

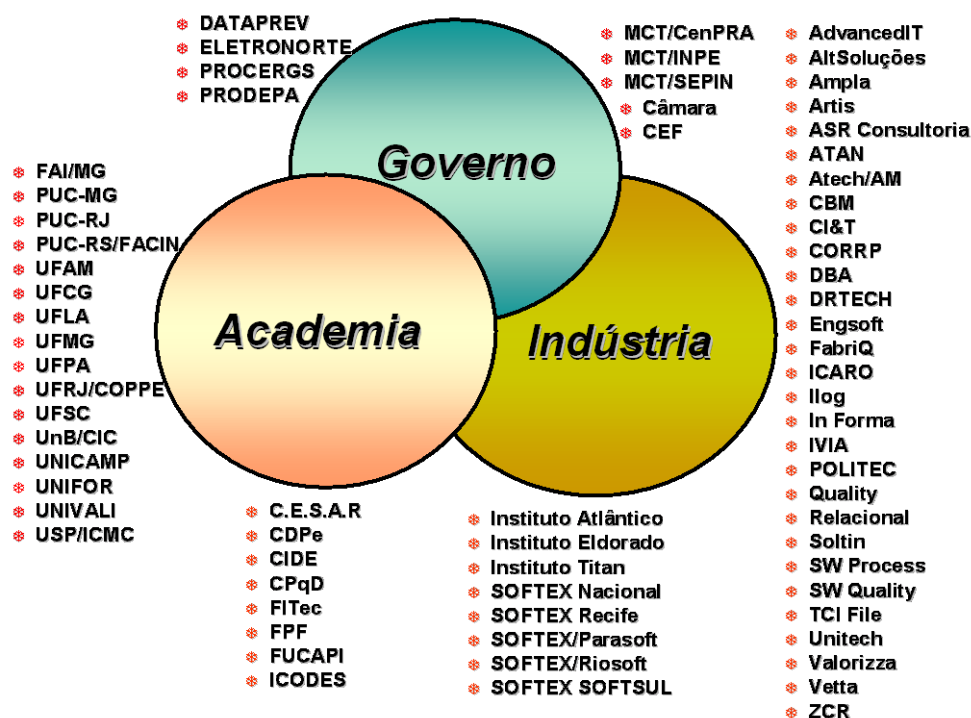
Ciclos mais Recentes do Programa - 2006

Projetos Selecionados do Ciclo 2006

Concebido como um programa mobilizador, a cada ano o chamado à participação vem despertando o interesse de entidades que submetem seus projetos, com a abertura de um espaço de discussão composto por especialistas de todo o País, interessados na melhoria da qualidade do software brasileiro.

Comparando-se esta distribuição com a das instituições que submeteram projetos aos ciclos anteriores, pode-se observar a adesão ao Programa por parte de novas instituições, em especial, no segmento industrial privado e da academia sendo mantida a representatividade e instituições relacionadas ao governo.

A figura a seguir ilustra a participação das entidades coordenadoras de projetos submetidos que foram selecionados em 2006.



Durante o Workshop da Qualidade e Produtividade em Software - WQPS Brasília, o Grupo de Avaliadores do PBQP-Software selecionou 99 projetos dentre os que foram submetidos ao Ciclo 2006.

A maior concentração nesse Ciclo provém de projetos relativos a Métodos de Gestão (47,5%) e Tecnologia de Software (30%), com a participação das demais categorias, todas representadas, não chegando a alcançar 25%.

Dos 99 projetos selecionados 47,4% foram relatados como concluídos e seus coordenadores apresentaram artigos que serão apresentados neste capítulo.

Número de Projetos por Categoria - 2006

Categorias	Projetos selecionados		Projetos concluídos	
	Nº	% sobre Total	Nº	% sobre selecionados
Conscientização e Motivação	7	7,1	4	57,1
Métodos de Gestão	47	47,5	25	53,1
Recursos Humanos	5	5,1	2	40,0
Serviços Tecnológicos	9	9,1	4	44,4
Articulação Institucional	-	-	-	-
Tecnologia de Software	30	30,0	12	40,0
Marketing de Software	1	1,0	-	-
Total	99	100	47	47,4

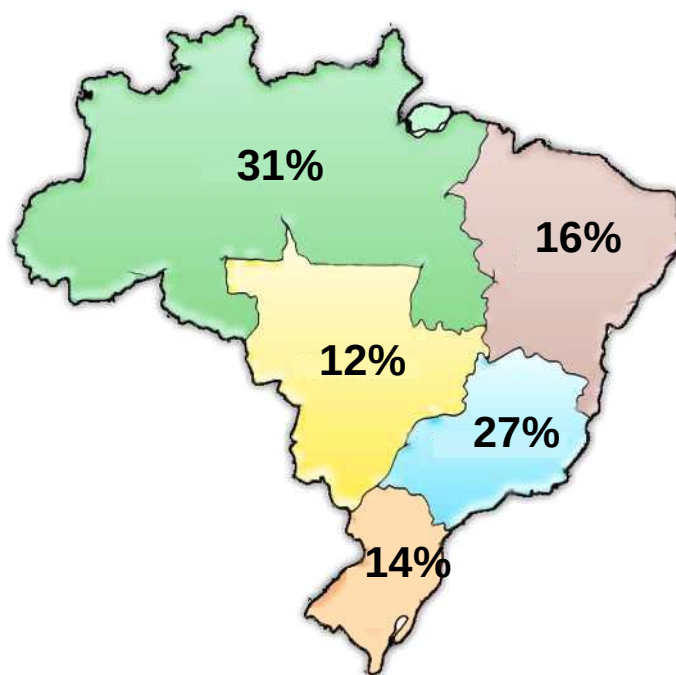
As regiões Norte e Sudeste continuam com o maior número de projetos submetidos, invertendo, no entanto, as posições alcançadas em 2005 e passando de 69 para 57, juntas.

As cidades de Manaus (26) e Campinas (12) permanecem na liderança, mantendo as posições do Ciclo anterior, enquanto Brasília (9) ocupa o terceiro lugar, ocupado em 2005 pela cidade do Rio de Janeiro.

Número de Projetos por Localização Geográfica - 2006

Região / UF	Nº	Municípios	UF	Nº
Norte	30	Belém	PA	4
Amazonas	26	Belo Horizonte	MG	3
Pará	4	Brasília	DF	9
Nordeste	16	Campinas	SP	12
Bahia	3	Caxias do Sul	RS	1
Ceará	7	Fortaleza	CE	7
Pernambuco	6	Florianópolis	SC	3
Sudeste	27	Formosa	GO	3
Minas Gerais	6	Ijuí	RS	2
Rio de Janeiro	6	Lavras	MG	2
São Paulo	15	Manaus	AM	26
Sul	14	Niterói	RJ	1
Santa Catarina	6	Porto Alegre	RS	5
Rio Grande do Sul	8	Recife	PE	6
Centro-oeste	12	Rio de Janeiro	RJ	5
Distrito Federal	9	Salvador	BA	3
Goiás	3	Santa Rita do Sapucaí	MG	1
Total	99	São Caetano do Sul	SP	1
		São Carlos	SP	1
		São José	SC	3
		São José dos Campos	SP	1
		Total	BR	99

Percentual de Projetos por Localização Geográfica - 2006



Número de Projetos por Entidade Coordenadora, segundo Categorias - 2006

Categorias	Entidades Coordenadoras	UF	Nº de Projetos
Conscientização e Motivação	CIDE	AM	1
	CPqD	SP	1
	FUCAPI	AM	1
	MCT/SEPIN	DF	2
	PUC-MG	MG	1
	SOFTEX/PARASOFT	PA	1
Métodos de Gestão	AdvancedIT	RS	1
	AMPLA	SP	1
	ASR Consultoria	SP	1
	ATECH	AM	1
	C.E.S.A.R	PE	1
	CBM	DF	1
	Câmara dos Deputados	DF	1
	CEF	DF	1
	CenPRA	SP	5
	CORRP	BA	1
	DATAPREV	RJ	1
	Eletronorte/PA	PA	1
	Eletronorte/Sede	DF	1
	FabriQ	AM	2
	FPF	AM	1
	FUCAPI	AM	4
	ICODES	GO	1
	In Forma	PE	1
	Instituto Atlântico	CE	3
	Instituto Titan	CE	1
	Instituto Eldorado	SP	1
	IVIA	CE	1
	POLITEC	DF	1
	PROCERGS	RS	1
	PRODEPA	PA	1
	Relacional	RJ	1
	SOFTEX Nacional	SP	2
	SOLTIN	AM	3
	SWQuality	MG	1
	TCI File	PE	1
	UFAM	AM	1
	UFLA	MG	1
	UFPA	PA	1
	UFRJ/COPPE	RJ	1
	UFSC	SC	1
	UnB/CIC	DF	1
	UNIFOR	CE	2
	UNIVALI	SC	1
	Valorizza	RS	1

Categorias	Entidades Coordenadoras	UF	Nº de Projetos
Recursos Humanos	Artis	RJ	1
	CIDE	AM	1
	PUC-RS/FACIN	RS	1
	SOLTIN	AM	1
	UFLA	MG	1
Serviços Tecnológicos	CenPRA	SP	1
	ILOG	SC	1
	PUC-RS/FACIN	RS	1
	SOFTEX Nacional	SP	2
	SOFTEX/Recife	PE	2
	SOFTEX/Riosoft	RJ	1
	SOFTEX/SOFTSUL	RS	1
	SW Process	RS	1
	UNIVALI	SC	2
Tecnologia de Software	ASR Consultoria	SP	1
	ATAN	MG	1
	ATECH	AM	1
	C.E.S.A.R	PE	1
	CDPe	RS	1
	CEF	DF	1
	CenPRA	SP	2
	CI&T	SP	2
	DBA	RJ	1
	DR TECH	AM	1
	Engsoft	RS	2
	FAI	MG	1
	FITec	MG	1
	FPF	AM	2
	FUCAPI	AM	2
	Icaro	SP	1
	ICODES	GO	2
	INPE	SP	1
	Instituto Atlântico	CE	1
	PUC-RJ/Inf	RJ	1
	PUC-RS/FACIN	RS	1
	QUALITY	AM	2
	SOFTEX Nacional	SP	1
	SOLTIN	AM	1
	UFCG	PB	1
	UFMG	MG	1
	UFPA	PA	1
	UFRJ/COPPE	RJ	1
	UFSC	SC	1
	UNICAMP	SP	1
	UNITECH	BA	1
	USP/ICMC	SP	1
	VETTA	MG	1
	ZCR	BA	1
Marketing de Software	Alt Soluções	AM	1

Artigos dos Projetos Concluídos do Ciclo 2006

Conforme previsto na etapa final de um Ciclo do PBQP-Software, quando da apresentação do Relatório Final do Ciclo 2006, os coordenadores de projetos concluídos foram responsáveis pela apresentação de um artigo relatando o desenvolvimento, descrição do produto e resultados relevantes alcançados. Os artigos encontram-se agrupados a seguir por categoria e ordenados segundo o número atribuído ao projeto.

Conscientização e Motivação

[1.03] Qualidade - é com isto que se desenvolve!

Entidade: 1Núcleo de Tecnologia de Software do Pará (PARASOFT).
Trav. Quintino Bocaiúva, 1588, 3º andar, Bloco A- 66035190- Belém - Pará – Brasil.

Autor: Dilma Farias Viana - dil_helena@yahoo.com.br

Abstract. *The awareness of importance about software processes improvement did not evolve equally in the regions of Brazil. In the state of Pará, it has happened gradually and there is a necessity of initiatives that motivate the local community in such way that it compromise to the culture of quality, in order to remain in this market each time more demanding and competitive. This article shows how actions have been delineated and executed so that academy, government and other organizations have knowledge about what is being produced in the area of quality and productivity of software, and how each one can be inserted in this context, so that they can take the benefits as much as other regions.*

Resumo. *A conscientização da importância da melhoria de processos de software não evoluiu igualmente em todo o Brasil. No estado do Pará, tem ocorrido de forma gradativa e há a necessidade de iniciativas que motivem a comunidade local a se comprometer com a cultura da qualidade, a fim de se manterem neste mercado cada vez mais exigente e competitivo. Este artigo mostra como foram delineadas e executadas ações que levassem ao conhecimento da academia, governo e demais organizações o que está sendo produzido na área de qualidade e produtividade de software, e como cada um pode ser inserido neste contexto, de forma a se beneficiar tanto quanto as demais regiões.*

1. Introdução

Apesar de não haver dados coletados a respeito, há uma grande carência de serviços de software que atendam às necessidades do mercado no estado do Pará. Sensível a este contexto, o Núcleo de Tecnologia de Software do Pará (PARASOFT), busca promover o desenvolvimento científico, tecnológico e mercadológico da indústria de software local, contribuindo assim para o seu desenvolvimento econômico e social.

Desta forma, e alinhado aos objetivos da sociedade SOFTEX para que o software brasileiro alcance padrões internacionais de qualidade e produtividade, o PARASOFT deu início a uma série de ações de fomento ao mercado local, que

culminaram em 2006 com reconhecida repercussão na academia, empresas e governo, sensibilizando-os e motivando-os para a realidade de que “sem qualidade não há competitividade”.

2. Objetivos e Justificativas

Este trabalho teve como objetivo principal a disseminação da cultura da melhoria da qualidade e produtividade de software no estado do Pará, buscando uma conscientização e motivação generalizada, pois o cenário era de pouco conhecimento e envolvimento das academias e empresas públicas e privadas do Estado, em relação aos programas, projetos e ações que vem acontecendo em âmbito nacional, capitaneados, principalmente, pela sociedade SOFTEX.

As organizações fornecedoras e compradoras de produtos e serviços de software tendem a se preocupar com o produto final, sem atentar para o fato de que a qualidade do produto está intimamente ligada à qualidade do processo de desenvolvimento empregado na sua implementação. Hoje se sabe que ter um processo de software forte e otimizado resulta em muitos benefícios mensuráveis como: maior qualidade, menor custo, menos tempo de desenvolvimento e maior satisfação do cliente (Wagner, Viana & Monteiro, 2007). Uma das ações para que os agentes SOFTEX abram novas perspectivas de solução, diante de problemas tão comuns enfrentados pelas organizações ao desenvolver ou adquirir softwares e serviços correlatos, e que tem se mostrado bastante efetiva, é a disseminação dos programas, projetos e ações relativas à qualidade e produtividade de software para que, principalmente, as micro, pequenas e médias empresas passem a ter maior conhecimento e acesso às oportunidades existentes.

Como produto esperado do trabalho realizado ao longo de 2006, esperava-se a realização de um curso de Introdução ao MPS-Br e a inclusão de temas voltados à qualidade e produtividade de software no Congresso Paraense de Software, que é um evento realizado pelo PARASOFT e seus parceiros.

3. Metodologia de Execução

A idéia do trabalho foi baseada nas necessidades de mercado, identificadas através de reuniões em que a autora participava, quer seja como Colaboradora do PARASOFT ou como Diretora-adjunta da SUCESU Regional Pará. Também havia a experiência de ter passado por uma empresa privada de grande porte e também de uma empresa pública, onde, em ambas, passou pelos mesmos problemas relacionados à falta de utilização de práticas adequadas de gestão no desenvolvimento e aquisição de software e serviços correlatos.

Através do *network* estabelecido pela participação em eventos e cursos fora do Estado (EQPSs, SBQSs, curso de MPS-BR, cursos de Engenharia e Processos de Software do Mestrado da UFPa) e de pesquisas em sites específicos, buscou-se delinear possíveis palestras com os assuntos de interesse no âmbito da qualidade de software.

Ao ter sido definida a data de realização do II Congresso Paraense de Software, de 24 a 25 de outubro de 2006, como um evento satélite da SEPAI (Semana Paraense de Informática), que configurava uma maior oportunidade de divulgação,

foram feitos os contatos previamente selecionados, buscando o agendamento de possíveis palestrantes e ministrantes dos cursos.

A parceria existente entre o PARASOFT, a SUCESU Regional Pará, e a PRODEPA (Empresa de Processamento de Dados do Estado do Pará) oportunizou o intercâmbio de palestrantes e ministrantes de cursos, fazendo com que diferentes públicos tivessem acesso a uma extraordinária gama de novas informações.

Durante o evento, o curso de Introdução ao MPS-Br foi realizado em forma de mini-curso. A razão é que aquele evento é que deveria sensibilizar muitas pessoas e organizações a participarem de um curso oficial, então optou-se em somente apoiar um mini-curso de Introdução ao MPS-Br proposto pela comissão organizadora da SEPAI.

Paralelamente a organização do II Congresso Paraense de Software e devido a contatos estabelecidos no SBQS 2006, solidificou-se a possibilidade de realização, pelo Ministério da Ciência e Tecnologia, de um EQPS (Encontro de Qualidade e Produtividade de Software) em Belém, com o apoio do PARASOFT e do Amazonsoft (Pólo de Software do Amazonas). Assim as ações foram evoluindo de forma que conseguíssemos os patrocínios necessários para concretizar o evento, no período de 23 a 24 de novembro. A divulgação do evento consistiu de comunicados e cartas específicas emitidas, dependendo do público-alvo, a fim de sensibilizarmos da necessidade de participação pelos interesses de cada um. Outra ação efetiva para que o I EQPS Belém tivesse um público interessante, foi a divulgação deste evento durante a SEPAI, ao final de palestras, painéis e mini-cursos.

4. Resultados Obtidos

Recursos humanos capacitados

Até fevereiro/2007, no estado do Pará, havia dez profissionais que realizaram as provas P-1 do MPS-Br e passaram. A autora é uma delas.

Devido a motivação do PARASOFT, e por considerar a área de qualidade promissora, a autora tem participado do curso de Mestrado em Ciência da Computação da Universidade Federal do Pará, com ênfase em engenharia de software e processos, visando especialização futura em Melhoria de Processo de Software.

Eventuais parcerias ou programas de transferência de tecnologia efetuados

A SECTAM (Secretaria de Tecnologia e Meio Ambiente), que faz parte do Conselho Administrativo do PARASOFT, porém sem uma representatividade efetiva, após ter participado do EQPS, buscou o fortalecimento da parceria através do convite para uma reunião de discussão sobre o Pólo de Software do Estado do Pará. A reunião foi realizada em dezembro/2006, resultando em uma possibilidade de realização de um projeto em parceria com o PARASOFT, visando um Arranjo Produtivo Local de Software para o Estado.

A academia também acabou por estreitar os laços de associado ao PARASOFT, pois teve a oportunidade de participar ativamente da organização do EQPS Belém, inclusive sediando este evento e participando através da apresentação de trabalhos em parcerias com empresas.

Outros resultados

Os resultados tangíveis deste trabalho podem ser resumidos na tabela a seguir:

Eventos realizados/ apoiados 4

Cursos 1

Empresas Aderentes ao Modelo MPS-Br 1

Provas realizadas (P-1) 2

Painéis 2

Palestras realizadas/apoiadas (II Congresso Paraense de Software) 8

Palestras realizadas/apoiadas (I EQPS Belém) 7

A seguir estão discriminadas algumas das realizações que a autora teve um papel decisivo para a realização ou que considera de grande relevância no cumprimento dos objetivos deste projeto.

Eventos

- Realização do II Congresso Paraense de Software
- Apoio à realização do I EQPS Belém, em parceria com outras organizações.

Cursos de Capacitação

- Mini-curso: "MPS.BR - Processo de Software", ministrado pela Profa. Dra. Ana Regina Rocha da COPPE/UFRJ, durante o III EIN/XX SEPAI.

Palestras

- "PMDS Programa de Modernização de Desenvolvimento de Sistemas da PRODEPA"-Edson Brabo e Dilma Viana – PRODEPA – SEPAI/COPIP/II CPS – 2006.
- "Dificuldades em Implantações de Processos de Software - Uma visão do cenário nacional" - Dilma Viana, Ana Paula e Renata Wariss – PRODEPA – SEPAI/COPIP/II CPS – 2006.
- "Experiências em implantação de processos de software visando certificações CMMI/MPS" - Coordenação: Dilma Viana (PRODEPA) – SEPAI/COPIP/II CPS – 2006.

Divulgações

A autora participou da elaboração e apresentação do artigo "Estudo das dificuldades em implantações de processos de software visando a implantação do Nível G do MPS.BR na PRODEPA" durante o Curso de Mestrado em Ciência da Computação da Universidade Federal do Pará (UFPA).

A autora apresentou o Trabalho: "Qualidade - é com isto que se desenvolve!" no EQPS Belém.

Além destes resultados previstos, com a participação no Curso de Formação de Agentes de Política Industrial, durante o evento “Estratégias Regionais de Inovação e Política Industrial”, realizado em Belém pela primeira vez em 2006, a autora procurou sensibilizar as entidades representadas (academia, empresas, governo), quanto a realidade de nosso Estado estar à parte das iniciativas de MPS já bastante evoluídas em outros estados.

As discussões geradas culminaram com a abertura de um canal de comunicação junto ao Programa de Desenvolvimento de Fornecedores do Estado do Pará (PDF), que é capitaneado pela Secretaria de Produção e Executiva de Indústria e Comércio SEICOM, e que tem como objetivo capacitar e promover as empresas paraenses como fornecedores dos grandes empreendimentos instalados no estado.

5. Aplicabilidade dos Resultados

Notadamente, no ano de 2006, o tema Qualidade e Produtividade de Software foi de grande destaque no maior evento de TI da região norte, a SEPAI/II Congresso Paraense de Software. Além deste, outro evento de grande repercussão foi o EQPS, que trouxe pela primeira vez ao Estado a apresentação de trabalhos que tratam especificamente do tema, além de propiciar palestras de grande importância para o empresariado local quanto aos programas e incentivos disponibilizados pelo governo federal, com fins de melhoria da qualidade e produtividade do software brasileiro.

Houve grande intercâmbio de informações durante as palestras dos eventos, os trabalhos apresentados, e as reuniões decorrentes. A interação do PARASOFT, da academia e do governo foi fortalecida e projetos futuros visando à melhoria de processos têm sido delineados. O modelo MPS-Br passou a ser considerado para a melhoria dos processos em organizações interessadas e já se enxerga a certificação como possível a um custo acessível.

A repercussão esperada foi alcançada em nível regional, já que durante os principais eventos, houve a participação representantes de outros estados da região Norte, como Manaus, Amazonas, Amapá, etc.

6. Características Inovadoras

Considerando, no estado do Pará, o cenário anterior de pouco conhecimento, quanto aos programas, projetos e ações relativas à qualidade e produtividade de software, o projeto trouxe ao conhecimento de muitos estudantes, empresários e governantes a importância da adesão ao movimento de qualidade e o quanto este pode melhorar a situação do mercado local de software, a formação de profissionais e a competitividade das empresas.

7. Conclusão e Perspectivas Futuras

A disseminação em massa do tema qualidade e produtividade nos eventos promovidos pelo PARASOFT, trouxe uma conscientização mais abrangente e maior motivação aos que já estavam convencidos desta necessidade. A academia tende a assumir maior compromisso com o tema e a apoiar as iniciativas relacionadas. Empresas públicas e privadas já buscam informações e demonstram

interesse em implantar a melhoria de processos de software (MPS). No entanto é necessário um trabalho que venha a reforçar esta necessidade de forma a efetivamente convencer, principalmente o empresariado local, dos benefícios esperados ao se implantar a MPS.

Para que as iniciativas de tecnologia da informação e comunicação sejam evolutivas em nosso Estado, é precípua que o governo venha a definir políticas de investimento nesta área, inclusive visando o alinhamento ao Governo Federal. Tal fato dificulta sobremaneira as ações dos voluntários que administram e executam os projetos da entidade PARASOFT e inviabiliza a manutenção deste como agente SOFTEX. Apesar das dificuldades encontradas, o trabalho de conscientização e motivação teve excelente repercussão e certamente foi um divisor de cenários no Estado. Pode-se vislumbrar a partir deste ano de 2007, a intenção do governo local em definir uma política de TIC e um arranjo produtivo local de software, em que o PARASOFT deve assumir um papel mais representativo, agregador e orientador, a fim de que se alavanque o mercado local de software.

8. Referências Bibliográficas

PARASOFT (2005). Núcleo de Tecnologia de Software do Pará. Ata de Reunião - 28 de junho de 2005.

Wagner, A. P. K., Viana, D. F. e Monteiro, R. W. (2007). Resumo do Artigo "Managing Organizational Change for Software Process Improvement". Processamento de Dados do Estado do Pará (PRODEPA) / Universidade Federal do Pará (UFPA).

Project Management Institute. (2000). A guide to the project management body of knowledge (PMBOK guide).

[1.05] Revitalização do SPIN-Manaus.

Entidade: Centro de Incubação e Desenvolvimento Empresarial - CIDE

Autor: Paulino Wagner Palheta Viana - wagner_palheta@yahoo.com.br

Objetivos e Justificativas:

Revitalização e estruturação do SPIN-Manaus (Software Process Improvement Network), criando um fórum para intercâmbio de experiências voltadas a Engenharia de Software e Qualidade de Software local.

Entre dez/2002 a dez/2004, o SPIN-Manaus atuou na disseminação de práticas de engenharia de software e modelos de qualidade do processo de software num mercado carente por informação e capacitação na área. Durante 2005, o SPIN-Manaus ficou em estado de hibernação, mas hoje a maioria das organizações de software na região possui uma iniciativa interna de promover a melhoria dos seus processos de software e re-surge a demanda por um fórum para intercâmbio de informações interinstitucional.

O projeto se propôs a revitalizar o SPIN-Manaus, através da articulação junto a profissionais envolvidos com os programas de melhoria de processo de software em suas instituições para determinar uma nova forma de atuação do SPIN que busque atender as demandas atuais da região.

O SPIN-Manaus, foi lançado na I Mostra de Educação Tecnológica FUCAPI, realizado nos dias 17,17 e 18 de novembro de 2006, infelizmente não pode contar com um grande numero de pessoas, mas as que estiveram presentes foram bastante colaborativas nos aspectos do propósito. Foi sugerido inclusive a mudança do nome para SPIN-Amazonas e que foi acatado imediatamente.

Resultados Relevantes:

Fóruns interinstitucionais e regionais para intercâmbio de experiências têm-se mostrado em todo o mundo um componente essencial na capacitação de recursos humanos e no desenvolvimento da maturidade das comunidades locais em engenharia de software. Diversa organização tem buscado promover o setor de software em Manaus e o profissional da área tem muito a se beneficiar com a estruturação de um fórum nos moldes propostos

Impacto:

O SPIN-Manaus pretende ser um ator relevante na mobilização e articulação do setor de software local visando não somente contribuir para o crescimento do setor na região, mas promover o uso e disseminação de tecnologias de Engenharia de Software, contribuindo para a competitividade das organizações no cenário nacional e na maturidade dos profissionais locais.

Abrangência:

O SPIN-Manaus tem abrangência multi-institucional local e seus participantes são voluntários. Por Manaus constituir um centro tecnológico regional, a esfera de influência potencial pode ser a região da Amazônia Ocidental como um todo.

Inovação:

Existem múltiplos SPINs no Brasil e no Mundo. Em geral, nos locais onde há tradição ou pólo do setor de software / engenharia de software. O SPIN-Manaus é o único SPIN que se tem notícias na Região Norte.

Conclusão:

Conseguimos atingir o objetivo, agora cabe aos colaboradores não deixar o movimento esfriar novamente.

Referências Bibliográficas:

SPIN-RIO
SPIN-BH
SPIN-Ceará
SPIN-Curitiba
SPIN-Campinas

[1.06] Seminário de Qualidade do Grupo de Pesquisa em Engenharia de Software GPES-CESF.

Entidade - Instituto de Ensino Superior FUCAPI - CESF

Autor: Paulino Wagner Palheta Viana - wagner_palheta@yahoo.com.br

Objetivos e Justificativas:

A criação do evento “Seminário de Qualidade do GPES-CESF” teve como principal objetivo, apresentar os trabalhos de pesquisas ou trabalhos finais de graduação realizados pelos alunos envolvidos com pesquisa no GPES-CESF.

O mercado hoje em dia necessita de profissionais qualificados e existe uma grande demanda no mercado de engenharia de software, o evento visa motivar os alunos de informática do CESF a pesquisar na área de Engenharia de Software e apresentar os resultados dos trabalhos no Seminário da Faculdade.

O evento foi realizado na I Mostra de Educação Tecnológica FUCAPI nos dias 16,17 e 18 de novembro de 2006 e contou com a participação de professores e dos alunos que tiveram seus trabalhos submetidos ao PBQP-SW do ciclo de 2006.

O Seminário teve o seguinte formato:

- Apresentação de abertura do GPES-CESF.
- O lançamento do SPIN-Amazonas. (Projeto 1.05).
- Apresentações dos projetos dos alunos inscritos no PBQP-SW no ciclo de 2006.
- Apresentação de trabalhos futuros do grupo.

Resultados Relevantes:

O Seminário serviu como uma vitrine dos trabalhos realizados dentro do GPES-CESF, como indicador tivemos três trabalhos realizados em Qualidade de Software pelo GPES-CESF.

Impacto:

Terá um impacto muito grande pois demonstrará como indicador o numero de trabalhos realizados pelo GPES-CESF

Abrangência:

O Seminário foi divulgado por diversas instituições de ensino e fundações de pesquisas em tecnologia na cidade de Manaus e alcançou uma boa visibilidade pelo próprio portal do GPES-CESF já renovado.

Inovação:

O evento foi inovador sendo o primeiro evento realizado pelo Grupo de Pesquisa de Engenharia de Software – GPES-CESF.

Conclusão:

Conseguimos atingir o objetivo, o Seminário foi realizado dentro do previsto, obteve participação dos alunos. Agora em 2007, o Seminário terá uma nova participação com os trabalhos desse ano.

Para nós professores, além da grande satisfação, fica a sensação do dever cumprido, na extensão do ensinamento fora de sala de aula.

[1.07] Projeto de tradução das PAs do nível 3 do CMMI

Autores: André Villas-Boas - villas@cpqd.com.br

José Marcos Gonçalves - jose@cpqd.com.br

1) Objetivos e Justificativa

O objetivo deste projeto é traduzir e disponibilizar as práticas específicas do nível 3 do CMMI.

Em continuação aos projetos de tradução do modelo CMM realizados no âmbito do PBQP nos anos de 2000 e 2001 e das traduções do CMMI realizadas no ano de 2005, este projeto tem o objetivo de disponibilizar uma tradução para o português das práticas específicas do nível 3, complementando o trabalho já realizado. Esse documento também será disponibilizado em formato eletrônico para toda a comunidade.

As traduções do modelo CMM foram muito utilizadas, seja no CPqD, em outras empresas e em várias instituições de ensino em cursos de pósgraduação.

Essa necessidade de se ter um texto em português foi percebida em várias oportunidades, como reuniões regulares do PBQP e outros eventos da área de qualidade de software.

2) Metodologia de execução

Este trabalho, assim como a tradução do CMM e da primeira parte do CMMI, foi realizado como atividade extra dos autores, feito em horário fora do trabalho. Foi feito um planejamento da tradução (uma divisão das tarefas) e a revisão executada foi cruzada, ou seja, um revisava o que o outro havia traduzido.

A experiência da outra tradução foi muito importante para o planejamento, bem como para a execução das tarefas de tradução e revisão.

Na média, cada Área de Processo levou perto de 16h de trabalho para o tradutor e 4h para o revisor.

3) Resultados relevantes

Os projetos do ciclo passado (2005) apresentaram a tradução da parte introdutória do CMMI (introdução do modelo, as duas representações, a característica de integração do modelo, que agora trata software, sistema, desenvolvimento integrado e SUPPLIER SOURCING, a estrutura os componentes do modelo,

comparação das duas representações), das áreas de processo de nível 2 do CMMI, que são:

- Gestão de requisitos;
- Planejamento de projeto;
- Controle e monitoramento de projeto;
- Gestão de fornecedores;
- Medição e análise;
- Gestão de configuração e
- Garantia da qualidade de produto e processo.

E das metas e práticas genéricas do CMMI, que habilitam a organização a institucionalizar as melhores práticas.

O projeto atual apresenta a tradução das 14 áreas de processo de nível 3 do CMMI, que são:

- Desenvolvimento de Requisitos;
- Solução Técnica;
- Integração de Produto;
- Verificação;
- Validação;
- Foco no Processo Organizacional;
- Definição do Processo Organizacional;
- Treinamento Organizacional;
- Gestão Integrada de Projeto;
- Gestão de Risco;
- Integração de Equipes;
- Gestão Integrada de Fornecedores;
- Análise de Decisão;
- Ambiente Organizacional para Integração.

4) Aplicabilidade

A idéia de se ter o material relativo ao CMMI disponível em português tem como base facilitar o acesso às informações do modelo aos membros da comunidade de qualidade de software que têm pouca familiaridade com a língua inglesa. Da mesma forma que as traduções anteriores do CMM/SW (resultados de projetos do PBQB mais antigos) essas traduções serão úteis na preparação de equipes para implantação do modelo, na preparação de empresas para avaliação e na disseminação do modelo em cursos de graduação e pós-graduação em instituições de ensino brasileiras, ou de países de fala portuguesa.

Outra utilização para esse material é como texto traduzido para o projeto do CMMI-Browser, que está sendo desenvolvido pela Univali sob coordenação da Profa. Dra. Christiane Gresse von Wangenheim.

Em várias edições das reuniões regulares do PBQP fomos informados da utilização do nosso material já traduzido do CMM/SW em vários cursos pelo Brasil, bem como em várias empresas que se preparavam para a avaliação oficial. A Gerência de Comunicação com o Mercado do CPqD já informou que o *link* da página institucional do CPqD, onde ficam os documentos da tradução do CMM/SW e do CMMI (Visão Geral e Práticas do Nível 2 - www.cpqd.com.br, *link* "Comunidade CpqD"), é o *link* mais visitado da página, além do fato de terem sido contactados por pesquisadores portugueses e por estudantes de países africanos de fala portuguesa.

5) Características inovadoras

A inovação desse projeto está no fato de gerar um documento sobre o CMMI em língua portuguesa e disponível na internet, ou seja, um material importante para a área de qualidade de software, mas na nossa língua e sem custos associados à aquisição.

6) Conclusões e Perspectivas Futuras

Dando continuidade ao trabalho e já preocupada em disponibilizar todo o material sobre o CMMI, a dupla pretende traduzir as áreas de processo dos níveis 4 e 5, e também já se prepara para elaborar a tradução da nova versão do CMMI (versão 1.2 lançada em setembro/2006), de tal forma que o público de fala portuguesa tenha o material completo do CMMI em português e disponível pela internet.

7) Referências Bibliográficas

[1] SEI, **Capability Maturity Model Integration** (CMMISM), Version 1.1, CMU/SEI-2002-TR-012, 2002.

Método de Gestão

[2.01] A implantação de um PMO (Escritório de Projetos) na Dataprev

Entidade: Empresa de Tecnologia e Inf. da Previdência Social - DATAPREV

Autor: Marcelo Portugal Fellows - marcelo.fellows@previdencia.gov.br

Objetivos e Justificativa

Em meados de 2004 o Departamento de Controladoria do Rio de Janeiro (DCRJ.N), órgão vinculado a Diretoria de Negócios da Dataprev (DNG), empresa de Tecnologia e Informações da Previdência Social, iniciou o processo de implantação de seu PMO - *Project Management Office*, Escritório de Projetos (EP).

A experiência é uma evolução de diversas outras tentativas anteriores e busca colher os benefícios desta metodologia, alcançando maior produtividade através de uma de organização orientada e apoiada pela gestão de projetos, mas também, e principalmente, na gestão "por projetos".

Em dezembro de 2005, a alta direção optou de vez pela gestão por projetos, criando o Departamento de Projetos de Software – DEPS com três subdivisões, duas voltadas à execução dos projetos e uma que seria o próprio PMO, a Divisão

de Gestão de Projetos de Softwares. Houve uma grande reestruturação na diretoria, e, em um movimento que já se ensaiava há um tempo, finalmente foram criadas as áreas de: relacionamento com cliente, qualidade e metodologia, fábrica de software, gestão de produtos, informações e projetos, este com seu respectivo Escritório.

O DEPS então, em sua gestão por projetos, patrocinado agora formalmente pela Diretoria de Relacionamento e Desenvolvimento de Sistemas - DRD – evolução da antiga DNG, e pelo seu principal cliente o INSS, impulsionado pelo PINMG – Projeto de Implantação do Novo Modelo de Gestão do INSS, passou a executar suas atribuições, acreditando que, entre outros benefícios haverá uma melhor otimização de seus recursos, tantos materiais como humanos, o que a delimitação de escopo acarretará em um esforço mais focado e desta forma em um trabalho mais eficiente, o que o estabelecimento de prazos para entrega de produtos vai gerar mais sinergia dentro das equipes demandadas, o que a reunião de vários colaboradores unidos pelo mesmo propósito aumentará a qualidade dos resultados, o que a adoção de melhores práticas recomendadas pelo mercado e evoluídas através de um processo de melhoria contínua, vai forçosamente contribuir para o oferecimento de melhores produtos para toda a previdência social especificamente e para toda sociedade brasileira consequentemente, o que a organização orientada a projetos vai aumentar o grau de eficácia dos processos e qualidade dos produtos e soluções oferecidas, o que vai gerar respostas imediatas às necessidades da organização e seu principal cliente, o vai permitir que o processo decisório por parte da alta administração, seja mais apoiado nas informações apresentadas pelo portfólio de projetos, organizado em programas, projetos e sub-projetos.

Metodologia de Execução

Como no momento desta nova iniciativa haviam diversos projetos em andamento, estabeleceu-se a premissa de que, somente os novos projetos, seriam acompanhados pelo E. P.

O modelo de EP escolhido foi do tipo Painel de Controle em que o escritório não assume a responsabilidade pelo sucesso dos projetos porém provê suporte na metodologia e ferramentas, atua na mediação de conflitos entre os departamentos, desenvolve os profissionais e a própria cultura de gerência de projetos (GP) na empresa, apóia diversos projetos simultaneamente, encarrega-se dos trâmites burocráticos de formalização do projeto, dá suporte ao diretor nas questões de priorização dos esforços, cuida do repositório de informações sobre projetos na empresa, elabora e oferece relatórios gerenciais à alta direção (nível executivo).

Atualmente face às demandas do PINMG, o DEPS focou inteiramente seu escopo na implementação do PINMG, que deverão propiciar uma notável melhoria na gestão da Previdência Social.

Resultados Relevantes

Abaixo relacionamos diversos resultados, destacando sobretudo, a melhoria da produtividade e o maior controle das ações da área de desenvolvimento de



sistemas da empresa, permitindo uma visão clara de onde são necessárias priorizações.

- Atualização do fluxo de processos para a formalização de projetos,
- Confeção de uma metodologia de GP, aderente ao PMBoK1 do PMI2 refletindo todos os processos, suas explicações e modelos de documentos,
- Esta cartilha está disponível na intranet da empresa contendo também
- textos teóricos sobre GP, e um extenso glossário
- Definição do modelo de *mentoring* e *coaching* que o DCRJ ofereceria,
- Definição das ferramentas de suporte aos projetos,
- Atualização do aplicativo de controle de portfólio de projetos da empresa,
- Obrigatoriedade da confecção da WBS que auxilia aos líderes de projeto e clientes a enxergarem melhor o trabalho a ser feito e principalmente os produtos a serem entregues, delimitando de forma inequívoca o escopo dos projetos,
- Formação de equipe de profissionais com formação em gerência de projetos,
- Check list para planejamento e acompanhamento de projetos,
- Realização de fóruns com os gerentes de projeto,
- Foi buscado patrocínio com os demais gerentes de departamento e divisão, para que se promovessem uma conscientização sobre os benefícios em se observar as melhores práticas em projetos,
- Definiu-se um modelo de acompanhamento de projetos onde os principais eventos de cada projeto são registrados em uma área específica de conhecimento - integração, escopo, tempo, custos, comunicação, RH, riscos, qualidade e aquisições.
- Estes registros são lançados em um outro aplicativo construído pelo DCRJ de onde é possível extrair diversos relatórios,
- Ao encerramento de cada projeto são colhidas as Lições Aprendidas de cada líder de projeto, promovendo assim um processo de melhoria contínua,
- Relatórios consolidados apresentando a posição dos portfólios de projetos são encaminhados mensalmente ao diretor de negócios, de forma que este possa ter uma visão panorâmica das ações de sua diretoria,
- Foi criada na home-page desta diretoria uma página sobre os projetos com acesso aos diversos endereços referentes a GP,
- Auxilia na obtenção da certificação CMMI no nível II e III,
- Fica disponível aos líderes um formulário eletrônico para acompanhamento dos projetos no serviço de correio, que está sendo utilizado inclusive para o líder manter sua equipe e patrocinador informando sobre o projeto,
- Foi preenchido um formulário avaliando-se a maturidade do escritório seguindo o método OPM3 do PMI, em nível 2,

- Produção de uma vasta gama de indicadores de desempenho que auxiliam no diagnóstico de problemas e oferecem os melhores caminhos para as suas soluções,
- Compilação de recomendações extraída do registro do acompanhamento de projetos e dos relatórios de lições aprendidas,
- Estamos encaminhado para aprovação da mesa diretoria de alteração da norma par que os pagamentos das gratificações sejam estendidos à toda equipe, obviamente de forma proporcional a sua participação e principalmente que o pagamento desta seja efetivado somente mediante a entrega dos produtos do projetos,
- Controle de custos por projeto, uma necessidade antiga que aos poucos vai sendo implantada, e, vai permitir que se tenha uma exata noção de onde devem ocorrer os novos investimentos,
- Foi desenvolvido um trabalho de conclusão do curso de MBA da Fundação Getúlio Vargas com este mesmo título.

Aplicabilidade dos resultados e principais impactos

Como toda mudança de cultura que acontece na Dataprev, ocorreu uma grande resistência a esta nova forma de trabalho. O DCRJ não tinha alçada suficiente para efetuar uma alteração na estrutura funcional da empresa, porém com os próprios resultados dos projetos a aceitação geral foi cada vez maior, resultando na própria adoção do modelo orientado à projetos desde então preconizado posteriormente reafirmado pelo seu bem sucedido sucessor, o DEPS.

A própria facilidade do diretor em ter em mãos um painel de controle refletindo o estado em tempo real de cada iniciativa foi um grande impulsionador do processo.

Características Inovadoras

Aos poucos a administração pública vem se dando conta dos benefícios da GP.

Algumas outras empresas já estão também em processo de implantação de seus PMO's porém na área de T. I. temos colhido o ônus e os bônus deste pioneirismo.

Não faz mais de uma década que o conceito de escritório de projetos passou a ser adotado pela empresas de tecnologia. Migrado da área de engenharia os conceitos de GP, revelarem-se bastante próprios a nossa área uma vez que tratam também de construção de um produto ou serviço novo, que para obtenção de maior qualidade devem passar por fases de concepção, planejamento, execução, controle e fechamento de forma estruturada.

Desta forma, as iniciativas dentro da gestão pública, podem ser agrupados de acordo com seu objetivo, abrangência ou cliente, em portfólios e programas de projetos, podendo ser acompanhadas dentro de uma metodologia focada nos resultados planejados nas linhas de bases, seja na área de tempo, custo, recurso, riscos, qualidade e principalmente, escopo.

Conclusão e Perspectivas Futuras

Conforme preconiza a literatura especializada na área, a gestão por projetos é a ferramenta mais eficiente de atingir na prática, os objetivos levantados em seu planejamento estratégico. O trabalho tem sido bastante árduo principalmente no que se refere a resistências de setores mais arraigados a matriz funcional tradicional. O esforço de aculturação é longo, porém tem sido exercido com bastante motivação pela equipe do DEPS, uma vez que a crença no sucesso da iniciativa e conseqüentemente no futuro da empresa, é compartilhado por todos.

A prática de se fechar o escopo com o cliente, embora desconfortável a princípio, propiciou muitos ganhos em eficiência pois está conseguindo eliminar as tradicionais, porém antiprodutivas, solicitações de novas funcionalidades. Temos frisado bastante no processo de consultoria aos líderes, que um minuto gasto em planejamento vai poupa-lo de outros três minutos de retrabalho na fase de implantação.

A maioria dos líderes nas reuniões de homologação e fechamento testemunhou que, apesar de sua rejeição inicial a um trabalho que lhes parecia meramente burocrático, a gerência de projetos por ocupar-se de forma tão minuciosa do planejamento de um sistema, mostrou-se bastante valiosa no momento da implantação homologação de novos produtos. A utilização dos mapas de riscos e plano de comunicações foi citada como uma forma de atuação que lhes preveniu de diversos acidentes de percurso.

Abaixo uma relação das iniciativas futuras ou em andamento:

- Implantação dos produtos previstos projetos do PINMG do INSS,
- Integração com EP do MPS,
- Evolução de ferramentas de apoio,
- Formalização da atividade na DTP,
- Revisão das normas internas sobre gestão de projetos,
- Integração total com a área de metodologia da Dataprev,
- Controle e qualificação dos artefatos de Software,
- Aperfeiçoamento de métodos de estimação de prazos (Banco de Métricas),
- Atualização do fluxo de processos e da metodologia com modelos próprios e documentos mínimos de acordo com o porte e tipo do projeto - se sistema novo, manutenção, versionamento ou *datawarehouse*,
- Realização de workshops para divulgação das melhores práticas em GP,
- Realização de capacitação em GP para líderes em fase de concepção de projeto,
- Realização de treinamento de 24hr com fundamentos da GP em parceria com consultoria de renome no mercado,
- Definição de uma política de alocação de mão de obra dos escritórios regionais e mesmo de demais áreas da empresa em forma de um banco de talentos

Referências Bibliográficas

- Choosing the Right PMO setup. CASEY, William e PECK, Wendi. PM Network Fev/2001.
- Gerência de Projetos. Kim Heldman, 2003. Editora Campus
- Gerenciamento de Projetos - Definindo e controlando o escopo em Projetos, Carlos Magno da Silva Xavier, 2005. Editora Saraiva.
- Gerenciamento de Projetos. Paul Campbell Dinsmore, Editora Qualitymark, 2003.
- PMP Exam Prep. Rita Mulcachy, RMC Publications, 1998.
- Gerenciamento de Projetos nas Organizações. Darcy Prado, Ed. DG, 2003.
- Os 7 Hábitos das Pessoas Altamente Eficazes. Stephen R. Covey, Editora Best Seller, 1989

[2.04] A Implantação do CMMI-SW Nível 3 no Instituto Atlântico

Entidade: Instituto Atlântico - Rua Chico Lemos, 946 - Caixa Postal 60822-780, Fortaleza - CE - Brasil

Autores: Fabiana G. Marinho, Tatiana C. Monteiro, Solange A. Alcântara, Márcia G. Sampaio, Carlo Giovano S. Pires - {fabiana, tatiana, solange, marcia, cgiovano}@atlantico.com.br

Abstract. *This paper describes the Instituto Atlântico experience in planning, defining and implementing CMMI-SW level 3. This experience was based on IDEAL's process improvement cycle and carried out according to the following phases: planning, diagnosis of the existing process, definition of new processes, implementation and use of the defined processes and assessments. The observed results of this experience have been the increase of the organizational development processes.*

Resumo. *Este trabalho relata a experiência de planejamento, definição e implantação do CMMI-SW nível 3 no Instituto Atlântico. Esta experiência foi planejada com base no modelo de melhoria de processos IDEAL e realizada de acordo com as seguintes etapas: planejamento das atividades, diagnóstico dos processos existentes, definição dos novos processos, implantação dos processos definidos e avaliações. Como principais resultados obtidos, ressaltamos o amadurecimento do processo de desenvolvimento da instituição.*

1. Introdução

O Instituto Atlântico é uma instituição de pesquisa e desenvolvimento localizada em Fortaleza, Ceará, fundada em novembro de 2001 por iniciativa do Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD).

Em fevereiro de 2006, o Instituto Atlântico atingiu o nível 3 de maturidade no modelo CMMI-SW versão 1.1, abrangendo todos os processos da área de



desenvolvimento. O Instituto Atlântico também é certificado na norma ISO 9001:2000 desde julho de 2005. Este trabalho relata a experiência de planejamento, definição e implantação do CMMI-SW nível 3 no Instituto Atlântico. Esta experiência foi planejada com base no modelo de melhoria de processos IDEAL e apresentou como principais resultados obtidos o amadurecimento do processo de desenvolvimento da instituição.

2. Objetivos e Justificativa

Este artigo está estruturado em sete seções. A seção 2 descreve os objetivos e justificativas deste trabalho. Na seção 3, a metodologia de execução do processo de implantação do CMMI nível 3, bem como as estratégias adotadas para a condução deste processo são apresentadas. As seções 4 e 5 discorrem sobre os resultados obtidos. A seção 6 destaca as características inovadoras do projeto. Finalmente, a seção 7 apresenta as conclusões e perspectivas futuras.

Este artigo tem como objetivo contribuir com o relato da experiência do Instituto Atlântico nas atividades de planejamento, definição e institucionalização do programa de melhoria de processos alinhado ao nível 3 do modelo CMMI. O CMMI (*Capability Maturity Model Integration*) [Chrissis, Konrad and Shrum 2004] é um modelo desenvolvido pelo SEI que visa orientar as organizações de software na implementação de melhorias contínuas em seu processo de desenvolvimento. O nível 3 de maturidade do CMMI reforça o controle da gestão nas organizações e expande o foco para as áreas de engenharia de software. O nível 3 enfoca também o nível de processos organizacionais, gestão do conhecimento, treinamento, tomada de decisão formal e institucionaliza a filosofia de melhoria contínua.

3. Metodologia de Execução

O método de pesquisa adotado neste trabalho foi o qualitativo. O uso de métodos qualitativos é justificado por envolver o estudo do processo de planejamento e implantação do CMMI-SW no seu contexto real, visando documentar e analisar os resultados de uma experiência prática. Para a coleta de dados, adotou-se uma solução baseada em entrevistas e análises de documentos.

O grupo responsável pela pesquisa participou de todas as atividades e eventos da implantação do CMMI-SW nível 3, atuando no *Engineering Process Group* (EPG) do Instituto Atlântico e vivenciando na prática todo o processo.

3.1. O processo de implantação do CMMI-SW nível 3

A implantação do CMMI-SW nível 3 foi planejada para ocorrer de forma gradual, visando reduzir o impacto das mudanças necessárias nas práticas institucionais e respeitando sua cultura organizacional. As atividades foram realizadas em cinco etapas principais, seguindo a estrutura do modelo de melhoria organizacional IDEAL: (i) planejamento das atividades; (ii) diagnóstico dos processos existentes; (iii) definição dos novos processos; (iv) implantação e uso dos processos definidos e (v) avaliações. Cada uma dessas etapas está descrita a seguir.

Planejamento das atividades

O planejamento e a gerência das atividades de melhoria dos processos do Instituto Atlântico foram de responsabilidade do grupo de Suporte ao Processo de Desenvolvimento (SPD), que contou com o suporte da consultoria de uma empresa externa certificada pelo SEI especializada em CMMI. O SPD é um grupo institucional com atuação nas áreas de qualidade, testes, arquitetura, usabilidade, processos, medição e análise, treinamento e segurança. Os subgrupos de qualidade (QAs), testes, usabilidade e arquitetura são compostos por colaboradores com alocação dedicada ao SPD. Já os subgrupos de processos (EPG), medição e análise, treinamento e segurança são compostos por integrantes do SPD e por integrantes das equipes dos projetos.

O cronograma contendo as principais atividades realizadas é mostrado na Tabela 1. O programa de melhoria passou por um planejamento inicial no início do ano de 2004. No início do ano de 2005, um novo planejamento foi necessário, de modo a adequar as atividades do programa de melhoria às novas necessidades estratégicas da instituição. Destaca-se a inserção da norma ISO 9001:2000 na arquitetura de processos existente.

Tabela 1: Cronograma de atividades

ETAPA / ATIVIDADE	MÊS	ANO
Planejamento		
Plano de Melhoria	Março	2004
Definição dos novos processos		
Identificação e priorização dos processos da organização	Março	2004
Criação dos <i>Process Action Teams</i> - PATs	Março	2004
Treinamento CMMI Intermediário	Abril	2004
Treinamento em Técnicas e Métodos de Definição de Processos	Abril	2004
Revisão e atualização do Plano de Melhoria	Janeiro	2005
Treinamentos V&V e M&A	Março e Maio	2005
Definição dos processos da organização	Maio a Julho	2005
Implantação e uso dos processos definidos		
Início dos treinamentos nos novos processos	Junho	2005
Institucionalização - projetos piloto	Junho a Agosto	2005
Institucionalização definitiva	Setembro a Novembro	2005
Avaliações		
Avaliação Classe C	Agosto	2005
Treinamento SCAMPI	Setembro	2005
Avaliação independente de SQA (CPqD)	Outubro	2005
Avaliação Classe B	Novembro	2005
Avaliação Interna	Janeiro	2006
Avaliação Classe A	Fevereiro	2006

Diagnóstico dos processos existentes

A primeira atividade a ser realizada pelo EPG foi analisar os procedimentos definidos e institucionalizados para o SW-CMM nível 2. Nesta época, existiam apenas seis procedimentos organizacionais que documentavam as práticas de gestão da instituição. Estes procedimentos foram analisados sob dois aspectos: (i) mapeamento para as práticas de gestão do modelo CMMI-SW e (ii) adequação ao novo contexto do Atlântico. O resultado obtido com o mapeamento inicial realizado é apresentado na Tabela 2.

Tabela 2: Diagnóstico dos procedimentos existentes

Procedimento SW-CMM	Processo CMMI-SW
Procedimento Institucional para Planejamento de Projetos	Processo de Gestão de Projeto
Procedimento Institucional para Estimativas	Processo de Gestão de Projeto
Procedimento Institucional para Acompanhamento de Projetos	Processo de Gestão de Projeto
Procedimento Institucional para Garantia da Qualidade	Processo de Garantia da Qualidade
Procedimento Institucional para Gestão de Configuração	Processo de Gestão de Configuração
Procedimento Institucional para Gestão de Requisitos	Processo de Requisitos

Definição dos novos processos

A arquitetura de processos do Instituto Atlântico foi definida e documentada com base na arquitetura de processos do RUP. O RUP foi escolhido por fornecer uma notação formal para a definição de processos baseada na UML [Pires, Marinho and Souza 2004].

Inicialmente, o EPG foi constituído por trinta e um colaboradores organizados em subgrupos denominados *Process Action Teams* (PAT's). Essa estratégia foi adotada com o objetivo de envolver integrantes das equipes de projetos com dedicação parcial como forma de obter contribuições diferenciadas para as diversas áreas de processo. Após a definição inicial, os representantes de cada PAT realizaram *workshops* para apresentar os processos aos principais envolvidos com a execução dos mesmos. Os participantes dos *workshops* fizeram sugestões de melhoria e adequação dos processos de acordo como as práticas e necessidades dos projetos. Após a incorporação das correções sugeridas, os processos ficaram mais aderentes à realidade da organização e os envolvidos mostraram-se mais engajados com a implementação dos mesmos.

Implantação e uso dos processos

O processo de implantação dos processos alinhados ao CMMI-SW nível 3 no Atlântico pode ser descrito em três fases. A primeira fase caracterizou-se pela realização dos treinamentos nos processos organizacionais. Durante esta fase foi criado o Grupo de Treinamento, responsável por dar suporte ao Programa de Treinamento.

A segunda fase iniciou com a seleção dos projetos pilotos. Foram selecionados três projetos para a análise da implantação dos processos definidos. Os projetos

selecionados representavam uma amostra significativa dos projetos existentes na instituição com relação ao número de integrantes, porte e complexidade. Essa estratégia foi adotada com o objetivo de avaliar a implantação dos processos em projetos que exigiam o suporte de uma arquitetura de processos robusta para o desenvolvimento de software. Algumas dificuldades foram encontradas durante a implantação dos processos nos projetos pilotos. Inicialmente, as equipes dos projetos selecionados mostraram-se resistentes ao volume de trabalho gerado pelo número elevado de processos criados, uma vez que todas já possuíam cronogramas acordados com seus clientes e argumentavam falta de tempo para a implantação dos novos processos. O envolvimento da alta gerência e a atuação do EPG e do grupo de *Quality Assurance* (QA) através de palestras e *workshops* motivou o engajamento de todos. As principais dificuldades identificadas por área de processo e ações tomadas estão descritas na Tabela 3.

Tabela 3: Principais problemas encontrados na implantação do CMMI-SW

ÁREA DE PROCESSO	PROBLEMAS IDENTIFICADOS	PLANO DE AÇÃO
Organizacional	Realização das avaliações dos processos organizacionais (treinamento, medição e análise e gestão de processos).	- Criação de grupos organizacionais constituídos de integrantes de equipes de projetos e integrantes do grupo de Suporte ao Processo de Desenvolvimento (SPD) para gestão das atividades de treinamento, gestão de processos e medição e análise. - Alocação do grupo de QA para a realização das avaliações das atividades dos grupos organizacionais, com a restrição de que os integrantes do grupo de QA não podem estar alocados nos grupos organizacionais.
	Diferenciação dos conceitos de treinamento interno ao projeto e treinamento organizacional.	Definição de diretrizes para os conceitos de treinamento interno ao projeto e treinamento organizacional incluídas no processo de Treinamento.
	Complexidade da gestão do programa de treinamento.	
	Criação da biblioteca organizacional contendo as lições aprendidas e melhores praticas da instituição.	Adoção de uma ferramenta de gestão de conhecimento – MC2® [Secrel 2006] – para auxiliar na gestão do programa de treinamento, criação da biblioteca organizacional e criação do repositório organizacional de medições.
	Criação do repositório organizacional de medições.	
	Difusão do uso da biblioteca organizacional	- Criação e divulgação de um projeto conceito denominado Sócrates que contém informações da média de produtividade, melhores documentos e lições aprendidas da organização. - Uso da ferramenta JIRA® [Atlassian 2006], onde são registradas as sugestões de melhoria, problemas encontrados e submissão de lições aprendidas e de melhores documentos.

Gestão de Projetos	Entendimento dos conceitos de planejamento e controle da gestão de dados	- Incluir o planejamento da gestão de dados no planejamento do projeto e no planejamento da gestão de configuração - Incluir o acompanhamento da gestão de dados nas auditorias de configuração e nas auditorias do grupo de QA.
	Definição e uso do processo adaptado do projeto.	Documentação de critérios e orientações para adaptação do processo organizacional baseado nas características dos projetos da instituição.
	Definição de fontes e categorias para os riscos	Identificação de fontes e categorias de riscos através de pesquisa na literatura disponível e entrevistas nos projetos.
Suporte	Definição das medições e indicadores alinhados às estratégias da instituição.	- Criação do mapa estratégico da instituição baseado em BSC para priorização dos objetivos e necessidades organizacionais. - Alocação de uma pessoa dedicada para a definição e institucionalização do processo
	Avaliações de qualidade baseadas no processo adaptado dos projetos.	Adaptação dos checklists de avaliação de qualidade com base nos itens documentados no processo adaptado do projeto.
	Coleta dos novos indicadores com base nas especificações das medições.	Forte orientação do grupo de Medição e Análise na coleta, análise e divulgação dos novos indicadores.
	Uso do processo de análise e tomada de decisão.	Documentação de situações indicadas para o uso de decisão formal. Treinamento e forte orientação do grupo de QA.
Engenharia	Entendimento dos conceitos de integração contínua e integração complexa.	Adoção de ferramentas que automatizam a realização de testes unitários, testes de integração e montagem do produto com técnicas de integração contínua.
	Uso da rastreabilidade horizontal e vertical de requisitos.	Uso de rastreabilidade em vários níveis, mapeando desde funcionalidades até criação de manuais utilizando a própria estrutura de construção e produtos baseados em casos de uso para geração da rastreabilidade de forma natural ao longo da criação dos produtos de trabalho.
	Maior formalização das entregas para o cliente proposto pelo processo de disponibilização.	- Uso de formulários para formalização de entregas e pesquisa de satisfação do cliente. - Planejamento das entregas, detalhando todas as restrições e exigências do cliente.
	Definição de critérios para seleção dos produtos que serão submetidos às revisões técnicas.	- Documentação das técnicas e abordagens de revisão técnica adotadas pela instituição. - Criação de orientações, indicando artefatos obrigatórios para revisão técnica. Para cada tipo de artefato são associadas as técnicas e abordagens aplicáveis.

Na terceira fase, os ajustes nos processos organizacionais foram realizados com base nos resultados obtidos com as atividades do EPG e do grupo de QA. Foi gerada a primeira *baseline* de processos organizacionais e os processos foram implantados em toda a organização. Neste momento, o número de integrantes do EPG foi reduzido para dezesseis colaboradores com alocação parcial, estando entre suas responsabilidades a manutenção e a avaliação de oportunidades de melhoria nos processos, além de divulgar os avanços realizados a todos os níveis da organização. Para a manutenção dos processos foi adotada a ferramenta JIRA® [Atlassian 2006] onde são registradas as sugestões de melhoria e problemas encontrados nos processos. Esta ferramenta também é utilizada pelos colaboradores da organização para a submissão de lições aprendidas e de melhores documentos dos projetos.

Avaliações

Durante o processo de melhoria, o Instituto Atlântico foi submetido a diversos níveis de avaliação. Em agosto de 2005 foi realizada a avaliação Classe C. O escopo desta primeira avaliação foi verificar o grau de aderência dos processos definidos com relação às áreas de processo do modelo CMMI. Em novembro de 2005 foi realizada a avaliação Classe B. O escopo desta avaliação foi verificar o grau de institucionalização dos processos na instituição. Esta avaliação seguiu todo o formalismo exigido pelo *Standard CMMI Appraisal Method for Process Improvement* (SCAMPI) que é o método utilizado para avaliações oficiais do CMMI, exigindo um grande esforço da instituição na preparação do PIID (*Practice Implementation Indicator Descriptions*). A montagem do PIID é o ponto crítico da avaliação, pois consiste no levantamento e mapeamento de todas as evidências necessárias para comprovar a institucionalização dos processos definidos. Foram gastas aproximadamente mil horas nesta atividade.

Os problemas e sugestões de melhoria identificadas nas avaliações Classe C e Classe B foram tratados por meio de planos de ação elaborados com base nas sugestões de correção e melhorias levantadas. Em fevereiro de 2005 foi realizada a avaliação Classe A. O escopo desta avaliação é verificar a aderência e institucionalização dos processos na instituição. O esforço necessário para a montagem do PIID para esta avaliação foi reduzido, pois a construção do mesmo foi iniciada com a realização das avaliações anteriores. Esta avaliação abrangeu todos os processos da arquitetura de processos do Instituto Atlântico, com exceção do Processo de Gestão de Fornecedores avaliado como não aplicável na instituição. Foram avaliados quatro projetos. O perfil médio dos projetos avaliados está descrito na Tabela 4.

Tabela 4: Perfil dos projetos avaliados

DURAÇÃO (MESES)	ESFORÇO (HOMEM.HORA)	TAMANHO DA EQUIPE (Nº DE PESSOAS)
18	26.000	24

4. Resultados Obtidos

No final do projeto de implantação do CMMI-SW Nível 3, o Instituto Atlântico foi certificado como uma organização CMMI-SW nível 3, sendo todas as práticas do modelo avaliadas como totalmente implementadas.

Outros resultados importantes obtidos com a implantação deste projeto estão descritos a seguir:

- Publicação de quatro artigos no Simpósio Brasileiro de Qualidade de Software 2006 (SBQS)
 - Evoluindo do SW-CMM nível 2 para o CMMI-SW nível 3 - a experiência do Instituto Atlântico
 - Experiência na definição e implantação do processo de análise e tomada de decisão alinhada ao CMMI-SW
 - Processo de Gestão de Processos: uma implementação da gerência de processos organizacionais do CMMI-SW Nível 3
 - Definindo e Implantando um Processo de Medição e Análise: abordagem para o CMMI-SW Nível 3
- Publicação de um artigo no Simpósio de Informática da PUCRS 2006 (SIMS)
 - Nível de Aderência das Práticas de Gestão do Conhecimento em uma Instituição CMMI Nível 3 e ISO 9001:2000
- Publicação de dois artigos na Conferência Latina Americana de Informática 2006 (CLEI)
 - Análise e Tomada de Decisão Formal Alinhada ao CMMI: um Estudo de Caso
 - Implementação de um Repositório de Medições para os Níveis 2 e 3 do CMMI
- Geração de três dissertações de mestrado na Universidade de Fortaleza (UNIFOR)
 - Pontos de Caso de Uso Técnico (TUCP): uma Extensão da UCP
 - Um Processo de Gerenciamento de Riscos para Projetos de Software
 - Um Modelo de Repositório Organizacional de Medições de Projetos de Software

5. Aplicabilidade dos Resultados

Com a certificação alcançada, o Instituto Atlântico figura em um seleto grupo de menos de dez empresas com esse nível de maturidade no Brasil. O nível 3 CMMI aumentou o grau de controle da gestão e expandiu o foco para a área de engenharia, teste, construção e projeto dos produtos, levando a um amadurecimento do processo de desenvolvimento da instituição e conseqüente melhoria nos serviços prestados.

A filosofia de melhoria contínua existente na organização também levou a resultados importantes em áreas como gestão do conhecimento e treinamento da equipe.

Também é importante destacar o número de artigos e dissertações gerados ao longo do projeto de implantação deste nível de maturidade, apontando mais uma vez a relevância das pesquisas desenvolvidas e possibilitando que outras organizações fiquem cientes do trabalho realizado.

6. Características Inovadoras

O Instituto Atlântico iniciou a implantação de um amplo programa de qualidade desde a sua fundação em novembro de 2001 e foi avaliado como uma organização SW-CMM nível 2 em outubro de 2003. Desde então, o *Engineering Process Group* (EPG) da instituição vem trabalhando na estruturação de uma arquitetura de processos organizacionais alinhada ao CMMI-SW nível 3.

O Instituto Atlântico foi a primeira instituição do norte e nordeste do Brasil a conquistar o CMMI nível 3. De acordo com o SEI (*Software Engineering Institute*) [SEI 2006], o Brasil ainda possui poucas organizações com esse nível de maturidade ou com nível de maturidade superior.

7. Conclusão

Este artigo descreveu as atividades realizadas desde o planejamento até a implantação dos novos processos, mostrando as principais dificuldades encontradas e as soluções adotadas. Ao longo deste processo, identificamos um conjunto de lições aprendidas que devem ser destacadas e estão descritas a seguir.

- A ênfase de atuação do grupo de qualidade como orientador facilita a identificação de deficiências nos processos.
- O planejamento da melhoria do processo de software deve ser formal.
- A gestão do Programa de Treinamento é um fator crítico.
- As avaliações intermediárias contribuem para o processo de melhoria.
- O suporte de uma empresa de consultoria especializada tem expressiva importância na condução do programa como um todo.

Referências bibliográficas

Chrissis, M. B., Konrad, M. and Shrum S. (2004) "CMMI - Guidelines for Process Integration and Product Improvement". SEI - Series in Software Engineering, Addison-Wesley.

Pires, C. G., Marinho F. G. and Souza, G.T. (2004) "Uma Arquitetura de Processos para ISO 9001:2000 e SW-CMM Nível 3". VI Simpósio Internacional de Melhoria de Processo de Software - SIMPROS. São Paulo.

Secrel Consultoria e Sistemas Ltda (2006) All rights reserved.
<http://www.mc2.com.br>. Acessado em 08/03/2006.

Rational Unified Process Tutorial (2002) Versão 2002 05 00.



Atlassian Software Systems Pty Ltd (2006) All rights reserved. Copyright © 2005.
<http://www.atlassian.com/software/jira/>. Acessado em 23/02/06.

SEI - Software Engineering Institute (2006) All rights reserved.
<http://www.sei.cmu.edu/>. Acessado em 13/03/06.

[2.08] Aumento dos índices de produtividade no desenvolvimento de softwares: remodelando o processo de gestão de recursos

Entidade: Faculdade de Tecnologia - Universidade Federal do Amazonas (UFAM)
Av. Gen. Rodrigo Octávio Jordão Ramos, 3000, Campus Universitário, Bairro Coroadó I. CEP 69077-000 Manaus - AM - Brazil

Autor: Rayfran Rocha Lima - rayfran.rocha@fucapi.br

Abstract. *This article presents the results of a field research, whose exploratory character is based on multiples study cases about the factors which lead to success or failure on software development. The first part of this article presents the conjuncture which provides the core of this work, moreover, it shows the methodology used to do it. After, it presents the results achieved, its applicability and new features for this kind of research. Finally, it concludes reporting the need of an effective path to use the productive resources.*

Resumo. Este artigo apresenta os resultados de uma pesquisa de campo, cujo caráter exploratório subsidia-se no estudo de múltiplos casos sobre fatores condicionantes de sucesso e fracasso no desenvolvimento de *software*.

Inicialmente, apresentam-se as inquietações que levaram a definição do cerne deste trabalho, bem como, a metodologia utilizada em sua realização.

Posteriormente, são apresentados os resultados alcançados, sua aplicabilidade e as características inovadoras da pesquisa. Por fim, conclui-se pela necessidade em utilizar, de forma efetiva, os recursos produtivos.

1. Introdução

Com o desenvolvimento do capitalismo, o enfoque na qualidade e a posse de tecnologia avançada deixaram de ser o único diferencial na luta pela perenidade das empresas de produção de *software*. No entanto, destaca-se que tais recursos podem até garantir a eficácia do desenvolvimento de seus produtos, contudo, não são capazes de garantir que tal meta seja alcançada com eficiência.

É nesse cenário que emerge o cerne dessa pesquisa, tendo em vista que, apesar das empresas disporem de equipamentos e sistemas com alta tecnologia unidos a profissionais qualificados, continua enfrentando problemas gerenciais, os quais se apresentam como um entrave para galgar uma posição confortável perante seus clientes e concorrentes. Isto se torna evidente, quando as fábricas de *software* fecham contratos com data e custos fixos e não os cumprem.

Além disso, com a introdução de novos concorrentes no mercado de *software* local e nacional, tais fragilidades tornam-se mais evidentes e importantes, à medida que os clientes, atuais e em potencial, aumentam a sua lista de fornecedores, os quais

possuem infra-estrutura compatível e, em muitos casos, superior à das empresas locais.

Diante destes fatos, esta pesquisa buscou responder a seguinte pergunta: quais os principais fatores levam uma fábrica de *software*, possuidora de infra-estrutura composta por alta tecnologia e profissionais qualificados, ao cumprimento ou não dos prazos e custos dos projetos de *software* contratados?

Os resultados obtidos apontam para o método de gestão adotado pelos líderes e gerentes. Durante o levantamento bibliográfico e de campo verificou-se uma grande quantidade de fatores condicionantes de sucesso e fracasso dos projetos de *software*. O método de pesquisa utilizado permitiu diminuir esta quantidade a partir do contraste entre dois fatores bidimensionais: conhecimento empírico *versus* conhecimento científico e opinião *versus* atitudes.

2. Objetivos e Justificativas

Como objetivo geral, pretendeu-se verificar a diferença existente entre teoria e prática no que tange à gestão de recursos adotada pelos gerentes de projeto, propondo diretrizes e ferramentas voltadas para a necessidade das fábricas de *software* instaladas em Manaus.

Deste modo, destaca-se que ao longo da participação nos projetos de *software* desenvolvidos em uma das fábricas de *software*, foram percebidas algumas peculiaridades formadoras de um cenário que gerou várias inquietações, a principal delas foi a dificuldade no cumprimento de prazos e custos contratados.

Além disso, verificou-se que a metodologia de desenvolvimento, ou seja, a gestão dos recursos era particular a cada projeto, possuindo como base, o arcabouço técnico-profissional de seu líder, gerência e demais componentes da equipe.

Outras características são advindas do tamanho das equipes, estas que ficam em torno de três a dez pessoas, tendo, no mínimo, um líder de projeto junto a sua equipe formada de analistas de sistemas e programadores, sem mencionar os arquitetos, projetista de banco de dados, gerente de configuração etc. Em algumas situações, as funções e responsabilidades são confundidas, já que muitas vezes, o analista programa e o programador analisa.

Os programadores são dependentes diretos dos analistas. É responsabilidade do analista, passar o projeto detalhado e com ausência de dúvidas para o programador.

Contudo, muitas vezes, este fica “ocioso”, esperando ser eliminada uma dúvida entre o analista e o cliente, pois não existe um controle de quando será necessária tal informação. Por isso, o tipo de produção sofre efeitos semelhantes ao da sazonalidade, a qual possui picos de demanda, traduzidas em um ciclo de produção desnivelado, possuidor de horas-extras e ociosidades não planejadas.

Diante deste cenário, acredita-se que muitos dos entraves podem ser sobrepujados a partir da aplicação de práticas de gestão de recursos. Então, pretende-se criar uma nova perspectiva para as empresas, ao que se refere à utilização dos recursos humanos e materiais, gerando um diferencial competitivo, bem como, uma nova percepção de sua imagem no mercado local e nacional.

Devido às várias pesquisas levantadas possuírem seus resultados gerados a partir da visão dos clientes, pretendeu-se analisar como o processo de gestão de recursos produtivos se efetiva nos vários projetos desenvolvidos sob o comando de gerentes de projetos atuantes nas fábricas de *software* de Manaus, gerando uma nova visão sobre os fatores condicionantes de sucesso ou fracasso, baseada na visão do fornecedor, ou seja, de quem faz e não de quem usa.

Postula-se que devido à particular do local da pesquisa, Zona Franca de Manaus (ZFM), tanto os métodos de gestão quanto os recursos materiais e humanos geridos, não podem ficar somente seguindo, na íntegra, metodologias utilizadas em grandes centros ou em países desenvolvidos e nem aquelas criadas a partir de experiências empíricas de uma organização isolada. Deste modo, fez-se necessário proporcionar uma nova visão aos métodos já existentes, uma visão conjunta, baseada em conhecimentos empíricocientíficos, mitigando o aparecimento de práticas inadequadas à realidade de uma região.

Neste contexto de ZFM, destaca-se que por muitos anos, indústrias e empresas comerciais instaladas neste pólo vêm sendo reféns de produtos tecnológicos fornecidos por fábricas de *software* localizadas em outros estados e em outros países devido à falta de alternativas locais.

Com o intuito de fortalecer os pólos de *softwares* regionais, o Governo Federal criou um programa chamado SOTEX, o qual é representado em nossa região pelo Pólo Amazonsoft, composto por aproximadamente nove *software houses*. Atualmente, o Pólo de *Software* localizado na ZFM se propõe em responder à demanda por *software*, porém ainda vem enfrentando dificuldades em competir neste mercado global.

3. Metodologia de Execução

Como foi mencionado, o local de realização desta pesquisa não se restringiu aos limites de uma única organização, mas contou com a participação de 23 fábricas de *software* e 40 gerentes de projeto entrevistados.

Este trabalho realizou uma pesquisa empírica, sustentada por um estudo de múltiplos casos reais. Como base para o levantamento e análise dos resultados utilizouse da dialética materialista proposta por Engels e Marx levantadas em Politzer (1979, apud Marconi e Lakatos, 2000:83). Portanto, compreende-se que os resultados apresentados não são imutáveis, mas estes representam um retrato da realidade num dado momento a partir da análise e interpretação da percepção dos informantes pelo pesquisador.

Com relação à operacionalização da pesquisa, é importante ressaltar que esta se deu em três grandes fases. A primeira se refere à busca de conhecimento sobre o assunto pesquisado, utilizando a revisão literária, troca de conhecimento com outros pesquisadores e alguns gerentes de projeto.

Na segunda fase, definiu-se a população, o formato e conteúdo do formulário de entrevista e, percebeu-se a necessidade de criar perguntas semi-abertas e fechadas visando capturar o máximo de informações, mitigando o risco dos vieses criados a partir de uma visão tendenciosa sobre os assuntos abordados. O formulário foi validado através de um pré-teste, sendo realizados os ajustes cabíveis. Ainda nesta fase, foi criado o modelo conceitual e operativo da pesquisa que subsidiou, de forma sistemática, a captura, tratamento e análise do resultado das entrevistas.

A última fase ficou responsável pela: realização das entrevistas, tabulação, apresentação e interpretação do seu resultado; verificação da hipótese; geração de novos conhecimentos e geração do modelo de gestão a ser proposto.

Como foi mencionado no final do detalhamento da Segunda Fase do trabalho, o Modelo Conceitual e Operativo da Pesquisa vêm estabelecer a estratégia de construção do resultado, ou seja, o modelo voltado para as necessidades dos gestores buscando o aumento da produtividade no desenvolvimento de *software*, a partir da manipulação dos dados coletados em três etapas distintas e complementares. Este modelo é ilustrado pela Figura 1.

Antes de discorrer sobre as etapas do modelo, faz-se necessário explicar as formas, figuras e sua representação para este modelo. Ao olhar para a Figura 1, percebe-se a existência de retângulos, uns com o fundo branco e outros com fundo cinza, os quais vão se fragmentando ao longo das três etapas. Os retângulos brancos representam os fatores que influenciam o desempenho dos projetos de *software*, levantados pelo pesquisador antes das entrevistas. Já os retângulos cinzas representam aqueles fatores que não foram previstos, portanto, advindos das respostas dos informantes da pesquisa.

Outro fenômeno a ser destacado é que, ao passar de uma etapa para outra, esses fatores se misturam e diminuem em quantidade, ou seja, alguns fatores continuarão sendo observados, mas não farão parte do conjunto de fatores críticos contido no modelo proposto. Este fato vem representar a estratificação e filtragem sofrida pelos fatores durante todo o ciclo de vida da pesquisa, buscando, ao final, obter um conjunto minimizado de fatores. A estratificação gerou uma visão tricotômica: fatores de sucesso, fatores de fracasso e aqueles que influenciam em ambos os resultados. A filtragem se deu a partir do cotejo entre opiniões e atitudes, ou seja, aquilo que acham interessante fazer por causa de *marketing* ou *benchmark* e aquilo que é essencial para o alcance das metas de um projeto.

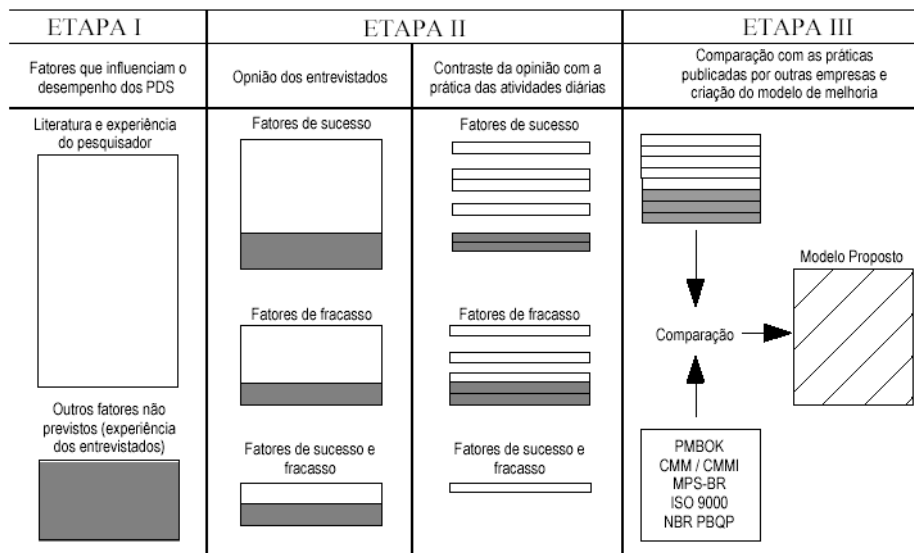


Figura 1 – Modelo Conceitual e Operativo da Pesquisa
Fonte: Adaptação de Lima (2006)

Para finalizar a explicação dos símbolos, destaca-se o retângulo que contém alguns modelos consolidados no mercado e na academia, tais como: PM-BOK, CMMI, MPS-BR, ISO9000 etc. Estes conhecimentos servirão de base para a criação do Modelo Proposto representado por um retângulo rachurado. Contudo, adverte-se que este modelo não pretende ser uma panacéia, mas um conjunto de diretrizes voltadas para a gestão dos recursos produtivos, as quais utilizarão as ferramentas e técnicas disponíveis no mercado pelo fato delas serem necessárias e não pelo fato de existirem.

Uma vez explicada as formas e figuras, bem como, a representação de cada uma, parte-se para dissertar sobre cada Etapa do modelo ilustrado na Figura 1. Assim, a Etapa I consistiu em levantar os principais fatores que, geralmente, exercem influência no desempenho dos projetos de *software* (PDS), a qual se valeu da primeira Fase desta pesquisa e da primeira entrevista. Para esta entrevista, utilizaram-se dois formulários, cuja finalidade foi coletar quatro categorias de informações: a) informações gerais sobre a organização; b) informações gerais sobre o entrevistado; c) informações sobre a experiência na gestão de recursos produtivos; d) informações sobre os fatores que influenciam o desempenho dos Projetos de *Software*.

A Etapa II do modelo refere-se à realização da aplicação de um terceiro formulário, o qual visa verificar, de acordo com a opinião dos entrevistados, quais são os principais fatores que levam ao alcance do sucesso, do fracasso e aqueles que podem influenciar em ambos os resultados.

Optou-se por se fazer, nas Etapas I e II, perguntas semi-abertas para evitar a indução dos entrevistados quanto às respostas e tentar capturar novos fatores até então não visualizados pelo pesquisador e pela literatura estudada.

Na Etapa III, após ter identificado os prováveis fatores que realmente fazem parte do cotidiano dos entrevistados, buscou-se compará-los com as práticas de sucesso de outras organizações, advindas da literatura e modelos já consolidados no mercado nacional e internacional, tais como: as práticas contidas no PMBoK, CMMI, MPS-BR, ISO 9000, SPICE etc., buscando elencar as melhores práticas que lidam com estes fatores, evitando a práticas de simulacros.

4. Resultados Obtidos

Durante este projeto foi desenvolvida uma metodologia chamada Teia Estratégica, a qual objetiva fornecer técnicas de negociação a partir da percepção de que o sucesso é composto por um conjunto de fatores dinâmicos e interdependentes.

Os resultados do projeto proporcionaram a geração de quatro artigos os quais foram publicados completos conforme lista abaixo:

LIMA, R. R., ANDRADE, J. B. L. Aumento dos índices de produtividade no desenvolvimento de *softwares*: remodelando o processo de gestão de recursos. In: II Simpósio de Engenharia de Produção, 2005, Manaus. Anais, 2005.

LIMA, R. R. Aumento dos Índices de Produtividade no Desenvolvimento de *Software*: Remodelando o Processo de Gestão de Recursos. In: Encontro da Qualidade e Produtividade em *Software* - Manaus, 2005, Manaus. ANAIS. Manaus: MCT, 2005.

LIMA, R. R., Teia Estratégica: uma nova abordagem para a estratégia produtiva orientada ao desenvolvimento de sistemas de informação. In: VII Simpósio Internacional de Melhoria de Processo de Software, 2005, São Paulo. ANAIS. São Paulo: SENAC, 2005.

LIMA, R. R., ROCHA, A. C. B. Softwares Houses: um ator importante para o sucesso das alianças estratégicas nas redes de empresas. In: I Simpósio de Engenharia de Produção, 2004, Manaus. Anais, 2004.

Os resultados parciais do trabalho foram apresentados na forma de resumo:

LIMA, R. R. The quest for success in software development: remodeling the resource management process. In: I Escola de Inverno em Gestão e Inovação Tecnológica da Escola de Química da UFRJ, Rio de Janeiro. ANAIS, 2005.

LIMA, R. R. Teia Estratégica: uma nova abordagem da estratégia produtiva orientada ao desenvolvimento de sistemas de informação. In: Reunião Regional da SBPC em Manaus. ANAIS, MCT, 2005.

Este trabalho está servindo de base para a conclusão de uma dissertação de mestrado realizada na Faculdade de Tecnologia da Universidade Federal do Amazonas, subsidiado pelo Programa de Pós-Graduação Strictu Senso em Engenharia de Produção.

A partir dos contrastes encontrados entre opinião e atitude dos gerentes de projetos junto à análise do conhecimento empírico com o conhecimento científico, foi possível reduzir a lista de fatores condicionantes de sucesso em 51% e a lista de fatores condicionantes de fracasso em 44%, como mostra a Figura 2.

Todos os fatores que obtiveram o percentual igual ou maior que 90% do somatório das opiniões de “concordo” e “concordo plenamente” permaneceram na lista.

Os demais fatores não devem ser esquecidos ou negligenciados, porém, segundo as respostas obtidas eles não são capazes de influenciar o resultado de forma significativa.

FATORES CONDICIONANTES DE SUCESSO

Categoria	Qtd. Fatores Inicial	Qtd. Fatores Final	Redução %
1 - Em relação às características da equipe de desenvolvimento	8	5	38%
2 - Em relação aos usuários	12	6	50%
3 - Em relação ao cliente	8	4	50%
4 - Em relação ao desenvolvimento do software	18	8	56%
5 - Em relação ao projeto	12	5	58%
6 - Em relação às características do gerente do projeto	8	4	50%
7 - Em relação a Alta Administração da Software House	4	2	50%
Total Geral	70	34	51%

FATORES CONDICIONANTES DE FRACASSO

Categoria	Qtd. Fatores Inicial	Qtd. Fatores Final	Redução %
1 - Em relação às características da equipe de desenvolvimento	8	3	63%
2 - Em relação aos usuários	12	5	58%
3 - Em relação ao cliente	8	7	13%
4 - Em relação ao desenvolvimento do software	18	11	39%
5 - Em relação ao projeto	12	8	33%
6 - Em relação às características do gerente do projeto	8	3	63%
7 - Em relação a Alta Administração da Software House	4	2	50%
Total Geral	70	39	44%

Figura 2 - Redução dos fatores condicionantes de sucesso e fracasso

Fonte: Adaptação de Lima (2006)

De posse da lista otimizada de fatores condicionantes de sucesso e da lista de fatores condicionantes de fracasso, verificou-se que 34% dos fatores são capazes de influenciar o resultado de um projeto tanto para o sucesso quanto para o fracasso.

Outro resultado importante está relacionado à desmistificação de que na ausência de um fator de sucesso o projeto está fadado ao fracasso e vice-versa. Com este estudo, evidenciou-se a existência de fatores que levam um projeto ao sucesso, outros fatores que levam ao fracasso e aqueles que são capazes de levar para ambos os resultados.

Percebeu-se ainda que nas empresas que possuíam algum tipo de certificação na área de qualidade, os gerentes de projetos seguiam uma linha de raciocínio produtivo padronizado. Por outro lado, aquelas empresas desprovidas de um processo organizacional definido ficavam a mercê das características e experiência de cada gerente de projeto.

5. Aplicabilidade dos Resultados

Os resultados desta pesquisa podem ser utilizados pela ciência como matéria-prima para trabalhos de pesquisa relacionados ao aumento da produtividade a partir da compreensão e gestão dos fatores condicionantes de sucesso e fracasso no desenvolvimento de *software*. Além disso, este trabalho se propõe em preencher uma lacuna encontrada nos cursos de gestão de projeto de *software*, pois muitos alunos sentem dificuldades em encontrar um ponto de partida, uma vez que a área de gestão de recursos produtivos é muito abrangente.

A partir da posse da nova lista de fatores condicionantes, sendo 34 inerentes ao sucesso e 39 inerentes ao fracasso, alunos e profissionais são capazes de saber por onde começar. Neste sentido, gera-se uma nova perspectiva sobre o planejamento, gestão e aprimoramento dos recursos produtivos. Para os gerentes novatos, existe uma lista de fatores iniciais que necessitam de atenção. Para os gerentes mais experientes, fornece uma forma de rever conceitos sob a luz focada para os fatores e situações que merecem maior atenção durante o desenvolvimento de um projeto.

Diante destes resultados, as fábricas de *software* são capazes de verificar a similaridade com o mercado, buscando assim, diferencial competitivo. Este trabalho vem gerando a necessidade da reflexão por parte das empresas no sentido de verificar as metodologias e ferramentas utilizadas para alcançar sucesso em seus projetos, a partir da resposta ao seguinte questionamento: onde estão sendo gastos nossos recursos? Será que estamos caminhando na direção correta? As metodologias, processos e ferramentas utilizadas são adequadas às reais necessidades da empresa?

6. Características inovadoras

Atualmente, muitos pesquisadores estão preocupados em realizar um levantamento sobre os fatores condicionantes de sucesso e/ou fracasso em projetos de *software*, além disso, tais estudos se baseiam em estudos de casos únicos. Entretanto, destaca-se que o resultado destas pesquisas vem aumentando, cada vez mais, a lista destes fatores.

Com o intuito de consolidar ao invés de aumentar a lista dos fatores condicionantes, esta pesquisa realizou uma revisão nas metodologias contemporâneas utilizadas pelos líderes de equipes, contrastando-as com as reais necessidades das fábricas de *software* pesquisadas, propondo uma metodologia mais adequada e enxuta, buscando evidenciar as discrepâncias existentes entre opinião e atitudes.

Outra faceta desta pesquisa é analisar como esse processo se efetiva nos vários projetos desenvolvidos por essas fábricas de *software*, o que se assemelha à pesquisa realizada por Moraes (2004) que, sob a ótica do cliente, estudou os resultados alcançados por várias empresas que optaram por adquirir e implantar projetos de *software*, sendo escolhidos apenas um projeto de cada empresa para compor a amostra da pesquisa.

Deste modo, acredita-se que este tipo de estudo, implementado dentro das fábricas de *software* gerou uma nova visão sobre os fatores condicionantes de sucesso ou fracasso em projetos de *software* desenvolvidos, pois, esta

perspectiva será gerada sob a ótica do fornecedor, subsidiado pelo estudo de casos múltiplos.

7. Conclusão e Perspectivas Futuras

Com esta pesquisa, verificou-se a existência de uma visão dicotômica sobre sucesso de projetos, muitas vezes negligenciada ou confundida. Esta visão se divide em: sucesso na gestão do projeto (processo) e sucesso do projeto (produto). Por esta razão, a falta de um conjunto de fatores gerenciáveis tornava cada vez mais difícil o alcance dos objetivos do projeto.

De acordo com os dados levantados neste estudo, defende-se que a falta de uma visão holística e o não envolvimento de todos os setores no processo de tomada de decisões gerenciais, acarretam na aquisição de soluções que contemplam a melhoria de alguns setores, tendo como consequência, o prejuízo de outros. Portanto, assevera-se que este fenômeno é responsável por conceber um dos maiores problemas enfrentados pelas organizações, a debilidade do seu processo decisório.

A concepção anterior pode ser verificada através da existência de várias metodologias, ferramentas e processos com características incompatíveis às organizações usuárias, os quais se propõem em aumentar a produtividade ou competitividade das organizações. No entanto, ao invés disso, estes são responsáveis por sobrecarregar algumas empresas que optaram por utilizá-los sem prévia análise de sua necessidade, além de prejudicarem o controle e planejamento da produção.

Depois da realização das entrevistas e divulgação dos resultados, foram criadas novas expectativas e visões direcionadas ao gerenciamento de recursos produtivos. Esta pesquisa veio ratificar a idéia de que a atuação de um processo de melhoria na causa principal é capaz de eliminar a maioria dos efeitos colaterais, ocasionando benefícios significativos para o sistema de forma global. Além disso, a atuação individual nos efeitos colaterais poderá trazer melhorias ao sistema, mas de uma ordem de magnitude muito menor.

Neste contexto, acredita-se que a capacidade gerencial pode ser debilitada pelo aumento do campo de atuação. Em outras palavras, quanto mais disperso for o alvo, menor precisão será necessária. Assim, quando não se sabe onde quer chegar, um caminho qualquer pode resultar em resultados diferentes das metas estabelecidas.

Portanto, destaca-se que a diminuição dos fatores condicionantes tornou mais visível o ponto de convergência. Com isso, o gerente de projeto pode direcionar seus esforços para aqueles fatores que trarão um resultado maior.

Destaca-se que este trabalho não procurou esgotar todas as frentes de trabalho em gestão de projeto de software, mas sua finalidade foi de contribuir à etapa inicial de uma série de pesquisas neste campo de conhecimento.

Como trabalho futuro, pretende-se realizar a mesma pesquisa, porém sob o ponto de vista dos subordinados para verificar o contraste entre as visões de líder e liderado.

Outro objeto de estudo para pesquisa futura deverá ser composto pelos gerentes de projetos que trabalham em organizações que constroem seus próprios sistemas, mesmo não tendo como negócio-fim a construção de software.

8. Referências Bibliográficas

1. LIMA, R. R., ANDRADE, J. B. L. Aumento dos índices de produtividade no desenvolvimento de *softwares*: remodelando o processo de gestão de recursos. In: II Simpósio de Engenharia de Produção, 2005, Manaus. Anais, 2005.
2. LIMA, R. R., Teia Estratégica: uma nova abordagem para a estratégia produtiva orientada ao desenvolvimento de sistemas de informação. In: VII Simpósio Internacional de Melhoria de Processo de *Software*, 2005, São Paulo. ANAIS. São Paulo: SENAC, 2005.
3. LIMA, R. R., ROCHA, A. C. B. *Softwares Houses*: um ator importante para o sucesso das alianças estratégicas nas redes de empresas. In: I Simpósio de Engenharia de Produção, 2004, Manaus. Anais, 2004.
4. LIMA, R. R. Sistemas de Informação: uma alternativa para eliminação de gargalos em uma empresa de manufatura com sistema de produção intermitente. Monografia. Especialização em Engenharia de Produção – Faculdade de Tecnologia. UFAM, 2004.
5. MARCONI, Marina de A. e LAKATOS, Eva Maria. Técnicas de Pesquisa. 5 ed. São Paulo: Editora Atlas, 2002.

[2.10] Adequação do Processo de Desenvolvimento de Software à Lei Sarbanes Oxley

Entidade: Centrais Elétricas do Norte do Brasil S.A - ELETRONORTE - Diretoria de Gestão Corporativa Superintendência de Tecnologia da Informação - SCN Quadra 6, Conj. A, Bl. C, Sala 805 - 70716-900 - Brasília - DF

Autores: Celso Patrício Cavalcanti Silva, Floriano Brandão Filho, Teresinha Moraes Falabella de Castro - celso@eln.gov.br, florianobrandao@eln.gov.br, tfalabella@eln.gov.br

1. Introdução

A Eletronorte é uma concessionária de serviço público de energia elétrica, sociedade anônima de economia mista, subsidiária das Centrais Elétricas Brasileiras S.A. – Eletrobrás e tem como finalidade principal, a realização de estudos, projetos, construção e operação de usinas geradoras e de sistemas de transmissão e distribuição de energia elétrica, além da comercialização da energia gerada pela Empresa.

A Eletrobrás é uma empresa de economia mista e de capital aberto, com ações negociadas nas Bolsas de Valores de São Paulo (Bovespa), de Madrid na Espanha, e de Nova Iorque, nos Estados Unidos. O Governo Federal possui mais da metade das ações ordinárias e preferenciais (52,45%) da Eletrobrás e, por isso, tem o controle acionário da empresa. O Grupo Eletrobrás atua de forma integrada. As

políticas e diretrizes adotadas são definidas pelo Conselho Superior da Eletrobrás (Consize), formado pelos presidentes das empresas do grupo, que se reúne regularmente. Criada em 1962 para promover estudos e projetos de construção e operação de usinas geradoras, linhas de transmissão e subestações, destinadas ao suprimento de energia elétrica do País, a Eletrobrás adquiriu características de holding, controlando empresas de geração e transmissão de energia elétrica. As empresas do Grupo Eletrobrás produzem cerca de 60% da energia elétrica consumida no país. São elas: Chesf, Furnas, Eletronorte, Eletronuclear e CGTEE. A Eletrobrás detém ainda 50% da Itaipu Binacional. Também integram o Grupo Eletrobrás a Lightpar, o Cepel, bem como a Eletrosul, empresa transmissora de energia elétrica.

2. Objetivos e justificativas

A Eletrobrás, ao subir o nível das ações da Bolsa de Valores de Nova York, se obriga, ela e suas subsidiárias, a implantarem os controles internos solicitados pela Lei Sarbanes-Oxley (lei americana aprovada em 2002). Tais controles visam mitigar riscos identificados relacionados às operações e à segurança dos sistemas de informações.

Na Eletronorte, a implantação dos controles exigidos pela lei, para a primeira certificação em 2006, foi conduzida na forma de um projeto, com conseqüente alteração de processos internos. Estas mudanças nos processos e a manutenção possibilitarão a re-certificação, a ser realizada anualmente por auditoria independente.

A necessidade de adequação às exigências da Lei Sarbanes-Oxley (SOX) exigiu a adequação dos sistemas informatizados e a implementação de processos que garantam que todas as alterações de sistemas que ocorrerem posteriormente, e/ou implementações de novos sistemas, também estarão de acordo com as exigências da Lei. Em função dessa necessidade, a Eletronorte optou por documentar os requisitos e as soluções-padrão para atendimento à Lei utilizando UML, e adequar o processo de desenvolvimento e implantação de sistemas para contemplar e adotar esse padrão. O trabalho apresentado consiste em modelo padrão desenvolvido e seu impacto na metodologia de desenvolvimento adotada pela Eletronorte.

Para que se possa compreender o escopo da implantação dos controles da Lei Sarbanes Oxley e conseqüentemente o escopo do modelo desenvolvido, faz-se necessário apresentar como foi estruturado e desenvolvido o Projeto de implantação da Lei na Tecnologia da Informação na Eletronorte – Projeto SOX - que apresentaremos a seguir:

Projeto SOX na Eletronorte

Na Eletronorte, o projeto foi dividido em 3 segmentos, sendo o de negócios, conduzido pela Auditoria Interna e os outros dois, pela Superintendência de Tecnologia da Informação:

TICA (Technology Information Control Applications) – controles necessários às aplicações que suportam os negócios empresariais, especialmente nos processos que geram lançamentos em “sistemas

financeiros-chave”, isto é, ligados a movimentos ou saldos contábeis/financeiros representativos no cálculo da “materialidade” da empresa.

TIGC (Technology Information General Controls) – controles exigidos aos processos internos da área de Tecnologia da Informação, sendo seis ao todo, conforme detalhado a seguir (para estes, foi utilizado, como base, o modelo COBIT de Governança em TI, nos processos de tecnologia aderentes à Lei):

Ambiente geral e físico: Acesso físico ao CPD, Condições ambientais do CPD

Backup e recuperação de dados: Execução de backups, Recuperação de dados

Mudança de controles de tecnologia: Requisição de mudanças, Desenvolvimento e testes, Liberação para produção, Mudanças emergenciais

Administração dos usuários: Gerenciamento de acesso, Revisão periódica, Monitoramento de acesso

Gerenciamento de configuração: Configuração de sistemas, Configuração de rede e infra-estrutura, Configuração de Banco de Dados

Processamento (batch Interface): Escalonamento de rotinas, Monitoramento de rotinas

3. Metodologia de execução

A metodologia incluiu os seguintes passos:

3.1. Desenvolvimento / manutenção de software – o segmento de desenvolvimento e manutenção de software na Eletronorte, objeto deste trabalho, foi influenciado pelos processos internos relacionados a seguir, definidos no Item 2 acima (Projeto SOX na Eletronorte), para os quais foram exigidos controles específicos visando reduzir os riscos identificados, após uma análise minuciosa.

Backup e recuperação de dados (*Execução de backups, Recuperação de dados*):

O processo de desenvolvimento de software, deve identificar os pontos de risco do sistema e implantar ações de mitigação destes riscos por meio de gravação de “logs” em pontos chave:

- Sistemicamente e conforme configuração prévia, os pontos sensíveis geram registros no arquivo de Log em local previamente determinado o qual será gravado em área segura (servidor segregado) com acesso controlado.
- Devem existir procedimentos e controles para que:

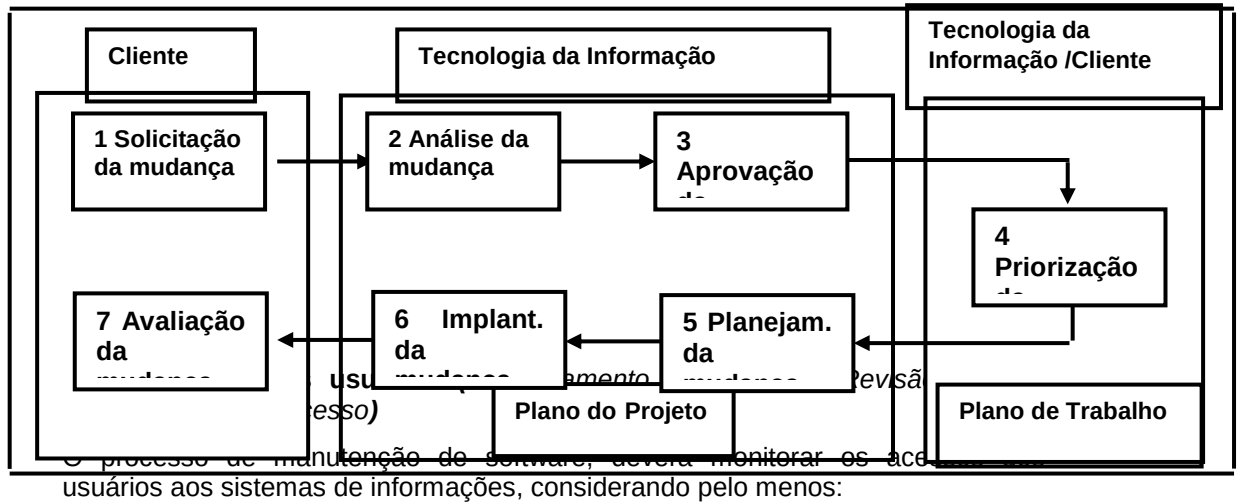
- na geração e na leitura se garanta a integridade do conteúdo do arquivo de Log;
- garantir a existência da periodicidade na geração de log's;
- garantir que o conteúdo dos arquivos de log seja analisado de acordo com as expectativas e necessidades dos gestores funcionais de cada sistema;
- o resultado da análise do conteúdo dos arquivos de log seja devida e formalmente encaminhado para a área funcional e usuária do sistema a qual tomará as devidas providências cabíveis;
- as ações que sejam adotadas em resposta a existência de atividade indevida no sistema, estejam respaldadas pelas recomendações da política de direitos e deveres no uso de recursos de tecnologia, na Eletronorte
- todas as mudanças, decorrente das recomendações da área gestora do sistema, sigam os mesmos procedimentos formais normatizados na respectiva instrução normativa;
- os arquivos de logs sejam armazenados em ambiente adequado, no prazo correto, e tenham possibilidade de restore a qualquer momento;
- o acesso aos arquivos de log seja realizado somente por pessoal autorizado;

Mudança de controles de tecnologia (*Requisição de mudanças, Desenvolvimento e testes, Liberação para produção, Mudanças emergenciais, Configuração de sistemas*)

O processo de desenvolvimento de software deve definir as ações de manutenção para novas versões, identificando os pontos de risco do sistema para futuras alterações, verificando:

- O Ciclo Desenvolvimento/Manutenção/Aquisição
- Formalização de Solicitação de Mudanças
- Autorização para Desenvolvimento de Mudança
- Homologação de Mudança
- Implantação de Mudanças em Produção (mudanças emergenciais)
- Processo Formal de Desenvolvimento
- Identificação de Mudança Referente aos sistemas
- Mudanças em Estrutura de Banco de Dados
- Acesso a Tabelas do Banco de Dados
- Segregação de Ambientes para Sistemas
- Consistência e Integridade de Interfaces entre Sistemas
- Testes de Interfaces
- Integridade na Conversão de Sistemas

O planejamento da mudança deverá seguir o esquema abaixo:



O processo de manutenção de software, deverá monitorar os acessos dos usuários aos sistemas de informações, considerando pelo menos:

- Criação de contas de acesso
- Alteração de contas de acesso
- Cancelamento ou bloqueio de contas de acesso
- Revisão periódica dos perfis atribuídos às contas de acesso
- Revisão periódica de cancelamento de contas de acesso

3.2. Análise completa das exigências da Lei Sarbanes Oxley (riscos e controles) que permitiu a identificação das seguintes premissas para o modelo gerado pelo projeto:

- Aumentar supervisão sobre demonstrações financeiras
- Garantir aderência à seção 404 da Lei Sarbanes Oxley
- Definir e implementar controles de infra-estrutura de TI
- Definir e implementar controles no processo de negócio
- Garantir controles de sistemas adequados (TICA)
- Garantir a geração de evidências para auditoria
- Garantir confidencialidade
- Garantir conhecimento gerencial das informações relevantes
- Garantir consistência e integridade da informação financeira
- Garantir disponibilidade
- Garantir efetividade de padrões e procedimentos
- Garantir Integridade
- Garantir segurança no acesso, manipulação e disponibilização das informações
- Implementar controles exigidos pela Lei Sarbanes Oxley
- Mitigar riscos de inconsistência das informações financeiras

3.3. Levantamento dos riscos nas aplicações que suportam os processos de negócios da Eletronorte e especificação dos controles necessários à mitigação destes riscos – conforme abaixo:

- Aprovação por alçada
- Autorização antecipada da chefia
- Autorização /Ciência da chefia
- Autorizar realização da tarefa
- Ciência posterior da chefia
- Controlar o processo
- Documentar ações, encadeamentos, pareceres e decisões
- Documentar problemas detectados
- Documentar soluções adotadas
- Encaminhar conforme classificação do controle
- Executar tarefa
- Garantir segregação de função
- Guardar os ativos
- Identificar os responsáveis
- Obedecer a periodicidade do controle
- Registrar ações executadas
- Registrar evidências dos controles
- Registro conforme tipo de controle

3.4. Desenvolvimento do Modelo de aderência à SOX: Que consiste no desenho e especificação de casos de uso correspondentes aos controles identificados para os processos de negócios da Eletronorte – são mais de 100 casos de uso levantados, agrupados em 4 módulos – definidos em conformidade com as áreas de conhecimento adotadas no Projeto SOX, que são:

- **Gerenciamento de Acesso:** Compreendendo todos os casos de uso referentes ao gerenciamento de controle de acesso ao sistema de informação.
- **Gerenciamento de Log:** Compreendendo todos os casos de uso referentes à geração e monitoramento dos logs dos sistemas de informação.
- **Interface:** Compreendendo todos os casos de uso referentes ao controle e monitoramento das soluções de interface entre sistemas de informações distintos, sejam eles internos da empresa, ou externos.
- **Gerenciamento de Histórico:** Compreendendo todos os casos de uso referentes à geração de arquivos e conseqüente recuperação de informações históricas relacionadas com o gerenciamento de acesso, gerenciamento de log e interface.

3.5. Adequação da metodologia de implantação e manutenção de sistemas – Consiste na adequação da metodologia de desenvolvimento e manutenção de sistemas próprios e de terceiros, contemplando a adoção do Modelo de Aderência à SOX, de forma a garantir o atendimento aos requisitos SOX por esses sistemas.

4. Resultados obtidos

Como resultado, obteve-se os seguintes documentos:

- 4.1. Modelo de aderência à SOX para desenvolvimento/manutenção de sistemas;
- 4.2. Metodologia de Aquisição, Desenvolvimento, Implantação e Manutenção de Sistemas, utilizando o Modelo de aderência à SOX.

5. Aplicabilidade dos resultados

Os resultados obtidos permitem a sua utilização para qualquer empresa que esteja implementando o SOX e para empresas que prestem serviço de Sistemas de Informação a estas empresas.

6. Características inovadoras

- 6.1. Documentação dos requisitos SOX em UML
- 6.2. Desenvolvimento de um modelo conceitual com soluções que permitam atender aos requisitos SOX, possibilitando a implementação de soluções físicas particulares, pela empresa que o adote.
- 6.3. Proposta de adequação da Metodologia de Aquisição, Desenvolvimento, Implantação e Manutenção de Sistemas, utilizando o Modelo de aderência à SOX

7. Conclusão e perspectivas futuras

O desenvolvimento desse trabalho contribuiu por contemplar o atendimento a requisitos de negócio padrão que estão sendo exigidos por cada vez mais empresas, em função da Lei Sarbanes-Oxley, e consequentemente por empresas fornecedoras de soluções de TI que atendam à essas empresas. Como evolução do trabalho apresentado pretende-se:

- Validar o Modelo de aderência à SOX para desenvolvimento/manutenção de sistemas com outras empresas e com fornecedores de serviços de TI, buscando adequá-lo a ser o mais geral e abrangente possível.
- Desenvolver como um componente modular a metodologia de Aderência à SOX para desenvolvimento/manutenção de sistemas, de forma a garantir a sua utilização de maneira opcional, em qualquer metodologia específica adotada pelas empresas interessadas em utilizá-la.

8. Referências bibliográficas

- IT Control Objectives for Sarbanes-Oxley – Discussion Document , IT Governance Institute, 2004
- COBIT 3º Edition, IT Governance Institute, 2000
- ISO IEC 17799
- IUML - Guia do Usuário / Grady Booch, James Rumbaugh, Ivar Jacobson; tradução de Fábio Freitas da Silva, Rio de Janeiro : Campus, 2000.
- UML 2.0: do requisito à solução / Adilson da Silva Lima - 1. ed., São Paulo : Érica, 2005.
- UML: Prático e Descomplicado / Alexandre Veloso de Matos, São Paulo : Érica, 2002.
- UML 2: Guia de Consulta Rápida / Gilleanes T.A. Guedes, São Paulo : Novatec, 2005

[2.11] Avaliação do Processo da Fundação Desembargador Paulo Feitoza para atender ao nível 2 de maturidade do CMMI

Entidade: Fundação Desembargador Paulo Feitoza - Caixa Postal 69.075-351 - Manaus - AM - Brasil

Autores: Andréia F. Vieira, Ellen Cristina V. Barroso - {avieira,ebarroso}@fpf.br

Abstract. *This paper describes an assessment done in the software development process from Fundação Desembargador Paulo Feitoza, using the CMMI model and appraisal method SCAMPI. The intention of this assessment was to show the strengths and weaknesses of the evaluated process, as well as the improvements opportunities. This way, the process can be improved and adequated to the CMMI level 2 maturity model.*

Resumo. *Este artigo descreve a avaliação executada no processo de desenvolvimento de software da Fundação Desembargador Paulo Feitoza, utilizando o modelo CMMI e o método de avaliação SCAMPI. O intuito desta avaliação foi levantar os pontos fortes e fracos e oportunidades de melhoria do processo, para que o mesmo possa ser melhorado e adequado ao modelo de melhoria de processo CMMI, nível 2 de maturidade.*

1 Introdução

A demanda por sistemas de software com alta qualidade e a um baixo custo está tendo um aumento significativo atualmente. Tal fato tem levado a uma preocupação constante com a produtividade no desenvolvimento e a qualidade dos produtos gerados.

Por conta desta demanda e pela alta competitividade do mercado, as empresas de desenvolvimento de software estão cada vez mais cientes de que obter um software com um padrão de qualidade aceitável não é uma tarefa simples, e um

dos fatores primordiais a serem considerados está relacionado à qualidade dos processos utilizados durante o desenvolvimento, pois se estes forem bem definidos e seguidos, conseqüentemente, o resultado será um produto com qualidade. Portanto, pode-se afirmar que, a qualidade do produto está diretamente relacionada à qualidade do processo.

Visando auxiliar tais empresas, vários modelos de melhoria de processos foram sendo desenvolvidos, dentre eles: SW-CMM (*Capability Maturity Model*) (SEI, 1993), CMMI (*Capability Maturity Model Integration*) (CHRISSIS, 2003) e MPSBR (Melhoria do Processo do Software Brasileiro) (SOFTEX, 2006). Existem também os modelos de avaliação de processo, como o SCAMPI (*Standard CMMI Appraisal Method for Process Improvement*) (SEI, 2001), que são capazes de avaliar tanto a capacidade dos processos quanto a maturidade da organização.

As pesquisas acadêmicas e as próprias organizações estão focando seus esforços na busca da melhoria dos processos de software. Em (CORRÊA, 2005) encontra-se um diagnóstico da área de requisitos de software usando IDEAL, CMMI, e SCAMPI que tem por objetivo melhorar o processo dessa área, identificando os pontos fracos e recomendando melhorias.

(VIVEIROS, 2005) mostra um programa de melhoria iniciado no Banco Nacional de Desenvolvimento (BNDES), para avaliar seu processo de desenvolvimento software. Para esta avaliação, foram levantadas informações com pessoas que faziam parte da equipe de desenvolvimento de softwares. A partir dessas informações coletadas, as atividades de implantação das melhorias a serem feitas foram planejadas, visando atender ao nível 3 do modelo de referência do CMMI.

Este artigo está organizado da seguinte forma: Neste capítulo é dada uma contextualização geral do projeto executado; no capítulo 2, são mostrados o objetivo e a justificativa do projeto; no capítulo 3 é apresentado como o projeto foi desenvolvido; no capítulo 4, os resultados obtidos são apresentados, detalhando o planejamento e a execução; no capítulo 5, a relevância, abrangência e impacto do projeto são explanados; no capítulo 6, são mostradas as contribuições inéditas do projeto; no capítulo 7, a conclusão e possíveis trabalhos futuros são apresentados e no capítulo 8, as referências são citadas.

2 Objetivos e Justificativa

A Fundação Desembargador Paulo Feitoza, visando continuar no mercado competitivo e atender às expectativas dos clientes e sabendo que todo processo precisa ser melhorado continuamente, vem almejando melhorar seu processo e alcançar nível 2 de maturidade do modelo CMMI. Para auxiliá-la, esse trabalho teve como objetivo realizar uma avaliação do processo de desenvolvimento de software da empresa, escrito aderente ao modelo de referência SW-CMM, identificando os pontos fortes e fracos, além de oportunidades de melhoria, de forma que o mesmo fique aderente ao modelo CMMI, nível de maturidade 2.

3 Metodologia de Execução

Para alcançar os objetivos propostos, as seguintes atividades foram executadas: 1) Estudo do modelo CMMI e do método de avaliação SCAMPI, além do estudo do

processo da organização avaliada. Este estudo foi necessário para dar embasamento à execução do projeto; 2) Adequação do SCAMPI para a realidade desta avaliação; 3) Elaboração do Plano de Avaliação, onde foram identificados: escopo, restrições, projetos avaliados, dentre outras informações relevantes para a avaliação; 4) Elaboração do questionário, que foi feito com o intuito de auxiliar nas entrevistas e foi baseado no questionário de maturidade do CMM (ZUBROW, 1994); 5) Realização da avaliação, conduzida de acordo com o plano: nessa atividade foram identificadas evidências objetivas, pontos fortes e fracos, caracterização da implementação das práticas, e recomendações de melhorias. Estas recomendações foram identificadas a partir dos pontos fracos identificados.

4 Resultados Obtidos

A avaliação do processo da FPF baseou-se em um processo de avaliação definido especificamente para a empresa com base no método de avaliação SCAMPI. De acordo com o SCAMPI, muitas atividades devem ser executadas durante uma avaliação. Tais atividades são definidas em três fases: Planejamento, Condução da Avaliação e Reporte de Resultados. Porém, muitas destas atividades não eram pertinentes a presente avaliação, visto que o principal objetivo da mesma era fazer o levantamento de pontos fortes, pontos fracos e possíveis melhorias do processo para adequação do mesmo ao modelo CMMI. Portanto, uma adequação do método se fez necessária, e as atividades executadas e as não executadas para esta avaliação são mostradas a seguir.

4.1 Planejamento

Na Fase de Planejamento, o que se julgou adequado para a presente avaliação, foram as seguintes atividades: (a) Definição dos objetivos da avaliação; (b) Definição de restrições da avaliação; (c) Definição do escopo da avaliação; (d) Definição das saídas da avaliação; (e) Definição dos membros da equipe de avaliação; (f) Definição dos participantes da avaliação; (g) Definição dos equipamentos e facilidades; (h) Definição do cronograma da avaliação; (i) Preparação dos participantes; (j) Administração dos instrumentos; (l) Planejamento da coleta de dados e (m) Documentação dos resultados da avaliação.

Os resultados obtidos da execução das atividades desta fase foram documentados no Plano da Avaliação que foi usado na execução da avaliação. Algumas das decisões documentadas no plano foram: (a) Todos os projetos a serem avaliados devem estar concluídos; (b) Três projetos: A, B e C, foram selecionados. Tais projetos receberam estes nomes para manter confidencialidade dos clientes. Os projetos B e C possuíam as mesmas características com relação à ambiente de desenvolvimento, ou seja, usaram mesma linguagem de programação, os projetos eram para o mesmo cliente, foram desenvolvidos em um mesmo laboratório e possuía uma equipe com um bom número de integrantes; (c) Os projetos A e B tiveram uma duração de 4 meses e o C teve duração de 6 meses.

As atividades que foram julgadas desnecessárias para o escopo da avaliação foram as seguintes: (a) No Processo 1.1 “Analisar requisitos”, atividade “Obtenção do comprometimento com as entradas da avaliação”; (b) No Processo 1.2 “Desenvolver Plano da Avaliação”, atividade “Planejamento e gerenciamento da

logística”; (c) Todas as atividades do Processo 1.3 “Seleção e Preparo Equipe”; (d) Todas as atividades do Processo 1.4 “Obtenção e Análise Evidências Objetivas Iniciais”; (e) No Processo 1.5 “Preparar para coleção de evidências objetivas”, atividades “Executar revisão rápida” e “Replanejar coleção de dados”.

4.2 Execução

Todos os processos da FPF foram avaliados e as atividades da avaliação seguiram de acordo com o plano elaborado para execução da mesma. No decorrer dessa seção serão mostrados exemplos de como ocorreu a avaliação e os resultados obtidos, porém como a avaliação foi feita para todos os processos, serão mostrados exemplos apenas de Garantia de Qualidade.

Primeiramente, foi feito o exame de evidências objetivas através de entrevistas.

Tais entrevistas foram realizadas com os membros das equipes dos projetos selecionados para avaliação e identificados no plano. Elas foram auxiliadas com o questionário elaborado.

Em seguida foi feito o exame de evidências objetivas através da busca de artefatos, verificação e validação das evidências objetivas, revisão das observações coletadas durante entrevistas e documentação das evidências objetivas.

Após validar os resultados com os membros das equipes dos projetos, as evidências objetivas presentes e ausentes foram documentadas. Além disso, para cada prática, seja específica ou genérica, estas evidências objetivas foram classificadas em AD: Artefato Direto, AI: Artefato Indireto, AF: Afirmação e AA: Artefato Ausente.

A tabela 1 abaixo mostra um exemplo dos artefatos diretos e indiretos, afirmações coletadas, e artefatos ausentes, identificados durante a avaliação do processo de Garantia de Qualidade, para os objetivos específicos e genéricos respectivamente.

Tabela 1 - Evidências Objetivas das Práticas Específicas

Práticas	Artefatos	Projeto A				Projeto B				Projeto C			
		AD	AI	AF	AA	AD	AI	AF	AA	AD	AI	AF	AA
SG1 SP1.1	Plano de GQ do Projeto	X				X				X			
	Resultado da Inspeção	X				X				X			
	Ferramenta de Controle de NC		X	X			X	X			X	X	
	Guia de Reporte de NC		X				X				X		
	Reporte de NC	X		X		X		X		X		X	
	Reporte de AC	X		X		X		X		X		X	
	Reporte de LA				X				X				X
SG1 SP1.2	Plano de GQ do Projeto	X				X				X			
	Resultado da Inspeção	X				X				X			
	Ferramenta de Controle de NC		X	X			X	X			X	X	
	Guia de Reporte de NC		X				X				X		
	Reporte de NC	X		X		X		X		X		X	
	Reporte de AC	X		X		X		X		X		X	
	Reporte de LA				X				X				X

- Pontos Fortes e Fracos**

Os pontos fortes foram levantados a partir da verificação da implementação das práticas, independente se ela era executada de forma eficiente ou não.

Os pontos fracos foram levantados a partir da identificação de artefatos ausentes, ou dos casos em que a prática era implementada, porém, de forma ineficiente ou inadequada. Exemplos de pontos fortes e fracos, para o processo de Garantia de Qualidade, podem ser visualizados na Tabela 2.

Tabela 2 - Pontos Fortes das Práticas Específicas e Genéricas**Caracterização da Implementação das Práticas:**

Uma vez levantadas todas as evidências objetivas, validados os resultados preliminares com os membros das equipes dos projetos e levantados os pontos fortes e fracos da área de processo, foi feita a caracterização da implementação das práticas do modelo de referência.

Exemplo de caracterização da implementação das práticas, para o processo de Garantia de Qualidade é mostrado na tabela 3.

Tabela 3 - Caracterização da Implementação das Práticas Específicas e Genéricas

Práticas Específicas e Genéricas	Projeto A	Projeto B	Projeto C	Resultado
SG 1: Avaliar objetivamente processos e produtos de trabalho				
SP 1.1-1: Avaliar objetivamente processos	LI	LI	LI	LI
SP 1.2-1: Avaliar objetivamente produtos e serviços de trabalho	LI	LI	LI	LI
SG 2: Prover visão objetiva				
SP 2.1-1: Comunicar e assegurar resolução das não-conformidades	TI	TI	TI	TI
SP 2.2-1: Estabelecer registros	TI	TI	TI	TI

Depois de finalizada a execução das atividades de caracterização do estado atual da empresa, todos os resultados obtidos foram documentados. Os resultados foram:

Evidências objetivas presentes e ausentes, pontos fortes e fracos, caracterização da implementação das práticas e oportunidades de melhorias.

As atividades desta fase, julgadas desnecessárias para o escopo da avaliação foram as seguintes: (a) No Processo 2.1 “Examinar Evidências Objetivas”, atividade “Exame de Evidências Objetivas de Apresentações”; (b) No Processo “Documentar Evidências Objetivas”, atividade “Revisão e atualização do plano de coleta de dados” e (c) No Processo “Gerar resultados da avaliação”, atividade “Determinação das áreas de satisfação do processo”.

4.3 Reporte de Resultados

A terceira fase do SCAMPI é o reporte de resultados. Nesta fase, os resultados da avaliação executada no processo da empresa foram apresentados à alta gerência e à alguns membros participantes do grupo de melhoria dos processos.

Foi feita uma apresentação contendo todos os pontos fortes e fracos e oportunidades de melhoria para todos os processos.

4.4 Oportunidades de melhoria

A definição das oportunidades de melhoria para o processo da Fundação Desembargador Paulo Feitosa foi feita a partir dos pontos fracos identificados no decorrer da avaliação.

Abaixo são apresentados alguns exemplos das oportunidades de melhorias identificadas para o processo da empresa, visando aderência ao CMMI:

- Gerência de Requisitos: Definir e documentar uma lista de critérios para distinguir bons provedores de requisitos, ou seja, os stakeholders relevantes ao projeto.
- Planejamento de Projetos: Criar uma atividade no processo para criação de um WBS baseado na arquitetura do produto.
- Acompanhamento de Projetos: Monitorar Custos adequadamente.
- Garantia de Qualidade: Identificar lições aprendidas para os processos e produtos desenvolvidos.
- Gerência de Configuração: O Plano de Gerência de Configuração deve ser acrescido da informação de quando cada item de configuração será colocado sob Gerência de Configuração.
- Medição e Análise e Acordo com Fornecedores: Definir e documentar os processos definidos para estas áreas.

5 Aplicabilidade dos Resultados

Para as empresas desenvolvedoras de software, a melhoria do seu processo de desenvolvimento é de grande importância, pois tal melhoria leva a um produto de qualidade. Este, por sua vez, faz com que as empresas alcancem ou permaneçam no topo do mercado de desenvolvimento de software.

Portanto, para auxiliar a Fundação Desembargador Paulo Feitosa a continuar neste mercado competitivo, este trabalho teve o intuito de realizar a primeira fase do processo de melhoria da empresa, que foi a identificação dos pontos fortes, fracos e oportunidades de melhorias de seu processo.

O alcance deste projeto, inicialmente, é de âmbito interno à empresa, pois avaliou o processo da empresa perante o nível 2 do CMMI, buscando adequação ao modelo e melhorias no processo que tragam benefícios em sua execução.

Posteriormente, caso essas melhorias sejam implantadas, haverá alcance em âmbito externo à empresa, pois a melhoria resultará num produto de maior qualidade, causando satisfação dos clientes com os produtos desenvolvidos.

6 Características Inovadoras

Esse projeto teve como inovação o início do processo de melhoria do processo de desenvolvimento de software da Fundação Desembargador Paulo Feitosa, na

tentativa de torná-lo aderente ao modelo CMMI em seu nível 2 de maturidade, que é um modelo importante para empresas desenvolvedoras de software para serem reconhecidas como empresas que desenvolvem softwares com qualidade.

7 Conclusão e Perspectivas Futuras

A realização da avaliação na Fundação Desembargador Paulo Feitoza teve como resultados, o levantamento de todas as evidências objetivas, a identificação de pontos fortes e fracos e a caracterização das práticas com relação à aderência ou não ao modelo CMMI, que foi o utilizado no trabalho. Além disso, recomendações de melhorias para o processo foram levantadas. Portanto, uma das contribuições do trabalho foi a de auxiliar no processo de melhoria que a empresa está tentando implantar para que o seu processo se adeque ao nível 2 de maturidade do modelo de referência CMMI.

Outra contribuição foi a adequação do método de avaliação SCAMPI. Este foi adaptado para que o mesmo refletisse as características da avaliação empresa em questão. A adaptação resultou na definição de um processo específico para a empresa, com remoção de algumas fases requeridas pelo modelo SCAMPI, e junção de várias fases em uma única, tornando-as mais simples. Com a execução do processo definido, pôde-se chegar muito mais rapidamente a conclusões sobre a adequação do processo de desenvolvimento às práticas do CMMI.

Como trabalho futuro, recomenda-se fazer a implantação das melhorias levantadas. A implantação de tais melhorias pode ser feita seguindo um modelo de melhoria, como por exemplo, o IDEAL, que possui fases pertinentes para dar continuidade ao trabalho. São elas: (a) Estabelecimento, que propõe o desenvolvimento de um plano detalhado, tendo como base os resultados da avaliação feita; (b) Ação, em que as atividades definidas no plano serão executadas; e (c) Aprendizado, em que todas as lições aprendidas do programa de melhoria serão levantadas e usadas nos próximos programas.

8 Referências

(CHRISSIS, 2003) CHRISSIS, Mary B., KONRAD, Shrum, Sandy, **CMMI®: Guidelines for Process Integration and Product Improvement**, Addison Wesley, 2003.

(CORRÊA, 2005) CORRÊA, George Marsicano, FIGUEIREDO, Rejane M. da Costa, OLIVEIRA, Kátia M. de, **Diagnóstico da área de Requisitos de Software usando IDEAL, CMMI e SCAMPI: Um Estudo de Caso**. Brasília, DF, SBQS 2005.

(SEI, 2001) SEI, **Standard CMMI Appraisal Method for Process Improvement (SCAMPISM) version 1.1: Method Definition Document**, 2001. Disponível em: <http://www.sei.cmu.edu/pub/documents/01.reports/pdf/01hb001.pdf>. Acesso em: 09 de Março de 2006.

(SOFTEX, 2006) SOFTEX, **MPS.BR – Melhoria de Processo do Software Brasileiro. Guia Geral versão 1.1**, 2006. Disponível em: http://www2.softex.br/media/MPS.BR_Guia.pdf. Acesso em: 25 de Maio de 2006.



(VIVEIROS, 2005) VIVEIROS, Sérgio M., TEIXEIRA, Cássio A. N., RAMOS, Jorge L. D., et al, **Estratégia para Melhoria de Processos com o CMMI e o MR mps Br no BNDS**. Brasília, DF, SBQS 2005.

(ZUBROW, 1994) ZUBROW, D., HAYES, W., SIEGEL, J., GOLDENSON, D., **Maturity Questionnaire Special Report, CMU/SEI-94-SR-7 – Empirical Methods**. Disponível em: <http://www.sei.cmu.edu/pub/documents/94.reports/pdf/sr07.94.pdf>. Acesso em: 15 de Abril de 2006.

[2.13] Consolidação do Processo de Manutenção de Software baseado na ISO 14764

Entidade: Atech - Fundação Tecnologias Aplicações Críticas - Rua do Rócio, 313, 11º andar - Vila Olímpia - São Paulo - SP - 04552-000 - Brasil

Autor: Claudia A. Tocantins - claudia@atech.br

Abstract. *This article describes and reports the results of the [2.13 – Consolidation of a Software Maintenance Process based on ISO 14764] project as proposed to the 2006 cycle of PBQP-SW.*

Resumo. *Esse artigo descreve e reporta os principais resultados do projeto [2.13 – Consolidação do Processo de Manutenção de Software baseado na ISO 14764] submetido ao ciclo 2006 do PBQP-SW.*

1. Introdução

O projeto descrito nesse artigo faz parte do programa de melhoria contínua do processo de software da Filial Manaus da Fundação Atech. Em 2006, foi dada atenção especial ao tópico manutenção de software de modo a integrar esforços da equipe de Engenharia de Software e de Prospeção de Negócios em atender a demanda de clientes por serviços específicos de manutenção.

O artigo reúne dados publicados na submissão do projeto e dados apresentados no EQPS Belém em novembro de 2006. A seção 2 indica os objetivos e justificativa do projeto; a seção 3 explica como o projeto foi desenvolvido; a seção 4 descreve o principal produto e resultados relevantes obtidos; a seção 5 explica a aplicabilidade dos resultados em termos de relevância, abrangência e impacto dos resultados obtidos; a seção 6 indica características inovadoras e a seção 7 conclui o artigo, apresentando perspectivas futuras.

2. Objetivos e Justificativa

O objetivo desse projeto foi consolidar o processo de manutenção de software vigente na organização, adequando-o as normas ISO 12207 e 14764, com o objetivo de incorporá-lo ao programa de melhoria contínua da qualidade e apoiar a prospecção de negócios específicos de manutenção de sistemas.

A organização tem se envolvido com projetos de diversos portes (projeto de poucos meses até vários anos) e nas várias etapas do processo de desenvolvimento, desde a concepção de sistemas inovadores até a implantação,

operação e manutenção evolutiva de sistemas e acumulado um conjunto de técnicas e melhores práticas. Consolidando-as e adequando-as a normas internacionais, o projeto visou evidenciar a experiência e capacitação da equipe na manutenção evolutiva dos sistemas e apoiar a prospecção de negócios de manutenção evolutiva, garantindo a seus clientes uma resposta confiável e perene na atualização de seus sistemas.

3. Metodologia de Execução

A elaboração do Processo de Manutenção se deu por em etapas, resumidas no seguinte fluxo:

- Análise e revisão dos processos existentes;
- Sessões de brainsorm com pontos focais da equipe de desenvolvimento;
- Elaboração da Definição do Processo usando a templates do processo organizacional;
- Estudo e análise de aderência com as normas;
- Adaptação do processo de acordo com as normas;
- Revisão e aprovação.

4. Resultados Obtidos

O principal produto gerado foi a definição do Processo de Manutenção, consolidando as práticas e técnicas vigentes na organização e sua implantação nos diversos projetos da organização. O processo proposto incluiu os requisitos da norma ISO 14764 e definiu o planejamento da manutenção, a análise de problemas, implementação de soluções, aceitação, migração e descontinuidade de sub-sistemas.

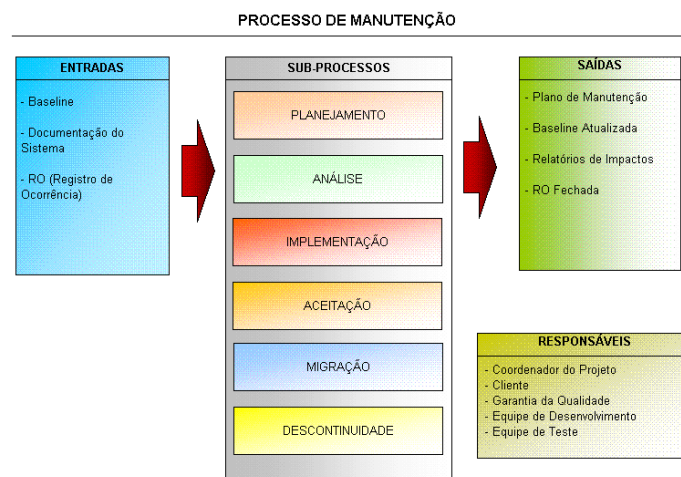


Figura 1 - Macro-fluxo do Processo

O processo definido tem como propósito modificar um sistema / produto de software após a sua implantação para: corrigir erros, melhorar a performance ou outros atributos, adaptá-lo a modificações na plataforma operacional. Ele incluiu instruções e formulários relacionados e a revisão e utilização de ferramentas de suporte para o Registro de Ocorrências (prevê acesso pelo cliente), o Registro de Problemas de Software. As ferramentas foram desenvolvidas através de customização sobre o Mantis e são adaptáveis conforme necessidade do projeto. As ferramentas estão disponíveis internamente a organização e em uso por projetos em andamento.

5. Aplicabilidade dos Resultados

Em termos de relevância, o projeto se insere na meta de adoção de métodos modernos de gestão da qualidade e produtividade em software por parte da organização. A definição do processo facilita, ainda, a capacitação das equipes envolvidas, uniformizando a qualidade dos serviços prestados e garantindo a longevidade e atualização dos sistemas do cliente.

Em termos de impacto, o Processo de Manutenção é parte do Programa de Melhoria do Processo de Qualidade da organização, tendo como finalidade a unificação de um processo único dentro da organização e eventualmente buscar uma certificação / avaliação em Qualidade de Software.

A abrangência do projeto é interna ao escopo da organização (Atech – Manaus) e foi utilizado em todos os projetos da filial Amazônia e tendo impacto direto na qualidade dos produtos já implantados nos clientes, garantindo uma resposta autônoma, confiável e perene no desenvolvimento, operação, manutenção e atualização tecnológica de sistemas inovadores que a organização tem fornecido.

6. Características Inovadoras

O Processo de Manutenção definido é uma consolidação / aprimoramento das técnicas e práticas vigentes na organização com adequação a normas de referência internacionais. A exigência da existência de processos consolidados é ainda prática pouco difundida no mercado brasileiro e o seu uso no apoio a prospecção de negócios é original.

7. Conclusão e Perspectivas Futuras

O Projeto “Consolidação do Processo de Manutenção de Software baseado na ISO 14764”, conforme submetido ao PBQP-SW, foi concluído com sucesso. Convém ressaltar, porém, que o Programa de Melhoria Contínua é “contínuo”, ou seja, sugestões de melhorias são identificadas com o uso e serão incorporadas.

Pode-se observar também que a formalização do Processo de Software apóia a capacitação de RH e também a prospecção de negócios. A equipe passa a ter melhor visibilidade do processo vigente e passa a ter um canal de contribuição formal para a melhoria do mesmo. O processo definido foi incluso em pelo menos uma prospecção relacionada a projeto de Manutenção de Software, onde o cliente solicitou aderência as normas ISO. Nessa ocasião, dúvidas e comentários do cliente foram discutidas e incorporadas. O projeto ainda não foi contratado.

Quanto a aderência a normas, nota-se que esse objetivo ajuda a identificar lacunas no processo vigente. A aderência a ISO 14764 resultou em inclusão de atividades não previstas e/ou não formalizadas anteriormente, como por exemplo os subprocessos de migração e descontinuidade.

[2.16] - Definição de uma Metodologia para Gerência de Manutenção de Software em uma Empresa do Pólo de Software Amazonsoft

Entidade: SOLTIN - Soluções Integradas em Tecnologia de Informações Ltda. - Av. Gen. Rodrigo Octávio, 1866, Sede do CIDE, sala: 23, São Lázaro, CEP: 69073-620 Manaus - AM - Brasil

Autor: Milka Romão Bezerra - milka_romao@soltin.com.br

1. Objetivos e Justificativas

A Soltin Ltda, é uma empresa de desenvolvimento de software participante do AmazonSoft. No ano de 2006 trabalhamos na definição e implantação do nível G do MPS.BR, e durante a fase de levantamento da situação, foi observado que, na maioria das vezes os softwares necessitariam passar por manutenção. Praticamente 80% dos programas tinham que ser revistos novamente, sofrendo dessa forma algum tipo de manutenção no código. Com isso surgiram muitas perguntas:

- Qual a razão de tantas manutenções?
- Onde está o problema, na especificação do programa? Na capacidade do programador?
- Qual o impacto gerado na manutenção?
- Como registrar o conhecimento adquirido na manutenção?

Após esse levantamento inicial e comparando as considerações acima, classificamos em problemas de manutenção de caráter gerencial: (i) Ausência de um processo de manutenção de software; (ii) Rotatividade de Equipe; (iii) Prazos não condizentes com a complexidade do software; (iv) Atrasos na entrega de versões; (v) Usuários (grande expectativa e/ou baixa comunicação). E problemas de caráter técnico: (i) Registro de Manutenções anteriores (Ausente/ Superficial); (ii) Documentação (Ausente/ Superficial); (iii) Validação insuficiente de manutenções efetuadas.

Uma maneira de se organizar a produção de software é utilizar-se de um processo de software. Este é importante, pelo fato de sistematizar as atividades desempenhadas pelas pessoas que lidam com sistemas de informação, permitindo a elas planejar e controlar as tarefas. Com isto, percebe-se a importância de se sistematizar e ordenar as atividades ligadas à manutenção de software. O processo de manutenção proposto na norma ISO 14764, oferece a opção de ser simplificado e adaptado a realidade de pequenas empresas de software.

Este projeto, propoem a definição de uma metodologia que será denominada MMSS - Metodologia de Manutenção de Software Soltin¹, objetivando a redução de tempo, esforço e custo, bem como gerar uma melhoria significativa na qualidade da manutenção do software desenvolvido, a partir das recomendações da ISO/IEC 12207 e 14764.

2. Metodologia de execução

Um processo de manutenção diz respeito a um conjunto de etapas bem definidas que direcionam a atividade de manutenção de software com o objetivo primordial de satisfazer as necessidades dos usuários de maneira planejada e controlada. Por esse caminho iniciou-se a pesquisa sobre a norma ISO/IEC 14764, que esclarece os requerimentos para o processo de manutenção de softwares, contendo também as atividades e tarefas do mantenedor. Essa Norma Internacional é parte da família de documentos ISO/IEC 12207.

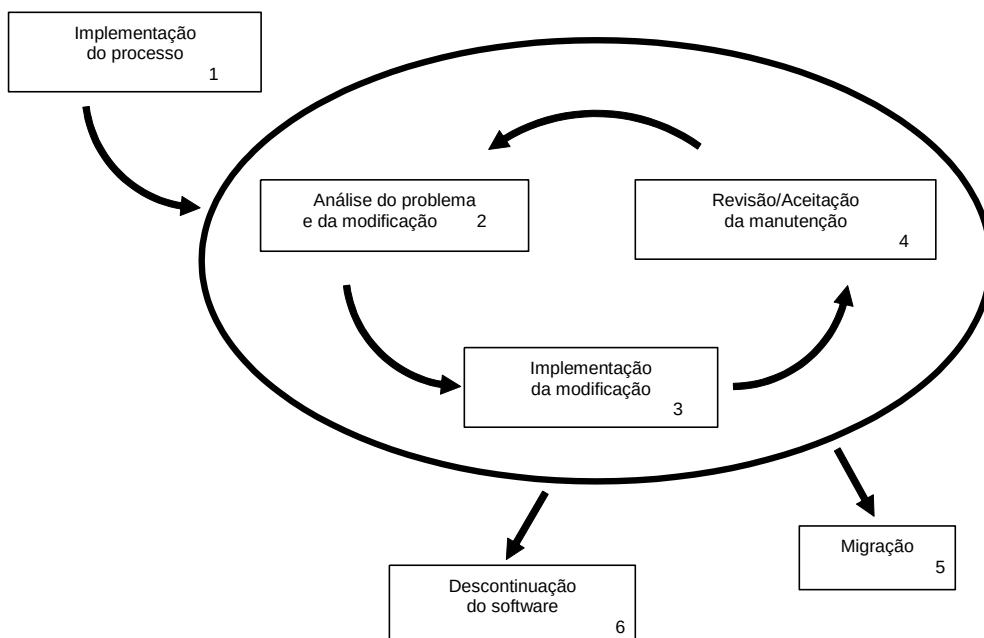


Figura 1: Processo de Manutenção ISO/IEC 14764. (retirado de ISO/IEC 14764, 1999)

3. Resultador relevantes

A criação da metodologia denominada MMSS – Metodologia de Manutenção de Software, envolve quatro processos, que estão resumidos na Figura 2, e interagem da seguinte forma:

¹ Soltin – Soluções Integradas em Tecnologia da Informação.

Na Fase I e II – queremos atender a **Implementação do processo (Figura 1 - ISO/IEC 14764)** onde, são estabelecidos planos e procedimentos para registrar e controlar a atividade de manutenção e os pedidos feitos pelos clientes.

Conforme descrito na Figura 2, a Gerência de Requisitos identifica uma necessidade de mudança de requisito, coletada pelo stakeholder relevante, através do registro de manutenção a partir de três formas distintas:

- (i) o próprio cliente solicita a manutenção;
- (ii) pelos consultores da empresa, após visita ao cliente e levantamento de necessidades de manutenção;
- (iii) pelos próprios desenvolvedores, que podem identificar necessidades de manutenção à medida que desenvolvem o software.

A partir da necessidade identificada, o documento de Solicitação de Mudança de Requisito (SMR) é preenchido, contendo informações sobre o projeto, o stakeholder responsável, o requisito a ser modificado, a caracterização da mudança (se foi sugerida pelo stakeholder final ou erro de especificação de requisito pelo analista de sistema). Em seguida é utilizada uma ferramenta de controle de rastreabilidade para preparar a Lista de Rastreabilidade do requisito. A Lista de Rastreabilidade contém informações sobre: (i) a quantos Objetos (produtos de trabalho) o requisito está relacionado; (ii) e quais desses Objetos são Itens de Configuração.

Fase II – descrito na Figura 2, a Gerência de Configuração recebe a SMR e a Lista de Rastreabilidade, coleta os Objetos relacionados na Lista de Rastreabilidade no repositório específico e preenche o documento de Especificação de Mudança de Requisito (EMR), contendo informações de todos os objetos a serem modificados e os entrega a Gerência de Projeto.

Na Fase III – atenderemos a **Análise do Problema e da Modificação (Figura 1)** Nesta etapa, o mantenedor fará: (i) Análise das Solicitações de Mudança MRs/PRs; (ii) repassa ou verifica o problema; (iii) Desenvolve opções para executar a modificação; (iv) Documenta as Solicitações de Mudança MRs/PRs, e os resultados das opções de implementação; (v) obtém aprovação para a opção selecionada da modificação (**ISO/IEC 14764**). Conforme descrito na Figura 2, a Gerência de Projeto recebe a EMR e os objetos para fazer a análise de impacto sobre o tempo, custo, risco e recursos necessários para realizar as mudanças. Depois disso, deve-se elaborar o plano de trabalho representado pelo documento Autorização de Mudança de Requisito (AMR), que contém a distribuição das atividades para as pessoas responsáveis e os seus respectivos Objetos. Esses documentos serão entregues para a Fábrica de Software.

Na Fase IV – será a **Implementação da Modificação**: Nesta etapa, são realizadas as tarefas propriamente ditas de alteração do produto de software, bem como de sua documentação. Nela, deve-se garantir a perfeita execução para se chegar à solução proposta, o mantenedor desenvolverá e testará a modificação (**ISO/IEC 14764**). Conforme descrito na Figura 2, a Fábrica de Software é responsável pela implementação da mudança. As atividades a serem executadas

pela Fábrica de Software serão monitoradas pela Gerência de Projeto para garantir a qualidade do produto.

A implementação deverá ser validada com as partes interessadas, e a partir dessa validação serão então atualizadas as versões entregues, seguindo o proposto na norma **Aceitação / Revisão da Modificação**: Nesta etapa, são feitos a homologação e aprovação junto ao solicitante para que o produto possa ser liberado, garantindo que o sistema está correto e de acordo com padrões aprovados usando a metodologia correta (**ISO/IEC 14764**).

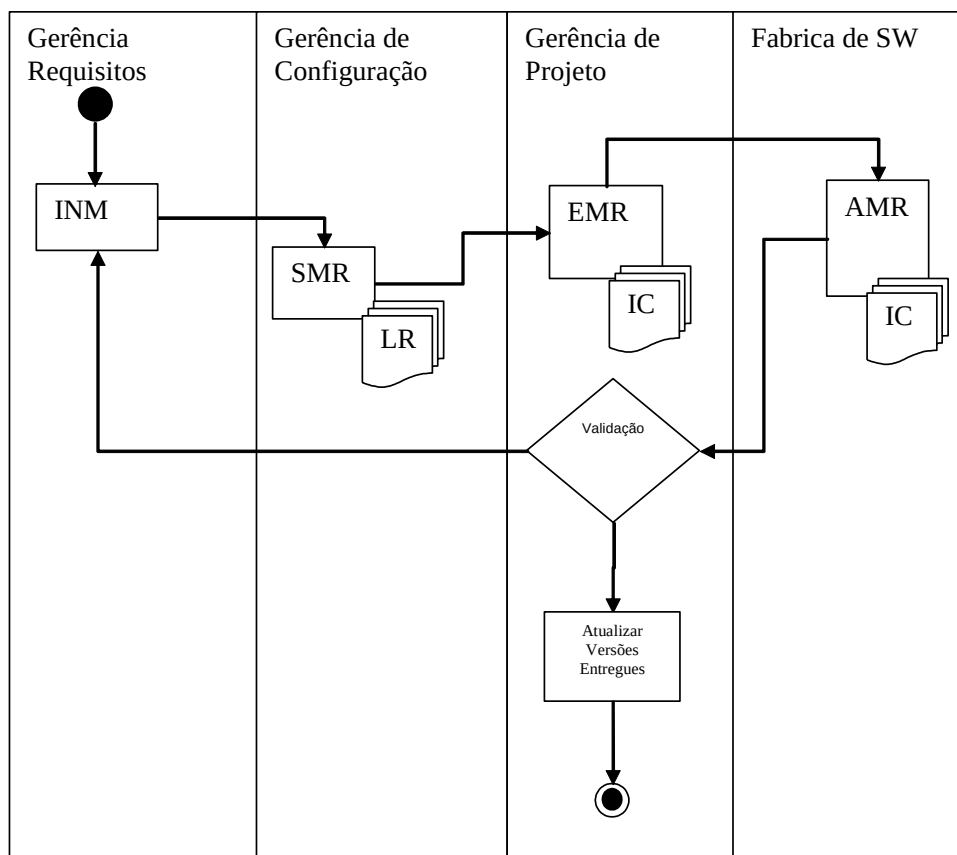


Figura 2 - Fluxo MMSS

4. Aplicabilidade dos resultados e principais impactos

Com a implementação dessa metodologia, ficou visível o quanto se fazia falta um processo de manutenção de software na empresa, nas etapas de treinamento sobre a metodologia já se obteve bons resultados demonstrado pelo interesse da equipe em padronizar um processo.

5. Conclusão e perspectivas futuras

As principais dificuldades durante a atividade de manutenção de software, são os problemas ligados a questões gerenciais, mais presentes do que os de caráter técnico. De fato, constatou-se que a manutenção de software e o que os usuários desejam não é uma tarefa simples de conciliar.

A metodologia proposta **MMSS** é uma evolução na busca de melhoria de processos pela empresa e a institucionalização será extremamente importante para alcançarmos maturidade e cumprirmos nossas metas e prazos em manutenção de software.

Durante esse projeto identificou-se também a necessidade do uso do processo de gerência de configuração para apoiar o processo de manutenção, principalmente no que se refere aos aspectos de rastreamento de requisitos e atualização de versões entregues aos clientes.

Como proposta para trabalhos futuros, sugere-se a definição de um processo de gerência de configuração para apoiar a metodologia proposta de manutenção. Esse trabalho vem agregar esforços em compreender melhor a atividade de manutenção de software, na metodologia proposta foi demonstrada a relação entre as áreas de: Gerência de Requisitos, Gerência de Configuração, Gerência de Projeto e Desenvolvimento, as boas práticas de cada área deverão contribuir para o sucesso e continuidade da **MMSS**.

6 Referências Bibliográficas pesquisadas

ISO/IEC 12207, (1995) “**Information Technology: software life cycle processes**”, International Standard Organization, Switzerland.

ISO/IEC FDIS 14764 (1999): “**Software Engineering- Software Maintenance**”. International Standard Organization, Switzerland.

Pressman, R.S. (2005) “**Software Engineering: a practitioner’s approach**”, 6.ed., McGrawHill Higher Education.

Rocha, a. r. c., Maldonado, j. c., Weber, K. C., “**Qualidade de Software: Teoria e Prática**”, São Paulo: Prentice Hall, 2001.

Souza, S.C.B. de; Neves, W.C.G. das; Anquetil, N.; Oliveira, K.M. de. (2004) “**Documentação Essencial para Manutenção de Software II**”, In: Anais do I Workshop de Manutenção de Software Moderna, Brasília, Brasil, Outubro.

[2.18] Elaboração de um Processo de Aquisição de Software Comerciais Prontos (COTS)

Departamento de Informática – Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI) - Av. Gov. Danilo de Matos Areosa, 381 - Distrito Industrial - CEP: 69075-351 - Manaus-AM / C. Postal: 631

Autores: Patrick O. Ramos¹, Miguel Grimm do Nascimento² {patrick.ramos, miguel}@fucapi.br

Abstract. *This Project describes the fundamentals concepts and characteristics of Software Products Acquisitions, applied to two methodologies: PMBOK, that describes the Acquisition Process's phases and activities for any common project; MPS-BR Acquisition Guide, that describes a Technology information's products specific acquisition process. Based on those concepts, this Project propose is the elaboration of an acquisition process, capable of handle specific aspects of COTS (Components out the shelves) products on a systematic and efficient way, and whose purpose is to obtain better quality products.*

Resumo. *Este trabalho descreve os conceitos e características importantes de aquisição de produtos de software aplicado em duas metodologias: PMBOK, que descreve as fases e suas respectivas atividades de um processo de aquisição voltado para projetos em geral; Guia de Aquisição MPS-BR, que descreve um processo de aquisição voltado para produtos de Tecnologia da Informação. Com base nesses conceitos, o trabalho propõe a elaboração de um processo de aquisição capaz de tratar de forma sistemática e eficiente aspectos específicos de produtos COTS, cuja finalidade é permitir a obtenção de produtos de maior qualidade.*

1 Introdução

Nos dias atuais o rápido e contínuo avanço tecnológico, a busca por soluções mais eficientes, as otimizações do tempo e de recursos estão cada vez mais presentes nas empresas, proporcionando constantes desafios e investimentos na definição e melhoria de seus processos.

A quantidade de softwares prontos disponíveis no mercado é um fator que dificulta na escolha de um software, muitas vezes induzindo a aquisição do produto inadequado.

Segundo (Capterra's, 2001), a causa de aquisição equivocada, deve-se ao fato do comprador não possuir habilidades suficientes realizar a avaliação e seleção do produto requerido.

Outro fator a ser considerado é a complexidade das tecnologias em decorrência de sua evolução.

Com isso, percebe-se que as empresas vêm investindo na definição e implantação de modelos metodológicos que procuram organizar seus processos.

Nesse contexto, identificou-se a necessidade de customizar um processo de aquisição de software voltado exclusivamente para COTS (Commercial-off-the-shelfsoftware).

Tal processo baseia-se no guia de aquisição MPS-BR e PMBOK.

O processo de aquisição definido visa auxiliar o comprador de software na avaliação e seleção do produto com características qualitativas afim de assegurar o atendimento de suas necessidades e expectativas.

2 Objetivo

O objetivo principal do projeto é elaborar um modelo sistemático em torno de um processo de aquisição voltado para produtos *COTS*.

2.1 Justificativa

A aplicação prática de uma metodologia que defina claramente as atividades necessárias que possibilite uma empresa a identificar, avaliar, selecionar e adquirir uma solução de software de produtos *COTS*.

3. Material e Método

O trabalho iniciou com o levantamento bibliográfico que visa à obtenção de conhecimentos conceituais sobre aquisição de *software*, para a formação de uma base científico-teórica no planejamento do projeto.

Para a definição do escopo de trabalho, identificaram-se os tipos de softwares existentes e suas classificações. A partir de então, definiu-se que o trabalho se limitaria ao tratamento das atividades pertinentes a compra de produtos *COTS*.

Com isso, foi realizado também um comparativo entre as metodologias *PMBOK* e Guia de Aquisição MPS-BR, sendo considerados os processos e atividades descritas em cada metodologia.

Em seguida, foram discutidos os requisitos mínimos que uma empresa deve ter para implantação da sistemática (processo de aquisição), tais requisitos foram obtidos através de reuniões das partes envolvidas no projeto.

Por fim, foi definido um processo de aquisição com suas respectivas atividades e produtos a serem gerados, cuja finalidade é auxiliar a avaliação e seleção de ferramenta que atendam as necessidades de uma empresa.

4. Embasamento Teórico

Um processo de aquisição é um guia geral que define as fases e atividades necessárias para aquisição de um software.

O *PMI* (2004), afirma que o processo planejar compras e aquisições identifica quais necessidades do projeto podem ser melhor atendidas pela compra ou aquisição de produtos, serviços ou resultados, direcionado a qualquer tipo de projeto.

Quadro 1. Processo de aquisição segundo a IEEE STD 1062:1998, descritos em (SOFTEX, 2005).

Fase	Início da fase	Fim da fase	Atividades a serem executadas
Planejamento	Identificação da necessidade	Pesquisa de mercado ou pedido de proposta.	1. Planejamento da estratégia organizacional 2. Implementação do processo organizacional 3. Definição dos requisitos do <i>software</i>
Contratação	Pesquisa de mercado ou pedido de proposta	Contrato assinado	4. Identificação dos potenciais fornecedores 5. Preparação dos requisitos do contrato 6. Avaliação das propostas e seleção do fornecedor
Desenvolvimento do <i>software</i>	Assinatura do contrato	Recebimento do <i>software</i>	7. Gerência do desempenho do fornecedor
Aceitação	Recebimento do <i>software</i>	Aceite do <i>software</i>	8. Aceitação do <i>software</i>
Acompanhamento	Aceitação do <i>software</i>	Encerramento do uso do <i>software</i>	9. Utilização do <i>software</i>

4.1 Classificação dos Produtos de Software

Os produtos de software são divididos em três classes: *Commercial-off-the-shelf software (COTS)*, *Modified-off-the-shelf software (MOTS)* e *Fully Developed Software (FD)*, conforme exposto em SOFTEX (2005).

_ **COTS**: refere-se aos produtos que estão comercialmente disponíveis, ao alcance de qualquer empresa e ou pessoa. É normalmente bem definido e documentado e usado por um grande número de usuários. O fornecedor não está disponível para modificar o software às necessidades de um cliente específico e nem para controlar a manutenção do software.

_ **MOTS**: software de prateleira modificável, o cliente não tem controle sob a qualidade de suas características, é parecido com o software do tipo COTS, com a diferença de que o fornecedor está disponível para efetuar modificações das funcionalidades do produto de software, segundo os requisitos do cliente.

_ **FD**: software desenvolvido sob encomenda (*fully developed software*) é único, tem um volume baixo de aplicação e é desenvolvido para atender completamente os requisitos de um cliente específico.

Nesse contexto, um processo de aquisição pode ser modelado de maneira a facilitar o fluxo das atividades de acordo com a classificação de cada software, conforme ilustrado na figura 1.



Figura 1. Decomposição do processo de aquisição.

A importância da decomposição do processo deve-se ao fato da independência das atividades relacionadas a cada software de acordo com sua classificação.

5 Resultados Obtidos

O modelo definido concentra as informações necessárias para atender em particular as atividades essenciais para aquisição de software de prateleira (COTS).

O processo de aquisição foi dividido em três fases, Identificação e Preparação da Aquisição, Avaliação e Seleção do Fornecedor e Aceitação do Fornecimento, conforme ilustrado na figura 2.



Riscos	Atividade de gerenciamento
Distribuição de novas versões pelo fornecedor	✓ pesquisa de mercado ✓ análise de impacto atualizada
Orçamento	✓ controle de artefatos ✓ planejamento e cálculo de custos
Garantia de segurança	✓ controle de qualidade ✓ apoio logístico
Qualidade do sistema	✓ controle de qualidade
Controle de configuração	✓ controle de artefatos ✓ apoio logístico

As atividades do processo precisam ser executadas por pessoas que, por sua vez, possuem um papel importante no sucesso de uma aquisição. Para tanto, os responsáveis pela interação com o processo são:

_ **Líder de equipe:** Responsável por identificar as necessidades de se adquirir um produto.

_ **Usuário:** Usuário simples que utiliza o *software* adquirido para realização de uma determinada tarefa.

_ **Equipe técnica:** Responsável pela identificação e análise dos requisitos tecnológicos e realizar a instalação, configuração, atualização e registro do produto.

_ **Equipe de aquisição:** Responsável pela análise financeira, e identificação e convite a possíveis fornecedores e contratação de fornecedores.

_ **Fornecedor:** Responsável pelo envio da proposta de fornecimento, e entrega do produto e realização dos serviços associados (suporte técnico e operacional).

5.1 Produto Gerado

Neste contexto, o produto gerado nesse projeto trata-se de um *workflow* para aquisição de COTS, ou seja, uma sequência de atividades com regras definidas e pessoas responsáveis por cada atividade em prol de um objetivo único. O modelo está ilustrado na figura 3.

Além disso, é importante destacar que a aquisição de um software com a qualidade desejada e que atenda as expectativas geradas, está diretamente relacionada à qualidade do processo utilizado.



Figura 3. Fluxo do processo de aquisição de COTS.

Contudo, pode-se afirmar que o processo de aquisição modelado neste trabalho possui os principais artefatos aplicados em qualquer processo organizacional. Tais artefatos são: fases, atividades, atores e um fluxo.

5.2 Recursos Humanos

A equipe responsável pela definição do processo de aquisição voltado para produtos COTS é formada por um mestre com alta qualificação e experiência em definição de processos e gerenciamento de pessoal e um agora especialista em Engenharia de Software com experiência em análise e desenvolvimento de sistemas. Tais envolvidos são respectivamente os seguintes:

- _ Miguel Grimm do Nascimento: (Orientador do trabalho)
- _ Patrick Oliveira Ramos (Orientando do trabalho)

5.3 Dissertações e/ou teses geradas

A partir desse projeto foi possível realizar o trabalho de conclusão do curso de Especialização em Engenharia de Software realizado no Centro Universitário Nilton Lins no Estado do Amazonas no período de outubro/2004 a dezembro/2005.

6. Aplicabilidade dos Resultados

O processo de aquisição definido neste trabalho, foi disponibilizado para uso institucional, no qual tornou-se parte de um PO (Procedimento Operacional – Aquisição de TI) na empresa.

No entanto, para disponibilizar o processo no mercado, é preciso que alguns pontos sejam concluídos e vivenciados, com intuito de obter resultados que demonstrem a eficácia do processo, tais pontos são: Validação dos *templates* (documentos) que serão utilizados na execução de cada atividade do processo; Validação do processo e suas respectivas atividades por seus usuários, a fim de extrair dados suficientes que comprovam a eficiência do processo.

7. Características Inovadoras

A característica principal do modelo proposto é a combinação das melhores praticas a respeito de aquisição de software e serviços correlatos existentes tanto MPS-BR e no *PMBOK* (versão 2004) sintetizadas em torno de um processo de aquisição de software específico para produtos COTS.

8. Conclusão e Perspectivas Futuras

O resultado deste trabalho é identificado com a definição de um processo de aquisição composto por três fases: Identificação e preparação da aquisição; Avaliação e Seleção do Fornecedor; Aceitação do Fornecimento.

A intenção deste trabalho é fornecer artefatos que auxiliam na melhoria dos produtos de *software* a serem adquiridos.

Considerando o rápido e contínuo avanço tecnológico aonde a todo instante, novos padrões vêm sendo definidos e produtos disponibilizados no mercado prometendo soluções aos problemas existentes, compreende-se que o diferencial seja estimulado pela inovação tecnológica e melhoria contínua dos processos organizacionais.

De posse Considerando os objetivos alcançados, e com intuito de obter melhorias futuras recomenda-se:

_ Aprofundamento, pesquisa e definição de processos de aquisição específicos para produtos *Modified-off-the-shelf-software* (MOTS) e *Fully Developed Software* (FD).

_ Definição, implantação de uma ferramenta que automatize e auxilie no uso do processo de aquisição.

8. Referências Bibliográficas

PMI - PROJECT MANAGEMENT INSTITUTE -, Um Guia do Conjunto de Conhecimentos em Gerenciamento de Projetos. Guia PMBOK®. Terceira edição; 2004.

SOFTEX. MPS.BR - Melhoria de Processo do Software Brasileiro. Guia de Aquisição (Versão 1.0). Maio de 2005.

MARTINS, Eliane. Processo de Manutenção (COTS). Instituto de Computação – UNICAMP. Setembro de 2002.

[IEEE 1062] - IEEE COMPUTER SOCIETY. **IEEE** - *Software Engineering Standards Collection. IEEE STD 1062 - IEEE Recommended Practice for Software Acquisition*. New York, NY. 1998a. 43p.

CAPTERRA'S. Software Selection Methodology. The Enterprise Software Center. março de 2001.

8.1 Outras Fontes Consultadas

PRESSMAN, R. S. Engenharia de Software Ed. 5, Rio de Janeiro – RJ, McGraw-Hill, 2002.

CMU/SEI-2000-TR-019, CMMi for Systems Engineering / Software Engineering, Version 102, SEI, Carnegie University Technical Report, 2000

[2.19] Proposta de Utilização da Mcda-C na Avaliação e Melhoria de Processos de Software

Entidade: Universidade Federal de Santa Catarina - UFSC

Autores: Leonardo Ensslin (UFSC) ensslin@deps.ufsc.br1, Luiz Carlos Mesquita Scheid (UFSC) lcmscheid@deps.ufsc.br2

Resumo

A avaliação e melhoria de processos de software (SPA e SPI) têm recebido atenção especial desde a década de 80. Vários modelos têm sido criados entre eles CMM, ISSO 9000, ISO 15.504, BOOTSTRAP. Estes modelos apóiam-se no paradigma racionalista.

Eles apresentam sugestão das áreas de processo a serem avaliadas. Quais práticas devem ser implantadas e institucionalizadas. Como e quem pode realizar a avaliação e certificação. A participação dos decisores de uma organização de software limita-se a escolher qual o modelo a adotar, ter em conta suas particularidades. A questão é que, a adoção destes modelos tem encontrada pouca receptividade pelos usuários. Quando se pesquisa a literatura sobre o assunto, encontram-se trabalhos que propõem alternativas para diminuir as fraquezas dos modelos. A proposta deste trabalho é a de apresentar a MCDA-C como metodologia para aperfeiçoar o SPA e SPI. A MCDA-C é baseada em paradigma construtivista. O modelo de avaliação e melhoria será construído com os decisores das organizações, partindo de suas preocupações e do contexto em que estão inseridos. No final o modelo construído representa a percepção dos decisores da organização de software, e é ajustado a sua realidade.

Palavras chave: avaliação de processos de software, melhoria de processos de software, metodologia multicritério de apoio a decisão, CMMI e SPICE.

1. Introdução

A avaliação e melhoria de processos de software (SPI3 e SPA4) têm recebido atenção especial dos governos, pesquisadores e da indústria (WANG *et al*, 1997 p.1).

Trabalhos publicados atestam a economia promovida pela melhoria da qualidade do software (PITTERMAN, 2000 p.96). Desde a década de 80 vários modelos foram desenvolvidos com este intuito. Os mais utilizados pelas organizações de software são o CMM5, recentemente substituído pelo CMMI6 e o SPICE7 (KUILBOER e ASHRAFI, 2000 p.28).

Os modelos propostos são baseados em um conjunto de atividades base de processos (BPA8). Estas atividades são pré-definidas pelos modelos (WANG *et al apud* WANG *et al*, 1998). Seguindo um paradigma racionalista os modelos determinam quais processos à organização deve ter. Sugerem a ordem em que as áreas de processo devem ser avaliadas e melhoradas quanto a sua performance. Finalmente, como julgar o estágio atual de capacidade de um processo ou adoção de uma prática.

Este trabalho apresenta o uso da metodologia Multicritério de Apoio a Decisão (MCDA), uma alternativa às organizações de software, para adoção em avaliação e melhoria de processos, através da opção pelo paradigma construtivista, que considera, entre outros aspectos, que o fluxo do processo decisório não é necessariamente predeterminado ou organizado de maneira lógica e como tal cada organização terá seu processo particularizado (ENSSLIN *et al*, 2001 p.17).

2. Objetivos e Justificativa

As diferenças entre o software e outras áreas da engenharia fazem com que a melhoria da qualidade a ser obtida em um produto de software seja alcançada através da avaliação do processo de software e, conseqüentemente, das sucessivas mudanças introduzidas no processo com o intuito de melhorar a performance.

Em meados dos anos 80, nos Estados Unidos, foi criado o Software Engineering Institute (SEI) na Universidade Carnegie-Mellow em Pittsburg. Desde então o SEI tem produzido diversos e significativos trabalhos baseados nos paradigmas da qualidade, centrados nos processos, propostos por Deming (1986) e Juran (1988), e na idéia de 3 maturidade de Crosby (1980). Em 1992, o Internacional Standards Organizations (ISO) estabeleceu um grupo para a elaboração dos padrões para avaliação de processos de software baseado no conceito de que a qualidade do produto é altamente dependente do processo utilizado para a sua criação. Recentemente em 2003, foi criado o projeto MPS.BR (SOFTEX, 2007) com apoio do Ministério da Ciência e Tecnologia (MCT) e do FINEP com objetivo de apoiar a melhoria do processo de software brasileiro.

Apesar da importância e interesse destes modelos, quando se procura estatística a respeito da quantidade de organizações de software que os tenham adotado, constata-se que poucos o fizeram. O relatório CMMI V1.1 – SCAMPI9 V1.1 (SEI, 2007), que periodicamente apresenta resultados de avaliações do CMMI pelo SEI,

mostra que 840 empresas participaram de avaliações oficial Classe “A” do SEI entre abril de 2002 e junho de 2006, em todo o mundo. No Brasil, a última pesquisa realizada pela Secretaria de Política de Informática (SEPIN) do Ministério de Ciência em Tecnologia sobre a Qualidade e Produtividade no Setor de Software, publicada em 2001 (BRASIL, 2005), mostra que 415 empresas responderam sobre o uso do SEI-CMM, destas, apenas 16 conhecem e usam sistematicamente o modelo. Quanto à utilização da ISO/IEC 15.504, das 409 empresas que responderam, apenas quatro informaram que conhecem e usam sistematicamente o modelo. Já o projeto MPS.BR (SOFTEX, 2007), segundo informações divulgadas em seu site, apresenta 13 empresas como avaliadas desde sua criação.

Ao procurar entender o problema da não adoção pelas organizações de software de um destes modelos, Gray e Smith (1998) publicaram um trabalho onde relatam algumas questões, de forma crítica, que explicam alguns motivos da fraca adesão. Segundo os autores os modelos possuem:

- Natureza “sintética” – eles foram criados com base em opiniões e asserções não testadas, assumindo que o problema da qualidade está no gerenciamento.
- Dificuldades na repetição e reprodução de resultados de avaliação – Não há certeza de que dois grupos diferentes de auditores ou avaliadores irão obter o mesmo resultado ao avaliar o mesmo processo de software.
- Problemas na sua validação – Era de se esperar que a melhoria da qualidade dos produtos de software deveria ser alcançada à medida que os níveis de maturidade fossem aumentando. Para os autores, há evidências nos Estados Unidos de organizações avaliadas em altos níveis de maturidade que produzem software com qualidade pobre.
- Questionários para avaliação do processo – Questionários pré-elaborados para validar o processo de software possuem problemas de semântica em questões que levam a interpretações diferentes. Uma mesma questão pode ter diferentes significados para vários auditores.
- Questionável impacto e relevância quanto a melhoria alcançada com a avaliação do processo de software – Os resultados obtidos até agora pelas organizações de software que utilizaram a abordagem de avaliação do processo para obtenção futura de melhorias são criticadas. Há indícios de que os planos de melhoria não são muito bem compreendidos pelos atores envolvidos.
- Dificuldades em demonstrar a economia gerada e o custo efetivo da avaliação do processo – Os indicadores de performance existentes atualmente, tais como, retorno sobre investimento (ROI), redução da duração do ciclo de vida do projeto, quantidade de erros, entre outros, ignoram os benefícios intangíveis gerados, dificultando qualquer análise de custo e benefício da utilização desta abordagem.

- Fraca estrutura científica – Esta questão é a raiz de todos os problemas acima mencionados. Não existe um processo científico onde os problemas são identificados em apenas um domínio empírico; as hipóteses não são formadas sobre como os problemas podem ser explanados e resolvidos e, finalmente, não são testadas em laboratório e em condições reais.

Motivado pela importância do assunto, avaliação e melhoria do processo de software, e pela baixa adesão das organizações de software aos modelos existentes, este trabalho de pesquisa foi desenvolvido para encontrar na Engenharia de Produção, dado à experiência existente em vários tipos de indústrias, ferramentas para a solução deste tipo de problema. Entre as ferramentas existentes, foi adotada uma abordagem que utiliza o paradigma construtivista, onde o processo a ser avaliado e melhorado inicialmente é compreendido em forma personalizada pelos atores envolvidos. Em seguida, as preocupações e os critérios utilizados são estruturados e explicitados por meio de escalas e finalmente é desenvolvido um modelo para avaliação propriamente dito que mensura o desempenho do processo de software em cada um dos aspectos julgados relevantes. A Metodologia Multicritério de Apoio à Decisão Construtivista (MCDA-C), já utilizada em várias áreas da gerência das organizações, é agora proposta como ferramenta para subsidiar o SPI e SPA.

3. Metodologia de Execução

Este trabalho de pesquisa iniciou-se com uma revisão bibliográfica dos principais modelos para avaliação e melhoria de processos de software e da metodologia multicritério de apoio à decisão. Em seguida, um estudo de caso foi realizado em uma organização de software onde um modelo de avaliação do processo de software foi construído, segundo a percepção dos atores envolvidos (diretoria, gerência e profissionais de tecnologia da informação) que obteve como resultado as conseqüentes melhorias propostas e seus indicadores de performance, utilizando como ferramenta a metodologia multicritério de apoio à decisão. A seguir, dado a natureza deste artigo, será apresentado um resumo deste estudo de caso. Para leitura da versão completa do trabalho de pesquisa consulte Scheid (2005).

Estudo de Caso

Em modelos que seguem o paradigma racionalista e monocritério (p.ex., CMMI, SPICE e MPS.BR) busca-se o aperfeiçoamento de uma função baseando-se em um critério único, objeto da avaliação. Para o decisor de uma organização de software nem sempre o ganho da melhoria pode ser percebido em tais modelos, pois o seu problema, o processo a ser melhorado, possui múltiplas dimensões (ENSSLIN *et al*, 2001 p.45). O estudo de caso realizado e aqui apresentado para melhoria do processo de software usando as ferramentas da MCDA-C, teve as seguintes etapas: (1) a definição dos atores, (2) definição do problema, (3) definição dos elementos primários de avaliação (EPA) e dos conceitos, (4) construção da hierarquia de conceitos, (5) árvore hierárquica de candidatos a pontos de vistas - aspectos relevantes, (6) construção dos descritores - escalas e funções de valor, (7) determinação das taxas de substituição, e (8) análise de sensibilidade da robustez do modelo.

Definição dos Atores

Na MCDA, para início dos trabalhos de SPI e SPA, deve-se definir o subsistema de atores (imprescindível), composto dos agidos (pessoas envolvidas de maneira passiva), dos decisores (pessoas que possuem a responsabilidade máxima pelas decisões tomadas, para este estudo de caso, o Diretor da Fábrica de Software), dos representantes (pessoas que representam o decisor na sua falta, neste caso, dois Gerentes de Projeto) e do facilitador, neste caso, o autor deste artigo (ENSSLIN *et al*, 2001 p.18).

Definição do Problema

No caso do problema SPI e SPA em organizações de software, o rótulo do problema será melhorar continuamente o processo (ou um conjunto de processos) de software através de avaliações, que podem ser passíveis (por exemplo, CMMI modelo por estágio) ou não de benchmarking (por exemplo, SPICE e CMMI modelo contínuo).

Com a MCDA-C estaremos no segundo caso, não passível de benchmarking, pois para o paradigma construtivista o problema pertence a uma pessoa (EDEN *et al apud* ENSSLIN *et al*, 2001 p. 23) dentro de um determinado contexto.

Definição dos Elementos Primários de Avaliação e Construção dos Conceitos

Na MCDA-C as oportunidades de melhoria são construídas para promover o aperfeiçoamento dos aspectos relevantes e estes usualmente não são conhecidos a priori. A MCDA-C utiliza-se de uma técnica para tal que inicia com a identificação pelos atores do contexto de decisão, segue-se a identificação dos elementos primários de avaliação (EPA). Os EPAs são identificados em reuniões onde, de forma livre, os atores mencionam seus valores, preocupações, problemas, ações, tudo relacionado aos processos que se deseja melhorar e ao conjunto de aspectos que, estará sendo considerado, ao se examinar o produto final (ENSSLIN *et al*, 2001 p.79). Neste artigo será utilizado, como exemplo, uma preocupação em que os atores da área da organização de software onde o estudo de caso foi realizado, teriam com a definição de tecnologia a ser utilizada no desenvolvimento de um software: “os problemas que a troca constante de tecnologia pode gerar”. Para este exemplo, em uma reunião de “brainstorming” os decisores mencionaram como EPAs:

- mesma tecnologia
- tecnologia ditada por terceiros
- prazo de definição da tecnologia

Outros 70 EPAs foram identificados Scheid (2005).

Em seguida, com o propósito de melhorar o entendimento de cada EPA, foram construídos conceitos orientando a ação a partir de cada EPA (ENSSLIN *et al*,

2001 p.81) e associando-os a um oposto psicológico separados por três pontos (...) que se lê “ao invés de”. Os EPAs acima geraram assim os três conceitos:

- 1) possuir projetos com mesma tecnologia ... cada projeto com tecnologia diferente;
- 2) definir a tecnologia para o projeto ... permitir que seja ditada por terceiros;
- 3) definir a tecnologia com antecedência ... ser definida somente no início do projeto;

Construção da Hierarquia de Conceitos

Para integrar os conceitos, e eventualmente identificar conceitos faltantes ou redundantes, é construído um mapa de relações meios-fins (mapas cognitivos) (ENSSLIN *et al*, 2001 p.82). A figura 1 apresenta um exemplo a partir dos conceitos construídos para este exemplo. O topo do mapa apresenta a área de preocupação *definição de tecnologia*. Os conceitos originais apresentam-se numerados e são colocados na parte central do mapa. Os demais são conceitos construídos em direção aos meios e fins.

Os agrupamentos de nós com ligações mais fortes, chamados intra-componentes (ENSSLIN *et al*, 2001 p.115), formam os cluster. No caso do exemplo da figura 1, dois clusters podem ser identificados.

Árvore de candidatos a ponto de vista.

O mapa cognitivo construído a partir de cada uma das áreas de preocupação dos decisores possui um conjunto de candidatos a pontos de vista fundamentais (PVF). Os PVFs representam as dimensões que os decisores julgam importantes para o contexto.

(ENSSLIN *et al*, 2001 p.139). Estes PVFs serão representados em estruturas arborescentes. A figura 2 apresenta a estrutura derivada do exemplo anterior. O ponto de vista global *Definição da tecnologia* é decomposto em duas áreas de interesse, *padrão de tecnologia* e *tempo de definição da tecnologia*. A área de interesse *padrão de tecnologia*, como exemplo, possui os candidatos a PVF *difundir tecnologia usada internamente* e *conceito da área de tecnologia*. Com isto a metodologia MCDA-C construiu, segundo a percepção do decisor, portanto em forma personalizada, as dimensões que deverão ser utilizadas para avaliar a performance do processo de software. Os passos seguintes irão, sempre em forma construtivista, construir as escalas para mensurar o desempenho dos PVFs, integrar as dimensões, construir o modelo global de avaliação, identificar o desempenho do status quo e identificar oportunidades de melhoria.

Construção dos descritores (escalas ordinais) e funções de valor (escalas cardinais)

Nos modelos CMMI e SPICE a avaliação deve ser feita por consultores especialistas credenciados pelo SEI ou pela ISO, respectivamente. Este fato ocorre devido à subjetividade do julgamento (ausência de instrumentos analíticos explícitos) a ser realizado na avaliação. Na metodologia MCDA-C é construído um modelo multicritério para avaliação das ações potenciais (melhorias). Duas

ferramentas serão utilizadas: descritores e funções de valor. O descritor serve para dar um melhor entendimento da preocupação do decisor (instrumento ordinal) e a função de valor (instrumento cardinal) mede a diferença de atratividade entre os diversos níveis do descritor (ENSSLIN *et al*, 2001 p.145). O descritor deve ser mensurável, operacional (facilidade de definição e coleta dos dados a serem medidos), e compreensível (KENNEY *apud* ENSSLIN *et al*, 2001 p.160).

A figura 3 apresenta um exemplo, da construção dos descritores para medir a performance do candidato a PVF *conceito da área de tecnologia*.

Para medir o desempenho do candidato a PVF *conceito da área de tecnologia*, os decisores, como mostra o exemplo da figura 3, foram construídos: (a) o número de trabalhos publicados por ano na mídia especializada e (b) número de apresentações em eventos na área de TI por ano. Os níveis de impacto estão apresentados na escala de valor. Por exemplo, trabalhos publicados foram definidos os seguintes níveis de impacto: 1 (um) trabalho publicado, 3 (três) trabalhos publicados, 5 (cinco) trabalhos publicados, 7 (sete) trabalhos publicados, e 9 (nove) trabalhos publicados.

Existem vários métodos para a construção de funções de valor. Neste trabalho foi utilizado o método de julgamento semântico (BANA E COSTA E VANSNICK *apud* ENSSLIN *et al*, 2001 p.195) MACBETH10. O MACBETH faz uso de um julgamento de diferença de atratividade entre duas ações potenciais (ver a função de valor obtida para o descritor *trabalhos publicados*, apresentada no quadro

Taxas de substituição

Quando submetemos uma ação potencial à avaliação em um modelo multicritério, raramente ocorre de ela ser a melhor com relação a todos os critérios (dimensões) analisados (ENSSLIN *et al*, 2001 p.218). Tornando difícil a tarefa de identificar em uma forma fundamentada a mais atrativa. Emergem então os modelos compensatórios visando integrar as várias dimensões em uma única medida sem, no entanto, descaracterizar o modelo multicritério. A taxa de substituição é a forma de agregar estas dimensões da avaliação. Uma ação potencial dita aqui pode ser o “status quo” do processo ou uma ação a ser avaliada. Esta avaliação irá representar ao decisor o quanto de melhoria se espera que será obtida após implantação da ação. A figura 4 mostra as taxas de substituição obtidas através do método de comparação par-a-par. A avaliação global de uma ação potencial “status quo” foi então calculada. Como exemplo de uma avaliação global o quadro 2 apresenta resultados baseados nos PVFs da figura 4. Para esse exemplo, segundo o julgamento do decisor, a ação potencial dos processos referentes à *definição da tecnologia* alcançou o valor global de 28,7.

Análise de sensibilidade

A última etapa do método MCDA-C serve para verificar a robustez das respostas do modelo frente a alterações nos parâmetros do mesmo. Um estudo detalhado pode ser obtido em ENSSLIN *et al* (2001).

4. Resultados Obtidos

Como resultado deste trabalho de pesquisa obteve-se a dissertação de mestrado:

□ Avaliação em melhoria de processo de software: Uma abordagem utilizando a Metodologia Multicritério de Apoio à Decisão – Construtivista. Título de Mestre concedido ao autor deste artigo (Luiz Carlos Mesquita Scheid) pelo Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina.

5. Aplicabilidade dos Resultados

A ferramenta MCDA-C possibilita aos decisores das organizações de software realizarem os trabalhos de melhoria continua do processo de software rumo a uma certificação de mercado (p.ex., CMMI, ISO ou MPS.BR) usando um modelo de avaliação que retrata suas preocupações imediatas e que permite o acompanhamento da evolução do nível de maturidade da organização através de uma escala cardinal.

6. Características Inovadoras

Apesar das ferramentas da MCDA-C serem utilizadas em diversas áreas das organizações, especificamente na melhoria de processos de software é inédito.

Contrapondo os modelos tradicionais apoiados no paradigma racionalista, a MCDA-C permite ao decisor a construção de um modelo que representa seus valores a cerca do problema existente no contexto da sua organização de software. Este modelo é único e apenas tem sentido enquanto os dois sujeitos (atores e contexto) estiverem presentes.

7. Conclusão e Perspectivas Futuras

Apesar do interesse os modelos tradicionais (CMMI, ISO e MPS.BR) não tem sido adotados em grande escala. A MCDA-C apresentada como uma alternativa para subsidiar o SPI e SPA que tem como vantagem apoiar-se em um paradigma construtivista. Nesta metodologia, o problema é estruturado junto com os atores e considerando as preocupações do contexto. Desde modo, os resultados de melhoria terão em conta as particularidades do contexto. Ao invés dos atores do processo priorizarem qual BPA deveriam trabalhar primeiro, através de um processo interativo, irão estruturar o problema segundo a suas percepções e com isto fazer com que o modelo seja personalizado ao decisor naquele contexto. Ademais o uso da metodologia MCDA-C permite construir um processo gerador de melhorias sem limites.

Como recomendação, quando uma área de preocupação coincidir com uma BPA existente em um dos modelos, CMMI ou SPICE, as práticas sugeridas pelos modelos deveriam ser utilizadas como exemplo, já que se trata de modelos amplamente estudados.

8. Referências Bibliográficas

BANA E COSTA, C.; DE CORTE, J.; VANSNICK, J. **Macbeth**. London School of Economics and Political Science, Work Paper, LSEOR 03.56, 2003.

BRASIL. **Secretaria de Política de Informática – SEPIN. Ministério de Ciência e Tecnologia.** Pesquisa Qualidade e Produtividade no Setor de Software 2001. Disponível em <http://www.mct.gov.br/Temas/info/Dsi/Quali2001/Public2001.htm>. Acesso em: 16 de Junho. 2005.

CROSBY, P. **Quality is Free: The Art of Making Quality Certain.** Mentor Books, New York, NY, 1980.

DEMING, W. **Out of the Crisis.** MIT Press, Cambridge, Massachusetts, 1986.

EDEN, C. **Analyzing cognitive maps to help structure issues or problems.** European Journal of Operational Research, No.159, Pág.: 673–686, 2004.

ENSSLIN, L.; MONTIBELLER NETO G.; NORONHA, S. **Apoio à Decisão:** Metodologias para Estruturação de Problemas e Avaliação Multicritério de Alternativas.

INSULAR, LabMCDA, Florianópolis - SC, 2002.

GRAY, E.; SMITH, W. On the limitations of software process assessment and the recognition of a required re-orientation for global process improvement. Software Quality Journal, Vol. 7, No. 1, Pag.: 21 – 34, 1998.

JURAN, J. **Quality Control Handbook, 4th Edition.** McGraw-Hill, New York, NY, 1988.

KEENEY, R. **Value-Focused Thinking: A Path to Creative Decision Making.** Harvard University Press, 1992.

KUILBOER, J.; ASHRAFI, N. **Software process and product improvement: na empirical assessment,** Information and Software Technology, Volume 42, Issue 1, Pages 27-34, 2000.

PAULK C. **The CMM – your questions answered.** BCS SPIN special event, London, 1995.

PETRI, S. **Modelo para Apoiar a Avaliação das Abordagens de Gestão de Desempenho e Sugerir Aperfeiçoamentos:** Sob a ótica Construtivista. Florianópolis, 2005. 235 f. Tese (Doutorado) – Universidade Federal de Santa Catarina. 2005.

PITTERMAN, B. **Telcordia technologies: the journey to high maturity.** IEEE Software (July–August), 89–96, 2000.

SCHEID, L. **Avaliação em melhoria de processo de software:** Uma abordagem utilizando a Metodologia Multicritério de Apoio à Decisão – Construtivista. Florianópolis, 2005. 281 f. Dissertação (Mestrado) – Universidade Federal de Santa Catarina. 2005.

SEI. Software Engineering Institute. Carnegie Mellon University. **CMMI Maturity Profile.** Disponível em <http://www.sei.cmu.edu/appraisalprogram/profile/pdf/CMMI/2006sepCMMI.pdf> . Acesso em 6 de Fevereiro. 2007.

SOFTEX. Associação para Promoção da Excelência do Software Brasileiro - SOFTEX,. **Avaliações.** Disponível em

http://www.softex.br/portal/mpsbr/_avaliacoes/default.asp . Acesso em 6 de Fevereiro. 2007.

WANG, Y., KING, G., DORLING, A., PATEL, D., COURT, I., STAPLES, G. ROSS, M., A **Worldwide Survey of Base Process Activities Towards Software Engineering Process Excellence**. IEEE, 1998.

9. Anexos

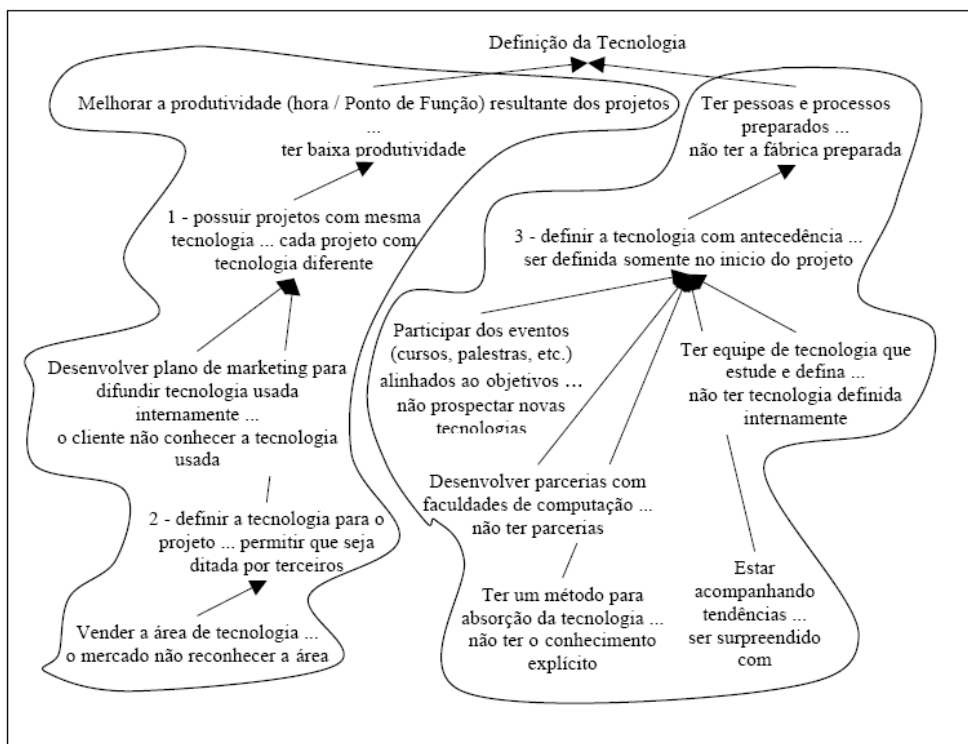
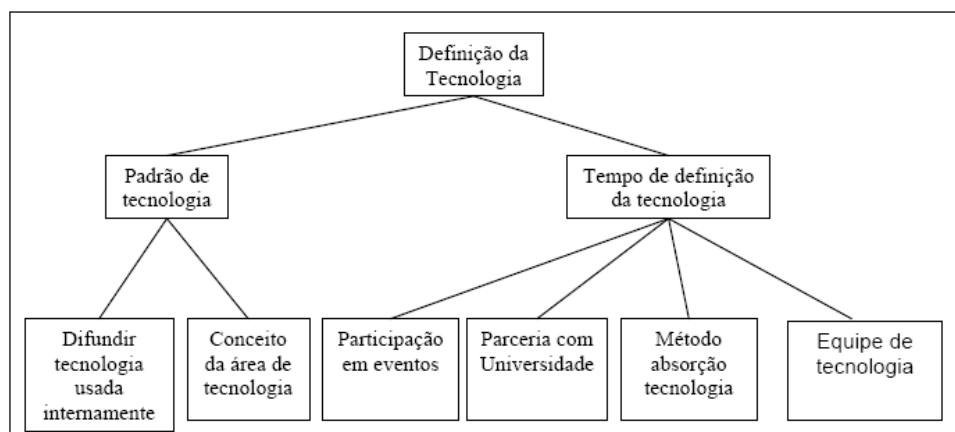


FIGURA 1 – Mapa de Hierarquia de Conceitos



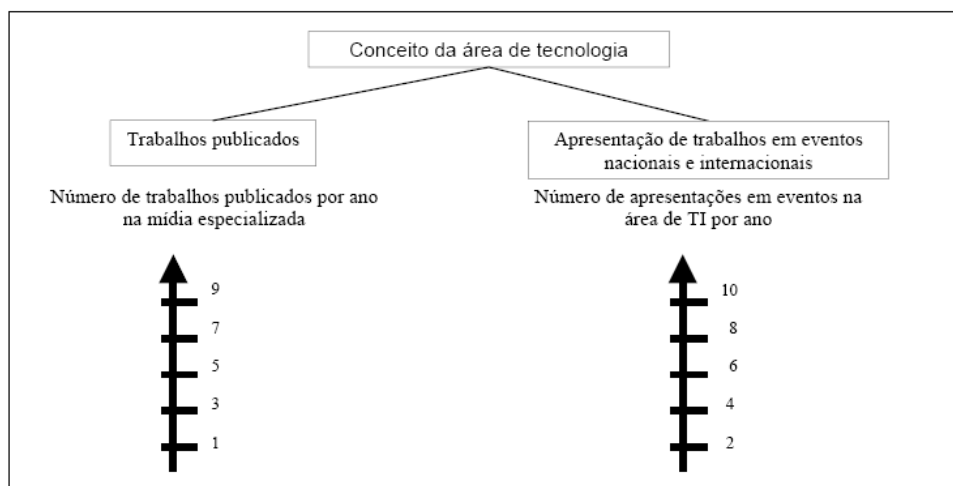


FIGURA 3 – Descritores

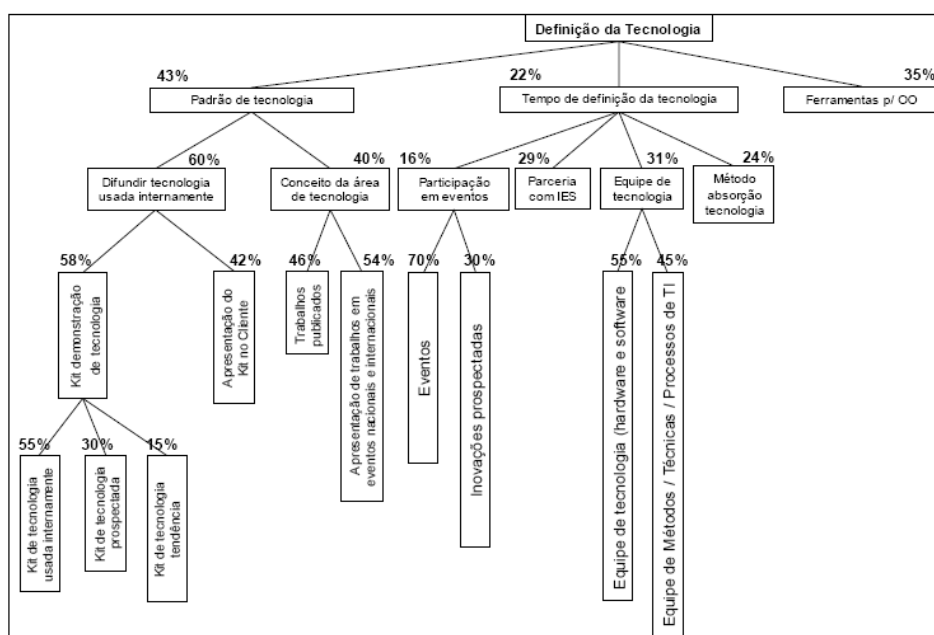


FIGURA 4 - Árvore de valor do PVF *definição da tecnologia* com as taxas de substituição

Trabalhos publicados			
Níveis de Impacto	Níveis de Referência	Descrição	Função de Valor
N5	BOM	Ter 9 trabalhos publicados	133,5
N4		Ter 7 trabalhos publicados	100
N3		Ter 5 trabalhos publicados	66,5
N2	NEUTRO	Ter 3 trabalhos publicados	0
N1		Ter 1 trabalhos publicados	-33,6

QUADRO 1 – Função de Valor para o descritor *trabalhos publicados*

Ponto de Vista Fundamental / Elementar ou Descritor	Taxa de Substi- tuição	Função de Valor					Ação Global
		N1	N2	N3	N4	N5	
Definição da tecnologia	78%						28,7
Difundir tecnologia usada internamente	60%						-15,8
Kit demonstração de tecnologia	58%						14,2
Kit de tecnologia usada internamente	55%	-16,7	0,0	66,7	100,0	133,3	66,7
Kit de tecnologia prospectada	30%	-75,0	0,0	50,0	100,0	125,0	-75,0
Kit de tecnologia tendência	15%	-50,0	0,0	50,0	100,0	150,0	0,0
Apresentação do kit no cliente	42%	-57,3	0,0	71,3	100,0	128,4	-57,3
Conceito da área de tecnologia	40%						30,6
Trabalhos publicados	46%	-33,6	0,0	66,5	100,0	133,5	66,5
Apresentação de trabalhos em eventos	54%	-73,2	0,0	69,1	100,0	130,1	0,0
Tempo de definição da tecnologia	22%						38,3
Participação em eventos	16%						42,0
Eventos	70%	-60,0	0,0	60,0	100,0	140,0	60,0
Inovações prospectadas	30%	-50,0	0,0	75,0	100,0	125,0	0,0
Parceria com IES	28%	0,0	50,0	100,0	137,5		50,0
Equipe de tecnologia	32%						55,0
Equipe de tecnologia (hardware e software)	55%	0,0	100,0	110,0			100,0
Equipe de Métodos / Técnicas / Processos de TI	45%	-200,0	0,0	100,0	200,0		0,0
Método absorção tecnologia	24%	-100,0	0,0	100,0	500,0		0,0
Ferramentas WEB/OO	35%	-21,7	0,0	54,6	100,0	130,4	54,6

QUADRO 2 – Exemplo de avaliação global

[2.21] Implantação e Certificação no Modelo de Referência MPS.BR no nível G em uma empresa do Pólo de Software AmazonSoft

Entidade: SOLTIN - Soluções Integradas em Tecnologia de Informações Ltda. - Av. Gen. Rodrigo Octávio, 1866, Sede do CIDE, sala: 23 - São Lázaro, CEP: 69073-620 - Manaus - AM - Brasil

Autor: Paulino Wagner Palheta Viana - contato: _soltin@yahoo.com.br

Objetivos e Justificativas:

No Amazonas está acontecendo um grande movimento nas empresas fabricantes de software, graças à divulgação do PBQP-SW, desde 2004, a representatividade dos projetos submetidos ao programa tem aumentado consideravelmente, e nos credenciou a ser sede do EQPS-Manaus no mesmo ano, isso serviu como uma grande motivação para as empresas. Para o ciclo de 2005, Manaus foi a capital brasileira com o maior número de projetos submetidos no PBQP-SW. Isso representa o esforço e comprometimento empregado pelas empresas de software da região, no que se diz respeito à qualidade de software [1]. Acompanhando esse movimento, em 2004 foi realizado em Manaus o curso oficial do MPS.BR e com grande satisfação, das nove inscrições para a prova do dia 3 de dezembro, todas as pessoas foram aprovadas. Isso evidencia, que os profissionais estão integrados com as realizações do projeto MPS.BR coordenado pela SOFTEX [2].

O AmazonSoft tem cumprido o seu papel como agente catalisador do *cluster* de software do Amazonas, haja vista a sua grande interação com os principais fomentadores do desenvolvimento da indústria local, representados pelas esferas governamentais, acadêmicas e empresariais, bem como pela indução ao surgimento de novas empresas desenvolvedoras de software no estado [3].

E dentro desse cenário, encontra-se a Soltin - Soluções Integradas em Tecnologia de Informação Ltda., empresa de base tecnológica incubada no CIDE – Centro de Incubação e Desenvolvimento Empresarial, com participação expressiva no SBQS2003, PBQP-SW 2004, PBQP-SW 2005 e PBQP-SW 2006. Com o objetivo de buscar através desse projeto a aderência do seu processo produtivo de software ao nível G do MPS.BR, com apoio de recursos do CNPq, pelo edital HRAE Inovação 2004.

Metodologia de execução:

A Soltin Ltda, é uma empresa de desenvolvimento de Sistemas de Informação e participante do *AmazonSoft*. A área de atuação da Soltin concentra-se no desenvolvimento de projetos de sistemas para atender demandas específicas das empresas do pólo industrial, comércio e serviços em geral. Trata-se de uma empresa de pequeno porte, constituída de dois sócios fundadores e cinco colaboradores.

Devido à preocupação e comprometimento dos sócios com a qualidade no desenvolvimento de software, a empresa já foi concebida, considerando a necessidade de adoção, de um processo padrão de desenvolvimento de software. Esse trabalho foi pioneiro no CIDE, por se tratar da primeira empresa incubada a

publicar um artigo no SBQS2003 na categoria de *Relato de Experiência* [4]. Isso nos trouxe bons resultados, pois essa iniciativa mostra que uma pequena empresa recém criada, também pode buscar competência e maturidade para produzir software atendendo um padrão de qualidade. Porém para manter o processo, é necessário comprometimento dos membros da equipe e infelizmente a empresa passou por uma reformulação societária, tendo como consequência a queda da qualidade no controle estabelecido no GSQS.

Atualmente, a empresa já reformulada, está se preparando para implantar o nível G do MPS.BR, uma iniciativa envolvendo Organizações Integrantes do Sistema SOFTEX, Instituições de Ensino, Pesquisa e Centros Tecnológicos e Sociedade de Econômica Mista. O comprometimento é tanto que, esse projeto foi submetido ao ciclo do PBQP-SW 2005, e com o apoio do CNPq junto ao Edital HRAE Inovação 2004, contemplando com bolsas para um mestre, dois graduados e dois graduandos, esse projeto teve continuidade no PBQP-SW 2006.

No período inicial do projeto, foi identificado quais eram as etapas do processo padrão remanescente. Foi constatado que 50% dos resultados esperados do Processo Gerência de Requisitos já vinham sendo aplicado, caracterizado como parcialmente implementado, pois necessitaria padronizar alguns *templates* e torná-los mais objetivos para auxiliar a rastreabilidade. Em relação à aderência ao Processo Gerência de Projeto, foi constatado que somente 38% dos resultados esperados do processo vínhamos sendo aplicados, e também caracterizado como parcialmente implementado, pois necessitaria padronizar todos os *templates*.

Com a nova versão da Guia Geral 1.1 [5], os esforços foram direcionados para atender as mudanças realizadas. A equipe concentrou a pesquisa no processo com maior prioridade, os resultados obtidos nos processos no nível G foram satisfatórios. Todos os artefatos necessários para aderência ao nível G do MPS-BR foram definidos e implementados.

Conclusão:

Este projeto, foi um esforço voluntário de uma equipe com o intuito de pesquisar e implementar o nível G do MPS.BR, em uma empresa de base tecnológica, Soltin Ltda., incubada no CIDE, que é sede do AmazonSoft.

Esse Projeto visa a melhoria do processo produtivo de software da empresa em questão, seguindo recomendação do MPS.BR, no que se diz respeito, a uma implementação e reconhecimento mais gradual da melhoria no processo de software.

Evidentemente que as dificuldades foram surgindo durante o período inicial, e durante o projeto também, felizmente terminamos o projeto junto ao CNPq sem maiores problemas, pois a experiência dos implementadores é muito importante, não basta somente base teórica, precisa conhecer o *chão de fábrica*.

Agradecimentos:

A FUCAPI, por ter incentivado o nosso grupo de estudo do CMMI que foi utilizado como preparatório para o exame MPS.BR e que custeou todo o treinamento e exame. Ao Francisco Vasconcellos, por toda sua orientação e coordenação do

grupo de estudo. A Jucele Vasconcellos, por colaborar como parte integrante da equipe do projeto. E aos demais membros da equipe.

Referências Bibliográficas e pesquisas.

[1] VASCONCELLOS, F. J. S. “Qualidade de Software na Região Norte”, Projeto premiado em terceiro lugar no PBQP-2004.

[2] www.softex.br

[3] MAUÉS, A. L. da Silva, “O AmazonSoft como núcleo SOFTEX no Amazonas”. Projeto 5.02. PBQP 2004.

[4] VIANA, P. W. P., TUPINAMBÁ, S. L. F., CONTE, T. U. “Definição e implantação do Processo de Software em uma empresa do Pólo de Software AmazonSoft”, SBQS2003, Fortaleza, 2003.

[5] MPS.BR Melhoria de Processo de Software Brasileiro - Guia Geral 1.1. www.softex.br

ROCHA, A. R. C., MALDONADO, J. C., WEBER, K. C., “Qualidade de Software: Teoria e Prática”, São Paulo: Prentice Hall, 2001.

NBR ISO/IEC 12207, “Tecnologia de informação: processos de ciclo de vida de software”, Rio de Janeiro: ABNT, 1998.

Project Management Institute (PMI), “PMBOK - Project Management Body of Knowledge”, Acessado em: 24/01/2003. Disponível na Internet: <http://www.pmi.org>.

PAULK, M. C. et al. “The Capability Maturity Model: Guidelines for improving the Software Process”, Carnegie Mellon University, Software Engineering Institute, Addison-Wesley, MA, 1995.

A Guide to the Project Management Body of Knowledge - PMBOK™, Project Management Institute (PMI), 2004.

The International Organization for Standardization and the International Electrotechnical Commission. **Software Engineering - ISO/IEC TR 16326: Software Engineering - Guide for the Application of ISO/IEC 12207 to Project Management**, Geneve: ISO, 1999.

The International Organization for Standardization. **Quality Management - Guidelines to Quality in Project Management**, Geneve: ISO, 1997.

[2.23] Melhoria de Processo de Software em uma Pequena Empresa: a Experiência da In Forma

Entidade: In Forma Software Ltda

Autor: Ciro Coelho, Pedro Cecconello - {ccoelho, pceconello}@informa.com.br

Introdução

A indústria do software vem experimentando um grande crescimento nas últimas décadas. Hoje, o software está presente na vida de praticamente todas as

peças, seja através do uso de computadores, sistemas de automação comercial e industrial ou software embutido em eletrodomésticos e telefones celulares. Como consequência desse crescimento, temos o aumento da complexidade do software e as exigências cada vez maiores do mercado. É exigido das empresas de software que os sistemas sejam desenvolvidos com prazo e custo determinados e obedeçam a padrões de qualidade.

Para atender essas exigências, tornou-se necessário investir no processo de desenvolvimento de software, já que é cada vez mais evidente a correlação entre a qualidade do produto de software desenvolvido e o processo de desenvolvimento adotado.

Como forma de guiar a melhoria e avaliar a qualidade dos processos de software, modelos de referência têm sido adotados. Possuir reconhecimento ou certificação nesses modelos tornou-se um importante diferencial para as empresas de software. Entre os principais modelos de qualidade, podemos citar a ISO 15504 (SPICE), o *Capability Maturity Model Integration* (CMMI) e o projeto de Melhoria de Processo do Software Brasileiro (mps.Br). Inserida nesse contexto, está a In Forma Software, empresa pernambucana de software, localizada no Porto Digital, em Recife. Contando atualmente com cerca de 30 colaboradores, que formam uma equipe altamente qualificada, a In Forma tem crescido continuamente e solidamente no mercado brasileiro e internacional desde 1993, oferecendo, a seus clientes, prazos, custos e qualidade bem definidos em sistemas de informação.

Como forma de continuar prestando serviços com qualidade, a empresa tem investido fortemente na melhoria de seus processos de software, sempre com base em modelos reconhecidos nacional e internacionalmente.

A In Forma participa, desde Abril de 2004, do projeto Consultoria no *Capability Maturity Model Integration*, promovido pelo Sebrae/PE – PSI, que tem como objetivo a melhoria de processos de software fundamentada no CMMI. Este projeto reúne empresas participantes do Projeto Setorial Integrado para Exportação de Software e Serviços Correlatos (PSI-SW) de Pernambuco.

Além da participação nesse projeto, a In Forma também tem tomado outras iniciativas no tocante à melhoria e certificação de seus processos. Como resultado dessas iniciativas, a In Forma obteve, em Setembro de 2005, a certificação no Nível G do mps.Br, tendo sido a primeira empresa certificada nesse modelo.

Objetivos e Justificativa

Os objetivos iniciais do programa de melhoria de processos de software da In Forma, determinados pela diretoria da empresa com base nos objetivos estratégicos da organização, eram os seguintes:

- Aumento da produtividade através da utilização de processos mais efetivos;
- Aumento da qualidade dos produtos de software entregues aos clientes;
- Obter certificações de qualidade que pudessem dar maior visibilidade à empresa e que servissem como “cartão de apresentação” da In Forma no mercado externo.

Um estudo realizado pelo *Software Engineering Institute* (SEI) demonstra o impacto e os benefícios do CMMI em cinco categorias: custo, prazo, qualidade (mais relacionado à redução de defeitos ao longo do tempo), satisfação do cliente e retorno de investimento.

Por ser reconhecido mundialmente e possuir resultados concretos de sucesso na melhoria de processos, o CMMI foi selecionado como base para a melhoria de processos na In Forma. Assim, o objetivo inicial passou a ser a obtenção do Nível 2 do CMMI.

Durante o andamento do projeto, o surgimento do mps.Br foi vislumbrado como uma oportunidade, já que possui os mesmos fundamentos do CMMI, que já vinha sendo implementado, além de ser um modelo mais adequado à realidade brasileira. Por isso, o reconhecimento como empresa mps.Br nível G também tornou-se um objetivo.

Metodologia de Execução

O projeto foi dividido em duas grandes fases: Definição e Institucionalização. Na fase de Definição, o grande objetivo era definir um processo aderente ao Nível 2 do CMMI. Já a fase de Institucionalização tinha como objetivo colocar o processo definido em funcionamento na empresa, sendo entendido, aceito e utilizado por todos os colaboradores.

A fase de Definição foi composta das seguintes atividades:

- Definição dos integrantes do *Software Engineering Process Group* (SEPG);
- Levantamento do processo já existente;
- Identificação do grau de aderência do processo existente;
- Elaboração de processo aderente ao modelo CMMI nível 2, considerando as características e cultura da organização e os processos já existentes;
- Validação do processo com os especialistas da empresa (SEPG) e consultores externos;
- Desenvolvimento de ferramentas de apoio ao processo.

A fase de Institucionalização foi composta pelas atividades abaixo:

- Treinamentos gerais para todos os colaboradores da empresa, incluindo o processo definido e ferramentas que o apóiam;
- Treinamentos específicos, de acordo com o papel exercido por cada colaborador;
- Institucionalização gradual do processo nos projetos. Em um primeiro estágio, foram institucionalizados os processos de Gestão de Projetos (*Project Planning*, *Project Monitoring and Control* e *Supplier Agreement Management*) e Garantia da Qualidade. Em seguida, foi institucionalizado o processo de Gerenciamento de Requisitos, seguido do processo de Gerência de Configuração. O processo de Medição e Análise foi

institucionalizado à medida que os demais processos começavam a ser utilizados;

- Ajustes no processo definido, levando-se em consideração as sugestões dos colaboradores, as dificuldades encontradas, os resultados das auditorias de garantia da qualidade e as observações do SEPG;
- Realização de dinâmicas, jogos, e-mails informativos etc. com o intuito de difundir o processo e facilitar a institucionalização;
- Simulação de avaliação conduzida pela equipe de garantia da qualidade;
- Avaliação externa nos moldes SCAMPI classe C;
- Avaliação mps.Br.

A escolha por um processo gradual de institucionalização teve o intuito de diminuição de esforço das equipes dos projetos. Imagina-se que, do contrário, o *overhead* causado pela introdução de processos pouco conhecidos naquele momento seria muito grande, podendo, inclusive, causar resistência ao processo por parte das equipes dos projetos.

Resultados Relevantes e Lições Aprendidas

O projeto de melhoria de processos da In Forma já obteve resultados significativos em diversos aspectos. Processos definidos e institucionalizados, aumento da qualidade e produtividade e o aprendizado de boas práticas que podem ser aplicadas nas outras etapas do projeto foram alguns dos benefícios obtidos.

No que diz respeito ao processo de software, destaca-se a definição, documentação e institucionalização de processos para todas as áreas do nível 2 do CMMI.

Outro resultado relevante está relacionado ao reconhecimento, por parte da Diretoria, dos colaboradores e dos clientes, da importância das ações de melhoria de processos implementadas. Dentre os benefícios observados, podemos destacar:

- Maior controle sobre prazos e escopo;
- Melhor entendimento dos requisitos;
- Estimativas mais acuradas;
- Produtos de trabalho melhor gerenciados;
- O cliente tem mais conhecimento sobre o andamento do projeto, sabe quando cada produto será entregue e reconhece a empresa como bem organizada.

Com relação ao aumento da visibilidade da empresa no mercado, a In Forma obteve, em setembro de 2005, a certificação no nível G do mps.Br, o qual abrange as áreas de Gerência de Projetos e Gerência de Requisitos. Confirmando o seu papel inovador e seu compromisso com a melhoria de processos de software, a empresa foi a primeira a ser

avaliada com base neste modelo.

Outro resultado importante foi o aprendizado de boas práticas. Essas boas práticas são úteis tanto para a sequência do projeto de melhoria quanto para outras empresas com características similares às da In Forma. Dentre as lições aprendidas durante o projeto, podemos destacar:

- Uma definição cuidadosa do processo pode diminuir o *overhead* durante a institucionalização. Essa diminuição é particularmente importante para empresas e equipes menores, onde qualquer esforço extra pode comprometer a viabilidade do projeto de desenvolvimento. Apesar de exigir um maior tempo na fase de Definição, essa preocupação com a definição de um processo adequado facilitou muito a institucionalização, que pôde ser realizada em menor tempo e com menor resistência por parte dos colaboradores. Uma definição cuidadosa de processos deve levar em conta os seguintes aspectos:
 - Procurar aproveitar o que a organização já possui de bom, os processos que já funcionam;
 - Entender a cultura da organização antes de tentar definir os processos;
 - Sempre validar os processos com as pessoas que irão utilizá-lo;
 - Ter sempre em mente que processo não é tudo. Acima dos processos, está a capacidade da empresa de produzir software e gerar lucro. O processo deve ajudar a organização a atingir esses objetivos;
- A disponibilização do processo em formato HTML, assim como templates, ferramentas, guias etc na intranet da empresa (Figura 1), tornou o acesso às informações fácil e rápido, facilitando a institucionalização;

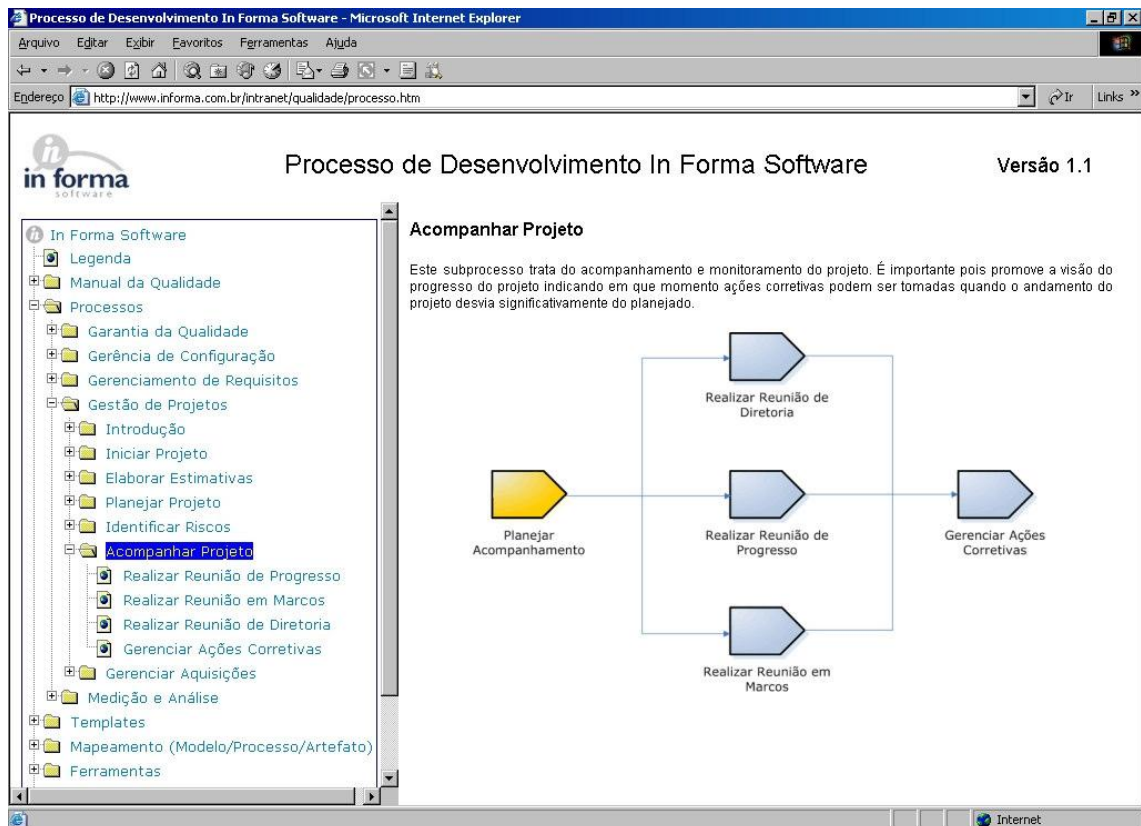


Figura 1 - Site do processo da In Forma na intranet da empresa

- Deve-se investir muito na capacitação das pessoas. Sem esse investimento, a institucionalização dos processos é praticamente impossível. Entre as iniciativas realizadas na In Forma para capacitar seus colaboradores, podemos citar:
 - Treinamentos para toda a equipe de colaboradores;
 - Consultoria em áreas onde a institucionalização se mostrou mais difícil, como, por exemplo, Gerência de Configuração;
 - Elaboração de palestras, e-mails informativos, dinâmicas e jogos para acelerar o aprendizado do processo;
 - Simulações de avaliação e auditorias de garantia da qualidade com o objetivo de identificar as áreas que precisavam de treinamentos adicionais;
- Ferramentas para apoiar o processo são essenciais, mas é possível diminuir significativamente os custos com a aquisição de ferramentas. Dentre as estratégias adotadas pela In Forma, podemos citar:

- Adoção de ferramentas gratuitas e largamente utilizadas, como CVS e Mantis;
- Desenvolvimento interno de ferramentas para necessidades específicas da empresa. Foram desenvolvidas ferramentas para estimativas de esforço e rastreabilidade de requisitos;
- Elaboração de planilhas eletrônicas para automatizar procedimentos mais simples;
- O apoio de uma consultoria externa é fundamental para avaliar a aderência do processo definido aos modelos de qualidade e para sugerir estratégias para a institucionalização dos processos;
- É fundamental que a alta direção da empresa não só apóie o projeto de melhoria como também demonstre esse apoio para os colaboradores da empresa. Se não ficar claro que a melhoria de processos é uma prioridade, as atividades de melhoria ficarão em segundo plano em relação às demais atividades dos projetos.

Conclusão e Perspectivas Futuras

O projeto de melhoria de processos de software da In Forma tem trazido muitos benefícios para a empresa, tanto em termos de aumento de qualidade e produtividade quanto em relação à visibilidade e reconhecimento da empresa nos mercados nacional e internacional.

Os resultados obtidos pela In Forma mostram que a melhoria de processos pode ser útil também para pequenas empresas. As empresas de menor porte precisam lidar com algumas dificuldades extras. Essas dificuldades, entretanto, podem ser bastante reduzidas com a utilização de técnicas e abordagens mais adequadas ao perfil da empresa, como mostram as lições aprendidas apresentadas na seção anterior.

Entre os aspectos fundamentais para se obter sucesso em projetos de melhoria, podemos destacar o apoio da alta direção da empresa, a necessidade de se definir um processo adequado às características da organização e a importância do investimento na capacitação dos colaboradores.

Após a certificação no nível G do mps.Br, obtida em Setembro de 2005, os próximos objetivos da In Forma, na área de melhoria de processos de software, incluem a obtenção do nível 2 do CMMI, ainda no primeiro semestre de 2007, e, a médio prazo, obter o nível 3 do CMMI e o nível C do mps.Br.

Referências Bibliográficas

1. Machado, L.F.C., Modelo para Definição de Processos de Software na Estação TABA, COPPE/UFRJ, Dissertação de Mestrado, 2000.
2. Goldenson, D.R., Gibson, D.L., *Demonstrating the Impact and Benefits of CMMI: Na Update and Preliminary Results*, Software Engineering Institute, Special Report CMU/SEI-2003-SR-009, 2003.
3. Weber, Kival C. *et al.* Uma Estratégia para Melhoria de Processo de Software nas Empresas Brasileiras. *Fifth Conference for Quality in Information and*

Communications Technology (QUATIC). Porto, Portugal, outubro de 2004, p. 73-78.

[2.24] ISO 9001:2000 - Pitang

Entidade: Pitang - High Performance IT - Recife - PE

C.E.S.A.R – Centro de Estudos e Sistemas Avançados do Recife - Recife - PE

Autores: Karina Rodrigues^{1,2}, Suzana Sampaio¹, Teresa Maciel^{1,2}, Thayssa Rocha¹ - {karina.rodrigues, suzana.sampaio, thayssa.rocha}@pitang.com.br,teresa@cesar.org.br

Abstract. *This paper presents a project of the ISO9001:2000 implementation in an IT organization, Pitang, which is sited in Recife. Pitang is a software factory, and was created through a partnership between C.E.S.A.R and Rio Bravo Nordeste I. This paper describes the experience of the project, including goals, adopted methodology, and the main results.*

Resumo. Este artigo descreve a experiência da implantação da ISO 9001:2000 em uma organização de TI sediada em Recife, a Pitang, a qual foi criada como fruto de uma cooperação entre o C.E.S.A.R e a Rio Bravo Nordeste I para atuar como Fábrica de Software. O artigo apresenta os objetivos do projeto, a metodologia utilizada, os resultados obtidos e trabalhos.

1. Introdução

A Pitang - empresa do ecossistema Porto Digital, criada em 2005 - oferece serviços de consultoria e tecnologia, fábrica de software, outsourcing, soluções e produtos com a qualidade proveniente do conhecimento do negócio de cada cliente. Fábrica de software do C.E.S.A.R - considerado o mais inovador Instituto de Pesquisa do Brasil (FINEP, 2004) e melhor empresa de serviços de software no Brasil (Info200, 2005) - a Pitang foi criada como fruto de uma cooperação entre C.E.S.A.R e Rio Bravo Nordeste I, fundo de investimento de capital de risco administrado pela Rio Bravo.

A Pitang possui atualmente em torno de 50 clientes ativos, 10 empresas parceiras, filiais em São Paulo, Brasília, Fortaleza e João Pessoa além de um escritório no Rio de Janeiro.

Desde sua criação, a Pitang tem como princípio básico o investimento no binômio Processo & Qualidade, confirmando seu compromisso com o cliente, fornecedores, parceiros e colaboradores. Nesse sentido, a busca pela certificação ISO 9001:2000 para os processos chave de sua atuação surgiu de forma natural para garantir o atendimento a este princípio.

Tendo como missão: “Desenvolver, manter e integrar produtos e soluções em Tecnologia da Informação e Comunicação, com ênfase no processo e qualidade, satisfazendo as necessidades e criando valor para os nossos clientes”, a busca pelas certificações de qualidade certamente estarão sempre presentes no caminho da empresa.

Diante deste contexto, a Pitang resolveu definir um programa de melhoria baseado em melhores práticas reconhecidas por organizações. Inserido neste programa, o projeto de certificação ISO9001:2000 foi considerado fundamental para apoiar a estruturação dos processos da Pitang, assim como para estabelecer uma identidade própria da organização.

2. Objetivos e Justificativa

Criada através da iniciativa entre o C.E.S.A.R e Rio Bravo, a Pitang precisava buscar sua identidade própria, e através desse objetivo buscou a padronização e melhoria interna de seus processos, o que proporcionaria, entre outros aspectos, um diferencial competitivo perante o mercado regional e nacional.

Neste contexto, uma das primeiras iniciativas da Pitang teve o foco na certificação ISO 9001:2000, uma vez que através do desenvolvimento dos requisitos da norma, estaria organizando seus processos operacionais (de Fábrica e desenvolvimento de software) e sua gestão organizacional. Adicionalmente, a criação de indicadores de gestão tornaria mais fácil e efetivo o acompanhamento da operação e o estabelecimento de melhorias.

Além de objetivos de melhoria interna, o atendimento da certificação também possibilitaria à nova empresa a possibilidade de participar de disputas em licitações públicas e privadas, aumentando a visibilidade e consequentemente as oportunidades de negócio.

3. Metodologia de Execução

Iniciado em Março de 2005 através de um diagnóstico realizado em conjunto com a consultoria SQS Qualidade, o projeto de certificação teve duração de 13 meses até a obtenção do certificado em Abril de 2006.

Inicialmente, a equipe foi alocada para definição do macro processo organizacional e foram necessários cinco meses de elaboração até sua divulgação e publicação, bem como da primeira versão do Sistema de Gestão de Qualidade. A equipe inicial foi composta por quatro pessoas em tempo parcial para definição dos processos e duas para execução de auditorias internas e acompanhamento do sistema até a certificação.

Definição, Documentação e Melhoria do Processo Organizacional

No sentido de melhor disseminar o conhecimento e fomentar o envolvimento de todos os colaboradores, o sistema de gestão da qualidade da Pitang foi definido e documentado em site na Intranet. Desde o início, ficou estabelecido que toda a documentação do sistema seria disponibilizada de forma digital.

Como o foco da certificação concentra-se no desenvolvimento de software e consultoria de TI, foi necessária a definição de todo o ciclo de vida do desenvolvimento, incluindo a elaboração da proposta, até a entrega do produto/serviço acordado. O processo organizacional foi definido através de um sistema de documentação estabelecido em seis níveis de documentos: Processo, Subprocesso, Atividade, Técnica/Método, Artefato e Ferramenta.

Os níveis garantiram independência entre os documentos de forma a garantir reuso (por exemplo, uma atividade pode ser referenciada em mais de um processo) e minimizar custo de manutenção nos ativos.

Toda documentação foi gerada e disponibilizada em formato digital, acessível através da Intranet da empresa, sob responsabilidade da Gerência de Qualidade da Pitang. Por outro lado, todos os colaboradores foram habilitados a submeter solicitações de mudanças ou melhorias a partir da ferramenta de gestão de mudanças da Pitang. Desta forma as solicitações são enviadas aos responsáveis pelos processos, analisadas e controladas até sua resolução.

Auditorias

Para garantir um controle efetivo e visibilidade do progresso do projeto, foram planejadas e realizadas cinco auditorias internas, além do diagnóstico inicial e da pré-auditoria. Ao final de cada auditoria, um plano de ação era definido, assim como acompanhada a sua execução, a partir das recomendações identificadas pela equipe auditora.

4. Resultados obtidos

A certificação da Pitang foi obtida sem nenhuma não-conformidade registrada, apresentando apenas algumas observações e sugestões para serem acompanhadas na auditoria de manutenção a ser realizada no ano de 2007.

Considerando-se uma empresa que já nasceu com projetos críticos e de longa duração, a Pitang conseguiu evoluir bastante a partir dos processos e sistema de gestão da qualidade estabelecidos.

A melhoria foi visível em toda área de operações, principalmente na Fábrica de Software, que no início desbravava um horizonte muito descrito em páginas de livros, mas pouco implementado na prática. Uma vez descritas e formatadas – através do processo – as fronteiras entre a Fábrica e seus clientes (projetos internos da organização que demandam serviços de construção de software), o controle da operação foi estabelecido.

O processo baseia-se em formalização dos relacionamentos entre a Fábrica e seus clientes, iniciando através do estabelecimento de um acordo de nível de serviço (ANS) entre as partes, em seguida formalizando o recebimento das demandas através de uma análise crítica dos artefatos que servirão como entrada para a fábrica. Durante o desenrolar do projeto a comunicação entre as partes também é formalizada através de ferramentas que mantêm bem definidas as fronteiras de atuação dos projetos enquanto clientes.

No âmbito de gestão organizacional, o sistema de gestão auxiliou não apenas no estabelecimento das diretrizes organizacionais, mas também no acompanhamento da melhoria de todas as áreas interligadas a operações.

Para a área de Operações, coração da organização, as melhorias puderam ser percebidas a partir do aumento da formalidade exigida para o acompanhamento dos projetos, proporcionando um aumento na satisfação do cliente quanto à visibilidade do projeto (progresso, dependências, riscos). Para a alta direção, o

projeto viabilizou informações para apoio a tomada de decisões, gerando ações preventivas e/ou corretivas a fim de antecipar e reverter problemas.

Os indicadores de processo implementados com a certificação auxiliaram todas as áreas envolvidas a conhecerem suas limitações para mais facilmente agirem sobre elas, ou ainda, identificarem quando e como devem agir para superá-las. Neste primeiro ano os indicadores apoiaram a Pitang especialmente no processo de auto conhecimento, para que sejam dadas em um segundo momento condições de planejar melhor os objetivos corporativos e buscar a melhoria contínua.

Atualmente os desafios estão relacionados à automação da coleta de métricas e indicadores.

Este processo quando realizado de forma manual toma muito tempo para sua operacionalização. Além da automação, a integração das ferramentas de gestão hoje adotada pela Pitang deve ser uma prioridade para este segundo ano.

5. Aplicabilidade dos Resultados

A aplicabilidade dos resultados obtidos é direta na capacidade que hoje a Pitang tem em gerar serviços de qualidade, obtendo visibilidade gerencial do alcance de suas metas, e identificando pontos de melhoria de forma mais concreta e objetiva. Apesar de parecer um projeto interno, assim como a própria Pitang, os clientes também têm um ganho considerável em termos de qualidade do software que recebem.

O fato de a Pitang ser uma empresa C.E.S.A.R, apresentando as mesmas preocupações com o desenvolvimento sócio-econômico da região, fortalece a multiplicação de conhecimento. O envolvimento de mais de 100 profissionais de software na definição e no uso de melhores práticas de mercado, promove o grau de qualificação do capital humano, especialmente de Pernambuco.

6. Características Inovadoras

Apesar de o projeto apresentar objetivos e metas focados em uma norma já conhecida e difundida internacionalmente, a metodologia para definição do processo de forma a distribuir os ativos em níveis de documentação, independentes entre si, pode ser considerado uma inovação em termos de processo, especialmente por ter sido consideradas lições aprendidas com a definição e melhoria de processos no C.E.S.A.R.

A experiência Pitang, repercutiu, por exemplo, na reestruturação da documentação do processo produtivo do C.E.S.A.R, que também passou a adotar a mesma sistemática.

7. Conclusão e Perspectivas Futuras

A certificação ISO é apenas a primeira dentre os objetivos da Pitang, que nos possibilitou estabelecer processos básicos, para um caminho sólido rumo à maturidade organizacional.

Neste sentido, o projeto promoveu uma base estrutural de processos de forma coerente e abrangente, além da definição de mecanismos de melhoria contínua nos mesmos.

Neste primeiro ano de certificação os indicadores estabelecidos foram de grande importância para a Pitang, pois nos apoiou em nosso auto conhecimento, garantindo um planejamento estratégico mais estruturado.

O próximo passo será recebermos a primeira auditoria de manutenção do certificado neste primeiro semestre de 2007 e em paralelo iniciando a busca de mais um objetivo da Pitang que é o CMMI3, o qual certamente será mais objetivo e produtivo a partir de um sistema de gestão devidamente embasado em normas internacionais.

Referências

ISO/IEC NBR ISO 9001 – Quality management systems – Requirements

[2.25] MPS.BR - Melhoria de Processo do Software Brasileiro (Dez2003-Dez2006)

Entidade: SOFTEX - Associação para Promoção da Excelência do Software Brasileiro

Autores: Kival Chaves Weber e Eratóstenes Araújo - {kival.weber, era}@nac.softex.br

Introdução

No início dos anos 2000, estudos mostraram que era necessário um esforço significativo para aumentar a maturidade dos processos de software nas empresas brasileiras e que, nos últimos anos, as empresas de software no Brasil favoreceram a ISO 9000 em detrimento de outras normas e modelos especificamente voltadas para a melhoria de processos de software [MCT/SEPIN 2001 e Veloso 2003].

Segundo dados do MCT - Ministério da Ciência e Tecnologia, em 2003, 214 empresas que desenvolviam software no Brasil tinham certificação ISO 9000, enquanto eram trinta as empresas com certificado SEI/CMU de avaliações CMM[®]. Estas últimas, a maioria subsidiária no país de grandes empresas multinacionais, compreendiam 24 empresas no nível 2, cinco no nível 3, uma no nível 4 e nenhuma no nível 5 do CMM.

Para ajudar na solução deste problema, a SOFTEX – Associação para Promoção da Excelência do Software Brasileiro lançou o Projeto MPS.BR – Melhoria de Processo do Software Brasileiro no dia 11 de dezembro de 2003, numa reunião realizada no MCT em Brasília-DF.

A SOFTEX é uma entidade privada, sem fins lucrativos, que promove ações com abrangência nacional visando transformar o Brasil em um centro de excelência mundial na produção e exportação de software. Dispõe de 31 Agentes SOFTEX, localizados em 23 cidades de 13 Unidades da Federação, com mais de mil

[®] CMM, CMMI e SCAMPI são marcas da CMU – Carnegie Mellon University. MPS.BR, MR-MPS, MA-MPS e MN-MPS são marcas da SOFTEX.

empresas associadas – com o seguinte porte segundo o número de funcionários: 11% grandes, 13% médias, 36% pequenas e 40% micro empresas. Para mais informação, veja o Portal SOFTEX em www.softex.br.

O propósito do Projeto MPS.BR é a Melhoria de Processo do Software Brasileiro, compreendendo dois processos: i) desenvolvimento e aprimoramento do Modelo MPS, baseado nas melhores práticas de Engenharia de Software, em conformidade com as normas ISO/IEC 12207 – Processos do Ciclo de Vida do Software e ISO/IEC 15504 – Avaliação de Processo, compatível com o CMMI – *Capability Maturity Model Integration* [Chrissis 2003, Bush 2005] e adequado à realidade das empresas brasileiras; ii) disseminação e adoção do Modelo MPS, a um custo acessível, tanto em grandes organizações públicas e privadas como em pequenas e médias empresas (PME), em todas as regiões do país.

Até agora, o Projeto MPS.BR contou com as seguintes fontes de recursos financeiros: apoio do BID, FINEP e MCT, complementado por receitas de serviços MPS tais como: taxas de inscrição em cursos, provas e workshops; taxas de análise de pleitos de Instituições Implementadoras – II, Instituições Avaliadoras – IA, Instituições Organizadoras de Grupos de Empresas – IOGE e Consultores de Aquisição – CA; taxas SOFTEX para avaliação MPS em empresas.

Este artigo descreve o MPS.BR no período Dez2003-Dez2006, tendo como finalidade complementar o relatório final do Projeto [2.25] – MPS.BR – Melhoria de Processo do Software Brasileiro (Dez2003-Dez2006), do PBQP Software – ciclo 2006. A seção 2 apresenta o objetivo e a justificativa do projeto. A seção 3 explica como o projeto foi desenvolvido. A seção 4 destaca os principais resultados obtidos. A seção 5 trata da aplicabilidade destes resultados. A seção 6 resume as características inovadoras do projeto. A seção 7 conclui o artigo e apresenta as perspectivas futuras do projeto.

Objetivo e Justificativa

No Projeto [2.25] – MPS.BR – Melhoria de Processo do Software Brasileiro (Dez2003-Dez2006), do PBQP Software - ciclo 2006, consta o seguinte:

- **Objetivo:** Criação de um modelo de Melhoria de Processo de Software (MPS) - composto de um Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS), a ser implantado em um número significativo de empresas de software no Brasil, a um custo acessível (MNC - modelo de negócio cooperado entre várias empresas).
- **Justificativa:** Melhorar os processos de software em 120 empresas de software no Brasil até 2006 e em mais 160 empresas nos dois anos subsequentes, a um custo acessível, com foco na base da pirâmide (processo gerenciado - níveis G e F do MR-MPS).

Metodologia de Execução

Esta seção descreve a estrutura organizacional, etapas e atividades do Projeto MPS.BR no período Dez2003-Dez2006.

Estrutura Organizacional do Projeto MPS.BR

Além de um Conselho de Gestão que se reúne semestralmente e da Unidade de Execução que se reúne mensalmente, o Projeto MPS.BR conta com duas estruturas de apoio para o desenvolvimento de suas atividades: o Fórum de Credenciamento e Controle - FCC e a Equipe Técnica do Modelo - ETM. Através destas estruturas, o MPS.BR obtém a participação de representantes de Universidades, Instituições Governamentais, Centros de Pesquisa e de organizações privadas, os quais contribuem com suas visões complementares que agregam qualidade ao empreendimento.

O FCC tem como principais objetivos assegurar que as Instituições Implementadoras - II e Instituições Avaliadoras - IA sejam submetidas a um processo adequado de credenciamento e que suas atuações não se afastem dos limites éticos e de qualidade esperados, além de avaliar e atuar sobre o controle dos resultados obtidos pelo MPS.BR.

Por outro lado, cabe à ETM atuar sobre os aspectos técnicos relacionados ao Modelo de Referência (MR-MPS) e Método de Avaliação (MA-MPS), tais como a concepção e evolução do modelo, elaboração e atualização dos Guias, preparação de material e definição da forma de treinamento (cursos oficiais e workshops de reciclagem) e de aplicação de provas individuais, publicação de Relatórios Técnicos e interação com a comunidade visando a identificação e aplicação de melhores práticas.

Etapas e Atividades do Projeto MPS.BR

A execução do Projeto MPS.BR compreende dez etapas: a primeira etapa, anual, teve como foco a sua organização (P do PDCA); as outras nove etapas, semestrais, são de execução, controle e aprimoramento.

A 1ª etapa – Organização do Projeto, realizada de Dez2003-Dez2004, teve como objetivo: i) organizar o projeto; ii) estabelecer seus objetivos; iii) definir uma versão preliminar do Modelo MPS (MR-MPS, MA-MPS e MN-MPS); iv) iniciar atividades de treinamento no modelo (cursos, provas e workshops do MPS.BR); v) realizar experiências-piloto de implementação do MR-MPS em grupos de empresas no Rio de Janeiro, Recife e Campinas.

As outras nove etapas semestrais do Programa MPS.BR, no período Jan2005-Jun2009, compreendem atividades de execução, controle e aprimoramento visando a Melhoria de Processo do Software Brasileiro.

Este artigo cobre a 1ª etapa anual (Dez2003-Dez2004), de organização do projeto, e as quatro etapas semestrais (Jan2005-Dez2006), de execução, controle e aprimoramento do projeto.

Resultados Obtidos

Esta seção destaca os principais resultados obtidos e os produtos relevantes do Projeto MPS.BR.

Principais Resultados Alcançados

A Tabela 1 apresenta os principais resultados alcançados no MPS.BR por atividade.

Tabela 1. Resultados Alcançados no Projeto MPS.BR

Atividades	Principais Resultados Alcançados no Projeto MPS.BR (Dez2003-Dez2006)
i) Monitoramento e avaliação do Projeto MPS.BR, incluindo o Modelo de Negócio.	Abrangência nacional do Projeto MPS.BR, a um custo acessível. Na formação do custo de implementação do MR-MPS e de avaliação MA-MPS em empresas, considerou-se como premissa a necessidade destes serem acessíveis à micro, pequena e média empresa de software no Brasil, seja pela disponibilidade de Instituições Implementadoras – II e Instituições Avaliadoras – IA em todas as regiões do país (implicando em presença local com menor custo de deslocamento e hospedagem), seja por basear-se na moeda corrente no país (não em dólares), seja pelo modelo de negócio adotado (tanto de serviços específicos para uma empresa como de um pacote de serviços a ser compartilhado por um grupo de empresas).
ii) Aperfeiçoamento do Modelo MPS, incluindo os Guias do MPS.BR	Criação e aperfeiçoamento do Modelo MPS. Disponibilidade de Guias do MPS.BR (Guia Geral, Guia de Implementação, Guia de Avaliação e Guia de Aquisição), criados e aprimorados anualmente pela Equipe Técnica do Modelo - ETM, com base nas melhores práticas da Engenharia de Software e em normas e modelos internacionais.
iii) Capacitação de Especialistas no MPS.BR (cursos, provas e workshops).	Capacitação de especialistas no MPS.BR seja ofertando regularmente cursos oficiais (C1- Introdução, C2 – Implementadores, C3-Avaliadores e C4-Melhoria do Processo de Aquisição de Software) em todas as regiões do país, seja aplicando semestralmente provas individuais em todo o país, seja organizando anualmente workshops de reciclagem. 30 instrutores foram treinados e estão autorizados a ministrar os cursos oficiais do MPS.BR. Mais de 2.100 pessoas participaram de cursos oficiais do MPS.BR, pagando taxas de inscrição acessíveis. Mais de 600 especialistas foram aprovados em provas individuais do MPS.BR (P1-Introdução, P2- Implementadores, P3-Avaliadores; P4-Melhoria do Processo de Aquisição de Software). As questões das provas aplicadas estão disponíveis na seção Cursos-Provas do Portal www.softex.br/mpsbr .
iv) Credenciamento de Instituições Implementadoras - II	15 Instituições Implementadoras – II credenciadas, em todas as regiões do país exceto na Região NO, mediante convênio assinado com a SOFTEX. Até Dez2008, espera-se exceder muito a meta original: 20 II, em todas as regiões do país.
v) Credenciamento de Instituições Avaliadoras - IA	2 Instituições Avaliadoras – IA credenciadas, uma na Região SE e outra na Região NE, mediante convênio assinado com a SOFTEX. Até Dez2008, espera-se exceder a meta original: 15 IA, em todas as regiões do país.
vi) Ferramentas de software para a gestão do Projeto MPS.BR.	A principal ferramenta de software para a gestão do MPS.BR é o Portal SOFTEX www.softex.br/mpsbr , sempre atualizado, com informações disponíveis a todos os interessados. Além disto: i) está sendo “customizado” o ambiente CoreKM, para aprimorar o controle das avaliações, avaliadores, IA, implementadores, II, grupos de empresas e IOGE; ii) está sendo desenvolvido um Repositório MPS contendo resultados de desempenho de empresas que adotaram o Modelo MPS.
vii) Análise de lições aprendidas com a implementação e avaliação do Modelo MPS nas empresas.	Lições aprendidas com a implementação e avaliação do Modelo MPS em empresas foram publicadas em duas edições especiais MPS.BR da Revista ProQualiti: volume 1, número 2, Nov2005; volume 2, número 2, Nov2006. Está começando a ser usado o ambiente CoreKM para apoiar a gestão do MPS.BR, aprimorando o controle das avaliações, avaliadores, IA, implementadores, II, grupos de empresas e IOGE, incluindo Gestão do Conhecimento para análise das lições aprendidas. Em 2007, será iniciada a coleta de resultados de desempenho de organizações que adotaram o Modelo MPS.
viii) Implementação e avaliação do Modelo MPS nas empresas.	Implementação do Modelo MPS em cerca de 120 empresas, em todas as regiões do país, até Dez2006. Dentre estas, coordenadas por Instituições Organizadoras de Grupos de Empresas – IOGE, 93 empresas (52% pequenas, 33% médias e 15% grandes empresas) estão usando o Modelo de Negócio cooperado, conforme o COMUNICADO SOFTEX MPS 20/2005. Estes grupos de empresas estão comprometidos a concluir a implementação do MR-MPS em até 12 meses, a

	contar da data de assinatura do convênio entre a SOFTEX e a respectiva IOGE, e a se submeterem a avaliação MA-MPS nos três meses subsequentes. Até Dez2008, espera-se implementar o Modelo MPS em mais 160 empresas. Foram realizadas 17 avaliações oficiais MA-MPS. Dentre estas, 34% são micro e pequenas empresas, 6% médias empresas e 60% grandes empresas; sendo 18% no nível A (o mais alto), 18% níveis D e E, e 64% níveis F e G. Até Dez2007, espera-se ter um total de 80 empresas com avaliação oficial MA-MPS; até Dez2008, espera-se totalizar 160 empresas avaliadas oficialmente.
ix) Administração do Projeto	Administração transparente, com resultados alcançados superando resultados esperados bastante desafiadores. Semestralmente, são realizadas reuniões do Conselho de Gestão do Projeto - CGP. Mensalmente, a Unidade de Execução realiza Reuniões de Acompanhamento do Projeto - RAP. O Projeto MPS.BR conta com duas estruturas de apoio: o Fórum de Credenciamento e Controle - FCC e a Equipe Técnica do Modelo - ETM. Registro da logomarca MPS.BR, registro das marcas MPS.BR, MR-MPS, MA-MPS e MN-MPS e registro de direitos autorais (<i>copyright</i>) dos Guias e cursos oficiais do MPS.BR.

Produtos Relevantes

Para fins do Projeto [2.25] do PBQP Software – Ciclo 2006, são produtos e resultados relevantes:

1) Pedidos de patente e/ou patentes registradas

Registro da logomarca MPS.BR. Registro das marcas MPS.BR, MR-MPS, MA-MPS e MN-MPS. Registro de direitos autorais (*copyright*) dos Guias e Cursos do MPS.BR.

2) Produtos de software gerados

Lançamento do Portal SOFTEX www.softex.br/mpsbr, sempre atualizado, com informações do MPS.BR disponíveis a todos os interessados. Além disto: i) está sendo “customizado” o ambiente CoreKM, para aprimorar o controle das avaliações, avaliadores, IA, implementadores, II, grupos de empresas e IOGE; ii) está sendo desenvolvido um Repositório MPS contendo resultados de desempenho de empresas que adotaram o Modelo MPS.

3) Outros produtos gerados

Até Dez2006, foram credenciadas 15 Instituições Implementadoras – II, em todas as regiões do país exceto na Região NO, mediante convênio assinado com a SOFTEX. Até Dez2008, espera-se exceder muito a meta original: 20 II, em todas as regiões do país.

Até Dez2006, foram credenciadas 2 Instituições Avaliadoras – IA, uma na Região SE e outra na Região NE, mediante convênio assinado com a SOFTEX. Até Dez2008, espera-se exceder a meta original: 15 IA, em todas as regiões do país.

4) Métodos e/ou algoritmos desenvolvidos

Desenvolvimento do Modelo MPS, composto de: Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS), documentado em quatro Guias do MPS.BR (Guia Geral, Guia de Implementação, Guia de Avaliação e Guia de Aquisição). Estes guias, criados e aprimorados anualmente pela Equipe Técnica do Modelo - ETM, com base

nas melhores práticas da Engenharia de Software e em normas e modelos internacionais, estão disponíveis na seção Guias do Portal www.softex.br/mpsbr.

5) Artigos publicados

Descrições mais completas do Projeto MPS.BR e Modelo MPS encontram-se em artigos publicados em renomados eventos no país e no exterior [Weber 2004a, 2004b, 2005a, 2005b e 2006], disponíveis na seção Artigos e Apresentações do Portal www.softex.br/mpsbr. Como se vê nas Referências Bibliográficas, um destes artigos foi premiado como o melhor Artigo Técnico do IV Simpósio Brasileiro de Qualidade de Software (SBQS 2005).

6) Recursos humanos capacitados

Capacitação de especialistas no MPS.BR seja ofertando regularmente cursos oficiais (C1- Introdução, C2 – Implementadores, C3-Avaliadores e C4-Melhoria do Processo de Aquisição de Software) em todas as regiões do país, seja aplicando semestralmente provas individuais em todo o país, seja organizando anualmente workshops de reciclagem. 30 instrutores foram treinados e estão autorizados a ministrar os cursos oficiais do MPS.BR. Mais de 2.100 pessoas participaram de cursos oficiais do MPS.BR, pagando taxas de inscrição acessíveis. Dentre estes, os representantes das empresas nas respectivas equipes de avaliação MA-MPS. Mais de 600 especialistas foram aprovados em provas individuais do MPS.BR (P1-Introdução, P2-Implementadores, P3-Avaliadores; P4-Melhoria do Processo de Aquisição de Software). As questões das provas aplicadas estão disponíveis na seção Cursos-Provas do Portal www.softex.br/mpsbr. Além disto, foram certificados 2 Consultores de Aquisição (CA) de software e serviços correlatos.

7) Dissertações e/ou teses geradas

Embora não haja acompanhamento direto disto no Projeto MPS.BR, sabe-se que há trabalhos de fim de curso de especialização, dissertações de mestrado e teses de doutorado, geradas ou em andamento, fortemente correlacionadas com a implementação e avaliação do Modelo MPS, principalmente em Universidades que têm participação ativa na Equipe Técnica do Modelo – ETM, tais como COPPE/UFRJ, UCB, Universidade de Lavras e UFRPE.

8) Lançamento de publicação

Lições aprendidas com a implementação e avaliação do Modelo MPS em empresas foram publicadas em duas edições especiais MPS.BR da Revista ProQualiti: volume 1, número 2, Nov2005; volume 2, número 2, Nov2006.

9) Implementação do Modelo MPS em empresas

Até Dez2006, implementação do Modelo MPS em cerca de 120 empresas, em todas as regiões do país. Dentre estas, coordenadas por Instituições Organizadoras de Grupos de Empresas – IOGE, 93 empresas (52% pequenas, 33% médias e 15% grandes empresas) estão usando o Modelo de Negócio cooperado, conforme o COMUNICADO SOFTEX MPS 20/2005.

Estes grupos de empresas estão comprometidos a concluir a implementação do MR-MPS em até 12 meses, a contar da data de assinatura do convênio entre a SOFTEX e a respectiva IOGE, e a se submeterem a avaliação MA-MPS nos três meses subsequentes. Na seção Implementações do Portal www.softex.br/mpsbr estão publicadas as implementações em grupos de empresas ora em curso.

Até Dez2008, espera-se implementar o Modelo MPS em mais 160 empresas.

10) Avaliação do Modelo MPS em empresas

Foram realizadas 17 avaliações oficiais MA-MPS. Dentre estas, 34% são micro e pequenas empresas, 6% médias empresas e 60% grandes empresas; sendo 18% no nível A (o mais alto), 18% níveis D e E, e 64% níveis F e G. Na seção Avaliações do Portal www.softex.br/mpsbr estão publicados os resultados oficiais das avaliações MA-MPS já realizadas, com validade de três anos a contar da data da sua conclusão. Até Dez2007, espera-se ter um total de 80 empresas com avaliação oficial MA-MPS; até Dez2008, espera-se totalizar 160 empresas avaliadas oficialmente.

Aplicabilidade dos Resultados

Para fins do Projeto [2.25] do PBQP Software – ciclo 2006, consideram-se os seguintes aspectos na aplicabilidade dos resultados do Projeto MPS.BR: relevância, abrangência e impacto dos resultados obtidos.

Relevância

A Melhoria de Processo do Software Brasileiro (MPS.BR) é um objetivo estratégico da SOFTEX, relevante no capítulo de software na PITCE - Política Industrial, Tecnológica e de Comércio Exterior

Abrangência

Projeto mobilizador, com abrangência nacional (todas as regiões do país).

É um projeto multiinstitucional, coordenado pela SOFTEX, com a participação de renomadas instituições na Equipe Técnica do Modelo – ETM (coordenada pela Prof Ana Regina Cavalcanti da Rocha/COPPE UFRJ) e no Fórum de Credenciamento e Controle – FCC (Universidade-Governo-Indústria/SOFTEX).

Impacto dos resultados obtidos

Projeto mobilizador, com grande poder de difusão e adesão junto às mais de mil empresas associadas aos Agentes SOFTEX no país, dentre outras.

Na formação do custo de implementação do MR-MPS e de avaliação MA-MPS em empresas, considerou-se como premissa a necessidade destes serem acessíveis à micro, pequena e média empresa de software no Brasil, seja pela disponibilidade de Instituições Implementadoras – II e Instituições Avaliadoras – IA em todas as regiões do país (implicando em presença local com menor custo de deslocamento e hospedagem), seja por basear-se na moeda corrente no país (não em dólares), seja pelo modelo de negócio adotado (tanto de serviços específicos para uma

empresa como de um pacote de serviços a ser compartilhado por um grupo de empresas).

É interessante observar que o foco inicial do Projeto MPS.BR eram as pequenas e médias empresas (PME) implementarem e avaliarem o Modelo MPS nos seus níveis mais baixos (níveis G e F) – usando o Modelo de Negócio Cooperado (MNC); entretanto, por razões de mercado, o Modelo MPS também começou a ser adotado em grandes organizações públicas e privadas no Brasil que implementaram e já foram avaliadas em seus níveis mais altos (E até A) – usando o Modelo de Negócio Específico (MNE).

Características Inovadoras

Projeto mobilizador, inédito e inovador.

Desde o início, o Projeto MPS.BR sempre deixou claro não ter como objetivo definir algo novo no que se refere a normas e modelos [Weber 2004a e 2004b].

Sua novidade está na estratégia seja de criação e aprimoramento do Modelo MPS seja de disseminação e adoção do Modelo MPS em empresas, ambas adequadas à realidade brasileira.

Conclusão e Perspectivas Futuras

Para fins de complementação do relatório final do Projeto [2.25], do PBQP Software - ciclo 2006, este artigo descreveu o Projeto MPS.BR, com destaque para os principais resultados alcançados no período Dez2003-Dez2006 (Tabela 1). Em síntese, neste período, dois grandes resultados alcançados comprovam o sucesso do MPS.BR: i) tecnicamente, foram criados e vêm sendo aprimorados anualmente um Modelo de Referência (MR-MPS) e um Método de Avaliação (MA-MPS) de processos de software - o que não é algo trivial em qualquer lugar do mundo; ii) do ponto de vista do mercado, fazendo uso do Modelo de Negócio (MN-MPS), houve melhoria dos processos de software no Brasil com a implementação do MR-MPS em cerca de 120 empresas em todas as regiões do país até Dez2006; das quais 17 unidades organizacionais foram bem sucedidas em avaliações MA-MPS no período Set2005-Dez2006 – o que deve se acelerar a partir de 2007.

No período Jan2007-Jun2009, de consolidação do MPS.BR, os principais desafios são: i) comprometer com o Projeto MPS.BR e o Modelo MPS mais pessoas qualificadas, instituições experientes, empresas e organizações interessadas; ii) continuar aprimorando o Modelo MPS e publicar anualmente novas versões dos Guias do MPS.BR; iii) realizar os calendários anuais de Cursos, Provas e Workshops do MPS.BR, em todas as regiões do país; iv) credenciar novas Instituições Implementadoras – II e Instituições Avaliadoras – IA, em todas as regiões do país; v) obter mais recursos para apoiar novas Instituições Organizadoras de Grupos de Empresas – IOGE na criação de novos grupos para implementação e avaliação do Modelo MPS, com foco nos níveis G e F do Modelo MPS, em todo o país; vi) até Dez2008, implementar o MR-MPS em mais 160 empresas e totalizar 160 avaliações oficiais MA-MPS.

Resumindo, o Projeto MPS.BR tem contribuído significativamente para a melhoria de processos de software em todas as regiões do Brasil, tanto em grandes organizações públicas e privadas como nas pequenas e médias empresas (PME), a um custo acessível. Após três anos de realização (Dez2003-Dez2006), este é um empreendimento magno no setor de software Brasileiro, com forte interação Universidade-Empresas-Governo, com muitos resultados alcançados, ativos construídos e valor agregado. Em seguida, há grandes e novos desafios a superar para consolidá-lo no período Jan2007-Jun2009.

Referências Bibliográficas

Bush, M., Dunaway, D. CMMI Assessments: Motivating Positive Change. Addison-Wesley, 2005.

Chrissis, M. B., Konrad, M., Shrum, S. CMMI: Guidelines for Process Integration and Product Improvement. Addison-Wesley, 2003.

ISO/IEC 12207:1995/Amd 1:1998/Amd 2:2002. Technology Information – Software Life Cycle Processes.

ISO/IEC 15504. Technology Information – Process Assessment. Part 1 – Concepts and vocabulary; part 2 – Performing an assessment; part 3 – Guidance on performing an assessment; part 4 – Guidance on use for process improvement and process capability determination; and part 5 – An exemplar process assessment model.

MCT/SEPIN – Ministério da Ciência e Tecnologia, Secretaria de Política de Informática. Qualidade e Produtividade no Setor de Software Brasileiro 2001. Brasília: 2001.

Veloso, F., Botelho, A. J., Tschang, A., Amsden, A. Slicing the Knowledge-based Economy in Brazil, China and India: a tale of 3 software industries. Report, MIT – Massachusetts Institute of Technology, Mass, September 2003.

Weber, K. C., Rocha, A. R., Alves, A., Ayala, A. M., Gonçalves, A., Paret, P., Salviano, C., Machado, C. F., Scalet, D., Petit, D., Araújo, E., Barroso, M. G., Oliveira, K., Oliveira, L. C. A., Amaral, M. P., Campelo, R. E. C., Maciel, T. Modelo de Referência para Melhoria de Processo de Software: uma abordagem brasileira, In: Proceedings of the XXX Conferencia Latinoamericana de Informatica (CLEI 2004). Arequipa, Peru: septiembre 2004.

Weber, K. C., Rocha, A. R., Rouiller, A. C., Crespo, A., Alves, A., Ayala, A., Gonçalves, A., Paret, B., Vargas, C., Salviano, C., Machado, C., Scalet, D., Petit, D., Araújo, E., Maldonado, J. C., Oliveira, K., Oliveira, L. C., Girão, M., Amaral, M., Campelo, R., and Maciel, T. Uma Estratégia para Melhoria de Processo de Software nas Empresas Brasileiras, In: Proceedings of QUATIC'2004 (5th Conference for Quality in Information and Communications Technology), pp. 73-78. Porto, Portugal, October 2004.

Weber, K. C., Araújo, E., Machado, C. F. M., Scalet, D., Salviano, C. F., Rocha, A. R. C. Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software – versão 1.0 (MR-MPS e MA-MPS), In: Anais do IV Simpósio Brasileiro

de Qualidade de Software (SBQS 2005). Porto Alegre, Brasil: junho de 2005. Prêmio de melhor Artigo Técnico do SBQS 2005.

Weber, K. C., Araújo, E. R., Rocha, A. R.C., Machado, C. F., Scalet, D., and Salviano, C. F. Brazilian Software Process Reference Model and Assessment Method, In: P. Yolum et al. (Eds.), Proceedings of ISCIS 2005 (20th International Symposium on Computer and Information Sciences), LNCS 3733, pp. 402-411. Istanbul, Turkey, October 2005. Copyright Springer-Verlag, Berlin Heidelberg, 2005.

Weber, K.C., Araújo, E. R., Rocha, A.R.C., Oliveira, K. M., Rouiller, A.C., von Wangenheim, C. G., Araújo, R., Salviano, C.F., Machado, C.F., Scalet, D., Galarraga, O., Amaral, M.P., e Yoshida, D. Melhoria de Processo do Software Brasileiro (MPS.BR): um programa mobilizador, In: Proceedings of the XXXI Conferencia Latinoamericana de Informatica (CLEI 2006). Santiago, Chile: agosto 2006.

[2.29] Metodologia de Gerenciamento de Projetos AdvancedIT

Entidade: Advanced Database & IT Sistemas de Informação S/A - AdvancedIT

Autor: MSC Roberto Petry, PMP - roberto.petry@advancedit.com.br

Resumo

Este documento apresenta um resumo de avaliação dos resultados do projeto Metodologia de Gerenciamento de Projetos AdvancedIT. O projeto objetivou o desenvolvimento de uma metodologia baseada no Modelo de Maturidade CMMI/MPS.BR, o que propiciou uma melhoria nos processos de desenvolvimento de software, da corporação e permitiu estruturar uma gestão de projetos baseada em indicadores. Também foram atingidos os objetivos de avaliação formal como um diferencial competitivo para mercado nacional e contratos internacionais, a qual foi centrada no Nível F do Modelo MPS.BR.

Palavras-chave: Gerenciamento de Projetos de Software; CMMI; MPS.BR.

1. Introdução

Nos dias de hoje as empresas brasileiras dispõem de uma razoável oferta de modelos de referência e normas de qualidade para auxiliar na definição de processos que assegurem qualidade nos seus produtos e serviços. Dentre eles, o modelo de maturidade CMMI – *Capability Maturity Model Integration* do SEI – *Software Engineering Institute* e o PMBOK – *Project Management Body of Knowledge* do PMI – *Project Management Institute*, estão entre os mais utilizados pelas empresas de desenvolvimento de software.

A adoção tão significativa dos modelos citados deve-se, principalmente, ao reconhecimento obtido dos mesmos em mercados compradores internacionais, às ações do governo brasileiro que visam a exportação de software, ao alinhamento dos modelos aos objetivos de negócio das empresas e as iniciativas de

disseminação de conhecimentos e formação de profissionais, promovidas por parceiros credenciados.

Porém, no cenário de empresas de TI locais no RS, este quadro apresenta um pequeno número de adesões a modelos formais baseados no CMMI, principalmente centrado em grandes empresas e/ou empresas multinacionais sediadas neste estado.

2. Objetivos e Justificativa

O objetivo geral do projeto era o desenvolvimento de uma metodologia baseada no Modelo de Maturidade CMMI/MPS.BR, que propicia-se a melhoria contínua nos processos de desenvolvimento de software da empresa. Também foi proposto o desenvolvimento de um sistema de apoio ao gerenciamento de projetos, automatizando o processo descrito e propiciando uma gestão de projetos baseada em indicadores.

Por fim, para permitir que a empresa tenha um crescimento consistente e sustentado foi definido como objetivo estratégico a obtenção de uma avaliação formal no modelo MPS.BR Nível F, como um diferencial competitivo para mercado nacional e futuros contratos internacionais.

Visto que os modelos de desenvolvimento de software similares ao CMM ainda são muito insipientes em nosso mercado local (RS) e a prática é reconhecida e amplamente utilizada nos grandes mercados internacionais, este foi o desafio estruturado neste projeto.

3. Metodologia de Execução e Resultados Gerados no Projeto

Inicialmente propôs-se montar um projeto de implementação da Metodologia baseada no modelo SW-CMM, visto que a empresa havia integrado um grupo cooperado para implementação do Modelo SW.CMM Nível 2. Porém, durante o projeto, surgiu a possibilidade de integrar a empresa em um grupo desenvolvido pela Softsul e SW-Process¹ de implementação e avaliação do MPS.BR².

O projeto também contou em sua parte final com apoio da SOFTEX e bolsas de iniciação científica concedidas pelo RAHE para atividades dentro do projeto.

3.1 O Processo de Desenvolvimento de Software

O principal objetivo do Processo de Desenvolvimento de *Software* (PDS) da Advanced IT é definir, melhorar e aprimorar as práticas de gerenciamento de projetos e de engenharia de *software* e, por consequência, capacitar a empresa nas práticas do MR-MPS.

Como no mercado existem várias maneiras de se vender um projeto, a escolha da Advanced IT foi que somente vendas a preço fechado devem compor o escopo de implementação e avaliação, podendo envolver serviços de levantamento, análise, projeto, desenvolvimento e implantação.

Ainda ficou definido que podem ser assumidos dois casos de negócio:

- Desenvolvimento de novas aplicações



- Manutenção ou customização de aplicações existentes, que podem ser próprias ou de terceiros.

Para a documentação e disseminação do PDS a Advanced IT criou diversos documentos estruturados, dentre eles o documento “P00 – Processo de Desenvolvimento de Software Advanced IT”, contendo explicações sobre políticas, papéis e responsabilidades, processos, formulários, manuais, tabelas de referência, anexos de formulários e os diagramas dos processos, bem como apresentando o índice dos documentos do processo.

A diagramação dos fluxos dos processos exige que se tenha a definição dos papéis e responsabilidades dos participantes. Então, ainda dentro do documento “P00 – Processo de Desenvolvimento de Software Advanced IT”, eles estão definidos. Veja abaixo os principais papéis:

- **DP - Diretor de Projeto:** É o papel de máxima responsabilidade sobre os projetos dentro da empresa, normalmente desempenhado por um dos sócios.
- **GCC - Gerente de Contas Corporativas:** é o papel de máxima responsabilidade comercial sobre os projetos junto ao cliente.
- **GP - Gerente de Projeto:** tem o papel de responsabilidade sobre um conjunto de projetos da empresa e representante da empresa junto ao cliente.
- **AC – Analista de Configuração:** tem o papel de dar suporte a gerência de configuração dos projetos.
- **GQ - Gerente de Qualidade:** Tem responsabilidade de controlar as atividades da garantia da qualidade.
- **CCC - Comitê de Controle da Configuração:** O CCC é responsável por autorizar a criação e reabertura das baselines do projeto.
- **LP - Líder de Projeto:** Papel de responsabilidade sobre um determinado projeto da empresa, podendo representar a empresa junto ao cliente.
- **AProc - Auditor de Processo:** é responsável por participar da elaboração do Plano de Garantia da Qualidade e auditar periodicamente o projeto de acordo com o estabelecido no Plano de Garantia da Qualidade do Projeto, buscando evidências da correta e completa realização dos processos gerenciais de desenvolvimento de *software* da Advanced IT.
- **AProd - Auditor de Produto:** é responsável por apoiar a elaboração do Plano de Garantia da Qualidade e auditar periodicamente o projeto.
- **ST - Suporte Técnico:** tem responsabilidade sobre a infraestrutura de hardware e *software* e apoio técnico às equipes dos projetos.

Além dos papéis e responsabilidades adotados dentro da empresa, existem outros

papéis que se referem a participante de fora da empresa. Veja abaixo os principais deles:

- **SP - Sponsor:** papel de máxima responsabilidade sobre os projetos no cliente, normalmente desempenhado por um Diretor ou Gerente de Informática.
- **RT - Responsável Técnico:** tem o papel de responsabilidade técnica sobre os projetos no cliente.
- **US - Usuário-chave:** tem o papel de responsabilidade negocial sobre os projetos dentro do cliente.

O foco da Advanced IT é obter uma gestão de projetos baseada em indicadores, que se fazem necessários para a melhoria contínua do PDS. Então foram definidos processos, de acordo com o MR-MPS, e que no CMM são chamados de áreas chaves, a fim de adquirir primeiramente o nível F do MR-MPS, contemplando os processos de Gerência de Requisitos, Planejamento de Projetos, Garantia da Qualidade de *Software*, Acompanhamento e Supervisão de Projetos, Gerência de Configuração, Aquisição e Medição.

Cada um destes processos está definido em um documento contendo as políticas e as etapas. Cada etapa também é definida neste documento, apresentando os objetivos, elaborador, entradas, definição do procedimento, as saídas e, quando existirem, os itens que farão parte da medição.

A partir disto, tem-se uma estrutura um pouco mais definida do PDS da Advanced IT, contendo o processo principal, demais processos e suas etapas. Com isto, já se pode apresentar outros principais documentos do processo, que já foram citados anteriormente, como os formulários, manuais, tabelas de referência e anexos de formulários, numa estrutura hierárquica. Veja abaixo:

- **Processo de Desenvolvimento de Software:** P00 - Processo de Desenvolvimento de Software Advanced IT.
- **Gerência de Requisitos:** P30 – Processo de Levantamento e Análise de Requisitos

Etapas: Contato e Avaliação, Levantamento de Requisitos, Análise de Requisitos, Orçamento de Requisitos, Negociação de Requisitos e Planejamento Prévio.

Principais Documentos:

- F10 – Registro de Contato: encaminhamento das demandas de projetos da área comercial para a área técnica de desenvolvimento de projetos;
- F30 – Documento de Requisitos: detalhamento do escopo e dos principais aspectos relacionados ao projeto.
- F20 – Orçamento: estimativa e orçamentação realizado a partir dos requisitos levantados, através de tabelas de produtividade padrão em cada tecnologia utilizada pela

empresa.

- **Garantia da Qualidade de Software:** P50 – Processo de Garantia da Qualidade

Etapas: Acompanhamento da Garantia da Qualidade, Auditoria do Processo de Requisitos, Planejamento da Garantia da Qualidade, Auditoria de Processo e Auditoria de Produto.

Principais Documentos:

- F51 - Cronograma de Auditorias: Contém as auditorias que deverão ser realizadas durante o projeto.
- F52 – Checklist: Existem diversos tipos de checklists. Dentre eles, auditoria de requisitos, de processo e de produto. Este último, por tecnologia.
- F56 - Proposta de Melhoria de Processos: Descreve os pontos do processo que apresentou dificuldades, assim como possibilidades de melhorias.

- **Planejamento de Projetos:** P60 – Processo de Planejamento de Projetos

Etapas: Planejamento de Recursos, Planejamento do Projeto, Revisão das Estimativas e Planejamento de Riscos.

Documentos:

- F60 – Plano do Projeto: O plano do projeto é o principal documento da etapa de planejamento. Descreve os recursos, os prazos, tempo de projeto, os riscos, as entregas e outras informações do planejamento.
- F61 – Cronograma Detalhado de Atividades: É o cronograma do projeto, descrevendo detalhadamente as atividades, datas, precedências e recursos do projeto.
- F70 – Documento de Análise: onde estão detalhadas as especificações técnicas dos requisitos. Um F70 para cada requisito ou grupo de requisito.

- **Acompanhamento e Supervisão de Projetos:** P65 – Processo de Acompanhamento de Projetos

Etapas: Acompanhamento Operacional, Acompanhamento Gerencial, Acompanhamento nos Marcos do Projeto, Replanejamento e Finalização do Projeto.

Documentos:

- F66 – Acompanhamento Gerencial: Onde se encontram o previsto e realizado do ponto de “Break Even”, esforço e custo por recurso, e andamento do projeto.

- **Gerência de Configuração:**

Etapas: Criação do Repositório do Projeto, Criação e Manutenção da *Baseline* Inicial, Planejamento da Gerência de Configuração, Criação e Manutenção de *Baselines*, Auditoria de Configuração e Encerramento do Projeto.

Documentos:

- F90 – Artefatos do Projeto: tem por objetivo complementar o Plano do Projeto com o planejamento da Gerência de Configuração para o projeto.
- F91 – *Baselines*: tem por objetivo detalhar a criação e manutenção das *baselines* de planejamento, replanejamento, testes e entrega do projeto.
- F92 – Matriz de Rastreabilidade: tem por objetivo manter a rastreabilidade entre os requisitos, artefatos do processo e artefatos do produto.

- **Gerência de Mudanças:** P80 – Processo de Gerência de Mudanças

Etapas: Criação do Repositório do Projeto, Criação e Manutenção da *Baseline* Inicial, Planejamento da Gerência de Configuração, Criação e Manutenção de *Baselines*, Auditoria de Configuração e Encerramento do Projeto.

Documentos:

- F80 – Registro da Mudança: contém as solicitações de mudanças de requisitos.
- F81 – Avaliação da Mudança: contém as especificações técnicas e impactos de cada mudança.

- **Medição e Análise:** as atividades e etapas foram distribuídas entre os demais processos, não tendo sido criado um processo a parte para suportar esta área de processo.

Outro processo que também foi definido é um processo para Gerência de Testes, não sendo um processo explícito do MR-MPS, mas que a empresa Advanced IT considerou fundamental para a garantia de qualidade de seus produtos.

Vistos os processos que fazem parte do projeto de aquisição do MR-MPS, podemos apresentar uma visão geral do processo principal, ou seja, o processo de desenvolvimento da Advanced IT, conforme Figura 1, com uma breve descrição de cada item.

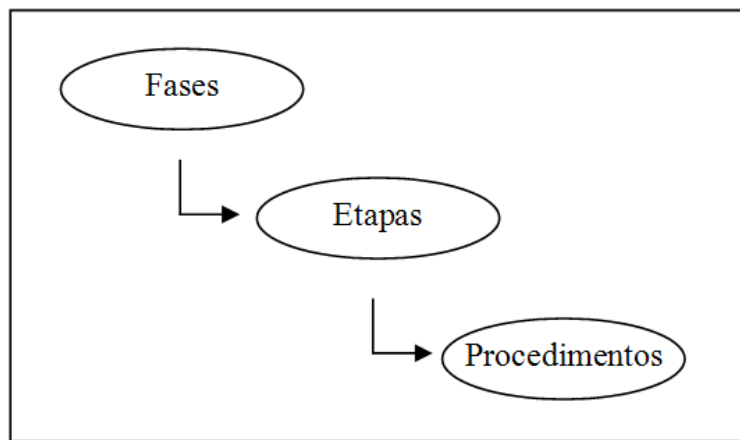


Figura 1 - Visão Geral do PDS da Advanced IT

- **Fases** – apresentam a visão macro das principais fases do PDS da Advanced IT. Exemplos: Levantamento e Análise de Requisitos, Planejamento de Projeto, Gerência de Mudanças, etc.
- **Etapas** – apresentam a subdivisão das principais fases do processo, agrupando os procedimentos necessários para a geração de documentos formais e completos. São identificados os responsáveis pela execução de cada uma das etapas e as entradas e saídas esperadas. Exemplos de etapas são: Contato e Avaliação, Orçamento, Levantamento de Requisitos, Proposta, Planejamento do Projeto, etc.
- **Procedimentos** – apresentam o maior nível de detalhe do processo, informando uma descrição dos procedimentos que deverão ser realizados para o atingimento dos objetivos. Exemplos de procedimentos da Etapa de Levantamento de Requisitos são: levantamento de dados, análise de requisitos, negociação de requisitos, validação de requisitos, etc.

Agora que se tem a visão da metodologia adotada pela Advanced IT, pode-se comparar os processos com as fases, verificando que estas estão teoricamente associadas e que expressam a visão da empresa em refletir os processos do MR-MPS.

Como todos os processos da Advanced IT foram diagramados, na imagem abaixo podemos mostrar o enquadramento do processo (fase) de Gerência de Requisitos e suas etapas, inserindo também os procedimentos executados.

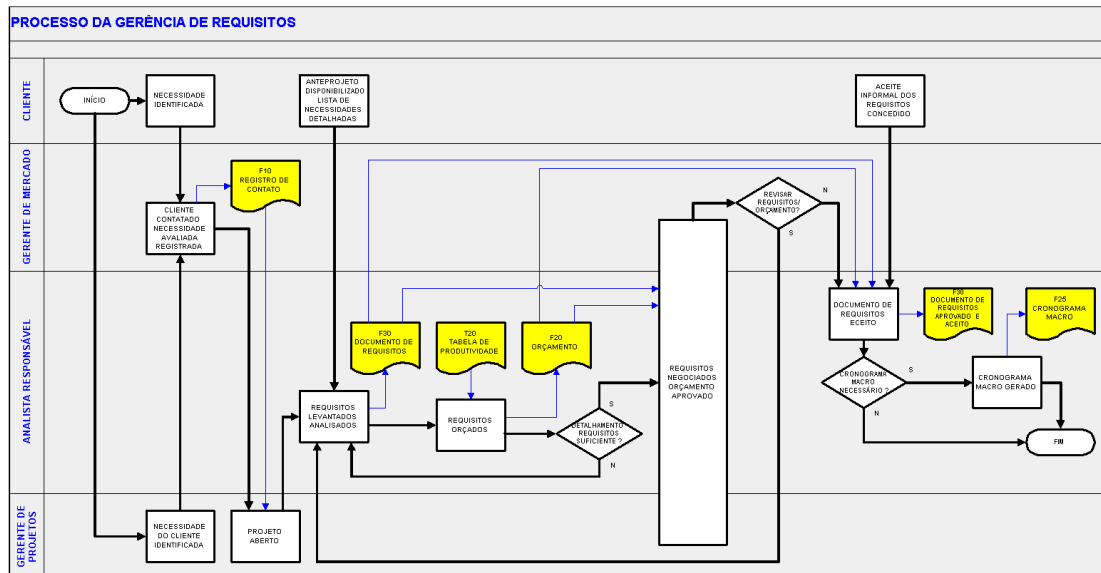


Figura 2 – Fluxo do Processo de Gerência de Requisitos

De acordo com a Figura 2, todos os processos estão diagramados, facilitando a visualização e entendimento dos mesmos, possibilitando a identificação de algumas atividades/procedimentos e seus responsáveis.

1 O Projeto de um Framework para Controle de Projetos

Para a aquisição do MPS.BR Nível F foi necessário a implementação de algumas práticas definidas nos processos de Gerência de Requisitos, Planejamento de Projeto, Acompanhamento e Supervisão de Projeto, Gerência da Qualidade, Gerência da Configuração, Aquisição e Medição, a fim de garantir o nível de capacidade e maturidade de cada um.

A partir disto, a equipe que realizou o diagnóstico apontou o grau de conformidades das práticas utilizadas na organização em relação ao MR-MPS, bem como os pontos fortes e as oportunidades de melhorias. Só então se pode sugerir o desenvolvimento de algumas ferramentas para auxiliar na aquisição do modelo.

Veja abaixo, no Quadro 2, os processos de Gerência de Requisitos e Gerência de Projetos e alguns resultados esperados (RE), elencados como os mais críticos para se implantar, e as idéias iniciais de como determinadas ferramentas poderiam auxiliar na aquisição do MR-MPS.

Quadro 1 - Resultados Esperados de Processos e Sugestões de Ferramentas

Processo	Principais Resultados Esperados do MR-MPS	Onde uma Ferramenta Poderia Auxiliar
Gerência de Requisitos (GRE)	- GRE 3: A rastreabilidade entre os requisitos, os planos do projeto e os produtos de trabalho é estabelecida e mantida - GRE 6: Inconsistências entre os planos do projeto, os produtos de trabalho e os requisitos são identificadas e corrigidas	Com o desenvolvimento de uma aplicação para criação e manutenção de uma matriz de rastreabilidade, tendo a maioria de seus itens (x ou y) gerados automaticamente, apontando de forma gráfica alterações nos itens da matriz, o que incentivaria o uso desta, tornando uma prática corrente dentro da empresa.
Gerência de Projetos (GPR)	- GPR 3. As fases do ciclo de vida do projeto são definidas; - GPR 4. A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. - GPR 5. As tarefas, os recursos e a infraestrutura necessários para completar o trabalho são planejados; - GPR 6. O cronograma e o orçamento do projeto são estabelecidos e mantidos; - GPR 8. Os dados relevantes do projeto são identificados, coletados, armazenados e distribuídos. Um mecanismo é estabelecido para acessá-los, incluindo (se pertinente) questões de privacidade e segurança; - GPR 13. O planejamento do projeto é monitorado no que se refere a cronograma, recursos, riscos, envolvimento dos interessados e dados; - GPR 14. Revisões são realizadas em marcos do projeto conforme estabelecido no planejamento;	Uma vez que o cronograma do projeto estará automatizado em uma ferramenta e todas as atividades estarão ali cadastradas na forma mais granularizada possível, estando também associadas a um procedimento do processo e este associado a uma etapa, temos claramente definido o andamento do projeto em relação às atividades executadas, pendentes e em andamento, sabendo o <i>status</i> da cada processo e etapa, bem como a facilidade do relato destas informações do Líder ao Gerente de Projeto. Isto seria possível, pois o lançamento de horas é dado pontualmente para uma atividade, possibilitando a precisa identificação do tempo gasto para determinado requisito do escopo, sendo obrigatório que toda e qualquer atividade de um projeto esteja pré-definida no cronograma, ocasionando assim a impossibilidade de mascaramento do esforço e custo do projeto.

3.2 Objetivos da Ferramenta

O diagnóstico realizado na Advanced IT apontou alguns tópicos que deveriam ser abordados para que a implementação no MPS.BR tenha sucesso. Isso se deve a algumas oportunidades de melhorias no PDS hoje em execução na empresa.

O projeto de desenvolvimento de um *framework* para controle de projeto tem como objetivo o desenvolvimento de algumas aplicações essenciais para implementar as oportunidades de melhorias encontradas, bem como fornecer sustentação às práticas exigidas pelo MPS.BR descritas no Quadro 1, sendo fundamental para sua aquisição.

3.3 Escopo da Ferramenta

Abaixo segue Figura 3 que ilustra o diagrama de caso de uso geral do *framework*.

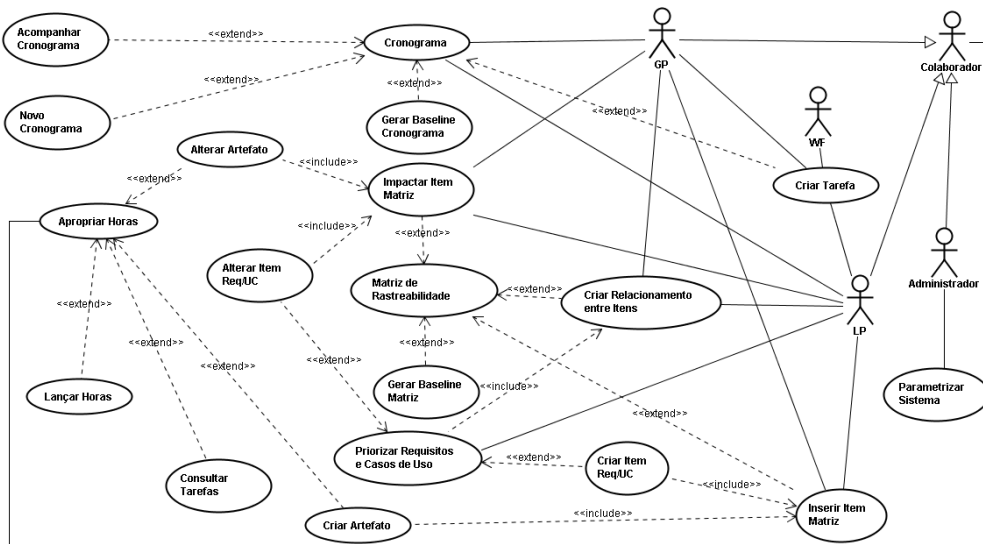


Figura 3 - Diagrama de Casos de Uso

3.4 Arquitetura do projeto do *framework*

Para que fosse possível a criação deste *framework*, algumas tecnologias foram necessárias. A escolha destas tecnologias se deu através de alguns fatores importantes, como, por exemplo, a disponibilidade de tecnologias já existentes dentro da empresa Advanced IT. A Figura 9 esboça as tecnologias empregadas.

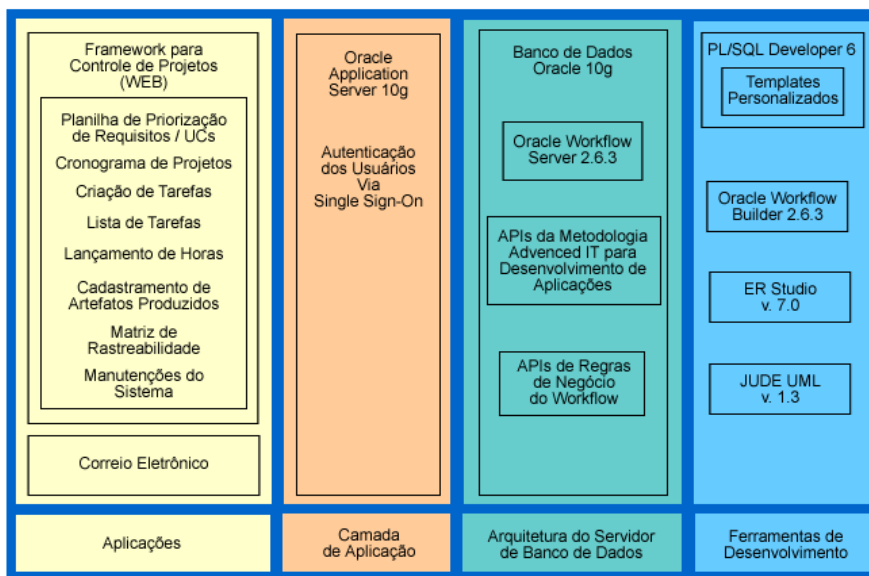


Figura 9 - Arquitetura da Solução

Dada a figura acima, cabe salientar alguns fatores importantes, como, por exemplo, a utilização da autenticação dos usuários através do Oracle Application Server, com uso do Single Sign-On, e também a utilização do Oracle Workflow para controle dos processos e suas regras.

3.5 Implementação da Ferramenta

3.5.1 Planilha de Priorização de Requisitos e Casos de Usos

Esta aplicação tem como objetivo manter o cadastramento dos requisitos funcionais, requisitos não-funcionais e Casos de Usos de um projeto. Ainda, é nesta aplicação que a manutenção entre relações entre estes itens se dá, ou seja, existem relações de “pai para filho” e “de-para” entre os Requisitos Funcionais e Não-Funcionais e Casos de Usos, com algumas regras definidas. Cabe salientar que cada item desta planilha gera um item na Matriz de Rastreabilidade, bem como suas relações, conforme Figura4.

Código	Descrição	Tipo	Tempo(h)	Prioridade	Requisito Pai	Operações
ADVMPs	ADVMPs - MPS BR - implementação do nível F.	RO				
REQ 1	Requisito 1	RF	15		ADVMPs	
UC01	Use case 01	UC			ADVMPs	
UC01.1	Use case 01.1	UC			UC01	
REQ01NF	Requisito 01 não funcional	RNF			ADVMPs	
REQ 1	Requisito 1	RF	15		ADVMPs	
REQ 2	Requisito 2	RF	0		ADVMPs	
UC02	Use Case 02	UC			ADVMPs	
UC02.1	Use case 02.1	UC			UC02	
UC01	Use case 01	UC			ADVMPs	
UC01.1	Use case 01.1	UC			UC01	
REQ02NF-VERSAO2	Requisito 02 não funcional	RNF	0		ADVMPs	
REQ 2	Requisito 2	RF	0		ADVMPs	

Figura 4 - Aplicação da Planilha de Priorização de Requisitos e Casos de Uso

3.5.2 Cronograma de Projeto

Esta aplicação tem como objetivo possibilitar a criação e manutenção do cronograma de um projeto. Além disto, é possível realizar o acompanhamento das atividades, bem como fazer o replanejamento delas.

O funcionamento do cronograma é bastante parecido com a ferramenta Microsoft Project. Essa semelhança não é mera coincidência, pois a primeira idéia antes da criação desta ferramenta era utilizar o Microsoft Project para esta finalidade. Porém, devido à impossibilidade de se customizar o Microsoft Project para aderir a algumas práticas que deveriam ser adotadas, permaneceu no escopo o

desenvolvimento do caso de uso Cronograma de Projeto.

Podemos citar algumas customizações. Por exemplo, o vínculo entre um requisito ou caso de uso do escopo com as tarefas a serem criadas no cronograma. Isto se faz necessário devido ao vínculo entre os artefatos produzidos e os requisitos do escopo. Explicando melhor, quando é criada uma atividade no cronograma, o responsável pelo cronograma tem o poder de gerar uma tarefa para o recurso informado. Assim, essa tarefa aparece na lista de tarefas deste recurso para que este possa lançar as horas gastas naquela atividade, bem como informar no sistema quais os artefatos produzidos, gerando, automaticamente, um vínculo entre o artefato produzido e o requisito informado para aquela atividade no cronograma. Isto é necessário para o cumprimento da prática de criação e manutenção de uma matriz de rastreabilidade bidirecional.

Ainda no Cronograma de Projeto, é possível que o responsável possa buscar no Calendário de Alocação de Recursos o recurso que está disponível para a data de início prevista para a atividade.

Outra funcionalidade do Cronograma de Projeto é a verificação de conflitos entre atividades versus recursos. Isto é bastante importante para que o responsável pelo cronograma não gere atividades sobrepostas para um mesmo recurso, pois o Calendário de Alocação de Recursos somente verifica tarefas geradas, e não atividades de cronograma. Veja maiores detalhes nos itens deste documento que falam sobre a Lista de Tarefas e o Calendário de Alocação de Recursos.

A Figura 5 representa a aplicação do Cronograma de Projeto.

Operações	Início	Fim	Concluído (%)	Tarefa	Requisito	Descrição	Horas	Duração	Data Início	Data Fim	Pred.	Real	Recurso
	75	10	10	ADVMP	ADVMP - ADVMP - MPS BR - implement	8	32	06/04/2006	28/04/2006			10:00	
	50	10	10	UC01.1	Use case 01.1	8	8	06/04/2006	06/04/2006			10:00	Sandra Leis Fiesch
	50	13	10	UC02.1	Use case 02.1	8	8	06/04/2006	06/04/2006			00:00	Robson Schneider de Barros
	100	13	20	REQ.1	Requisito 1	8	8	25/04/2006	06/04/2006			00:00	
	100			UC01.1	Use case 01.1	4	8	25/04/2006	26/04/2006			00:00	Marcos César Zanboni Paul
	100			Tarefa 2		4	8	27/04/2006	28/04/2006			00:00	Marcos César Zanboni Paul

Figura 5 - Aplicação do Cronograma de Projeto

3.5.3 Calendário de Alocação de Recursos

Esta aplicação tem como objetivo detalhar as tarefas que um determinado recurso tem para executar em determinado dia. Temos que salientar que tarefas são diferentes de atividades. As tarefas compõem a Lista de Tarefas do usuário. Já as

atividades estão presentes somente dentro do cronograma. Ou seja, atividades de cronograma podem gerar tarefas da Lista de Tarefas.

Assim, o Calendário de Alocação de Recurso apresenta somente as tarefas dos usuários, não apresentando as atividades de cronograma. Caso o responsável pelo cronograma queira alocar um recurso, ele deve gerar uma tarefa e, se necessário, deixá-la bloqueada temporariamente.

A Figura 6 ilustra a aplicação do Calendário de Alocação de Recursos.



Figura 6 - Calendário de Alocação de Recursos

Nota-se que existem cores diferentes para os dias. Isso se deve pois a cor verde indica que o recurso está com seu tempo totalmente disponível. Já a cor amarela indica que o recurso possui atividades naquele dia, mas ainda tem tempo disponível da sua carga de horário de trabalho. Já a cor vermelha indica que o recurso está com todo seu tempo alocado ou ainda sobrecarregado de atividades naquele dia.

3.5.4 Lista de Tarefas

Esta aplicação tem como objetivo informar as tarefas que um recurso tem para executar, conforme Figura 7.

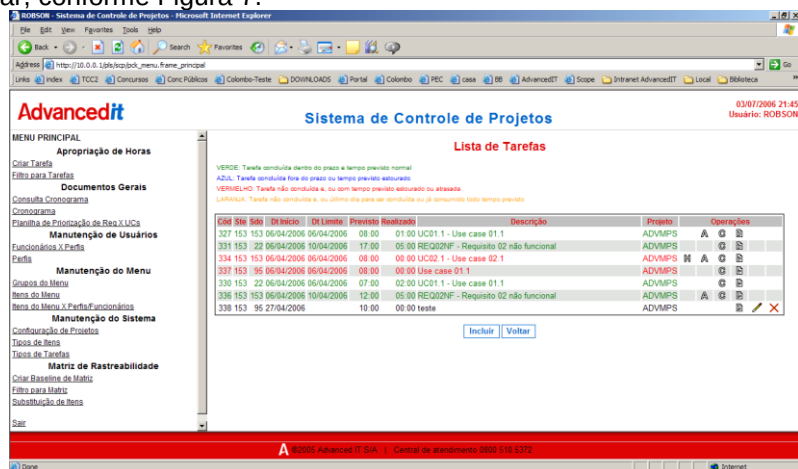


Figura 7 - Lista de Tarefas

Veja que as tarefas da lista possuem cores diferentes. Essas cores servem para alertar o usuário sobre atrasos, tamanho realizado maior que o previsto, tarefas já concluídas, etc. Além disto, na coluna Operações é possível acessar as aplicações de Lançamento de Horas (letra H), Cadastramento de Artefatos (letra A), Cronograma de Projeto (letra C), ou, ainda, possibilidade de consultar os detalhes, alterar ou excluir, sendo que essas duas últimas operações somente quando o usuário tem permissão.

3.5.5 Matriz de Rastreabilidade

Esta aplicação tem como objetivo cumprir uma prática bastante complexa. É a prática que exige que seja criada e mantida uma matriz bidirecional para rastreabilidade entre requisitos, artefatos do processo e artefatos do produto.

É nesta aplicação que podemos identificar, a partir de um item dela, quais os outros itens que estão vinculados a ele. Para que isto fosse possível, foi implementada uma alimentação destes itens praticamente automática, ou seja, todo requisitos da Planilha de Priorização de Requisitos e Casos de Uso viram itens da matriz, incluindo suas relações, que podem ser de “pai-para-filho” ou “de-para”. Além destes itens, os artefatos cadastrados na aplicação de Cadastro de Artefatos também geram itens na Matriz de Rastreabilidade, bem como a relação com os requisitos do escopo que originaram estes itens. Isto é possível porque, para cada atividade do Cronograma de Projeto, deve ser informado um item da Planilha de Priorização de Requisitos e Casos de Uso. Então, após a geração de uma tarefa, somente através desta tarefa que o colaborador pode informar se criou um artefato novo ou alterou um já existente. Caso ele criou um artefato, é gerado um item na Matriz de Rastreabilidade, bem como seu vínculo com o requisito que originou a atividade do cronograma e que, por sua vez, gerou a tarefa da Lista de Tarefas. Caso o usuário altere um artefato, é gerado um alerta na Matriz de Rastreabilidade para que o responsável por ela dê sua devida atenção.

A Figura 8 ilustra a aplicação de Matriz de Rastreabilidade.

Advancedit Sistema de Controle de Projetos
Matriz de Rastreabilidade

Projeto: ADVMP5 - MPS BR - implementação do nível F.

	AAAA	ADVMP5	F70-ADVMP501	REQ_TESTE	REQ_1	REQ2	REQ2NF-VERSAO2	REQ2	REQ2NF-VERSAO2	REQ_1	REQ2	REQ2NF-VERSAO2	REQ_1	REQ2	REQ2NF-VERSAO2
AAAA															
ADVMP5															
F70-ADVMP501															
REQ_TESTE															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2															
REQ_1															
REQ2															
REQ2NF-VERSAO2					</										

Figura 8 - Aplicação da Matriz de Rastreabilidade

7. Conclusões e Perspectivas Futuras

A conquistou no dia 20/12/2006, o Nível F na avaliação do MPS.BR - modelo de referência para melhoria de processos de software brasileiro. Ela foi uma das cinco empresas do 1º Grupo do Projeto Cooperativa MPS.BR – SOFTSUL a obter a certificação.

Dentro outras metas alcançadas com este projeto, podemos citar diversos impactos alcançados com o projeto, o que dividimos em sua classe de contribuição.

Como impactos gerados para a empresa: o desenvolvimento de uma metodologia que propicie a melhoria contínua nos processos de desenvolvimento de software e o desenvolvimento de um sistema de gestão de projetos baseada em indicadores; a avaliação formal do MPS.BR Nível F, gerando um diferencial competitivo para mercado nacional e futuramente exportação; o fortalecimento da parceria com a entidades de ensino e pesquisa, visando o desenvolvimento de inovações tecnológica a disseminação dos resultados obtidos; a maior visibilidade oferecida aos clientes sobre o andamento dos serviços; a agregação de melhores práticas de mercado, estabelecendo padrões mínimos para todos os projetos desenvolvidos;

Como impactos gerados para a sociedade: envolver bolsistas, pesquisadores e alunos da Universidade na pesquisa e aplicação da metodologia de desenvolvimento de projeto de software; qualificar os profissionais da empresa para projetos de pesquisa; ser agente facilitador para a formação de profissionais da empresa no corpo discente da universidade.

Como impactos econômicos: Contribuir para a formação do Cluster de TI, com alto grau de desenvolvimento tecnológico no estado do Rio Grande do Sul; credenciamento para novos mercados internacionais direta e indiretamente (via clientes).

Como impactos tecnológicos e científicos: incentivar o intercâmbio com estudantes/profissionais visando a transferência de tecnologia; permitir a permanente troca de experiências entre o público acadêmico e empresarial; geração de Trabalhos Técnicos, Artigos Técnicos e Trabalhos de Conclusão de Curso; tornar-se um bechmark para o Setor de TI do RS.

A perspectiva futura é consolidar a Metodologia e a Ferramenta no ano de 2007 e planejar uma nova escala de crescimento de maturidade do processo baseado em MPS.BR ou CMMI a partir de 2008.

8. Notas

1 O **Projeto Cooperativa MPS.BR SOFTSUL** é uma iniciativa da SOFTSUL em parceria com a Software Process Consultoria. As empresas que integraram esta iniciativa são pioneiras no RS, na implantação deste modelo que já é referência nacional. Algumas avaliações deverão ser concluídas até o final de janeiro nas demais empresas.

2 O **MPS.BR** é um programa para Melhoria de Processo do Software Brasileiro, está em desenvolvimento desde dezembro de 2003 e é coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX). A iniciativa conta com o apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e Projetos (FINEP) e do Banco Interamericano de Desenvolvimento (BID). Existem hoje 93 empresas em onze Estados brasileiros participando de grupos cooperados para implantação do modelo e que estão sendo apoiadas pela SOFTEX com recursos do BID.

9. Referências Bibliográficas

PROJECT MANAGEMENT INSTITUTE; **PMBOK 3rd Edition – A Guide to the Project Management Body of Knowledge**, 2004. Disponível em: <<http://www.pmi.org>>.

SOFTWARE ENGINEERING INSTITUTE, CARNEGIE MELLON UNIVERSITY. **Capability Maturity Model Integration – SE/SW**, Version1.1. Pittsburgh: SEI, 2002.

BARROS, Robson Schneider & SILVA, Ranusa Dias. **Projeto de um Framework para Controle de Projetos Baseado no MPS.BR**. Trabalho de Conclusão de Curso, Canoas: ULBRA, 2006.

BECKER, Carlos Alberto & PETRY, Roberto. **Utilização Complementar dos Modelos PMBOK e CMMI no Gerenciamento de Projetos de Software**. Porto Alegre: PMIRS, 2006.

10. Autor

Roberto Petry, MSC, PMP (roberto.petry@advancedit.com.br):

. , com foco nas Áreas de Engenharia de Software (Banco de Dados e Gerenciamento de Projetos). RS e SUCESU NACIONAL. Sócio-fundador /A. por .

[2.30] Gestão por Indicadores: medir para evoluir

Entidade: Caixa Econômica Federal - CEF

Autor: Rubens Alves de Souza

1. Introdução



Gerenciar uma empresa sem o mínimo de informações sobre as decisões a serem tomadas é uma atividade difícil. Possuir dados, números e informações sobre os problemas a serem resolvidos é uma ferramenta eficaz que, associada a outras ferramentas, leva o gerente a decidir de forma mais apropriada.

Antes de decidir, o gerente deve ter subsídios para que possa estudar a melhor forma de se resolver um determinado problema. No livro “Gerente Minuto” há um diálogo em que o subordinado deseja resolver um problema e gostaria de saber a opinião do seu gerente. Ao explicar o problema, o gerente pede mais detalhes:

“– Quero dizer – explicou o gerente – que não desejo ouvir somente algo a respeito de atitudes e sentimentos. Conte o que é que está acontecendo em termos observáveis, mensuráveis.” (BLANCHARD; JOHNSON, 2004, p.29).

Todas as informações necessárias para a tomada de decisões estão na própria empresa. O problema é definir como coletar essas informações e agrupá-las de maneira a permitir que as decisões apropriadas sejam tomadas. Há que se definir os indicadores de maneira correta, conforme os problemas a serem resolvidos.

Para entender o que são indicadores: “Indicadores são formas de representação quantificáveis das características de produtos e processos. São utilizados pela organização para controlar e melhorar a Qualidade e o desempenho dos seus produtos e processos ao longo do tempo”.

(TAKASHIMA; FLORES, 1996, p.19).

2. Objetivos e Justificativa

2.1. Objetivos

O trabalho desenvolvido apresenta um Processo de Medição, – baseado no *Practical Software and Systems Measurement (PSM)* - com proposta de definição de Indicadores de Gestão que permitam avaliar o desempenho da Área de Tecnologia e estabelecer metas para sua evolução.

Busca realizar a implementação de um Processo de Medição e acompanhamento de Indicadores de Gestão, de forma sistemática, para obter respostas aos problemas identificados e atingir as metas definidas. Utiliza os princípios descritos no PSM com os devidos ajustes à estrutura e com a customização necessária para a cultura existente na empresa.

2.2. Justificativas

Tomar decisões numa empresa que possui grande volume de dados tratados, sem um mínimo de conhecimento dos números gerados, pode incorrer em decisões erradas.

O Processo de Medição visa facilitar a tomada de decisão com base em números coletados do seu processo interno. Auxilia na tomada de decisões baseada em fatos e direcionados para as metas organizacionais. Busca atingir os objetivos definidos pela empresa.

3. Metodologia de Execução

Descrição da Metodologia:



Antes de se iniciar as atividades centrais previstas no PSM foram realizadas reuniões gerenciais para definição das necessidades de informação e identificar: a) o que precisa ser medido; b) como os resultados devem ser mostrados; c) que metas a empresa possui hoje e d) como as medidas identificadas podem ajudar no alcance das metas.

A metodologia utilizada nesse trabalho utilizou as quatro atividades principais do processo de medição do PSM:

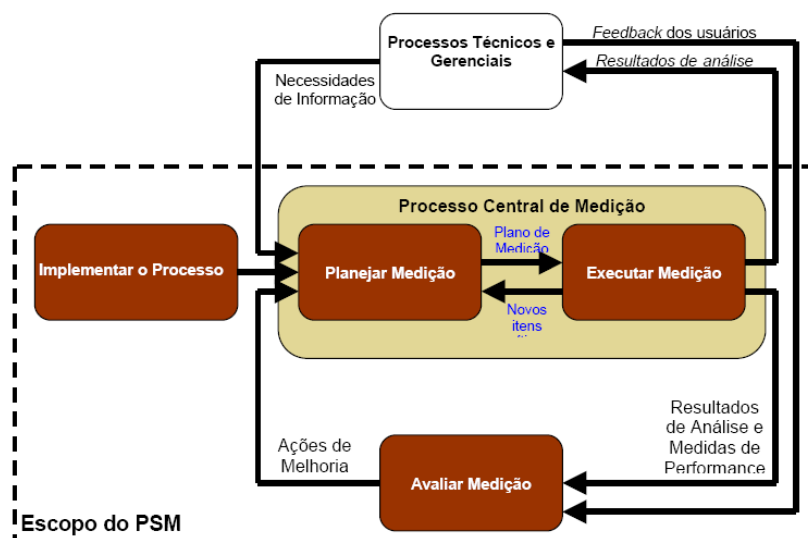
Implementar o processo de medição;

- Planejar a medição;
- Executar a medição e;
- Avaliar a medição.

Implementar o Processo de Medição: foi subdividido nas atividades de: a) identificação das necessidades de informação; b) classificação das informações em categorias e c) criação de um Plano de Medição.

Planejar a Medição: foi subdividido nas atividades de: a) definição das medições, b) identificação e caracterização das medições e c) planejamento do Processo de Medição.

Ex
me
ati
Inc
Av
b)
res
coi
pai
A
im



dos de
Nessas
ico dos
dações,
ntar os
co que
ntam o
ado na

Figura 1 - Processo de Medição do PSM

4. Resultados Obtidos

Recursos humanos capacitados

Para a implementação do processo e avaliação das medições (atividades de coleta e análise), há o trabalho dedicado de 16 recursos humanos; sendo um recurso especialista. Novos recursos serão agregados ao processo, com intuito de aumentar a base o número de indicadores.

Dissertações e/ou teses geradas

Gestão por Indicadores: medir para evoluir. Dissertação apresentada à Caixa Econômica Federal no “1º Concurso de Estudos e Pesquisas para Melhores Práticas em TI”.

Outros resultados

Foram avaliados 43 Indicadores no prazo de um ano (12 meses); com dados consolidados por mês (coleta mensal) o que permitiu os seguintes resultados:

- Formação de uma base histórica dos indicadores (12 meses).
- Revisão dos Acordos de Nível de Serviço.
- Avaliação do volume de demandas recebido mensalmente e capacidade de atendimento.
- Estudo das pendências de atendimento.
- Comparação dos motivos das pendências de atendimento (anos 2005/2006).
- Elaboração de diretrizes para eliminação das pendências de atendimento.
- Elaboração de sugestões em processos internos, para melhorar o desempenho das equipes.
- Segregação das demandas por tamanho (revisão de processo de atendimento diferenciado).

5. Aplicabilidade dos Resultados

Nas análises efetuadas, com os dados obtidos houve os seguintes resultados:

- Formação de uma base histórica dos indicadores (12 meses): permite identificar o que mais precisa ser medido.
- Revisão dos Acordos de Nível de Serviço: permite observar se os acordos de serviço firmados podem ser melhorados.
- Avaliação do volume de demandas recebido mensalmente e capacidade de atendimento: permite discutir com o gestor da informação novas formas de atendimento das demandas e prioridades de atendimento.

- Estudo das pendências de atendimento: permite revisar os processos com vistas a identificar os motivos de pendência, para eliminá-los.
- Comparação dos motivos das pendências de atendimento (anos 2005/2006): permite avaliar a evolução da capacidade de atendimento e se o processo de eliminação de pendências atende às expectativas.
- Elaboração de diretrizes para eliminação das pendências de atendimento: aumentar capacidade de atendimento.
- Elaboração de sugestões em processos internos, para melhorar o desempenho das equipes.
- Segregação das demandas por tamanho: permite revisão do processo de atendimento, de forma a criar processos e atividades conforme tamanho da demanda (sob medida).

6. Características Inovadoras

Trata-se de um Processo de Medição com conceitos de conhecimento do mercado e apresenta quatro ferramentas ajustáveis às necessidades da organização. O destaque do processo é um documento intitulado “Diagnóstico dos Indicadores” que traz um resumo dos resultados obtidos em cada indicador, com menção dos indicadores que ainda não alcançaram as metas.

Esse documento “Diagnóstico dos Indicadores” apresenta as seguintes informações:

- Nome do Indicador
- Resultado obtido na medição
- Média do resultado obtido no ano
- Meta (onde a empresa pretende chegar)
- Percentual de obtenção da meta (quanto já se alcançou com foco no resultado)

7. Conclusão e Perspectivas Futuras

Previsibilidade, transparência, controle, alcance de metas e alinhamento à estratégia da organização são características que um Processo de Medição bem implantado traz à empresa.

Não se pode perder o foco de atingir as metas estratégicas da empresa. Como mencionado por Goldratt e Cox (1997, p. 37): “ – O que estou lhe dizendo é que a produtividade não tem sentido, a menos que você saiba qual é a sua meta [...]”.

Um Processo de Medição com o uso de indicadores associados às metas organizacionais e coleta sistemática dos resultados é uma ferramenta poderosa para a tomada de decisões e o sucesso da empresa.

Futuramente, alguns indicadores serão substituídos e novos indicadores serão agregados ao processo.

Os resultados obtidos com um Processo de Medição consistente, padronizado e com uma linguagem transparente possibilitam a revisão dos Acordos de Nível de Serviço estabelecidos, permitem a revisão dos contratos junto a empresas contratadas, a revisão e estabelecimento de metas exequíveis, baseadas em fatos, com o uso de dados históricos.

O Processo de Medição permite formar uma base histórica de resultados obtidos no ciclo de vida do software e com este resultado fazer uma comparação com os resultados obtidos em outras empresas no Brasil e até, internacionalmente. É uma excelente ferramenta para comparação. Medir para evoluir: sem medição não há evolução.

8. Referências Bibliográficas

KARDEC, Alan; FLORES, Joubert F.; SEIXAS Eduardo. **Gestão estratégica indicadores de desempenho** – Rio de Janeiro: Qualitymark: ABRAMAN, 2002.

TAKASHIMA, Newton Tadachi; FLORES, Mário C. X. **Indicadores da qualidade e do desempenho: como estabelecer metas e medir resultados** – Rio de Janeiro: Qualitymark Ed., 1996.

GOLDRATT, Eliyahu M.; COX Jeff; **A meta: um processo de melhoria contínua**; tradução de Thomas Corbett Neto. – São Paulo: Nobel, 1997.

PSM. Practical Software Measurement: A Foundation for Objective Project Management, v. 4.0b1. **Disponível em:** <http://www.psmc.com/members/default.asp>. Acesso em 04 mai 2005.

[2.35] A Implantação do MPS.BR nível G na IVIA

Entidade: IVIA - Av. Senador Virgílio Távora, 1701 2º andar - Fortaleza - CE

Autores: Edgy Paiva¹, Fábio de Castro Leite¹, Josyleuda Oliveira¹, Luciana Almeida¹ - {edgy.paiva,fabio.leite,josy.oliveira,luciana.almeida}@ivia.com.br

Abstract.

This paper shows the difficulties, the strong and weak points during the MPS.BR implantation at IVIA, a Brazilian software company. Additionally is showed the results of MPS.BR level G evaluation and the gains archived with the software quality.

Resumo.

Este trabalho mostra como ocorreu a implantação do MPS.BR na IVIA, empresa de software cearense, sendo exibidas as dificuldades, os pontos fortes e pontos fracos. Aqui também é exibido o resultado da avaliação e ainda os ganhos do uso da qualidade de software para a melhoria da organização.

1. Introdução

A IVIA é uma instituição de desenvolvimento de software que atua no mercado desde 1996. É certificada ISO 9001:2000 desde novembro de 2003 e possui uma área de pesquisa e desenvolvimento (P&D).

A organização, procurando a melhoria contínua de seus processos, passou a trabalhar com os processos do MPS.BR (Melhoria de Processo de Software Brasileiro) em 2006, mais especificamente com os do nível G, que trata do processo de gerência de projetos e do processo de gerência de requisitos.

Inicialmente foi diagnosticado que seriam necessários alguns ajustes no atual processo da empresa e esses ajustes foram implementados e instanciados para os projetos. Em dezembro de 2006, a IVIA foi avaliada e aprovada com nota máxima, sendo a primeira empresa brasileira a ter todos os requisitos avaliados como totalmente implementados.

Este trabalho descreve o projeto de implantação do nível G do MPS.BR na IVIA e foi dividido nas seguintes seções: na seção 2, são descritos os objetivos e justificativa do projeto. A seção 3 apresenta a metodologia de execução do projeto, ou seja, como ocorreu a implantação do nível G do MPS.BR (2006) na IVIA. Os resultados obtidos

com o projeto são apresentados na seção 4. A seção 5 apresenta a aplicabilidade dos resultados do projeto. Na seção 6, são descritas as características inovadoras do projeto.

A conclusão do trabalho e as perspectivas futuras para a empresa estão na seção 7. As referências bibliográficas são exibidas na seção 8.

2. Objetivos e Justificativa

O principal objetivo da IVIA, com esse projeto, não é só atingir o nível G na avaliação do MPS.BR (2006), mas, principalmente, melhorar os seus processos de software, implantando um modelo compatível com o CMMI (2006), voltado para o mercado brasileiro. A IVIA tem consciência de que, melhorando seus processos, através do MPS.BR, conseqüentemente ela estará melhorando seus produtos e, com isso, aumentando a satisfação de seus usuários. Para uma empresa cearense de capital privado, é um marco significativo.

3. Metodologia de Execução

Inicialmente foi definida a equipe interna na IVIA, responsável pela definição dos processos de gerência de projeto e gerência de requisitos, lembrando que a IVIA já apresentava, antes de iniciar os trabalhos do MPS.BR, uma metodologia baseada no PMBOK (2004) e na ISO 9001 (2000). Era necessário detectar primeiramente quais pontos desta metodologia não atendiam ao nível G, do MPS.BR. Em seguida, foi contratada uma empresa de consultoria em MPS.BR.

Assim, foi realizada uma apresentação da situação dos processos da empresa, na época, para a realização de um diagnóstico conduzido pela equipe de consultoria com entrevistas a membros da equipe de SEPG, análise de documentos e validação com consultor sênior e patrocinador do projeto.

De posse dessas informações, foi elaborado um planejamento da execução dos passos para a implantação do MPS.BR, nível G, na organização.

O diagnóstico foi realizado com projetos concluídos pela IVIA e o seu resultado foi bem satisfatório.

Esse trabalho inicial foi de extrema importância para direcionar os esforços nos

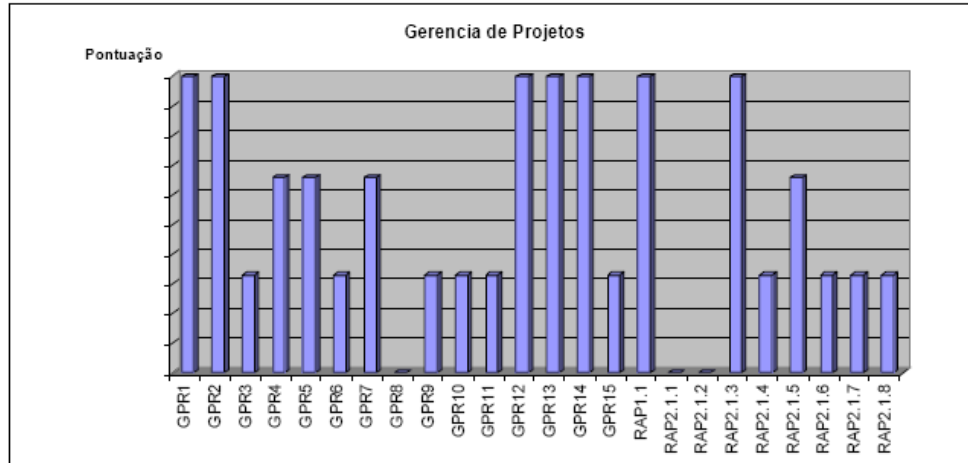


Figura 1 – Avaliação da gerência de projetos na IVIA.

- O escopo do trabalho para o projeto está definido;
- O escopo, os produtos de trabalho e as tarefas do projeto são estimados através de métodos apropriados;
- O envolvimento dos interessados do projeto é planejado;
- O Plano do Projeto é revisado pelos interessados do projeto e o compromisso com o Plano é obtido e;
- Periodicamente, o projeto é monitorado com relação ao Plano do Projeto e o resultado é relatado aos interessados.
- Foram detectados também alguns desvios como:
- O acompanhamento do projeto não está bem claro na metodologia;
- A planilha de estimativa do projeto não contempla infra-estrutura e recursos;

- Os riscos do projeto não são documentados;
- O controle de acesso aos projetos deve ser melhorado;
- Não existe um método de estimativa de tamanho do projeto.

3.2. Diagnóstico da Gerência de Requisitos

Da mesma forma do processo de gerência de projetos, no processo de gerência de requisitos (GRE) todos os itens de resultados esperados (GRE) e todos os atributos de processos (RAP) foram verificados, chegando-se à conclusão listada no gráfico da figura 2.

E com esse trabalho, também se perceberam os seguintes pontos fortes existentes na IVIA antes de implantar o processo de gerência de requisitos:

- uma comunicação contínua com o cliente é estabelecida;
- o entendimento sobre os requisitos é obtido;
- o comprometimento com os requisitos é estabelecido, registrado e mantido;
- mudanças nos requisitos são gerenciadas ao longo do projeto.
- E também foram detectados alguns desvios como:
- não estão definidos critérios objetivos para aceitação dos requisitos;
- não é gerada uma matriz de rastreabilidade bidirecional entre os requisitos, os planos do projeto e produtos de trabalho.

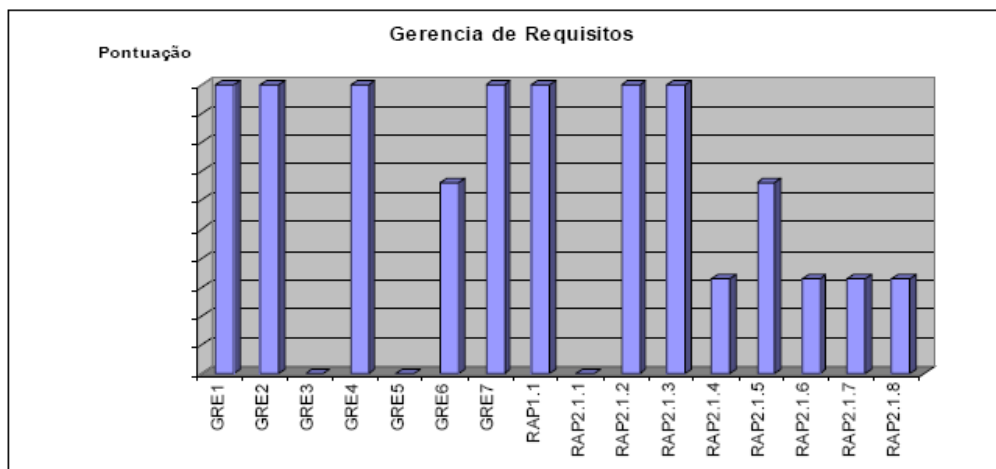


Figura 2 – Avaliação da gerência de requisitos na IVIA.

3.3. Definição dos Processos

Após a conclusão do diagnóstico, iniciaram-se as atividades de adequação dos processos da IVIA para o MPS.BR, nível G.

Assim, foram incluídos alguns procedimentos, itens na política organizacional, inclusão/atualização de papéis e *templates*.

Um exemplo de um novo artefato foi a matriz de rastreabilidade de requisitos, que permitiu a realização da rastreabilidade dos requisitos a partir de um baixo custo para a organização, conforme exibido na Figura 3 abaixo.

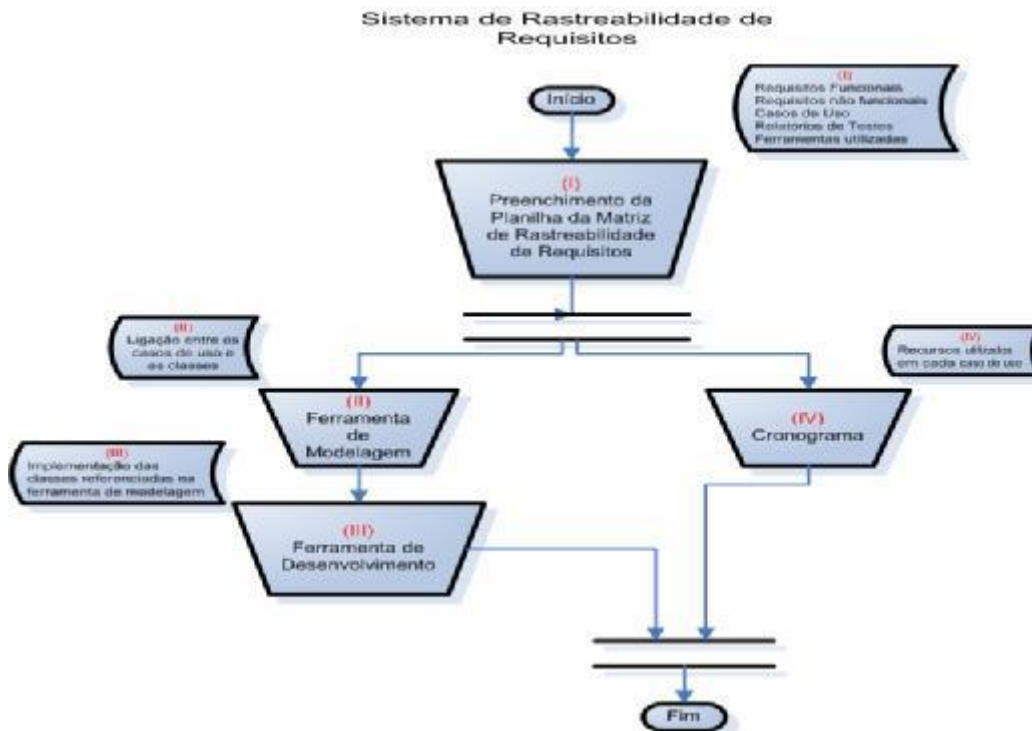


Figura 3 - Fluxo de rastreabilidade dos requisitos da IVIA

A fim de garantir que os processos estariam sendo utilizados conforme descritos na metodologia/política da empresa, foi criada uma equipe de garantia da qualidade que, mediante uma lista de itens, verificava se cada projeto estava ou não seguindo a metodologia da IVIA e, no caso de detecção de não-conformidade, encaminhavam aos responsáveis para a sua correção, dando o devido acompanhamento de tais correções.

4. Resultados Obtidos

Os processos foram implantados inicialmente em dois projetos-piloto, que chamaremos de Projeto A e Projeto B. Em seguida, a consultoria detectou algumas não conformidades e pontos de melhorias que foram implantadas num terceiro projeto (Projeto C). Seguem, abaixo, alguns exemplos das não-conformidades e melhorias detectadas pela consultoria:

- Atas de reunião da equipe do projeto não identificam os papéis dos participantes – Recomendação da consultoria: Identificar os papéis dos participantes da reunião;

- Alguns retornos de aprovações de atas e relatórios não estão ocorrendo – Recomendação da consultoria: Solicitar confirmação/retorno dos artefatos enviados a *stakeholders* para revisão/aprovação (pode ser via e-mail);
- A identificação de riscos é feita com base subjetiva – Recomendação da consultoria: Incluir no template de Plano de Projeto uma lista de riscos comuns nos projetos da IVIA para auxiliar a identificação de riscos dos projetos.

Logo depois, foram implantados os processos em um quarto projeto, a fim de implementar todas as melhorias trabalhadas. Esse último projeto, por ser de grande porte, foi essencial para demonstrar que os processos se aplicavam a todos os projetos da organização.

4.1. Recursos Humanos Capacitados

Os novos processos foram apresentados a todos os colaboradores da empresa através de treinamento, no qual todos foram informados da importância dos novos processos, de que todos os artefatos estariam disponíveis na intranet da empresa e de que as dúvidas poderiam ser esclarecidas pela equipe de garantia da qualidade. Já os gerentes de projeto tiveram treinamentos mais específicos como:

- Treinamento na gerência de requisitos, dando enfoque para a ferramenta matriz de rastreabilidade;
- Project (ferramenta utilizada como apoio na gerência de projetos) e;
- Planilha de estimativa de projetos.

A equipe de SEPG da empresa, hoje composta por três profissionais, participou do curso de introdução ao MPS.BR.

5. Aplicabilidade dos Resultados

Como uma forma de a IVIA avaliar o uso dos processos de qualidade na melhoria do desenvolvimento de software da empresa, foi realizada uma análise organizacional com base no processo de medição da IVIA [Oliveira, 2006a], [Oliveira, 2006b].

Para isso, foram considerados a base histórica da organização antes dos processos do MPS.BR, nível G, bem como os dados colhidos pela organização após implantação do MPS.BR, nível G.

Como exemplo da análise realizada para cada indicador organizacional, será considerado o indicador de estimativa de esforço (figura 4). Para esse indicador, pode-se verificar que, antes do uso de processos de qualidade do MPS.BR – Projetos de A a G –

a estimativa de esforço não atingia o índice mínimo de referência da organização (13,71% - linha horizontal que corta o gráfico)[Oliveira, 2006b]. Após o uso de processos de qualidade do MPS.BR, nível G (Gerência de Projetos e Gerência de Requisitos) – Projetos H e I – pôde-se verificar a melhoria nas estimativas de esforço dos projetos. Ficou evidenciada, portanto, a importância do uso de processo de qualidade no desenvolvimento de software, propiciando, dessa forma,

um estímulo nos colaboradores, a buscarem cada vez mais a melhoria nos processos de desenvolvimento por meio da garantia da qualidade.

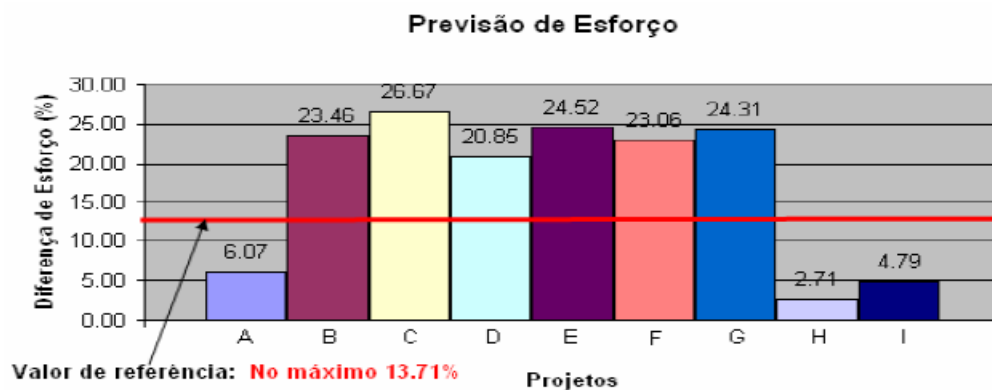


Figura 4 – Análise do Indicador Previsão de esforço.

6. Características Inovadoras

Com esse processo de qualificação, a IVIA espera se tornar mais competitiva no mercado e obter uma melhoria interna, com uma maior previsibilidade dos resultados, maior produtividade, cumprimento de prazos, melhor qualidade do produto final, melhor habilidade para gerenciar complexidade, satisfação do cliente e satisfação da equipe. Seus clientes receberão, em todos os projetos de software desenvolvidos por ela, mais uma segurança: a de que a IVIA segue padrões de confiabilidade, qualidade e segurança.

7. Conclusão e Perspectivas Futuras

Em dezembro de 2006, a IVIA foi avaliada e aprovada com nota máxima no MPS.BR, nível G. A equipe avaliadora divulgou o resultado da avaliação, sugerindo algumas melhorias para os processos de gerência de projetos e gerência de requisitos, bem como algumas melhorias para a organização. Não foram detectados pontos fracos para nenhum dos processos avaliados. Seguem algumas observações divulgadas pela equipe avaliadora.

7.1. Modelo da IVIA

Pontos fortes:

- Possuir um processo de desenvolvimento integrando os processos específicos do MPS.BR.
- Garantia da Qualidade implantada.
- Ter iniciado a implantação de Gerência de Configuração, Medição, e Treinamento.
- Base de lições aprendidas e procedimentos para sugestões de melhorias implementadas.

Oportunidades de Melhoria:

- Aumentar a periodicidade das reuniões do SEPG e Gerência de alto nível.

7.2. Gerência de Projetos da IVIA

Pontos fortes:

- Processo de Gerência do Projeto definido e institucionalizado na organização.

Oportunidades de Melhoria:

- A representação gráfica do Ciclo de Vida é sempre a mesma, independente do modelo de ciclo de vida do projeto. Deveria ser criada uma figura para os diferentes ciclos de vida.
- No plano do projeto, devem constar as responsabilidades pela aprovação dos artefatos. O que há na descrição do processo não é suficiente para todos os artefatos.
- A alocação de colaboradores para os projetos envolve várias consultas na base de competências, além dos cronogramas dos projetos. É necessário um procedimento mais ágil, principalmente com o crescimento da empresa.
- Utilizar um método de estimativa como, por exemplo, pontos de função, pontos de caso de uso.

Resultado Final da avaliação:

- Processo de Gerência de Projeto Satisfeito.

7.3. Gerência de Requisitos da IVIA

Pontos fortes:

- Processo de Gerência de Requisitos definido e institucionalizado na organização.
- Representação da matriz de rastreabilidade possibilitando consulta rápida ao conteúdo de seus relacionamentos.

Oportunidades de Melhoria:

- Gerar versões do documento de requisitos, incluindo as mudanças de requisitos

Resultado Final da avaliação:

- Processo de Gerência de Requisitos Satisfeito.

Como o resultado final da avaliação do processo de gerência de projetos e gerência de requisitos foi satisfeito, a IVIA obteve o nível G do MPS.BR.

Para o segundo trimestre de 2007, a IVIA tem como meta organizacional obter o CMMI, nível 2.

8. Referências Bibliográficas

CMMI (2006). CMMI Product Team. Capability Maturity Model Integration, version 1.1. CMMI for software engineering (CMMI-SW/IPPD, v1.2) staged representation. Software Engineering Institute, 2006.

ISO 9001 (2000) - ABNT – ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 9000:2000 – Sistemas de gestão da qualidade e garantia da qualidade – Fundamentos e Vocabulário. Rio de Janeiro: ABNT, 2000.

MPS.BR (2006), Melhoria de Processo do Software Brasileiro, **Guia Geral**, versão 1.1., Maio Oliveira (2006a), Josyleuda M. M. de; Oliveira, Karlson B. de; Belchior, A. D.

Institutionalization of an Organizational Measurement Process. In: International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 06), IEEE, University of Bridgeport, December.

Oliveira (2006b), Josyleuda M. M. de et al. Application of Multi-Criteria to Perform an Organizational Measurement Process. In: International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 06), IEEE, University of Bridgeport, December.

PMBOK (2004). A guide to the project management body of knowledge. Syba: PMI Publishing Division, 2004. Disponível em: <www.pmi.org>.

[2.36] Melhoria de Processo de Software baseado no MPS.BR Nível G - Um Estudo de Caso

Entidade: Soluções Integradas em Tecnologia de Informação Ltda – SOLTIN e Instituto de Ensino Superior Fucapi - CESF

Autor: Juarez Souza Ávila - juarez_avila@yahoo.com.br

Abstract. *The intention of this article is to stimulate microns, small e average companies (MPEs) of development of software of the Amazonsoft Polar region in Manaus, to improve its processes, using the model of quality of software MPS.BR. Characterized for seven levels of maturity, distributed of the following form: A , B, C, D, E, F and G, being level called G Partially Managed is the initial level for a work of implementation of the Brazilian model. On the basis of this Model, will be shown a case study that displays some steps in the implementation of level G in the Soltin company, will be argued important aspects for the implementation of this level in the company.*

Resumo. O propósito deste artigo é incentivar as micro, pequenas e médias empresas (MPE's) de desenvolvimento de software do Pólo Amazonsoft em Manaus, a melhorar seus processos, utilizando o modelo de qualidade de software MPS.BR. Caracterizado por sete níveis de maturidade, distribuído da seguinte forma: A, B, C, D, E, F e G, sendo o nível G denominado Parcialmente Gerenciado é o nível inicial para um trabalho de implementação do modelo brasileiro. Com base neste Modelo, será mostrado um estudo de caso que expõe alguns passos na implementação do nível G na empresa Soltin, serão discutidos aspectos importantes para a implementação deste nível na empresa.

1. Introdução

Para uma empresa de desenvolvimento de software produzir bons produtos de software, é de suma importância a existência de um processo de desenvolvimento de software. Este cenário é mais fácil de ser encontrado nas grandes empresas de software, pela facilidade de recursos financeiros. Entretanto, no que se refere às MPE's de desenvolvimento de software, a existência de um processo é bastante raro e quando existe não é executado de forma adequada, comprometendo os bons resultados do projeto realizado naquele momento.

Esta pesquisa está alinhada a modelos de melhoria de processo de software, que primam à construção de um produto de software com qualidade e principalmente atendendo a satisfação do consumidor. Será usado como apoio, o modelo Melhoria de Processo do Software Brasileiro (MPS.BR). Este modelo apresenta vantagem relacionada aos demais, o baixo custo em obtenção de uma avaliação, que para as MPE's de desenvolvimento de software é de vital importância.

O propósito é incentivar, motivar as MPE's de desenvolvimento de software do pólo Amazonsoft, a buscarem melhorias em seus processos de desenvolvimento, utilizando para isso, um modelo brasileiro.

2. O Modelo MPS.BR

O foco principal do modelo de qualidade de *software* MPS.BR é proporcionar melhoria do processo de *software* para MPE's de desenvolvimento de software que formam a grande maioria das empresas que desenvolvem *software* no Brasil.

O Guia Geral MPS.BR, baseia-se nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de produtos de *software* e serviços correlatos. Dentro desse contexto, o MPS.BR possui três componentes: Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS).

O Modelo de Referência MR-MPS, diz respeito aos requisitos que os processos das unidades organizacionais devem atender para estar em conformidade com o MR-MPS. Este modelo possui as definições dos níveis de maturidade, processos e atributos do processo.

Alem do Guia Geral, este modelo possui o Guia de Aquisição, trata-se de um documento complementar destinado a organizações que pretendem adquirir *software* e serviços correlatos. O Guia de Aquisição faz referência a boas práticas para a aquisição de *software* e serviços correlatos.

De acordo com o [Guia Geral MPS.BR, 2006], o Modelo de Referência MR-MPS define: a) níveis de maturidade; b) processos e; c) capacidade do processo.

A definição dos processos está em conformidade com a forma apresentada na Emenda 1 da norma [ISO/IEC 12207, 1998], mostrando o propósito e os resultados esperados de sua execução.

Com relação à capacidade do processo, trata-se da habilidade do processo para alcançar os objetivos de negócio, atuais e futuros; estando relacionada com os atributos de processo associados aos processos de cada nível de maturidade.

O MR-MPS será o ponto principal para embasamento teórico deste artigo.

3. Estudo de Caso

Este trabalho, no decorrer do ano de 2006, vem sendo desenvolvido na micro empresa de desenvolvimento de software Soltin, instalada nas dependências do CIDE – Centro de Incubação e Desenvolvimento Empresarial, com o propósito de colaborar para a obtenção do nível G no modelo MPS.BR. Como resultado do esforço empregado, foi submetido e posteriormente aceito no Programa Brasileiro da Qualidade e Produtividade em Software (PBQP – SW), foi apresentado no EQPS BELÉM – Encontro da Qualidade e Produtividade em *Software*.

3.1. Contextualizando a Soltin

A partir de 2004 até os dias atuais, o Amazonas vem participando com assiduidade no PBQP-SW criado em 1993, submetendo projetos em diversas categorias. No ano de 2005, Manaus foi a capital com o maior número de projetos submetidos. Inserida neste contexto, encontra-se a Soltin, caracterizada como micro empresa.

Para dar suporte à melhoria de processos de *software* nas MPE's, o modelo MPS.BR possibilita, como consequência, certificação acessível. Este modelo vem ganhando força em algumas regiões do país, como no Nordeste, Sul e Sudeste. Infelizmente na região Norte a disseminação e utilização do modelo MPS.BR ainda é reduzida. Por este motivo, uma das metas deste trabalho é torna o modelo MPS.BR conhecido primeiramente no universo das empresas de pequeno porte e também no meio acadêmico.

O Amazonas está vivendo um momento inédito no que se refere à participação no PBQP-SW. Essa participação tem sido um dos fatores motivadores que impulsionaram as MPE's em submeter projetos, buscar implantar um processo de *software* e conseqüentemente procurar melhorar seus processos de desenvolvimento de *software*. Para que isso venha a ocorrer é sugerido a utilização do MPS.BR.

O AmazonSoft - Pólo de Desenvolvimento *Software* tem contribuído consideravelmente para essa manifestação de grande parte das empresas do Amazonas no PBQP-SW, situado no CIDE é apoiado pela SUFRAMA – Superintendência da Zona Franca de Manaus, surge como um agente catalisador do cluster de software da região Amazônica, e tem como missão contribuir para o progresso tecnológico-científico, econômico e social da Amazônia e do Brasil, com sua sede em Manaus. O Amazonsoft é um dos programas prioritários da Lei de Informática, e tem como objetivos: promover alianças estratégicas entre empresas associadas e parceiros em pesquisa e desenvolvimento, bem como marketing e vendas, no país e no exterior; favorecer o surgimento de novas empresas de software, particularmente com produtos e serviços baseados em diferenciais regionais; apoiar empresas existentes agregando valor (capacitação, design, marketing, promoção de negócios, comercialização e outros) e facilitar o acesso a financiamentos, dentre outros. Traz benefícios à Zona Franca de Manaus, criando

uma nova alternativa de perfil industrial e agregando valores aos empreendimentos industriais no Pólo Industrial de Manaus – PIM.

3.2. Caracterização da Soltin no Mercado de Software de Manaus

A Soltin Ltda, é uma empresa de desenvolvimento de Sistemas de Informação e participante do AmazonSoft. O campo de atuação da empresa está voltado para Desenvolvimento de Sistemas Integrados, Projetos P & D (Pesquisa e Desenvolvimento), Consultoria Técnica e Tecnologia Web. A Soltin foi a primeira empresa incubada no CIDE a participar de um Simpósio de Grande relevância, no caso o SBQS – Simpósio Brasileiro da Qualidade em *Software*.

Esse feito tem proporcionado bons resultados, pois tal iniciativa deixa transparecer que uma micro empresa pode ter competência e maturidade para produzir *software* atendendo um padrão de qualidade. No entanto, para manter o trabalho em andamento é de suma importância o interesse dos membros que compõem a equipe. Infelizmente a empresa vem sofrendo constantes reformulações no seu quadro de pessoal.

Inserida neste cenário a Soltin, está constantemente buscando meios para adquirir experiência no mercado nacional de desenvolvimento de *software*. Para obtenção de experiência, a Soltin vem participando expressivamente em eventos, dentre eles podemos citar: SBQS de 2003 e participação com projetos no PBQP – SW 2004, 2005 e 2006. Acreditamos que ao participarmos destes eventos, certamente nos trará mais experiências e conseqüentemente espera-se com isso o amadurecimento.

A Soltin tem como objetivo inicial, buscar a aderência do seu processo produtivo de *software* ao nível G do modelo de qualidade MPS.BR, para isso, a empresa tem como apoio recursos do CNPq, publicados através do edital RHAE Inovação 2004.

3.3. Capacitação como foco principal

A busca pela capacitação, vem sendo tratada com paciência, dedicação e otimismo, acreditamos que, ao tornarmos o pessoal apto para o trabalho, o tempo para alcançar o objetivo diminuirá. A Soltin, com incentivo do CIDE vem disponibilizando oportunidades de treinamento em cursos voltados para a área de gestão e afins. O propósito de treinar o pessoal em cursos dessa natureza é de tornar a Soltin uma empresa competitiva inicialmente no mercado Regional e futuramente no mercado Nacional e Mundial.

A participação no curso de Planejamento da Qualidade e Mapeamento de Processos realizado na FIEAM – Federação das Indústrias do Estado do Amazonas, trouxe novos conhecimentos que nos proporcionaram visão relevante sobre processos, ou seja, permitiu por exemplo, através do mapeamento, conhecer o processo da empresa, onde identificamos pontos fracos e fortes, forma de gerenciá-los e também a importância das sugestões da equipe para a melhoria dos processos juntamente com a alta gerencia.

Apesar dos inúmeros obstáculos enfrentados pela empresa em questão, os resultados tem sido satisfatórios, na medida do possível estamos capacitando o pessoal envolvido, para que resultados futuros também possam ser obtidos. Tornar o pessoal apto para o trabalho é extremamente necessário, atividade dessa natureza é primordial para a Soltin, pois, com o pessoal adequadamente capaz de desenvolver determinada tarefa, a obtenção do nível de maturidade G do modelo MPS.BR será inevitável.

Buscar melhoria no processo de desenvolvimento de *software* é uma realidade para a Soltin e para tornar esse objetivo visível há o cuidado em identificar as áreas que estão próximas da nossa realidade. A [NBR ISO 9001:2000] também é um instrumento que nos auxiliará a melhorar nosso processo (GSDS – Gestão Soltin de Desenvolvimento de *Software*), para que isso venha a ocorrer também há preocupação em capacitar o pessoal nesta área.

3.4. Aderência do GSDS ao Nível G do MPS.BR

O GSDS foi projetado como o objetivo de organizar um conjunto de atividades relacionadas com criação e entrega de soluções de *software*. O GSDS define o que fazer (artefatos), quando e como executar as principais atividades de desenvolvimento, além de especificar quem deve estar envolvido no processo (papeis).

O GSDS é um processo em fase de desenvolvimento na Soltin, que por sua vez, vem encontrando dificuldades em estabelecê-lo.

De forma abrangente, um dos fatores que dificultam o estabelecimento do processo GSDS é a imaturidade. Com o intuito de tentar controlar problemas desse nível, a Soltin vem adequando-se as exigências do mercado, aderindo seu processo de desenvolvimento aos processos do MPS.BR.

Com auxílio do [PMBOK, 2004], estamos buscando a melhor forma para definir nossos projetos, dado que este Guia está repleto de conhecimentos dessa natureza.

O processo GSDS está organizado em quatro fases:

- Fase 1 - Análise de Negócio
- Fase 2 - Análise e Projetos
- Fase 3 - Fábrica de Software
- Fase 4 - Implantação

É de suma importância obter um processo de desenvolvimento de *software* e mais importante que isso é buscar melhoria para esse processo de *software*, o modelo MPS.BR proporciona meios acessíveis para uma micro empresa como a Soltin a alcançar melhoria de processo e conseqüentemente certificação neste modelo.

O Modelo de Referência MR-MPS, defini níveis de maturidade, ou seja, trata-se de uma combinação entre processos e capacidade de processos, de acordo com orientação do [Guia Geral MPS.BR, 2006].

Ao iniciar o projeto de implantação do nível G do MPS.BR na Soltin, foi identificado que a empresa deveria aplicar esforços extras, pois, o quadro de pessoal é extremamente reduzido, ou seja, em maio de 2005 eram cinco funcionários, hoje são quatro. Isso vem se tornando um desafio para a empresa.

No modelo MPS.BR, para obter o nível G, faz-se necessário atender aos resultados esperados dos seguintes processos: Gerência de Projetos e Gerência de Requisitos. Antes de iniciarmos o trabalho de implantação do Nível G, foi observado pela equipe da Soltin, que as atividades realizadas dentro da empresa, estavam aderentes em 42% no que se refere aos resultados esperados do processo de Gerência de Projeto e 43% dos resultados esperados do processo de Gerência de Requisitos, ambos caracterizados como parcialmente implementado. Hoje, no que se refere aos Artefatos empregados para os dois processos do Nível G, tem-se aderência igual a (100%), para visualizar melhor esta situação, segue abaixo nas (Tabelas 1 e 2 respectivamente) a aderência entre os artefatos do processo GSDS e os resultados esperados para os Processos de Gerência de Projetos e Gerência de Requisitos.

Processo	Resultado Esperado	Artefato do GSDS	Em (%)
Gerência de Projetos	GPR 1	Proposta de Especificação de Software (Escopo do projeto);	100
	GPR 2	EAP – Estrutura Analítica do Projeto;	100
	GPR 3	Documento de Planejamento do Projeto;	100
	GPR 4	Documento de Planejamento do Projeto (Plano de Riscos);	100
	GPR 5	Documento de Planejamento do Projeto;	100
	GPR 6	Documento de Planejamento do Projeto;	100
	GPR 7	Documento de Planejamento do Projeto (Plano de Riscos);	100
	GPR 8	Documento de Planejamento do Projeto (Plano de comunicação);	100
	GPR 9	Documento de Planejamento do Projeto (Plano de Equipe);	100
	GPR 10	Documento de Planejamento do Projeto (Cronograma Geral);	100
	GPR 11	Documento de Planejamento do Projeto (Plano de Comunicação);	100
	GPR 12	Ata de Reunião;	100
	GPR 13	Documento de Planejamento do Projeto (Plano de Equipe);	100
	GPR 14	Documento de Planejamento do Projeto (Cronograma);	100
	GPR 15	Ata de Reunião (Devidamente analisada pela gerência);	100
	GPR 16	Ata de Reunião (com as ações corretivas dadas pela gerência).	100
Total			100

Tabela 1 - Aderência do processo GSDS ao processo Gerência de Projeto do MPS.BR nível G. - Fonte: Soltin Ltda.

Processo	Resultado Esperado	Artefato do GSDS	Em (%)
Gerência de Requisitos	GRE 1	Plano de Comunicação (Ata de Reunião; Carta Proposta, etc).	100
	GRE 2	Documento de Especificação de Requisitos do <i>Software</i> ;	100
	GRE 3	Plano de Aprovação (Plano de Aceitação).	100
	GRE 4	Documento de Especificação de Requisitos do <i>Software</i> ;	100
	GRE 5	Matriz de Rastreabilidade;	100
	GRE 6	Controle de mudanças (Documento de Consistência dos Requisitos).	100

	GRE 7	Planilha de Controle de Mudanças ou Plano de Ação.	100
Total			100

Tabela 2 - Aderência do processo GSDS ao processo Gerência de Requisitos do MPS.BR nível G. - Fonte: Soltin Ltda.

3.5. Gerência de Processo de Software Seguindo o Ciclo PDCA

O ciclo PDCA é uma das metodologias mais conhecidas utilizadas para dar base a sistemas da qualidade, como métodos de análise e solução de problemas, como método de melhoria contínua e até para definição de novos processos.

Como a metodologia de PDCA é um ciclo “infinito”, este é amplamente utilizado em processos de melhoria contínua, das quais os resultados de um ciclo podem ser referências para o giro do novo ciclo [DEMING, 1990]. A idéia de ciclo contínuo permite ao usuário descrever como “giro do PDCA”.

O PDCA é formado por quatro etapas, representadas de acordo com a (Figura 1) abaixo.

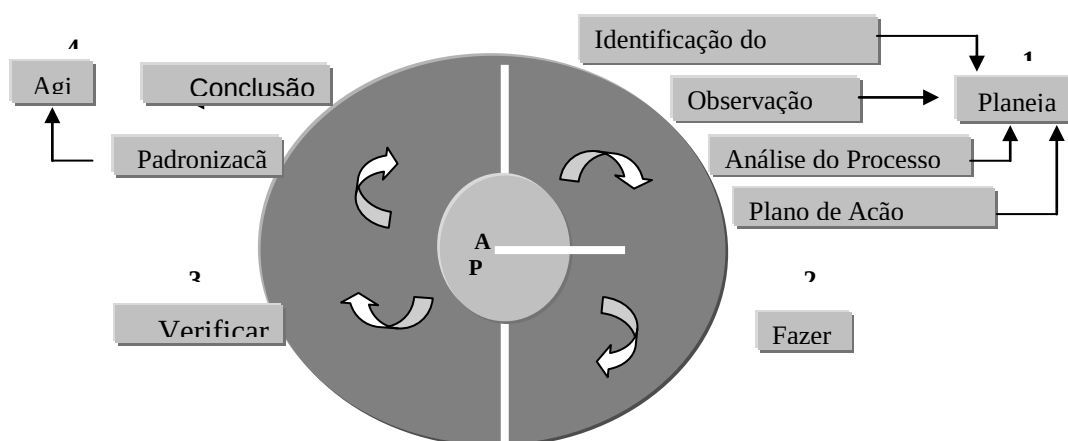


**Figura 1 - Ciclo do PDCA como metodologia para melhoria
Fonte: DEMING (1990)**

Será apresentado a seguir o gerenciamento do Processo, Gerência de Requisitos de Software da Soltin, utilizando o Ciclo de Deming ou PDCA.

Processo: Gerência de Requisitos do Projeto de Implantação e Certificação no MR-MPS nível G.

Fase 1 do Ciclo PDCA: PLAN (Planejar)

Definição das Metas: Consiste em definir as metas a serem alcançadas.

Objetivo / Tarefa: Gerenciar Requisitos de *Software*

Prazo: 12 meses

Definição de Métodos: Consiste em criar métodos que possibilitem atingir as metas estipuladas.

- 1 - Comunicação contínua com os fornecedores de requisitos;
- 2 - Obtenção de entendimento dos requisitos;
- 3 - Definir critérios para a aceitação dos requisitos;
- 4 - Comprometimento com os requisitos;
- 5 - Matriz de rastreabilidade bidirecional entre os requisitos;
- 6 - Verificar inconsistências entre os requisitos;
- 7 - Gerenciar mudanças nos requisitos.

Fase 2 do Ciclo PDCA: DO (Fazer)

Educar e Treinar: Consiste em educar a equipe para a importância dos métodos a serem seguidos e capacitar a equipe nos métodos utilizados.

- Conscientizar a equipe, por meio de reuniões, para a importância do cumprimento rigoroso dos métodos estipulados;
- Capacitar a equipe para atender os métodos propostos, por meio de cursos como, Mapeamento de Processos, ISO 9000, MPS.BR, etc.

Execute o Trabalho: Consiste em criar meios para verificar se os métodos estão sendo colocados em prática.

- Através do preenchimento de documentos (*templates*: Ata de Reunião, Carta Proposta, Documento de Especificação de Requisitos, Plano de Aprovação, Documento para Mapeamento do Processo de Gerência de Requisitos, Documento de Consistência dos Requisitos, Plano de Ação e etc.) será verificado o cumprimento dos métodos propostos.

Fase 3 do Ciclo PDCA: CHECK (Checar/Monitorar)

Verificar os efeitos do Trabalho Executado: Consiste em confirmar a efetividade da ação, ou seja, checar se as metas estão sendo cumpridas.

- Revisões periódicas;
- Reuniões;
- Preenchimento da Documentação.

Fase 4 do Ciclo PDCA: ACT (Agir)

Atue no Processo em Função dos Resultados: Agir, o que está bom, pode ser melhorado e o que é negativo é recomendado encontrar formas para melhorar.

A Soltin, já vinha praticando muitos dos resultados esperados dos dois processos exigidos no nível G do MPS.BR (Gerência de Projetos e Gerência de Requisitos).

Hoje, a Soltin atende a todos os resultados esperados, (no que se refere a documentação), dos processos exigidos para certificação MPS.BR. O que a empresa possuía em termos de artefatos, foi melhorado e atendem a necessidade requerida. Este fato, está melhor explicado no capítulo 3.4, que diz respeito a

aderência dos artefatos da Soltin com os resultados esperados do MPS.BR nível G.

Para atender a esta fase do ciclo PDCA faz-se necessário obter aperfeiçoamento dos artefatos da seguinte forma:

- Aderindo conhecimentos da ISO 9001:2000, no que se refere a elaboração de documentos;
- Aderindo conhecimentos sobre Engenharia de *Software* direcionada para Processos, CMMI e MPS.BR;

4. Conclusões e Perspectivas Futuras

O mercado de *software* onde as MPE's estão inseridas é extremamente concorrido, essa realidade é bem refletida nas MPE's do pólo Amazonsoft. Neste trabalho foi apresentada uma abordagem para que empresas de pequeno porte possam ter maturidade em seus processos de desenvolvimento de *software* e conseqüentemente qualidade em seus produtos.

A implantação de processo de *software* é mais que uma tendência nas empresas de desenvolvimento de *software* do pólo Amazonsoft, trata-se de um requisito indispensável para enfrentar o mercado competitivo e atender exigências dos clientes. No entanto esta não é uma tarefa simples.

Como exemplo de melhoria em processo de *software*, foi apresentado o esforço da micro empresa Soltin, incubada no CIDE, sede do Amazonsoft, com o intuito de implementar o nível G do Modelo MPS.BR. O esforço em querer obter melhorias em processos de desenvolvimento de *software*, no caso da Soltin, foi recompensado com aceitação de trabalhos submetidos em eventos reconhecidos a nível nacional, como por exemplo: PBQP-SW, EQPS e SBQS. A qualidade do *software* será bem sucedida se devidamente submetida à padrões de qualidade estipulados por Normas, Modelos, Guias, todos referenciando a melhoria do processo de *software* e conseqüentemente o produto que é o próprio *software*.

Durante a pesquisa, foi observada a grande dificuldade que as MPE's de *software* têm em obter melhoria nos seus processos. São vários os fatores que contribuem para tal, como foi citado o principal deles é o pequeno poder aquisitivo de empresas desta natureza e também a falta de maturidade, dentre outros.

Foi observado também que, praticamente todo e qualquer conflito, que diz respeito a processo de *software*, apresentado dentro das MPE's são por falta de gerenciamento. Para tentar solucionar este problema, foi usado como alicerce para a melhoria do processo de *software* o modelo MPS.BR nível G que atende as necessidades aqui apresentadas.

O estudo aqui exposto, reflete a realidade de muitas empresas de pequeno porte. É notório observar que, assim como a empresa Soltin com esforço e dedicação está conquistando resultados, outras empresas desta natureza podem também adquirir resultados satisfatórios.

Para trabalhos futuros, pretende-se disponibilizar esta pesquisa, com o intuito de dar continuidade na progressão aos níveis de maturidade do Modelo MPS.BR.

A cada dia, espera-se por parte das MPE's do Pólo Amazonsof a utilização de um Modelo de Qualidade de *Software*, pensando nisto, torna-se real a possibilidade de obter aperfeiçoamento para que uma MPE's venha a se tornar uma empresa Implementadora ou Avaliadora MPS.BR. Assim como existe consultor ISO, o mercado de consultoria MPS.BR esta se expandindo gradativamente.

Com a disseminação do modelo MPS.BR nas MPE's e Academia, acredita-se que a qualidade dos processos de desenvolvimento de *software* será maior e conseqüentemente o produto gerado alcançará maiores índices de confiabilidade.

O motivo pelo qual o processo de desenvolvimento de *software* é exaustivamente enfatizado é que a qualidade do produto esta fortemente ligada à qualidade do processo.

Referências

[Guia Geral MPS.BR, 2006] Associação para Promoção da Excelência do Software Brasileiro –SOFTEX. *MPS.BR - Melhoria de Processo de Software Brasileiro - Guia Geral*. 2 ed. 2006.

[CMMI, 2002] Software Engineering Institute – SEI. *CMMI - Capability Maturity Model Integration*. Disponível em: <<http://www.sei.cmu.edu/cmmi>>, Acessado em 01 set. 2006.

[ROCHA, 2001] Rocha, Ana Regina Cavalcante da. Maldonado, José Carlos. Weber, Kival Chaves. *Qualidade de Software Teoria e Prática*. São Paulo: Prentice Hall, 2001.

[NBR ISO/IEC 12207, 1998] *Tecnologia de informação: processos de ciclo de vida de software*. 1 ed. Rio de Janeiro: ABNT, 1998.

[ISO/IEC 15504, 2003] INTERNATIONAL STANDARD ORGANIZATION. ISO/IEC TR 15504: Information Technology - software process assessment. Disponível em: <<http://www.sqi.cit.gu.edu.au/spice/>>, Acessado em 15 set. 2006.

[PMBOK, 2004] Project Management Institute (PMI). *PMBOK - Project Management Body of Knowledge*. 3 ed. 2004. Disponível em: <<http://www.pmimg.org.br>>, Acessado em: 11 out. 2006.

[PRESSMAN, 2006] Pressman, Roger S. *Engenharia de Software*. 6 ed. São Paulo: McGraw-Hill, 2006.

[PÁDUA, 2003] Filho, Wilson de Pádua Paula. *Engenharia de Software Fundamentos Métodos e Padrões*. 2 ed. LTC 2003.

[NBR ISO 9000: 2000] NBR ISO 9000 – Sistemas de Gestão da Qualidade – Fundamentos e Vocabulário. Rio de Janeiro: ABNT – Associação Brasileira de Normas Técnicas, 2000.

[DEMING, 1990], William Edwards. *Qualidade: A Revolução da Administração*. Rio de

Janeiro: Marques Saraiva, 1990.

[2.38] O Programa de Medição e Análise Através dos Níveis do CMMI

Entidade: Instituto Atlântico, Rua Chico Lemos, 946 - Caixa Postal 60822-780, Fortaleza - CE - Brasil

Autores: Solange A. Alcântara, Fabiana G. Marinho, Tatiana C. Monteiro, Márcia G. Sampaio, Carlo Giovano S. Pires - {fabiana, tatiana, solange, marcia, cgiovano}@atlantico.com.br

1. Introdução

Um dos objetivos estratégicos do Instituto Atlântico é a melhoria contínua dos seus processos. A organização já conseguiu chegar no nível de maturidade três do modelo CMMI (CMMI, 2006), agora tem como meta atingir o nível cinco. Neste nível, a melhoria contínua é baseada no entendimento de causas comuns de variação do processo. Neste contexto este trabalho visa apresentar soluções para institucionalização de um Programa de Medição e Análise atendendo cada nível de maturidade do CMMI.

2. Objetivos e Justificativa

Um dos objetivos estratégicos do Instituto Atlântico é a melhoria contínua dos seus processos. A organização já conseguiu chegar no nível de maturidade 3 do modelo CMMI, agora ela almeja alcançar o nível 5. Neste nível, a melhoria contínua é baseada no entendimento de causas comuns de variação do processo. Neste contexto este trabalho visa apresentar soluções para institucionalização de um Programa de Medição e Análise aderente aos níveis 2 ao 5 do modelo CMMI.

O processo de medição e análise ganhou maior destaque no CMMI (Capability Maturity Model Integration) do que no SW-CMM (Capability Maturity Model for Software) que tratava a medição e análise como uma característica comum (comum feature) (Goethert, 2004). A incorporação da medição e análise como uma área de processo distinta no CMMI fornece a gerencia a visibilidade e o foco que as organizações têm precisado para guiar o uso da medição nos esforços de melhoria. Ainda assim, a medição não é muito integrada nas práticas de engenharia de software e engenharia de sistemas, permanecendo a medição um desafio para muitas organizações (Brown, 2004).

Neste contexto, este projeto gerou uma Dissertação de Mestrado que defini um modelo para estruturação do Repositório Organizacional de Medições aderente a cada nível de maturidade do CMMI. O repositório de medições é o principal artefato de um programa de medição e análise e tem a função de armazenar as medidas coletadas e analisadas em uma organização (Araújo, 2006).

3. Metodologia de Execução



Através de análises no modelo CMMI, foram identificados os requisitos para implementação do modelo.

4. Resultados Obtidos

Dissertação de mestrado (Em fase de defesa)

- Um modelo de Repositório Organizacional de Medições para Projetos de Software

Artigo Publicado

- Implementação de um Repositório de Métricas para os Níveis 2 e 3 do CMMI, XXXII Conferencia Latinoamericana de Informática CLEI 2006.

Apresentação Realizada

- Implementação de um Repositório de Medições para os Níveis 2 e 3 do CMMI, Terceira Conferência Anual SEPG Latino Americano, 2006.

5. Aplicabilidade dos Resultados

Um modelo proposto pode ser utilizado por equipes envolvidas na institucionalização da área de processo de medição e análise.

6. Características Inovadoras

Vários trabalhos com temas semelhantes foram analisados, mas em todos eles o foco era definir um repositório para armazenar as métricas. No modelo proposto o objetivo principal é definir como as medidas devem ser armazenadas e organizadas em cada nível de maturidade do CMMI, sendo também definidas as métricas necessárias.

7. Conclusão

O modelo foi aplicado em organizações com nível de maturidade inferior ao nível 4. É importante a validação desse modelo em outras organizações com níveis de maturidade superiores. Outra aplicação interessante seria a utilização deste modelo em pequenas e médias empresas.

Referências bibliográficas

- (ARAÚJO, 2006a) Araújo, Solange A.; Sampaio, Márcia G.; Pires, Carlo Giovano S.; Belchior, Arnaldo D., *Implementação de um Repositório de Métricas para os Níveis 2 e 3 do CMMI*, XXXII Conferencia Latino Americana de Informática **CLEI** 2006.
- (BROWN, 2004) Brown, Maureen; Goldenson, Dennis, *Measurement And Analysis: What Can and Does Go Wrong?* Journal Title: IEEE International Software Metrics Symposium, 2004
- (CMMI, 2006) Capability Maturity Model Integration, *CMMI for Software Engineering, Version 1.1-SW, Staged Representation* CMU/SEI-2002-TR-029.ESC-TR-2002-029, Pittsburgh, Software Engineering Institute - Carnegie Mellon University, 2002.

- (ARAÚJO, 2006a) Araújo, Solange A.; Sampaio, Márcia G.; Pires, Carlo Giovano S.; Belchior, Arnaldo D., *Implementação de um Repositório de Métricas para os Níveis 2 e 3 do CMMI*, XXXII Conferencia Latino Americana de Informática **CLEI** 2006.
- (BROWN, 2004) Brown, Maureen; Goldenson. Dennis, *Measurement And Analysis: What Can and Does Go Wrong?* Journal Title: IEEE International Software Metrics Symposium, 2004
- (CMMI, 2006) Capability Maturity Model Integration, *CMMI for Software Engineering, Version 1.1-SW, Staged Representation* CMU/SEI-2002-TR-029.ESC-TR-2002-029, Pittsburgh, Software Engineering Institute - Carnegie Mellon University, 2002.

[2.40] - Processo de Desenvolvimento de Software em Conformidade com o Nível G do Modelo de Referência MPS.BR

Entidade: Instituto Centro-Oeste de Desenvolvimento de Software (ICODES) - Rua Herculano Lobo, 206, Sala 01 – 73801-260 – Formosa – GO – Brasil

Autor: Cristiano Lehrer - cristiano@ICODES.org.br

Resumo. Este artigo apresenta o estudo de caso da implementação dos processos Gerência do Projeto (GPR) e Gerência de Requisitos (GRE) do modelo de referência da Melhoria de Processo do Software Brasileiro (MPS.BR), no Instituto Centro-Oeste de Desenvolvimento de Software (ICODES), para o cumprimento dos requisitos necessários para a obtenção do nível de maturidade Parcialmente Gerenciado (G) do referido modelo de referência.

1. Introdução

A indústria de software nacional se encontra num estágio prematuro em relação à adoção de modelos de gestão e desenvolvimento de software, comparado com países como Estados Unidos, Japão, Índia e outros países considerados grandes produtores de software. Mas o panorama vem se alterando rapidamente, principalmente em virtude do aumento da competição internacional nesse segmento, que movimenta bilhões de dólares anuais [Fernandes, 2004].

Com o intuito de atuar nesse panorama mundial, o Instituto Centro-Oeste de Desenvolvimento de Software está estruturando suas atividades de desenvolvimento de software de acordo com uma Fábrica de Software, buscando aumentar a produtividade e a qualidade dos serviços prestados, através da adoção de processos de desenvolvimento em conformidade com o Modelo de Referência da Melhoria de Processo do Software Brasileiro (MPS.BR).

O presente artigo apresenta o estudo de caso da implementação, no Instituto Centro-Oeste de Desenvolvimento de Software, dos processos de Gerência do Projeto (GPR) e Gerência de Requisitos (GRE) definidos no Modelo de Referência MPS.BR, para o cumprimento dos requisitos necessários para a obtenção do nível de maturidade Parcialmente Gerenciado (G) do referido modelo de referência.

2. Objetivos e Justificativa



Em tempos de competitividade acirrada entre organizações desenvolvedoras de software, a consciência da necessidade de processos de produção mais eficientes, que garantam o equilíbrio perfeito entre qualidade e produtividade, vem crescendo substancialmente, de modo que o fator qualidade tem sido considerado fundamental para o sucesso de qualquer empresa [Vasconcelos, 2003].

Com o intuito de agilizar e melhorar a produção de software, as organizações que provêm serviços de desenvolvimento de sistemas começaram a utilizar processos de desenvolvimento de software bem definidos e tecnologia de ponta, com o intuito de produzirem software de alta qualidade, a baixo custo e de forma rápida [Herbsleb, 1999].

Para comprovar a eficiência e a eficácia do processo de desenvolvimento de software, as organizações buscam implementar certificações que demonstrem a qualidade dos produtos desenvolvidos por elas, dentre as quais a ISO (*International Organization for Standardization*) e o CMMI (*Capability Maturity Model Integration*) [Fernandes, 2004].

Os custos despendidos para implantação e posterior avaliação das principais certificações disponíveis no mercado são proibitivos para micro e pequenas empresas, pois exigem um esforço de vários anos de seus colaboradores, consultores e equipes de avaliação. Buscando conduzir essa grande massa de micro e pequenas empresas de software a alcançarem melhorias significativas nos seus processos de software, a Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) inicia em dezembro de 2003 o programa de Melhoria de Processo do Software Brasileiro (MPS.BR) [SOFTEX, 2006].

O MPS.BR é realizado em parceria com diversas instituições de ensino, pesquisa, centros tecnológicos e sociedades de economia mista, visando a melhoria de processo do software Brasileiro, através do desenvolvimento e aprimoramento de um Modelo de Referência, compatível com o CMMI e em conformidade com as normas ISO/IEC 12207 e ISO/IEC 15504 [Weber, 2005].

3. Metodologia de Execução

Um processo pode ser definido de várias maneiras, dentre as quais pode-se considerar, que um processo é uma seqüência de passos realizados para um determinado propósito, ou que um processo é um conjunto de atividades intra-relacionadas, que transformam entradas em saídas [Machado, 2003].

3.1. Gerência do Projeto – GPR

A área de gerência de projetos vem recebendo atenção crescente por parte das organizações [Prikladnicki, 2004], principalmente devido ao crescente número de organizações que aderem à gestão orientada a projetos [Demarco, 2003].

Tipicamente, um processo de gerência de projetos envolve três atividades principais [Falbo, 2005]:

- planejamento: no início do projeto, um plano organizado de como o projeto será conduzido deve ser elaborado, contendo a definição do escopo do projeto, do processo de software do projeto, da realização de estimativas,

da elaboração de um cronograma e da identificação e tratamento dos riscos associados ao projeto;

- acompanhamento: por estarem sujeitos a mudanças, é fundamental realizar o acompanhamento do progresso do projeto, refinando escopo e estimativas, alterando o processo do projeto e o cronograma, além de monitorar riscos e tomar ações corretivas;
- encerramento: realizar uma análise crítica do que deu certo e o que não funcionou na execução do projeto, procurando registrar as lições aprendidas de sucesso e oportunidades de melhoria.

O objetivo primordial da atividade de planejamento é a elaboração do plano de como o projeto será conduzido durante a sua execução, o qual servirá de referência para as demais atividades do processo de Gerência do Projeto. O resultado da execução dessa atividade é a elaboração do artefato Plano de Projeto.

O plano de projeto é um documento aprovado formalmente, usado para gerenciar e controlar a execução do projeto [PMBOK, 2004]. A figura 01 apresenta o índice do documento de Plano de Projeto utilizado pelo Instituto Centro-Oeste de Desenvolvimento de Software para registrar o planejamento do projeto, o qual é uma adaptação do modelo de Plano de Projeto apresentado por Rouiller [2003].

1. INTRODUÇÃO

1.1. Visão geral deste documento

1.2. Convenções, termos e abreviações

1.2.1. Identificação dos riscos

1.2.2. Probabilidade e impacto dos riscos

2. VISÃO GERAL

3. PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

3.1. O processo de software do ICODES

3.2. Artefatos gerados

3.3. Padrões adotados

3.4. Ferramentas utilizadas

3.5. Revisões, verificações e validações

4. ENTRADAS E SAÍDAS DO PROJETO

5. ORGANIZAÇÃO DO PROJETO

5.1. Organograma

5.1.1. Capacidades necessárias para o projeto

5.1.2. Recursos Humanos

5.2. Infra-estrutura

- 5.2.1.Ferramentas
- 5.2.2.Equipamentos
- 5.2.3.Outros itens relevantes
- 5.3. Interfaces técnicas e organizacionais
 - 5.3.1.Reuniões da equipe técnica
 - 5.3.2.Reuniões de garantia da qualidade
 - 5.3.3.Interface entre a equipe técnica e os usuários (clientes)
- 5.4. Controle de documentos e dados
- 5.5. Treinamento
- 6. ANÁLISE DE RISCOS
- 7. ARMAZENAMENTO, CÓPIA, RECUPERAÇÃO E PRESERVAÇÃO
- 8. CRONOGRAMA
- 9. REFERÊNCIAS

Figura 01 - Índice do artefato Plano de Projeto

Com o propósito de coletar e analisar as informações relativas aos produtos desenvolvidos e aos processos implementados na organização em seus projetos, de forma a permitir um planejamento mais objetivo através de estimativas confiáveis e reais, o Instituto Centro-Oeste de Desenvolvimento de Software implementou o *Personal Software Process* (PSP) entre seus colaboradores [Albuquerque, 2004].

O plano de projeto identifica e registra a periodicidade no qual o projeto será acompanhado, indicando a periodicidade das reuniões da equipe técnica e da equipe de garantia da qualidade. As reuniões são documentadas através de uma ata de reunião, no qual são registrados o desempenho do projeto durante o seu ciclo de vida.

Além das reuniões da equipe técnica e da garantia da qualidade, o ciclo de vida utilizado para o desenvolvimento do projeto também possui marcos estabelecidos no qual serão realizadas revisões no planejamento do projeto. Normalmente, os marcos de revisão são estabelecidos nas entregas dos módulos do produto ou serviço solicitado ao cliente.

O objetivo desse acompanhamento é monitorar todos os aspectos do planejamento do projeto – cronograma, custos, recursos, riscos, envolvimento dos interessados e dados – de forma a garantir a efetiva conclusão do projeto dentro dos atributos especificados inicialmente.

Caso seja identificado algum problema nas monitorações, o mesmo é registrado e analisado, e ações corretivas são estabelecidas quando necessário e gerenciadas até a sua conclusão. Esse processo é realizado utilizando o modelo IDEAL (*Initiating, Diagnosing, Establishing, Acting and Leveraging*) [McFeekey, 1996].

No encerramento do projeto é identificado se o projeto foi bem-sucedido ou não, pois a mera entrega do produto ou serviço almejado para o cliente não significa que ele foi finalizado de forma satisfatória.

3.2. Gerência de Requisitos - GRE

Os requisitos consistem em uma série de sentenças que descrevem de maneira clara, concisa, consistente e não-ambígua todos os aspectos significativos do sistema a ser desenvolvido, contendo os requisitos funcionais e não-funcionais do sistema, além das restrições funcionais e de projeto [Rocha, 2001].

Os requisitos do projeto são levantados através de entrevistas com os clientes e/ou principais usuários do sistema. Antes da realização da entrevista, o entrevistador deve elaborar um plano para a entrevista, contendo os principais questionamentos que devem ser abordados, visando tornar a entrevista mais produtiva, para não exigir do usuário um longo período de tempo.

Após a realização da entrevista com o cliente, os colaboradores preparam um relatório, contendo os principais pontos discutidos na entrevista, a ser entregue ao cliente para conferência das informações prestadas por ele na ocasião. Caso ainda haja pontos dúbios, os mesmos são registrados no relatório, para servirem de base para as próximas entrevistas.

O processo de identificação dos requisitos é realizado até o momento que a equipe de desenvolvedores tenha adquirido conhecimento suficiente sobre o projeto a ser desenvolvido, quando os requisitos são validados pela equipe técnica e de usuários, formalizando o comprometimento entre as partes.

Os requisitos são registrados no documento de requisitos, onde são registrados todos os requisitos funcionais e não-funcionais do produto ou serviço solicitado pelo cliente. A figura 02 apresenta o índice do documento de requisitos utilizado pelo Instituto Centro-Oeste de Desenvolvimento de Software para registrar os requisitos do projeto, o qual é uma adaptação do modelo de documento de requisitos apresentado por Rouiller [2003].

1. INTRODUÇÃO

1.1. Convenções, termos e abreviações

1.1.1. Identificação dos requisitos

1.1.2. Prioridades dos requisitos

2. VISÃO GERAL DO PRODUTO/SERVIÇO

2.1. Abrangência e sistemas relacionados

2.2. Descrição do cliente

2.3. Descrição dos usuários

2.3.1. <Nome de um tipo específico de usuário>

2.3.2. <Nome de um tipo específico de usuário>

3. REQUISITOS FUNCIONAIS

3.1. <Nome de subseção para agrupar requisitos funcionais correlacionados>

- 3.1.1. RF001 - <Nome do requisito funcional>
- 3.1.2. RFXXX - <Nome do requisito funcional>
- 3.2. <Nome de subseção para agrupar requisitos funcionais correlacionados>
- 4. REQUISITOS NÃO FUNCIONAIS
 - 4.1. Usabilidade
 - 4.1.1. RNF001 - <Nome do requisito>
 - 4.1.2. RNFXXX - <Nome do requisito>
 - 4.2. Confiabilidade
 - 4.2.1. RNFXXX - <Nome do requisito>
 - 4.3. Desempenho
 - 4.3.1. RNFXXX - <Nome do requisito>
 - 4.4. Segurança
 - 4.4.1. RNFXXX - <Nome do requisito>
 - 4.5. Distribuição
 - 4.5.1. RNFXXX - <Nome do requisito>
 - 4.6. Padrões
 - 4.6.1. RNFXXX - <Nome do requisito>
 - 4.7. Hardware e Software
 - 4.7.1. RNFXXX - <Nome do requisito>
- 5. RASTREABILIDADE
 - 5.1. Entre requisitos funcionais
 - 5.2. Entre requisitos funcionais e não funcionais
- 6. REFERÊNCIAS

Figura 02 – Índice do artefato Documento de Requisitos

4. Resultados Obtidos

Durante o desenvolvimento e implantação do processo, várias dificuldades foram observadas, principalmente em relação a mudança de comportamento por parte dos colaboradores envolvidos no processo.

Em relação aos gerentes de projeto, a grande dificuldade observada para a implantação do processo proposto foi conscientizar os mesmos da necessidade de realizarem um planejamento rigoroso e minucioso do projeto logo no seu início.

Anteriormente, os planos de projeto produzidos eram superficiais em vários pontos, principalmente na definição do escopo do projeto e nas atividades a serem executadas para a sua conclusão, ocasionando aumentos expressivos nos custos e no cronograma dos projetos, em virtude da necessidade de se realizar grandes alterações no plano de projeto.

Outra dificuldade observada foi institucionalizar o registro das reuniões realizadas com a equipe técnica e com os clientes, tanto para o acompanhamento do projeto como para o levantamento dos requisitos, e principalmente, a monitoração e o

gerenciamento dos problemas identificados, que anteriormente eram feitos de maneira informal.

Em relação aos demais colaboradores da organização, a grande dificuldade observada foi a implantação do *Personal Software Process*, para a coleta e análise das informações relativas aos produtos desenvolvidos e aos processos implementados na organização em seus projetos, de forma a permitir um planejamento mais objetivo através de estimativas confiáveis e reais.

Inicialmente, os colaboradores relutaram na adoção dessa metodologia, pela dificuldade de alterarem seus hábitos de desenvolvimento e por questionarem o real benefício que essa metodologia poderia agregar ao trabalho desenvolvido pelos mesmos.

Essas dificuldades foram sendo solucionadas com o tempo, com treinamentos e conversas informais, através da conscientização dos colaboradores dos benefícios que a metodologia trazia para eles e para a organização.

Além do emprego do *Personal Software Process*, a utilização de uma arquitetura de desenvolvimento padrão para componentes de software fornece um apoio extra nas estimativas necessárias para o planejamento do projeto, e principalmente, no acompanhamento do plano traçado para a execução do projeto.

A execução do presente projeto também resultou na monografia Implementação do Processo Gerência do Projeto do Modelo de Referência MPS.BR no Instituto Centro-Oeste de Desenvolvimento de Software, apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação Lato Sensu Modelo de Maturidade e Capacidade do Processo com CMMI® e MPS.BR®, defendida em novembro de 2006.

5. Aplicabilidade dos Resultados

O processo de desenvolvimento de software em conformidade com o nível G – Parcialmente Gerenciado, do modelo de referência MPS.BR, é estratégico para que o Instituto Centro-Oeste de Desenvolvimento de Software e as empresas associadas a ele alcancem níveis internacionais na produção de software, tanto na produtividade como na qualidade dos serviços prestados aos clientes.

As organizações com processos de desenvolvimento bem definidos possuem um grande diferencial competitivo sobre as demais organizações que ainda não adequaram seus processos de desenvolvimento aos novos modelos e padrões exigidos pelo mercado consumidor.

A implementação e implantação do modelo de referência MPS.BR no Instituto Centro-Oeste de Desenvolvimento de Software está abrindo novas perspectivas para o processo de melhoria de software empregado na instituição, objetivando aumentar a qualidade e a produtividade dos serviços prestados para a comunidade através da definição de um processo de desenvolvimento reconhecido por organizações independentes.

O processo de desenvolvimento proposto está sendo implantado inicialmente no Instituto Centro-Oeste de Desenvolvimento de Software, e após a homologação do

mesmo pela Associação para Promoção da Excelência do Software Brasileiro, o mesmo será disponibilizado para as demais organizações associadas ao Instituto e para o público em geral, que poderão utiliza-lo como guia para adequarem seu próprios processos ao modelo de referência MPS.BR.

6. Características Inovadoras

O Instituto Centro-Oeste de Desenvolvimento de Software é um instituto de pesquisa e desenvolvimento de software que iniciou as suas atividades no primeiro trimestre de 2006, com uma equipe técnica composta por cinco colaboradores, que atualmente é composta com nove colaboradores. A implementação e implantação de um processo de desenvolvimento de software em conformidade com o modelo de referência MPS.BR servirá como estudo de caso para que empresas imaturas e pequenas consigam adequar seus processos conforme um modelo de referência.

7. Conclusão e Perspectivas Futuras

Para desenvolver e implantar os processos de Gerência do Projeto e Gerência de Requisitos no Instituto Centro-Oeste de Desenvolvimento de Software, foram utilizados aproximadamente quinhentas horas da equipe de colaboradores, incluindo as horas despendidas na implantação do *Personal Software Process* entre os colaboradores da organização.

O maior custo no desenvolvimento e implantação do processo foram com o treinamento dos colaboradores, tanto no próprio processo como no *Personal Software Process*. Neste contexto, também está sendo considerado como treinamento o tempo despendido na avaliação do processo proposto, quando a equipe se reunia para avaliação do próprio processo, sugerindo melhorias no processo, e principalmente, nos artefatos a serem utilizados.

Conforme avaliações preliminares, a execução do processo proposto pelo Instituto Centro-Oeste de Desenvolvimento de Software se encontra em conformidade com os requisitos do modelo de referência MPS.BR para o nível de maturidade Parcialmente Gerenciado (G).

O Instituto Centro-Oeste de Desenvolvimento de Software almeja alcançar o nível de maturidade G – Parcialmente Gerenciado, do modelo de referência MPS.BR ainda no ano de 2007, trabalhando para desenvolver, e principalmente, refinar seus processos em conformidade com o referido nível de maturidade.

A expectativa é que ainda sejam necessários alocar cerca de trezentas e cinquenta horas da equipe de colaboradores para se alcançar o nível de maturidade desejado, não incluindo nesse cálculo a quantidade de horas a serem despendidas com a validação dos processos por um consultor externo e nem os custos com a avaliação oficial.

8. Referências Bibliográficas

Albuquerque, Jones de Oliveira. (2004). "Gerencia de Processo de Software em Pequena Escala: ênfase em PSP, TSP e P-CMM". Lavras: UFLA/FAEPE. 113p.

Demarco, T. (2003). "Waltzing with bears: managing risk on software projects." New York: Dorset House.

Falbo, Ricardo de Almeida. (2005). "Engenharia de software: notas de aula". Disponível em: <www.inf.ufes.br/~falbo/download/aulas/es-g/2006-2/NotasDeAula.pdf>. Acesso em: set. 2006.

Fernandes, Aguinaldo Aragon; Teixeira, Descartes de Souza. (2004). "Fábrica de software: implantação e gestão de operações". São Paulo: Atlas. 304p.

Herbsleb, J. D.; Grinter, R. E. (1999). Splitting the organization and integrating the code: Conway's law revisited. In: "Proceedings of ICSE". Los Angeles: IEEECS. Pág. 85-95.

Machado, Cristina Ângela Filipak. (2003). "Definindo processos do ciclo de vida de software usando a norma NBR ISO/IEC 12207". Lavras: UFLA/FAEPE. 101p.

Mcfeeley, Bob. (1996). "IDEAL: a user's guide for software process improvement." Pittsburgh: Software Engineering Institute. 222p.

PMBOK. (2000). "A guide to the project management body of knowledge". Pennsylvania USA: Project Management Institute. Prikladnicki, R. (2004). Risk management in distributed software development: a process integration proposal. In: "Proceedings of 5th IFIP World Computer Congress." Toulouse, France.

Rocha, Ana Regina Cavalcanti da. (2001). "Qualidade de Software". São Paulo: Prentice Hall. 303p.

Rouiller, Ana Cristina. (2003). "Gerência de Projetos de Software". Lavras: UFLA/FAEPE.

SOFTEX – Associação para Promoção da Excelência do Software Brasileiro. (2006). "Guia geral". Versão 1.1.

Vasconcelos, Alexandre Marcos Lins de; Maciel, Teresa Maria de Medeiros. (2003). "Introdução à engenharia de software e aos princípios de qualidade". Lavras: UFLA/FAEPE. 104p.

Weber, Kival Chaves; Araújo, Eratóstenes; Machado, Cristina Ângela Filipak; Scalet, Danilo; Salviano, Clênio Figueiredo; Rocha, Ana Regina Cavalcanti da. (2005).

Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software – versão 1.0 (MR-MPS e MA-MPS). In: "IV Simpósio Brasileiro de Qualidade de Software". Porto Alegre: SBQS.

[2.41] Um Processo para Controle e Garantia de Qualidade no Desenvolvimento de Software

Entidade: Centro de Informática (CENIN) da Câmara dos Deputados Praça dos Três Poderes, Palácio do Congresso Nacional, Anexo I, 24º andar, Sala 2402
70160-900 - Brasília -DF - Brasil

Autor: Carlos Constantino Moreira Nassur - carlos.nassur@camara.gov.br

Abstract. *This paper describes the development report of project “2.41 – Um processo para controle e garantia de qualidade no desenvolvimento de software”, submitted by Informatics Center of Brazilian House of Representatives in 2006’ Cycle of Brazilian Program for Software Quality and Productivity, which is promoted by Secretary of Politics for Informatics and Technology from Ministry of Science and Technology.*

Resumo. *Este artigo apresenta o relato do desenvolvimento do projeto “2.41 – Um processo para controle e garantia de qualidade no desenvolvimento de software”, submetido pelo Centro de Informática da Câmara dos Deputados ao Ciclo 2006 do Programa Brasileiro da Qualidade e Produtividade em Software, promovido pela Secretaria de Política de Informática e Tecnologia do Ministério da Ciência e Tecnologia.*

1 Introdução

O Centro de Informática da Câmara dos Deputados vem desenvolvendo diversas iniciativas para aumentar a qualidade geral dos produtos e serviços que oferece à Casa e à sociedade.

Entre tais iniciativas encontram-se (I) a de se estabelecer os processos adequados ao desenvolvimento e manutenção de sistemas de informação; (II) a aquisição de ferramentas de *software* com vistas no incremento de produtividade das equipes; e (III) o permanente treinamento e aperfeiçoamento de seu corpo técnico.

Neste cenário, o CENIN elaborou e submeteu ao Ciclo 2006 do Programa Brasileiro da Qualidade e Produtividade em *Software* (PBQP *Software*) uma dessas iniciativas, que foi a de desenvolver um processo para controle e garantia de qualidade dos projetos de desenvolvimento e manutenção de *software*. Este artigo apresenta o relato deste empreendimento.

2 Objetivos e Justificativa

A qualidade de um produto ou serviço pode ser aferida, ou obtida, pelo processo que é utilizado para a sua elaboração. O processamento, ou o processo, é um dos principais condicionantes da qualidade que um produto, seja este material (um objeto) ou conceitual (uma doutrina, uma teoria etc), possa eventualmente apresentar.

Nessa linha, o modelo *CMMI*, do *Software Engineering Institute*, uma referência internacional para projetos de melhoria de qualidade em organizações intensivas em *software*, adota como premissa básica a de que a qualidade de um produto de *software* depende fundamentalmente dos atributos do processo mediante o qual é desenvolvido.

Assim, considerando-se verdadeiras tais proposições, segue-se então que o estudo e a constante melhoria dos processos executados pelas organizações, para a criação de produtos ou para a prestação de serviços, é fundamental para sua própria sobrevivência.

Não há, porém, em razão do envolvimento humano, garantia de que uma organização que possua um portfólio perfeito de processos seja capaz de produzir

bens ou prestar serviços com índices de qualidade grandemente diferenciados e, com eles, obter satisfação de seus clientes. Assim, a qualidade dos processos é também função da habilidade/qualificação e da atuação dos agentes que por eles respondem. Isto porque é o empenho e o preparo operacional e gerencial dos atores da organização que lhes definem a forma final.

A melhoria da qualidade de produtos de *software* tem sido uma das preocupações do CENIN. Por isto, o Centro de Informática da Casa patrocinou diversas iniciativas para que fossem definidos os processos necessários, especialmente aqueles concebidos com base nas normas NBR ISO/IEC 12207, ISO/IEC 15504-5 e no modelo *CMMI*. Apesar dos esforços desenvolvidos nesta direção, o Centro ainda não possuía um processo de controle e garantia de qualidade efetivo e adequado aos desafios enfrentados no desenvolvimento e manutenção de sistemas, que garantisse tanto a qualidade dos produtos, quanto a dos processos utilizados.

O projeto, assim, visou a contribuir para a solução desse problema, mediante o desenho de um processo de controle e garantia de qualidade com forte embasamento teórico e efetivo na garantia dos produtos elaborados pelo CENIN, mantendo, contudo, o desempenho adequado das equipes responsáveis pelo desenvolvimento e pela manutenção de sistemas.

3 Metodologia de Execução

A execução do projeto deu-se em três etapas distintas: (I) revisão teórica, (II) proposição do processo e (III) execução piloto.

Na primeira etapa, foi efetuada ampla pesquisa e revisão teórica do assunto, com especial atenção para as normas e os modelos de referência. Foram estudadas as normas NBR ISO/IEC 12207, ISO/IEC 15504-5 e o modelo *CMMI*, além da literatura tradicional de Engenharia de *Software*.

Foi o embasamento teórico que gerou, na segunda etapa, a proposição do processo. Nesta, foram elaborados um fluxo e um modelo de trabalho para o processo e uma descrição das atividades e dos padrões dos artefatos para seu suporte; e, por fim, foram treinadas as pessoas responsáveis por sua execução.

O processo foi concebido (I) pela identificação das principais atividades diretamente relacionadas com o controle e a garantia de qualidade listadas na norma NBR ISO/IEC 12207, (II) pela inclusão das metas, objetivos e resultados preconizados pelo modelo *CMMI*, e pela norma ISO/IEC 15504-5. Importa também ressaltar que uma atribuição importante para o processo, também inspirada nesta última norma, foi a preocupação com a melhoria do próprio processo de controle e garantia de qualidade, e do Processo CENIN de Desenvolvimento de Sistemas (PROCEDE) como um todo, a partir das experiências recolhidas na execução das atividades de controle e garantia de qualidade. A Figura 1 apresenta o diagrama de atividades do processo criado.

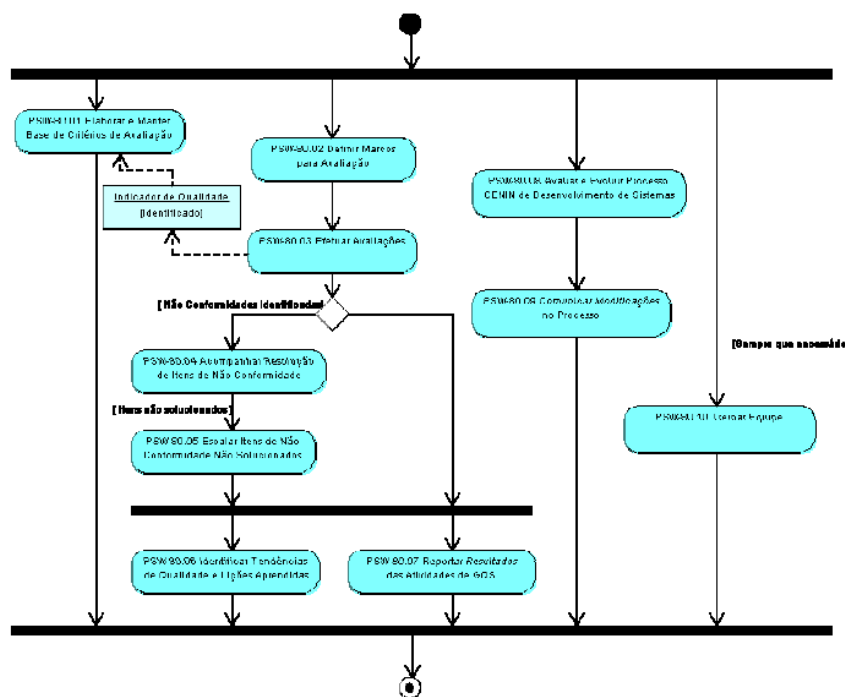


Figura 1: O processo de controle e garantia de qualidade

Finalizado o desenho do processo de controle e garantia de qualidade de *software* e de seus instrumentos de apoio, passou-se à última etapa, a sua execução sobre um projeto-piloto de desenvolvimento, que permitiu a identificação de diversas oportunidades de melhoria do processo, tais como a automação de tarefas, a implementação de alguns artefatos nas ferramentas de apoio ao desenvolvimento de *software* e o ajuste e o sincronismo dos cronogramas de desenvolvimento do projeto e da equipe responsável pela execução do processo de controle e garantia de qualidade.

4 Resultados Relevantes

No segundo semestre de 2006, o processo de controle e garantia de qualidade de *software* elaborado foi implantado no corpo de disciplinas do PROCEDE.

A Coordenação de Engenharia de Sistemas (COENS) do Centro de Informática é o órgão responsável pelo desenvolvimento e pela manutenção dos produtos de *software* de uso corporativo da Casa. Desta forma, é o segmento do CENIN com maior atuação no PROCEDE. Incidentalmente, a implantação do novo processo resultou na sua imediata aplicação em oito dos projetos em execução naquele momento na Coordenação, a saber: “seldor_me_002”, “silegdep_no_001”, “cotapi_no_001”, “orcad_no_001”, “sicod_no_001”, “sisgraf_no_001”, “folhacd_no_001” e “sicat_no_001”. Este conjunto de projetos reúne sistemas de grande importância para a Câmara dos Deputados. Inclui os de desenvolvimento, em sua ampla maioria, e um de manutenção; e abrange sistemas representativos de todas as grandes áreas de atuação do CENIN.

O projeto “seldor_me_002” atua sobre sistemas de apoio às funções de elaboração e fiscalização do orçamento da União; o “silegdep_no_001” tem como objetivo a criação do novo sistema que reunirá e centralizará todas as bases de dados dos Deputados Federais; o “folhacd_no_001” deverá elaborar o novo sistema de folha de pagamentos para o Departamento de Pessoal da Casa; e os restantes atuam em sistemas administrativos tradicionais da Câmara dos Deputados.

A aplicação do processo de controle e garantia de qualidade nos sobreditos projetos foi de grande valia, pois produziu algumas informações inéditas e confirmou suspeitas prévias. A Tabela 1 reúne, por projeto avaliado, as discrepâncias identificadas.

Tabela 1: Principais não conformidades identificadas nos projetos avaliados

Projeto	Não conformidades identificadas
cotapi_no_001	Cronograma, Plano de Gestão de Projeto, Plano de Gestão de Riscos e Plano de Garantia de Qualidade de <i>Software</i> não foram elaborados. Plano de Gestão de Configuração e Ambiente parcialmente criado. O documento não possui todas as informações necessárias à condução do projeto, especialmente quanto ao planejamento das <i>baselines</i> dos artefatos. Sumário Executivo parcialmente criado, mas também não possui todas as informações necessárias à condução do projeto, especialmente quanto à descrição da arquitetura proposta para o sistema.
folhacd_no_001	Plano de Gestão de Projeto, Plano de Gestão de Riscos e Plano de Gestão de Configuração e Ambiente criados; mas não completamente preenchidos. Tabela de Casos de Uso não criada, o que dificulta sobremaneira a associação entre as diversas <i>releases</i> do sistema elaborado pelo projeto com as funcionalidades implementadas.
orcad_no_001	Plano de Gestão de Projeto e Plano de Garantia de Qualidade de <i>Software</i> não foram elaborados para o projeto.

	Plano de Gestão de Configuração e Ambiente e Sumário Executivo não estavam completamente preenchidos Roteiro de Homologação do produto também não contempla os Casos de Uso críticos que deveriam ser executados para a homologação do produto. Sumário Executivo não contempla a descrição da arquitetura proposta para o sistema.
seldor_me_002	As atividades do Grupo de Qualidade de <i>Software</i> (GQS), conforme definido no Plano de Garantia de Qualidade de <i>Software</i> do projeto, não possuem as tarefas em harmonia de datas com o Cronograma do próprio projeto.
sicat_no_001	Plano de Gestão de Projeto e Plano de Gestão de Riscos não foram elaborados para o projeto. Tabela de Casos de Uso criada; mas não estava atualizada com o desenvolvimento do projeto.
sicod_no_001	Plano de Gestão de Risco requer, apenas, pequenos ajustes nas ações de contingência.
silegdep_no_001	Plano de Gestão de Projeto não elaborado. Plano de Garantia de Qualidade de <i>Software</i> necessita definir quais são os membros do GQS e o Plano de Gestão de Configuração e Ambiente não define quais <i>baselines</i> deverão ser estabelecidas para o projeto.
sisgraf_no_001	Plano de Gestão de Projeto, Plano de Gestão de Riscos, Plano de Garantia de Qualidade de <i>Software</i> e Cronograma não criados para o projeto. Plano de Gestão de Configuração e Ambiente não define nenhuma <i>baseline</i> para o projeto. Sumário Executivo não contempla a descrição da arquitetura proposta para o sistema.

Sumariando, os principais problemas identificados nos projetos:

- baixo nível de gerenciamento. Os planos dos projetos, quando existentes, não eram obedecidos ou sistematicamente atualizados. Também ficou evidente que seus cronogramas eram descompassados e, em sua maioria, não retratavam a realidade ou as suas necessidades; os cronogramas também não eram utilizados para o controle de desenvolvimento dos respectivos projetos. Precários planejamentos e gerenciamentos dos riscos, o que inviabilizou algumas decisões neles tomadas, felizmente revistas;
- grande dificuldade de planejamento e estabelecimento de *baselines* para os artefatos dos projetos e, conseqüentemente, inabilidade para univocamente segregar as versões adequadas dos sistemas e para manter a rastreabilidade entre os requisitos de *software* e sua implementação;
- dificuldade e falha constante na comunicação e na colaboração entre as equipes da própria COENS e com as demais Coordenações do CENIN; e - abandono, em alguns projetos, dos processos estabelecidos quando suas equipes enfrentavam restrições de tempo ou pressões dos interessados pelos sistemas.

Em contrapartida, também foram identificadas inúmeras oportunidades de melhoria, tanto no PROCEDE, quanto na organização da Coordenação e em suas equipes de trabalho. O planejamento de *baselines*, por exemplo, assim como estava definido no PROCEDE, foi substancialmente simplificado e estudos estão se desenvolvendo para promover a maior automação possível utilizando-se as ferramentas de apoio ao desenvolvimento, objetivando mecanizar os passos necessários à implementação das *baselines* e diminuir a interferência humana.

Além dessas, outras iniciativas também se fazem necessárias para se incrementar a qualidade geral dos projetos. Assim, de posse dessas informações, foi sugerida à direção da COENS a implementação de três projetos para o ano de 2007, com vistas na solução dos problemas identificados pela execução do processo de controle e garantia de qualidade de *software*:

- a) elaboração e execução de um plano para institucionalização do PROCEDE;
- b) avaliação de ferramentas de *software*, com vistas na aquisição de solução para automação e apoio das atividades de gerenciamento de projetos; e
- c) instalação de comitês técnicos, formados por especialistas nos diversos assuntos inerentes ao desenvolvimento de *software*, do próprio quadro efetivo de servidores da Casa, para auxiliar o Gestor do PROCEDE na sua constante manutenção, adaptação e evolução.

Por fim, puderam ser observados dois outros importantes resultados diretamente relacionados com a execução deste projeto: o primeiro foi a publicação de artigo científico no “VII Congresso de Qualidade em Produção de Software”, realizado na Universidade Federal de Lavras – UFLA, em Lavras (MG), no qual se descreve em pormenores o modelo proposto e implementado para o processo de controle e garantia de qualidade de *software*. O segundo, também de grande valor, apesar de imaterial, foi a desmistificação do assunto “qualidade de *software*”. Previamente à existência de um processo estabelecido para controle e garantia de qualidade de *software*, muitos dos colaboradores do corpo técnico entendiam que o assunto carecia de valores e dados reais e que se apresentava de maneira muito abstrata para ser devidamente implementado nas rotinas cotidianas de trabalho. O projeto conseguiu apresentar tarefas, dados, atribuições e responsabilidades concretas, que encerraram a alusão à abstração.

5 Aplicabilidade dos Resultados

O processo de controle e garantia de qualidade de *software* desenvolvido pode ser adotado por qualquer organização que possua um segmento interno de desenvolvimento de sistemas de informações, especialmente naquelas que possuem algum processo estabelecido. Ademais, como sua execução não requer uma numerosa equipe, é possível sua implantação em empresas ou instituições de pequeno porte, que não possuam numerosas equipes de trabalho.

6 Características Inovadoras

A execução do processo de controle e garantia de qualidade desenvolvido pelo Centro de Informática da Câmara dos Deputados resultou na identificação de uma característica muito significativa: ele pode ser facilmente parametrizado para emprego em projetos em geral, que se desenvolvam conforme modelos diversos

de processo daqueles a partir do qual foi originalmente concebido. Há, ainda, indicações que este mesmo processo também possa ser aplicado para controle e garantia de qualidade em projetos de outras áreas negócio, não necessariamente se restringindo ao escopo do desenvolvimento e manutenção de *software*.

O processo, conforme foi concebido, apresenta uma estrutura bastante modular, o que facilita sua adaptação. Sua aplicação em outro contexto requer, inicialmente, que se construa uma base de critérios de avaliação adequada ao contexto no qual se deseja aplicá-lo. As demais atividades e artefatos de suporte que o processo requer, podem ser reutilizados com pequenas adequações.

7 Conclusão e Perspectivas Futuras

O empreendimento resultou na apresentação, pelo CENIN, de um processo para o controle e garantia de qualidade de projetos de desenvolvimento e manutenção de *software* que já foi implantado e está em plena operação.

Tal processo, elaborado a partir das atividades da norma NBR ISO/IEC 12207 e recomendações da norma ISO/IEC 15504-5 e do modelo *CMMI*, é formado por dez atividades e um fluxo de trabalho, além dos modelos dos artefatos que, em conjunto, compõem seus instrumentos de trabalho. Também foi elaborado um roteiro para a implantação do processo, onde a recomendação para sua execução sobre um projetopiloto, destinado à identificação de eventuais necessidades de ajustes, se mostrou bastante prudente e proveitosa.

A proposição deste processo não encerra o trabalho pela contínua busca e garantia de qualidade nos produtos de *software*. Ao contrário, apenas dá forma a uma iniciativa que, seguramente, pode ser aperfeiçoada, principalmente mediante a adição de mecanismos de automação ou mesmo a adoção de novos paradigmas de controle e garantia de qualidade, nestes incluídos os operacionalizáveis por intermédio de rotina e os emergentes pelas vias atitudinais.

O processo criado já contribuiu significativamente para o incremento de qualidade, tanto dos produtos elaborados pelo Centro de Informática, quanto para seu próprio processo de desenvolvimento, o PROCEDE. Novas iniciativas já se vislumbram, como, por exemplo, a automação de tarefas de avaliação de qualidade através de sistemas de *workflow*, a comparação dos processos executados pela COENS com os modelos de referência, especialmente com o modelo MPS.BR e a elaboração de *checklists* para os diversos processos e artefatos. Todas deverão ser em breve implementadas, para que o controle e garantia de qualidade de *software* também obtenha ganhos de desempenho.

8 Referências Bibliográficas

[1] SALVIANO, Clênio F. **Melhoria e Avaliação de Processo com ISO/IEC 15504 (SPICE) e CMMI**. 2ed. Lavras: Universidade Federal de Lavras – UFLA/FAEP, 2004. 127 p.

[2] PROCEDE. **Web site do Processo CENIN de Desenvolvimento de Sistemas**. <http://prod.camara.gov.br/processo/>. Disponível na *Intranet* do Centro de Informática da Câmara dos Deputados. Acesso em 18 dez. 2006.

- [3] KRUCHTEN, Philippe. **The Rational Unified Process – An Introduction**. 2ed. Reading, Massachussets, EUA. Addison Wesley, 2000. 298 p.
- [4] ABREU, Estela dos Santos; TEIXEIRA, José Carlos Abreu. **Apresentação de Trabalhos Monográficos de Conclusão de Curso**. 6. ed. rev. Niterói: EdUFF, 2003.
- [5] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO/IEC 12207.1998: tecnologia de informação: processos de ciclo de vida de *software*. Rio de Janeiro,1998. 35p.
- [6] VASCONCELOS, Alexandre Marcos Lins de; MACIEL, Teresa Maria de Medeiros; ROUILLER, Ana Cristina. **INTRODUÇÃO À ENGENHARIA DE SOFTWARE E AOS PRINCÍPIOS DE QUALIDADE**. Lavras: Universidade Federal de Lavras – UFLA/FAEP, 2004. 128 p.
- [7] FILIPAK MACHADO, Cristina Ângela. **DEFININDO PROCESSOS DO CICLO DE VIDA DE SOFTWARE USANDO A NORMA NBR ISO/IEC 12207**. Lavras: Universidade Federal de Lavras – UFLA/FAEP, 2003. 101 p.
- [8] PÁDUA FILHO, Wilson de. **ENGENHARIA DE SOFTWARE**. Fundamentos, Métos e Padrões. 2.ed. Rio de Janeiro: LTC, 2003. 602 p.
- [9] PRESSMAN, Roger S. **ENGENHARIA DE SOFTWARE**. 5.ed. Rio de Janeiro: McGraw-Hill, 2002. 843 p.

[2.43] Gerência de Configuração Proposta de Metodologia Híbrida

Entidade: Instituto de Ensino Superior FUCAPI - CESF, Manaus , Brasil,

Autor: Nilmara Salgado - nilmarasalgado@gmail.com.br

1. Introdução

Vive - se em um período de grande concorrência empresarial. Esse aspecto não é diferente entre as empresas de desenvolvimento de *software*.

Outro ponto visível é a complexidade na criação dos seus produtos. Por isso surgiu a necessidade de adaptar a produção de *software* conforme as exigências de mercado.

Buscando atingir a qualidade que o mercado procura e garantir o prazo determinado, as empresas recorreram à adoção de metodologias de desenvolvimento.

Dentro dessas metodologias surgiu a gestão de configuração de *software* que é o foco desta pesquisa.

Atualmente, são poucas as empresas de desenvolvimento de *software* que adotam um processo de gerência de configuração. Porém, controlar documentações , seguir padrões , documentar códigos, utilizar ferramentas , criar históricos e outras atividades dentro de todo ciclo de vida do sistema é de suma importância para um projeto. Por tanto, ao propor essa metodologia, este trabalho busca incentivar a melhoria constante dos processos de desenvolvimento de *software* .

2. Objetivo e Justificativa

Visar a criação de uma metodologia de gerência de configuração híbrida sob os modelos de processo de desenvolvimento de *software Rational Unified Process* (RUP) e *Extreme Programming* (XP), adotando como modelo de maturidade e de *software* Melhoria de Processos do Software Brasileiro (MPS.BR), para atingir micros, pequenas e médias empresas de desenvolvimento de *software* em Manaus é o objeto deste estudo para o trabalho de conclusão de curso. Para tanto faz-se imprescindível dividi-lo em duas fases. A primeira busca verificar a viabilidade da proposta, a qual será o escopo deste artigo, a ser realizada através de investigação referente ao processo de Gerência de Configuração, metodologias RUP e XP e modelo de maturidade MPS.BR.

Sabe-se que já existe esse processo dentro dessas metodologias, portanto a idéia é criar um processo de Gerência de Configuração híbrido que atue como agente facilitador na implantação dessa disciplina e que possa contribuir para adaptação da equipe, lembrando que a sua adoção tem que vir sempre de cima para baixo, ou seja, da gerência superior.

3. Gerência de Configuração (GC)

Após pesquisa de campo, efetuando entrevistas com profissionais da área, aliada a averiguação realizada através do raciocínio indutivo, segue uma síntese da investigação.

3.1. O que é Gerência de Configuração

Gestão de configuração de *software* (*Software Configuration Management*, SCM), é uma atividade guarda-chuva¹ que é aplicada ao longo de todo o processo de *software*. Como modificações podem ocorrer em qualquer época, às atividades de SCM são desenvolvidas para (1) identificar modificações, (2) controlar modificações, (3) garantir que as modificações sejam adequadamente implementadas e (4) relatar as modificações a outros que possam ter interesse [1].

3.2. Histórico

Um breve histórico mostra que a Gerência de Configuração surgiu após a Segunda Grande Guerra entre as técnicas que visavam melhorar a qualidade dos produtos e suprir as necessidades de gerenciar alterações documentais da indústria aeroespacial norte-americana.

1 Atividades guarda-chuva são atividades de apoio aplicadas ao longo de todo o processo de *software*.

Nas duas décadas seguintes a GC passou a controlar também o *software* desenvolvido.

Mas, foi por meio da associação de Instituições, tais como: *Institute of Electrical and Electronics Engineers* - IEEE, *International Organization for Standardization* - ISO, *Electronic Industries Alliance* - EIA e *American National Standards Institute* e ANSI ocorreram definições de padrões de informatização para o desenvolvimento

de *software* . E a partir da definição do *Capability Maturity Model* CMM ® a GC se estabilizou como processo fundamental da Engenharia de Software [9].

3.3. Objetivos

A Gerência de Configuração veio preencher os espaços vazios de como, quando e o quê gerenciar dentro de um projeto. Portanto, sua finalidade é identificar e documentar itens de configuração ao longo do ciclo de vida visando sua integridade , manutenibilidade e rastreabilidade.

É importante ressaltar que não pode ser um processo burocrático que cause transtornos e atrasos dentro do projeto.

3.4. Conceitos fundamentais

Os conceitos descritos a seguir dão subsídios para um melhor entendimento do processo de GC.

Item de configuração (IC) – é a informação criada como parte do processo de engenharia de software [1].

Linha-base (*baseline*) – o IEEE Std nº 610.12- 1990 define linha- base como uma especificação ou produto que foi formalmente revisto e aprovado, o qual daí em diante serve como base para o desenvolvimento futuro e que pode ser modificado apenas por meio de procedimentos formais de controle de modificação [1].

Repositório – um banco de dados que atua como o centro tanto para acumulação quanto para armazenamento de informação de engenharia de *software* [1]. Pode-se dizer também que é uma área do servidor ou um servidor onde os itens de configuração estão fisicamente armazenados .

Espaço de trabalho (*workspace*) - a equipe não trabalha diretamente nos arquivos do repositório. Cada membro possui seu *workspace* , que é o espelho do repositório, trabalhando isoladamente em suas atividades .

Desenvolvimento concorrente (*branch*) – é uma ramificação da linha principal do projeto.

Cada projeto possui uma linha original e, ao longo do desenvolvimento, normalmente é necessário criar versões sem comprometer a linha principal.

Sincronização de mudanças (*merger*) – em algum momento do desenvolvimento essas ramificações precisam voltar à linha principal, ou seja, precisam ser mescladas . Outra forma de utilização do *merger* se dá quando dois ou mais membros da equipe trabalham em partes diferentes de uma mesma atividade e em determinado momento precisa ser mesclada.

Observa-se na Figura 1 uma demonstração gráfica dos conceitos de *branch* e *merger* .

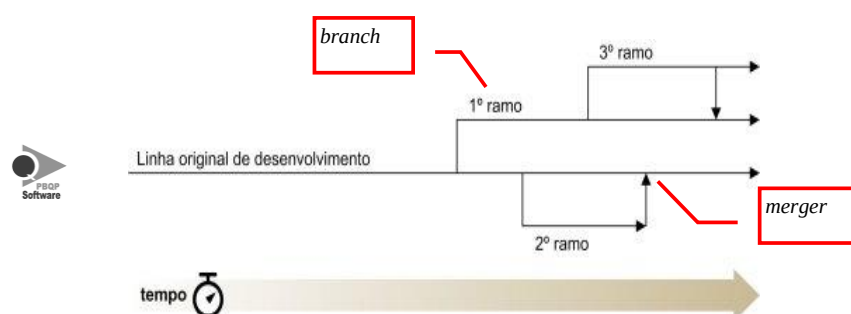


Figura 1 - Demonstração de *branch* e *merge*
Fonte: Encontrada no site [10] (2006)

3.6 Normas que contribuem para GC

Conforme apresentado em [7], dentre as contribuições de Instituições Internacionais que valorizaram a Gerência de Configuração de Software (GCS) encontram-se as seguintes:

IEEE STD 828-1998, *Standard for Software Configuration Management Plans*, define uma estrutura básica para o plano de GCS, estabelecendo os itens que devem ser abordados e uma sucinta explicação para cada um deles.

IEEE STD 1042-1986, *Guide for Software Configuration Management*, é um guia de apoio para a aplicação da norma de plano, fazendo comentários adicionais ao seu preenchimento.

ISO 10007, *Quality Management – Guidelines for Configuration Management*, é um guia geral para gerência de configuração, enfocando produtos de engenharia, não sendo específica para software. Procura cobrir todo o ciclo de desenvolvimento de um produto e foi concebida como forma de atender os requisitos de GCS estabelecidos na família de normas ISO9000.

ISO/IEC 12207, *Information Technology – Software Life Cycle Process*, define uma estrutura básica (framework) para processos do ciclo de vida de software. É uma norma bastante sucinta dedicando uma página para cada processo, e seu objetivo maior é estabelecer uma base para o entendimento comum sobre o software e sua evolução.

ISO/IEC TR 15846, *Information Technology – Configuration Management for Software*, é um relatório técnico que detalha o processo de GCS definido pela ISO/IEC 12207. Segundo as normas ISO/IEC 12207 e ISO/IEC TR 15846, o processo de GCS é caracterizado pelas seguintes atividades:

- Implementação do Processo;
- Identificação da Configuração;
- Controle da Configuração;
- Relato da Situação da Configuração;
- Gerência de Liberação e Distribuição.

3.6 Metodologias

As normas especificam “o quê” deve ser feito e as empresas perguntaram como fazer. A forma encontrada foi a adoção de metodologias. Deve-se ressaltar que as

metodologias serão utilizadas como referência e as empresas precisam aplicá-las segundo as suas realidades.

3.6.1. Extreme Programming (XP)

Metodologia de desenvolvimento que se baseia em doze princípios, considerada ágil, não sendo obrigatório o uso de artefatos.

A Figura 2 apresenta o modelo de processo do XP. Após uma abordagem verificou-se a possibilidade de agregar GC a um processo que utiliza metodologia XP sem afetar diretamente seus princípios através de suas *User Stories*, *Release Planning* e *Test Scenarios*.

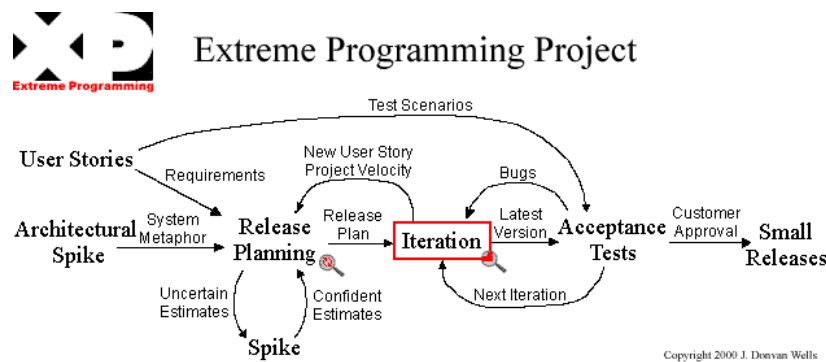


Figura 2 - Modelo de processo do XP

3.6.2. Rational Unified Process (RUP)

O RUP é um processo de engenharia de software organizado em disciplinas e que se caracteriza pela grande quantidade de artefatos a cada atividade. Trata o desenvolvimento de um sistema de forma iterativa e incremental. Observa-se na Figura 3 que a GC está presente em praticamente todas as fases de um projeto.

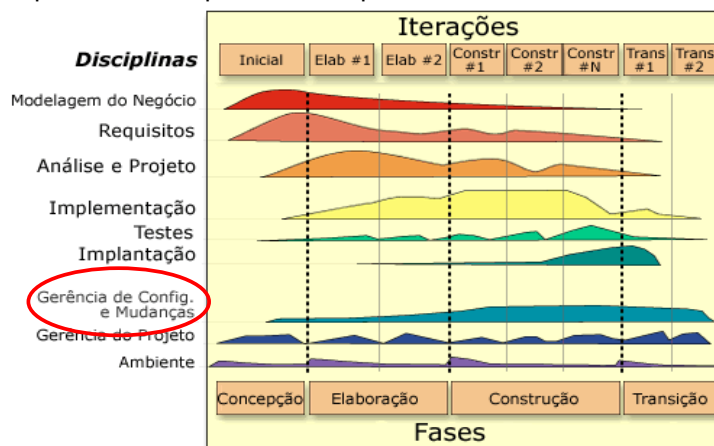


Figura 3. Modelo RUP

3.7 Pesquisa de Campo

Para entender as duas metodologias escolhidas foi realizada uma pesquisa de campo² em dois Institutos que são referência em inovação tecnológica: Instituto Certi Amazônia – ICA e Genius Instituto de Tecnologia. O primeiro com desenvolvimento voltado para os princípios do XP. Já o Instituto Genius necessitou para o desenvolvimento de seus projetos de um processo mais complexo.

3.7.1 Instituto Certi Amazônia - ICA

O ICA utiliza os princípios do XP, SCRUM e PMBOK (*Project Management Body of Knowledge*). O desenvolvimento de seu projeto possui as seguintes características: curto prazo, equipes pequenas, entregas semanais, presença constante do cliente.

A ferramenta *Trac*³ garante o sincronismo do desenvolvimento, pois, trata-se de um projeto distribuído.

Para garantir o sucesso no seu processo de GC foram realizados vários treinamentos, seminários e vídeo conferências.

Com a implantação adequada da GC o Instituto obteve vários benefícios, tais como: documentação controlada e atualizada, controle das versões, históricos dos seus itens de configuração bem como melhoria no processo de comunicação com a equipe.

3.7.2 Genius Instituto de Tecnologia

O Instituto Genius enfrentou várias dificuldades antes da adoção do processo de GC, pois, o mesmo possui uma diversidade grande de projetos, tanto de *hardware* como de *software*.

Como o intuito do Instituto era ter um processo facilitador para auxiliar no desenvolvimento de seus projetos, foram tomadas as seguintes providências: aquisição uma ferramenta, no caso, *Rational ClearCase*⁴, treinamento para adaptação da equipe, palestras, acompanhamentos e auditorias; objetivando resultados em médio prazo.

A Gerência de Configuração adotada pelo Instituto trouxe vários benefícios, dentre eles pode-se citar: controle de códigos, disseminação do conceito, melhoria no processo de comunicação com a equipe bem como, abrangência para processos

² Pesquisa realizada em abril de 2006. Revisada em novembro de 2006.

³ *Trac* – fornece controle de versão e controle de mudança bem como documentação e visualização do repositório.

⁴ *Rational ClearCase* – fornece o gerenciamento durante todo o desenvolvimento de um projeto. Com controle de versão, espaço de trabalho, desenvolvimento paralelo, construção e release.

administrativos, por exemplo, documentos de Recursos Humanos. Atualmente existe até uma forma bem humorada na empresa de tratar seus artefatos. Quando algum membro pergunta por um artefato o outro já responde: “Ta na VOB?”, referindo-se ao repositório.

3.8 Gerência de Configuração e MPS.BR

Como a proposta é criar uma metodologia de GC, se faz necessário embasar-se das normas e ter a capacidade de atender e passar por um processo de qualidade e maturidade de software e Melhoria do Processo de Software Brasileiro (MPS.BR) foi o escolhido.

Segundo [6], uma das metas do MPS.BR visa definir e aprimorar um modelo de melhoria e avaliação de processo de software, visando preferencialmente as micro, pequenas e médias empresas, de forma a atender as suas necessidades de negócio e ser reconhecido nacional e internacionalmente como um modelo aplicável à indústria de software. O MPS.BR também estabelece um processo e um método de avaliação, o qual dá sustentação e garante que o MPS.BR está sendo empregado de forma coerente com as suas definições.

Conforme Figura 4, o Modelo de Referência MR-MPS define níveis de maturidade que são uma combinação entre processos e sua capacidade.

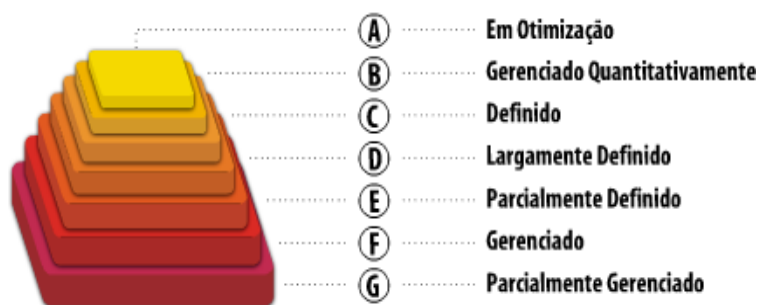


Figura 4. Níveis de maturidade – MPS.BR

A capacidade é representada por um conjunto de atributos descrito em termos de resultados esperados. A capacidade do processo expressa o grau de refinamento e institucionalização com que o processo é executado na organização.

A capacidade do processo no MPS possui cinco (5) atributos de processos (AP) que são: AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2. Cada AP está detalhado em termos de resultados esperados do atributo de processo (RAP) para alcance completo do atributo.

Os níveis de maturidade estabelecem patamares de evolução de processos, caracterizando estágios de melhoria da implementação de processos na organização. O nível de maturidade em que se encontra uma organização permite prever o seu desempenho futuro ao executar um ou mais processos.

3.8.1. Fluxo de Atividades GC

Apresenta-se na Figura 5 o fluxo de atividades de Gerência de Configuração sob a visão do MPS.BR.

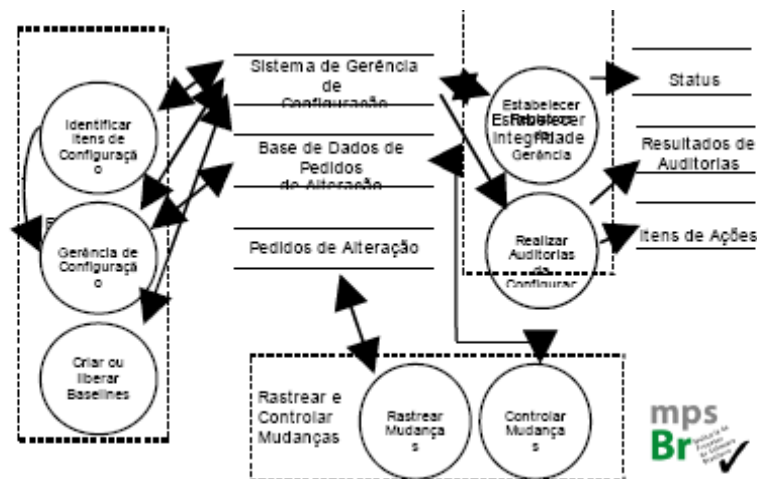


Figura 5. Fluxo de atividades preparado por Ana Regina, Dra.⁵
Fonte: Curso Oficial – Introdução ao Modelo de Referência para Melhoria de Processo de Software (2004).

3.8.2 Nível MR-MPS: F - Gerenciado

Cabe ressaltar que o MPS.BR está dividido em níveis de maturidade e o processo de gerência de configuração está no nível F – Gerenciado.

Cada processo possui um propósito, bem como resultados esperados. Conforme [6] o propósito do processo de Gerência de Configuração é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto e disponibilizá-los a todos os envolvidos. Os resultados esperados são:

GCO 1. Os Itens de Configuração - I.C. são identificados;

GCO 2. Os I.C. gerados pelo projeto são definidos e colocados sob uma linha base;

GCO 3. É estabelecido e mantido um Sistema de Gerência de Configuração;

GCO 4. As modificações e liberações dos I.C. são controladas;

GCO 5. As modificações e liberações são disponibilizadas para todos os envolvidos;

GCO 6. A situação dos I.C. e as solicitações de mudanças são registradas, relatadas e o seu impacto é analisado;

GCO 7. A completeza e a consistência dos I.C. são asseguradas;

GCO 8. O armazenamento, o manuseio e a entrega dos produtos de trabalho são controlados;

GCO 9. A integridade das linhas bases (*baselines*) é estabelecida e mantida, através de auditoria da configuração e de registros da Gerência de Configuração.

3.9. Pontos em Comum

Pode-se citar no Quadro 1, no que diz respeito às metodologias de desenvolvimento escolhidas para pesquisa, algumas características em comum que contribuem para viabilidade da proposta.

Característica	Descrição
Adoção de modelo de maturidade	O propósito da GC não é definir quando e como devem ser realizadas as modificações nos artefatos, papel este reservado ao processo de desenvolvimento.
Desenvolvimento iterativo incremental	A cada iteração uma parte do sistema é desenvolvida, sendo o produto de cada nova iteração superior à da iteração anterior.
Integração Contínua	Uma das características do desenvolvimento iterativo é a integração contínua, que torna possível manter o sistema integrado durante todo o seu desenvolvimento.
Refatoração	Para que o sistema possa evoluir de forma incremental, a equipe deve fazer com que ele expresse os seus objetivos facilmente e esteja sempre claro e fácil de compreender. Observa-se uma restrição para grandes sistemas, pois é necessária uma definição de arquitetura.
Código Padronizado	Torna-se necessário para que possa haver compartilhamento entre os programadores.

Quadro 1 - Pontos em comum entre as metodologias RUP e XP
Fonte: Encontrada no site [5, 16, 17, 18] (2006, 2006, 2004, 2004).

4. Resultados Relevantes

Dentre os resultados que este projeto alcançou assinala-se os de contexto acadêmico, dentre os quais encontram-se:

- Abordagem do processo de Gerência de Configuração e Metodologias de Desenvolvimento;
- Abordagem das Normas Técnicas relacionadas ao processo;
- Divulgação do Modelo de Maturidade de Software Brasileiro MPS.BR.

Ressalta-se também apresentação desta proposta na “I Mostra de Educação Tecnológica FUCAPI”, no dia 17 de novembro de 2006.

Há, ainda, o desenvolvimento de trabalho acadêmico que será apresentado em junho de 2007.

5. Aplicabilidade e Impacto

A aplicabilidade está na forma de contribuição para propagar, no que diz respeito aos conhecimentos de processo de Gerência de Configuração e modelos de desenvolvimento e maturidade, fornecendo uma primeira visão de que o mesmo pode ser aplicado de forma eficiente e eficaz na melhoria de um processo dentro de micros, pequenas e médias empresas de desenvolvimento de *software* em Manaus, conseqüentemente, trazendo impacto positivo para qualidade do produto.

6. Características Inovadoras

Este projeto inova no que se refere à divulgação e transferência de conhecimento na área de processo de Gerência de Configuração, apresentando características de adaptabilidade, tanto para equipe quanto para empresa, uma vez que a mesma não precisa ficar restrita ao uso de uma metodologia.

7. Conclusão e Perspectivas Futuras

Após o estudo realizado, observa-se que é possível criar uma metodologia híbrida de Gerência de Configuração que atue como agente facilitador e que possa ser utilizada tanto no RUP quanto no XP tendo como modelo de maturidade de software o MPS.BR. Cabendo ainda lembrar que GC não deve ser dependente de ferramentas e que visa contribuir para adaptação e formalização dos princípios que devem nortear a equipe no que concerne ao controle de mudanças dentro do processo do ciclo de vida do software, e que nesse contexto sua utilização deve se dá dos níveis hierárquicos mais altos até os colaboradores envolvidos.

Para segunda fase do trabalho está previsto a criação de um processo e seus respectivos artefatos utilizando-se de pesquisa aplicada para demonstrar a aplicabilidade da proposta.

8. Referências

- [1] PRESSMAN, Roger S. Engenharia de Software. – 6.ed.- Rio de Janeiro:

McGraw-Hill, 2006

[2] ARAUJO, Kleitor Franklinton Correa de. Engenharia para Web – Análise das metodologias XP e RUP sobre o contexto da KFC Araújo – 2005.

[3] VILLAS-BOAS, André. I Encontro de Implementadores MPS.BR – Mini-Curso Implementando o Nível F – 29/08 a 01/09/2005.

[4] ROCHA, Ana Regina. Curso Oficial – Introdução ao Modelo de Referência para Melhoria de Processo de Software, setembro 2004.

[5] MPS.BR – Melhoria de Processo do Software Brasileiro. Guia de Implementação – Parte 2: Nível F Versão 1.0. Softex, dezembro 2006. Disponível em: <http://www.softex.br/portal/mpsbr/_guias/default.asp>. Acesso em: 04/02/2007.

[6] MPS.BR – Melhoria de Processo do Software Brasileiro. Guia Geral. Versão 1.1. Softex, maio 2006. Disponível em: <http://www.softex.br/portal/mpsbr/_guias/default.asp>. Acesso em 04/10/2006.

[7] Gerência de Configuração. Disponível em: <<http://www.cin.ufpe.br/~gfn/qualidade/gc.html>>. Acesso em 07/03/2006.

[8] Aplicando práticas de eXtreme Programming(XP) em equipes SW-CMM nível 2. Disponível em: <http://www.simpros.com.br/simpros2004/Apresentacoes_PDF/Artigos/Art_05_Simpros2004.pdf>. Acesso em 04/03/2006.

[9] A Hora da GCS. Disponível em: <http://www.serpro.gov.br/publicacoes/tema_175/materias/art_175_02>. Acesso em 07/03/2006.

[10] O que é Gerência de Configuração. Disponível em: <http://www.pronus.eng.br/artigos_tutoriais/gerencia_configuracao/gere.php>. Acesso em: 07/03/2006.

[11] MPS.BR (Melhoria de Processo do Software Brasileiro). Disponível em: <http://www.cits.br/mps.do>. Acesso em: 07/03/2006.

[12] RUP – Conceitos Básicos. Disponível em: <http://w3.datasus.gov.br/pts/pts.php?area=4063A1B0C0D0E0F4063G3313HIJd1L2M0N&Vinclude=../site/texto.php>. Acesso em: 12/03/2006.

[13] Introduzione alla XP Extreme Programming. Disponível em: <http://www.jonathan.cristoforetti.name/programming/xp/xp.html>. Acesso em: 12/03/2006.

[14] Guidelines_for_Extended_Abstracts. Disponível em: http://www.metrologia2006.org.br/guidelines/downloads/Guidelines_for_Extended_Abstracts.pdf. Acesso em: 05/02/2007.

[15] Acreditar que Gerência de Configuração de Software é controle de versão é um engano. Disponível em: http://www-306.ibm.com/e-business/br/campaign/rational/columns_articles_gdd_gerency.shtml. Acesso em: 04/02/2007.

- [16] Comparação entre Metodologias RUP e XP. Disponível em: <http://www.ppgia.pucpr.br/~alcides/teaching/mestrado/FundamentosEngenhariaSoftware/artigos/ResumosApresentacoes/RUPvsXP_draft.pdf>. Acesso em: 04/02/2007.
- [17] Como usar o XP (extreme Programming) dentro do RUP. Disponível em: <<http://www.vhmartins.com/Documentos/Adriana.pdf>>. Acesso em: 04/02/2007.
- [18] Análise Comparativa das Metodologias de Desenvolvimento de Softwares Tradicionais e Agéis. Disponível em: <http://www.cci.unama.br/margalho/portaltcc/tcc2004/oscar.pdf>. Acesso em: 04/02/2007.

[2.44] Aumentando a capacidade produtiva de desenvolvimento de software e a qualidade através de uma rede de fornecedores certificados

Entidade: ¹ Companhia de Processamento de Dados do Estado do Rio Grande do Sul - www.procergs.rs.gov.br; ² SOFTSUL – Associação Sul-Riograndense de Apoio ao Desenvolvimento de Software - www.softsul.org.br; ³ Software Process Consultoria; ⁴ Qualità Informática; www.qualita.inf.br; ⁵ Advanced IT - www.advancedit.com.br

Autores: Ana Ruth Schirmer Garcia¹, Carlos Alberto Becker^{2,3}, Luís Alberto Richter⁴, Paulo Roberto de Miranda Samarani¹, Roberto Petry⁵, Vitor Hugo Berenhauser de Aguiar¹ - {ana-garcia,samarani,vitor}@procergs.rs.gov.br, carlos.becker@terra.com.br, richter@qualita.inf.br, roberto.petry@advancedit.com.br

Abstract. This article describes the experience of a public enterprise, an organization that supports the IT area development and some software enterprises of Rio Grande do Sul State to create a certified net suppliers to improve quality and the productive capacity in system development, using methodologies and technologies arose from ISO/IEC 12207.

Resumo. Este artigo descreve a experiência de uma empresa pública, de uma entidade representativa do setor de software e de empresas de software do Estado do Rio Grande do Sul na implementação de uma rede de fornecedores certificados, visando o aumento da qualidade e da capacidade produtiva de desenvolvimento de software do Estado, através da utilização das metodologias e técnicas derivadas da norma ISO/IEC 12207.

1. Introdução

Atualmente, as organizações desenvolvedoras de *software* cada vez mais são pressionadas pela entrega de produtos com maior qualidade, de forma mais rápida e a um custo menor. Ao mesmo tempo, o surgimento de novas tecnologias na área de hardware, redes e telecomunicações, e a diversidade de metodologias, métodos e técnicas de desenvolvimento de *software* têm aumentado a complexidade dos processos de ciclo de vida de *software* [ABN 12207], tais como Aquisição, Fornecimento e Desenvolvimento.

Os últimos anos têm demonstrado uma crescente demanda de trabalhos em Tecnologia da Informação e da Comunicação (TIC) pelo Governo, não somente para a automatização e gestão dos processos internos dos órgãos públicos estaduais, como também para a aplicação do conceito de Governo Eletrônico. Esse conceito estabelece a aplicação da TIC no aprofundamento da reforma da gestão pública, nos aspectos de melhoria da prestação de serviços, acesso à informação, redução de custos e controle social. Isto resulta na promoção da informatização da administração pública e no uso de padrões nos seus sistemas aplicativos, no fomento a aplicações em serviços de governo e na capacitação em gestão de Tecnologias de Informação.

Muitos modelos e normas de qualidade têm sido desenvolvidos na intenção de melhorar os processos dos produtos e serviços e alcançar uma maior

qualidade. Com o intuito de criar uma rede de produtores de *software* que trabalhem de forma integrada em um processo homogêneo e eficiente de produção, adotando as melhores práticas de engenharia de *software* e gerência de projetos, foi criado um projeto chamado Fábrica de *Software* RS - FSRS [FAB 2006]. Esse projeto consiste numa parceria entre a SOFTSUL (Associação Sul-Riograndense de Apoio ao Desenvolvimento de *Software*) e empresas representativas do setor de informática do Estado do Rio Grande do Sul com a PROCERGS (Companhia de Processamento de Dados do Estado do Rio Grande do Sul). Um dos produtos elaborados pela FSRS foi uma metodologia de desenvolvimento de *software* (MDS-FSRS), que teve como base a norma internacional NBR ISO/IEC 12207, além da consideração de diversos outros modelos e normas, tais como CMM, CMMI, ISO/IEC 15504 e PMBOK.

A partir das novas necessidades de mercado, da crescente demanda por TIC pelo Governo e das experiências obtidas a partir da FSRS com a integração de entidades e da metodologia gerada, iniciou-se um novo projeto, denominado Rede de Fornecedores Certificados PROCERGS (REDEPRO). Esse projeto se propôs, ainda, em considerar outros requisitos, tais como padronização de métodos, técnicas, uso de ferramentas e componentes. Para tanto, concentrou o foco em processos, tais como, aquisição, fornecimento, desenvolvimento e gerência [RED 2006].

O artigo está estruturado como segue. A seção 2 descreve os participantes que estão envolvidos com o modelo REDEPRO, a seção 3 descreve a importância de parcerias para o modelo e na seção 4 o modelo é descrito em mais detalhes, identificando-se os objetivos, papéis envolvidos e processos. Ainda nesta seção, são descritas brevemente, algumas experiências da utilização do modelo pelos envolvidos. Finalmente, as seções 5 e 6 contêm os resultados obtidos e as conclusões do artigo.

2. Projeto REDEPRO: envolvidos

O projeto elaborado pela PROCERGS, contou com a participação da SOFTSUL e das empresas do setor de *software* do Rio Grande do Sul.

A PROCERGS é uma empresa de economia mista, com 33 anos de atuação como órgão executor da política de informática do Estado. Atualmente é a maior empresa de informática e Internet do Rio Grande do Sul, tendo a responsabilidade de atender os órgãos públicos estaduais, cidadãos e empresas que utilizam seus produtos e serviços no dia-a-dia. A empresa tem trabalhado lado a lado com os demais órgãos do Estado, gerando soluções que contribuem para apoiar as ações do Governo, racionalizar e modernizar a máquina pública, melhorar os serviços prestados ao cidadão e aumentar a competitividade das empresas do Estado, através do emprego de telecomunicações e da tecnologia de informação.

A SOFTSUL é uma organização não-governamental constituída na forma de uma sociedade civil, sem fins lucrativos, de direito privado e de caráter público, filiada ao Programa SOFTEX - Programa para Promoção da Excelência do *Software* Brasileiro. A SOFTSUL é uma instituição que executa ações alinhadas às políticas públicas para desenvolvimento do setor de TIC.

O projeto REDEPRO conta com a importante participação de empresas, dentre as quais, algumas participaram do projeto FSRS desde o seu início. As empresas Advanced IT, Datum Informática Ltda.[DAT 2006], DBServer Assessoria em Sistemas de Informação Ltda.[DBS 2006], Qualità Informática Ltda. e Softdesign[SOF 2006], formaram um consórcio e são, atualmente, os fornecedores de serviços de construção REDEPRO.

3. Projeto REDEPRO: parcerias

Para que um projeto da magnitude do REDEPRO obtenha os ganhos desejados, acredita-se que seja necessário o fortalecimento e criação de parcerias entre a PROCERGS e seus fornecedores. Para tanto, é importante que ambos migrem para um patamar de maior amadurecimento nas suas relações.

Na tentativa de atender às necessidades citadas, o Projeto REDEPRO estabeleceu dentre suas definições técnicas, a utilização conjunta da metodologia de especificação da PROCERGS e da FSRS, por se tratar de metodologias que já vinham sendo utilizadas e por apresentarem forte aderência a normas e padrões internacionais.

4. Descrição do modelo REDEPRO

Nesta seção será apresentado, de forma sucinta, o modelo REDEPRO.

4.1. Os objetivos e benefícios

O Projeto REDEPRO tem como objetivo criar as condições técnicas, administrativas e legais, para que a PROCERGS, em parceria com as empresas do setor de TIC passem a investir na adequação de seus processos e produtos, de modo que: do ponto de vista técnico, seja transparente para a PROCERGS se o fornecedor é externo ou interno; do ponto de vista administrativo, seja um processo ágil e que não implique num acréscimo significativo na linha de produção interna para o atendimento a picos de demanda; do ponto de vista legal, obedeça à legislação vigente.

Dentre os principais benefícios, pode-se citar o fortalecimento do setor de TIC, gerando emprego, renda, crescimento das empresas do setor, aumento da arrecadação do Estado e agilização do atendimento aos clientes no desenvolvimento de sistemas, mantendo com a PROCERGS a inteligência sobre as regras de negócios dos projetos.

4.2. Os papéis

Para a operacionalização do projeto, devem ser considerados os seguintes papéis que interagem nos processos de certificação, de contratação e de execução:

- Empresas de desenvolvimento de sistemas: desenvolvem sistemas especificados pela PROCERGS com base na metodologia estabelecida;
- Agentes de Certificação: promovem cursos destinados a desenvolver e certificar profissionais e empresas em termos de metodologia e arquiteturas tecnológicas;
- Agente REDEPRO: atua como mediador entre a PROCERGS e os fornecedores, verificando a conformidade das especificações elaboradas pela PROCERGS com relação à metodologia prevista, validando o

dimensionamento do projeto com relação ao seu tamanho e monitorando a qualidade dos serviços e dos produtos gerados;

- Fábrica de *Software* RS: é responsável pela metodologia e padrões de especificação, atuando também, através de um das suas entidades coordenadoras, a SOFTSUL, como agente certificador da metodologia FSRS;
- PROCERGS: gerencia todo o processo, é responsável pela análise e projeto, pelas especificações, testes funcionais e de integração e pela implantação dos sistemas desenvolvidos, além de atuar como agente certificador nas arquiteturas tecnológicas.

4.3. O processo

Para a implementação do projeto REDEPRO, foi necessária a conjugação de três processos básicos (Certificação, Contratação, Execução), apoiados em três grandes pilares de sustentação: metodologia FSRS, arquiteturas metodológicas e tecnológicas utilizadas na PROCERGS e código de ética do relacionamento da PROCERGS com seus fornecedores. Esses processos podem ser visualizados através da figura 4.1.

4.3.1. Processo de Certificação

O processo de certificação possui como agentes a SOFTSUL e a PROCERGS. A SOFTSUL, que vem coordenando juntamente com a PROCERGS desde 1998 o projeto FSRS, com a colaboração de várias empresas do mercado, promove os cursos sobre a metodologia FSRS. Este curso é pré-requisito para os cursos de certificação promovidos pela PROCERGS para os ambientes tecnológicos utilizados nos seus projetos.

4.3.2. Processo de Contratação

A implementação da REDEPRO prevê a contratação do Agente REDEPRO e dos fornecedores de serviços de construção (programação e testes). No edital para a contratação de fornecedor dos serviços de construção foi utilizado o critério “técnica e preço”, onde existe expressiva pontuação para os profissionais e empresas certificadas. Já no edital para a contratação do Agente REDEPRO, também com o critério “técnica e preço”, além da valorização das certificações, foi considerada a experiência dos profissionais em consultoria no desenvolvimento de sistemas orientados a objeto, a participação em programas federais na área de TI e a experiência em consultoria a empresas em projetos de certificação CMM, CMMI e MPSBR.

4.3.3. Processo de Execução

Contratados o Agente REDEPRO e o fornecedor de serviços de construção, inicia-se o processo de produção propriamente dito. Neste processo, a PROCERGS possui os sistemas a serem desenvolvidos, nos quais executa os papéis de Gerência de Projeto, Análise, Projeto e Testes, produzindo as especificações para a construção. Estas especificações são entregues para o Agente, que as analisa para verificar a conformidade em relação às metodologias PROCERGS e FSRS.

O método utilizado para o dimensionamento de sistemas (estimativa de tamanho) na REDEPRO está baseado em UCPs (*Use Case Points*). Compete também ao Agente verificar o correto emprego do método.

Segundo o modelo REDEPRO, a remuneração dos fornecedores dos serviços de construção é realizada a partir do tamanho do *software* (número de UCPs), substituindo o método tradicional que tem, como base, o número de horas.

Somente após o dimensionamento, as especificações são entregues ao fornecedor dos serviços de construção, que tem um prazo definido para dar o seu aceite e iniciar a construção, segundo métricas previamente estabelecidas, podendo ainda convocar o Agente para o consenso das estimativas.

Durante todo o processo de construção, são realizadas reuniões periódicas entre a Gerência do Projeto na PROCERGS, o Coordenador Técnico do fornecedor dos serviços de construção e o Agente REDEPRO. A partir destas reuniões, são feitos os ajustes no andamento dos trabalhos, sempre com a mediação do Agente que, ao final de cada etapa e do projeto, emite relatórios de avaliação dos serviços e produtos gerados.



Fig. 4.1 - Processos da REDEPRO

4.4. O relato da experiência

4.4.1. Os processos do Agente REDEPRO

O Agente REDEPRO foi criado para atuar como uma terceira parte neutra, mediando o relacionamento que a PROCERGS estabelece com seus fornecedores.

Inicialmente, os processos de trabalho definidos previam a execução de atividades diferentes e com pouco relacionamento entre si: Avaliação da Conformidade, Classificação e Avaliação dos Produtos. Porém, a experiência demonstrou que as atividades citadas possuem alto grau de inter-relacionamento,

uma vez que se baseiam no mesmo contexto de processos a serem seguidos e se referem aos mesmos projetos de desenvolvimento de *software*.

Essa constatação motivou um redesenho dos processos originais, unificando os processos em torno de um único papel (Agente) responsável por todas as atividades.

A figura 4.2 exibe o fluxo geral de informações no contexto dos projetos REDEPRO e destaca as principais atividades e documentos envolvidos.

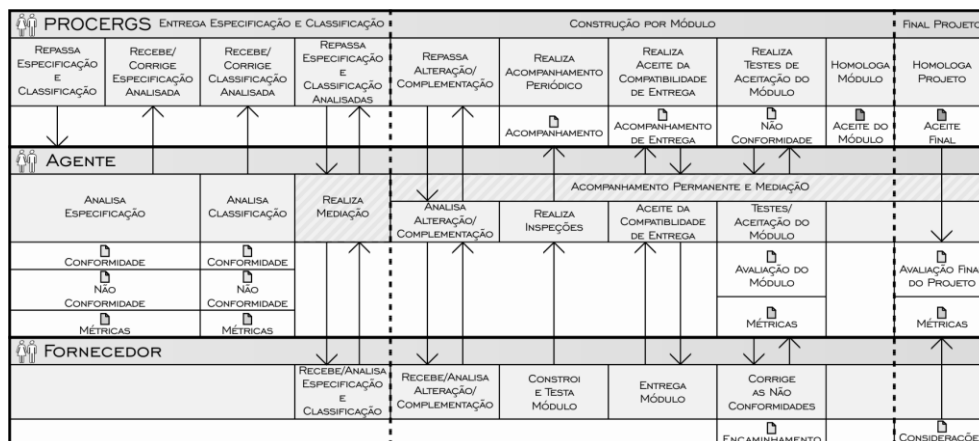


Figura 4.2 - Fluxo geral de informações

As atividades do Agente REDEPRO são mais intensas nas etapas preliminares do projeto, quando o pacote ou módulo de especificações é repassado para uma avaliação de conformidade e dimensionamento.

Posteriormente, durante o desenvolvimento, o Agente atua como mediador em situações de conflito entre PROCERGS e Fornecedor, além de coletar informações do projeto para elaboração de relatórios intermediários e finais.

O Agente ainda possui participação importante nos eventos de entrega de produto, quando acompanha o aceite de compatibilidade de produto e coleta informações sobre quantidades de defeitos relatados e corrigidos, bem como a sua responsabilidade.

4.4.2. Os processos do Fornecedor de serviços de construção REDEPRO

Para a execução das responsabilidades atribuídas ao fornecedor, o consórcio concentrou-se em alguns pilares, como o Processo Gerencial entre as partes contratuais envolvidas, a definição do Ciclo de Vida do Projeto e sua Gerência, o Processo de Construção propriamente dito e a Garantia de Qualidade.

O Processo Gerencial foi estabelecido com a constituição de um Comitê Gerencial, concebido para ser um fórum de apoio na resolução de conflitos, indefinições ou confirmação de acordos gerados na esfera técnica. Seu funcionamento ocorreu através de reuniões periódicas (quinzenais) entre todos os envolvidos (contratante, fornecedor e agente) e suas decisões foram formalizadas através de atas.

A definição do ciclo de vida do projeto constituiu-se na seleção das atividades necessárias para o cumprimento das responsabilidades assumidas e esteve fortemente embasado nas experiências vivenciadas pelas diversas empresas que compõem o consórcio. Foram estabelecidos também os percentuais de cada uma das atividades previstas, que serviram para definição do perfil da equipe e constituição do cronograma do projeto, conforme demonstra a figura 4.3.

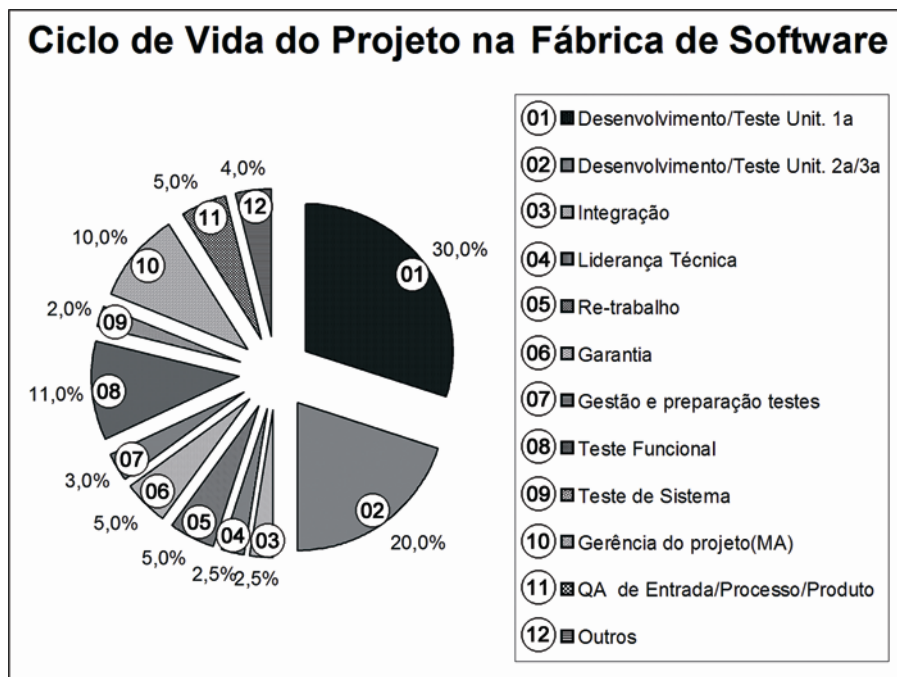


Fig. 4.3 - Ciclo de Vida do Projeto na Fábrica de Software - Consórcio REDEPRO

O Gerenciamento do Projeto utilizou-se de artefatos para planejamento e controle, destacando-se Plano de Projeto, Cronograma, Plano de Comunicação, Registro de atividades e horários e *Status Report*, bem como reuniões semanais de acompanhamento, envolvendo o Gerente de Projeto e empresas do consórcio.

O Processo de Construção considerou aspectos técnicos e comportamentais para sua execução. Entre os primeiros, citamos o conhecimento da arquitetura adotada pela PROCERGS, garantido pela certificação formal no ambiente de todos os componentes da equipe, o pleno domínio tecnológico e a definição de um ciclo de construção.

O ciclo adotado considerou que o modelo de fábrica fosse exercido de forma contínua e com atividades em sequência apropriada, desde a validação das especificações recebidas, codificação em camadas, testes unitários e funcionais até a entrega, com o processo de integração no ambiente do cliente. Ainda deve-se ressaltar a gestão de testes, com validação e complementação dos planos de testes e execução de testes (unitários, de integração e de sistema).

Na gerência de configuração, os itens que colaboraram expressivamente foram a padronização de equipamentos e *softwares* e o versionamento dos artefatos. Nos aspectos comportamentais, destacamos a motivação e comprometimento da equipe, a sinergia estabelecida entre as empresas e os colaboradores envolvidos e o ótimo ambiente de trabalho durante o desenvolvimento do projeto.

Por fim, o Processo de Garantia de Qualidade de *Software* teve por objetivo, verificar e facilitar a aderência à metodologia adotada. Para tanto, foram realizadas auditorias de processo e produto, e seus resultados foram repassados à equipe, visando sua melhoria contínua.

Para dimensionamento das atividades e resultados atingidos, a experiência aqui relatada constituiu-se de um processo de 4 meses de trabalho, com uma equipe formada por 6 profissionais, desempenhando 9 papéis (gerente de projeto, líder técnico, integrador, gestor de testes, analista de testes, construtor, testador, auditor de processo e produto) com 3 entregas realizadas para o cliente, em um total de 268 UCPs construídas. A quantidade de erros reportada aponta a média de 1 erro para cada 11,16 UCPs, o que pode ser considerada uma média que a alto índice de qualidade obtido pelo projeto.

5. Resultados obtidos

A avaliação dos resultados obtidos até o momento é positiva, tanto em termos de qualidade dos produtos gerados, como no aperfeiçoamento do próprio modelo, além de ter demonstrado a viabilidade do desenvolvimento de aplicações utilizando os modernos conceitos de fábrica de *software*, mantendo o domínio dos conhecimentos sobre os sistemas na PROCERGS.

6. Conclusão

Para que a capacidade produtiva de desenvolvimento de *software* do Estado possa acompanhar o aumento das demandas de TIC, com garantia de qualidade dos produtos gerados, tornam-se necessárias algumas mudanças. Em função disso, a REDEPRO propôs inovações nos processos de Aquisição, Fornecimento e Desenvolvimento de *software*, que provaram ser viáveis, atendendo às necessidades apontadas. Além disso, o projeto colabora em aspectos importantes para o Governo, tais como geração de emprego, renda e promoção do desenvolvimento econômico, com total aderência aos eixos da administração estadual (modernização, combate às desigualdades regionais e novo modelo de gestão).

Referências

[ABN 12207], ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO/IEC 12207/1997: Processos de Ciclo de Vida de Software. Rio de Janeiro, 1997.

[FAB 2006], FSRS - Fábrica de Software RS. Projeto disponível em www.fsrs.com.br.

[RED 2006], REDEPRO - Rede de Fornecedores Certificados. Projeto disponível em www.redepro.rs.gov.br.



[DAT 2006], Datum - Datum Informática Ltda. Site disponível em www.datum.inf.br.

[DBS 2006], DBSserver - DBServer Assessoria em Sistemas de Informação Ltda. Site disponível em www.dbserver.com.br.

[SOF 2006], Softdesign - Softdesign Consultoria e Sistemas. Site disponível em www.softdesign-rs.com.br.

[2.47] Rumo ao CMMI-SW Nível 2 para micro, pequenas e médias empresas

Entidade: 1Laboratório de Qualidade e Produtividade de Software (LQPS), Mestrado em Computação Aplicada ; Universidade do Vale do Itajaí (UNIVALI) - Campus São José - Rod. SC 407 km 4, s/n - 88.122-000 - São José/SC e 2Centro de Pesquisas Renato Archer - CenPRA, Rod. Dom Pedro I km 143,6 – 13069-901 Campinas/SP

Autores: Marcello Thiry 1, Christiane Gresse von Wangenheim 1, Alessandra Zoucas 1, Kênia Pickler 1, Clenio F. Salviano 2 - {thiry,gresse,azoucas,kenia}@univali.br, clenio.salviano@cempra.gov.br

Abstract. *This article presents an introduction of a project to developer and application of approach to implementation of the process improvement programs in alignment with maturity level 2 of the CMMI-SE/stage adapted to small software companies. This project was executed by LQPS/UNIVALI with CenPRA and with 5 software companies of ACATE in Florianópolis/SC.*

Resumo. *Este artigo apresenta um resumo de um projeto para desenvolvimento e aplicação de uma abordagem para implementação de programas de melhoria de processo enfocando o nível 2 do CMMI-SW/em estágios para micro, pequenas e médias empresas. Este projeto foi realizado pelo LQPS/UNIVALI em parceria com o CenPRA e com 5 empresas de software da ACATE em Florianópolis/SC.*

1. Introdução

No Brasil, o crescimento do setor de software incentiva à criação de novos e pequenos empreendimentos e a maior parte destes empreendimentos é representada por micro, pequenas e médias empresas (MPMEs), que passam a ter, desta forma, um papel essencial na economia do país. Entretanto, devido à informalidade de seus processos, este tipo de empresa sofre diversas dificuldades em relação à qualidade e produtividade de software e conseqüentemente, acabam prejudicando a qualidade de seus produtos e também a competitividade no mercado. Para uma empresa alcançar a melhoria da qualidade de seu produto final, geralmente, é necessário antes atingir a garantia da qualidade do próprio processo produtivo.

Por este motivo, o Laboratório de Qualidade e Produtividade de Software (LQPS) da UNIVALI em conjunto com o Centro de Pesquisas Renato Archer – CenPRA, e com 5 empresas da ACATE (Associação Catarinense das Empresas de Tecnologia), realizou um projeto de pesquisa para o desenvolvimento e a aplicação de uma abordagem para implementação de programas de melhoria de processo enfocando o nível 2 do CMMI-SW/em estágios [SEI 2006] para MPMEs.

Para a validação da abordagem, ela foi aplicada em 5 empresas da ACATE da Grande Florianópolis. A parceria LQPS/UNIVALI, CenPRA e ACATE viabilizou o desenvolvimento, aplicação e avaliação dos resultados do projeto.

Este projeto corresponde ao projeto de pesquisa “PLATIC: Plataforma de Tecnologia da Informação e Comunicação de Santa Catarina - Meta 1: Rumo ao CMMI-SW Nível 2 para MPMEs” financiado pelo Programa Encomenda CTVerde-Amarela/Ação Regional (FINEP).

2. Objetivos e Justificativa

O presente projeto teve como principal objetivo o desenvolvimento e a aplicação de uma abordagem para implementação de programas de melhoria de processo com ênfase no nível 2 do CMMI-SW/em estágios para MPMEs de software.

A maior parte das empresas brasileiras de software é classificada como MPME.

Identificar áreas problemáticas e sistematicamente estabelecer ações de melhoria são atividades vitais para o sucesso, a longo prazo, de uma empresa. Isso mostra a importância da qualidade e produtividade de software para o aumento da competitividade do setor. Apesar de existirem atualmente vários métodos para a melhoria de processo de software, a maioria é mais adequada para grandes empresas, requerendo alto custo e experiência considerável para a sua adaptação. Assim, a abordagem em desenvolvimento buscou facilitar a aplicação do modelo CMMI-SW para as MPMEs, permitindo um uso mais amplo em uma maior quantidade de empresas nacionais, aumentando sua competitividade no mercado interno e externo.

A aplicação desta abordagem desenvolvida para implementação de programas de melhoria desenvolvida foi realizada alinhada aos objetivos, contexto e estratégia de negócios das 5 empresas da ACATE envolvidas no projeto.

3. Metodologia de Execução

As atividades do presente projeto iniciaram com a busca pela capacitação no modelo CMMI-SW/em estágios (ênfase ao nível 2 de maturidade), por meio de pesquisas bibliográficas e de reuniões periódicas. Também foram realizadas análises para traçar comparações e identificar pontos fortes e fracos, assim como similaridades e pontos de compatibilidade entre os outros modelos e normas existentes na área de melhoria de processo de software. Esta capacitação foi realizada de forma contínua durante a execução do projeto, com o envolvimento de visitas do pesquisador Clenio Salviano do CenPRA.

A partir deste período de estudos e pesquisas inicial, foi realizada a elaboração de um Programa de Melhoria Cooperado para as 5 empresas participantes do projeto.

Este Programa de Melhoria Cooperado foi apresentado, validado e disponibilizado para todas as empresas participantes do projeto.

A aplicação do projeto estabeleceu uma metodologia organizada em 5 fases visando garantir o atendimento de seu objetivo principal.

- **Conscientização.** Esta fase iniciou com uma reunião de kick-off onde as empresas participantes puderam conhecer a proposta do projeto, a equipe do LQPS, bem como a metodologia que a ser aplicada. Ainda como parte da Conscientização, foi realizada uma palestra de 4 horas, abordando o tema de Melhoria de Processo de Software e uma visão inicial do Modelo CMMI, assim como a apresentação do planejamento das aplicações a serem feitas no contexto deste Projeto.
- **Diagnóstico.** Nesta fase, as empresas tiveram seus processos de software analisados em alto-nível por meio de avaliações de contextualização [Gresse Von Wangenheim, Pickler, Thiry, Zoucas e Salviano 2005]. Como resultado, foi possível identificar características relevantes, como metas de negócio, tamanho da equipe de desenvolvimento, estratégias de gerenciamento, execução das áreas de processo referente ao nível 2 de maturidade do CMMI-SW/em estágios, ferramentas, metodologias e técnicas utilizadas. O diagnóstico foi individualmente realizado em cada empresa. A análise foi feita por meio de questionários e entrevistas com responsáveis envolvidos no processo de desenvolvimento de software. Esta fase foi aplicada pelos consultores do LQPS e, na primeira avaliação, houve a participação do professor Clênio Salviano do CenPRA. Como resultado, foram elaborados relatórios para cada empresa.
- **Revisão.** Após o diagnóstico, foi necessário executar uma revisão das áreas de processo que eram de interesse das empresas participantes. Embora cada empresa tivesse características e demandas próprias, o objetivo desta revisão foi identificar uma maneira de alinhar a execução das fases seguintes de forma cooperada. Assim, foi elaborado um plano cooperado de melhoria de processo, que foi apresentado em uma reunião a todas as empresas. A elaboração deste plano contou com a participação ativa das empresas e, embora tenha uma linha comum para a execução do projeto, o plano contou com algumas das características individuais de cada uma das empresas.
- **Treinamento.** Esta fase teve a finalidade de capacitar os consultores e os profissionais das empresas para que eles pudessem identificar e avaliar metodologias de desenvolvimento e manutenção de software adequadas às áreas de processo do nível 2 do modelo. O treinamento foi realizado direcionado para as pessoas responsáveis pelas atividades da melhoria do processo de software na empresa. Esta fase aconteceu em paralelo com a fase de consultoria e ambas seguiram até o final do projeto. Desta forma, as ações de melhoria identificadas durante a fase de consultoria seguiram a sequência de treinamentos. A sequência destes treinamentos também fez parte do plano cooperado e foi um dos resultados da fase de revisão.
- **Consultoria.** A Consultoria teve como objetivo acompanhar as empresas participantes do projeto na implantação e controle de ações de melhoria seguindo as práticas do nível 2 do CMMI-SW/em estágios. A consultoria

foi realizada pela equipe do LQPS, semanalmente para a maior empresa e quinzenalmente para as demais, com 4 horas de duração. Para a primeira atividade realizada nesta fase, a modelagem detalhada dos atuais processos adotados em cada uma das empresas, utilizou-se uma abordagem colaborativa [Thiry, Gresse Von Wangenheim, Zoucas e Pickler 2006]. Após, foi realizada uma avaliação para identificar as diferenças entre a situação atual de cada empresa e as práticas definidas pelo modelo CMMI-SW/em estágios. A seguir, acompanhando a sequência dos treinamentos, as ações de melhoria foram identificadas, implementadas em projetos piloto, acompanhadas e revisadas.

4. Resultados Obtidos

O principal resultado do projeto é abordagem para implementação de programas de melhoria de processo adaptado a MPMEs. A abordagem está sendo detalhadamente documentada e disponibilizada publicamente na internet. A seguir são apresentados os resultados obtidos durante a execução:

- Definição do Programa de Melhoria Cooperado, o qual foi apresentado, validado e disponibilizado para todas as empresas participantes do projeto.
- Desenvolvimento de uma abordagem para avaliação de contextualização adaptada à realidade das MPMEs. O objetivo destas avaliações é conhecer de forma geral a atual situação do processo de software das empresas visando a melhoria alinhada ao nível 2 de maturidade do CMMI-SW/em estágios.

Entretanto, a abordagem pode ser utilizada também com outros modelos como o MPS.BR [SOFTEX 2006] e a norma ISO/IEC 15504 [ISO/IEC 15504 2006].

- Aplicação de avaliações de contextualização nas 5 empresas participantes, na fase de diagnóstico. A compilação destas avaliações, juntamente com a abordagem definida, resultou na publicação de um artigo no Simpósio Internacional de Melhoria de Processo de Software – SIMPROS 2005 [Gresse Von Wangenheim, Pickler, Thiry, Zoucas e Salviano 2005].
- Desenvolvimento de uma abordagem colaborativa e visual para a modelagem do processo de software. Esta abordagem é executada por meio de seções entre os consultores e os profissionais da empresa, onde são utilizados símbolos específicos que facilitam a visualização e o entendimento sobre o processo. A abordagem compreende também seções de validação, onde os participantes revisam o processo definido. Assim que o processo é validado, a empresa participante inicia o detalhamento dos processos utilizando templates específicos.
- Aplicação da modelagem do processo de software nas 5 empresas participantes, na fase de consultoria. Tanto a definição quanto a aplicação desta abordagem colaborativa, resultou na publicação de um artigo no

Simpósio Brasileiro de Qualidade de Software – SBQS 2006 [Thiry, Gresse Von Wangenheim, Zoucas e Pickler 2006].

- Realização de treinamentos com o objetivo de buscar uma maior capacitação tanto dos consultores quanto dos profissionais das empresas participantes. Com os treinamentos foram adquiridos conhecimentos práticos e foram solucionadas dúvidas provenientes das atividades de pesquisa bibliográfica e das consultorias nas empresas. Estes treinamentos foram selecionados com foco nas sete áreas de processo do nível 2 do CMMI-SW/em estágios. Além disso, um treinamento extra sobre metodologia de desenvolvimento de software foi adicionado.
- Realização de atividades de consultoria nas empresas participantes, seguindo o plano de melhoria definido. A consultoria teve como objetivo fazer o acompanhamento da implementação e controle das ações de melhoria, nas empresas, de acordo com as práticas sugeridas pelo nível 2 de maturidade do CMMI-SW/em estágios. Esta implementação foi realizada de acordo a sequência dos treinamentos e respeitando as necessidades, características e limitações de cada uma das empresas.

Além dos resultados pontuais aqui apresentados, este projeto deu suporte para a criação de uma pesquisa, em andamento, para a avaliação da interface entre os diferentes modelos de negócios das empresas de desenvolvimento de software e os requisitos estabelecidos pelo modelo CMMI-SW/em estágios.

E com o objetivo de disponibilizar às empresas participantes do projeto uma base de conhecimento mais direcionada também foi iniciado o desenvolvimento de guias de processo e de ferramentas relacionados às áreas de processo do nível 2 do modelo CMMI-SW/em estágios.

4.1 Artigos Publicados

Durante a realização do projeto foram publicados 2 (dois) artigos em conferências nacionais:

- Aplicando Avaliações de Contextualização em Processos de Software Alinhados ao CMMI-SE/SW - SIMPROS 2005 [Gresse Von Wangenheim, Pickler, Thiry, Zoucas e Salviano 2005].
- Uma Abordagem para a Modelagem Colaborativa de Processos de Software em Micro e Pequenas Empresas - SBQS 2006 [Thiry, Gresse Von Wangenheim, Zoucas e Pickler 2006].

Além destas publicações, como resultado da avaliação de contextualização foram produzidos 5 relatórios detalhados, mostrando o perfil dos processos avaliados e as observações feitas durante a fase de diagnóstico de cada uma das 5 empresas de participantes do projeto. Além disso, foram produzidos, durante a fase de consultoria guias para auxiliar na implantação de cada uma das sete áreas de processo do nível 2 do modelo CMMI-SE/em estágios.

4.2 Recursos Humanos Capacitados

No contexto do projeto, por meio dos treinamentos e por visitas científicas para transferência de tecnologia realizada pelo professor pesquisador Clênio, foram capacitados, os pesquisadores do LQPS/UNIVALI participantes do projeto na área melhoria de processo de software. Também foram capacitados os representantes de cada uma das empresas participantes do projeto. A seguir é apresentada a seqüência dos treinamentos realizados durante a execução do projeto:

Tabela 1: Seqüência de treinamentos realizados no projeto.

Treinamento	Execução
Visão Geral do CMMI	ESICenter/UNISINOS
Planejamento de Projeto	Fundação Vanzolini
Monitoração e Controle de Projeto	Fundação Vanzolini
Gerência de Requisitos	PUC-PR
Introdução ao RUP	LQPS/UNIVALI
Gerência de Configuração	Fundação Vanzolini
Medição e Análise de Projeto	LQPS/UNIVALI
Gerência de Acordo com Fornecedores	Fundação Vanzolini
Garantia da Qualidade de Processo e Produto	Fundação Vanzolini

A equipe do projeto, pelo LQPS/UNIVALI contou com 2 doutores Marcelo Thiry e Christiane Gresse von Wangenheim, , 2 graduadas Kênia Pickler e Alessandra Zoucas, e 1 aluno de graduação Osvaldo Capelari Junior e pelo CenPRA o doutor Clênio F. Salviano.

4.3 Parcerias

Para a criação de um centro de excelência, este projeto foi desenvolvido em cooperação com instituições de pesquisa, associações de empresas e diretamente com MPMEs.

Além da parceria com as 5 empresas da ACATE, o projeto manteve um vínculo com o CenPRA, por meio do professor pesquisador Clênio Salviano, referência nacional na área de melhoria de processo de software. Este vínculo, objetivando a transferência de tecnologia, foi garantido por visitas feitas pelo professor pesquisador durante a execução do projeto. A parceria LQPS/UNIVALI, CenPRA e ACATE viabilizou o desenvolvimento, aplicação e avaliação dos resultados do projeto.

5. Aplicabilidade dos Resultados

Este projeto é extremamente significativo para a melhoria de processo de software no contexto de MPMEs brasileiras de software, fornecendo uma adaptação de abordagens existentes às suas características inerentes. Os resultados deste projeto facilitam uma maior aplicação do modelo CMMI-SE/em estágios para

MPMEs de software no Brasil, contribuindo na melhoria do processo de software e auxiliando o aumento da sua competitividade no mercado nacional e internacional podendo também aumentar as exportações de software. Entretanto, os resultados também podem ser aplicados em outros tipos de empresa de software (nacionais e internacionais), desde que feitas pequenas adaptações. O principal impacto deste projeto é a contribuição para viabilizar que MPMEs sejam capazes de melhorar a qualidade e produtividade por meio da melhoria de processo de software.

As nossas experiências com a aplicação da abordagem em 5 empresas de software da Grande Florianópolis, mostraram uma primeira indicação da aplicabilidade da abordagem principalmente no contexto de MPMEs.

6. Características Inovadoras

Este projeto, junto com outros projetos do CenPRA, LQPS/UNIVALI e outras instituições, representam um avanço na utilização dos modelos de melhoria de processo de software para MPMEs e também representam uma alternativa viável e eficaz para a melhoria, com foco nos objetivos, contexto e estratégia de negócios específicos de cada empresa.

A contribuição científica deste projeto foi o desenvolvimento e a aplicação de uma abordagem para implementação de programas de melhoria de processo enfocando o nível 2 do CMMI-SW/em estágios para MPMEs. Como fruto do projeto, experiências inéditas sobre a aplicação de uma abordagem, de forma cooperada, enfocando o modelo CMMI-SW/em estágios em MPMEs brasileiras foram geradas. Isto iniciou a consolidação de uma base para melhoria da qualidade e produtividade dos processos de MPMEs brasileiras auxiliando no aumento da sua competitividade.

A principal inovação da abordagem desenvolvida é a sua adaptação específica para o contexto das MPMEs brasileiras, facilitando a sua aplicação neste tipo de empresa com um custo baixo. Embora a certificação CMMI possa estar fora do alcance destas empresas, o alinhamento a este modelo permitiu que elas pudessem estar em sintonia com as demandas de qualidade do mercado e também com outros modelos mais acessíveis como, por exemplo, o MPS.BR. Desta forma, a abordagem demonstrou a viabilidade da utilização, por MPMEs de software brasileiras, de um modelo internacionalmente reconhecido. Além disso, a aplicação da abordagem nas 5 empresas participantes do projeto permitiu a coleta de dados e experiências sobre a aplicação do modelo CMMI em MPMEs brasileiras.

7. Conclusão e Perspectivas Futuras

Os resultados deste projeto indicam que o método proposto pode ser adequado para avaliar o processo de software em micro empresas incubadas em alto-nível e com baixo custo considerando as características peculiares deste tipo de empresa.

Como resultados de pesquisa significativos o projeto gerou uma abordagem para avaliações de contextualização voltadas para as MPMEs, uma abordagem colaborativa e visual para a modelagem de processos em MPMEs e uma abordagem para avaliação integrada de processos de software, além de guias

para a implementação das sete áreas de processo do nível 2 do modelo CMMI-SW/em estágios. Somados aos resultados da pesquisa, pode-se também observar benefícios nas empresas participantes, por meio do entendimento e documentação de seus processos, avaliação de sua situação atual em relação ao modelo CMMI-SW/em estágios e ainda com a capacitação de seus profissionais com os treinamentos realizados.

Outro benefício foi a execução das ações de melhoria por parte dos representantes das empresas como acompanhamento desta execução por parte dos consultores do LQPS. Estas ações de melhoria realizadas nas empresas seguindo a ordem de execução dos treinamentos como uma forma de aplicação prática destes treinamentos. As fases concomitantes de treinamento e consultoria seguiram até o final do projeto, permitindo a revisão, melhoria e consolidação das abordagens desenvolvidas. Como indicam os resultados do projeto nas empresas participantes, a aplicação destas abordagens em projetos piloto, pode contribuir para a melhoria nos seus processos de software e também auxiliar para o aumento de sua competitividade das MPMEs no mercado global podendo também aumentar as exportações de software.

Em geral, os resultados do projeto também podem ser aplicados internacionalmente ou em outros tipos de empresa de software com pequenas adaptações.

Com base nestas experiências, a abordagem está sendo evoluída, aplicada e validada em outros projetos incluindo em projetos em organizações de pesquisa. Esta evolução da abordagem inclui também a utilização de outros modelos de melhoria de processo, como por exemplo o MPS.BR.

Referências Bibliográficas

Gresse Von Wangenheim, C., Pickler, K., Thiry, M., Zoucas, A., e Salviano, C. (2005) "Aplicando Avaliações de Contextualização em Processos de Software Alinhados ao CMMI-SE/SW". In: SIMPROS 2005, Simpósio Internacional de Melhoria de Processo de Software, São Paulo, Brasil, 2005.

ISO/IEC Std.15504 (2006): Information Technology – Process Assessment, International Standard, 2003-2006.

SEI (2006)- Software Engineering Institute. Capability Maturity Model Integration (CMMI) Product Suite, 2006. <http://www.sei.cmu.edu/cmmi> SOFTEX (2006). Guia Geral do MPS.BR - Melhoria de Processo do Software Brasileiro (Versão 1.1). Abril, 2006.

Thiry, M., Gresse Von Wangenheim, C., Zoucas, A. e Pickler, K. (2006) "Uma Abordagem para a Modelagem Colaborativa de Processos de Software em Micro e Pequenas Empresas". In: V SBQS 2006, Simpósio Brasileiro de Qualidade de Software, Vila Velha, Brasil, 2006.

Recursos Humanos

[3.02] Processo de Treinamento do nível E do MPS.BR com estratégia de uma empresa do Pólo de Software AmazonSoft.

Entidade: SOLTIN – Soluções Integradas em Tecnologia de Informações Ltda. - Av. Gen. Rodrigo Octávio, 1866, Sede do CIDE, sala: 23, São Lázaro, CEP: 69073-620 Manaus - AM - Brasil

Autor: Paulino Wagner Palheta Viana - wagner_palheta@yahoo.com.br

Objetivos e Justificativas:

A empresa passou por um processo de implementação do nível G do MPS.BR, e com isso, sentimos a necessidade de muitas pesquisas por parte dos integrantes da equipe. E procuramos uma resposta para a seguinte pergunta, como poderíamos formalizar esses treinamentos? Esse projeto teve como principal objetivo prover a empresa com profissionais que possuam conhecimentos e as habilidades para executar suas funções de forma eficiente.

Decidimos implementar o processo de Treinamento localizado no nível E do MPS.BR. Buscamos dentro do próprio modelo o processo para apoiar essa iniciativa.

Metodologia de execução:

A metodologia criada descreve os seguintes passos:

- Descreve atividade e subatividades
- Responsáveis
- Artefatos de entrada
- Artefatos de saída
- Critérios

Resultados relevantes:

A estratégia de Treinamento adotada pela Soltin atende os resultados esperados do Processo de Treinamento do MPS.BR. descritos a seguir:

- Uma estratégia de treinamento (TR1)
- Levantamento de necessidades: (TR2)
 - Questionário de treinamento
 - Solicitação de treinamento
- Planejamento de treinamento: (TR3)
 - Plano de Projeto (PMBok), tempo, riscos, custos, etc.
 - Aprovação do plano (V&V), comprometimento da Gerência ou descontinuidade

- Agendamento do treinamento e comunicação aos participantes
- Execução do treinamento: (TR4)
- Recursos necessários (TR3)
- Registros do treinamento (TR4)
- Certificados de participação (TR4)
- Avaliação do treinamento (TR4)
- Avaliação da eficácia do treinamento (TR5)
 - Avaliação formal, baseado em indicadores de produtividades, retrabalhos.

Aplicabilidade dos resultados e principais impactos:

Obtivemos sucesso com a implementação, pois foram realizados dos cursos para aumentar nossas habilidades

- Formação de Auditor Interno de Sistema de Qualidade
- Planejamento da Qualidade e Mapeamento de Processo

Conclusão e perspectivas futuras:

Com isso acreditamos que consigamos mitigação dos riscos do nível de capacidade dos colaboradores em realizar suas atividades, colabora para o crescimento intelectual do principal ativo da empresa, o colaborador. Sendo assim, surge novas oportunidades a serem pesquisadas, como por exemplo, como gerenciar esse conhecimento adquirido pelos colaboradores e manter na empresa.

[3.03] Um Jogo para Apoiar o Ensino de Gerência de Projetos de Software

Entidade: Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), Faculdade de Informática (FACIN)

Autor: Rafael Prikladnicki, Ricardo Rosa, Eric Kieling

1. Objetivos e Justificativa

Um projeto é definido como um esforço temporário, conduzido por um grupo de pessoas, para criar um produto, serviço ou resultado único. Um projeto é caracterizado por possuir objetivo e requisitos claramente definidos (Escopo), obedecer ao orçamento aprovado (Custo) e deve ser finalizado no prazo especificado (Tempo) (Schwalbe, 2002).

Com o desenvolvimento tecnológico, os projetos começaram a ficar cada vez mais complexos, surgindo a necessidade de criar técnicas e padrões para sua execução. Com isso, vários processos de gerenciamento foram criados, tais como gerenciamento de custos, criação de uma agenda de tarefas, organização das equipes, alocação de recursos, análise de riscos, entre outros. Tudo isso precisa ser observado na criação, elaboração e execução de projetos, independente da

área. Para que um projeto seja bem executado é imprescindível que haja ferramentas apropriadas, técnicas bem desenvolvidas, conhecimento e habilidades dos responsáveis por sua execução. Para isso existe o gerenciamento de projetos.

Hoje em dia, o gerenciamento de projetos de software está muito difundido entre diversos profissionais como diretores, gerentes, clientes e outros interessados. Para auxiliar estes profissionais foi criado um conjunto de conhecimentos em gerenciamento de projetos (PMBOK) que, segundo o PMI (PMI, 2004) é a soma dos conhecimentos intrínsecos à profissão de gerenciamento de projetos. Com base neste corpo de conhecimentos, o gerenciamento de projetos constitui-se da aplicação e integração de processos de nove áreas de conhecimento em cinco fases do ciclo de vida de gerenciamento, quais sejam: iniciação, planejamento, execução, monitoramento e controle, e encerramento.

O gerente de projetos deve seguir estes processos ao longo destas fases, sendo responsável pelos objetivos do projeto e que esses sejam corretamente realizados. Para isso, o gerente deve identificar as necessidades, estabelecer os objetivos de forma clara e que esses sejam alcançáveis, balancear as demandas conflitantes de qualidade, escopo, tempo e custo e adaptar as especificações, dos planos e da abordagem para que tanto a organização como os clientes fiquem satisfeitos. Mas aprender a ser um gerente de projetos não é uma tarefa fácil. Para isso existem vários tipos de treinamentos, e esses, cada vez mais priorizam os profissionais que já possuem um bom embasamento teórico sobre gerenciamento de projetos, assim dificultando o aprendizado daquelas pessoas que estão começando o seu aprendizado na área, tais como alunos e profissionais sem experiência.

Desta forma, existe uma demanda cada vez maior pelo aprendizado de gerenciamento de projetos, e um número cada vez maior de pessoas interessadas em aprender, mas que não possuem experiência nem teórica e nem prática. Foi observado que os treinamentos disponíveis, tanto em cursos isolados como em disciplinas nas faculdades, acabam sendo caracterizados por duas abordagens principais: (1) dão ênfase a conceitos não tão básicos de gerência de projetos, visando àqueles profissionais que pretendem aperfeiçoar seus conhecimentos sobre gerência; e (2) são em grande parte teóricos, sem um entendimento prático dos conceitos. Assim, acredita-se que a abordagem didático-pedagógica pode ser repensada, de forma a abranger também aqueles alunos e profissionais que estão começando a aprender os conceitos sobre gerência de projetos, buscando então uma abordagem mais prática e interativa, para que a falta de maturidade e a inexperiência não comprometam o ensino e a aprendizagem nesta área.

Dentro deste contexto, o objetivo deste artigo é apresentar os resultados do projeto de desenvolvimento do jogo *Planager*, que foi criada visando facilitar o aprendizado caracterizado anteriormente. O projeto foi desenvolvido vinculado a um trabalho de conclusão do Bacharelado em Ciência da Computação da Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS). Além do desenvolvimento da ferramenta, o foco deste trabalho permitiu que o objeto em estudo fosse visto e desenvolvido com maior profundidade pelos alunos envolvidos.

Este trabalho utilizou uma metodologia diferenciada dos treinamentos tradicionais. Esta metodologia envolveu a elaboração de um jogo envolvente, onde o aluno se familiariza mais facilmente com os conceitos sobre gerenciamento de projetos (Gramigma, 1994). Assim, diferentemente dos treinamentos tradicionais, foi desenvolvida uma ferramenta que visa a didática, a interatividade e a possibilidade de praticar os conceitos ao mesmo tempo. Desta forma, o objetivo foi proporcionar um aprendizado mais prático e divertido (Betz, 1996).

Na versão inicial desenvolvida, foram utilizados alguns processos definidos por duas áreas de conhecimentos definidas pelo PMBOK, e dois módulos foram criados: o módulo de tutorial, onde o aluno pode revisar os conceitos de gerência de projetos aprendidos em aula e ter uma visão de como é o jogo; e o módulo jogo, onde o aluno pode praticar seus conhecimentos de uma forma interativa. O objetivo do jogo é fazer com que o jogador passe por várias fases, sendo avaliado ao final de cada uma delas.

Assim, entende-se que a ferramenta desenvolvida supre, pelo menos em parte, a carência de ferramentas específicas na área de ensino de gerência de projetos de software, adotando uma abordagem interativa e com uma abordagem didático-pedagógica inovadora. Dentre as necessidades atendidas pode-se citar: uma ferramenta gratuita, em língua portuguesa, adaptável, extensível, com facilidade para criação de cenários em diferentes níveis de dificuldades e para diferentes jogadores, e que engloba alguns dos principais conceitos existentes no PMBOK.

2. Metodologia de Execução

O principal produto desenvolvido ao longo do projeto foi a ferramenta *Planager*. O projeto foi desenvolvido durante dois semestres do Bacharelado em Ciência da Computação da PUCRS. No primeiro semestre realizou-se uma pesquisa sobre o tema abordado (gerência de projetos), buscando-se também apoio em literatura de jogos e de planejamento didático-pedagógico. As principais referências utilizadas foram áreas de conhecimento de Escopo e Tempo do PMBOK. Os principais resultados desta primeira etapa do projeto foram: o melhor entendimento do problema (necessidade e importância de mudar o enfoque nos treinamentos de gerência de projetos para alunos de graduação e profissionais interessados na área mas ainda sem experiência) e da solução (arquitetura, funcionalidades e uma definição de um jogo interativo com cinco processos de áreas de conhecimento do PMBOK), o que permitiu que o projeto prosseguisse para a etapa de desenvolvimento. O segundo semestre foi utilizado para desenvolver a solução projetada. Ao término deste, chegou-se ao produto final do projeto: uma ferramenta plenamente funcional, desenvolvida com base em alguns processos das áreas de gerência de Tempo e Escopo propostos pelo PMBOK (PMBOK, 2004).

Entendendo a importância de se definir e seguir um processo em projetos de desenvolvimento de software, optou-se por incorporar ao projeto práticas do PMBOK, do *Rational Unified Process* (RUP) e a utilização de diagramas da UML (*Unifying Modeling Language*). Foram desenvolvidos diagramas de casos de uso, de seqüência e de classes, além da elaboração da EAP (Estrutura Analítica do Projeto – do inglês WBS), e do planejamento completo do projeto. Além disso, o

próprio planejamento da ferramenta *Planager* foi utilizado para criar o primeiro cenário do jogo. Acredita-se que o fato de ter-se utilizado o RUP e o PMBOK durante o desenvolvimento deste trabalho de conclusão de curso seja inovador, pois não costuma ser prática comum entre os trabalhos de conclusão definir e utilizar um processo de desenvolvimento de software nem um processo de gerenciamento de projetos da forma como ambos foram utilizados.

3. Resultados Relevantes

O projeto produziu os seguintes resultados:

- **módulos/produtos ou programa de computador resultantes de projeto, disponibilizado para o mercado:** a ferramenta *Planager* e seus manuais de instalação, de usuário e de sistema estão disponíveis e podem ser obtidos no endereço <http://www.inf.pucrs.br/~rafael/Planager/>.

- **métodos e/ou algoritmos desenvolvidos:** para o desenvolvimento da ferramenta, foram desenvolvidos algoritmos para pontuar os jogadores em cada fase, aplicando heurísticas especificadas no trabalho, dependendo do tipo de fase. Além disso, foi elaborado um processo de treinamento de acordo com a abordagem didático-pedagógica proposta.

- **artigos publicados:** um artigo descrevendo a ferramenta desenvolvida está sendo escrito para submissão ao Simpósio Brasileiro de Qualidade de Software em 2007. Outro artigo está sendo escrito com foco na abordagem didático-pedagógica utilizada na ferramenta, visando o uso com alunos de graduação, e será submetido para eventos na área de Informática na Educação. Além disso, outro artigo relatando o uso da ferramenta será escrito para eventos científicos no segundo semestre de 2007 e ao longo de 2008.

- **recursos humanos capacitados:** a ferramenta desenvolvida fez parte do trabalho de conclusão do curso de Bacharelado em Ciência da Computação dos dois últimos autores deste relatório, que se tornaram bacharéis em Ciência da Computação em Janeiro de 2007. Além disso, os alunos utilizaram os conceitos de gerência de projetos aprendidos com o trabalho para aplicar profissionalmente. Ambos os alunos também possuem planos de ingressar em um Programa de Pós-Graduação em um futuro próximo, de forma a aprofundar os conhecimentos adquiridos na área. E por fim, mas não menos importante, a ferramenta desenvolvida foi testada com um pequeno grupo de alunos de forma a avaliar a viabilidade de capacitação futura de outros alunos em disciplinas e cursos na área de gerência de projetos, com foco em cursos de graduação.

- **dissertações e/ou teses geradas:** foi gerado um volume de trabalho de conclusão de curso de graduação em Ciência da Computação (Kieling & Rosa, 2006).

- **eventuais parcerias ou programas de transferência de tecnologia efetuados:** no semestre de conclusão deste projeto, o professor orientador entrou em contato com algumas instituições e pesquisadores na área de gerência de projetos de software, de forma a apresentar o trabalho desenvolvido e buscar parceiros para testar a utilidade da ferramenta *Planager*, discutir melhorias para dar continuidade ao trabalho e torná-la um produto. Desta forma, contatos feitos

com profissionais e pesquisadores nos Estados de Minas Gerais, Paraná, Rio de Janeiro e Rio Grande do Sul resultaram no compartilhamento da ferramenta para avaliação e teste.

4. Aplicabilidade dos resultados e principais impactos na infraestrutura física da instituição, da região, do município e do país.

Do ponto de vista de aplicabilidade dos resultados, a ferramenta desenvolvida foi planejada e está pronta para ser utilizada em disciplinas de graduação que ensinam conceitos básicos de gerência de projetos de software. Mas apesar de ter sido pensada com foco em projetos de desenvolvimento de software, a ferramenta pode ser utilizada para apoio ao ensino de conceitos de gerência de projetos como um todo. Além disso, também pode ser utilizada em cursos de especialização, ou até individualmente por profissionais que necessitam de uma ferramenta interativa com foco em jogo de estratégia, diferente de um ambiente de *e-learning* ou um treinamento presencial com ênfase na teoria e aulas expositivas. Cabe salientar que atualmente o projeto encontra-se na fase de pedido de registro do software, para poder disponibilizar para uso de forma mais abrangente.

5. Características Inovadoras

Do ponto de vista de inovação, a ferramenta incorpora algumas características importantes, entre as quais podemos citar:

- É uma ferramenta gratuita, em língua portuguesa, e adaptável. É possível ser utilizada por organizações interessadas na capacitação de profissionais, bem como para alunos de cursos de graduação e pós-graduação.
- De forma a incorporar os principais conceitos de gerência de escopo e gerência de tempo, a ferramenta foi desenvolvida considerando cinco processos principais: definição do escopo, criação da EAP, definição de atividades, seqüenciamento de atividades e identificação do caminho crítico.

Para todas as fases, o conteúdo deveria ser configurado através da criação de um cenário. A modelagem das classes (Figura 1) foi pensada de forma a respeitar alguns requisitos básicos definidos, entre os quais os principais foram: generalização de cenários, diferentes níveis de usuário, níveis de dificuldade de cenários, pontuação de fase e ranking geral.

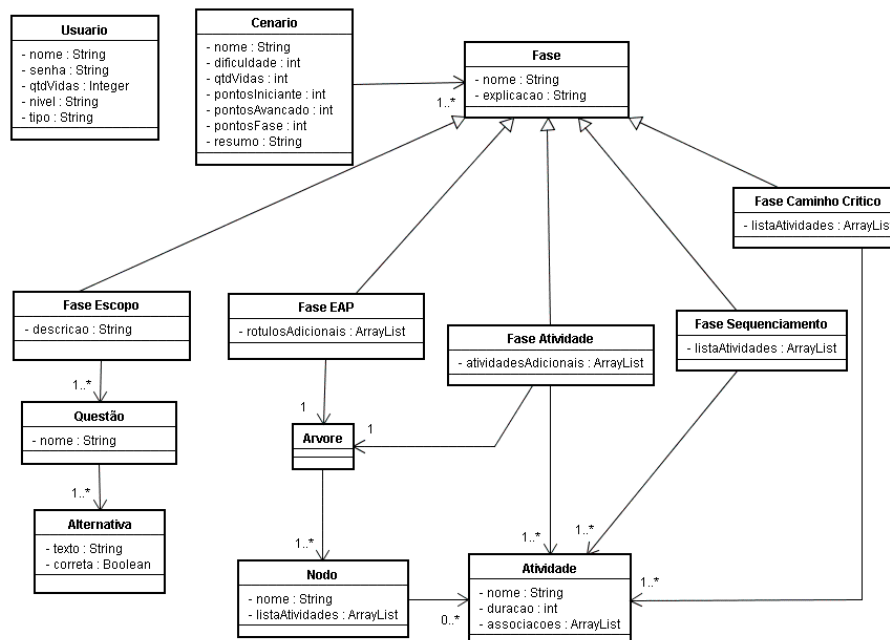


Figura 1 - Diagrama de classes conceitual da ferramenta *Planager*

- a mudança da abordagem didático-pedagógica para o ensino de gerência de projetos de software e a simulação de situações reais através de um jogo também são características inovadoras. Jogos são utilizados para apoiar o ensino de diversas áreas do conhecimento e muitas vezes despertam maior interesse e motivação por parte do aluno (Gramigma, 1994). Os jogos de computador permitem a visualizar e experimentar conceitos abstratos, além de trabalhar a criatividade dos jogadores. Neste projeto trabalhou-se com os conceitos de gerência de projetos de forma visual e intuitiva. Além disso, Betz (1996) afirma que em jogos deste tipo, competição e definição de objetivos são componentes motivadores. O jogo *Planager* possui um sistema de pontuação e um ranking que são utilizados para motivar o jogador a aprender.

- No intuito de abranger usuários avançados e usuários mais experientes, um dos requisitos identificados foi a possibilidade de criar cenários de jogo com diferentes níveis de dificuldade e para diferentes tipos de jogadores, tudo configurado por usuários com permissão de administração. A figura 1 ilustra a fase de criação de EAP para um projeto “churrasco”.

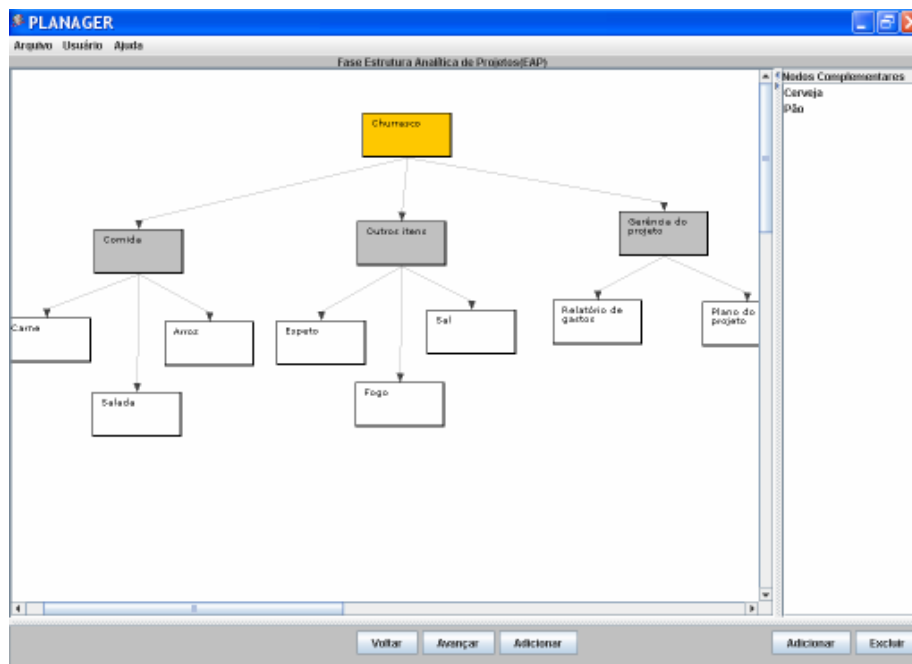


Figura 2 - Criação da EAP no cenário da ferramenta *Planager*

- Como um dos objetivos era permitir o uso sem depender de acesso à internet, tomou-se o cuidado de fazer uso apenas de tecnologias apropriadas para este fim (o jogo foi desenvolvido utilizando a linguagem de programação Java, caracterizando uma ferramenta flexível e portátil, podendo ser utilizada pelas diversas plataformas que suportam a JVM - *Java Virtual Machine*).

6. Conclusão e Perspectivas Futuras

Este projeto foi desenvolvido devido à percepção de que existe uma demanda cada vez maior pelo aprendizado na área de gerência de projetos. Além disso, a grande maioria dos treinamentos existentes não supre a necessidade de buscar o equilíbrio entre a formação teórica e prática do aluno, em uma área que precisa não apenas cursos com foco em certificações profissionais, tais como *Project Management Professional* (PMP). É necessário simular a vivência em um projeto. E quando o público envolvido não possui experiência na área, esta tarefa torna-se mais difícil e cansativa.

A ferramenta *Planager* buscou facilitar o aprendizado de gerência de projetos, principalmente em disciplinas de graduação, buscando tanto apresentar conceitos quanto simular cenários reais. Ela foi desenvolvida na forma de um jogo para que os usuários possam aprender de uma maneira mais interativa do que as existentes atualmente.

Como a ferramenta possui um módulo de criação de cenários, além do jogo propriamente dito, os professores podem encorajar os alunos mais experientes a participar do jogo criando novos cenários, pois isto ajuda a aprimorar os conceitos.

Mas é importante dizer que, para criar cenários, o usuário deve possuir bons conhecimentos em planejamento de projetos e deve conseguir definir muito bem o escopo do cenário.

Entre as limitações do projeto, esta versão abrangeu apenas alguns dos processos de planejamento das áreas de gerência de tempo e gerência de escopo. Por outro lado, ela foi especificada de forma a ser facilmente expansível para incluir outros processos de outras áreas de conhecimento, através de novos módulos.

A principal dificuldade encontrada durante o desenvolvimento foi a subjetividade existente nos processos de gerência de projetos. É muito difícil desenvolver um software que seja capaz de avaliar atividades tão subjetivas como a criação de uma EAP ou a definição das atividades que estão no escopo de um projeto. Por este motivo, é importante ressaltar a necessidade de descrever bem o escopo do projeto e o cenário como um todo.

Devido a limitações de tempo, não foi possível desenvolver a interface gráfica de acordo com os padrões utilizados em jogos convencionais. Planos futuros incluem continuar o desenvolvimento da ferramenta, e utilizá-la como apoio nas disciplinas de gerência de projetos de software dos cursos de graduação da Faculdade de Informática da PUCRS.

7. Referências

Betz, J. A. "Computer Games: Increases learning in an interactive multidisciplinary environment" - Journal of Educational Technology Systems, 1996.

Gramigma, M.R.M. "*Jogos de empresa*". São Paulo: Makron Books, 1994.

Kieling, E., Rosa, R. "Planager - Um Jogo para apoio ao Ensino de Conceitos de Gerência de Projetos de Software". Relatório Final da Disciplina de Trabalho de Conclusão do Curso de Bacharelado em Ciência da Computação, FACIN, PUCRS, 2006.

PMBOK. "A guide to the project management body of knowledge". PMI Publishing Division, 2004.

Schwalbe, K. "*Information Technology. Project Management*", Cambridge, MA: Course Technology, 2002.

Serviços Tecnológicos

[4.01] Avaliação do Modelo MPS em Empresas em 2005 e 2006

Entidade: SOFTEX - Associação para Promoção da Excelência do Software Brasileiro

Autores: Kival Chaves Weber e Eratóstenes Araújo - {kival.weber, era}@nac.softex.br}

1 Introdução

Um estudo do MIT mostrou em 2003 que, nos últimos anos, as empresas de software no Brasil favoreceram a ISO 9000 em detrimento de outras normas e modelos especificamente voltadas para a melhoria de processos de software [Veloso 2003].

Na análise deste problema, constatou-se que um dos maiores entraves à adoção de modelos como o CMM® no Brasil era o alto custo destas avaliações, principalmente para as pequenas e médias empresas (PME) que não podiam pagá-los embora necessitassem melhorar seus processos de software para aumentar a competitividade.

Além disto, na época, o seguinte fato novo decorrente do advento da Norma Internacional ISO/IEC 15504 configurava uma oportunidade: 'As organizações e associações industriais que percebem uma vantagem competitiva estratégica nos seus processos de negócio, agora tem a opção de criar o seu próprio Modelo de Referência de Processo (*Process Reference Model - PRM*) ou usar um modelo de processo existente. Daí, estas também podem criar um Modelo de Avaliação de Processo (*Process Assessment Model - PAM*) baseado neste modelo de referência. Entretanto, o esforço necessário para assegurar que os modelos estão de acordo ou em conformidade com a ISO/IEC 15504 não pode ser subestimado. É provável que apenas grandes organizações possam assumir um empreendimento desta magnitude. Na Europa, as indústrias espacial e automotiva estão seguindo este caminho através de suas associações industriais' [van Loon 2004].

Para ajudar na solução deste problema, a SOFTEX – Associação para Promoção da Excelência do Software Brasileiro lançou o Projeto MPS.BR – Melhoria de Processo do Software Brasileiro no dia 11 de dezembro de 2003, numa reunião realizada no MCT – Ministério da Ciência e Tecnologia, em Brasília-DF.

A SOFTEX é uma entidade privada, sem fins lucrativos, que promove ações com abrangência nacional visando transformar o Brasil em um centro de excelência mundial na produção e exportação de software. Dispõe de 31 Agentes SOFTEX, localizados em 23 cidades de 13 Unidades da Federação, com mais de mil empresas associadas – com o seguinte porte segundo o número de funcionários: 11% grandes, 13% médias, 36% pequenas e 40% micro empresas. Para mais informação, veja o Portal SOFTEX em www.softex.br.

O propósito do Projeto MPS.BR é a Melhoria de Processo do Software Brasileiro, compreendendo dois processos: i) desenvolvimento e aprimoramento do Modelo MPS, baseado nas melhores práticas de Engenharia de Software, em conformidade com as normas ISO/IEC 12207 – Processos do Ciclo de Vida do Software e ISO/IEC 15504 – Avaliação de Processo, compatível com o CMMI – *Capability Maturity Model Integration* [Chrissis 2003, Bush 2005] e adequado à realidade das empresas brasileiras; ii) disseminação e adoção do Modelo MPS, a um custo acessível, tanto em grandes organizações públicas e privadas como em pequenas e médias empresas (PME), em todas as regiões do país.

No período Dez2003-Dez2006, o Projeto MPS.BR contou com as seguintes fontes de recursos financeiros: apoio do BID, FINEP e MCT, complementado por receitas de serviços MPS tais como: taxas de inscrição em cursos, provas e workshops;⁶ taxas de análise de pleitos de Instituições Implementadoras – II, Instituições Avaliadoras – IA, Instituições Organizadoras de Grupos de Empresas – IOGE e Consultores de Aquisição – CA; taxas SOFTEX para avaliação MA-MPS em empresas.

Este artigo descreve as avaliações MA-MPS realizadas no Brasil, tendo como finalidade complementar o relatório final do Projeto [4.01] – Avaliação do Modelo MPS em Empresas em 2005 e 2006, do PBQP Software – ciclo 2006. A seção 2 apresenta o objetivo e a justificativa do projeto. A seção 3 explica como o projeto foi desenvolvido.

A seção 4 destaca os principais resultados obtidos. A seção 5 trata da aplicabilidade destes resultados. A seção 6 resume as características inovadoras do projeto. A seção 7 conclui o artigo e apresenta as perspectivas futuras do projeto.

2 Objetivo e Justificativa

No Projeto [4.01] – Avaliação do Modelo MPS em Empresas em 2005 e 2006, do PBQP Software - ciclo 2006, consta o seguinte:

- Objetivo: Avaliação do Modelo MPS em empresas a um custo acessível, em 2005 e 2006, no âmbito do Projeto MPS.BR - Melhoria de Processo do Software Brasileiro.
- Justificativa: Realizar avaliações oficiais do Modelo MPS em empresas e organizações em que o modelo foi implantado anteriormente.

3 Metodologia de Execução

No âmbito do Projeto MPS.BR, tecnicamente, foi desenvolvido e vem sendo aprimorado anualmente o Modelo MPS, composto do Modelo de Referência (MRMPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS), documentado em quatro Guias do MPS.BR: i) Guia Geral; ii) Guia de Implementação; iii) Guia de Avaliação; iv) Guia de Aquisição.

O Método de Avaliação (MA-MPS) está descrito no Guia de Avaliação. Este guia é composto basicamente pelo processo de avaliação MPS, método de avaliação MPS e características da qualificação dos avaliadores. Este método permite a realização de avaliações segundo o Modelo MPS.

O MA-MPS foi definido em conformidade com os requisitos para modelos de referência de processo (*Process Reference Model - PRM*) e métodos de avaliação de processos (*Process Assessment Model - PAM*) estabelecidos na norma ISO/IEC 15504-2 e atende aos requisitos específicos do Projeto MPS.BR. Desta forma, o método está em conformidade com a ISO/IEC 15504 e é compatível com

⁶ ® CMM, CMMI e SCAMPI são marcas da CMU – Carnegie Mellon University. MPS.BR, MR-MPS, MA-MPS e MN-MPS são marcas da SOFTEX.

o método SCAMPI [Bush 2005] para avaliação segundo o modelo CMMI, também definido com base na ISO/IEC 15504.

O processo de avaliação é composto por seis fases: Planejar e Preparar Avaliação; Realizar Avaliação Inicial e Rever o Plano; Realizar Acertos; Conduzir Avaliação; Relatar Resultados; Registrar Resultados. A Figura 1 apresenta o fluxo destas fases e os principais produtos de entrada e saída do processo de avaliação. Uma avaliação MAMPS tem validade de três anos a contar da data da sua conclusão.

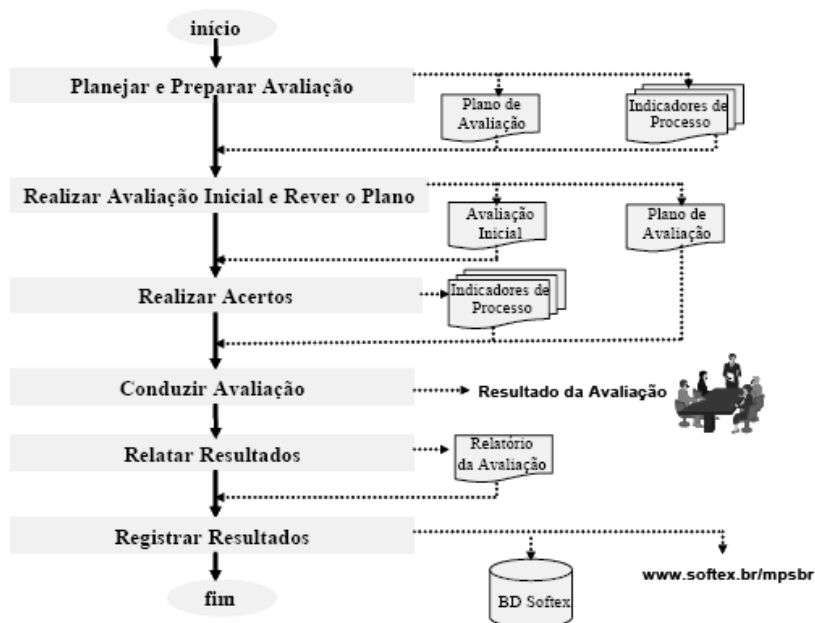


Figura 1. Processo de Avaliação MPS

Em paralelo com o desenvolvimento do Modelo MPS, foram realizadas atividades do Projeto MPS.BR tais como:

- Capacitação de Implementadores e Avaliadores do Modelo MPS (cursos, provas e workshops do MPS.BR)
- Credenciamento de Instituições Implementadoras – II
- Implementação do Modelo MPS em Empresas
- Credenciamento de Instituições Avaliadoras – IA
- Avaliações do Modelo MPS em Empresas

Para isto, o Projeto MPS.BR conta com duas estruturas de apoio: o Fórum de Credenciamento e Controle - FCC e a Equipe Técnica do Modelo - ETM. Através destas estruturas, o MPS.BR obtém a participação de representantes de Universidades, Instituições Governamentais, Centros de Pesquisa e de

organizações privadas, os quais contribuem com suas visões complementares que agregam qualidade ao empreendimento.

O FCC tem como principais objetivos assegurar que as Instituições Implementadoras - II e Instituições Avaliadoras - IA sejam submetidas a um processo adequado de credenciamento e que suas atuações não se afastem dos limites éticos e de qualidade esperados, além de avaliar e atuar sobre o controle dos resultados obtidos pelo MPS.BR.

Por outro lado, cabe à ETM atuar sobre os aspectos técnicos relacionados ao Modelo de Referência (MR-MPS) e Método de Avaliação (MA-MPS), tais como a concepção e evolução do modelo, elaboração e atualização dos Guias do MPS.BR, preparação de material e definição da forma de treinamento (cursos oficiais e workshops de reciclagem) e de aplicação de provas individuais, publicação de Relatórios Técnicos e interação com a comunidade visando à identificação e aplicação de melhores práticas.

4 Resultados Obtidos

Para fins do Projeto [4.01] do PBQP Software – ciclo 2006, são produtos e resultados relevantes:

- 1) Pedidos de patente e/ou patentes registradas Registro da logomarca MPS.BR. Registro das marcas MPS.BR, MR-MPS, MAMPS e MN-MPS. Registro de direitos autorais (*copyright*) dos Guias e Cursos do MPS.BR.
- 2) Produtos de software gerados Lançamento do Portal SOFTEX www.softex.br/mpsbr, sempre atualizado, com informações do MPS.BR disponíveis a todos os interessados. Além disto: i) está sendo “customizado” o ambiente CoreKM, para aprimorar o controle das avaliações, avaliadores, IA, implementadores, II, grupos de empresas e IOGE; ii) está sendo desenvolvido um Repositório MPS contendo resultados de desempenho de empresas que adotaram o Modelo MPS.
- 3) Outros produtos gerados Até Dez2006, foram credenciadas 15 Instituições Implementadoras – II, em todas as regiões do país exceto na Região NO, mediante convênio assinado com a SOFTEX.

Até Dez2008, espera-se exceder muito a meta original: 20 II, em todas as regiões do país.

Até Dez2006, as seguintes Instituições Avaliadoras - IA demonstraram capacidade para avaliar o MPS.BR em organizações que implementaram o Modelo de Referência MR-MPS, seguindo o Método de Avaliação MA-MPS, conforme o COMUNICADO SOFTEX MPS 10/2006 e convênio assinado com a SOFTEX:⁷

Até Dez2008, espera-se exceder a meta original: 15 (quinze) IA, em todas as regiões do país.

⁷ 1) COPPE/UFRJ - FUNDAÇÃO COPPETEC, do Rio de Janeiro-RJ (validade até 29 de novembro de 2008)
2) SWQuality Consultoria e Sistemas, de Recife-PE (validade até 30 de outubro de 2008)

4) Métodos e/ou algoritmos desenvolvidos Desenvolvimento do Modelo MPS, composto de: Modelo de Referência (MRMPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS), documentado em quatro Guias do MPS.BR (Guia Geral, Guia de Implementação, Guia de Avaliação e Guia de Aquisição). Estes guias, criados e aprimorados anualmente pela Equipe Técnica do Modelo - ETM, com base nas melhores práticas da Engenharia de Software e em normas e modelos internacionais, estão disponíveis na seção Guias do Portal www.softex.br/mpsbr.

5) Artigos publicados Descrições mais completas do Projeto MPS.BR e Modelo MPS encontram-se em artigos publicados em renomados eventos no país e no exterior [Weber 2004a, 2004b, 2005a, 2005b e 2006], disponíveis na seção Artigos e Apresentações do Portal www.softex.br/mpsbr. Como se vê nas Referências Bibliográficas, um destes artigos foi premiado como o melhor Artigo Técnico do IV Simpósio Brasileiro de Qualidade de Software (SBQS 2005).

6) Recursos humanos capacitados Capacitação de especialistas no MPS.BR, através de cursos oficiais, provas individuais e workshops de reciclagem.

Anualmente, há oferta regular dos seguintes cursos oficiais do MPS.BR, em todas as regiões do país: C1 – Introdução; C2 – Implementadores; C3 – Avaliadores; C4 – Melhoria do Processo de Aquisição de Software. Há 30 instrutores treinados, autorizados a ministrar os cursos oficiais do MPS.BR. Mais de 2.100 pessoas participaram destes Cursos do MPS.BR, pagando taxas de inscrição acessíveis; das quais, 804 participaram de cursos C2 – Implementadores e 67 de cursos C3 – Avaliadores. Os representantes das empresas nas respectivas equipes de avaliação MA-MPS devem ter participado previamente de um curso C1 – Introdução ao MPS.BR.

Semestralmente, em todo o país, são aplicadas as seguintes provas individuais do MPS.BR em todas as regiões do país: P1 – Introdução, P2 – Implementadores, P3 – Avaliadores; P4 – Melhoria do Processo de Aquisição de Software. As questões das provas aplicadas estão disponíveis na seção Cursos-Provas do Portal www.softex.br/mpsbr. Mais de 600 pessoas foram aprovadas em provas individuais do MPS.BR; das quais, 235 aprovadas em provas P2 – Implementadores e 28 em provas P3 – Avaliadores.

Anualmente, são realizados workshops de reciclagem do MPS.BR: até agora, dois workshops W2 – Implementadores com 160 participantes e um workshop W3 – Avaliadores com 40 participantes.

Além disto, já foram certificados 2 Consultores de Aquisição (CA) de software e serviços correlatos.

7) Dissertações e/ou teses geradas Embora não haja acompanhamento direto disto no Projeto MPS.BR, sabe-se que há trabalhos de fim de curso de especialização, dissertações de mestrado e teses de doutorado, geradas ou em andamento, fortemente correlacionadas com a implementação e avaliação do Modelo MPS, principalmente em Universidades que têm participação ativa na Equipe Técnica do Modelo – ETM, tais como COPPE/UFRJ, UCB, Universidade de Lavras e UFRPE.

8) Lançamento de publicação Lições aprendidas com a implementação e avaliação do Modelo MPS em empresas foram publicadas em duas edições especiais MPS.BR da Revista ProQualiti: volume 1, número 2, Nov2005; volume 2, número 2, Nov2006.

9) Implementação do Modelo MPS em empresas Até Dez2006, implementação do Modelo MPS em cerca de 120 empresas, em todas as regiões do país. Dentre estas, coordenadas por Instituições Organizadoras de Grupos de Empresas – IOGE, 93 empresas (52% pequenas, 33% médias e 15% grandes empresas) estão usando o Modelo de Negócio cooperado, conforme o COMUNICADO SOFTEX MPS 20/2005. Estes grupos de empresas estão comprometidos a concluir a implementação do MR-MPS em até 12 meses, a contar da data de assinatura do convênio entre a SOFTEX e a respectiva IOGE, e a se submeterem a avaliação MA-MPS nos três meses subsequentes. Na seção Implementações do Portal www.softex.br/mpsbr estão publicadas as implementações em grupos de empresas ora em curso.

Até Dez2008, espera-se implementar o Modelo MPS em mais 160 empresas.

10) Avaliação do Modelo MPS em empresas Até agora, foram realizadas 17 avaliações oficiais MA-MPS. Dentre estas, 34% são micro e pequenas empresas, 6% médias empresas e 60% grandes empresas; sendo 18% no nível A (o mais alto), 18% níveis D e E, e 64% níveis F e G. Na seção Avaliações do Portal www.softex.br/mpsbr estão publicados os resultados oficiais das seguintes avaliações MA-MPS.⁸

Um caso interessante de evolução da maturidade dos processos de software é o da empresa BL Informática que: i) em Set2005, após implementar o MR-MPS como integrante de um grupo de empresas (Modelo de Negócio Cooperado - MNC), foi bem sucedida em uma avaliação MA-MPS de nível F - Gerenciado; ii) até Dez2007, isoladamente (Modelo de Negócio Específico - MNE), espera ter sucesso em uma avaliação MA-MPS de nível A – Em Otimização (o mais alto).

⁸

- 1) BL INFORMÁTICA, de Niterói-RJ - Nível F do MPS.BR (validade até 22 setembro 2008)
- 2) BRÁSILIA INFORMÁTICA, de Brasília-DF - Nível G do MPS.BR (validade até 16 novembro 2009)
- 3) BRQ, do Rio de Janeiro-RJ - Nível A do MPS.BR (validade até 8 fevereiro 2010)
- 4) CCA SJ, de São José dos Campos-SP - Nível E do MPS.BR (validade até 4 outubro 2009)
- 5) COMPERA, de Campinas-SP - Nível F do MPS.BR (validade até 20 outubro 2008)
- 6) DATA TRAFFIC, de Goiânia-GO - Nível G do MPS.BR (validade até 14 novembro 2009)
- 7) DBA, do Rio de Janeiro-RJ - Nível A do MPS.BR (validade até 17 de dezembro de 2009)
- 8) FORTES, de Fortaleza-CE - Nível G do MPS.BR (validade até 7 dezembro 2009)
- 9) IN FORMA, de Recife-PE - Nível G do MPS.BR (validade até 12 setembro 2008)
- 10) IVIA, de Fortaleza-CE - Nível G do MPS.BR (validade até 5 dezembro 2009)
- 11) LINKNET DOTNET, de Brasília-DF - Nível G do MPS.BR (validade até 15 março 2009)
- 12) MARLIN, do Rio de Janeiro-RJ - Nível D do MPS.BR (validade até 13 julho 2009)
- 13) POLITEC, de Brasília-DF - Nível A do MPS.BR (validade até 25 maio 2009)
- 14) PROCENGE, de Recife-PE - Nível G do MPS.BR (validade até 20 dezembro 2009)
- 15) PROGRAMMER'S, de Campinas-SP - Nível F do MPS.BR (validade até 24 novembro 2008)
- 16) RELACIONAL, do Rio de Janeiro-RJ - Nível E do MPS.BR (validade até 28 setembro 2008)
- 17) 7COMm, de São Paulo-SP - Nível E do MPS.BR (validade até 14 fevereiro 2010)

Uma das motivações apontadas por três empresas (BRQ, DBA e Politec), bem sucedidas em avaliações MA-MPS de nível A – Em Otimização (o mais alto), foi o fato do mercado (nas aquisições de software e serviços correlatos) começar a pontuar o Modelo MPS nas mesmas condições de modelos equivalentes - como o CMMI.

Até Dez2007, espera-se ter um total de 80 empresas com avaliação oficial MAMPS; até Dez2008, espera-se totalizar 160 empresas avaliadas oficialmente.

5 Aplicabilidade dos Resultados

Para fins do Projeto [4.01] do PBQP Software – ciclo 2006, consideram-se os seguintes aspectos na aplicabilidade dos resultados do Projeto MPS.BR: relevância, abrangência e impacto dos resultados obtidos.

5.1 Relevância

Avaliação de processo segundo o Método de Avaliação (MA-MPS), desenvolvido no Projeto MPS.BR - um empreendimento magno no setor de software Brasileiro, com forte interação Universidade/Empresas/Governo, implicando em mudança cultural e econômica significativa.

5.2 Abrangência

O Projeto é multiinstitucional e tem abrangência nacional, o que também vale para as avaliações do Modelo MPS em empresas em 2005 e 2006.

5.3 Impacto dos resultados obtidos

Fechar um ciclo do Projeto MPS.BR, realizando avaliações segundo o MA-MPS em empresas e organizações em que foi implementado o Modelo de Referência (MR-MPS), ambos desenvolvidos no âmbito do Projeto MPS.BR.

6 Características Inovadoras

O Projeto MPS.BR criou um novo Modelo de Referência de Processo (MR-MPS) e um novo Método de Avaliação de Processo (MA-MPS) - documentados em quatro guias:

Guia Geral, Guia de Implementação, Guia de Avaliação e Guia de Aquisição, em conformidade com normas internacionais (ISO/IEC 12207 e 15504), compatível com modelos renomados (como o CMMI) e adequado à realidade das empresas brasileiras.

7 Conclusão e Perspectivas Futuras

No período inicial do Projeto MPS.BR (Dez2003-Dez2006, praticamente três anos), foram alcançados dois grandes resultados: i) tecnicamente, foram criados e vêm sendo aprimorados anualmente um Modelo de Referência (MR-MPS) e um Método de Avaliação (MA-MPS), o que não é algo trivial em qualquer lugar do mundo; ii) do ponto de vista do mercado, fazendo uso do Modelo de Negócio (MN-MPS), houve melhoria dos processos de software com a implementação do MR-MPS em cerca de 120 empresas em todas as regiões do país, das quais 17 unidades organizacionais foram bem sucedidas em avaliações MA-MPS.

Este artigo tratou de implementações e avaliações do Modelo MPS realizadas em empresas no Brasil em 2005 e 2006, tendo como finalidade complementar o relatório final do Projeto [4.01] – Avaliação do Modelo MPS em Empresas em 2005 e 2006, do PBQP Software - ciclo 2006.

Pode-se concluir que a adoção do Modelo MPS está acelerando no Brasil, com destaque para: i) a capacitação de pessoas através de cursos, provas e workshops do MPS.BR, em especial de Implementadores (C2, P2 e W2) e Avaliadores (C3, P3 e W3), mediante taxas de inscrição acessíveis; ii) as implementações do MR-MPS em todas as regiões do país, especialmente em grupos de pequenas e médias empresas (PME); iii) as avaliações MA-MPS em um número crescente de organizações, a um custo acessível, seja nos níveis mais baixos (G e F) em grupos de PME seja nos níveis mais altos (E até A) isoladamente em grandes organizações públicas e privadas.

Nos próximos três anos (Jan2007-Dez2009), a perspectiva é de: i) consolidação do Projeto MPS.BR, garantindo sua sustentabilidade institucional, operacional e financeira; ii) aprimoramento do Modelo MPS, especialmente do Modelo de Referência (MRMPS) e do Método de Avaliação (MA-MPS) - em conformidade com a nova ISO/IEC 12207 (prevista para Dez2007) e mantendo compatibilidade com o CMMI-DEV; iii) crescimento acelerado das avaliações MA-MPS em todos os níveis do MR-MPS, em todas as regiões do país; iv) disponibilidade de dados sobre resultados de desempenho de organizações que adotaram o Modelo MPS, em categorias tais como: custo, prazo, produtividade, qualidade, satisfação do cliente e retorno do investimento (ROI), com destaque para o índice de satisfação das empresas que implementaram o MPS.BR, bem como o incremento de suas exportações e redução de custos.

8 Referências Bibliográficas

Bush, M., Dunaway, D. CMMI Assessments: Motivating Positive Change. Addison-Wesley, 2005.

Chrissis, M. B., Konrad, M., Shrum, S. CMMI: Guidelines for Process Integration and Product Improvement. Addison-Wesley, 2003.

ISO/IEC 12207:1995/Amd 1:1998/Amd 2:2002. Technology Information – Software Life Cycle Processes.

ISO/IEC 15504. Technology Information – Process Assessment. Part 1 – Concepts and vocabulary; part 2 – Performing an assessment; part 3 – Guidance on performing an assessment; part 4 – Guidance on use for process improvement and process capability determination; and part 5 – An exemplar process assessment model.

van Loon, H. Process Assessment and ISO/IEC 15504: a reference book. The Kluwer International Series in Engineering and Computer Science, Volume 775. Springer, 2004. p. 85.

Veloso, F., Botelho, A. J., Tschang, A., Amsden, A. Slicing the Knowledge-based Economy in Brazil, China and India: a tale of 3 software industries. Report, MIT – Massachusetts Institute of Technology, Mass, September 2003.

Weber, K. C., Rocha, A. R., Alves, A., Ayala, A. M., Gonçalves, A., Paret, P., Salviano, C., Machado, C. F., Scalet, D., Petit, D., Araújo, E., Barroso, M. G., Oliveira, K., Oliveira, L. C. A., Amaral, M. P., Campelo, R. E. C., Maciel, T. Modelo de Referência para Melhoria de Processo de Software: uma abordagem brasileira, In: Proceedings of the XXX Conferencia Latinoamericana de Informatica (CLEI 2004). Arequipa, Peru: septiembre 2004.

Weber, K. C., Rocha, A. R., Rouiller, A. C., Crespo, A., Alves, A., Ayala, A., Gonçalves, A., Paret, B., Vargas, C., Salviano, C., Machado, C., Scalet, D., Petit, D., Araújo, E., Maldonado, J. C., Oliveira, K., Oliveira, L. C., Girão, M., Amaral, M., Campelo, R., and Maciel, T. Uma Estratégia para Melhoria de Processo de Software nas Empresas Brasileiras, In: Proceedings of QUATIC'2004 (5th Conference for Quality in Information and Communications Technology), pp. 73-78. Porto, Portugal, October 2004.

Weber, K. C., Araújo, E., Machado, C. F. M., Scalet, D., Salviano, C. F., Rocha, A. R. C. Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software – versão 1.0 (MR-MPS e MA-MPS), In: Anais do IV Simpósio Brasileiro de Qualidade de Software (SBQS 2005). Porto Alegre, Brasil: junho de 2005. Prêmio de melhor Artigo Técnico do SBQS 2005.

Weber, K. C., Araújo, E. R., Rocha, A. R.C., Machado, C. F., Scalet, D., and Salviano, C. F. Brazilian Software Process Reference Model and Assessment Method, In: P. Yolum et al. (Eds.), Proceedings of ISCIS 2005 (20th International Symposium on Computer and Information Sciences), LNCS 3733, pp. 402-411. Istanbul, Turkey, October 2005. Copyright Springer-Verlag, Berlin Heidelberg, 2005.

Weber, K.C., Araújo, E. R., Rocha, A.R.C., Oliveira, K. M., Rouiller, A.C., von Wangenheim, C. G., Araújo, R., Salviano, C.F., Machado, C.F., Scalet, D., Galarraga, O., Amaral, M.P., e Yoshida, D. Melhoria de Processo do Software Brasileiro (MPS.BR): um programa mobilizador, In: Proceedings of the XXXI Conferencia Latinoamericana de Informatica (CLEI 2006). Santiago, Chile: agosto 2006.

[4.02] Alinhando a Seleção de Processos de Software ao Modelo de Negócio de uma MPE

Entidade: ¹Laboratório de Qualidade e Produtividade de Software (LQPS), Mestrado em Computação Aplicada, Universidade do Vale do Itajaí (UNIVALI) - Campus São José; Rod. SC 407 km 4, s/n - 88.122-000 - São José - SC - Brasil

²Centro de Pesquisas Renato Archer - CenPRA, Rod. Dom Pedro I km 143,6 - 13069-901 Campinas - SP - Brasil

Autores: *Marcello Thiry*¹, *Christiane Gresse von Wangenheim*¹, *Eliza Coral*¹, *Clênio Salviano*², *Anderson Santos*¹, *Lúcio Frantz*¹ - {thiry,gresse,eliza}@univali.br, clenio.salviano@cenpra.gov.br, {andersonrs,luciofrantz}@univali.br

Abstract. *This paper presents an approach to business-driven guidance for defining target process profiles in Small Software Organizations. We analyzed the*



different business models adopted by the organizations and with base on the models characteristics we identified the related process areas from CMMI continuous model and ISO/IEC 15504. The presented results are based on our experiences in assessing and implementing software process improvement programs in several small software organizations in Brazil.

Resumo. *Este artigo apresenta uma abordagem para orientar a definição de perfis alvo de processo em micro e pequenas empresas de software a partir dos modelos e objetivos de negócio destas organizações. Foram avaliados os modelos de negócio utilizados pelas organizações e a partir das características destes modelos, foram relacionados com as áreas de processo dos modelos CMMI contínuo e ISO/IEC 15504. Os resultados apresentados estão baseados nas experiências que os autores possuem na condução de avaliações e implementações de programas de melhoria de processos em diversas micro e pequenas organizações de software no Brasil.*

1 - Introdução

Em vários países, a maior parte das organizações de desenvolvimento de software é de porte pequeno, com menos de 50 empregados. No Brasil, o MCT (2005) classifica como sendo uma micro empresa, aquela organização que possui até 9 funcionários e como pequenas empresas aquelas com até 49 funcionários. Ainda de acordo com o MCT (2005), as micro e pequenas empresas (MPes) representam aproximadamente 80% da indústria de software. Desta forma, visualizando a representatividade e importância deste tipo de organização, este trabalho foi desenvolvido com foco nas MPes.

Tipicamente, uma MPE é caracterizada por ter um baixo nível de maturidade, e apresenta dificuldades na aplicação direta de modelos de referência de processos de software, tais como o CMMI-DEV (SEI, 2006), a ISO/IEC 15504 (2005) ou o MPS.BR (SOFTEX, 2006). Isto acontece porque geralmente os processos de softwares nestas organizações são executados de modo informal, improvisado e com pouca visibilidade. Este cenário pode ser responsável pelas altas taxas de mortalidade, baixa competitividade e falta de acesso a determinados mercados.

Outro problema também encontrado é a variedade de modelos de negócio nas organizações de software e a falta de cobertura dos modelos de referência para diferentes modelos de negócio. A maioria dos modelos de referência está focada em projetos para o desenvolvimento de produtos sob encomenda, não fornecendo suporte direto para o desenvolvimento de produtos de prateleira (COTS – *Commercial off-the-shelf*), ou o fornecimento de serviços como manutenção.

Um passo na direção de aumentar a flexibilidade é o surgimento de modelos de referência contínuos como a ISO/IEC 15504 e CMMI com representação contínua, os quais permitem a seleção dos processos mais relevantes para auxiliar a organização a alcançar seus objetivos de negócio e melhoria. Ou seja, é possível definir um perfil alvo de capacidade baseado no modelo e objetivos de negócio da organização.

A questão está justamente na dificuldade de definir os processos relevantes para a organização sem que seja realizado algum tipo de avaliação para identificar os modelos e objetivos de negócio desta organização. É neste contexto que este trabalho analisou modelos e objetivos típicos de negócio em MPEs de software, identificando os diferentes ciclos de vida de software e fornecendo um conjunto de possíveis perfis alvo com base nos modelos de negócio levantados. Os processos foram selecionados a partir do modelo CMMI e da norma ISO/IEC 15504.

2 - Objetivos e Justificativa

Este projeto teve por objetivo disponibilizar às organizações de desenvolvimento de software uma avaliação de processo de software seguindo modelos contínuos (CMMI e ISO/IEC 15504) que esteja alinhada com seus modelos de negócio, com ênfase em MPEs de software. Foram avaliados os modelos de negócio utilizados pelas organizações e relacionadas às características destes modelos com as áreas de processo dos modelos CMMI contínuo e ISO/IEC 15504.

Apesar de existirem diversos modelos e normas para a área de melhoria de processo de software, uma das dificuldades encontradas nos modelos contínuos é a ligação entre as áreas de processo destes modelos e normas com o modelo de negócio da organização, uma vez que a priorização dos processos para ações de melhoria precisa ser alinhada ao tipo de modelo e objetivos de negócio da organização. Isto é fundamental para buscar um resultado mais significativo com menor custo na melhoria de processo.

Atualmente, avaliações de contextualização (overview assessments) são usualmente executadas no início de programas de melhoria para a identificação do modelo e dos objetivos de negócio das organizações e para eliciar as características do seu principal produto e serviço (WANGENHEIM, PICKLER, THIRY, ZOUCCAS e SALVIANO, 2005). A partir das informações coletadas, é estabelecido um perfil alvo de processos (ou uma série de perfis alvo) como base para a implementação do programa de melhoria de processo.

Entretanto, a grande variedade de modelos de negócio encontrada nas organizações torna complexa a tarefa de selecionar os processos. Além disso, existem poucas orientações disponíveis, requerendo alta experiência de profissionais em engenharia de software, e esforços e custos consideráveis.

3 - Metodologia de Execução

As atividades do presente projeto iniciaram com uma série de entrevistas para a identificação e análise dos modelos de negócio. Para tal, foram disponibilizados formulários para a caracterização do processo de negócio, aplicados durante 5 avaliações de contextualização (WANGENHEIM, PICKLER, THIRY, ZOUCCAS e SALVIANO, 2005).

Como atividade suplementar à realização das entrevistas, os autores também fizeram uma revisão sobre as experiências coletadas com diferentes organizações durante avaliações e implementações de programas de melhoria. A

partir destas informações e daquelas obtidas com as entrevistas, foi possível estabelecer os principais modelos de negócio adotados.

A partir dos modelos de negócio encontrados, buscou-se a análise dos processos existentes no modelo de referência CMMI e também na ISO/IEC 15504 para identificar a aderência dos processos em relação aos modelos de negócio. Esta atividade contou com a participação do pesquisador Clenio Salviano do CenPRA. Novamente, a experiência dos autores com a avaliação em implementação de programas de melhoria foi fundamental para a construção do mapeamento. Também foram estudados ciclos de vida de software adaptados aos processos das organizações, cruzando com os modelos de negócio empregados.

Como validação inicial do mapeamento, foram feitas 5 avaliações em MPEs da Grande Florianópolis. A partir dos resultados obtidos, os perfis identificados nas avaliações foram cruzados com aqueles definidos pela abordagem. Esta etapa do projeto permitiu ajustes na abordagem e também nas orientações de adaptação.

Para uma validação mais ampla da abordagem desenvolvida, foi construído um site onde as organizações, após se cadastrarem e responderem algumas perguntas que estabelecem o modelo de negócio adotado, podem obter perfil alvo de processo relacionado. Os resultados destas avaliações instantâneas pelas organizações estão sendo revistos pelos autores para que possam ser realizados os devidos ajustes e o refinamento das orientações de adaptação.

4 - Resultados Obtidos

O principal resultado deste projeto foi a abordagem para auxiliar as MPEs que querem implantar ações de melhoria nos seus processos de software de forma alinhada com seus respectivos modelos de negócio. Neste sentido, o trabalho caracterizou e mapeou a aderência dos diferentes modelos de negócio com as áreas de processos do modelo CMMI contínuo e processos da norma ISO/IEC 15504 no contexto de MPEs de software.

A identificação dos modelos de negócio foi baseada na classificação definida pelo MCT (2005) para as atividades da organização no tratamento do software. Embora sejam definidas outras possibilidades, este trabalho focou inicialmente em três modelos de negócio:

- **Software Pacote (COTS - Commercial off-the-shelf):** produto de software pronto para o uso, que apresenta um conjunto de características funcionais e de qualidade pré-definidas pelo fornecedor, às quais os usuários se adaptam durante o uso. São casos clássicos as suítes de escritório, tais como os processadores de texto e planilhas eletrônicas.
- **Software sob Encomenda (FD - Partially to Fully Outsourced):** produto de software desenvolvido a partir de um conjunto de requisitos obtidos pelo fornecedor junto ao cliente.
- **Software Modificado (MOTS - Modified off-the-shelf):** produto “customizável”, ou seja, um software pacote adaptado pelo fornecedor para atender a necessidades específicas de determinados clientes. Esta

situação é híbrida, pois normalmente ambas as partes, comprador e fornecedor, têm que realizar algum tipo de adaptação. O caso clássico é um software de gestão empresarial do tipo ERP.

Uma decisão importante foi considerar apenas o produto principal das organizações, uma vez que várias organizações possuem mais de uma possibilidade de tratamento do software. O resultado do mapeamento pode ser observado na tabela 1. A primeira coluna representa o processo (ISO/IEC 15504) ou área de processo (CMMI). A segunda coluna indica a origem do processo (Modelo de referência ISO/IEC 15504 ou CMMI). Em seguida, estão as colunas com a capacidade esperada em cada um dos 3 modelos de negócio avaliados. Eventualmente, quando o processo não é relevante para aquele modelo, a célula é deixada em branco.

Tabela 1. Mapeamento dos processos vs modelos de negócio

Processo ou área de processo	Modelo de Referência	FD (sob encomenda) Desenvolvimento	MOTS Customização	COTS Desenvolvimento Inicial
RD – Desenvolvimento de Requisitos	CMMI	2 ⁺ opt	2 ⁺	2 ⁺
REQM – Gerência de Requisitos	CMMI	2 ⁺	2 ⁺	1 ⁺
TS – Solução Técnica	CMMI	2 ⁺	2 ⁺	2 ⁺
PI - Integração de Produto	CMMI	2 ⁺	2	
VER - Verificação	CMMI	2 ⁺	2 ⁺	2 ⁺
VAL - Validação	CMMI	2 ⁺	2 ⁺	2 ⁺
ENG.11 – Instalação do Software	ISO/IEC 15504	2 ⁺ opt	2 ⁺ opt	
PP – Planejamento de Projeto	CMMI	1 ⁰	1 ⁰	1 ⁰
PMC – Monitoração e Controle de Projeto	CMMI	1 ⁰	1 ⁰	1 ⁰
SAM - Aquisição	CMMI	1 ⁰	*	1 ⁰ opt
RSKM – Gerência de Riscos	CMMI			1 ⁰
MAN. 4 - Gerência de Qualidade	ISO/IEC 15504		2 (x1)	
RIN.4 – Infra-estrutura	ISO/IEC 15504	1	1	1
SPL.1 – Prospecção de Fornecimento	ISO/IEC 15504	1 ⁰	1 ⁰	
SPL.2 – Liberação de Produto	ISO/IEC 15504		2	
SPL.3 – Apoio para Aceitação do Produto	ISO/IEC 15504		2	
CM – Gerência de Configuração	CMMI	2 ⁺	2 ⁺	2 ⁺
MA – Medição e Análise	CMMI	1 ⁰	1 ⁰	1 ⁰
PPQA – Garantia da Qualidade de Processo e Produto	CMMI	2 ⁺	2 ⁰	2 ⁺
DAR - Análise de Decisão e Resolução				1
SUP.10 - Gerência de Solicitação de Mudança	ISO/IEC 15504		2	

Construção de Manual / Help (baseado no Processo SUP.7 - Documentação) ¹	ISO/IEC 15504	1	2	2
OPE.2 – Suporte ao Cliente	ISO/IEC 15504	1	2 opt	
REU.1 – Gerência de Ativos	ISO/IEC 15504	1	1	
REU.2 – Gerência de Programa de Reuso	ISO/IEC 15504	1	1	
REU.3 – Engenharia de Domínio	ISO/IEC 15504		1	1
¹ Supondo que a documentação técnica do sistema de software está sendo feita em paralelo junto aos respectivos processos técnicos.				

Para a interpretação adequada dos níveis de capacidade associados com cada modelo na tabela 1, foi estabelecida uma legenda e um mapeamento com a visão de capacidade do modelo CMMI e da norma ISO/IEC 15504. Esta legenda pode ser observada na tabela 2. Para permitir uma maior granularidade entre os níveis de capacidade estabelecidos, foram criados níveis intermediários (p. ex. 2+) que não existem no modelo CMMI ou na norma ISO/IEC 15504.

Tabela 2. Legenda dos níveis de capacidade adotados no mapeamento

Níveis de capacidade usados para os perfis alvo de processo	Níveis de capacidade	Atributos de processo (ISO/IEC 15504)
3: Processo estabelecido	3: Processo estabelecido	PA 3.1 Process definition attribute PA 3.2 Process deployment attribute
2⁺ : Processo é gerenciado e definido, com templates para os produtos de trabalho.	2: Processo gerenciado	PA 2.1 Performance management attribute PA 2.2 Work product management attribute
2⁰ : Processo é gerenciado e existem templates para os produtos de trabalho críticos		
2: Processo é gerenciado		
1⁺ : Processo é executado e definido, com templates para os produtos de trabalho	1: Processo executado	PA 1.1 Process performance attribute
1⁰ : Processo é executado e existem templates para produtos de trabalho críticos		
1: Processo é executado		
0: Processo é incompleto	0: Processo incompleto	--
opt: Processos opcionais selecionados com base nas orientações de adaptação definidas na tabela 3.		

Para os processos opcionais, é utilizado um conjunto de orientações de adaptação que podem ser selecionadas de acordo com as características de cada organização. Este conjunto é mostrado na tabela 3.

Tabela 3. Orientações de adaptação

SAM* - Gerência de Acordos com Fornecedores	Se o produto adquirido é um COTS então o nível de capacidade deve ser 1º Se o produto adquirido é sob encomenda ou MOTS então o nível de capacidade deve ser 2º
RD – Desenvolvimento de Requisitos	Se, no projeto considerado, a organização já recebe os requisitos definidos (por exemplo, em uma fábrica de software), este processo pode não ser relevante
ENG.11 – Instalação	Se a organização não instala o sistema de software, este processo pode não ser relevante.
MAN. 4 – Gerência de Qualidade	(x1) A necessidade do processo depende da criticidade da qualidade para o produto de software.
OPE.2 – Suporte ao Cliente	Se a organização não oferece suporte ao cliente, este processo pode não ser relevante.

A partir do mapeamento, foram feitas 5 avaliações em MPEs de Santa Catarina, permitindo uma validação inicial e ajustes nas orientações de adaptação. Ainda, como forma de validação em uma escala maior, foi desenvolvido um site para que as organizações possam se cadastrar e identificar os perfis alvo de processo a partir do seu modelo de negócio. Os resultados deste site estarão sendo avaliados pelos autores deste trabalho como base para sucessivos refinamentos e melhoria das orientações de adaptação. Estas informações também servirão de base para a ampliação dos modelos de negócio considerados.

Um outro resultado do trabalho foi a confirmação de que muitas organizações iniciam seu trabalho por meio de desenvolvimento de produtos sob encomenda e, em algum momento, passam a focar em produtos customizados. Desta forma, existe uma migração da relevância dos processos ao longo da vida da organização.

5 - Aplicabilidade dos Resultados

Este projeto é extremamente significativo para a melhoria e avaliação de processos de software no contexto de MPEs brasileiras de forma mais adequada, uma vez que auxilia na identificação de quais processos são mais relevantes para a organização de acordo com seu modelo de negócio. Desta forma, os resultados de melhoria trarão resultados mais efetivos para as organizações, pois serão focados em suas necessidades, contribuindo para o aumento de sua competitividade no mercado nacional e internacional.

Os resultados do projeto são direcionados para MPEs de software brasileiras. Inicialmente, a avaliação foi realizada em 5 MPEs de Santa Catarina. Estas mesmas participaram também da validação dos resultados obtidos. Para uma ampla divulgação e disseminação da avaliação foi desenvolvido também um site que permite uma avaliação rápida de qualquer organização interessada em participar do projeto. A partir destas outras avaliações, é possível ampliar a validação dos resultados e executar os ajustes necessários nos perfis alvo definidos.

Este projeto é voltado para as MPEs brasileiras, oferecendo um mapeamento entre os modelos de negócio tipicamente adotados pelas organizações e as áreas de processo estabelecidas por um modelo e por uma norma internacionalmente reconhecidos. Apesar da avaliação estar inicialmente limitada às organizações catarinenses, os resultados deste projeto poderão ser estendidos também para organizações de outros estados e países, uma vez que os diferentes modelos de negócio são também adotados por estas organizações.

6 - Características Inovadoras

A principal inovação obtida com o resultado deste projeto foi disponibilizar publicamente um alinhamento da seleção das áreas de processo importantes com base no modelo de negócio de uma organização. Atualmente, não existem guias que suportem esta seleção durante uma avaliação ou melhoria de processo com base em um modelo contínuo.

A contribuição científica deste projeto foi o desenvolvimento e a aplicação de uma abordagem para a identificação instantânea dos processos relevantes na organização de acordo com o modelo de negócio adotado. Estes processos foram selecionados com base no modelo CMMI e na norma ISO/IEC 15504 e permitem que a organização possa ter uma primeira sugestão para iniciar seu programa de melhoria de processo.

Outro aspecto inovador da abordagem desenvolvida é a sua adaptação específica para o contexto das MPEs brasileiras, facilitando a sua aplicação neste tipo de organização sem o custo inicial de uma avaliação. Além disso, a aplicação da abordagem nas 5 organizações permitiu a coleta de dados e experiências sobre a aplicação e ajustes da abordagem em MPEs brasileiras.

7 - Conclusão e Perspectivas Futuras

Como resultados de pesquisa significativos, o projeto gerou uma abordagem para a identificação de perfis alvo de processos a partir dos objetivos e modelos de negócio de uma organização, especificamente no contexto das micro e pequenas empresas.

Deve-se compreender que o perfil apresentado é uma primeira idéia a partir de uma análise feita em organizações com projetos similares e que um perfil mais específico só poderia ser feito a partir de uma avaliação da organização. Entretanto, uma vez que esta identificação é instantânea, ela permite que a organização possa contar com uma sugestão inicial de melhoria alinhada com seus interesses.

A partir dos perfis alvos estabelecidos de acordo com os modelos de negócio identificados, pretende-se agora a realização de mais avaliações de contextualização que permitam a comparação entre os resultados da avaliação em termos dos processos relevantes e suas respectivas capacidades e aqueles obtidos com a abordagem proposta neste trabalho. Desta forma, objetiva-se refinar a abordagem e ajustar as orientações de adaptação, permitindo que as MPes possam ter uma orientação inicial para organizar seus programas de melhoria de processo com a certeza que o perfil está devidamente alinhado com seus objetivos e modelos de negócio.

Ainda, como perspectivas futuras, pretende-se ampliar o mapeamento para outros modelos de negócio. Este trabalho é importante para permitir atender um número maior de organizações e tornar as orientações de adaptação mais eficazes. Neste sentido, pretende trabalhar com a visão de manutenção, a qual é uma atividade comum dentro das MPes. A manutenção pode ser combinada com diferentes aspectos do desenvolvimento (por exemplo, COTS e MOTS). Desta forma, a abordagem poderá ser refinada a ponto de atender não apenas MPes, mas quaisquer organizações que adotem os modelos de negócio abordados.

Referências Bibliográficas

- Gresse Von Wangenheim, C., Pickler, K., Thiry, M., Zoucas, A., E Salviano, C. Aplicando Avaliações de Contextualização em Processos de Software Alinhados ao CMMI-SE/SW. In: SIMPROS 2005, Simpósio Internacional de Melhoria de Processo de Software, São Paulo, Brasil, 2005.
- ISO/IEC Std. 15504: Information Technology – Process Assessment, Part 1 to Part 5. International Organization for Standardization, 2003-2006.
- MCT – Ministério de Ciência e Tecnologia (Brasil). Pesquisa sobre Qualidade e Produtividade no Setor de Software, 2005. <http://www.mct.gov.br>.
- SEI - Software Engineering Institute. Capability Maturity Model Integration (CMMI) Product Suite, 2006. <http://www.sei.cmu.edu/cmmi>
- SOFTEX. Guia Geral do MPS.BR - Melhoria de Processo do Software Brasileiro (Versão 1.1). Abril, 2006.

[4.06] Next - Núcleo de Excelência em Testes de Sistemas

Entidade: Centro de Excelencia em Tecnologia de Software - SOFTEX/RECIFE - Rua Madre de Deus 27 - Recife - PE - Brasil

Autores: João Carlos Braczek e Marcos André Wanderley Gomes - joaocarlos@recife.softex.br e marcos@recife.softex.br

1. Introdução

Estima-se (2004) que o ecossistema de empresas de TIC de Pernambuco seja formado por cerca de 500 empresas que se distribuem por um largo espectro de atividades que vão desde desenvolvimento de soluções até aquelas de consultoria em informática e sistemas. Desse universo se estima que pelo menos

um quinto, ou pouco mais, delas, ou seja, entre 100 a 150 empresas se dedicam ao desenvolvimento de software. Os produtos e serviços produzidos por esse universo de empresas são sobremodo diversificados e vão desde software de entretenimento até soluções para a área de saúde passando pelo desenvolvimento de gestão integrada e aplicações wireless e indo até a soluções para a área bancária.

No que pese o seu tamanho e a diversificação dos seus produtos e serviços, o cluster de TIC em Pernambuco ainda carece de algumas características que possam alavancar a sua expansão nos mercados nacional e internacional. Em particular pode-se apontar o controle da qualidade do software por ele desenvolvido como um ponto fraco para uma maior penetração no mercado e, portanto, para a consolidação da sua reputação como centro produtor de Tecnologia da Informação.

Criar, investir e disseminar a cultura de qualidade no setor significa em última instância contribuir com uma ferramenta eficiente para consolidar o pólo local de TI. Essa consolidação por seu turno tem o condão de contribuir para a criação de um círculo virtuoso de crescimento capaz de conduzir a uma inserção dinâmica da economia estadual na economia nacional.

2. Objetivos e Justificativos

O projeto tem por objetivo criar condições para que o Laboratório de Teste e Qualidade de Software expanda sua atividade principal que é a melhoria da qualidade no setor de Tecnologia da Informação em Pernambuco, contribuindo para tornar as empresas do pólo de TI local mais competitivas na disputa por novos mercados dentro e fora do país. O resultado principal desse laboratório é colocar a disposição das empresas do ecossistema local uma instalação para testes de produtos, seus ou de terceiros. Neste local elas encontrarão capital humano, competência e infra-estrutura adequada para execução de testes de maneira mais eficaz e com menor custo do que os testes executados nas suas próprias instalações. Adicionalmente, mas não menos importantes, outros resultados serão gerados pela interação das empresas com o laboratório, como, por exemplo, a melhoria de processos e a capacitação dos seus profissionais. Trata-se, conforme se pode observar, de um projeto que contempla além da melhoria da infra-estrutura, a formação de capital humano e melhoria dos processos de desenvolvimento de software, criando elementos que tornem mais competitivas as empresas do pólo de TIC de Pernambuco.

3. Metodologia de Execução

O projeto será desenvolvido em três fases: Preparação, Definição de Processos e Operação do laboratório. Cada uma delas tem objetivos e atividades bem definidas e se encontram descritas abaixo.

3.1 Fase de Preparação

Esta fase tem por objetivos selecionar os bolsistas que comporão a equipe do projeto, capacitar esta equipe e definir as políticas de utilização do laboratório juntamente com as empresas, conforme descrito nas atividades abaixo.

- **Seleção da Equipe**

Serão selecionados para completar a equipe do projeto: um Gerente do Laboratório, um Administrador de Sistemas, e quatro Engenheiros de Teste. A equipe completa terá a configuração apresentada abaixo.

O Gerente do laboratório será responsável pela gestão do laboratório, administração dos seus serviços e promoção do mesmo, através da divulgação do projeto junto às empresas do ecossistema e aos fornecedores de hardware e software. Além disto, o Gerente do laboratório deverá reportar status e resultados obtidos com o laboratório ao Coordenador do SOFTEXRECIFE.

Os Engenheiros de Teste definirão um processo padrão de testes e fornecerão suporte às empresas na implantação ou melhoria do processo de testes, no desenvolvimento e execução de testes das aplicações, além da utilização de ferramentas de teste.

O Administrador de Sistemas será responsável pela configuração, suporte ao ambiente e gerenciamento da rede do laboratório.

- **Capacitação da Equipe**

A equipe do laboratório será capacitada para a execução das suas atividades no projeto. A capacitação será de acordo com o papel do bolsista no projeto, conforme descrito abaixo:

Capacitação do Engenheiro de testes:

- Treinamento em Qualidade de Software
- Treinamento em Testes de Software
- Treinamento de Introdução ao RUP (Rational Unified Process)

Capacitação do Gerente do Laboratório:

- Todos os treinamentos aplicáveis aos engenheiros de testes
- Treinamento de preparação para certificação PMP (Project Management Professional)

Capacitação do Administrador de Sistemas:

- Todos os treinamentos aplicáveis aos engenheiros de testes
- Treinamento em Banco de Dados (Oracle ou SQL Server).

- **Definição de políticas e modo de operação do laboratório**

Serão definidas as políticas de uso do laboratório pelas empresas e estabelecido um planejamento detalhado do modo de operação do laboratório. As empresas serão envolvidas na discussão e definição destes pontos.

3.2 Fase de Definição de Processos

Esta fase tem por objetivos desenvolver os estudos necessários à definição do processo do teste padrão do laboratório, documentar o processo, selecionar as

ferramentas de testes e planejar o ambiente operacional do laboratório, conforme detalhado nas atividades abaixo.

Estudo e definição do processo

A equipe do laboratório trabalhará na definição de um processo padrão de teste. Este processo será composto por procedimentos, guias e templates, e servirá como base para a definição do processo de testes das empresas, tendo suas características adequadas ao seu porte, maturidade e tipos de produtos que desenvolve. Este processo estará alinhado aos padrões adotados internacionalmente como: norma IEEE 829-1998, CMMI, ISO, RUP e XP.

O processo estará à disposição de qualquer empresa do ecossistema local que tenha interesse no mesmo. Tendo acesso além do processo, ao apoio da equipe do laboratório para sua adequação e implantação na empresa.

Seleção de ferramentas de testes

Serão selecionadas ferramentas para suporte ao processo de testes. O projeto avaliará e dará prioridade a soluções de software livre, mas avaliará também as ferramentas dos principais fornecedores de ferramentas de teste. Serão avaliadas ferramentas para:

- **Planejamento e gerenciamento de teste:** provêem suporte ao planejamento de testes, armazenamento dos testes, gerenciamento e controle, acompanhamento de defeitos e geração de relatórios de teste.
- **Cobertura de código:** ferramentas para análise do código fonte com o propósito de detecção de erros e violações de regras, testando a compatibilidade com padrões definidos.
- **Testes funcionais:** ferramentas que dão suporte ao desenvolvimento de scripts, captura de comportamento da aplicação, execução e avaliação de testes funcionais.
- **Testes de carga e de performance:** dão suporte ao desenvolvimento de cenários de teste de carga.

Ao final desta etapa será gerada uma lista de ferramentas indicadas para o suporte ao processo de testes. As ferramentas serão adquiridas para o laboratório de testes e poderão ser utilizadas pelas empresas quando estas estiverem fazendo uso da infra-estrutura do laboratório.

Planejamento do ambiente operacional do laboratório

O gerente do laboratório irá conduzir, junto às empresas, a um levantamento dos softwares necessários para realização dos testes dos seus produtos. Esta pesquisa incluirá softwares para: servidores de aplicação, sistemas operacionais, web browsers e outros.

Com base nesta pesquisa, será definido o ambiente inicial a ser provido pelo laboratório para os testes. O gerente do laboratório apoiará iniciativas para estabelecer parcerias a fim de viabilizar o melhor ambiente possível.

3.3 Fase de Operação do Laboratório

Esta fase se caracteriza pela entrada efetiva em operação do laboratório com sua equipe completa, e pela natureza contínua destas atividades, e tem por objetivos divulgar dos serviços do laboratório, capacitar as empresas no processo de testes, executar testes de produtos, negociar parcerias, planejar e controlar as atividades do laboratório, conforme detalhado nas atividades abaixo.

- **Divulgação do laboratório**

É fundamental para o sucesso do laboratório que haja um constante trabalho de divulgação junto às empresas do ecossistema, visando conscientizar os empresários do setor da importância do estabelecimento e amadurecimento do processo de testes para o aumento da qualidade dos produtos e da competitividade das empresas.

Serão necessárias a realização de ciclos de palestras, visitas e elaboração de material de divulgação, bem como a criação de um site na internet para o laboratório.

- **Consultoria e capacitação das empresas no processo de testes**

Para promover a implantação de processos de testes nas empresas, o projeto atuará fortemente em consultoria, capacitação e definição de processos na área de testes, visando contemplar o maior número de empresas possível, conforme descrito abaixo.

- Capacitação em testes: será fornecida capacitação em testes para até três profissionais de cada empresa.
- Consultoria na definição do processo de teste das empresas e avaliação do seu processo atual: tomando com base o processo geral de testes definido para o projeto, os profissionais do laboratório atuarão como consultores auxiliando as empresas na definição/adequação do seu processo de testes.
- Mentoring na implementação do processo de testes: através de ação conjunta com os profissionais das empresas, os consultores atuarão em projetos reais, juntamente com os profissionais da empresa para implantação do processo de testes.

- **Execução de testes de produtos**

A realização de testes nos produtos das empresas é um dos principais objetivos e resultados diretos gerados pelo projeto. Servindo como um forte apoio às empresas, ter esta atividade executada fora das suas instalações por uma equipe externa e qualificada, resultando em testes mais eficientes e redução dos seus custos. É importante destacar que atividades de testes, principalmente a execução dos mesmos, consomem bastante tempo e pessoal, significando uma parcela

considerável do custo e tempo demandados por um projeto de desenvolvimento ou manutenção de software.

As atividades de teste são muito importantes e determinantes da qualidade final de um software.

O laboratório poderá também executar testes em produtos que a empresa pretende adquirir, principalmente produtos de software livre que pretenda utilizar no seu ambiente de desenvolvimento ou operação dos seus produtos e serviços.

- **Negociar parcerias com fornecedores**

A estratégia para crescimento do laboratório é a formação de parcerias com fornecedores de hardware e software, que, visando o fortalecimento dos seus canais e de suas plataformas poderão disponibilizar equipamentos e softwares para testes das aplicações desenvolvidas utilizando seus produtos.

A negociação de novas parcerias é uma atividade constante para possibilitar uma maior diversidade hardware e software a serviço dos usuários do laboratório.

- **Planejamento e controle das atividades do laboratório**

Consiste em receber solicitações de serviços, negociar prazos e realizar o acompanhamento das atividades a serem realizadas para garantir o bom nível de serviço do laboratório durante sua operação.

- **Cronograma de Execução**

A tabela abaixo representa um cronograma de execução do projeto. As atividades e fases aqui apresentadas estão descritas na seção anterior. O Planejamento completo, incluindo datas, e maiores detalhes (como subdivisão das atividades) será elaborado iterativamente com o andamento do projeto.

Atividade / Mês	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Fase de Preparação																			
Seleção da equipe																			
Capacitação da equipe																			
Definição de políticas e modo de operação do laboratório																			
Fase de definição de processos																			
Estudo e definição do processo																			
Seleção de ferramentas de testes																			
Planejamento do ambiente operacional do laboratório																			
Fase de Operação do Laborat																			
Divulgação do laboratório																			
Consultoria e capacitação das empresas no processo de testes																			

Execução de testes de produtos																				
Negociar parcerias com Fornecedores																				
Planejamento e controle das atividades do laboratório																				

4 . Resultados Obtidos

Durante o período do projeto, foi realizada a estruturação na área de teste servida pelo NEXT. Dentro do seu processo de implantação o NEXT promoveu um total de 25 cursos, treinamento, capacitação em processos de desenvolvimento e teste de software, totalizando 236 horas e envolveu 416 participantes. Além destas atividades, foram dispendidas mais de 3.000 horas, entre projetos pilotos e **mentoring** em teste de software.

Os cursos foram ministrados como parte do trabalho de difundir a cultura de qualidade e teste de software no pólo local de Tecnologia da Informação. A sua execução se deu dentro do espírito do SOFTEXRECIFE em não manter um corpo próprio de instrutores, mas sim contar com as **expertise** das empresas locais, a maioria das quais associadas ao SOFTEXRECIFE. Apenas quando não se conseguiu identificar uma **expertise** local, o SOFTEXRECIFE foi buscar em outras praças, profissionais para ministrar novos cursos.

Seguem os resultados obtidos por fase do projeto:

4.1. Fase de Preparação

Esta fase teve por objetivos selecionar os bolsistas para compor a equipe do projeto, capacitar esta equipe e definir as políticas de utilização do laboratório juntamente com as empresas, conforme descrito nas atividades abaixo.

Seleção da Equipe

Foram selecionados para completar a equipe do projeto: um Gerente do Laboratório, um Administrador de Sistemas, e quatro Engenheiros de Teste.

O Gerente do laboratório é o responsável pela gestão do laboratório, administração dos seus serviços e promoção do mesmo, através da divulgação do projeto junto às empresas do ecossistema e aos fornecedores de hardware e software. Além disto, o Gerente do laboratório deverá reportar status e resultados obtidos com o laboratório ao Coordenador do SOFTEXRECIFE.

Os Engenheiros de Teste definirão um processo padrão de testes e fornecerão suporte às empresas na implantação ou melhoria do processo de testes, no desenvolvimento e execução de testes das aplicações, além da utilização de ferramentas de teste.

O Administrador de Sistemas será responsável pela configuração, suporte ao ambiente e gerenciamento da rede do laboratório.

Capacitação da Equipe

A equipe do laboratório foi capacitada para a execução das suas atividades no projeto. A capacitação foi de acordo com o papel do bolsista no projeto, alocando-se as bolsas BEP do projeto, que contemplaram os treinamentos, conforme descrito abaixo:

Capacitação do Engenheiro de testes:

- Treinamento em Qualidade de Software
- Treinamento em Testes de Software
- Treinamento de Introdução ao RUP (Rational Unified Process)
- Treinamento em ferramentas de automação de testes

Capacitação do Gerente do Laboratório:

- Todos os treinamentos aplicáveis aos engenheiros de testes
- Treinamento de preparação para certificação PMP (Project Management Professional)

Capacitação do Administrador de Sistemas:

- Todos os treinamentos aplicáveis aos engenheiros de testes
- Treinamento em Banco de Dados (Oracle ou SQL Server)

Definição de políticas e modo de operação do laboratório

Foram definidas as políticas de uso do laboratório pelas empresas e estabelecido um planejamento detalhado do modo de operação do laboratório.

4.3. Definição de Processos

Esta fase tem por objetivos desenvolver os estudos necessários à definição do processo de teste padrão do laboratório, documentar o processo, selecionar as ferramentas de testes e planejar o ambiente operacional do laboratório, conforme detalhado nas atividades abaixo.

Estudo e Definição do Processo

O processo padrão de teste foi definido pela equipe e implementado nos projetos piloto realizados. Este processo é composto por procedimentos, guias e templates, e serve como base para a definição do processo de testes das empresas, tendo suas características adequadas ao seu porte, maturidade e tipos de produtos que desenvolve. Este processo está alinhado aos padrões adotados internacionalmente como: norma IEEE 829-1998, CMMI, ISO, RUP e XP.

O processo está à disposição de qualquer empresa do ecossistema local que tenha interesse no mesmo. Tendo acesso além do processo, ao apoio da equipe do laboratório para sua adequação e implantação na empresa.

Seleção de Ferramentas de Testes

Foram selecionadas ferramentas para suporte ao processo de testes. O projeto avaliou e deu prioridade a soluções de software livre, mas foram avaliadas

também as ferramentas dos principais fornecedores de ferramentas de teste e adquiridas ferramentas para:

Planejamento e Gerenciamento de Teste: provêem suporte ao planejamento de testes, armazenamento dos testes, gerenciamento e controle, acompanhamento de defeitos e geração de relatórios de teste.

Cobertura de Código: ferramentas para análise do código fonte com o propósito de detecção de erros e violações de regras, testando a compatibilidade com padrões definidos.

Testes Funcionais: ferramentas que dão suporte ao desenvolvimento de scripts, captura de comportamento da aplicação, execução e avaliação de testes funcionais.

Testes de Carga e de Performance: dão suporte ao desenvolvimento de cenários de teste de carga.

As ferramentas foram adquiridas para o laboratório de testes e podem ser utilizadas pelas empresas quando estas estiverem fazendo uso da infra-estrutura do laboratório.

Planejamento do Ambiente Operacional do Laboratório

O gerente conduziu, junto às empresas, levantamento dos softwares necessários para realização dos testes dos seus produtos. Esta pesquisa incluiu softwares para: servidores de aplicação, sistemas operacionais, web browsers e outros. Com base nesta pesquisa, foi definido o ambiente inicial do laboratório, criando as condições necessárias para a pesquisa e desenvolvimento dos pilotos de teste.

4.3 Operação do Laboratório

Está é a fase atual do projeto, na qual estamos desenvolvendo atividades efetivas junto as empresas e instituições do pólo de TIC de Pernambuco.

Projetos Pilotos de Teste

Para início das suas atividades o NEXT desenvolveu dois projetos pilotos, com o objetivo de validar e melhorar os seus processos e a capacitação do seu corpo técnico. Estes projetos tiveram como foco, a sua execução em um ambiente real do cliente.

Projeto Piloto na Agência de Tecnologia da Informação - ATI

O objetivo deste Projeto era validar a metodologia de execução de testes de sistema e exercitar o uso das ferramentas adotadas para o NEXT, em um projeto real. O Projeto selecionado foi o SG.NET / ABP – Sistema de Gestão de Governo – módulo Bens Patrimoniais. Os testes para este projeto foram planejados sobre um subconjunto importante e pequeno de requisitos funcionais.

Este projeto abrangeu todas as fases da metodologia desenvolvida para o NEXT, desde a elaboração de casos de teste até a sua execução (automatizada e manual). Para isto foram envolvidos analistas de negócio, engenheiros de teste,

projetistas de testes, administrador de rede e engenheiro de qualidade, além de técnicos da ATI, perfazendo um total de 9 pessoas. O projeto teve uma duração de 2 meses e dispendeu mais de **1.000 horas**, atingindo plenamente os seus objetivos, tanto pela parte da ATI como do NEXT.

Projeto Piloto na Capital Login

O objetivo deste Projeto para o NEXT era validar e melhorar a metodologia de execução de testes de sistema definida para o laboratório em um ambiente real e exercitar uma nova estratégia de testes. A partir do piloto realizado com a ATI, muitas melhorias foram implementadas e havia necessidade de testarmos em um ambiente real a sua efetividade. Para a Capital Login este projeto será um projeto experimental com objetivo de prover conhecimentos para sua equipe técnica em processos de testes de sistemas.

O sistema escolhido para o projeto foi o **ServiceDesk**. A diretoria da Capital Login selecionou este sistema por ser um projeto interno da empresa e desta forma com maior flexibilidade em prazos.

As prioridades deste projeto estão descritas abaixo:

- Detalhar as atividades requisitadas para preparar e conduzir os testes do sistema.
- Comunicar para as pessoas envolvidas as tarefas executadas e o escalonamento de cada tarefa.
- Acompanhamento e validação pela Capital Login de todos os artefatos produzidos.

O projeto foi mais abrangente que o da ATI, pois envolveu mais funcionalidades e novas ferramentas e artefatos implementados pelo NEXT. Para uma melhor avaliação e ajustes nos processos, o projeto foi dividido em duas etapas de 2 meses cada uma. Foram dispendidas mais de **1.500 horas** por uma equipe de 8 técnicos formados pelas duas empresas. Os resultados do projeto foram plenamente atingidos e amadureceram os processos elaborados pelo NEXT, num ambiente real de execução.

Prova de Certificação em Teste (CBTS)

ALATS criou a Certificação Brasileira de Teste de Software (CBTS) para atender a uma exigência do mercado brasileiro que sempre demandou um processo de certificação e qualificação de profissionais em teste de software.

A CBTS visa estabelecer padrões para uma avaliação da qualificação dos profissionais de TI com funções na área de Testes. Adquirir o certificado CBTS para o profissional da área de Teste é um grande diferencial, pois indica que o mesmo possui um excelente nível de competência profissional nos princípios e nas práticas de Teste/Qualidade de Software, dentre os demais profissionais de TI.

Tornar-se um certificado CBTS, significa tornar-se membro de um grupo seleto de profissionais reconhecidos na área de teste de software, e receber este

reconhecimento de sua competência, é conseguir uma ascensão potencialmente mais rápida em sua carreira, e uma maior aceitação no mercado de TI.

O exame realizado em 30 de Setembro de 2006, contou com a presença de 113 candidatos e foi realizado simultaneamente em 8 estados Brasileiros. Seu índice de aprovação foi de 38,94%. Agora são 54 profissionais certificados na CBTS em todo o Brasil. **Pernambuco** foi o único estado a atingir **100%** de aprovação. Todos os aprovados fizeram curso preparatório promovido pelo NEXT.

Mentoring de Teste na Procenge

O objetivo deste projeto foi capacitar a equipe UGE nas melhores práticas no contexto de testes de software, especialmente na automação de testes, de forma a proporcionar ganhos de produtividade e da qualidade do produto da empresa. As frentes do trabalho a serem executadas foram organizadas conforme relacionadas a seguir:

- **Capacitação:** capacitar à equipe UGE no corpo de conhecimento relacionado a testes de software.
- **Melhoria do processo:** promover melhorias no processo atual da empresa, de forma a adicionar as melhores práticas de testes. A melhoria do processo deverá se basear nos modelos de qualidade para a área de software, o CMMI e o MPS.BR, bem como nas melhores referências da engenharia de software.
- **Automação de testes:** capacitar a equipe UGE na automação de testes, usando a ferramenta de teste TestComplete.

Neste projeto o envolvimento do NEXT, foi desde a seleção da equipe de automação, capacitação e acompanhamento dentro do cliente. A metodologia aplicada, de acompanhar os resultados alcançados junto com a equipe da PROCENGE, demonstrou ser bastante eficiente, conforme aponta os resultados alcançados pelo projeto, em menos de três meses:

Resultados alcançados:

- Indicadores de performance voltaram a evoluir positivamente
 - ☐ Redução do % de **bugs** detectados em casa X clientes
 - ☐ Redução do **backlog** de **bugs**
- Detecção de problemas que antes eram quase impossíveis de serem verificados
- Testes analíticos, repetíveis e cobrindo combinações de eventos que manualmente eram inimagináveis
- Base de conhecimento sobre testes completamente documentada

- Significativa diminuição do *stress* de fim de projeto (hora-extra ≈ 0)
- Automação de mais de 250 casos de teste críticos
- Eliminação dos **Betas**
- Equipe motivada

Mentoring de Teste na MV

O objetivo deste projeto é formar uma célula de teste automatizados na MV Sistemas, empresa de desenvolvimento de sistemas para área hospital. Os trabalhos foram iniciados em dezembro de 2006 e deverão se estender até meados de 2007. Neste projeto temos 3 fases distintas conforme descrito a seguir: relacionadas a seguir:

- **Capacitação:** capacitar à equipe da MV no corpo de conhecimento relacionado a testes de software.
- **Melhoria do processo:** promover melhorias no processo atual da empresa, de forma a adicionar as melhores práticas de testes. A melhoria do processo deverá se basear nos modelos de qualidade para a área de software, o CMMI e o MPS.BR, bem como nas melhores referências da engenharia de software.
- **Automação de testes:** capacitar a equipe MV na automação de testes, usando a ferramenta de teste TestComplete.

Cooperação para realização do I Simpósio Brasileiro de teste de Software (SBTS) Nov/2006

O NEXT participou com apoio administrativo e técnico para realização do I SBTS, em parceria com a UNIBRATEC. Durante o evento ministramos o curso de introdução a teste de software, para mais de 50 participantes.

Implementação de diretoria regional da ALATS (Associação Latino Americana de teste de Software)

Com a criação da diretoria regional, nos qualificamos para divulgar as melhores práticas de teste de software de forma a permitir a evolução de toda a comunidade envolvida neste trabalho, e para isso, promove eventos, tais como, cursos, seminários, congressos, grupos de trabalho, workshops, dentre outros. A idéia é sempre fomentar o debate que permitirá a troca constante de conhecimentos entre a comunidade latino americana de teste de software.

Parceria do NEXT com a BSTQB/ISTQB para formação de instrutores para certificação em teste de software

Com a parceria firmada com o RSI/ISTBQ, estaremos habilitados na formação de instrutores visando a certificação do ISTBQ, uma das maiores organizações de certificação em teste de software do mundo, com isto estaremos elevando o padrão dos técnicos do ecossistema de TIC de Pernambuco a padrões aceitos mundialmente, elevando a nossa competitividade tanto no mercado local com internacional.

Modelo de Negócio e Captação de Novos Clientes

No decorrer do ano de 2006, desenvolveu-se diversas ações de divulgação do portfólio de competências desenvolvidas pelo Next, o qual resultou na concretização dos primeiros negócios junto ao mercado local. Pelas premissas, vislumbra-se para o próximo período, a consolidação do NEXT como agente de soluções inovadoras em mentoring em processos de desenvolvimento e testes de software, com excelentes perspectivas de novos negócios no ecossistema local e estados vizinhos.

5. Aplicabilidade dos resultados

O mercado para os serviços prestados pelo Laboratório de Teste e Qualidade de Software é formado prioritariamente pelo pólo de Tecnologia da Informação e Comunicação localizada em Pernambuco, Conforme já se teve oportunidade de mencionar, esse pólo é constituído por algo em torno de 500 empresas das quais entre 100 e 150 se dedicam prioritariamente ao desenvolvimento de software e serviços associados.

Trata-se, portanto, de um universo bastante amplo posto que muito pouco foi feito em termos de teste de software mesmo pelas empresas maiores. Com efeito, a questão de teste de software ainda é incipiente e as empresas quando necessitam desses procedimentos tem, elas próprias, que criarem os seus sistemas e estruturas para tanto. Isso tem uma implicação em termos de eficiência e, por consequência, de custos para as empresas uma vez que a empresa precisa manter

um departamento ou, pelo menos uma equipe, encarregada desses serviços.

As questões vinculadas à teste de software, conforme visto anteriormente, ainda não se fazem muito presentes entre as empresas que compõem o pólo de TIC em Pernambuco. Isso, se por um lado, emerge como uma dificuldade inicial haja visto que um amplo trabalho pedagógico será requerido para demonstrar o impacto da questão de qualidade nos resultados, por outro se apresenta como um terreno virgem a ser explorado.

Inicialmente, portanto, o grande esforço será o de disseminar essa cultura e demonstrar que qualidade é uma outra face da produtividade e que, ademais, há uma relação positiva entre qualidade e resultado empresarial, ainda que a primeira vista o empresário possa não se aperceber. Decorre dessa constatação a necessidade inicial de se colocar um serviço e uma equipe treinada para demonstrar essa relação com os seus impactos sobre os resultados das empresas.

Na medida em que se avance na difusão e que, portanto, as empresas incorporem o teste como uma parte inerente ao desenvolvimento dos seus produtos, ou seja, quando as empresas perceberem a agregação de valor trazida por essa atividade, o foco do laboratório irá mudando da ação pedagógica para a prestação de serviços, movimento este que já vem ocorrendo com o NEXT no mercado de TIC de Pernambuco.

6. Características Inovadoras

A proposta de criação do NEXT se mostra inovadora no sentido de viabilizar para empresas de pequeno e médio porte, acesso a processos e ferramentas de teste, que de outra forma elas não teriam acesso. Outro fator inovador é a forma de atuação integrada a realidade e necessidades das empresas do pólo de TIC de Pernambuco.

7. Conclusão e Perspectivas Futuras

Com o aumento do nível de exigência da qualidade do software, tanto nacional como internacionalmente, um grande numero de empresas buscam no processo de teste de software, a saída para se tornarem mais competitivas. A implantação de um processo de teste de software é caro e complexo para a grande maioria das empresas nacionais, que são compostas por micro e pequenas empresas. Diante disto o modelo implantado pelo projeto do Laboratório de Testes - NEXT, que provê a utilização de uma infra-estrutura compartilhada e capaz de viabilizar a cultura de teste de software às empresas, que sem sua existência, não teriam acesso a ela.

Durante a implementação do projeto surgiram alguns fatores, que provocaram atrasos na sua conclusão, tais como:

Dificuldade na alocação de capital humano qualificado para preenchimento das bolsas do projeto, ocasionado pelo alto grau de exigência para implementá-las e concorrência no mercado de trabalho.

Falta de capacitação das empresas em processos de teste de software. Isto resultou na elaboração de um amplo programa de capacitação, não previsto inicialmente, mas que é de vital importância para o sucesso do projeto, pois ele viabiliza os serviços prestados pelo Laboratório de Testes – NEXT às empresas do ecossistema Local. Apesar de ter provocado um atraso considerável no projeto, a ação tomada, gerou benefícios que irão se estender para um número mais amplo de empresas do que o originalmente previsto.

Atualmente o Laboratório de Testes – NEXT está em sua fase operacional, com as etapas de preparação, definição de processos e planejamento do ambiente operacional, implementadas. Com isto já disponibilizamos para as empresas: infraestrutura de TI, pessoal qualificado, processos definidos, treinamentos e consultorias, que juntos fomentam a cultura de teste de software no ecossistema de Pernambuco.

Com o início efetivo da atuação comercial do NEXT e a carência do mercado em relação a teste de software, a perspectiva futura se mostra bastante promissora em relação ao sucesso do laboratório de teste. Outro fator relevante são as parcerias firmadas com empresas e instituições de renome, que alavanca o projeto e facilita a disseminação da cultura de teste de software na região e no país.

[4.07] Processo de Gerência de Pedidos de Alteração Relato de Experiência - Ilog Tecnologia

Entidade: 1Ilog Tecnologia Ltda. - Rua Lauro Linhares, 589 - 1º andar, B. Trindade, Florianópolis-SC, 88036-000, Brasil

Autores: Demetrius Ribeiro Lima¹, Gisele Trentin da Silva¹ - {gisele, ribeiro}@ilog.com.br

Resumo. *Este artigo relata a experiência da empresa Ilog Tecnologia na definição, modelagem e implantação sistematizada do seu processo de desenvolvimento de software. A Ilog é uma empresa de desenvolvimento de sistemas e soluções para e-learning, sendo o Parla! e o Web Ensino seus principais produtos no mercado. O trabalho discorre, especificamente, sobre a modelagem e implantação do processo de Gerência de Pedidos de Alteração que visa garantir que os pedidos de alteração solicitados pelos clientes sejam gerenciados, rastreados e controlados, melhorando assim a qualidade e a produtividade da manutenção dos softwares desenvolvidos pela empresa. A definição e implantação dos processos de software na Ilog segue a metodologia ASPE/MS (Approach for Software Process Establishment in Micro and Small Companies), desenvolvida pelo Laboratório de Qualidade e Produtividade de Software (LQPS) da Universidade do Vale do Itajaí (UNIVALI), em parceria com o Geness (UFSC).*

Abstract. *This paper narrates the experience of Ilog Technology company on the definition, modeling and systemize implantation of its software development process. Ilog is a company of systems and solutions development for e-learning, being the Parla! e the Web Ensino its main business products.*



The paper specifically discern about the modeling and implantation of the Management of Alteration Order process that aims to guarantee that the customers requested of alteration order may be managed, tracked and controlled, improving the maintenance's quality and productivity of the softwares developed by the company. The definition and implantation of the software processes at Ilog are based on the ASPE/MSc methodology (Approach for Software Process Establishment in Micro and Small Companies), developed by the Laboratory of Software Quality and Productivity of the University of the Itajai Valley (UNIVALI) in partnership with Geness (UFSC).

1. Introdução

A qualidade dos produtos de uma empresa está ligada à qualidade de seus processos.

Micro e pequenas empresas têm geralmente um processo de software informal e, conseqüentemente, dependem principalmente da competência das pessoas envolvidas (Ministério da Ciência e Tecnologia, 2004). Nesse contexto, o estabelecimento sistemático de processos pode contribuir significativamente na sua melhoria e assim, aumentar sua competitividade e suas chances de sobrevivência (Weber et. al., 2005).

Um **processo de software** é definido como um conjunto de atividades, métodos, práticas e transformações que pessoas empregam para desenvolver e manter software e produtos associados (Paulk et. al., 1994).

Segundo a Softex (2005), a ISO/IEC 15504 presta-se à realização de avaliações de processos de software com dois objetivos:

- 1) A melhoria de processos e a determinação da capacidade de processos de uma unidade organizacional. Dessa forma, a unidade organizacional pode realizar uma avaliação com o objetivo de gerar um perfil dos processos que será usado para a elaboração de um plano de melhorias. A análise dos resultados identifica os pontos fortes, os pontos fracos e os riscos inerentes aos processos.
- 2) A organização tem o objetivo de avaliar um fornecedor em potencial, obtendo o seu perfil de capacidade. O perfil de capacidade permite ao contratante estimar o risco associado à contratação daquele fornecedor em potencial para auxiliar na tomada de decisão de contratá-lo ou não.

Para a melhoria dos seus processos de software, a Ilog firmou parceria com a equipe de Qualidade da Incubadora Centro GeNESS da Universidade Federal de Santa Catarina (UFSC) e o Laboratório de Qualidade e Produtividade de Software da Universidade do Vale do Itajaí (UNIVALI), adotando a metodologia ASPE/MSc (Approach for Software Process Establishment in Micro and Small Companies).

A abordagem ASPE/MSc está organizada em fases visando, de forma incremental, estabelecer e melhorar continuamente os processos. A Figura 1 mostra uma visão geral da abordagem e suas fases: Diagnóstico do Processo de Software Atual, Análise Estratégica, Definição do Processo, Implantação do Processo e o Gerenciamento.

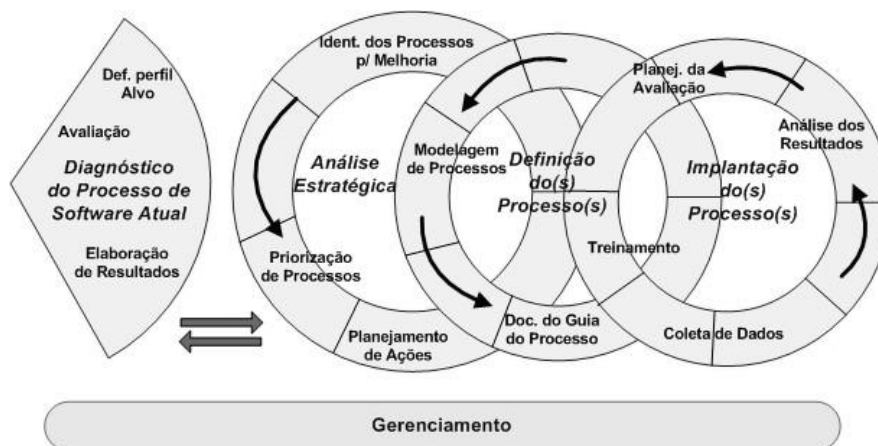


Figura 1. Abordagem ASPE/MSD

2. Objetivos e Justificativa

Em dezembro de 2003 a empresa iniciou um programa de melhoria de seus processos de software, em cooperação com o LQPS/UNIVALI e o grupo QS/UFSC, com o objetivo de melhorar a qualidade dos seus produtos e serviços e buscar a avaliação MPS.BR.

O primeiro passo do programa de melhoria foi a realização de uma avaliação dos processos da organização, utilizando o método MARES (Gresse von Wangenheim et al, 2006a), em conformidade com a norma ISO/IEC 15504 (ISO, 2003). Através dessa avaliação, a empresa percebeu que era necessário melhorar seus processos e, em maio de 2004, decidiu dar continuidade ao programa de melhoria através do estabelecimento de seus processos, seguindo a abordagem ASPE/MSD (Weber, 2005) (Weber et al, 2005) (Gresse et al, 2006b).

A seguir, descreve-se como foi modelado e implantado o processo de Gerência de Pedidos de Alteração na Ilog, conforme a abordagem ASPE/MSD.

3. Metodologia de Execução

Em maio de 2004 iniciou-se a aplicação da abordagem na Ilog, que contou com o auxílio de alguns funcionários da própria empresa, do autor da abordagem e a participação da diretoria. No início, foram feitas algumas reuniões com os diretores para definir os objetivos do programa de melhoria e selecionar os responsáveis pela sua execução. Um pesquisador júnior do LQPS foi designado como Engenheiro do Processo e um dos colaboradores como Representante da Organização. Durante a fase de análise estratégica, os processos foram priorizados e o processo de Gerência de Pedidos de Alteração foi selecionado para ser estabelecido. Na ocasião, os processos de Gerência de Projetos e Instalação de Software foram escolhidos para serem estabelecidos na sequência da aplicação da abordagem.

Buscando definir o processo de Gerência de Pedidos de Alteração, foram feitas várias reuniões com os envolvidos, sobretudo com a equipe de suporte técnico,

que era a responsável por gerenciar os pedidos de alteração feitos pelos clientes. Durante essa etapa, as principais atividades foram levantadas, bem como a sequência entre elas e seus responsáveis. Além disso, alguns artefatos foram identificados e uma representação inicial do processo foi criada e documentada.

O processo de Gerência de Pedidos de Alteração é iniciado sempre que o cliente manifesta a necessidade em realizar alguma alteração na versão atual do software que o mesmo utiliza. Esse desejo deve ser formalizado através de um e-mail e enviado para o endereço eletrônico do suporte técnico da Ilog. Quando o setor de suporte recebe o email e o qualifica como um pedido de alteração, deve ser feita uma análise, verificando os procedimentos subsequentes a serem realizados. Caso o cliente solicitante tenha contrato de manutenção e esteja em dia com suas obrigações contratuais, o pedido é encaminhado para um analista de sistemas que possua conhecimentos no software a ser alterado. Do contrário, o pedido deve ser passado para a diretoria da Ilog para uma negociação contratual (Weber & Silva, 2005).

Dando prosseguimento ao processo, o analista de suporte avalia o pedido recebido e dá um parecer técnico sobre esse. Finalizada a análise, o pedido é encaminhado para a gerência de projetos, a fim de planejar o atendimento da solicitação. Nas etapas subsequentes, o pedido é encaminhado para o desenvolvimento para que os analistas de suporte desenvolvam e implantem a alteração. Finalizadas essas atividades, o cliente é notificado, sendo informado da conclusão do atendimento da solicitação. Para auxiliar na visualização e compreensão do processo, o mesmo foi modelado por meio de um fluxograma, conforme Figura 2.

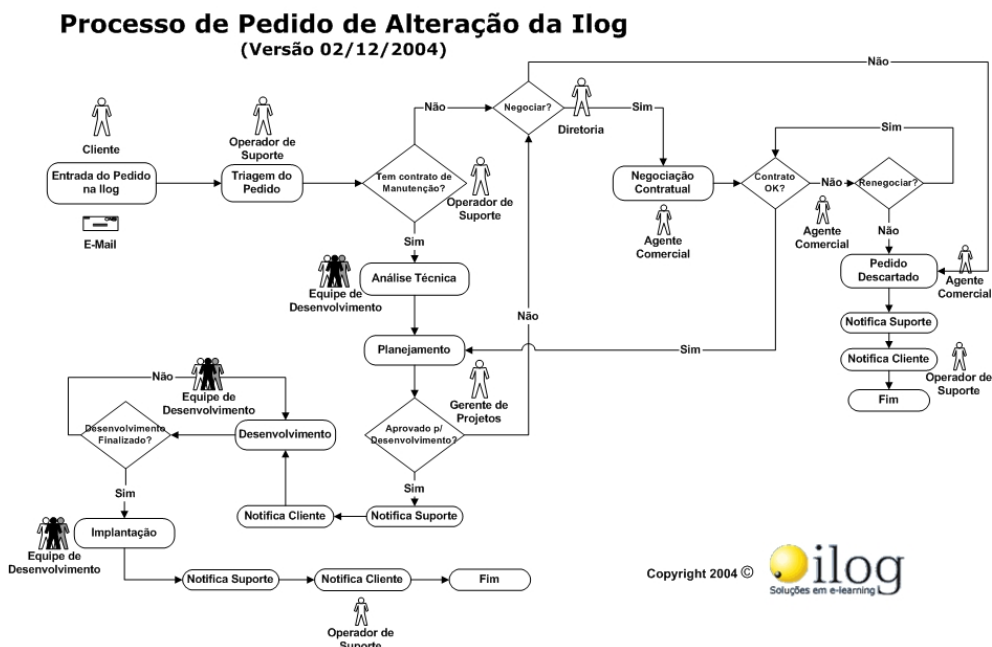


Figura 2. Fluxograma do Processo de Gerência de Pedidos de Alteração

Todas as atividades identificadas foram detalhadas pelos envolvidos na sua execução e revisadas pela equipe, conforme segue:

- **Entrada do Pedido:** entrada de um pedido do cliente via e-mail no suporte técnico.
- **Triagem do Pedido:** os atendentes de suporte fazem a triagem dos pedidos de alteração e redirecionando-os para a ferramenta. Se o cliente tem contrato de manutenção, o pedido é movido para a Análise Técnica; caso contrário, o pedido é movido para a Análise da Diretoria para ser renegociado com o cliente.
- **Análise Técnica:** os analistas de suporte dão um parecer técnico sobre o pedido efetuado e movem o mesmo para o Planejamento. Caso o pedido seja uma correção urgente, esse é movido diretamente para o Desenvolvimento.
- **Planejamento:** o pedido é planejado pelo Gerente de Projetos que define os prazos de entrega e as horas previstas para desenvolvimento.
- **Notificação ao cliente:** os atendentes de suporte notificam o cliente se o pedido foi aprovado para desenvolvimento. Também notificam o cliente se um pedido foi descartado, se o pedido precisa ser homologado ou se o pedido foi implantado no servidor.
- **Desenvolvimento:** o analista de suporte executa o desenvolvimento.
- **Implantação:** após o desenvolvimento, o pedido é movido para a Implantação.
- **Negociação Contratual:** quando cliente não possui contrato de manutenção, o setor comercial deve entrar em contato com o cliente para renegociação.

Para permitir o controle, a busca e o gerenciamento das solicitações realizadas, a empresa adotou uma ferramenta *open source* chamada OTRS (<http://otrs.org>), na qual foram feitas algumas customizações para atender os propósitos estabelecidos. Além disso, foi necessário dar toda a orientação aos envolvidos sobre como desempenhar as atividades descritas no modelo do processo utilizando a ferramenta. Para facilitar essa orientação, foi construído um guia do processo (Weber & Silva, 2005), que foi disponibilizado num servidor de arquivos da empresa. A partir daí houve vários refinamentos no guia, de modo que a fase de implantação do processo iniciou-se no final de novembro do ano de 2004.

A implantação foi efetivada na empresa em janeiro de 2005, quando foi instituído oficialmente o novo processo na organização através de um comunicado da diretoria aos funcionários e de um treinamento com todos os envolvidos. A partir da implantação do processo na empresa, todos os envolvidos passaram a utilizar a ferramenta e o processo passou a ser utilizado de forma obrigatória dentro da empresa, podendo, a qualquer momento, sofrer alterações visando sua melhoria.

4. Resultados Obtidos

Nesta seção são apresentados os resultados obtidos com a execução do projeto:

Pedidos de patente e/ou patentes registradas: o projeto não resultou em pedidos de patentes ou patentes registradas.

Produtos de software gerados (módulos ou programas de computador resultantes do projeto, disponibilizados para o mercado): o projeto não resultou em produtos de software disponíveis para o mercado.

Artigos publicados: C.Gresse von Wangenheim, S. Weber, J. C. R. Hauck, G. Trentin. **Experiences on Establishing Software Processes in Small Companies.**

Aprovado para publicação em Information and Software Technology, Elsevier, 2006b.

Weber, S.; Hauck, J.C.R.; Wangenheim, C. G. Von. **Estabelecendo Processos de Software em Micro e Pequenas Empresas.** Florianópolis, 2005.

Recursos humanos capacitados (especialistas, mestres, doutores, etc.): a colaboradora Gisele Trentin da Silva, mestre na área da Ciência da Computação, foi capacitada para trabalhar na área de Qualidade de Software.

Dissertações e/ou teses geradas: S. Weber. **ASPE / MSC: Uma Abordagem para Estabelecimento de Processos de Software em Micro e Pequenas Empresas.**

Dissertação de Mestrado, Programa de Pós-Graduação em Ciência da Computação, UFSC, 2005.

Eventuais parcerias ou programas de transferência de tecnologia efetuados: parceria com o Laboratório de Qualidade e Produtividade de Software (LQPS) da Universidade do Vale do Itajaí (UNIVALI) e com o grupoQS/Geness (UFSC).

5. Aplicabilidade dos Resultados

O projeto foi relevante porque conduziu a organização a uma melhora significativa na produtividade e na qualidade dos serviços prestados. Os papéis e responsabilidades dos envolvidos no processo ficaram mais claros e facilitaram a coleta de métricas, melhorando as estimativas enviadas aos clientes.

Assim, através dos resultados obtidos nas avaliações e observações do processo na empresa, pode-se afirmar que o investimento em modelagem de processos é um excelente caminho para organizações que buscam excelência em qualidade e produtividade.

6. Características inovadoras

O projeto foi inédito porque foi a primeira empresa de Florianópolis – SC a utilizar o método MARES (Gresse von Wangenheim et al, 2006a) e a abordagem ASPE/MSC (Weber, 2005) (Weber et al, 2005) (Gresse et al, 2006b) para melhoria do processo de Gerência de Pedidos de Alteração.

7. Conclusão e Perspectivas Futuras

Neste artigo foi apresentado o projeto da empresa Ilog Tecnologia que utilizou a abordagem ASPE/MSC para a melhoria dos seus processos de software.

Atualmente, o processo de Gerência de Pedidos de Alteração está sendo devidamente gerenciado e controlado para garantir que as tarefas estão sendo executadas conforme o planejado. O guia do Processo de Gerência de Pedidos de Alteração está em constante atualização e os envolvidos são notificados na ocorrência de novas alterações. Dando prosseguimento à modelagem de processos, a empresa está melhorando outros processos (Gerência de Projetos e Gerência de Requisitos) para buscar a avaliação MPS.BR, no nível G.

8. Referências Bibliográficas

Anacleto, A., Wangenheim, C. G. von, Salviano, C. F. **Relatório Final da Avaliação de Processos Seleccionados segundo a ISO/IEC TR 15504 no Projeto LQPS – ILOG**. Dezembro de 2003.

C. Gresse von Wangenheim, A Anacleto, C. F. Salviano. **Helping Small Companies Assess Software Processes**. *IEEE Software*, Vol. 23, No. 1, Jan/Feb 2006a.

C.Gresse von Wangenheim, S. Weber, J. C. R. Hauck, G. Trentin. **Experiences on Establishing Software Processes in Small Companies**. Aprovado para publicação em *Information and Software Technology*, Elsevier, 2006b.

ISO/IEC Std. 15504: **Information Technology – Process Assessment**, Part 1 to Part 5. International Organization for Standardization, 2003-2006.

Ministério da Ciência e Tecnologia. **Qualidade e Produtividade no Setor de Software Brasileiro: Resultados da Pesquisa 2001**. Disponível em: <http://www.mct.gov.br/Temas/info/Dsi/Quali2001/Public2001.htm>. Acessado em 28 fevereiro de 2004.

Paulk, M. C.; Weber, C. V.; Curtis, B.; Chrissis, M. B. **The Capability Maturity Model: Guidelines for Improving the Software Process**. Addison-Wesley, 1994.

Softex, Sociedade. **MPS.BR - Melhoria de Processo do Software Brasileiro Guia Geral**. Versão 1.0. Abril de 2005.

S. Weber. ASPE / MSC: **Uma Abordagem para Estabelecimento de Processos de Software em Micro e Pequenas Empresas**. Dissertação de Mestrado, Programa de Pós-Graduação em Ciência da Computação, UFSC, 2005.

Weber, S.; Hauck, J.C.R.; Wangenheim, C. G. Von. **Estabelecendo Processos de Software em Micro e Pequenas Empresas**. Florianópolis, 2005.

Weber, S.; Silva, G. T. **Guia do Processo de Gerência de Pedidos de Alteração**. Florianópolis, 2005.

Tecnologia de Software

[6.05] Especificando Casos de Uso com Qualidadei

Entidade: 1 Faculdade de Informática - Pontifícia Universidade Católica do Rio Grande do Sul - Av. Ipiranga, 6681 - Prédio 30, Bloco C, CEP: 90619-900 - Porto Alegre - RS



Autores: Fabiana Zaffalon Ferreira Rocha¹, Ricardo Melo Bastos¹, Marcelo Hideki Yamaguti¹ - {frocha, bastos, yamaguti}@inf.pucrs.br

Abstract. *The software requirements specification (SRS) is decisive for a good final product. The metrics are an essential role for the detection of requirements defects, supplying ways to discrepancies' visualization and the identification of points out of projected situation. Thus, the objective of this paper is to propose a model that evaluates the quality of the translation of the requirements to use cases.*

Resumo. *O documento de especificação de requisitos de software (Software Requirements Specification - SRS) é decisivo para o desenvolvimento de um bom produto final. As métricas desempenham um papel essencial na detecção de defeitos dos requisitos, fornecendo meios para a visualização de discrepâncias e identificação de pontos fora de uma situação projetada.*

Assim, o objetivo deste trabalho é propor um modelo com a finalidade de avaliar a qualidade da tradução de requisitos para casos de uso.

1 Introdução

Determinar e gerenciar os requisitos são atividades essenciais no processo de desenvolvimento de um software, pois uma vez que os requisitos estejam incorretos, inconsistentes ou incompletos, o sistema provavelmente não atenderá às reais necessidades do cliente [Sommerville 2005]. Segundo [Jacobson 1999], um caso de uso é um "documento narrativo que descreve a sequência de eventos de um ator que usa um sistema para completar um processo". Assim, os casos de uso são artefatos de fácil compreensão para expressar os requisitos funcionais de um sistema.

Os requisitos podem ter diferentes aspectos de qualidade, tais como, consistência, não ambigüidade, etc. As métricas são definidas para medir a qualidade total de um requisito. São aplicadas nos requisitos expressos em linguagem natural e em casos de uso através dos atributos de qualidade [Mora 2003] e desempenham um papel essencial na detecção de defeitos dos requisitos, fornecendo meios para a visualização de discrepâncias e identificação de pontos fora de uma situação projetada [Costello 1995]. Partindo de modelos de qualidade, algumas métricas são derivadas a fim de executar uma avaliação quantitativa de uma especificação de requisitos [Fabbrini 2001].

Este trabalho visa identificar os elementos envolvidos em um modelo de qualidade de requisitos, tais como a estrutura do documento de especificação dos requisitos e os atributos de qualidade. A partir disto, é proposto um modelo para avaliar a qualidade da tradução dos requisitos para casos de uso, bem como a correlação existente entre os atributos dos documentos.

2 Objetivos e Justificativa

2.1 Objetivos

O estudo realizado neste trabalho tem como objetivo geral desenvolver um modelo para avaliação da qualidade do mapeamento de requisitos para especificação de casos de uso.

Como objetivos específicos têm-se:

- Aprofundar o estudo teórico sobre especificação de requisitos, métricas e atributos de qualidade de requisitos de software.
- Identificar informações necessárias do SRS para especificação de casos de uso consistentes e de qualidade.
- Identificar os atributos e métricas de qualidade que serão aplicados no mapeamento.
- Aplicar o modelo proposto em documentos de especificação de requisitos e casos de uso correspondentes.

2.2 Justificativa

Existem diversas abordagens para escrever um SRS, porém não se encontrou na literatura estudos envolvendo a geração de casos de uso de qualidade a partir do SRS.

Assim, verifica-se a importância de investigar os principais elementos presentes em um SRS e a correlação entre estes e casos de uso, para que, a partir do SRS, seja possível a geração de casos de uso com qualidade, evitando os problemas decorrentes de um produto implantado a partir de falhas de qualidade nos seus estágios iniciais.

3 Metodologia

O projeto foi realizado por meio de várias etapas, as quais são descritas a seguir:

Etapas 1: realizou-se o levantamento bibliográfico e estudo do referencial .

Etapas 2: foram estudadas as informações presentes no SRS e na especificação de casos de uso. A partir desses estudos, foi criado o modelo de Mapeamento que reúne as informações dos dois documentos.

Etapas 3: os resultados das etapas 1 e 2 foram analisados e identificou-se as informações recomendáveis do SRS para, a partir dele, especificar os casos de uso.

Etapas 4: baseado na etapa 3, foi proposto um modelo para avaliar a qualidade do mapeamento de requisitos para especificação de casos de uso foi proposto.

4. Resultados Obtidos

4.1 Modelo de Mapeamento

Para a identificação dos elementos foi elaborado um modelo conceitual para os elementos que compõem o SRS e outro para os elementos do modelo de caso de uso. A partir da análise destes modelos foi criado um modelo de Mapeamento que apresenta o relacionamento entre os elementos presentes nos modelos conceituais.

A importância do modelo de Mapeamento para este trabalho é permitir a visualização dos relacionamentos entre as classes dos modelos conceituais

permitindo verificar quais os cuidados que devem ser tomados no SRS para que seja possível a especificação de casos de uso de qualidade.

A Tabela 1 mostra o relacionamento entre as classes do modelo conceitual gerado a partir do SRS e da especificação de casos de uso.

Tabela 1 Relacionamento das classes no modelo de Mapeamento [Rocha 2006]

Classes da Especificação de Casos de Uso	Classes do SRS
nome	introduçãoRF
escopoUC	escopoSRS
contexto	funçõesProduto
participante	introduçãoRF
stakeholders	funçõesProduto
preCondições	processamento
garantiasMinimas	funcaoProduto e introducaoRF
garantiasSucesso	introducaoRF e saida
cenarioSucessoPrincipal	entrada e funçõesProduto
fluxoAlternativo	respostaSituaçõesAnormais
variações	processamento e funçõesProduto
informacoesRelacionadas	atributo e performance

Através de uma aplicação constatou-se que a presença de alguns elementos do SRS identificados na Tabela acima não são obrigatórias, tais como: Entrada, Saída, Processamento, Resposta a Situações Anormais, Atributo e Performance, entretanto estas informações contribuem no entendimento de cada requisito bem como na tradução dos mesmos para casos de uso.

Os modelos conceituais do SRS, Modelo de Caso de Uso e Mapeamento foram trabalhos realizados dentro do grupo de Engenharia de Requisitos do Centro de Desenvolvimento e Pesquisas DELL/PUCRS (CDPe).

4.2 Modelo de Avaliação da Qualidade da Transcrição de Requisitos para Casos de Uso

Um dos objetivos do modelo proposto é identificar quais os elementos do SRS são necessários para especificar casos de uso consistentes visando à qualidade na transcrição. Estas informações foram identificadas através do trabalho desenvolvido por [Belgamo 2005] e através da análise do modelo de Mapeamento.

De acordo com [Belgamo 2005], os campos Introdução, Funções do Produto, Requisitos Funcionais e Atributos são suficientes para extrair as informações necessárias para gerar os casos de uso, entretanto conforme o estudo realizado no modelo de Mapeamento e através de uma aplicação verificou-se que outras informações (seção 4.1) também são importantes para facilitar a transcrição.

4.2.1 Atributos de Qualidade do Modelo

Os atributos adotados para o modelo proposto são os mesmos apresentados na literatura, porém com abordagens diferentes. Enquanto que na literatura o foco está na qualidade do documento em si, avaliando cada requisito individualmente, o foco deste trabalho está na qualidade da transcrição dos requisitos para especificação de casos de uso, ou seja, na qualidade da tradução do requisito para caso de uso correspondente.

Na Tabela 2 está a relação dos atributos identificados na literatura que são utilizados no trabalho, bem como suas definições e adaptações para a avaliação da qualidade de tradução do modelo proposto.

Tabela 2 Comparação entre os atributos

Atributo	Literatura [Davis 1993]	Adaptação [Rocha 2006]
Compleitude	Um SRS é completo se descreve todas as demandas de interesse dos usuários; mostradas as respostas esperadas do sistema; se todas as páginas estiverem numeradas; todas as figuras e tabelas estiverem nomeadas e enumeradas; todos os termos definidos e todo material presente referenciado.	A transcrição será completa se todos os elementos presentes no requisito como por exemplo, atores e funcionalidades estiverem também presentes no caso de uso correspondente.
Não ambigüidade	Um SRS é não ambíguo de todos os requisitos presentes no documento tiverem uma única interpretação.	Os casos de uso e os requisitos correspondentes devem ter a mesma interpretação.
Consistência	Um SRS é consistente se não existir contradições entre os requisitos presentes no documento.	Capacidade de não existir contradição entre os casos de uso e os requisitos correspondentes

4.2.2. Métricas

Para guiar a seleção dos elementos que irão compor o modelo de medição, o trabalho seguiu a sequência de passos proposta pelo método GQM.

Na fase de planejamento, foram definidos os atributos de qualidade que serão medidos (seção 4.2.1). As métricas foram adaptadas para avaliar a qualidade da tradução dos requisitos para casos de uso.

4.2.1. Definição

Seguindo o GQM, foram definidos os objetivos de medição. A partir desses objetivos, a sequência de passos proposta pelo método GQM foi executada. Para cada meta de medição foram identificadas as perguntas, com o objetivo de traduzir as metas em aspectos quantitativos que pudessem ser alvos de medição. A partir das questões, derivaram-se então as métricas. A Tabela 3 apresenta o conjunto de métricas para a avaliação da qualidade da tradução de requisitos para casos de uso (U.C.).

Tabela 3 Modelo GQM proposto [Rocha 2006]

OBJETIVO G1	
Propósito:	Avaliar
Aspecto:	Qualidade
Objeto:	Atributo: completude
Ponto de Vista:	Gerente
Questão Q1	Todos os elementos presentes nos requisitos estão presentes nos casos de uso correspondentes?
Métrica M1	$M1 = ((A1 + A2 + \dots + An) + (F1 + F2 + \dots + Fn) + (RNF1 + RNF2 + \dots + RNFn)) / (A + F + RNF)$ Onde: A = total de atores identificados no requisito. F = total de funções encontradas no requisito. RNF = total de requisitos não funcionais (RNF) presentes no requisito. A1 = total de atores mapeados no caso de uso que correspondem aos do requisito. A2 = total de atores do U.C.2 que não estão presentes U.C.1 e que correspondem aos do requisito. An = total de atores do U.C.n que não estão presentes nos casos de uso anteriores e que correspondem aos do requisito. F1 = total de funções encontradas no U.C. 1 que correspondem aos do requisito. F2 = total de funções encontradas no U.C. 2 que não estão presentes U.C.1 e que correspondem aos do requisito.

	requisito. $RNF2$ = total de RNF encontrados no U.C. 2 que não estão presentes U.C.1 e que correspondem aos do requisito. RNF_n = total de RNF encontrados no U.C.n que não estão presentes nos casos de uso anteriores que correspondem aos do requisito. INTERPRETAÇÃO: $M1 = 1$ → tradução completa. $M1 < 1$ → nem todos os elementos presentes no requisito estão mapeados nos casos de uso.
Questão 2	Todos os atores identificados no requisito estão presentes no caso de uso correspondente?
Métrica M2 Quando 1 requisito gera 1 caso de uso	$M2 = A1 / A$ Onde: A = total de atores identificados no requisito. A1 = total de atores identificados no caso de uso que correspondem aos do requisito. INTERPRETAÇÃO: $M2 = 1$ → os atores presentes no requisito estão mapeados no caso de uso correspondente. $M2 < 1$ → existem atores presentes nos requisitos que não estão nos casos de uso.
Métrica M3 Quando 1 requisito gera N casos de uso	$M3 = (A1 + A2 + \dots + A_n) / A$ Onde: A = total de atores identificados no requisito. A1 = total de atores identificados U.C.1 e que correspondem aos do requisito. A2 = total de atores do U.C.2 que não estão presentes U.C. 1 e que correspondem aos do requisito. A_n = total de atores do U.C.n que não estão presentes nos U.C. anteriores e que correspondem aos do requisito. INTERPRETAÇÃO: $M3 = 1$ → todos os atores do requisito estão mapeados nos casos de uso correspondentes. $M3 < 1$ → existem atores presentes nos requisitos que não estão nos casos de uso.
Questão 3	Todas as funções presentes no requisito estão presentes também no caso de uso?
Métrica M4 Quando 1 requisito gera 1 caso de uso	$M4 = F1 / F$ Onde: F = total de funções identificadas nos requisitos. F1 = total de funções encontradas no caso de uso que correspondem aos do requisito. INTERPRETAÇÃO: $M4 = 1$ → o caso de uso apresenta todas as funções do requisito. $M4 < 1$ → nem todas as funções presentes no requisito estão mapeadas no caso de uso.
Métrica M5 Quando 1 requisito gera N casos de uso	$M5 = (F1 + F2 + \dots + F_n) / F$ Onde: F = total de funções encontradas no requisito. F1 = total de funções encontradas no U.C.1 que correspondem aos do requisito. F2 = total de funções encontradas no U.C.2 que não estão presentes no U.C.1 e que correspondem aos do requisito. F_n = total de funções encontradas no U.C.n que não estão presentes nos casos de uso anteriores e que correspondem aos do requisito.

Quando 1 requisito gera N caso de uso	RNF = total requisitos não funcionais (RNF) encontradas no requisito. RNF1 = total de RNF encontrados no U.C.1 que correspondem aos do requisito. RNF2 = total de RNF encontrados no U.C.2 que não estão presentes nos U.C.1 e que correspondem aos do requisito. RNFn = total de RNF encontrados no U.C.n que não estão presentes nos casos de uso anteriores e que correspondem aos do requisito. INTERPRETAÇÃO: M7 = 1 → os casos de uso apresentam todos os RNFs do requisito correspondente. M7 < 1 → nem todos RNFs presentes no requisito estão mapeadas nos casos de uso.
Questão Q5	A tradução dos requisitos para casos de uso está completa?
Métrica M8	$M8 = n_A / n_r$ Onde: n_A = número de requisitos no bloco A. n_r = número de requisitos do SRS. INTERPRETAÇÃO: M8 = 1 → tradução completa. Todos os requisitos possuem pelos menos um caso de uso correspondente. M8 < 1 → tradução incompleta.
OBJETIVO G2	
Propósito:	Avaliar
Aspecto:	Qualidade
Objeto:	Atributo: não ambigüidade
Ponto de Vista:	Gerente
Questão Q6	Os requisitos e casos de uso correspondentes possuem a mesma interpretação?
Métrica M9	$M9 = X$
Quando 1 requisito gera 1 caso de uso	Onde: X = 0 os requisitos e os casos de uso correspondentes são ambíguos. X = 1 os requisitos e os casos de uso correspondentes não são ambíguos.
Métrica M10	$M10 = UC_{un} / UC$ Onde:
Quando 1 requisito gera N casos de usos	UC_{un} = número de casos de uso que tem a mesma interpretação do requisito. UC = total de casos de uso gerados a partir do requisito. INTERPRETAÇÃO: M10 = 1 → o requisito e o caso de uso correspondente têm a mesma interpretação. M10 < 1 → o requisito e o caso de uso correspondente não possuem a mesma interpretação.
OBJETIVO G3	
Propósito:	Avaliar
Aspecto:	Qualidade
Objeto:	Atributo: consistência
Ponto de Vista:	Gerente
Questão Q7	Os requisitos e casos de uso correspondentes estão consistentes entre si?
Métrica M11	$M11 = X$
Quando 1 requisito gera 1 caso de uso	Onde: X = 0 os requisitos e os casos de uso não são consistentes.

A métrica M8 é uma adaptação do modelo proposto por Alexander [Davis 1993], onde os requisitos são distribuídos em quatro blocos dentro de um quadrante. A Figura 1, ilustra a adaptação.

No bloco A, são os requisitos que possuem cobertura completa dos casos de uso, ou seja, possui pelo menos um caso de uso correspondente, e todos os elementos presentes, no requisito, estão nos casos de uso. No bloco B, estão todos os requisitos que possuem casos de uso correspondente, porém nem todos os elementos foram mapeados para os casos de uso. No bloco C, estão os requisitos que não possuem casos de uso correspondente, ou seja, não foram traduzidos para casos de uso. E, finalmente, no bloco D, estão os requisitos desconhecidos, isto é, são os casos de uso gerados sem estarem associados aos requisitos especificados no SRS.

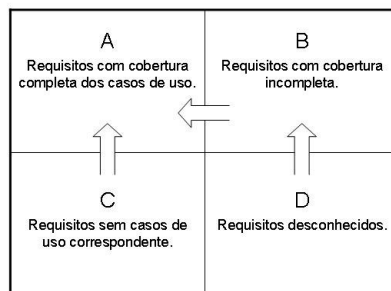


Figura 1 - Adaptação do Modelo Proposto por Alexander [Rocha 2007]

Os resultados obtidos com os estudos se mostraram satisfatórios, já que através das métricas foi possível detectar o foco do problema da tradução dos requisitos para casos de uso permitindo a correção dos mesmos, sem que fosse necessário analisar todos os documentos e todas as partes da tradução, e a descoberta de requisitos omissos no SRS. Através desta aplicação constatamos também a facilidade do uso das métricas, bem como a facilidade de interpretá-las.

Este trabalho resultou em publicação no III Simpósio Brasileiro de Sistemas de Informação e dissertação de mestrado em Ciência da Computação da Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS).

5 Conclusão e Trabalhos Futuros

Especificar requisitos de elevada qualidade não é tarefa trivial. Por isto a necessidade da análise com o objetivo de capturar falhas, isto é, identificar elementos no SRS que estão ausentes nos casos de uso correspondentes. As métricas podem ajudar a avaliar a qualidade do mapeamento de requisitos para casos de uso, capturar os defeitos e erros de requisitos permitindo as mudanças necessárias, bem como descobrir requisitos ausentes no SRS, requisitos que receberam cobertura completa e incompleta dos casos de uso correspondentes e aqueles que não foram mapeados. O que se encontra na literatura é somente a análise do SRS ou dos casos de uso em separado.

Através do modelo é possível descobrir os requisitos omissos no SRS por pessoas que não estejam envolvidas no processo de desenvolvimento do software, aplicando a métrica que analisa a completude do mapeamento. Também, pode-se citar como contribuição a não obrigatoriedade de quem preenche o SRS preencher também o documento de especificação de casos de uso, ou seja, não há uma dependência de conhecimento de quem desenvolveu o SRS para descrever os casos de uso, uma vez que o SRS esteja completo, pois o modelo dá

as diretrizes de onde as informações, para especificar os casos de uso, são encontradas no SRS.

O trabalho apresentado neste relatório é de um modelo que permite a avaliação da qualidade da tradução de requisitos para casos de uso, com o objetivo de gerar documentos na fase de especificação de um sistema (SRS e casos de uso) com qualidade e consistência, bem como identificar a correlação existente entre eles. Tal aspecto contribui para a qualidade do produto final do software evitando os custos decorrentes de um produto implantado a partir de falhas de qualidade nos seus estágios iniciais.

Atualmente, as métricas estão sendo aplicadas em SRS de sistemas de maior complexidade para avaliar o modelo e as métricas, refinando a proposta. Os SRS utilizados são documentos de projetos desenvolvidos em uma grande empresa de desenvolvimento de software com atuação internacional.

Como trabalhos futuros prevêem-se o estudo e a adaptação de novos atributos e métricas que possam ser aplicados na análise da tradução, por exemplo, atributo correção. Isso permitiria uma análise mais completa em relação à transcrição. Identificase também como trabalho futuro, a aplicação do modelo em projetos de maior complexidade, com a finalidade de comparar os resultados do modelo proposto.

Finalmente, a criação de uma ferramenta de apoio para a aplicação do modelo.

Referências Bibliográficas

Belgamo, A. and Fabbri, S. (2005) "TUCCA: Técnica de Leitura para apoiar a Construção de Modelos de Casos de uso e a Análise de Documentos de Requisitos".

In: XIX Simpósio Brasileiro de Engenharia de Software. Brasil. Costello, R. and Liu, D. (1995) "Metrics for Requirements Engineering". *Journal of Systems Software*, pp. 39-63.

Davis, A., et al. (1993) "Identifying and Measuring Quality in a Software Requirements Specification." In: *Proceedings of the 1st International Software Metrics Symposium*, pages. 141–152, Baltimore.

Fabbrini, F., et al. (2001) "An automatic quality evaluation for natural language requirements". In: *Proceedings of the 7th International Workshop on Requirements Engineering: Foundation for Software Quality REFSQ'01*.

IEEE Std 830-1998. (1998) "IEEE Recommended Practice for Software Requirements Specifications". New York, USA: The Institute of Electrical and Electronics Engineers.

Jacobson, I.; Booch, G. and Rumbaugh, J. (1999) "The unified software development process", Addison-Wesley, 463 p.

Mora, M. and Denger, C. (2003) "Requirements Metrics - An initial literature survey on measurement approaches for requirement specifications". Fraunhofer Institut Experimentelles Software Engineering, IESE-Report No. 096.03/E.

Rocha, F.; Cerri, E.; Bastos, R.; Yamaguti, M. (2006) "Uma Proposta de Modelo para Avaliar a Qualidade da Tradução de Requisitos para Casos de Uso". In: *III Simpósio Brasileiro de Sistemas de Informação – III SBSI*. Curitiba / PR.

Rocha, F.; Bastos, R. (2007) "Modelo para Avaliar a Qualidade da Tradução de Requisitos para Casos de Uso". Dissertação de Mestrado. Programa de Pós-Graduação em Ciência da Computação. Pontifícia Universidade Católica do Rio Grande do Sul – PUCRS. Porto Alegre, RS. 95 p.

Sommerville, I. (2005) "Integrated Requirements Engineering: A Tutorial". IEEE Software, pp. 16-23.

Estudo realizado pelo Grupo de Pesquisa em Engenharia de Requisitos do PDTI, financiado pela Dell Computadores do Brasil Ltda. com recursos da Lei 8.248/91.

[6.09] Ferramenta Web para Implantação do Personal Software Process

Entidade: Instituto Centro-Oeste de Desenvolvimento de Software (ICODES); Rua Herculano Lobo, 206, Sala 01 - 73801-260 - Formosa - GO - Brasil

Autor: Cristiano Lehrer - cristiano@ICODES.org.br

Resumo. *O presente artigo descreve o desenvolvimento e implantação do Personal Software Process (PSP), um processo disciplinado de auto-melhoria, nas dependências do Instituto Centro-Oeste de Desenvolvimento de Software (ICODES), fornecendo aos seus colaboradores um modo para melhorar a qualidade, a predictibilidade e a produtividade do seu trabalho.*

1. Introdução

Atualmente, qualidade é um dos assuntos mais importantes na indústria de software, e provavelmente será ainda mais importante num futuro próximo, pois há cada vez mais sistemas controlados por software, fazendo com que a economia de praticamente todos os países seja muito dependente da qualidade dos software por eles usados [Vasconcelos, 2003].

Entretanto, a qualidade dos produtos de software está fortemente relacionada à qualidade do processo de software. A preocupação com o processo de software está relacionada à necessidade de entender, avaliar, controlar, aprender, comunicar, melhorar, prever e certificar o trabalho dos colaboradores da área de software [Rocha, 2001].

Muitas organizações têm alcançado melhorias significativas nos seus processos e modos de trabalho através da aplicação de um modelo de gerenciamento de processo de software, entretanto, perceberam que para obterem índices melhores dependiam, ainda, do talento individual de seus colaboradores, de forma que estudos e pesquisas direcionados para a capacitação e melhoria do processo individual fossem realizadas [Albuquerque, 2004].

O presente artigo apresenta o estudo de caso da implantação, no Instituto Centro-Oeste de Desenvolvimento de Software, do processo pessoal *Personal Software*



Process (PSP), projetado para resolver as necessidades de melhoria de colaboradores individuais e de organizações produtoras de software de pequeno porte.

2. Objetivos e Justificativa

O *Personal Software Process* (PSP), traduzido como Processo Pessoal de Software, foi desenvolvido por Watts Humphrey em 1989, apoiado pelo *Software Engineering Institute* (SEI). Humphrey é mundialmente conhecido por desenvolver o processo do *Capability Maturity Model* (CMM) e o *Team Software Process* (TSP) [Koscianski, 2006].

O PSP é um *framework* de formulários, diretrizes e procedimentos de desenvolvimento de software. Corretamente usado, o PSP provê os dados históricos que o engenheiro de software precisa para fazer e conhecer melhor seus compromissos, tornar os elementos rotineiros de seu trabalho previsíveis e mais eficientes, além de possibilitar a medição da qualidade dos produtos gerados [Albuquerque, 2004].

O modelo proposto por Humphrey concentra-se no projeto, codificação e fases de testes do desenvolvimento de software. Porém, o PSP não se aplica somente a eles, mas as outras fases do processo de desenvolvimento de software, inclusive na especificação de requisitos, manutenção de produto, planejamento de testes e desenvolvimento de documentação [Tavares, 2006].

O modelo do PSP consiste de quatro etapas, sendo os três primeiros subdivididos em dois níveis, totalizando sete níveis. Na etapa do PSP 0, é medido o tempo gasto em cada uma das fases do desenvolvimento, os defeitos inseridos e os encontrados, para cada fase. O modelo apresenta tabelas cujos campos fornecem dados para compor os relatórios [Albuquerque, 2004].

Dentro do PSP 0 existe o PSP 0.1, cujo objetivo é formar um consenso entre os desenvolvedores, para a adoção de um padrão de escrita para os programas, de modo a uniformizar os programas produzidos por uma equipe [Albuquerque, 2004].

O PSP é um caminho rápido, eficaz e de qualidade assegurada para se alcançar os níveis de maturidade exigidos pela maioria dos modelos e normas de qualidade utilizados no mercado, como o CMMI (*Capability Maturity Model Integration*) e o Modelo de Referência da Melhoria de Processo do Software Brasileiro (MPS.BR), auxiliando as organizações a melhorarem sua maturidade no desenvolvimento de software [Koscianski, 2006].

3. Metodologia de Execução

Inicialmente, o *Personal Software Process* foi implantando entre os colaboradores do Instituto Centro-Oeste de Desenvolvimento de Software de forma manual, sem o auxílio de sistemas computacionais, utilizando-se exclusivamente planilhas eletrônicas.

As informações pertinentes ao *Personal Software Process* foram coletadas através de planilhas, conforme exibido na tabela 01, onde cada colaborador registra as

tarefas realizadas no decorrer do dia, bem como o esforço (em horas) despendido para a realização das mesmas.

Tabela 01 – Planilha de coleta de informações para o PSP

<i>Data</i>	<i>Início</i>	<i>Atividade</i>	<i>Projeto</i>	<i>Descrição</i>	<i>Concluído</i>	<i>Interrupção</i>	<i>Fim</i>
02	08:00	1.1.1	OBID06	Apresentação do serviço	Sim	00:00	09:30
02	09:30	1.1.2	OBID06	Análise do serviço	Sim	00:05	10:00
02	10:00	1.1.3	OBID06	Elaboração do Termo de Abertura	Sim	00:15	12:00
02	12:00	1.1.4	OBID06	Elaboração da Proposta Técnica	Não	00:20	18:00
02	08:00	1.1.4	OBID06	Elaboração da Proposta Técnica	Sim	00:15	12:00

A seguir, as informações provenientes das planilhas eletrônicas dos colaboradores são consolidadas, visando serem utilizadas pela organização na elaboração de estimativas confiáveis e reais para o planejamento do esforço e do custo dos produtos de trabalho e tarefas a serem executados nos novos projetos. A tabela 02 exibe o esboço da planilha consolidada.

Tabela 02 – Consolidação das informações do PSP

Atividade	OBID06	GPR06	GRE06	MÉDIA
1. Fase de Concepção				
1.1. Solicitação de Serviço	5,0	7,9	3,6	5,5
1.1.1. Reunião com o cliente	1,5	2,5	1,1	1,7
1.1.2. Avaliação da solicitação	0,5	0,9	0,4	0,6
1.1.3. Elaboração do Termo de Abertura	3,0	4,5	2,1	3,2

Uma vez que o *Personal Software Process* estava estabelecido, iniciou-se a implementação de um sistema Web, desenvolvido sobre a plataforma J2EE (*Java 2 Enterprise Edition*), para automatizar o registro e a consolidação das informações provenientes de todos os colaboradores.

Dessa forma, o próprio desenvolvimento da ferramenta Web para o registro do *Personal Software Process* foi registrado pela metodologia, contribuindo para o levantamento de requisitos das necessidades requeridas pelo Instituto Centro-Oeste de Desenvolvimento de Software.

4. Resultados Obtidos

Desenvolvimento de um sistema Web para auxiliar na implantação do *Personal Software Process* dentro das dependências do Instituto Centro-Oeste de

Desenvolvimento de Software, automatizando o registro e a geração dos relatórios requeridos pelo processo individual adotado.

O foco do projeto foi a automatização dos dois primeiros níveis do PSP, conhecidos como PSP 0 e PSP 0.1, onde são medidos apenas o tempo gasto pelo desenvolvedor em cada fase, bem como estimula a adoção de um padrão para a escrita dos programas produzidos pelos colaboradores.

Inicialmente, os colaboradores relutaram na adoção dessa metodologia, pela dificuldade de alterarem seus hábitos de desenvolvimento e por questionarem o real benefício que essa metodologia poderia agregar ao trabalho desenvolvido pelos mesmos.

Essas dificuldades foram sendo solucionadas com o tempo, com treinamentos e conversas informais, através da conscientização dos colaboradores dos benefícios que a metodologia trazia para eles e para a organização.

O Instituto Centro-Oeste de Desenvolvimento de Software espera com a implantação do *Personal Software Process*, que a coleta e análise das informações relativas aos produtos desenvolvidos e aos processos implementados na organização em seus projetos, permitam um planejamento mais objetivo através de estimativas confiáveis e reais.

5. Aplicabilidade dos Resultados

A implantação do *Personal Software Process* é estratégico para o Instituto Centro-Oeste de Desenvolvimento de Software, contribuindo para que o Instituto possa obter a certificação do seu processo de desenvolvimento, no modelo de referência MPS.BR, para o nível G (Parcialmente Gerenciado), além de contribuir para o aumento da produtividade dos serviços prestados, colaborando para a sustentabilidade financeira e econômica da organização.

A utilização da ferramenta abrirá novas perspectivas para o gerenciamento das atividades desempenhadas pelos colaboradores do Instituto Centro-Oeste de Desenvolvimento de Software, objetivando uma maior visibilidade gerencial das atividades e desempenho de cada colaborador, permitindo uma aferição mais precisa dos serviços executados, além de permitir o levantamento das necessidades de treinamento a serem ministrados para os colaboradores, a fim de corrigir as deficiências observadas.

A ferramenta implementada será inicialmente utilizada no Instituto Centro-Oeste de Desenvolvimento de Software, para posteriormente ser disponibilizada para as demais organizações associadas ao Instituto e para o público em geral, que poderão fazer o *download* da ferramenta diretamente do *site* do Instituto.

6. Características Inovadoras

Apesar de ter sido originalmente desenvolvido em 1989 por Watts Humphrey, a utilização do *Personal Software Process* nas empresas brasileiras ainda é muito remota, motivada principalmente pelo desconhecimento dos desenvolvedores quanto a sua utilização e a falta de ferramentas computacionais que auxiliem na sua utilização.

O Instituto Centro-Oeste de Desenvolvimento de Software almeja facilitar e agilizar o preenchimento dos formulários requeridos pelo *Personal Software Process*, de forma que o desenvolvedor não desperdice tempo preenchendo os formulários, além de automatizar as tarefas de consolidação dos registros, fornecendo um retorno mais rápido e preciso sobre o desempenho do desenvolvedor.

7. Conclusão e Perspectivas Futuras

A implantação do *Personal Software Process* exige um esforço considerável da organização e, principalmente, dos seus colaboradores, não apenas ao executar o trabalho e ao estudar os próprios métodos, mas na interferência com as práticas atuais.

A utilização do *Personal Software Process* pelo Instituto Centro-Oeste de Desenvolvimento de Software contribuiu para a melhora das estimativas utilizadas para a elaboração do Plano de Projeto, melhorando o planejamento e o acompanhamento dos cronogramas.

O Instituto Centro-Oeste de Desenvolvimento de Software pretende adequar a ferramenta desenvolvida para suportar as outras etapas do *Personal Software Process*, buscando sempre alcançar níveis maiores de qualidade.

8. Referências Bibliográficas

Albuquerque, Jones de Oliveira. (2004). "Gerencia de Processo de Software em Pequena Escala: ênfase em PSP, TSP e P-CMM". Lavras: UFLA/FAEPE. 113p.

Koscianski, André; Soares, Michel dos Santos. (2006). "Qualidade de Software". São Paulo: Novatec. 395p.

Rocha, Ana Regina Cavalcanti da. (2001). "Qualidade de Software". São Paulo: Prentice Hall. 303p.

Tavares, Pablo. (2006). "PSP Home". Disponível em:

<<http://www.psphome.hpg.ig.com.br>>. Aceeso em: jun. 2006.

Vasconcelos, Alexandre Marcos Lins de; Maciel, Teresa Maria de Medeiros. (2003).

"Introdução à engenharia de software e aos princípios de qualidade". Lavras: UFLA/FAEPE. 104p.

[6.11] Incrementando produtividade através de reuso de componentes



Entidade: Atech - Fundação Tecnologias Aplicações Críticas - Rua do Rócio, 313 / 11º andar - Vila Olímpia ; São Paulo - SP - 04552-000 - Brasil

Autor: Claudia A. Tocantins - claudia@atech.br

Abstract. *This article describes and reports the results of the [6.11 – Improving productivity through component reuse] project as proposed to the 2006 cycle of PBQP-SW.*

Resumo. *Esse artigo descreve e reporta os principais resultados do projeto [6.11 Incrementando produtividade através de reuso de componentes] submetido ao ciclo 2006 do PBQP-SW.*

1. Introdução

O projeto descrito nesse artigo visou o desenvolvimento de uma Biblioteca de Componentes que surgiu do próprio amadurecimento da equipe de desenvolvimento na tentativa de incorporar e/ou reutilizar pedaços de software em projetos sucesivos. O projeto institucionalizou a iniciativa na forma de um projeto específico que foi desenvolvido como um projeto de software independente.

O artigo reúne dados publicados na submissão do projeto e dados apresentados no EQPS Belém em novembro de 2006. A seção 2 indica os objetivos e justificativa do projeto; a seção 3 explica como o projeto foi desenvolvido; a seção 4 descreve o principal produto e resultados relevantes obtidos; a seção 5 explica a aplicabilidade dos resultados em termos de relevância, abrangência e impacto dos resultados obtidos; a seção 6 indica características inovadoras e a seção 7 conclui o artigo, apresentando perspectivas futuras.

2. Objetivos e Justificativa

O objetivo desse projeto, nomeado AtechCommons, foi consolidar uma biblioteca de componentes e modelos que forneça tanto ferramentas adicionais para incrementar a produtividade, quanto elementos que direcionem o processo de desenvolvimento, incluindo práticas como:

- a criação e execução de testes unitários
- a geração automática de código redundante integradas ao processo de construção de sistemas (arquivos XML, interfaces de negócio, etc., tipicamente presentes no ambiente J2EE)
- integrar o resultado dos testes ao processo de *build* para examinação de relatórios e construção condicional (baseada nos resultados de testes) de versões de software

O projeto se justifica pela análise das consequências diretas de sua utilização no ambiente de desenvolvimento, tais como: (1) promover a adoção de métodos orientados a objetos para o desenvolvimento de software, garantindo reuso e facilidades de manutenção; (2) visar a diminuição de defeitos nos softwares produzidos, a partir do incentivo ao desenvolvimento e execução de testes unitários como parte do processo; (3) estímulo direto ao desenvolvimento para a Internet, através do fornecimento de vários componentes específicos.

Em outras palavras, a experiência diversa em alguns projetos envolvendo tecnologias e contexto semelhantes (Linguagem Java, ambiente web) demonstrou que existiam problemas bastante parecidos no ambiente de desenvolvimento, como:

- Diferença muito grande na produtividade de programadores de níveis diferentes (jr., pleno e senior) – indução de práticas erradas para programadores menos experientes
- Falta de padronizações na produção do código e bibliotecas utilizadas
- Soluções diferentes para problemas semelhantes – maior ou menor performance dependendo da solução
- Muita repetição de código e pouca reutilização de soluções que já tinham sido desenvolvidas

A resolução destes problemas com a consolidação de uma biblioteca de componentes e práticas agregadas teria como principais justificativas:

- Promover a adoção de métodos orientados a objeto para o desenvolvimento, facilitando o reuso e manutenção de forma geral
- Visar a diminuição de defeitos nos softwares através da indução à construção de testes unitários
- Diminuir o esforço para construção de artefatos de software, através de uma série de soluções já desenvolvidas e consolidadas em diferentes projetos

3. Metodologia de Execução

O desenvolvimento da Biblioteca de Componentes surgiu do próprio amadurecimento da equipe de desenvolvimento que tentava incorporar e/ou reutilizar pedaços de software em projetos sucesivos. O projeto institucionalizou a iniciativa na forma de um projeto específico que foi desenvolvido como um projeto de software independente (porém usando recurso humano compartilhado com outros projetos). As seguintes atividades foram desenvolvidas:

- Definição dos requisitos reais
 - Necessidades atuais dos projetos, e futuras áreas de abrangência dos componentes
- Levantamento concreto das soluções e problemas existentes
 - Código repetitivo
 - Ausência ou escassez de testes
 - Ineficiência de soluções
- Definição do domínio de aplicações beneficiadas pelo projeto

- Tipos de aplicação que podem se beneficiar da utilização desta biblioteca (ambiente, linguagem, sistema operacional, etc.)
- Construção efetiva dos componentes
 - Reaproveitando o código existente e agrupando os artefatos produzidos em diferentes projetos, acrescentando funcionalidades e validando a qualidade final
- Estabelecimento das práticas subseqüentes à utilização dos componentes

A partir desse ponto, componentes adicionais desenvolvidos no âmbito de um projeto, mas que se encontrava dentro da linha de requisitos da Biblioteca de Componentes passou a ser desenvolvida para a Biblioteca e automaticamente disponibilizada para reuso.

4. Resultados Obtidos

O produto gerado é um framework Java que inclui uma série de componentes de infra-estrutura e manipulação de objetos reutilizáveis, além de incorporar melhores práticas de projeto e codificação. O framework faz uso extenso de bibliotecas e componentes livres, além de incorporar código reutilizável desenvolvida em projetos históricos e/ou correntes da organização.

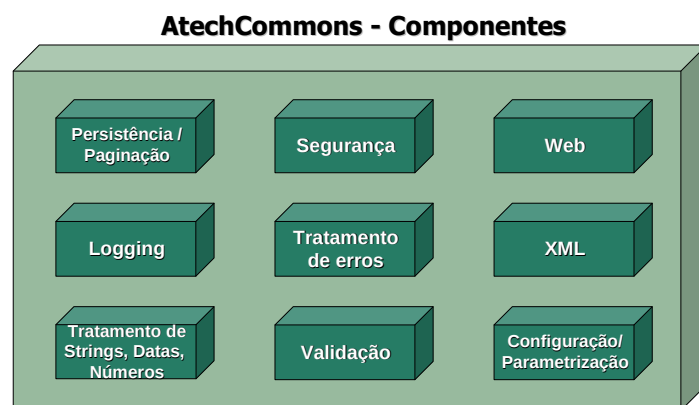


Figura 1. Principais Componentes

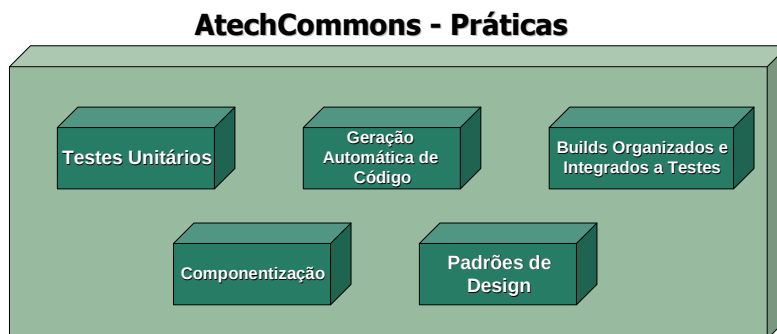


Figura 2. Práticas incorporadas ao Framework

5. Aplicabilidade dos Resultados

O projeto é relevante pela consolidação do uso de técnicas de Engenharia de Software, uma meta da organização, visando principalmente o aumento da qualidade e também produtividade. Ao empregar técnicas avançadas de orientação a objetos, como a aplicação de padrões de design com boa documentação, permite um alto nível de reaproveitamento, modularização e manutenibilidade, facilitando também sua extensibilidade. Em adição, tem como premissa a aplicação de práticas de processo tais como o desenvolvimento e execução de testes unitários e a geração de código automática, possibilitando um incremento na qualidade e na produtividade.

Em termos de impacto, a utilização do framework implica na manutenção de um repositório único de práticas e componentes, capaz de incrementar a qualidade e produtividade de equipes envolvidas em projetos distintos, ao estabelecer o framework como base para o processo de desenvolvimento para variados contextos de aplicação. O impacto direto do emprego do framework é uma menor curva de aprendizado para os membros da equipe, assim como a utilização de componentes bem testados e já utilizados em outros projetos. Sua manutenção e evolução garantem benefícios a diversos projetos em paralelo.

A abrangência do projeto, que teve origem em práticas e artefatos aplicados e desenvolvidos ao longo de vários projetos, já nasceu em um contexto de multidisciplinaridade. A utilização do mesmo já alcança vários setores da empresa, e tem crescido gradualmente através da adoção em novos projetos, assim como no interesse de integração a outros projetos já desenvolvidos na empresa. Por

utilizar ferramentas de código livre, o projeto se beneficia da amplitude das informações disponíveis para tais ferramentas.

6. Características Inovadoras

O projeto faz uso de diversas ferramentas complexas de código livre em um nível de abstração mais alto, através de componentes específicos, facilitando assim a adoção de tais ferramentas por equipes menos experientes e a manutenção de software que compartilhem das mesmas ferramentas. Trabalhou-se com o aprimoramento de padrões de design bem difundidos na comunidade.

Em adição, através da extensão de várias destas ferramentas, o projeto originou novas funcionalidades e características não encontradas anteriormente, e que podem ser exploradas com ganho de produtividade.

7. Conclusão e Perspectivas Futuras

O projeto “Incrementando produtividade através de reuso de componentes” foi concluído com sucesso.

A evolução da biblioteca (e práticas associadas) é continua, e é realimentada com a experiência de novos projetos. Obteve-se parte da evolução dos sistemas que fazem uso da biblioteca como consequência da evolução da própria biblioteca. Além disso, os sistemas se tornaram mais seguros pela utilização de mecanismos eficientes e bem testados de segurança (autenticação, autorização e validação) e mais rápidos, pela utilização de componentes otimizados em substituição às soluções pontuais com menor grau de otimização.

A medida que o uso do framework é difundido e experimentado, mais pessoas se capacitam a otimizar o código comum, assim como a treinar novos programadores de maneira rápida e produtiva. Observou-se uma maior performance para equipes inexperientes após a consolidação e adoção da biblioteca como consequência de uma maior padronização do código produzido, uma menor quantidade de erros, e uma maior rapidez na conclusão das tarefas.

Novos escopos e componentes estão sendo buscados, inclusive a integração com outras soluções dentro da organização e áreas e experiências distintas das convencionais

[6.12] Laboratório de Engenharia de Software e Inteligência Artificial: Construção do ambiente WebAPSEE

Entidade: 1Laboratório de Engenharia de Software (LABES) - Instituto de Ciências Exatas Universidade Federal do Pará (UFPA) - Belém - Pará - 66075-110

Autores: Carla Alessandra Lima Reis¹, Rodrigo Quites Reis¹ -
<http://www.labes.ufpa.br> - clima@ufpa.br, quites@computer.org

Abstract. *This paper describes the evolution of the WebAPSEE software system during 2006. WebAPSEE is an open source process-centered software engineering environment developed by the Software Engineering Lab at the Federal University of Pará (Brazil) with the cooperation of SERPRO (Belém Office), Eletronorte and*

Universität Stuttgart (Germany). This text emphasizes the description of the adopted methodological strategies, the results from the actual use of the system in the industry, while discusses current and future enhancements for the system.

Resumo. Este artigo descreve a evolução do WebAPSEE, ambiente de gestão de processo de software desenvolvido como Software Livre pelo Laboratório de Engenharia de Software da Universidade Federal do Pará com participação do SERPRO-Belém, Eletronorte e Universität Stuttgart – Alemanha. O texto enfatiza a descrição dos aspectos metodológicos adotados, os resultados obtidos com o uso do ambiente, e discute as perspectivas para o avanço atual e futuro.

1. Introdução

Ambientes para automação de processos de software [Gimenes 1994], aqui genericamente denominados PSEEs - *Process-Centered Software Engineering Environments* possibilitam a coordenação das atividades de equipes de desenvolvimento de software, realizando o acompanhamento dos prazos e consumo de recursos, além de facilitar a reutilização de boas práticas gerenciais a partir de diferentes projetos já concluídos. As soluções nesta área devem considerar as especificidades deste contexto, tais como: o caráter criativo do processo, a tendência a mudanças nos produtos e processos, a natureza abstrata do produto resultante (software), dentre outros aspectos.

O ambiente WebAPSEE, apresentado neste artigo, é um PSEE com desenvolvimento liderado pelo Laboratório de Engenharia de Software da Universidade Federal do Pará (LABES-UFPa) que visa prover simultaneamente automação e flexibilidade em níveis não alcançados por outros PSEEs propostos pela Academia e Indústria. Este ambiente destaca-se ainda pelo uso de ferramentas de Software Livre e padrões abertos que foram adotados, variando desde a linguagem de programação (Java), passando pelos *frameworks* para desenvolvimento de componentes, e chegando ao sistema de gerenciamento de banco de dados e ferramentas externas integradas.

Este artigo apresenta os avanços obtidos no ano de 2006 com respeito à evolução da ferramenta WebAPSEE principalmente no que diz respeito à incorporação de funcionalidades voltadas à gestão organizacional e apoio à tomada de decisão. Além disso, é feita uma avaliação inicial do uso da ferramenta em organizações de desenvolvimento de software localizadas no estado do Pará, enfatizando também questões relacionadas com as demandas de melhorias por parte dos usuários.

2. Objetivos e Justificativa

O desenvolvimento do WebAPSEE é baseado em três premissas básicas, descritas a seguir, e expostas em maiores detalhes por Lima Reis (2003):

- O apoio automatizado à **execução de processos** constitui um elemento fundamental para transformar os modelos de processo em realidade nas organizações de software de software possuindo, assim, enorme potencial no auxílio à implantação de processos;
- O aumento no nível de **flexibilidade** fornecido pelas ferramentas no acompanhamento da execução de processos é necessário, pois o

processo de software é inerentemente dinâmico. Os modelos de processo não evoluem de maneira determinística, o que implica na impossibilidade de antever todas as características dos modelos de processo e produtos de software resultantes;

- A elevação do nível de automação fornecido pelas ferramentas (e, por conseguinte, do apoio à tomada de decisão gerencial) é proporcional à **riqueza das informações (métricas e decisões) registradas** pelo ambiente acerca da realização prática dos processos na organização.

O desenvolvimento e evolução do ambiente WebAPSEE são motivados pela inexistência de produtos que atendam simultaneamente a todas as premissas listadas acima. O ano de 2006 foi decisivo para evolução do ambiente e início do seu uso por parte da indústria. As seções a seguir descrevem a evolução do projeto.

3. Metodologia de Execução

O início do desenvolvimento do WebAPSEE foi em 2005 com financiamento da FINEP¹ em um projeto em parceria com a regional Belém do SERPRO. Uma primeira versão do ambiente – denominada versão zero - foi desenvolvida e disponibilizada neste período, a qual fornecia funcionalidades para modelagem e execução de processos acessíveis através de serviços Web.

Em 2006, o projeto caminhou em duas direções complementares: 1) a evolução da ferramenta como **produto**, visando melhorias na usabilidade e performance geral do sistema, e 2) a evolução do WebAPSEE como **pesquisa**, visando a incorporação de melhorias para gestão que contribuam para que o ambiente atinja o estado da arte tecnológico (com apoio do CNPq²). O desenvolvimento contou ainda com o apoio de um projeto financiado pela Eletronorte³ com o objetivo de evoluir a ferramenta e utilizá-la como instrumento para a descrição e experimentação de um modelo de processo de software para ser implantado pelo Centro de Tecnologia da empresa.⁹

O projeto evoluiu em três versões de distribuição restrita (interna), até que no dia 2 de outubro de 2006 a versão 1.0 do produto foi disponibilizada publicamente para a comunidade de Software Livre. Esta versão foi desenvolvida a partir do levantamento das demandas tanto das empresas formalmente vinculadas com o LABES quanto da comunidade externa. Por exemplo, a integração do WebAPSEE com o sistema *Concurrent Version System* (CVS) foi baseada nesta estratégia.

4. Resultados Obtidos

As sub-seções a seguir apresentam os produtos e resultados relevantes obtidos em 2006.

⁹ ¹ Edital MCT/FINEP/Software Livre/2004

² Projeto LABES-IA apoiado pelo edital PDPG-TI/CNPq.

³ Projeto número 50528 apoiado no ciclo 2004/2005 do Programa de P&D da Eletronorte.

4.1. Produtos de software gerados

O principal produto gerado por este projeto é a versão 1.2 do sistema WebAPSEE, a qual está disponível para a comunidade de Software Livre no endereço do LABESUFPA.

A ferramenta possui duas visões principais: a do gerente e a do desenvolvedor.

A figura 1 mostra a visão do gerente (*Manager Console*) como parte mais ampla da figura e a visão do desenvolvedor (*Task Agenda*) na parte inferior direita da imagem.

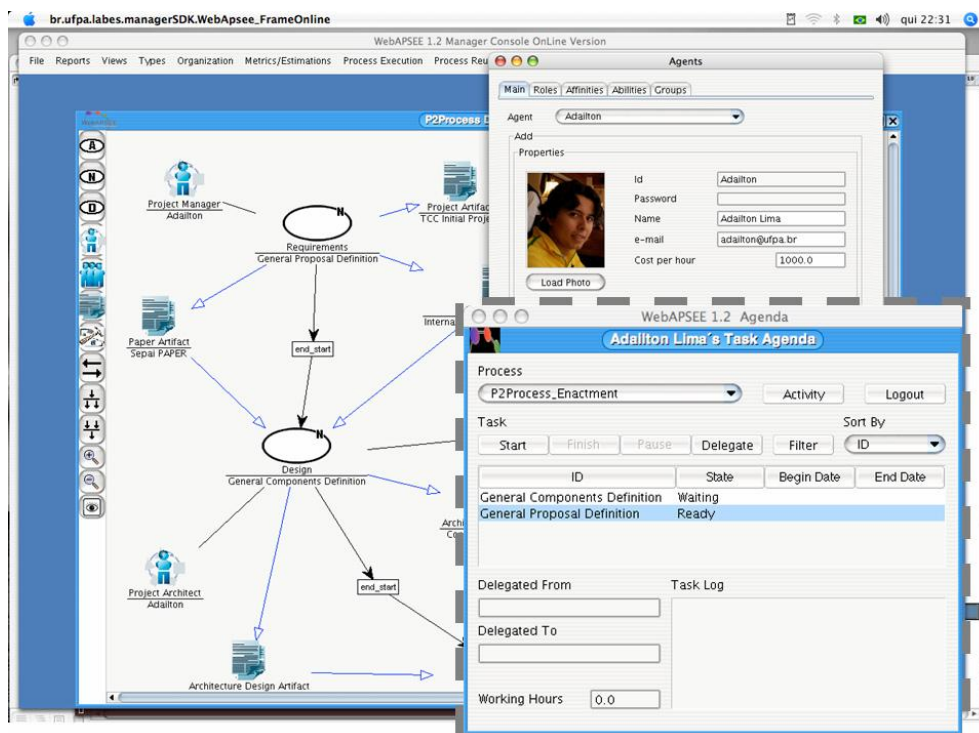


Figura 1 Tela principal do *ManagerConsole* e *TaskAgenda*

O fato de permitir a execução de processos de software significa que o ambiente fornece acompanhamento em tempo real do progresso das atividades e da produção dos artefatos do processo. As atividades - representadas por elipses no editor gráfico de processos mostrado na figura 1 - constituem os elementos principais de modelagem. Ao redor das atividades são definidos os artefatos (de entrada/saída), recursos humanos e de apoio e políticas de processo. As atividades têm suas cores (estados) alteradas na tela do gerente em resposta às ações realizadas pelos desenvolvedores em suas agendas.

Os elementos de software que compõem o WebAPSEE são desenvolvidos para atender prioritariamente as demandas relacionadas com os níveis de maturidade propostos pelo CMMi e MPS-BR. Como uma discussão detalhada acerca da

aderência do WebAPSEE às inúmeras práticas exigidas por estes modelos de referência foge do escopo deste texto¹⁰⁴, o trecho que segue fornece uma descrição das funcionalidades vinculadas à **gestão de configuração de software** e **gerência de projeto** exigidas, respectivamente, pelos níveis F e G do MPS-BR.

O WebAPSEE fornece uma maneira inovadora para implantar **gestão de configuração de software** visto que possibilita o controle de todos os artefatos do processo ocultando a complexidade dessa atividade. Tal funcionalidade é visualizada pelo desenvolvedor como operações de *upload* e *download* de artefatos, respectivamente associadas ao prévio planejamento da sua produção e consumo por uma atividade. Ressalta-se que os desenvolvedores só manipulam os artefatos definidos nas atividades, obedecendo ao controle de acesso previsto nos projetos da organização.

Assim, toda a interação com o CVS (ou semelhante) é escondida do usuário.

Quanto à **gestão de projetos**, o MPS-BR estabelece como um dos resultados da sua aplicação que “os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo” (Softex, 2007). Tomando o ponto de vista específico da alocação de recursos humanos, o WebAPSEE possui detalhado registro das informações organizacionais. Além disso, é fornecido um mecanismo automático para alocação de pessoas baseado em Políticas (Silva, 2007), o qual ilustra o nível de automação fornecido pelo ambiente no auxílio à tomada de decisão gerencial.

A linguagem de definição de Políticas permite a definição de estratégias de alocação genéricas as quais podem ser reusadas em diferentes processos. Além disso, as políticas representam uma extensão da linguagem de modelagem do WebAPSEE, fornecendo uma estrutura Orientada a Aspectos, transversal ao modelo de processo.

Funcionalidades adicionais, não detalhadas aqui por limitação de espaço, incluem; 1) apoio à recuperação e reutilização de processos (Sales, 2006); 2) coordenação de processos de *outsourcing* a partir da execução descentralizada (Lima, 2007); 3) extração de relatórios gerenciais para auxiliar na avaliação da qualidade de software (Paxiúba, 2007); 4) Políticas para alocação de recursos de apoio (análogas às de alocação de pessoas); 5) Apoio à Medição/Análise de Projetos (Nascimento, 2006) e 6) Modelo de Simulação de Processo de Software (França, 2007).

4.2. Artigos publicados

Este projeto originou seis artigos em conferências nacionais em 2006, listados a seguir.

Lima (2006A) e (2006B) são artigos voltados à descrição arquitetural do ambiente.

¹⁰⁴ Uma avaliação da aderência do WebAPSEE ao MPS-BR é disponível em Covre [2007].

Lima (2006C), premiado como Melhor Artigo Científico da XX Semana Paraense de Informática (SEPAI), fornece uma análise detalhada do ambiente com respeito às práticas de gestão de projetos. Sales (2006) descreve o mecanismo de recuperação e reutilização de processos. Nascimento (2006) fornece uma visão geral de uma metodologia apoiada pelo WebAPSEE para guiar atividades de medição de projetos e sua análise. Por fim, Paxiúba (2006) descreve o uso de conhecimento organizacional aplicado à simulação e melhoria de processos.

4.3. Recursos humanos capacitados

O WebAPSEE foi assunto principal de cinco trabalhos de conclusão do Curso de Bacharelado em Ciência da Computação da UFPA em 2006. Além disso, o projeto contou com a participação de quinze bolsistas (entre Iniciação Científica e de Mestrado) os quais atuaram em atividades do projeto e implementação de ferramentas. Tais alunos têm tido a oportunidade de participar em um projeto de software de alta complexidade, o qual utiliza diversas técnicas de Engenharia de Software e paradigmas inovadores de Programação de Computadores no desenvolvimento do núcleo principal e extensões do WebAPSEE. Além disso, seis destes alunos se especializaram em tarefas diversas envolvendo: o suporte técnico à instalação e uso do produto; a confecção e apresentação de cursos e materiais didáticos relacionados; e atuação no mapeamento e implantação de processos e iniciativas de melhoria organizacional em empresas diversas.

4.4. Dissertações geradas

Duas dissertações de mestrado foram geradas em 2006, tendo seus resultados preliminares apresentados no Workshop de Teses e Dissertações do Simpósio Brasileiro de Engenharia de Software de 2006 estando, no momento desta submissão, aguardando data para realização das defesas na Pós-Graduação em Engenharia Elétrica da UFPA.

4.5. Parcerias e programas de transferência de tecnologia efetuados

Antes da liberação da versão 1.0 do sistema para a comunidade de Software Livre, versões preliminares da ferramenta estavam disponíveis desde março de 2006 para os parceiros institucionais, a saber: o Centro de Tecnologia (LACEN) da Eletronorte e SERPRO-Belém. Avaliações de usabilidade do produto se seguiram, além de demandas de melhoria foram registradas e priorizadas para o lançamento oficial do produto.

No **SERPRO-Belém**, a ferramenta foi disponibilizada para acompanhar o andamento de projetos de curta duração. No **LACEN**, o WebAPSEE vem sendo usado na definição de um modelo de processo para guiar o desenvolvimento de software naquela instituição em um projeto de cooperação tecnológica com a *Universität Stuttgart* (Alemanha) e o Instituto de Estudos Superiores da Amazônia (IESAM). O modelo de processo foi proposto pela equipe do IESAM no primeiro semestre de 2006 e foi mapeado para o WebAPSEE para conduzir pequenos projetos entre agosto e novembro. Um projeto-piloto está sendo acompanhado desde novembro de 2006.

O WebAPSEE é utilizado ainda pela própria equipe do LABES-UFPA para controlar o desenvolvimento de software relacionado com demandas diversas. Sabe-se também que a ferramenta (ou suas variações, desenvolvidas a partir do código-fonte disponibilizado) vem sendo utilizada por outras empresas no país.

4.6. Outros resultados

As seções a seguir apresentam resultados que não se enquadram nas categorias acima.

4.6.1. Formação em Engenharia e Gestão de Processos de Software

A experiência citada anteriormente na implantação da ferramenta na indústria foi determinante para a capacitação de um conjunto de professores e alunos em situações práticas de definição e avaliação de projetos. A partir desta experiência, foi criada a disciplina de “Engenharia de Processos de Software” a qual é fortemente baseada na adoção de tecnologia para apoiar a definição e implantação de processos de software, onde o WebAPSEE é usado tanto em exercícios práticos de modelagem quanto no acompanhamento de projetos conduzidos em outras disciplinas. Esta disciplina foi ministrada no primeiro semestre de 2006 para o curso de Bacharelado em Sistemas da Informação da UFPA. No segundo semestre de 2006, uma versão expandida foi ministrada em turmas para a graduação e no Mestrado em Ciência da Computação. A experiência com essa disciplina foi determinante para que os autores propusessem em dezembro de 2006 a criação de um curso de Especialização em Gerência de Projetos de Software, com proposta aprovada na UFPA para início em março de 2007.

4.6.2. Repercussão do produto na mídia local e em eventos especializados

O produto WebAPSEE foi oficialmente lançado à comunidade de Tecnologia de Informação em outubro de 2006 no auditório do CAPACIT-UFPA, com a presença de aproximadamente 100 pessoas⁵¹¹. Nessa ocasião, foi possível demonstrar a ferramenta e disseminar o trabalho sendo realizado pela equipe do LABES. A repercussão foi muito positiva, com destaques de reportagens na Imprensa local e em sites de notícias relacionados à Computação.

Além da premiação de melhor artigo já citada na seção 4.2, outras palestras convidadas foram apresentadas, onde podemos citar: I Workshop de Integração de Ferramentas para a Melhoria do Processo de Gestão de Desenvolvimento e Comercialização Colaborativos de Software (promovido pelo ICMC-USP em maio/2006), apresentações na SEFA (Secretaria de Estado da Fazenda), Banpará (Banco do Estado do Pará) e Banco da Amazônia, além da apresentação no próprio EQPS 2006 em Belém, dentre outras. Foi bastante positiva também a participação na Sessão de Ferramentas do SBES 2006, onde houve

¹¹ 5 Fotografias do lançamento do WebAPSEE 1.0 estão disponíveis no item Eventos no site do LABES.

demonstração da ferramenta e suas funcionalidades a pesquisadores da área e elogios foram feitos ao trabalho.

5. Aplicabilidade dos Resultados

O ano de 2006 representou a evolução de uma proposta acadêmico-científica para um produto de alta tecnologia de propósito geral para atender a forte demanda por ferramentas especializadas na gestão do processo de software.

A adoção prática de PSEEs na indústria é um desafio relatado pela literatura especializada desde as primeiras propostas da década de 1990. A concepção do WebAPSEE se valeu, então, simultaneamente de tecnologias recentes e formalismos adequados para facilitar o seu uso e evolução. Assim, o fato de ser baseado em tecnologia Web (Lima, 2006a), técnicas de inteligência artificial, e Gramática de Grafos (entre outros métodos formais) na especificação do seu núcleo funcional (Lima Reis, 2006) constituem exemplos do cuidado tomado desde a concepção do produto.

A adoção satisfatória do WebAPSEE em empresas preocupadas com certificações de qualidade industrial, como o SERPRO-Belém e a Eletronorte – é natural. Nestes casos, as empresas estão plenamente convencidas das vantagens na adoção de modelos de processos rigorosos e a tarefa se resume ao mapeamento destes modelos para a notação fornecida pelo ambiente. Entretanto, deve-se observar o grande interesse que a tecnologia desperta nas micro e pequenas empresas que se sentem atraídas pela possibilidade de se definir processos e imediatamente obter seus resultados através da monitoração em tempo real fornecida pelo ambiente. A facilidade para lidar com mudanças durante a evolução dos processos – tão comuns em organizações com baixa maturidade e recursos limitados - também tem se demonstrado atrativa. Outra questão frequentemente elogiada é a facilidade na adoção de práticas de gestão de configuração de software fornecida pelo ambiente. Assim, embora o WebAPSEE e todo ferramental tecnológico fornecido esteja alinhado aos níveis mais altos de maturidade de software, organizações que buscam certificações iniciais ou mesmo apenas desejam uma melhoria no seu controle têm se valido da ferramenta para auxiliá-los nesta tarefa.

6. Características Inovadoras

O ambiente WebAPSEE apresenta uma série de componentes que fornecem funcionalidades inovadoras mesmo quando analisadas em separado. A base formal – que dá sustentação à arquitetura de software utilizada - estabelece mecanismos que foram pensados para atender os requisitos de qualidade de uma ferramenta de grande porte.

Assim, a arquitetura foi desenhada para atender uma demanda crescente de usuários, permitir a fácil integração de novas ferramentas, e integrar diferentes protocolos de comunicação distribuída. As técnicas de Inteligência Artificial integraram-se nos componentes que permitem apoio a decisão, como alocação de pessoas (no mecanismo baseado em Políticas de Instanciação), simulação de processos de software (baseada em conhecimento), e reutilização de processos

(com uso da técnica de raciocínio baseado em casos). Toda a documentação está disponível no *website* do projeto (LABES, 2007), constituindo elemento de distinção perante seus congêneres, cuja documentação normalmente se restringe ao código-fonte.

Ressalta-se que, na opinião dos autores, a maior inovação do projeto consiste na integração de todas estas funcionalidades em uma única ferramenta. A sinergia desta integração permite, por exemplo, que o mecanismo de alocação de pessoas possa definir políticas reutilizáveis em diferentes processos, e estas políticas possam tomar decisão com base no estado corrente (registrado pelo *log* de execução). Assim, é possível obter mais benefícios que o obtido com soluções isoladas.

7. Conclusão e Perspectivas Futuras

O uso prático da ferramenta WebAPSEE na definição e implantação de processos de software em organizações de pequeno a médio porte constituem a comprovação, com base nas evidências desta realidade, do grande interesse dedicado a este tema.

As principais limitações existentes na ferramenta atual dizem respeito à dificuldade em sua instalação e de seus componentes. Por ser baseada em outros produtos (como o MySQL e CVS) a instalação exige a configuração de diversos elementos, os quais são influenciados pela plataforma de execução (e.g., Windows ou Linux), características da rede e outros detalhes técnicos que paulatinamente vão sendo identificados e incorporados à documentação do projeto.

A proposta da ferramenta vem demonstrando o impacto social, técnico e econômico que ela pode causar: sua construção ajuda na formação de recursos humanos altamente capacitados, na contribuição científica à Engenharia de Software, enquanto que o seu, livre, ajuda a empresas locais na implantação de melhoria de processos de software. Por fim, a visibilidade do projeto também traz bons frutos para a Região Norte, e amplia os horizontes do grupo para mais projetos inovadores.

8. Referências Bibliográficas

Covre, V.; Lima Reis, C.A. (2007). Avaliação da Conformidade do WebAPSEE ao MPS-BR. Relatório de Pesquisa. Laboratório de Engenharia de Software – UFPA.

França, B. (2007). Proposta de um Modelo de Simulação Processo de Software para o Ambiente WebAPSEE. Trabalho de Conclusão de Curso. Belém: CBCC-UFPA.

Gimenes, I.M. (1994). Uma Introdução ao Processo de Engenharia de Software: Ambientes e Formalismos. 13ª Jornada de Atualização em Informática. SBC.

LABES-UFPA. (2007). Doc. de Referência do WebAPSEE, <http://www.labes.ufpa.br>

Lima Reis, C.A. (2003). Uma Abordagem Flexível para Execução de Processos de Software Evolutivos. Tese de Doutorado. Porto Alegre: PPGCC-UFRGS.

Lima Reis, C.A. (2007). Projeto do Curso de Especialização em Gestão de Projetos de Software da UFPA. http://www.labes.ufpa.br/gerencia_projetos.htm (Março)

Lima, A.M.; et al. (2006A) WebAPSEE: Um Ambiente Livre e Flexível Para Gerência de Processos de Software. In: VII Workshop sobre Software Livre, 2006, Porto Alegre. Fórum Internacional de Software Livre 7.0, 2006. p. 163-168.

Lima, A.M.; França, B.; Costa, A.; Reis, R. Q.; Lima Reis, C. A. (2006B) Gerência Flexível de Processos de Software com o Ambiente WebAPSEE. Sessão de Ferramentas do 21o. Simpósio Brasileiro de Engenharia de Software.

Lima, A.M.; Lima Reis, C.A.; Reis, R.Q. Análise do Ambiente WebAPSEE no Atendimento aos Requisitos de Gerência de Processos de Software. (2006C). XX Semana Paraense de Informática (SEPAI). SBC, SUCEsu-PA

Lima, A.M. (2007). Execução Descentralizada de Processos de Software Baseada em Contratos Eletrônicos no Ambiente WebAPSEE, TCC Belém: CBCC-UFPA.

Nascimento, L.M.; Reis, R.Q. (2006). Medição e Análise de Processo de Software Utilizando Técnicas de Inteligência Artificial. XI Workshop de Teses e Dissertações em Engenharia de Software. XX Simpósio Brasileiro de Engenharia de Software.

Paxiúba, C.M. (2007). Utilização de um Mecanismo de Gerência de Eventos em um Ambiente de Desenvolvimento de Software Centrado em Processo. Dissertação de Mestrado. Programa de Pós-Graduação em Engenharia Elétrica (a ser defendida).

Paxiúba, C.M.; Reis, R.Q. (2006). Simulação de Processo de Software Baseada em Conhecimento. XI Workshop de Teses e Dissertações em Engenharia de Software. XX Simpósio Brasileiro de Engenharia de Software, p.25-30.

Sales, E.O.; Freitas, S.; Reis, R. Q. (2006). Uma Ferramenta para Recuperação de Modelos de Processo de Software Reutilizáveis. Sessão de Ferramentas do XX Simpósio Brasileiro de Engenharia de Software, Florianópolis.

Silva, M. A. (2007). WebAPSEE-Planner - auxílio à alocação de pessoas em projetos de software através de políticas. Trabalho de Conclusão de Curso. Belém: CBCC-UFPA. Softex (2007). MPS-BR. <http://www.softex.br/mpsbr>

[6.17] Definição de um modelo para mensurar projetos de manutenção de sistemas

Entidade: 1Caixa Econômica Federal - Brasília - Brasil

Autor: Angélica Toffano Seidel Calazans¹ - angelica.calazans@caixa.gov.br

Abstract. *The organizations need to estimate the size of maintenance and development systems to control time, cost and resources in a better way. There are different approaches to estimate the size of a system, and the majority of them propose to measure the size in any phase of the life cycle of a system (new development or maintenance). However, some writers argue that the maintenance*

has specific particularities and that it is important to have an estimative approach which considers these characteristics. This work shows an analysis and the application of some measurement approaches to maintenance projects of Caixa Econômica Federal.

Resumo. *Para melhor controlar tempo, custo e recursos em projetos de software as organizações necessitam estimar o tamanho de novos desenvolvimentos e das manutenções de sistema. Existem diferentes abordagens para estimar o tamanho de um sistema e, em sua maioria, se propõem a medir o tamanho de qualquer etapa do ciclo de vida do sistema (novo desenvolvimento ou manutenção). Contudo, alguns autores argumentam que a manutenção tem particularidades específicas e é importante ter uma abordagem de estimativa que considere estas características. Este trabalho apresenta a análise e aplicação de algumas abordagens de medições para projetos de manutenção da Caixa Econômica Federal.*

1. Introdução

A medição na engenharia de software é um dos fatores importantes para a geração de um produto com qualidade. Segundo Fenton e Pfleeger (1997, p.12-13), mensura-se para compreender e controlar o processo de produção de software. A mensuração do tamanho do software na gestão de projetos também está vinculada à necessidade de avaliar e medir resultados, conhecer melhor o patrimônio de software, gerar indicadores para tomada de decisão, avaliar o impacto da introdução de novas tecnologias e obter e/ou melhorar estimativas de prazo, custo e recursos gerando expectativas mais realistas para o usuário (SIMÕES, 1999).

Várias abordagens têm sido propostas e aperfeiçoadas desde a década de 1960, com o objetivo de definir o tamanho de um software, entre elas: Lines of code – LOC (1950/1960) (FENTON e NEIL, 2000), a abordagem Halstead (1972) (BATENBURG, RIDDER e KERF, 1998), Function Point Analysis (FPA) ou Análise por Pontos de Função (APF) (1979) (IFPUG, 2000), o MKII Function Point Analysis (1991) (LOKAN e ABRAN, 1999) e o Full Function Points/COSMIC Full Functions Points (1997) (ABRAN et al, 1999). A maior parte dessas propostas busca medir o tamanho de qualquer tipo de software, independente da etapa do ciclo de vida do sistema, ou seja, se é um novo desenvolvimento ou uma alteração de alguma funcionalidade já existente.

Após a implantação, grande parte dos sistemas sofre alterações ao longo do período em que estão sendo utilizados. Seus requisitos são modificados para atender alguma necessidade do usuário, mudança de ambiente, introdução de novo hardware ou software, correção de erros e outros. Este processo de mudança do software após sua implementação é chamado de manutenção de sistemas. A manutenção de sistemas é uma realidade no mercado de software sendo considerada em estimativas como até 70% de todo o trabalho de engenharia de software. Uma parcela significativa do orçamento em Tecnologia da Informação (TI) de muitas organizações é dedicada à manutenção de sistemas e, segundo Stuzke (2000), as empresas necessitam estimar de forma acurada o

tamanho dos produtos de software no início do processo de desenvolvimento e/ou manutenção para melhor planejar e prever a construção de produtos de software.

A estimativa em projetos de software envolve, na maioria das vezes, a previsão de quatro variáveis: tamanho (dimensão do software ou da manutenção produzido ou a produzir), esforço (trabalho necessário para o desenvolvimento ou manutenção do software), prazo (tempo necessário para o desenvolvimento ou manutenção do software) e qualidade (envolvendo fatores como manutenibilidade, confiabilidade e outros)(AGUIAR, 2002).A primeira variável a ser determinada é sempre o tamanho do software, produzido ou a produzir, pois é a partir da definição da dimensão que é possível definir o esforço e o prazo necessários para o desenvolvimento ou manutenção do software. Isso demonstra a importância de uma mensuração de tamanho o mais próxima possível da realidade, pois ela é um fator que causa impacto nas demais variáveis.

Considerando o exposto, torna-se necessário estudar e adequar, se necessário, as abordagens de medição de tamanho para que se considerem as características específicas de projetos de manutenção e com isso gerem estimativas mais reais.

2. Objetivos e justificativa

Este projeto tem por objetivo a identificação e definição de uma proposta de mensuração de tamanho para projetos de manutenção.

São objetivos específicos:

- (i) Estudar as características principais de Projetos de Manutenção, identificando aspectos diferenciados;
- (ii) Estudar algumas abordagens de métricas de tamanho existentes, analisar sua aplicabilidade a esse contexto e identificar as melhores alternativas para projetos de manutenção;
- (iii) Utilizar e avaliar algumas abordagens em projetos de manutenção;
- (iv) Comparar os resultados da aplicação destas abordagens de mensuração com os resultados da abordagem utilizada pela CAIXA.

3. Metodologia de execução

O tipo de pesquisa utilizada classifica-se como pesquisa aplicada, uma vez que busca a resolução de problema concreto, que é a mensuração de projetos de manutenção de software.

Com relação aos meios de investigação, foram utilizados: a pesquisa bibliográfica baseada em material publicado em livros, revistas, jornais, redes eletrônicas, anais; estudo de caso circunscrito aos projetos possíveis de serem mensurados na Caixa Econômica Federal e investigação documental, apoiada em pesquisa documental dos sistemas a serem pontuados.

Foram analisadas, por meio de pesquisa bibliográfica, as características diferenciadas de um processo de manutenção de um sistema, as deficiências que as métricas possuem quando aplicadas a projetos de manutenção e as críticas existentes.

Foram identificadas três abordagens existentes, com base na pesquisa bibliográfica e aplicadas nos projetos de Manutenção. Foram comparados os resultados dessas propostas.

Foram escolhidas para análise as abordagens APF, a NESMA e o COSMIC. A APF foi escolhida por possuir maior maturidade tanto no meio acadêmico como na indústria, pois é uma das abordagens mais utilizadas e estudadas atualmente (Garmus e Herron, 2000, p. xvi). Por ser uma das métricas mais antigas, a APF possui um diferencial em relação a abordagens mais novas, pois existem dados de produtividade (relação horas por pontos de função) da indústria coletados por vários órgãos entre eles, o International Software Benchmarking Standards Group – ISBSG. Esses dados possibilitam o estudo ou mesmo a utilização desses índices por organizações que não possuem um histórico de contagens. Outro fator decisivo na escolha foi que a APF é a métrica utilizada pelas Representações de desenvolvimento de sistemas da CAIXA e nos contratos de terceirização.

A análise da abordagem NESMA foi escolhida por ser uma adequação da APF para projetos de manutenção. A escolha da abordagem COSMIC se justifica pois esta abordagem foi identificada pela literatura como uma abordagem mais atual, mais fácil, rápida de ser entendida e aplicada.

3.1 Coleta e análise de dados

Foram coletados dados sobre projetos nas Representações de Desenvolvimento e Manutenção de Sistemas da Caixa Econômica Federal. Foram também realizadas entrevistas com os participantes do processo (desenvolvedores, líderes, métricos).

4. Resultados obtidos

4.1 Outros produtos gerados

O principal produto gerado foi a utilização deste trabalho para definições internas e externas da Caixa com relações aos serviços de manutenção.

Outro produto gerado foi a premiação deste trabalho no 3º. lugar no Concurso de Estudos e Pesquisas para Melhores práticas em TI realizado pela Caixa Econômica Federal.

4.2 Artigos publicados

08/2005 - nos Anais da 34ª. Jornadas Argentinas de Informática e Investigación Operativa – Rosário/Argentina, com o artigo “Avaliação de Estimativa de Tamanho para Projetos de Manutenção de Software”, Autores: Calazans, Angélica; Oliveira, Marcelo.

A seguir são descritos, sucintamente, os resultados da aplicação da pesquisa.

4.3 Aplicação da APF em manutenções da CAIXA

Foram analisados 17 projetos de manutenção já finalizados. O objetivo era comparar o esforço estimado (qtd horas estimadas), inferido a partir da aplicação de fatores de produtividade ao tamanho do projeto, obtido com a métrica APF, e o esforço real (qtd horas reais) gasto pelas equipes de manutenção.

Os projetos foram pontuados seguindo a metodologia da APF para manutenção. Para cálculo do esforço estimado, foram utilizados os percentuais de esforço definidos para cada uma das etapas da metodologia utilizada pela CAIXA e a produtividade (Horas/PF) de cada etapa. A produtividade foi obtida dos contratos de terceirização da CAIXA.

A Figura 4.1 demonstra de forma gráfica a diferença de esforço obtido entre a estimativa utilizando APF e o esforço real.

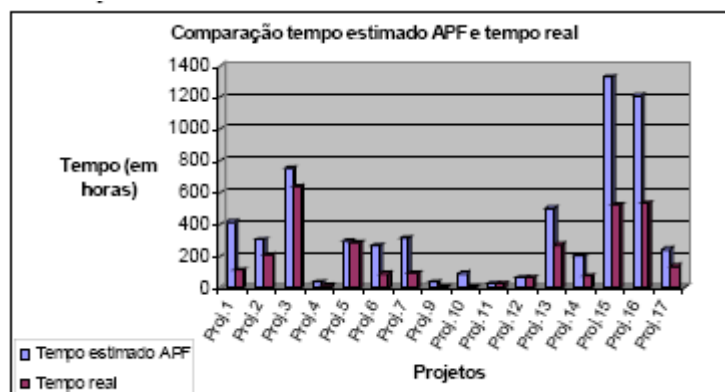


Figura 4.1 – Impacto do tempo (esforço) real/tempo (esforço) estimado APF

Em 88% dos projetos da amostra estudada, o cálculo do esforço (qtd de horas) obtido com a utilização da APF, ficou acima do tempo real gasto para a efetiva construção da manutenção. Após esta análise, foi estudado o fator de impacto (valor obtido a partir da divisão do tempo real e o tempo estimado). Este estudo possuiu o objetivo de identificar o impacto médio das diferenças observadas entre o esforço real e o esforço estimado APF (Figura 4.2).

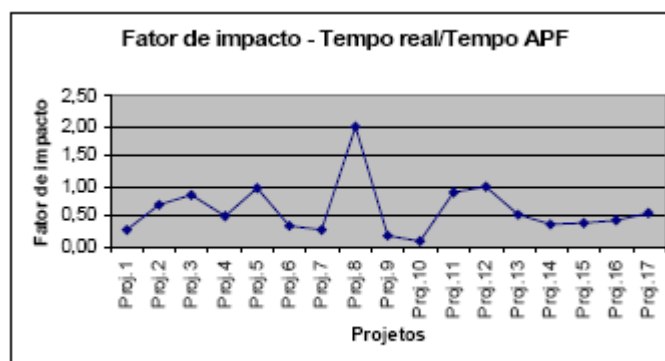


Figura 4.2 – Impacto do tempo real/tempo estimado APF

Em 94% dos projetos analisados o fator de impacto ficou abaixo de 1 (inclusive).

Considerando este resultado, foi desconsiderado o dado referente ao Proj. 8 e calculada a média do fator de impacto para os 16 projetos restantes. A média do fator de impacto para esta amostra ficou em 0,52 e a mediana em 0,47.

Considerando os resultados obtidos, pode-se concluir, que há evidências, na amostra estudada, que os resultados obtidos por meio da aplicação da métrica APF não possuem um grau de acurácia razoável em 94% dos projetos de manutenção analisados.

Foi realizada uma entrevista estruturada junto às equipes de manutenção e foi consenso que pequenas manutenções, onde há a alteração/exclusão de um atributo de um ALI, ou seja, uma tarefa relativamente simples, a pontuação de todo o ALI na melhoria tem aumentado substancialmente a contagem final da manutenção. Estas equipes reconhecem que, com relação aos Projetos de Melhoria, a APF superestima o tamanho do projeto.

É necessário reconhecer esta deficiência para que as organizações possam identificar caminhos alternativos para a mensuração de projetos de manutenção. Aumentar a produtividade (qtd horas/PF) para manutenção pode ser uma alternativa. Mas, pesquisa feita em bases históricas de produtividade de organizações (ISBSG, 2002), demonstrou que a produtividade de manutenção e a produtividade de novo desenvolvimento destas bases, tem sido aproximada. Outra alternativa seria pontuar os projetos com a utilização de outras abordagens.

4.4 Aplicação da NESMA em manutenções da CAIXA

O gráfico 4.3 abaixo demonstra a comparação APF X NESMA X Esforço Real.

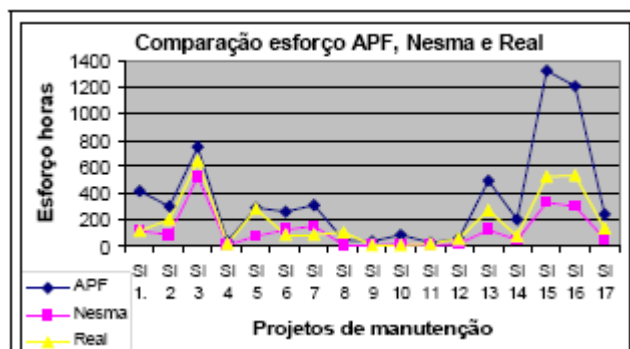


Figura 4.3 – Impacto do esforço estimado APF/esforço estimado Nesma/esforço real

O cálculo de esforço obtido pela APF foi maior do que o método NESMA em 100 % dos casos. Em 70% dos casos o valor de esforço obtido pelo método NESMA foi menor do que o esforço efetivamente realizado.

Por não haver disponível uma base histórica sobre a aplicação do método, não foi possível obter a produtividade de horas por EFP, dificultando sobremaneira a

análise e validação do método, pois foi utilizada a produtividade que a CAIXA utiliza em seus contratos de terceirização (baseada no ISBSG v.6).

A base do ISBSG versão 9 possui bases históricas de produtividade, inclusive com produtividade de manutenção para Nesma. Mas a CAIXA somente possui a versão 6 da base do ISBSG, que possui poucas manutenções utilizando a Nesma, o que inviabiliza a utilização de uma produtividade mais adequada a esta métrica.

4.5 Aplicação COSMIC em manutenções da CAIXA

A mensuração de tamanho com relação ao escopo de projetos de manutenção, nesta abordagem, é mais pontual uma vez que considera somente os dados modificados/alterados ou excluídos. A COSMIC não pontua funções de dados. Para este trabalho foram analisados alguns projetos já pontuados em APF e foram pontuados utilizando a abordagem COSMIC.

A base com produtividade de mercado que a autora possui acesso é o ISBSG v.6, e esta base possui poucas medições com a métrica COSMIC. A versão do ISBSG v.9 possui uma quantidade razoável de sistemas mensurados utilizando COSMIC, mas a CAIXA não adquiriu esta base e assim sendo, não foi possível identificar a média de produtividade por CSFU (unidade de medida da COSMIC), optou-se então por aplicar a produtividade de 8 h por CSFU, esforço indicado por Koppenberg e Dekkers (2004).

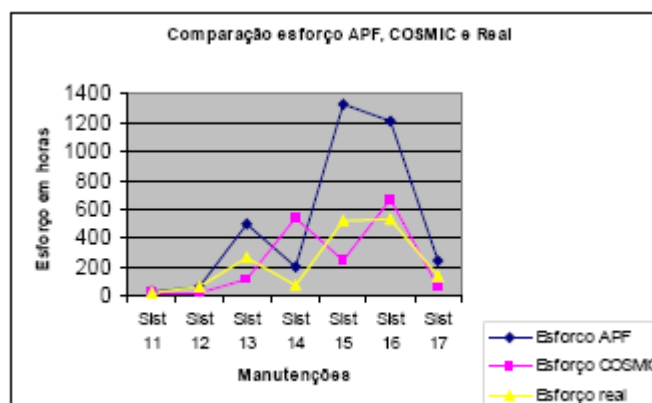


Figura 4.4 - Impacto do esforço APF/esforço estimado COSMIC/Esforço real

5. Aplicabilidade dos Resultados

Uma das maiores dificuldades encontradas pela gestão de projetos é estimar o porte do que está sendo construído. Existem muitas abordagens para mensurar o tamanho de um software e não existe uma abordagem que seja melhor que outra, sob todos os aspectos, em qualquer situação. A abordagem de mensuração de tamanho deve ser escolhida e/ou adequada dependendo das características particulares do sistema que se pretenda desenvolver ou manter ou do problema que se pretenda solucionar (CALAZANS, OLIVEIRA e SANTOS, 2003) .

Manter sistemas propõe uma visão diferenciada do processo de desenvolvimento.

Diferentemente de novo desenvolvimento, o processo de manutenção é mais complexo, especializado e com características específicas. Isso requer que outros processos, como os processos de medição (atualmente voltados para novo desenvolvimento), que interagem com o processo de construção de forma a viabilizar um melhor gerenciamento do processo de construção de um software, sejam adaptados de forma a garantir o gerenciamento adequado de recursos, custos e tempo.

Com essa motivação em mente, este trabalho teve por objetivo a identificação e definição de uma proposta de mensuração de tamanho para projetos de manutenção e os seguintes objetivos específicos foram elaborados:

- (i) Estudar as características principais de projetos de manutenção, identificando aspectos diferenciados em relação ao desenvolvimento de sistemas ;
- (ii) Estudar algumas abordagens de métricas de tamanho existentes, analisar sua aplicabilidade a esse contexto e identificar a melhor alternativa para projetos de manutenção;
- (iii) Utilizar e avaliar algumas abordagens em projetos de manutenção;
- (iv) Comparar os resultados da aplicação destas abordagens de mensuração com os resultados da abordagem utilizada pela CAIXA.

Para atender a esses objetivos, foram estudadas as principais características da manutenção e identificadas as principais diferenças com relação aos novos desenvolvimentos. Foram estudadas algumas abordagens de métricas de tamanho existentes, seus pontos fortes e as críticas existentes. Dessa forma atende-se aos objetivos (i) e (ii).

Para os objetivos (iii) e (iv), foi definida a APF, NESMA e COSMIC. Foram aplicadas as métricas em 17 projetos de manutenção (APF e NESMA) e 07 projetos (COSMIC).

Foram comparados e analisados os resultados.

6 Características inovadoras

Poucos estudos focam as peculiaridades do processo de manutenção de sistemas e a adequação das métricas de tamanho existentes no mercado para este tipo de serviço. No presente trabalho foram analisadas 3 métricas e aplicadas no contexto organizacional. É importante identificar pontos positivos e negativos das abordagens existentes de forma a garantir uma maior eficácia das medições realizadas.

7. Conclusão e perspectivas futuras

A manutenção faz parte do ciclo de vida de um sistema. Os sistemas são como os negócios, estão constantemente em evolução. Mensurar o tamanho da manutenção para inferir prazos em que a demanda será atendida, é necessidade de todas as organizações.

Neste trabalho, foram aplicadas 3 abordagens de métricas em alguns projetos de manutenção. Foram efetuadas análises para verificar a adequabilidade das abordagens ao contexto de manutenção de sistemas. Ao longo do estudo de casos reais, foi demonstrado todas as abordagens apresentam falhas, e que devem ser adequadas (APF, Nesma, COSMIC) ou melhor estudadas para serem utilizadas pela CAIXA para projetos de manutenção.

O trabalho realizado permitiu identificar alguns pontos que requerem melhorias e que poderão ser alvos de estudos futuros:

□ o estudo foi aplicado somente para projetos de manutenções pequenos. Contudo sua adequação a outros tamanhos de projetos de manutenção necessita ser verificada por meio de aplicação em outros projetos, de forma a enriquecer ou mesmo validar os resultados;

□ a abordagem COSMIC precisa ser melhor estudada para verificar sua aplicabilidade;

□ é necessário, também uma flexibilização das metodologias existentes de forma a uma maior adequação ao contexto diferenciado da manutenção de sistemas.

8. Referências bibliográficas

Aguiar, Maurício. (2002) Estimando os projetos com Cocomo II no RUP. "Developers Magazine".

Abran, A., Desharnais, J., Oigny, S., St-Pierre, D., Symons C. (1999). Cosmic FFP Measurement Manual. v. 2.2, Ed. S.Oigny, "Software Engineering Management Research Laboratory". Université du Quebec a Montreal. Canada.

Batenburg, F., Ridder, E., Kerf, J. (1998). APL Extended compared with other languages according to Halstead's theory. "ACM Sigplan Notices". 33(6), p. 54-60.

Calazans, Angélica; Oliveira, Kathia; Santos, Rildo.(2003). Dimensionando Data Marts : Uma adequação de uma métrica funcional. In: "Anais do II Simpósio Brasileiro de Qualidade de Software- SBQS". UNIFOR, CE.

Fenton, N., Pfleeger, S.(1997). "Software metrics a rigorous & practical approach". 2nd. Ed., PWS Publishing Company.

Fenton, N., Neil, M.(2000). Software Metrics: Roadmap. Future of Software Engineering. Limerick Ireland "ACM". p. 359-370.

Garmus, D., Herron, D.(2000). "Function point analysis – measurement practices for successful software projects". Addison-Wesley Information Technology Series.

IFPUG. (2000). International Function Point Users Group. Function Point Counting Practices "Manual: Release 4.1". Ohio: IFPUG. 1 v. ISBSG. (2002) Benchmarking Repository, Release 6. "ISBSG".

Koppenberg, T., Dekkers, T. (2004) Estimating maintenance projects using COSMIC FFP. "IWSM – Metricon".

Lokan, C., Abran, A. (1999). Multiple viewpoints in functional size measurement. In: "International Workshop on Software measurement - IWSM'99". Canada. p. 121-132.

Stutzke, R. Predicting Estimation Accuracy. (2000). In: "The European software control and metrics conference – ESCOM", Alemanha, p. 211 – 220.

[6.18] A Importância do CMMI para uma Pequena Empresa de Desenvolvimento de Software

Entidade: Fábrica de Software - ZCR Informática Ltda. - Rua Rubens Guelli, nº134, CEP 41.815-150 - Salvador - BA - Brasil

Autor: Ruben Arnaldo Soto Delgado - ruben@zcr.com.br

Abstract. *The purpose of this work is to write about the experience, lived by ZCR - Informática LTDA, in obtaining the Level 2 CMMI award and the result achieved based in the vision of the organization high-level management. This paper talks about the importance of the measurement data's selection, collection and management attendance. It also shows that is possible for a small company to follow an international software development model.*

Resumo. *Este trabalho tem o objetivo descrever o relato da experiência da obtenção do Nível 2 de maturidade CMMI pela empresa ZCR – Informática Ltda, bem como os resultados por ela obtidos na visão da alta gerência da organização. O trabalho mostra a importância da escolha, coleta e principalmente do acompanhamento gerencial dos indicadores trabalhados no programa de Medição da empresa. Este relato mostra que é possível e necessário uma pequena empresa ter um padrão de qualidade de desenvolvimento de software de padrão internacional.*

1. Introdução

A área de desenvolvimento de software é uma área difícil de analisar como negócio. O ciclo de vida de produção de software não é comum a todas as empresas, e a sua variação é muito grande comparado com outras áreas da indústria. Pode-se fazer um paralelo com uma indústria comum, ou com uma prestadora de serviços médicos, por exemplo. É fácil verificar, nestas duas últimas áreas, que duas empresas concorrentes apresentam pequenas variações nos seus modelos de produção, passando para a área de gestão os diferenciais competitivos entre elas. Obviamente, levando em consideração um posicionamento claro do seu produto e/ou serviço.

Ao contrário, as fábricas de softwares apresentam uma grande diferenciação entre elas. Existem empresas que não tem absolutamente nenhum compromisso com um padrão de qualidade no seu processo produtivo, ao contrário daquelas que investem na excelência de produção e que almejam não apenas o mercado nacional como meio de sustentação.. Este relato, não está orientado para estas últimas, que tem um posicionamento bem claro, uma carteira de clientes de grandes empresas e um padrão de qualidade definido ou pelo menos planejado.

Está direcionado a empresas que atendem um mercado SMB (Small and Middle Business), que por muitas vezes não sabem como comprar, nem como comparar empresas de níveis de maturidade tão diferentes.

A visão que este relato tenta passar é que o investimento na qualidade é muito mais do que um diferencial competitivo, mas uma questão de sobrevivências da companhia. Somente com um nível de maturidade reconhecido e validado, o empresário passa a conhecer a sua empresa, passa a fundamentar o seu *feeling* e a poder tomar decisões estratégicas na sua organização.

Este relato mostra a importância do nível de maturidade atingido pela ZCR – Informática, empresa pequena, com uma fábrica de pouco mais de 30 pessoas, que tem um mercado local na RMS¹² e foi a 10ª empresa brasileira a obter o laudo positivo de CMMI – Nível 2, e a primeira no Norte e Nordeste do Brasil.

2. ZCR - Uma pequena empresa obtendo o CMMI

Compartilhar despesas. Isto aconteceu através da Assespro-Ba, que com 8 empresas inicialmente formaram o grupo CMMI-Bahia, onde até o presente momento, apenas a ZCR conseguiu obter o laudo, mas mais três empresas devem conseguir até o final de junho de 2006. Neste caso específico, gastos com treinamento, passagens de consultores, entre outras coisas podem ser compartilhadas com as empresas. Vale a pena salientar que estes custos compartilhados, embora significativos, não chegam a representar 25% dos gastos totais do projeto, portanto esta iniciativa não deve ser encarada como uma solução financeira para o projeto.

O segundo ponto é a automatização de processos. O investimento em ferramentas é necessário e imprescindível. Como toda implantação de um novo modelo a ser seguido, ou seja, um novo processo, a organização deve implementar uma rotina de institucionalização muito efetiva. Para tal a empresa deve ter como base alguns procedimentos e ferramentas que auxiliem neste momento. A utilização de uma Intranet torna-se fundamental para a institucionalização do projeto, e é um ponto chave para a obtenção do laudo positivo. No caso específico da ZCR, tudo o que o colaborador precisava em relação ao CMMI estava no portal, na Intranet da empresa, sejam *templates*, cursos, notícias, processos, etc. Esta ferramenta é de fácil implementação e implantação.

Outra ferramenta fundamental é a que ofereça um controle de apropriação de horas de todos os recursos da empresa. Desta forma a alta gerência da organização poderá ver em que a empresa está gastando. Esta ferramenta é uma ferramenta simples, onde a principal dificuldade está na implantação da mesma, e em conscientizar esta cultura por todos os colaboradores da empresa, portanto recomendamos o seu desenvolvimento interno. Deve-se ter em mente que este sistema irá ajudar a empresa em duas funções fundamentais, a 1ª como forma de medir e controlar o investimento no próprio projeto, e a segunda como poderosa fonte de dados para as medidas (indicadores) que serão obtidas.

¹² RMS – Região Metropolitana de Salvador

Outras ferramentas que devem compor a estrutura da empresa estão ligadas diretamente com o processo de desenvolvimento do software. Neste caso diversas opções de mercado são oferecidas, como por exemplo, a da Borland e a suíte Rational da IBM, sendo que a empresa optou pela última. É importante lembrar que nem todas as ferramentas que fazem parte do processo de desenvolvimento, como por exemplo, ferramentas ligadas ao procedimento de testes, precisam ser implementadas no nível 2 do CMMI, já que este nível foca muito mais o gerenciamento dos projetos.

O terceiro e último ponto é o compromisso e envolvimento da alta direção. É recomendável que todo e qualquer projeto que se deseje implantar dentro de uma organização, tem que ter seus frutos cobrados a partir da alta direção, e daí para baixo, no melhor estilo *top-down*. Se o colaborador não percebe que os dados por ele gerados, são úteis e se de alguma forma, o colaborador não recebe as informações (pelo menos algumas) como *feed-back*, então é muito difícil que se tenha sucesso na implantação deste projeto. É natural que o colaborador não se sinta a vontade para, todos os dias, colocar no sistema as tarefas por ele desempenhadas, o colaborador na nossa área, tem um perfil elevado, portanto questionador, logo tem que entender e respeitar este perfil, e buscar inseri-lo e mostrar como é útil o dado por ele gerado dentro de todo o contexto do projeto. É fundamental, portanto, que a alta direção faça reuniões periódicas de acompanhamento das métricas, e que estes resultados sejam publicados e divulgados pela empresa.

3. Colhendo frutos com o PPQA

Como foi descrito na introdução, o empresário de software deve olhar o CMMI como uma forma de controlar e conhecer o seu processo produtivo e não como um diferencial competitivo em um mercado exigente. Desta forma, duas práticas devem ser rigorosamente acompanhadas pela alta direção, a PPQA¹³ e a MA¹⁴. As outras 5 práticas não possuem um envolvimento tão direto com a gerência, portanto não serão abordadas aqui neste relato.

Além de acompanhar as métricas destas duas práticas, a alta direção deve estudar a tendência das curvas que estes índices apontam. Não é raro, encontrar empresas que obtêm estes índices e não há comparativo histórico entre eles. Desta forma, a empresa passa a controlar projetos, mas não controla para onde está indo. Isto é literalmente assustador, pois os planos de MA e PPQA exigem ou devem exigir planos de ação quando os índices em estudo não atingem os números indicados pela empresa, mas não se preocupam com a tendência da curva que estes números estão mostrando. Para exemplificar, determinada empresa só consegue perceber que a sua produtividade caiu abaixo do nível mínimo, após os projetos chegarem nesta situação, ao contrário um estudo de tendência mostrará que a produtividade vai cair abaixo do nível mínimo nos projetos que vierem a acontecer, possibilitando planos de ação preventivos.

¹³ PPQA – *io jhsdciods cs*

¹⁴ MA – Medição e Análise

O perfil de um profissional que assume o PPQA de uma empresa deve ser uma pessoa rigorosa e ao mesmo tempo comunicativa e técnica. É importante que este profissional tenha acesso direto e livre com a alta direção, pois ele é o responsável por verificar semanalmente o andamento de todos os projetos, verificando a

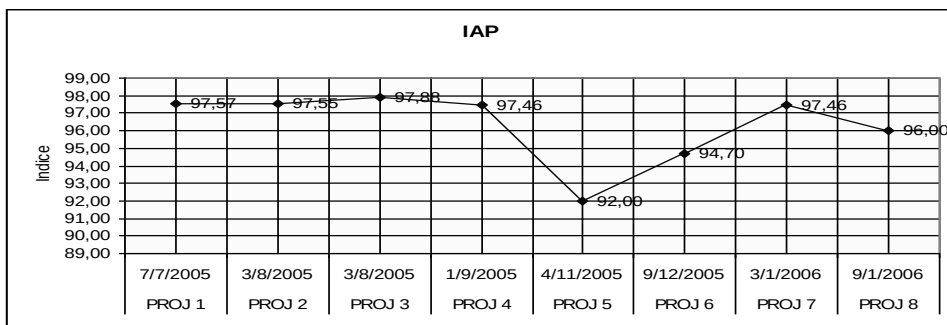


Figura 1. IAP – Índice de Aderência aos Processos

Duas informações interessantes são obtidas de suas auditorias, a primeira e mais importante é o nível de aderência aos processos. Esta informação vai mostrar de uma forma clara se os colaboradores estão seguindo os processos, e gerando os artefatos requeridos. Isto é o começo, pois se isto não está acontecendo, todo o processo é penalizado. Normalmente, as empresas não apresentam maiores problemas neste sentido, visto que a maioria dos colaboradores são analistas de sistemas, e acabam abraçando o projeto da obtenção do laudo rapidamente. Na figura 1 mostramos um gráfico de IAP – Índice de Aderência aos Processos x Data Final do Projeto. Pode-se verificar que a média está em 96,33%, isto levando em consideração um projeto que está fora da curva, que foi o projeto No 5. Neste caso, poderíamos tê-lo tirado para o cálculo da média, pois pontos fora da curva podem ser descartados para que não destorçam o resultado final. Neste caso a média iria para 96,95%. Podemos verificar que este gráfico não apresenta uma tendência clara, mas também mostra claramente que a aderência ao processo sempre (com uma exceção) varia entre 96 e 98%,

Outra informação que deve ser acompanhada pela alta direção é o ISMR – Índice de Solicitações de Mudança de Requisitos. Para quem desenvolve projetos de software há muito tempo, sabe que o cliente tem uma dificuldade muito grande para ajudar o analista a elicitar requisitos. Isto pode ser um problema muita grande para as empresas de software que vendem seus projetos da forma preço-fixo. Todos já passaram e continuam passando pela mudanças de requisitos no meio do projeto, e não é raro encontrar empresas que o requisito é mudado, o prazo é prolongado e o setor comercial não fica sabendo ou simplesmente abona este custo. Com as práticas do CMMI isto é devidamente registrado, e o projeto só continua após a aprovação formal do comercial que a área técnica pode acatar esta mudança. Apenas com esta ação, a empresa já passa a entender uma mudança é prejudicial e que o cliente tem que pagar por ela. O fundamental é que o índice vai acompanhando e mostrando isto entre todos os projetos da empresa.

Infelizmente, não temos histórico suficiente, mas já podemos notar com 8 projetos em estudo que 2 deles (projeto 4 e 8) apresentam um ISMR de 11 e 18% respectivamente. Percebemos que estes dois projetos vinham de um mesmo cliente que é fidelizado já a 5 anos, e que historicamente a nossa empresa não cobrava por estas mudanças de requisitos. Se considerarmos um lucro na operação de 10%, podemos chegar a conclusão que a empresa trabalhava para o cliente, sem nenhum lucro aparente. Ou seja, existe a possibilidade de ter tido 5 anos de desenvolvimento sem lucro é assustadora. Não há outra forma disto ser verificado sem um processo de maturidade como o CMMI, e indicadores como este podem ser a diferença entre sobreviver e morrer no mercado. Só para continuar exemplificando, a empresa passou a não aceitar mais mudanças de requisitos no cliente em estudo, e esta decisão foi tomada graças ao ISMR fornecido pelo PPQA.

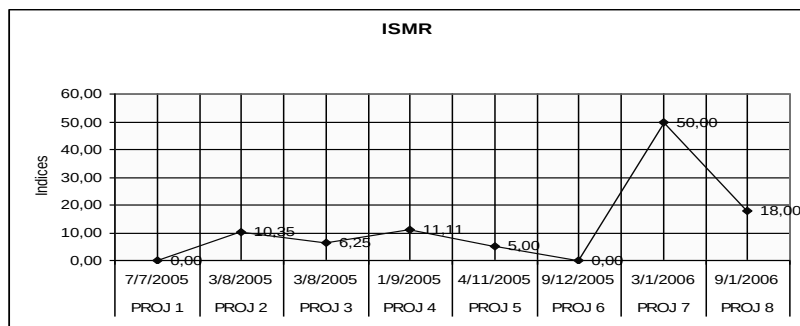


Figura 2. ISMR – Índice de Solicitação de Mudança de Requisitos

4. Empresa Mensurada e Controlada

Atualmente, nas empresas que não atingiram um nível de maturidade aceitável, as áreas comerciais passam a conviver internamente com um fator de erro para dimensionamento do preço dos serviços. Infelizmente, isto não é um padrão para toda a empresa, e sim um padrão para cada técnico que faz o levantamento. Esta metodologia beira o caos, e fatalmente gera prejuízos para a empresa, que só são sentidos pelo acompanhamento das horas gastas no projeto pós-concluído. Mesmo assim, não são levadas em consideração as horas que o recurso ficou parado na espera da aprovação de determinado artefato, por exemplo. Infelizmente, este recurso parado representa um custo fixo para a empresa. Nesta avaliação estamos considerando que a direção tem controle sobre as horas que cada recurso utilizou, caso contrário a situação é bem pior.

Um preço na nossa área já leva em consideração diversos fatores, capaz de desafiar Michael Porter, [PORTER 2000] para um estudo, mas sem o controle e sem o histórico da própria empresa, o índice mais utilizado vem do “feeling” dos vendedores que normalmente tendem a questionar a área técnica tentando-os convencer que o projeto consegue-se fazer com menos esforço.

Para tentar sair, deste problema que é comum na ares de informática, definimos como prioritário o acompanhamento das precisões, com destaque para: PET – Precisão de Estimativa de Tempo, PETAM – Precisão de Estimativa de Tamanho,

PEE – Precisão de Estimativa de Esforço e PEC – Precisão de Estimativa de Custo.

Aparentemente o trabalho da Medição e Análise serve para acompanhar quão são eficazes os nossos técnicos para estimar os trabalhos levantados pela área comercial, mas é muito mais do que isto. Podemos ver nos gráficos a seguir que no nosso caso, o esforço, tamanho e o custo (diretamente ligados), mostram que as estimativas não são boas. Podemos ver na PETAM que o tamanho médio estimado foi 16% menor do que foi realizado, mas se retirarmos o projeto 3, isto cai para uma média aceitável de 1,11. Devemos levar em consideração que alguns destes desvios foram oriundos de mudanças de requisitos com renegociação com o cliente.

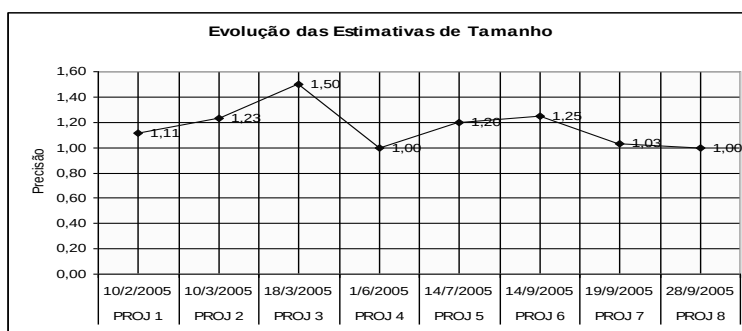


Figura 3. Evolução das Estimativas de Tamanho

Por outro lado a estimativa de esforço (PEE) média foi de 1,08, ou seja 8% a mais do que previsto. Mais uma vez, analisando sem o projeto 3, a média cai para a aceitável marca de 1,02. Isto demonstra que os nossos técnicos sabem estimar o esforço em projetos.

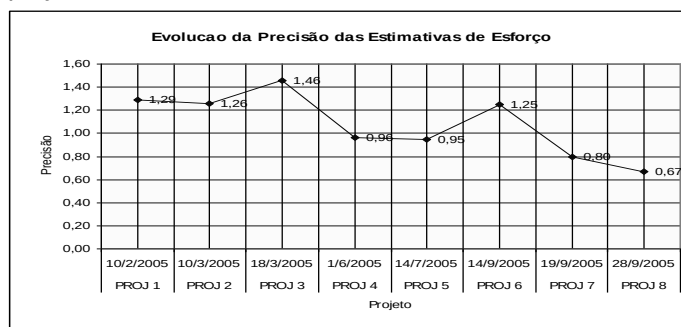


Figura 4. Evolução da Precisão das Estimativas de Esforço

A mesma análise pode ser feita na PEC – Precisão de estimativa de Custo, que normalmente acompanha a PEE. Pode-se ver que a tendência da curva é descendente, mostrando a importância da já citada divulgação e publicação de informações para todos os colaboradores. Podemos também verificar que a

tendência da curva do PEE, (e consequentemente a do PEC) mostra uma melhora sensível na empresa, atribuída à maturidade adquirida ao rodar os processos.

Ao contrário dos outros índices, o PET, é onde todas as atenções estão voltadas no momento. Um estudo desta métrica indica que não tem uma tendência de curva, e ao mesmo tempo, dá uma assustadora marca de 1,30, ou seja 30% a mais de tempo do que o estimado. Levando em consideração que em média gastamos 8% a mais do que estimado, e erramos no prazo em 30%, devemos estar com um bom tempo dedicado a espera de validações de artefatos, ida e volta de documentos, volta para melhor elicitação de requisitos, entre outros problemas. Fica claro e assustador que o lucro da empresa é todo revertido nesta métrica. Mas ficamos de certa forma, felizes por poder passar a controlar o que antes não tínhamos controle. Todo problema só é solucionável se o conhecemos primeiro, e neste caso diversos planos de ação já foram tomados para baixar este índice, e o resultado na empresa já começa a aparecer.

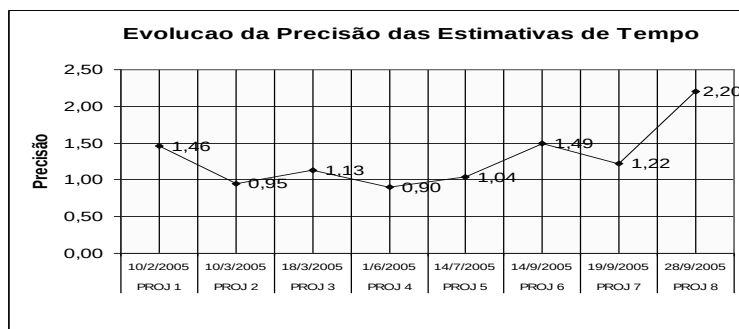


Figura 5. Evolução da Precisão das Estimativas de Tempo

5. Conclusão

A implantação de um modelo de maturidade com o CMMI, ou similar é fundamental para uma empresa produtora de software. O investimento para conseguir o laudo positivo tem um retorno claro quando a empresa passa a ter controle sobre o ciclo de vida de produção e passa a acompanhar as métricas e suas tendências, possibilitando a tomada de planos de ações preventivos, quando uma curva está tendenciosa para baixo, por exemplo.

Defendemos a idéia que é possível que uma empresa de 30 pessoas na fábrica, possa atingir um nível de maturidade internacional. Estamos cientes que o histórico não nos favorece nesta argumentação, visto que as empresas que tem obtido o CMMI são empresas de porte e com altos investimentos, mas a ZCR – Informática Ltda é prova que isto é possível.

O custo é um fator crítico, mas foram apresentadas alternativas para poder suavizar este investimento. Temos que ter claro, que devemos poder contar com políticas públicas de incentivo a obtenção destes laudos, assim foi com a Índia que mudou a sua economia por uma visão de longo prazo que o governo teve e pelas

políticas públicas que incentivaram e conseguiram mudar as empresas e a economia daquele país.

Os índices aqui discutidos obtidos tanto pelo PPQA, quanto principalmente pelo responsável pelo MA, são de fundamental importância para a gestão de uma empresa, sem eles torna-se mais difícil a tomada de decisões que envolvam produtividade, tempo, custo e esforço. Ficou claro na análise dos índices, uma melhora acentuada nas tendências das curvas, mostrando que o investimento no CMMI, tem gerado frutos positivos para a empresa. Também estas métricas, são armas poderosas para decidir a forma de agir com o cliente, e aumentar ou diminuir a negociação com o mesmo. A empresa deve ser comparável como uma fábrica, sem um PCP (Planejamento e Controle de Produção) qualquer indústria para, não é diferente para uma empresa de software. Se de fato queremos colocar o Brasil como *player* mundial de produção de software, o empresariado brasileiro deve deixar de ser amador, e passar a ser grande e a controlar a sua empresa, independente do tamanho que tenha. Este próprio relato mostrou que a empresa, achou o seu problema e passou a controlá-lo, e isto é sobrevivência da mesma, não há como controlar um problema, se não se sabe qual é, e isto só é conseguido com um nível de maturidade definido.

6. Referências Bibliográficas

CHRISSIS, Mary Beth; Konrad, Mike; Shrum, Sandy. (2003) CMMI – Guidelines for Process Integration and Product Improvement. Addison Wesley.

CMMI Product Team. (2002) CMMI for Software Engineering, Version 1.1, Staged Representation (CMMI-SW, V1.1, Staged). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

MALBOUISSON, Carlos Augusto Lima. (2006) Metodologia para Obtenção dos Custos Associados à Execução do Programa de Medição de uma Organização. Dissertação de Mestrado. Universidade Salvador - BA.

PORTER, Michael. (2000) A Nova Era da Estratégia. Revista HSM Management – Ed.Especial 2000, pág.17 a 28.

[6.19] RCCS - Rede de Compartilhamento de Componentes de Software

Entidade: 1 Laboratório de Inovação em Software - Ci&T Labs / UNICAMP Inovasoftware - Unicamp; 2 Instituto de Computação - Universidade Estadual de Campinas (UNICAMP) - Caixa Postal 6176 - 13.084-971 - Campinas - SP

Autores: Kleber Bacili¹, Ricardo Anido², Marcílio Oliveira¹⁻² - krbacili@cit.com.br, ranido@ic.unicamp.br, marcilio@cit.com.br

Abstract. *The Software Component Sharing Network is one of the results of a project of the Software Innovation Laboratory Ci&T-Unicamp. This is a Peerto-Peer network which is based on open patterns and can be used by any person or any entity interested in sharing or searching for components. A reference*

implementation has been developed using the J2EE platform and its related technologies such as Web Services, hibernate, lucene, among others.

This implementation has also been used as way of studying and validating new technologies.

Resumo. A RCCS, Rede de Compartilhamento de Componentes de Software, é fruto de um projeto do Laboratório de Inovação em Software Ci&t-Unicamp.

Trata-se uma rede Peer-to-Peer, baseada em padrões abertos e que pode ser utilizada por qualquer pessoa ou entidade interessada em buscar ou compartilhar componentes. Foi desenvolvida uma implementação de referência, utilizando a plataforma J2EE e tecnologias afins como Web Services, hibernate, lucene, dentre outras, visando também o estudo e validação destas novas tecnologias.

1. Introdução

A Rede de Compartilhamento de Componentes de Software (RCCS) é fruto de um projeto de pesquisa e desenvolvimento de solução tecnológica para viabilização de uma rede pública de distribuição e compartilhamento de componentes através de uma arquitetura independente de plataformas e da utilização dos padrões *Web Services*, em desenvolvimento no Laboratório de Inovação em Software Ci&T/Unicamp.

A RCCS é uma rede aberta, baseada em padrões abertos para comunicação, arquitetura e modelagem dos componentes e pretende atingir diversas categorias de entidades relacionadas a desenvolvimento e utilização de componentes de software. A figura 1 ilustra a forma com que a rede foi idealizada, com a participação de universidades e centros de pesquisa, órgãos governamentais, *hubs* de distribuição de software livre, fábricas de componentes, dentre outras entidades interessadas em compartilhar ou simplesmente buscar componentes na rede. Os pontos da rede podem se comunicar diretamente, ponto a ponto.

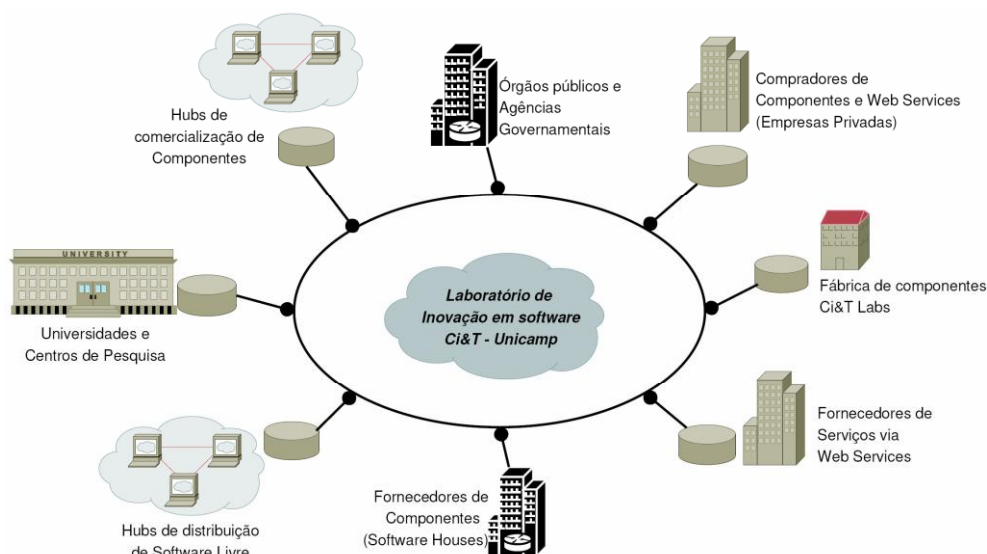


Figura 1. Arquitetura da RCCS

Conceitualmente, *Peer-to-Peer* (P2P) é uma alternativa aos modelos computacionais do tipo cliente-servidor. Através desta tecnologia, um sistema pode acessar diretamente recursos disponibilizados por outro, independente de terceiros atuando como servidores centrais.

2. Objetivo e justificativa

O principal objetivo do projeto é a criação de uma rede pública de busca e compartilhamento de componentes de software através de uma arquitetura distribuída e independente de plataforma.

Frente à necessidade de se construir software mais rápido e com mais qualidade em períodos cada vez mais curtos, o reuso de software pode ser visto como uma forte tendência para solucionar tais problemas. Reuso pode ser considerado uma denominação genérica, envolvendo uma variedade de técnicas utilizadas desde as etapas de design até a implementação. O principal objetivo é evitar o re-trabalho no desenvolvimento de um novo projeto, capitalizando trabalhos anteriores, fazendo com que as soluções já desenvolvidas sejam imediatamente implementadas em novos contextos. Desta forma, tem-se melhores produtos em um menor intervalo de tempo, com redução nos custos de manutenção pois as partes do sistema são independentes e facilitando a inclusão de novas funcionalidades. Além disso, a qualidade aumenta devido ao reuso de componentes previamente bem testados [D'Souza 99] [Jacobson97].

Uma das principais motivações na utilização de desenvolvimento baseado em componentes é exatamente a premissa de reutilização. Isto implica no

reaproveitamento de partes já desenvolvidas. Esta abordagem, por reduzir o tempo de desenvolvimento e aumentar a qualidade do produto final é, de fato, bastante atraente [Krueger92].

Para que seja possível reusar algo, é necessário saber onde está. É sob esta perspectiva que este trabalho foi desenvolvido. O principal desafio foi desenvolver uma arquitetura distribuída visando o compartilhamento de componentes de software, envolvendo também a concepção do repositório, dos serviços de compartilhamento, mecanismo de busca e descoberta de repositórios.

3. Metodologia de execução

Para atingir seu propósito, a rede foi idealizada considerando-se algumas premissas fundamentais (ou requisitos iniciais). E foi em torno destas premissas que as pesquisas e o desenvolvimento se realizaram. A seguir, são listados esses aspectos pré-definidos:

_ **Utilização de padrões.** A rede deve ser modelada seguindo padrões abertos [RAS06] e [IBM04] para facilitar e incentivar o surgimento de novas implementações e participações na rede.

_ **Interoperabilidade.** É importante que a rede não esteja restrita a uma plataforma de desenvolvimento. Como a comunicação deve ser possível entre pontos com diferentes implementações, é muito importante que a proposta proporcione interoperabilidade.

_ **Repositórios distribuídos e independência de entidade central.** Não deve existir um conjunto de repositórios centrais onde os componentes são armazenados. Estes repositórios estarão de forma distribuída, descentralizando os recursos da rede. Deste modo, a RCCS não deve ser dependente de nenhuma entidade específica e a queda de um ponto (ou simplesmente sua ausência na rede) não torne a rede indisponível momentaneamente.

_ **Escalabilidade.** Tendo em vista que a rede é aberta, é imprescindível que sua arquitetura suporte escalabilidade em relação ao número de participantes.

Vale ressaltar também que uma das diversas atividades executadas no projeto foi a validação de novas tecnologias na implementação da RCCS. As principais tecnologias pesquisadas e utilizadas no desenvolvimento da rede foram Arquiteturas Peer-to-Peer, SOA (*ServicesOriented Architecture*), RAS (*Reusable Asset Specification*) [RAS06], Lucene – *Text Search Engine*, AJAX (*Asynchronous Javascript And XML*) e JSF (*Java Server Faces*). Foram escritos *papers* sobre tais tecnologias e disponibilizados no site do Ci&T Labs (<http://labs.cit.com.br>) para acesso público.

3.1. Modelagem e arquitetura da rede

Um dos aspectos mais importantes é a modelagem da rede. Como dito anteriormente, trata-se de uma rede P2P, sem um servidor central. Mas isto não é suficiente. É necessário identificarmos a arquitetura, determinarmos como os pontos da rede se comunicarão, qual a tecnologia utilizada pra fazer a troca de informações e como estas informações serão transmitidas na RCCS.

Em cada repositório de componentes deve haver um servidor onde os serviços fiquem disponíveis e a comunicação é feita na forma de *Web Services*. A arquitetura permite que os pontos da rede comuniquem-se diretamente, sem a interferência ou dependência de um servidor central. A inexistência deste servidor central proporciona à rede uma característica interessante de independência de qualquer entidade.

Protocolo de comunicação e modelagem de objetos da RCCS

O protocolo de comunicação determina quais são e como serão representadas as informações transmitidas na rede. Como a comunicação é feita através de *Web Services*, as informações serão trocadas através de objetos, ou seja, o protocolo trata de quais objetos são transportados.

O modelo do componente foi baseado no RAS [RAS06], que é uma especificação proposta por um consórcio de empresas de tecnologia e adotada pela OMG (*Object Management Group* - <http://www.omg.org>) como padrão para especificação de componentes. É uma representação totalmente baseada em meta modelos. O principal objetivo do RAS é especificar a quantidade mínima de meta informações que um componente deve ter. Além disso, ele determina quais artefatos um componente possui, desde diagramas e casos de uso, até os arquivos de teste, código fonte e binário.

Possibilita também classificar estes artefatos. Podemos citar também alguns outros objetivos do RAS, como estimular o desenvolvimento baseado na utilização de componentes e aumentar a comunicação entre os produtores e os consumidores de componentes.

3.2. Funcionalidades principais da rede

Foi vislumbrada uma rede que pudesse trazer, de fato, funcionalidades que impulsionem e incentivem a utilização de componentes. Para isso, algumas funcionalidades são básicas e, sobretudo, essenciais. Pensando na forma mais natural de utilizarmos um sistema de compartilhamento, vimos que a busca de material é o primeiro passo.

Além de encontrar um componente, o usuário deve ter a opção por fazer o *download*. Para isso, a funcionalidade de *download* também se faz necessária. Visando agilizar a busca e diminuir o tráfego de dados, uma etapa é inserida entre a busca e o *download*: o *Asset Info*. Ou seja, uma funcionalidade para captura de informações sobre o *asset*/componente. A busca retornará informações básicas sobre os componentes encontrados, e, a partir disso, o usuário poderá requisitar maiores informações sobre o componente. E então, o *Asset Info* retornará todas as informações. A partir das informações fornecidas, o usuário poderá decidir por fazer ou não o *download* do componente. A busca e descoberta de repositórios é a parte mais relevante e é apresentada a seguir.

Busca de componentes

A busca é o primeiro passo para o funcionamento da rede. Podemos considerá-la como a parte mais importante do sistema, uma vez que ela determinará, muitas vezes, a qualidade do serviço. Uma busca deve ser simples e rápida, porém eficiente.

Tendo em vista que temos uma arquitetura *Peer-to-Peer* para a rede de compartilhamento e baseados nas duas principais arquiteturas existentes (pura e híbrida) [Walkerdine02], a arquitetura de busca deveria utilizar o que consideramos como pontos fortes de cada arquitetura existente.

No momento da busca, o usuário seleciona, entre um conjunto de repositórios conhecidos, onde deseja realizar a busca e, além disso, pode optar por buscar por novos repositórios em servidores *yellow page*, utilizando o protocolo de publicação de serviço dos *Web Services*, o UDDI (Figura 2a). Conhecendo os demais participantes da rede a busca é disparada para todos os repositórios – exceto os pesquisados (Figura 2b). Desta forma, cada *RAB* pode montar sua própria base de conhecimento, selecionando quais os repositórios serão utilizados para uma consulta futura.

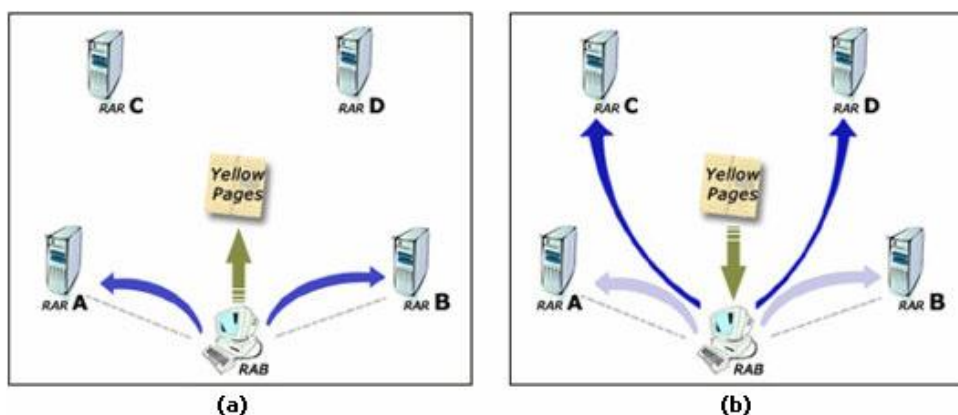


Figura 2. Arquitetura de busca na RCCS. [Oliveira05a]

O resultado da busca é apresentado com alguns detalhes. Os componentes retornados na busca são listados, apresentando informações relevantes, como repositório de origem, categoria, distribuição e descrição. Mais informações sobre as demais funcionalidades podem ser encontrados nos artigos publicados [Oliveira05b] ou no site do Ci&T Labs (<http://labs.cit.com.br>).

4. Resultados obtidos

4.1. Produtos de software gerados

A RCCS implementa uma ferramenta de compartilhamento de componentes de software, que podem ser instalados localmente e utilizados para disponibilizar e buscar componentes na rede. A figura 3 apresenta algumas interfaces da RCCS.



Figura 3. Interfaces RCCS (busca, resultado e detalhamento).

Foi implementado um plug-in para eclipse (www.eclipse.org) integrado à RCCS.

Através do plug-in é possível buscar, utilizando o eclipse, componentes de software distribuídos e integrá-los a um projeto ou biblioteca, fazendo *download* e instalando componentes instaláveis na aplicação.

O projeto da rede foi finalizado e integrado ao repositório de ativos digitais DigitalAssets Manager – DAM (<http://www.digitalassets.com.br>) possibilitando o compartilhamento de componentes entre implantações do DAM e da RCCS.

4.2. Outros produtos gerados (que foram disponibilizados para o mercado)

Foi desenvolvido e publicado um protocolo para compartilhamento de componentes de software, baseado no RAS. O documento de especificação está disponível no site do Ci&T Labs (<http://labs.cit.com.br>).

4.3. Artigos publicados

Além da realização de apresentação em eventos técnicos, como o III Fórum de Inovação Ci&T e na IX Semana de Informática UFV, o trabalho originou publicações nacionais e uma internacional:

_ O mecanismo de descoberta de repositórios (*Resource Discovery*) foi publicado no **I Workshop de P2P**, em Fortaleza-CE, sob o título de "Resource discovery em redes Peer-to-Peer aplicado à busca e compartilhamento de componentes de software" [Oliveira05a].

_ A rede de compartilhamento de componentes de software foi apresentada no **Salão de Ferramentas do 23º SBRC** [Oliveira05b]. Neste salão, além de uma apresentação formal, a RCCS foi instalada em diferentes computadores para realização de demonstrações práticas.

_ A integração da RCCS com o DigitalAssets Manager foi publicada no salão de demonstrações do **OOPSLA (21st Annual ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications)** [Bacili06].

Este evento é considerado um dos mais importantes do mundo na área de orientação de objetos e aplicações. Foi possível realizar exibições práticas do mecanismo da RCCS integrado a outra ferramenta, e foi possível demonstrar ambas realizando buscas distribuídas, uma vez que o protocolo de compartilhamento é o mesmo.

O perfil prático do projeto pôde ser afirmado através das publicações, que foram, em sua maioria, em salões de demonstração. Nestes salões foi possível instalar e permitir a interação dos participantes do evento com a ferramenta.

4.4. Recursos humanos capacitados (especialistas, mestres, doutores, etc.)

O projeto contou com pesquisadores do Instituto de Computação da Unicamp, consultores e gerentes de projeto da Ci&T e com alunos de graduação e mestrado. Um dos principais resultados do projeto foi a capacitação de pessoal, gerando especialistas em nas tecnologias relacionadas ao projeto, sobretudo em SOA (*Service Oriented Architecture*), arquiteturas Peer-to-Peer, além de um mestre em ciência da computação, que teve a RCCS como tema de projeto de mestrado (Busca e Compartilhamento de Componentes de Software em Redes Peer-to-Peer).

4.5. Dissertações e/ou teses geradas

Foi gerada uma tese de mestrado, desenvolvida pelo membro do laboratório Marcílio da Silva Oliveira, no Instituto de Computação (IC) da Unicamp, sob a orientação da Prof^a. Dr^a. Islene Garcia. A dissertação foi defendida em 15 de dezembro de 2006, no IC.

4.6. Eventuais parcerias ou programas de transferência de tecnologia efetuados

O Ci&T Labs tem parceria com diversas instituições de pesquisa, e vêm realizando projetos em conjunto com C.E.S.A.R., UFPE, UNICAMP, UFPB, IBTA, CenPRA e UFV. O projeto da RCCS foi desenvolvido juntamente com a Unicamp, e algumas ferramentas (como o plug-in) foram desenvolvidas no laboratório da UFV, em Viçosa/MG. Na equipe da UFV, além das ferramentas, foram realizados trabalhos acadêmicos sobre a Rede de Compartilhamento. Objetiva-se a aplicação de resultados da RCCS em novos projetos de parceria entre instituições.

5. Aplicabilidade dos resultados

O projeto desenvolveu, além de um protocolo de compartilhamento, uma ferramenta para manter localmente e compartilhar componentes de software. Atualmente os resultados já são aplicados na prática. A RCCS está integrada à principal ferramenta nacional de gerenciamento de ativos digitais (Prêmio B2B de qualidade: http://www.digitalassets.com.br/release_12.html). Além disso, a RCCS

já possui pontos remotos na Unicamp (Ci&T Labs), UFV (Depto. de Informática) e Ci&T.

O modelo de repositório definido também pode ser aplicado em outras ferramentas. Foi definido um padrão de banco de dados baseado no RAS e pode ser implementados em diferentes bases de dados. Isso já vem acontecendo, e ferramentas como DigitalAssets Manager já utilizam parte do repositório da RCCS para representar a flexibilidade de atributos de ativos digitais.

Também foi implementado um mecanismo de empacotamento de ativos em modelo RAS (Export-RAS). Este mecanismo gera o pacote “.ras”, contendo as informações os artefatos do componente, conforme definido na especificação [RAS06].

Os principais gerenciadores de ativos internacionais utilizam este padrão para intercâmbio de componentes, e essa funcionalidade alavanca as ferramentas nacionais a um padrão internacional de representatividade de ativos, que muitas vezes é requisito obrigatório para escolha de ferramentas.

Além da utilização da arquitetura da rede em outras ferramentas, o repositório desenvolvido também é utilizado em outros projetos. Atualmente, o modelo de repositório e banco de dados é utilizado no projeto de construção de um **Identificador Automático de Componentes de Software (IACS)**. O foco central do projeto consiste em montar um mecanismo capaz de reconhecer, em parques de aplicações já desenvolvidas, grupos de artefatos que componham componentes. Visando, sobretudo, a extração destes componentes para catalogação e reuso.

6. Características inovadoras

Relacionado às características inovadoras, podemos citar:

_ Modelagem e implementação de um protocolo específico para transmissão de componentes utilizando serviços independentes de plataforma (*Web Services*)
_ Implementação de arquitetura híbrida Peer-to-Peer de compartilhamento de componentes de software, utilizando bases de conhecimento locais e servidores de referência (*Yellow Pages*).

_ Desenvolvimento de ferramenta de gerenciamento e compartilhamento P2P de componentes. Apesar de existirem muitos exemplos de softwares de compartilhamentos de conteúdos, esta é a primeira solução específica para componentes de software, integrando mecanismos de busca, comunicação e transferência de componentes.

_ Modelagem e implementação de um modelo extensível de repositório compatível com o padrão OMG (RAS). Este modelo pode ser replicado em diferentes ferramentas/contextos.

7. Conclusão e Perspectivas futuras

As ferramentas de compartilhamento de conteúdo sempre foram vistas como ferramentas para *download* de música e vídeos. A proposta deste projeto foi de utilizar o conceito de aplicações P2P em prol da produtividade dos

desenvolvedores, visando facilitar o intercâmbio de informações úteis, neste caso componentes de software. Ou seja, foi possível utilizar tecnologias e arquiteturas já existentes, em um contexto diferente, proporcionando uma solução específica para novas necessidades.

Além de criar uma rede para compartilhamento de componentes, o projeto apresentou uma série de efeitos positivos, desde o fomento de novas tecnologias nacionalmente à promoção do reuso de componentes no desenvolvimento de software.

Referências

[Bacili06] Bacili K. R. and Oliveira M. S. *Digitalassets manager, managing and sharing software components*. OOPSLA - 21st Annual ACM Conference on Object-Oriented Programming, Systems,

[D'Souza99] D. F. D'Souza and A. C. Wills. *Objects, Components, and Frameworks With UML: The Catalysis Approach*. Addison Wesley. USA., 1999.

[IBM04] *Technology options for application integration and extended enterprise patterns*. IBM Red books, 2004. Disp. Em: <http://www.redbooks.ibm.com/redpapers/pdfs/redp3840.pdf>. Último acesso em 10/02/2006.

[Jacobson97] L. Jacobson, M. Griss, and P. Jonsson. *Software Reuse: Architecture, Process and Organization for Business Success*. Addison Wesley. USA., 1997.

[Krueger92] C. W. Krueger. Software Reuse. *ACM Computing Surveys*, 24(02):131{182, 1992.

[Oliveira05a] M. S. Oliveira, I. Garcia, e A. Nunes. *Resource Discovery em uma rede Peer-to-Peer aplicado a busca e compartilhamento de componentes de software*. I Workshop de Peer-to-Peer- XXIII Simpósio Brasileiro de Redes de Computadores - Fortaleza, CE - maio de 2005, 2005.

[Oliveira05b] M. S. Oliveira, I. Garcia, e A. Nunes. *RCCS, Rede de Compartilhamento de Componentes de Software*. X Salão de Ferramentas - XXIII Simpósio Brasileiro de Redes de Computadores - Fortaleza, CE - maio de 2005, 2005.

[RAS06] RAS. *Reusable Asset Specification, OMG Available Specification*. Object Management Group, 2006. Disponível em <http://www.omg.org/technology/documents/formal/ras.htm>. Último acesso em 01/12/2006.

[Walkerdine02] Walkerdine, J., Melville, L., Sommerville, I., *Dependability Properties of P2P Architectures*, In the Second International Conference on Peer-to-Peer Computing (P2P'02)

[6.22] Desenvolvimento de Recursos para Treinamento e Capacitação da Indústria Nacional em Teste e Validação de Software

Entidade: 1Departamento de Sistemas de Computação - ICMC/USP, São Carlos (SP); 2Departamento de Computação - UFSCar, São Carlos (SP)

Autores: Ellen Francine Barbosa¹, José Carlos Maldonado¹, Sandra Fabbri² - {francine,jcmaldon}@icmc.usp.br, sfabbri@dc.ufscar.br

1. Introdução

A demanda por formação e capacitação de pessoal, tanto no meio acadêmico – apoiando o estudo e o domínio de novos conhecimentos – como no âmbito industrial – oferecendo atividades de treinamento, complementação curricular e formação continuada na disseminação e aplicação de conhecimentos consolidados –, tem impulsionado a pesquisa em temas relacionados ao Ensino e Aprendizado nas mais diversas áreas de atuação.

Uma das linhas que tem sido objeto de intensa pesquisa refere-se ao desenvolvimento de módulos educacionais – unidades concisas de estudo, compostas por conteúdos teóricos integrados a atividades práticas e avaliações, cuja disponibilização aos aprendizes é apoiada por recursos tecnológicos e computacionais [Barbosa 2004]. Em termos de seus componentes principais um módulo educacional é composto por: (1) material didático associado, envolvendo os conteúdos referentes ao domínio de conhecimento que se deseja ensinar; e (2) ambientes educacionais e mecanismos de apoio, representando a infra-estrutura necessária à utilização do módulo.

Dentro desse cenário, no decorrer das atividades associadas ao trabalho de doutorado de Barbosa (Processo FAPESP 98/16490-5) [Barbosa 2004] foram investigadas linhas de pesquisa relacionadas à área de Ensino e Aprendizado, com ênfase na definição de mecanismos de apoio ao processo de desenvolvimento e modelagem de módulos educacionais.

Os mecanismos estabelecidos foram aplicados no decorrer do presente projeto, inserido no contexto do *Programa Brasileiro da Qualidade e Produtividade* (PBQP), a fim de propiciar a reestruturação de materiais didáticos na área de Engenharia de Software, mais especificamente dentro da temática de teste de software. Em linhas gerais, os aspectos de modelagem foram utilizados na estruturação e organização do domínio de conhecimento sobre teste, sendo que os processos definidos serviram como mecanismos de apoio para o desenvolvimento sistemático de módulos educacionais nesse domínio.

2. Objetivos e Justificativas

Entre as áreas de conhecimento tidas como essenciais dentro da Engenharia de Software, a atividade de teste de software desempenha um papel fundamental na temática de qualidade, sendo fundamental para a avaliação e a melhoria dos produtos de software desenvolvidos [da Rocha et al. 2001, Pressman 2001].

Ressalta-se, entretanto, que apesar dos avanços e pesquisas na área – definição e estudo de critérios de teste, especificação e implementação de ferramentas de apoio e condução de estudos teóricos e empíricos –, a utilização sistemática de teste em ambientes de desenvolvimento ainda não é uma prática comum. No

contexto nacional, pesquisas realizadas pelo Ministério da Ciência e Tecnologia [DSI/CGSA 1998, DSI/CGSA 2000, DSI/CGSA 2002], indicaram que apenas parte das empresas desenvolvedoras de software no país utilizam algum tipo de teste como método para detecção de erros. De modo geral, os resultados obtidos demonstram a dificuldade da indústria em incorporar e aplicar em seu processo de desenvolvimento tanto recursos já disponíveis comercialmente como novas tecnologias investigadas no ambiente acadêmico.

Além disso, a percepção de que o teste é uma atividade indispensável dentro do processo de desenvolvimento tem provocado um crescimento significativo na demanda por pessoal capacitado, sobretudo no ambiente industrial. No entanto, apesar de algumas iniciativas serem identificadas, o meio acadêmico não tem conseguido atender satisfatoriamente à demanda existente. Como exemplos de universidades que oferecem a seus alunos cursos específicos envolvendo teste de software e garantia de qualidade podem-se destacar a *George Mason University*, a *Kansas State University* e a *Purdue University*, esta última possuindo dois cursos de graduação em Engenharia de Software abordando teste e confiabilidade. No Brasil, embora exista uma grande quantidade de cursos de Computação/Informática no país, poucos deles possuem uma disciplina especificamente voltada ao ensino dos conceitos relacionados à atividade de teste. No caso do ICMC-USP, duas disciplinas no domínio de teste (*VV&T: Verificação, Validação e Teste de Software* e *Teste e Inspeção de Software*) são oferecidas aos alunos de graduação, sendo abordados tanto aspectos teóricos como práticos da atividade de teste.

Dentro da perspectiva apresentada, o presente projeto procurou definir e estabelecer mecanismos de apoio ao processo de ensino e aprendizado em teste de software, favorecendo o domínio e a disseminação do conhecimento na área. Objetivou-se, com isso, fornecer meios para a formação e a capacitação de pessoal tanto em conceitos de teste já consolidados em alguns setores da indústria quanto em tecnologias emergentes, estimulando o processo de transferência tecnológica entre universidade e indústria e contribuindo para a melhoria da qualidade dos produtos de software desenvolvidos.

3. Metodologia de Execução

As principais atividades e tarefas conduzidas durante o projeto são sintetizadas a seguir.

- Levantamento, estudo e avaliação de ambientes e sistemas educacionais, caracterizando, em especial, os requisitos necessários para apoiar a atividade de modelagem de conteúdos educacionais.
- Estudo e definição de mecanismos para a modelagem de conteúdos educacionais, estabelecendo subsídios para a identificação e estruturação das partes relevantes do domínio de conhecimento a serem ensinadas.
- Definição de parâmetros para comparação e seleção de modelos para representação de conteúdos educacionais e estabelecimento de uma abordagem de modelagem – *AIM/CID* (Abordagem Integrada de Modelagem / Conceitual,

Instrucional e Didática) –, procurando ressaltar pontos fortes e minimizar limitações identificadas nas abordagens existentes.

- Definição de um processo padrão para a elaboração de módulos educacionais, fornecendo diretrizes para o desenvolvimento sistemático dos módulos.
- Estudo de mecanismos para apoiar a especialização e instanciação do processo padrão definido e definição de um modelo de maturidade de processos para a elaboração de módulos educacionais – *CMMI/Educacional*.
- Instanciação do processo padrão definido e aplicação prática do processo instanciado no desenvolvimento de módulos educacionais no domínio de conhecimento sobre teste de software.
- Desenvolvimento, disponibilização e avaliação de módulos educacionais no domínio de teste de software, a partir da aplicação dos mecanismos de modelagem e padronização estabelecidos anteriormente.
- Desenvolvimento de um ambiente web para submissão e avaliação automática de trabalhos práticos com base em atividades de teste – *ProgTest*.

4. Resultados Obtidos

Os principais resultados obtidos a partir da execução do projeto são sintetizados a seguir.

Foram considerados *Produtos Desenvolvidos*, *Artigos Publicados* e *Dissertações e Teses Geradas*.

4.1. Produtos Desenvolvidos

4.1.1. Módulo Educacional *Teste de Software: Teoria e Prática*

Conceitos, fatos, princípios, procedimentos, exemplos, informações complementares e exercícios pertinentes ao domínio de teste de software foram estruturados, modelados e implementados na forma de um conjunto de transparências, ao qual foram integradas páginas HTML, documentos-texto, ferramentas específicas do domínio e ferramentas educacionais.

A transparência principal do módulo, ilustrando a integração entre seus diversos componentes, é apresentada na Figura 1.

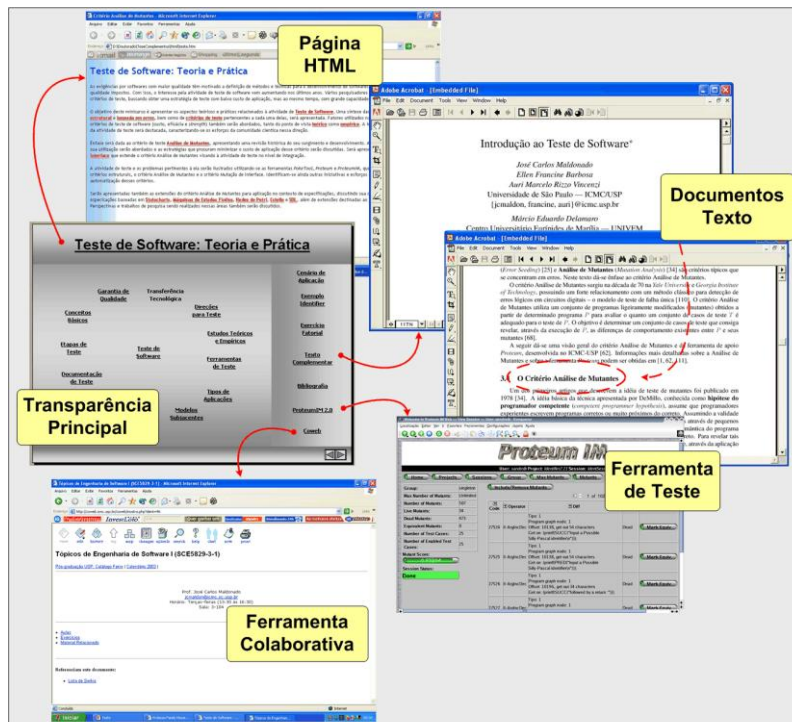


Figura 1. Transparência Principal do Módulo Teste de Software: Teoria e Prática.

O módulo foi disponibilizado na forma de palestra, proferida durante o *II Simpósio Brasileiro de Qualidade de Software (SBQS 2003)*, e na forma de minicursos, ministrados respectivamente no *XVII Simpósio Brasileiro de Engenharia de Software (SBES 2003)*, na *V Escola de Informática Norte (ERI – Norte)* e na *IV Escola Regional de Informática de Minas Gerais (ERI-MG 2005)*. Avaliações informais referente às características técnicas e pedagógicas foram conduzidas, possibilitando a identificação de problemas e melhorias associadas.

Além disso, no primeiro semestre de 2006, o módulo foi utilizado no contexto da disciplina *SCE-221 – Verificação, Validação e Teste de Software*, oferecida aos alunos de graduação em Ciências de Computação do ICMC/USP. Os resultados observados a partir da utilização do módulo estão sendo sintetizados em um artigo a ser submetido ao *SBQS 2007*. Atualmente, o módulo vem sendo utilizado na disciplina *SCE-702 – Teste e Inspeção de Software*, oferecida aos alunos de Engenharia de Computação do ICMC/USP.

4.1.2. Módulo Educacional Integrado de Fundamentos de Programação OO e Teste de Software

Um módulo educacional integrado abordando os principais tópicos de fundamentos de programação OO e teste de software também foi desenvolvido no contexto de um trabalho de mestrado (Processo FAPESP 03/04567-3) [Corte 2006]. O módulo integrado é composto de dois outros módulos – um para

fundamentos de programação e outro para teste de software –, os quais foram integrados por meio de pontos de conexão em locais específicos de cada um dos módulos. Um cenário de navegação entre os módulos é ilustrado na Figura 2.

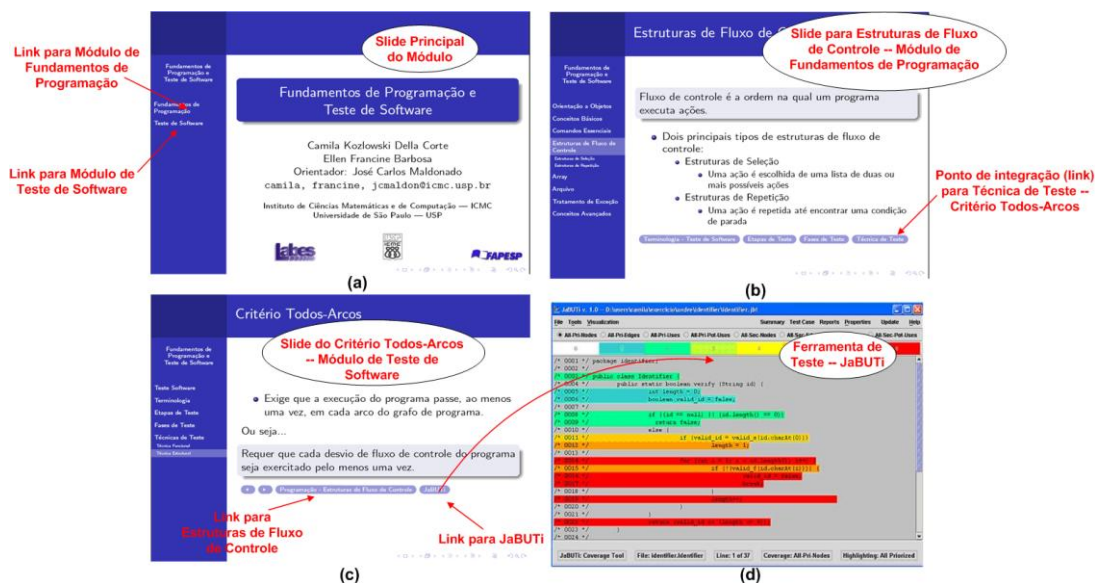


Figura 2. Cenário de Navegação entre os Módulos de Programação e de Teste

Cada módulo é implementado como um conjunto de slides e documentos-texto integrados a ferramentas educacionais e ferramentas de suporte específicas ao domínio em questão. No caso de teste de software, a ferramenta JaBUTi (Java Bytecode Understanding and Testing) [Vincenzi 2004], que apóia análise de cobertura para testar programas e componentes Java, foi incorporada ao módulo.

O módulo encontra-se em fase de validação, devendo ser disponibilizado em breve. Para isso, um experimento sistemático e controlado envolvendo estudantes de graduação do ICMC/USP vem sendo planejado.

4.1.3. ProgTest – Ambiente de Submissão e Avaliação de Trabalhos Práticos

Outro produto resultante do trabalho de mestrado mencionado anteriormente refere-se ao desenvolvimento de um ambiente para submissão e avaliação automática de trabalhos práticos – PROGTEST. O ambiente é baseado na web e na concepção de software livre.

A avaliação dos trabalhos dos alunos é realizada com base em atividades de teste e, para isso, ferramentas de teste são integradas ao ambiente. Em seu atual estágio de desenvolvimento, foi integrada à PROGTEST a ferramenta de teste JaBUTi [Vincenzi 2004].

A utilização de um ambiente como PROGTEST possibilita que os programas sejam avaliados no mesmo nível de eficiência e os resultados da avaliação sejam baseados nos mesmos padrões. Além disso, os professores podem ter uma redução na sua carga de trabalho, uma vez que o próprio sistema avalia automaticamente tanto os casos de teste quanto o código do programa. Ainda, após a avaliação o sistema gera relatórios para cada programa (Figura 3), fornecendo aos alunos *feedback* a respeito de seu desempenho.

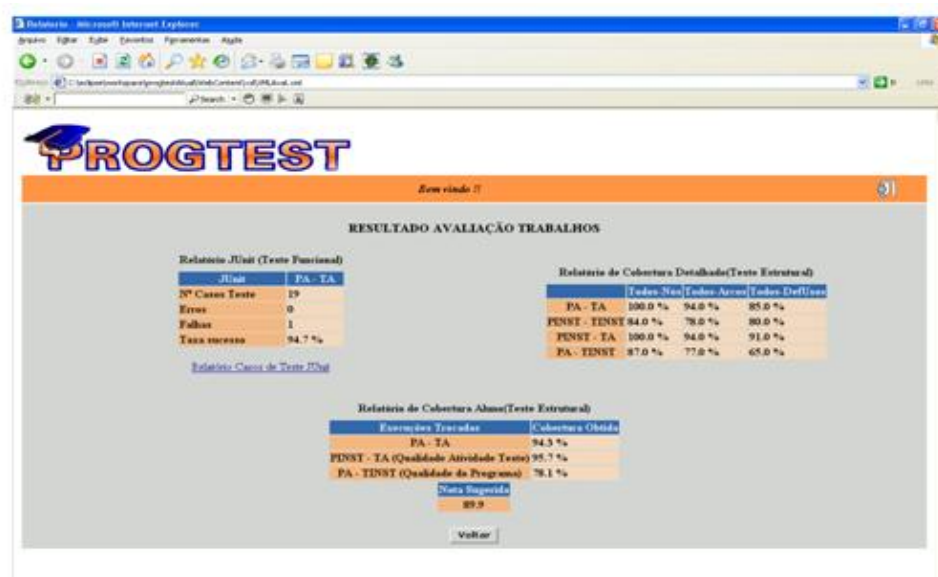


Figura 3. Ambiente PROGTEST – Relatórios Gerados

O ambiente encontra-se em fase final de implementação e deverá ser disponibilizado em conjunto com o módulo educacional integrado discutido na seção anterior.

4.2. Publicações Relacionadas

- Barbosa, E. and Maldonado, J. (2006a). Establishing a mutation testing educational module based on IMA-CID. In *2nd Workshop on Mutation Analysis (Mutation 2006)*, Raleigh, NC. CD-ROM.
- Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006b). Integrated teaching of programming foundations and software testing. In *17th IEEE International Symposium on Software Reliability Engineering (ISSRE 2006) – (Fast Abstract)*, Raleigh, NC. CD-ROM.
- Barbosa, E. and Maldonado, J. (2006c). Towards the establishment of a standard process for developing educational modules. In *36th Annual Frontiers in Education Conference (FIE 2006)*, San Diego, CA. CD-ROM.
- Barbosa, E. and Maldonado, J. (2006b). An integrated content modeling approach for educational modules. In *IFIP 19th World Computer Congress –*

International Conference on Education for the 21st Century, pages 17–26, Santiago, Chile.

5. Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006c). Integrated teaching of programming foundations and software testing. In *IFIP 19th World Computer Congress – International Conference on Education for the 21st Century (Pôster Session)*, page 425, Santiago, Chile.

6. Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006a). Estabelecimento de mecanismos de apoio ao ensino integrado de fundamentos de programação e teste de software. In *XXVI Congresso da Sociedade Brasileira de Computação (SBC 2006) – XIV Workshop de Educação em Computação (WEI 2006)*, Campo Grande, MS. CD-ROM.

7. Barbosa, E., Vincenzi, A., Delamaro, M., and Maldonado, J. (2005). Teste estrutural e de mutação no contexto de programas oo. In *IV Escola Regional de Informática de Minas Gerais (ERI-MG)*, pages 313–362, Belo Horizonte, MG.

8. Barbosa, E. F. and Maldonado, J. C. (2004). Mecanismos de apoio à modelagem de conteúdos: uma contribuição ao processo de desenvolvimento de módulos educacionais. In *XV Simpósio Brasileiro de Informática na Educação (SBIE 2004)*, Manaus, AM. CD-ROM.

9. Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2004). Ensino integrado de fundamentos de programação e teste de software. In *XXIV Congresso da Sociedade Brasileira de Computação (SBC 2004) – XII Workshop de Educação em Computação (WEI 2004)*, Salvador, BA. CD-ROM.

10. Barbosa, E. F., Maldonado, J. C., and Maidantchik, C. L. L. (2003c). Padronização de processos para o desenvolvimento de módulos educacionais. In *XXIX Latin- American Conference on Informatics (CLEI 2003)*, La Paz, Bolívia. CD-ROM.

11. Barbosa, E. F., Maldonado, J. C., and Ricarte, I. L. M. (2003d). Exploring learning objects under conceptual, instructional and didactic perspectives. In *I PGL Data Base Research Conference (PGL DB 2003)*, Rio de Janeiro, RJ.

12. Barbosa, E. F., LeBlanc, R., Guzdial, M., and Maldonado, J. C. (2003b). Introducing testing practices into objects and design course. In *16th Conference on Software Engineering Education and Training (CSEET 2003)*, Madrid, Spain.

13. Barbosa, E. F., LeBlanc, R., Guzdial, M., and Maldonado, J. C. (2003a). The challenge of teaching software testing earlier into design courses. In *II Workshop on the Teaching of Software Testing (WTST 2003)*, Melbourne, FL.

14. Barbosa, E. F., Maldonado, J. C., and Ricarte, I. L. M. (2002). Learning materials: Towards the establishment of guidelines for domain modeling. In *Informatics Curricula, TEaching Methods and best practice Working Conference (ICTEM 2002)*, Florianópolis, SC.

15. Barbosa, E. F., Adriano, C. M., Maldonado, J. C., Ricarte, I. L. M., and Jino, M. (2000). Fostering theoretical, empirical and tool specific knowledge in a software

testing learning scenario. In *International Conference on Engineering and Computer Education (ICECE 2000)*, São Paulo, SP.

4.3. Dissertações e Teses Geradas

1. Corte, C. K. D. (2006). Ensino integrado de fundamentos de programação e teste de software. Master's thesis, ICMC/USP, São Carlos, SP.
2. Barbosa, E. F. (2004). *Uma Contribuição ao Processo de Desenvolvimento e Modelagem de Módulos Educacionais*. PhD thesis, ICMC-USP, São Carlos, SP.

5. Aplicabilidade dos Resultados

Os aspectos observados durante o desenvolvimento e utilização dos módulos educacionais evidenciaram a viabilidade de aplicação dos mecanismos de apoio propostos, tanto no que se refere à modelagem de conteúdos como no contexto de padronização de processos para a elaboração dos módulos. Nesse sentido, pretende-se continuar a aplicação e avaliação de tais mecanismos em projetos mais abrangentes e com características diferenciadas, visando a elaboração de material didático e de capacitação tecnológica também em outros domínios, sobretudo aqueles relacionados à Engenharia de Software.

Ressalta-se, ainda, a viabilidade de aplicação dos módulos desenvolvidos para o domínio de teste de software não somente em contextos acadêmicos mas também em ambientes reais de desenvolvimento. Cursos para capacitação da indústria tendo como base tais módulos educacionais vêm sendo planejados com esse objetivo.

6. Características Inovadoras

Como principais características inovadoras do projeto têm-se:

- Proposição da abordagem *AIM-CID* (Abordagem Integrada de Modelagem – Conceitual, Instrucional e Didática), estabelecendo um conjunto de modelos genéricos – Modelo Conceitual, Modelo Instrucional e Modelo Didático – para a representação de conteúdos educacionais.
- Definição de um processo padrão para a elaboração de módulos educacionais, adaptado a partir da norma ISO/IEC 12207, incorporando práticas de projeto instrucional e modelagem de conteúdos educacionais.
- Definição de um modelo de maturidade de processos para a elaboração de módulos educacionais – *CMMI/Educacional*.
- Instanciação do processo padrão definido e aplicação prática do processo instanciado no desenvolvimento de módulos educacionais específicos no domínio de conhecimento sobre teste de software.

7. Conclusão e Perspectivas Futuras

Em continuidade às atividades conduzidas, podem-se destacar como perspectivas futuras de pesquisa:

- Evolução da abordagem *AIM-CID* e do processo padrão, considerando aspectos de desenvolvimento aberto, cooperativo e distribuído nos contextos de ensino presencial e a distância.
- Especificação e implementação de uma ferramenta de apoio à modelagem de conteúdos educacionais, tendo como base os modelos estabelecidos na abordagem *AIM-CID*.
- Desenvolvimento de módulos educacionais no domínio de teste de software, incorporando características específicas para sua disponibilização em cursos a distância.
- Os módulos desenvolvidos deverão ser disponibilizados na *Incubadora Virtual* do Projeto *TIDIA*.
- Aplicação dos mecanismos de apoio identificados em outros domínios da Engenharia de Software, resultando na reestruturação dos materiais didáticos associados e no desenvolvimento de módulos educacionais pertinentes.

Referências

Barbosa, E. and Maldonado, J. (2006a). Establishing a mutation testing educational module based on IMA-CID. In *2nd Workshop on Mutation Analysis (Mutation 2006)*, Raleigh, NC. CD-ROM.

Barbosa, E. and Maldonado, J. (2006b). An integrated content modeling approach for educational modules. In *IFIP 19th World Computer Congress – International Conference on Education for the 21st Century*, pages 17–26, Santiago, Chile.

Barbosa, E. and Maldonado, J. (2006c). Towards the establishment of a standard process for developing educational modules. In *36th Annual Frontiers in Education Conference (FIE 2006)*, San Diego, CA. CD-ROM.

Barbosa, E., Vincenzi, A., Delamaro, M., and Maldonado, J. (2005). Teste estrutural e de mutação no contexto de programas oo. In *IV Escola Regional de Informática de Minas Gerais (ERI-MG)*, pages 313–362, Belo Horizonte, MG.

Barbosa, E. F. (2004). *Uma Contribuição ao Processo de Desenvolvimento e Modelagem de Módulos Educacionais*. PhD thesis, ICMC-USP, São Carlos, SP.

Barbosa, E. F., Adriano, C. M., Maldonado, J. C., Ricarte, I. L. M., and Jino, M. (2000). Fostering theoretical, empirical and tool specific knowledge in a software testing learning scenario. In *International Conference on Engineering and Computer Education (ICECE 2000)*, São Paulo, SP.

Barbosa, E. F., LeBlanc, R., Guzdial, M., and Maldonado, J. C. (2003a). The challenge of teaching software testing earlier into design courses. In *II Workshop on the Teaching of Software Testing (WTST 2003)*, Melbourne, FL.

Barbosa, E. F., LeBlanc, R., Guzdial, M., and Maldonado, J. C. (2003b). Introducing testing practices into objects and design course. In *16th Conference on Software Engineering Education and Training (CSEET 2003)*, Madrid, Spain.

Barbosa, E. F. and Maldonado, J. C. (2004). Mecanismos de apoio à modelagem de conteúdos: uma contribuição ao processo de desenvolvimento de módulos educacionais. In *XV Simpósio Brasileiro de Informática na Educação (SBIE 2004)*, Manaus, AM. CD-ROM.

Barbosa, E. F., Maldonado, J. C., and Maidantchik, C. L. L. (2003c). Padronização de processos para o desenvolvimento de módulos educacionais. In *XXIX Latin-American Conference on Informatics (CLEI 2003)*, La Paz, Bolívia. CD-ROM.

Barbosa, E. F., Maldonado, J. C., and Ricarte, I. L. M. (2002). Learning materials: Towards the establishment of guidelines for domain modeling. In *Informatics Curricula, TEaching Methods and best practice Working Conference (ICTEM 2002)*, Florianópolis, SC.

Barbosa, E. F., Maldonado, J. C., and Ricarte, I. L. M. (2003d). Exploring learning objects under conceptual, instructional and didactic perspectives. In *I PGL Data Base Research Conference (PGL DB 2003)*, Rio de Janeiro, RJ.

Corte, C. K. D. (2006). Ensino integrado de fundamentos de programação e teste de software. Master's thesis, ICMC/USP, São Carlos, SP.

Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2004). Ensino integrado de fundamentos de programação e teste de software. In *XXIV Congresso da Sociedade Brasileira de Computação (SBC 2004) – XII Workshop de Educação em Computação (WEI 2004)*, Salvador, BA. CD-ROM.

Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006a). Estabelecimento de mecanismos de apoio ao ensino integrado de fundamentos de programação e teste de software. In *XXVI Congresso da Sociedade Brasileira de Computação (SBC 2006) – XIV Workshop de Educação em Computação (WEI 2006)*, Campo Grande, MS. CD-ROM.

Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006b). Integrated teaching of programming foundations and software testing. In *17th IEEE International Symposium on Software Reliability Engineering (ISSRE 2006) – (Fast Abstract)*, Raleigh, NC. CDROM.

Corte, C. K. D., Barbosa, E. F., and Maldonado, J. C. (2006c). Integrated teaching of programming foundations and software testing. In *IFIP 19th World Computer Congress – International Conference on Education for the 21st Century (Poster Session)*, page 425, Santiago, Chile.

da Rocha, A. R. C., Maldonado, J. C., and Weber, K. C. (2001). *Qualidade de Software: Teoria e Prática*. Prentice Hall, São Paulo, SP.

DSI/CGSA (1998). Qualidade no setor de software brasileiro: 1997. Revista do SEPIN/ MCT. Brasília, DF.

DSI/CGSA (2000). Qualidade no setor de software brasileiro: 1999. Revista do SEPIN/ MCT. Brasília, DF.

DSI/CGSA (2002). Qualidade e produtividade no setor de software brasileiro: 2001. Revista do SEPIN/MCT. Brasília, DF.

Pressman, R. S. (2001). *Software Engineering - A Practitioner's Approach*. McGraw-Hill, 5 edition.

Vincenzi, A. M. R. (2004). *Orientação a Objetos: Definição, Implementação e Análise de Recursos de Teste e Validação*. PhD thesis, Instituto de Ciências Matemáticas e de Computação – ICMC-USP, São Carlos, SP.

[6.25] Uma Solução para Gerência Integrada de Requisitos e de Configuração, Facilmente Replicável, Baseada em Softwares Livres

Entidade: ¹Icaro Technologies– Campinas - SP - Brasil; ²ASR Consultoria e Assistência em Qualidade - São Paulo - SP - Brasil

Autores: Elena Balachova¹, Kleber Stroeh¹, Ana Cecília Peixoto Zabeu² -
elena.balachova@icaro.com.br, kleber.stroeh@icaro.com.br,
anazabeu@asrconsultoria.com.br

Abstract. *Many companies face difficulties during the Configuration Management and Requirements Management processes implantation. To help to solve this difficulty, there were developed a system to help the implantation and future use of processes of Configuration Management and Requirements Management. The developed solution is low-cost to implement as well as to maintain and is highly accepted by employees of the pilot firm.*

Resumo. *Muitas empresas enfrentam dificuldades durante a implantação de processos de gerência de configuração e requisitos. Visando endereçar este desafio, desenvolveu-se um sistema que auxiliasse na implantação e posterior uso dos processos de Gerência de Requisitos e Gerência de Configuração segundo modelo CMMI-SW [CMMI]. A solução desenvolvida apresentou baixo custo de implantação e de manutenção, e alta aceitação pelos funcionários da empresa piloto.*

1. Introdução

A Ícaro Technologies, fundada em 1997, integradora líder em soluções de gerência, atua nas áreas de gerência de redes e sistemas, gerência de segurança e gerência de processos de negócios (BPM). Possui clientes em mais de 10 países e diversos casos de sucesso no Brasil e exterior. Suas soluções buscam forte aderência aos objetivos de negócio dos clientes, o que tem contribuído decisivamente para o elevado índice de fidelização alcançado pela empresa. A Ícaro Technologies tem entre seus clientes nomes como Carrefour, Claro, Copel, CSFB, GTech, IBM, Impsat, Nextel México, Roche, Smart.Net, TecBan, Telefônica, Telemar, Telmex, entre outros.

Buscando constante crescimento e alta qualidade em seus produtos e serviços sentiu-se a necessidade de controlar melhor os seus processos. Para tal tarefa foi escolhido o modelo CMMI-SW nível 3 [CMMI] que foi implantado com o auxílio da ASR Consultoria e Assessoria em Qualidade. Uma das áreas chaves mais críticas identificadas foi a Gerência de Configuração e Gerência de Requisitos. Ao estudar melhor o problema foi decidido que seria implantado um

sistema que auxiliasse nesses gerenciamentos. Esse sistema deveria possuir uma alta aceitação dos funcionários para ser facilmente difundível em toda a empresa e deveria ter os custos de desenvolvimento e manutenção baixos.

2. Objetivos e Justificativa

O objetivo principal do trabalho é desenvolver e obter um aplicativo de baixo custo e eficaz que integrasse, de forma completa, as disciplinas de gerência de requisitos e gerência de configuração. Este aplicativo deveria, também, ser aplicável ao desenvolvimento e integração de software, para agilizar os processos envolvidos, facilitar a visualização dos requisitos do cliente e melhorar a garantia do atendimento da integridade do produto frente aos mesmos.

As ferramentas disponíveis no mercado possuem atuação limitada para o gerenciamento integrado entre requisitos de software e sua gerência de mudanças e de configuração, e apresentam preços elevados para a maioria das empresas que desenvolvem e mantêm produtos de software. Assim sendo, o desenvolvimento de um aplicativo utilizando ferramentas de código aberto que possibilite a gerência de requisitos de forma eficaz e liberação segura de versões dos produtos agregaria muito valor para a comunidade.

3. Metodologia e Execução

A realização do projeto foi dividida em quatro fases: Pesquisa, Desenvolvimento Inicial, Período de Testes/Ajustes e Manutenção.

1. Pesquisa:

Durante essa fase foi realizado um estudo para descobrir tecnologias e metodologias existentes que se adequassem aos requisitos do modelo CMMI-SW [CMMI]. Durante essa pesquisa, um sistema em especial foi destacado: o Bugzilla [BGZ]. Esse sistema já era usado pela empresa para controlar a execução de testes e registrar e gerenciar as falhas (bugs) encontradas durante essas execuções, tendo, portanto, uma equipe já familiarizada com o sistema. Assim, foi decidido que o Bugzilla sofreria modificações e seria ampliado para servir de suporte aos processos de Gerência de Requisitos e Gerência de Configuração. Outro sistema destacado foi o Subversion [SVN], gerenciador de versões de arquivo. A opção foi feita por integrar os dois sistemas. Essa integração deveria possibilitar visibilidade, ser facilmente controlável pelos gerentes e líderes da empresa e ser transparente para usuários comuns.

Durante essa fase foram identificadas algumas novas funcionalidades como coleta de métricas, suporte ao fluxo de PPQA e integração com o service desk da empresa. Um requisito em especial foi destacado nessa fase: o sistema resultante não poderia “engessar” o desenvolvimento de processo. Em resumo, o sistema resultante deveria ser bastante flexível e adaptar-se ao processo da empresa e não o contrário.

2. Desenvolvimento Inicial

Foi elaborado um plano para o desenvolvimento da ferramenta e alterações nos processos de Gerência de Requisitos e Gerência de Configuração de modo a

garantir que todas as alterações no processo da empresa estivessem refletidas na ferramenta. As atividades principais estavam relacionadas ao modelo do sistema, que previa funcionalidades básicas, interligação “inteligente” e regras processuais bem definidas.

Atualmente a equipe de projeto usa o MonsterZilla para controlar arquivos do projeto, requisitos, mudanças de requisitos e bugs. Algumas atividades relacionadas ao projeto, como planejamento, auditorias e testes também são controladas pela ferramenta.

3. Período de Testes/Ajustes

Durante essa fase a ferramenta foi disponibilizada para ser usada pela equipe de projetos da empresa. Uma versão inicial do processo foi disponibilizada e foi planejada a introdução em alguns projetos já existentes, para realização dos primeiros testes, e depois de ajustes. A ferramenta foi utilizada em todos os projetos, já iniciados ou novos. Erros, sugestões de melhoria e impressões gerais de uso eram relatadas para a equipe desenvolvedora, que após análise com EPG, resultavam em novas versões da ferramenta.

4. Manutenção

A ferramenta de apoio vem sendo amplamente usada pela Ícaro. Para que a manutenção e mudanças nela sejam controláveis foi criado um canal de comunicação entre o grupo de EPG e os usuários. Hoje, os usuários da ferramenta abrem tickets que relatam falhas e/ou melhorias. É possível acompanhar o andamento de cada pedido (ticket). Falhas são corrigidas pela equipe de suporte e sugestões de melhoria são discutidas com a equipe de EPG da empresa.

5. Esforço gasto

A fase de pesquisa teve a duração de 2 semanas e nela foram usados 1 analista e 1 desenvolvedor, sendo nenhum deles alocado 100% do tempo.

Para a fase de desenvolvimento, foi designada uma equipe com 1 analista e 1 desenvolvedor, sendo a gerência do projeto realizada pela equipe de EPG composta por 3 membros (um deles o próprio analista citado anteriormente). É importante ressaltar que a alocação dos recursos não chegou a ser 100% em nenhum momento. Essa fase teve a duração total de 3 meses (90 dias), sendo 75 dias gastos com desenvolvimento e 15 com integração.

A fase de testes/ajustes teve duração de 45 dias.

O trabalho de manutenção é feito pela equipe de suporte que tem um gasto mensal de 6 horas. A equipe de EPG gerencia mudanças na ferramenta. Essa equipe é composta por 4 membros e tem um gasto mensal de 6 horas.

Para o desenvolvimento completo da ferramenta foram gastos 149 homens-hora. A ferramenta ficou dentro do custo inicial estimado.

4. Resultados Obtidos

A ferramenta final é disponível via Web para a empresa. A visão geral pode ser visualizada na Figura 1.



Figura 1: visão geral do MonsterZilla

O conceito original de bugs do Bugzilla foi modificado e agora engloba entidades que suportam bugs, requisitos, mudanças de requisito e atividades do projeto. É possível cadastrar e configurar requisitos de diferentes granularidades (macro e micro) e associar mudanças aos requisitos originais. Isso permite rastreabilidade e controle do projeto. É possível associar itens de configuração e de repositório às entidades e associar entidades entre si. É possível também planejar as baselines que serão liberadas ao longo do projeto e associar os requisitos a essas baselines. Desta maneira, o MonsterZilla permite realizar a análise de impacto de mudanças ao projeto e controlar cada release (tanto internas quanto para o cliente). A tela principal de busca de entidades é mostrada na Figura 2.

This is MonsterZilla

Lucilla (Webka 2.0.3)

[Give me some help](#) (reloads page.)
[Find a Specific CR](#)
[Advanced Search](#)

Summary:

Req Type: Product: Component: Version:

A Comment: The URL:

Status: Resolution: Severity: Priority: Hardware: OS:

Email and Numbering: Any of: ☒ CR owner ☐ reporter

Any of: ☒ CR owner ☐ reporter

CR Changes: Only bugs changed between: and Now (YYYY-MM-DD or relative dates)

Localizer: Concluido mz

Figura 2: tela principal de busca

Pelo fato do MonsterZilla ser integrado com o Subversion, é possível realizar o controle de quem acessa o repositório e suas ações. É possível definir que somente itens de configuração pré-cadastrados podem ser inseridos no projeto e as inserções somente são permitidas quando os requisitos estiverem estáveis. O MonsterZilla permite também controlar as baselines do projeto, garantindo que apenas requisitos aceitos pelo cliente sejam colocados nas baselines. Uma tela de exemplo de controle de inserção de arquivos na árvore de projeto pode ser visualizada na Figura 3.

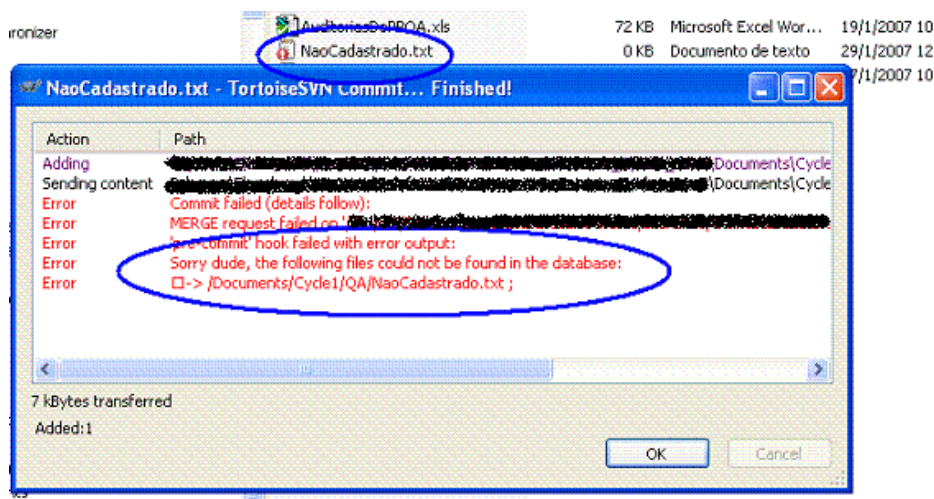


Figura 3: inserção controlada no repositório

É possível realizar coleta de várias métricas dos projetos. Por exemplo, é possível saber a quantidade de mudanças de requisitos e separar essas ocorrências por fases do projeto.

O MonsterZilla dá o suporte ao fluxo de PPQA (Process and Product Quality Assurance). Na ferramenta é possível controlar as auditorias do projeto e as não-conformidades associadas a estas auditorias. É possível, também, tirar as métricas relacionadas ao PPQA. Na figura pode ser visualizado o menu que permite acessar relatórios do projeto, área relacionada a testes e a área usada para o controle do fluxo de PPQA.

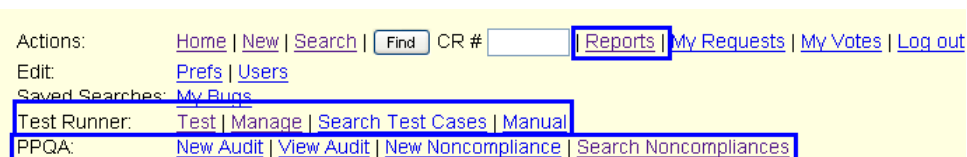


Figura 4: relatórios, área de testes e fluxo de PPQA

A ferramenta é integrada com o Service Desk da Icaro Technologies [ICT]. Isso permite que usuários consigam se comunicar facilmente com a equipe de EPG (gerencia mudanças) e a equipe de suporte (corrige erros da ferramenta). A integração foi feita via ticket e cada usuário consegue acompanhar o andamento dos seus tickets. É possível tirar relatórios dos tickets como por exemplo relação entre o número total de tickets e aqueles que foram fechados com sucesso. Na figura a seguir é mostrada a visão geral da tela de Service Desk da Icaro Technologies [ICT] com os pedidos de mudanças e bugs relacionados ao MonsterZilla e um relatório de status de fechamento dos tickets.

5 Aplicabilidade Dos Resultados

Inicialmente, a Icaro Technologies [ICT] tinha a proposta em disponibilizar a ferramenta. Porém, depois da fase de testes decidiu-se que o MonsterZilla será disponibilizado apenas para o uso interno da empresa, mas os conceitos e métodos desenvolvidos poderão ser utilizados por outras empresas parceiras, de grande, médio ou pequeno porte.

6 Características Inovadoras

O projeto é inovador na medida em que apresentou uma solução baseada em software livre que atende a uma demanda latente das organizações envolvidas com qualidade de software, que é a de ferramentas para integração completa das disciplinas de Gerência de Requisitos e de Configuração, com qualidade e custo acessível. As características inovadoras da ferramenta são:

- Modelagem de requisitos e relacionamentos para rastreabilidade fim-a-fim;
- Integração de softwares livres;
- Baixo custo;
- Grande flexibilidade e
- Baixo nível de acoplamento.

7 Conclusão e Perspectivas Futuras

No início do projeto foram levantados riscos, como a resistência inicial dos usuários devido ao maior controle e a falta da flexibilidade nas ferramentas usadas. Estes riscos não se concretizaram.

A ferramenta atendeu aos requisitos inicialmente estabelecidos. O período de testes e ajustes da ferramenta foi feito com sucesso e alguns projetos que existiam antes do MonsterZilla ter ficado “pronto” adequaram-se para poder usar a ferramenta. Todos os novos projetos da Ícaro utilizam o fluxo de Gerência de Configuração, Gerência de Requisitos e PPQA com o auxílio da ferramenta. O MonsterZilla já se encontra em uso geral pela empresa e a integração com o service desk da Icaro Technologies [ICT] está em funcionamento.

A Icaro Technologies [ICT] pretende contribuir com a sociedade através da manutenção e aprimoramento constantes do seu processo e da divulgação dele entre as empresas parceiras. O código-fonte da ferramenta estará disponível no site da Icaro Technologies [ICT] em formato “AS IS”. Ou seja, nenhuma garantia de funcionamento ou suporte é provida pela Icaro Technologies [ICT].

8 Referências Bibliográficas

[ICT] – Icaro Technologies; <http://www.icaro.com.br>

[SVN] – Subversion; <http://subversion.tigris.org/>; site acessado em 06/02/2007

[BGZ] – Bugzilla; <http://www.bugzilla.org/>; site acessado em 06/02/2007

[CMMI] – Livro “Capability Maturity Model® Integration (CMMISM), Version 1.1” (formato pdf), Carnegie Mellon, Software Engineering Institute, Pittsburgh, PA 15213-3890, Agosto de 2002

[6.30] Projeto Qualidade do Software Embarcado em Aplicações Espaciais

Entidade: ¹Instituto Nacional de Pesquisas Espaciais (INPE), Caixa Postal 515 - 12.227-010 - São José dos Campos - SP e ²DBA Engenharia de Sistemas Ltda, Av. Presidente Vargas 3131, 3º Andar, CEP 20210-030 - RJ

Autores: Coordenadora: Maria de Fátima Mattiello-Francisco ¹, Co-executor: Ricardo Costa ² - {fatima@dss.inpe.br rcosta@dba.com.br}

1. Introdução

A terceirização de TI é um Mercado crescente no mundo. Um estudo visando elaborar uma agenda estratégica para incrementar as exportações do setor de serviços de TI no Brasil mostrou que o mercado de *outsourcing* é bastante desenvolvido na América do Norte e na Europa onde se concentram demandas por serviços de maior valor agregado. Juntas, a América do Norte e a Europa representam quase 78% da demanda por *outsourcing* de serviços de TI. Em 2004, o mercado global de *outsourcing* de serviços de TI totalizou US\$ 607 bilhões e a

perspectiva de crescimento é de 6% ao ano até 2008. [Governo Federal, BRASSCOM 2004].

Esta tendência é claramente observada no setor espacial pela particular atenção dada na última década ao software em normas do tipo *European Cooperation for Space Standardization* – ECSS. Fruto de um esforço conjunto entre as Agências Espaciais Européias – ESA e a associação das indústrias Européias, as normas ECSS foram criadas nos anos 90 com o propósito de facilitar a relação cliente/fornecedor no desenvolvimento de projetos na área espacial. A preocupação das agências espaciais mundiais com a qualidade dos projetos de satélites, veículos lançadores, operação de missões em solo, entre outros, se estende para qualidade do software de forma crescente. A evolução dos sistemas computacionais tem permitido a implementação de funcionalidades por software tornando o item de software mais complexo e, portanto sujeito a erros. Soma-se a inserção do setor industrial na área espacial demandando grandes investimentos na melhoria dos processos adotados para o desenvolvimento de softwares críticos, o que especialmente envolve a aplicação de metodologias e técnicas da qualidade na gestão de projeto de software.

Com o propósito de envolver a indústria Brasileira no processo de desenvolvimento de missões de satélites como fornecedores de subsistemas e serviços para a área espacial, o Instituto Nacional de Pesquisas Espaciais – INPE tem adotado as normas ECSS para o desenvolvimento de vários subsistemas. Porém esforços orientados à aplicação das normas específicas de software (ECSS-E-40 e ECSS-Q-80) para o desenvolvimento de software embarcado pela indústria nacional nunca havia sido feitos de forma sistemática de modo a avaliar a aderência dos processos seguidos pela indústria de software aos requisitos recomendados pelas normas ECSS.

Neste contexto, o projeto Qualidade do Software Embarcado em Aplicações Espaciais – QSEE, financiado pela FINEP, foi concebido visando dois objetivos complementares: experimentar o modelo de Fábrica de Software no fornecimento de software embarcado em aplicações espaciais para o INPE através do uso das normas ECSS; e desenvolver um processo de aceitação de software para o INPE apoiado na abordagem Verificação e Validação Independente de Software preconizado pela ECSS. Caracterizado como um projeto de transferência tecnológica do INPE para a indústria brasileira do setor de software DBA Engenharia de Sistemas LTDA, o projeto QSEE iniciou em Janeiro de 2005.

2. Objetivos e Justificativa

No período correspondente ao ciclo 2006, as atividades do projeto QSEE estavam orientadas a conduzir a produção de um software embarcado piloto na Fábrica de Software da DBA aplicando os requisitos das normas ECSS da qualidade [ECSS-Q-80] e engenharia de software [ECSS-E-40] cuja adequação (*tailored form*) havia sido feita no primeiro ano do projeto.

O maior desafio do projeto neste período era analisar em tempo de execução (durante o desenvolvimento do software) a aderência dos processos de desenvolvimento de software já estabelecidos na Fábrica de Software da DBA, uma empresa formalmente avaliada CMMI nível 3, mas não familiarizada com a

tecnologia de desenvolvimento de software crítico embarcado em aplicações espaciais, aos requisitos impostos pelo cliente INPE.

A escolha de uma empresa do setor de software não envolvida no domínio da aplicação espacial deve-se ao desafio de o INPE experimentar *outsourcing* para um fornecedor que segue o modelo Fábrica de Software. Dois fatores sustentaram esta decisão: o fato de uma empresa CMMI 3 garantir que seus processos são aderentes às áreas-chaves de gestão de requisitos, planejamento de projeto, supervisão e acompanhamento de projeto e garantia da qualidade de software, bem como gerência de configuração de software e engenharia de produto de software. Soma-se o fato da gestão de projetos instituída na empresa ser orientada “por projeto”. Bibliotecas de planos de projetos entregues, por domínio da aplicação, são mantidas na empresa, estabelecendo uma base de conhecimento da organização nas tecnologias dominadas. Assim, a organização é capaz de sistematizar de forma menos burocrática seu envolvimento em projetos que requerem novas tecnologias, capturando e memorizando as peculiaridades do novo domínio de aplicação. Desta forma, os riscos de os investimentos do governo e esforços do INPE em um trabalho de transferência de tecnologia para a empresa brasileira serem perdidos com o tempo, dada a baixa demanda por projetos desta natureza, foram minimizados.

3. Metodologia de Execução

A metodologia de execução do projeto QSEE compreendeu cronologicamente as três fases abaixo relacionadas:

- Fase 1: constituição, treinamento das equipes, estudo das normas ECSS e processo de adequação (*tailored form*)
- Fase 2: especificação do projeto piloto (software SWPDC), concepção do processo de aceitação do software pelo INPE aplicando Verificação e Validação Independente.
- Fase 3: desenvolvimento do software SWPDC e validação das duas versões do projeto piloto usando a metodologia de aceitação.

Durante o período do ciclo 2006, foram cumpridas as fases dois e três integralmente. Elas serão relatadas a seguir, destacando-se as atividades realizadas e os produtos gerados.

3.1 Fase 2 : especificação do projeto piloto e concepção do método de aceitação

Esta fase compreendeu a definição de escopo do software piloto SWPCD, uma simplificação do software embarcado em instrumento a bordo de satélite científico. No caso, o instrumento caracteriza-se como carga útil do satélite MIRAX – uma missão científica de astronomia cujo objetivo é o imageamento do centro da galáxia em raios X. O software SWPDC embarcado neste instrumento, carga útil do MIRAX, foi especificado como o software piloto do projeto QSEE e desenvolvido em duas versões pelas equipes INPE e Fábrica de software da DBA. O software SWPDC constituiu um “case” para a realização da transferência

tecnológica da qualidade em software embarcado para a indústria nacional, na figura da interveniente, seguindo as recomendações das normas ECSS. Basicamente, nesta fase foram executadas as seguintes atividades:

1. Elaboração do documento de requisitos do software SWPDC
2. Concepção da metodologia de aceitação e ajustes à abordagem de Verificação e validação Independente adotada pelo projeto QSEE
3. Definição do cronograma de desenvolvimento do software piloto incluindo as revisões do projeto
4. Definição do processo de verificação
5. Revisão dos Requisitos Base do software SWPDC – **SSR** – Software Specification Review
6. Especificação Técnica do SWPDC pela DBA;
7. Especificação Técnica do SWPDC pelo INPE;
8. Revisão **PDR** – Preliminar Design Review

3.2 Fase 3 : desenvolvimento do software piloto SWPDC

Esta fase compreendeu o projeto, implementação e testes do software piloto SWPCD em duas versões. Um desenvolvida no INPE por especialistas da área espacial em engenharia de software embarcado, por uma equipe liderada por um engenheiro pleno do Laboratório de Altas Energias do INPE. A outra versão desenvolvida no Fábrica de Software da DBA Engenharia de Sistemas LTDA seguindo os processos do ciclo de vida de desenvolvimento de software da empresa avaliada CCMI 3. Basicamente, nesta fase foram executadas as seguintes atividades:

9. Projeto do software SWPDC, versão DBA, na Fábrica de Software, sob a liderança de Benhur Barbosa Lima;
10. Projeto do software SWPDC, versão INPE, no laboratório da CEA sob a liderança de Valdivino Santiago Jr.;
11. Revisão **DDR** - Detailed Design Review
12. Desenvolvimento e Testes do SWPDC, versão DBA, na Fábrica de Software, sob a liderança de Benhur Barbosa Lima;
13. Desenvolvimento e Testes do SWPDC, versão INPE, no laboratório da CEA sob a liderança de Valdivino Santiago Jr.;
14. Revisão **CDR** - Critical Design Review realizada em outubro de 2006
15. Aceitação do software no INPE seguindo o planejamento de Validação Independente.

Os seguintes documentos foram gerados e revisados ao longo das duas fases descritas acima:

1. RB - Requisitos base (preparado pelo cliente INPE)

2. Protocolo de comunicação PDC-OBDR (preparado pelo cliente INPE)
3. Protocolo de comunicação PDC-EPP (preparado pelo cliente INPE)
4. PDS - Plano de Desenvolvimento de Software (duas versões)
5. PVVIS – Plano de Verificação e Validação Independente (preparado pela equipe de IVV)
6. Relato da revisão SRR,
7. ETS - Especificação Técnica do Software (duas versões)
8. DPS – Documento de Projeto do Software (duas versões)
9. PTS – Plano de Teste do Software (duas versões)
10. Relatos das revisões PDR e DDR;
11. DET – Documento de especificação dos Testes (duas versões)
12. DRT – Documento de Relato de Testes (duas versões)
13. MUS – Manual do Usuário do Software (duas versões)
14. ETIVV – Especificação dos Testes de IVV (preparado pela equipe de IVV)
15. Relato da revisão CDR.

Os seguintes produtos foram entregues:

1. **Código Fonte:** Duas versões do Software Piloto **SWPDC** entregue pelos fornecedores INPE e DBA
2. **Código Fonte:** Duas versões dos simuladores **EPPs** entregue pelos fornecedores INPE e DBA
3. **Ferramenta de apoio à execução dos testes QSEE-TAS** desenvolvida no INPE e disponibilizada para os fornecedores INPE e DBA para os testes e aceitação do software SWPDC.

Artigos publicados, disponibilizados no site do projeto:
<http://www.das.inpe.br/~qsee/>

Verificação e Validação na terceirização de software embarcado em aplicações espaciais - Mattiello-Francisco, M.F. (INPE) ; Santiago, V. A. (INPE). ; Ambrosio, A.M. ; Costa, R. (DBA), Jogaib, L. (DBA). SBQS2006 – V Simpósio Brasileiro de Qualidade de Software – Apresentação Oral, Vila Velha - ES, Brasil – 29 Maio a 03 Junho 2006 – pp. 367- 374.

Safety in a Web-based Satellite Flight Plan Supporting System - Mattiello-Francisco, M.F. ; Sakugawa, B.M. ; Yano, E.T. SpaceOPS2006 – Oral Presentation – Session 41-OS-19 – Planning Tools II - AIAA International Conference, Rome, Italy – 19-23 June 2006

An Evolutionary Operational Profiles Approach for Integration Tests - Mattiello-Francisco, M.F. – Supplement Proceedings of the 2006



International Conference on Dependable Systems and Networks – DSN
2006 – 25- 28 June 2005 – Philadelphia, PA, USA, pp 210.

A Practical Approach for Automated Test Case Generation using Statecharts

- Santiago, V. A. ; Amaral, A.S.M.; Vijaykumar, N.L.; Mattiello-Francisco, M.F.; Martins E.; Lopes, O.C. . COMPSAC2006 – 30th Annual International Computer Software and Applications Conference – Apresentação Oral, Chicago, Illinois, USA – 17-21 Setembro 2006 – Vol. II, pp. 183- 188.

QSEE-TAS: Uma Ferramenta para Execução e Relato Automatizados de Testes de Software para Aplicações Espaciais – Silva, W.; Santiago, V. A.; Mattiello-Francisco, M.F. ; Passos, D. - Em: XX Simpósio Brasileiro de Engenharia de Software – SBES2006 - Sessão de Ferramentas, 16 – 20 de Outubro/2006, Florianópolis, Santa Catarina

Método desenvolvido:

Método de Aceitação de software pelo cliente usando a abordagem de Verificação e Validação Independente encomendada a uma terceira parte. Aplicação de técnicas Model-based Testing para a especificação de Testes de Conformidade e Injeção de Falhas, usando a metodologia CoFI combinada à geração automática de casos de testes usando a ferramenta CONDADO desenvolvida no IC/Unicamp.

Recursos Humanos capacitados:

- 2 estagiários com mestrado concluído, absorvidos pela Fábrica de Software DBA
- 2 estagiários com graduação concluída em Engenharia da Computação e Ciência da Computação;
- 1 técnico em computação.
- 1 Líder da Fábrica de Software

4. Aplicabilidade dos Resultados

Os resultados obtidos no projeto QSEE durante o período correspondente ao Ciclo 2006 foram extremamente positivos, pois o objeto de desenvolvimento, o software SWPDC, foi efetivamente produzido no modelo Fábrica de Software através da transferência de tecnologia proposta e realização de todo o processo de desenvolvimento planejado. Assim a meta fundamental para a avaliação da aderência dos processos da Fábrica de Software da DBA aos requisitos das normas ECSS foi atingido. Conseguiu-se que o processo fosse totalmente realizado, com extensão de 20% do prazo inicialmente estimado para o desenvolvimento do software.

A maturidade da empresa DBA permitiu identificar a sistemática de Revisão Formal recomendada pela ECSS e adotada no projeto como um mecanismo efetivo de interação com o cliente para a fase de análise dos requisitos. Observou-se que este procedimento evitou interpretações equivocadas do fornecedor em relação à especificação entregue pelo cliente, mitigando os riscos de eventuais problemas na entrega do produto do software e consequente

retrabalho. Constatou-se a incorporação da Revisão de Requisitos de Software - SSR no processo de desenvolvimento da Fábrica de Software da DBA

Adicionalmente, o projeto contribuiu para melhorias no processo de testes existente na Fábrica de Software. Um checklist para software embarcado foi criado para guiar a revisão por pares praticada pela empresa e, maior atenção foi dada ao planejamento dos testes de integração de componentes e testes de sistema, complementando a revisão por pares.

5. Características Inovadoras

As seguintes características do projeto QSEE podem ser citadas como inovadoras:

- Adaptação das normas ECSS para um projeto de software embarcado em aplicações espaciais terceirizado a uma Fábrica de Software Brasileira;
- Desenvolvimento e aplicação de um processo de aceitação de software em um caso real de terceirização de software embarcado;
- Aplicação da abordagem Verificação e Validação Independente de Software recomendada pela ECSS para o desenvolvimento de aplicações críticas;
- Transferência de tecnologia da área espacial para uma empresa brasileira do setor de software.

6. Conclusão e Perspectivas Futuras

O nível de maturidade 3 CMMI da Fábrica de Software da DBA permitiu à equipe da Garantia da Qualidade do Software identificar, de forma relativamente simples, as necessidades requeridas pelo cliente INPE para o produto de software embarcado. Em função do engajamento deste produto no ciclo de vida da missão espacial, as exigências de revisões formais ao longo do desenvolvimento do software foram compreendidas pelo fornecedor. Os mecanismos existentes nos processos da qualidade da empresa facilitaram a implantação das adequações que se mostraram necessárias como, por exemplo, a elaboração do conjunto de documentos exigidos pelo cliente nas revisões formais de projeto, seguindo os padrões ECSS.

Adicionalmente, o treinamento em desenvolvimento de software embarcado dado no INPE, por seis meses, para a equipe do projeto QSEE (2 analistas de sistemas pleno com mestrado) posteriormente absorvida pela Fábrica de Software da DBA, facilitou a instanciação dos processos da Fábrica de Software para a produção de software embarcado em aplicações espaciais. A experiência da DBA no modelo de Fábrica de Software para produtos de software embarcado em celulares ajudou muito na correlação dos conceitos “modelo de engenharia”, adotado em projetos espaciais, com “ambiente alvo”, domínio da DBA. As especificidades do domínio espacial contribuíram com a implantação de melhorias nos sub-processos do processo de Teste da Fábrica.

A continuidade do projeto após a revisão CDR caracterizou-se como **Fase de Aceitação** compreendendo a validação independente das duas versões do

software piloto e a consolidação da metodologia de aceitação desenvolvida no projeto. O processo de aceitação desenvolvido e experimentado neste projeto será adotado pelo INPE na terceirização de futuros desenvolvimentos de software para aplicações espaciais.

Agradecimentos

Os autores agradecem à Financiadora de Estudos e Projetos (FINEP) pelo suporte dado ao projeto Qualidade do Software Embarcado em Aplicações Espaciais (QSEE).