

Ministério da Ciência, Tecnologia e Inovação  
Secretaria de Política de Informática



# Programa Brasileiro da Qualidade e Produtividade em Software

Artigos - Ciclo 2011



3ª Edição  
Maio de 2012

Esplanada dos Ministérios – Bloco E- 3º andar  
[www.mct.gov.br/sepin](http://www.mct.gov.br/sepin)

Secretaria de Política  
de Informática

Ministério da  
Ciência e Tecnologia





# **Programa Brasileiro da Qualidade e Produtividade em Software Projetos Ciclo 2011**

**Presidente da República**

Dilma Vana Rousseff

**Ministro da Ciência e Tecnologia**

Aloizio Mercadante Oliva

**Secretário de Política de Informática**

Virgílio Augusto Fernandes Almeida

**Diretora de Políticas e Programas Setoriais em Tecnologias da Informação**

Marylin Peixoto da Silva Nogueira

**Coordenador Geral de Serviços e Programas de Computador**

Rafael Henrique Rodrigues Moreira

**PBQP Software**

**Coordenador Geral**

Kival Chaves Weber

**Coordenador de Projetos e Indicadores**

Diva da Silva Marinho

**Compilação, organização e revisão**

Diva da Silva Marinho

Antonio Fonseca Neto

**Equipe Técnica**

Eurípedes José de Sousa



**Ministério da Ciência e Tecnologia**  
**Secretaria de Política de Informática**

# **Revista do Programa Brasileiro da Qualidade e Produtividade em Software**

3ª edição  
Maio/2012

## Editorial

Essa é mais uma publicação que a Secretaria de Política de Informática - SEPIN elabora, com foco na qualidade do setor de software brasileiro. Trata-se da segunda edição da Revista - Programa Brasileiro da Qualidade e Produtividade em Software, que contém artigos que descrevem os projetos concluídos que foram submetidos ao Ciclo de Projetos 2011 do Programa.

Os projetos concluídos foram avaliados por um Comitê de Avaliadores do Programa com objetivo de conceder o Prêmio Dorgival Brandão Junior da Qualidade e Produtividade em Software aos três melhores avaliados.

Os vinte e seis artigos reunidos nessa segunda edição da revista representam uma parte significativa do trabalho que a comunidade brasileira envolvida com o tema de qualidade de software desenvolve com vistas a alcançar a melhoria dos processos e produtos do software brasileiro. Congratulo os autores e as instituições que contribuíram para mais uma edição da revista.

Rafael Henrique Rodrigues Moreira  
Especialista em Políticas Públicas e Gestão Governamental  
Coordenador Geral de Software e Serviços  
MCT/SEPIN

# Sumário

## Artigos dos Projetos 2011

Programa de análise de qualidade em TI - PraTIQ .....	10
Cintia Eitelwein <sup>1</sup> , Felipe Grando Sória <sup>2</sup> , Celia Marli Tokarski Dieguez <sup>3</sup> .....	10
Segurança de Sistemas da Informação .....	12
Amândio Balcão Filho, Ana Cervigni Guerra, André Abed Grégio, Ferrucio de Franco Rosa, Edson Corrêa Teracine, Luiz Otávio Duarte, Regina M. Thienne Colombo.....	12
SPIN São Paulo 15 anos de contribuição à melhoria da qualidade de software. ....	18
Renato Luiz Della Volpe <sup>1</sup> , Ernesto de Carvalho Bedrikow <sup>2</sup> , Antônio Carlos Tonini <sup>3</sup> , Valdir Cardoso da Silva <sup>4</sup> .....	18
Consolidação do Programa MPS.BR – Melhoria de Processo do Software Brasileiro (2008-2011) 25 Kival Weber <sup>1</sup> , José Antonio Antonioni <sup>1</sup> , Nelson Franco <sup>1</sup> , Elidiane Barroso <sup>1</sup> , Cleide Silva <sup>1</sup> , André Sotovia <sup>1</sup> .....	25
Proposta de Modelo de Qualidade de Requisitos de Segurança para Aplicativos de Software na Web .....	33
Regina M Thienne Colombo, Amândio Balcão Filho, Ana Cervigni Guerra, Marcelo Schneck de Paula Pessoa.....	33
Metodologia para Execução de uma Avaliação Integrada .....	40
André Villas Boas, Raquel Moreda P. Gutierrez, Silvia R. Lopes, Valmir Martins, Valquíria M. Fagundes ....	40
Ontologias e Mapas Conceituais na Transferência de Conhecimento sobre Processo de Software .....	43
José Augusto Fabri <sup>1</sup> , André Luís dos Santos Domingues <sup>1</sup> , Alexandre L’Erário <sup>1</sup> , Alessandro Silveira Duarte <sup>1</sup> , Elias Canhadas Genvigir Rodrigo Henrique Cunha Palácios <sup>1</sup> .....	43
Ambiente Integrado – Uma Abordagem Automatizada e Colaborativa para Gestão de Processos do MPS.Br .....	49
Simone Vasconcelos Silva, Aline P. V. de Vasconcelos, Jonivan Lisboa Coutinho .....	49
Consolidação de um Método para Customização de Modelos de Capacidade/Maturidade de Processos de Software .....	57
Jean Carlo Rossa Hauck .....	57
Chrisriane Gresse von Wangenheim .....	57
AgileCCPM – Uma integração entre Scrum e Corrente Crítica.....	67
Daniel Tadeu Martínez C. Branco <sup>1,2</sup> , Rafael Prikkladnicki <sup>3</sup> , Tayana Conte <sup>1</sup> .....	67
Institucionalizar Segurança no RUP-BNB.....	72
Francisco José Barreto Nunes, Joaquim Manoel Pitombeira de Oliveira .....	72
Um <i>framework</i> para implantação de gerenciamento de configuração baseado no MPS.BR e RUP 79 Marcelo Hideki Yamaguti, Helder Lino Moura Cardoso, Eder John Delgado de Pina.....	79
Ferramenta de Desenvolvimento Quellon Studio .....	83
Carlos José Pereira .....	83

O processo de Inovação Centrada no Ser Humano: boas práticas de desenvolvimento de software utilizadas no projeto Portal do Aluno.....	87
Gabriel C. Cardoso <sup>1</sup> , Ivan L. Martins <sup>2</sup> .....	87
BRISA QPM – Qualimobile Process Management.....	95
Kátia Argolo de Castro, André Luís Gobbi Sanches, Johnny Souza, Thamer de Jesus Hatem .....	95
Spider-Reuse: Ferramenta Aplicada à Gerência de Reutilização de Ativos de Software aderente às recomendações dos Modelos/Normas de Qualidade .....	100
Fabrcio Medeiros Alho <sup>2</sup> , Kleverton Oliveira Macedo <sup>1</sup> , Sandro Ronaldo Bezerra Oliveira <sup>1,2</sup> .....	100
Apoio Sistematizado de Software Livre para Implementação do Processo de Desenvolvimento de Requisitos do MPS.BR .....	106
Ewelton Yoshio Chiba Yoshidome <sup>1</sup> , Sandro Ronaldo Bezerra Oliveira <sup>1</sup> .....	106
Uma Proposta de Processo para Aquisição de Software e Serviços Correlatos Baseada no CMMI-ACQ e no Guia de Implementação do MPS.BR .....	109
Julio Cezar Costa Furtado <sup>1</sup> , Sandro Ronaldo Bezerra Oliveira <sup>1</sup> .....	109
Avaliação de Usabilidade de Aplicações em Dispositivos Móveis .....	115
Bruno Bonifácio, Mario J. S. C. Filho, Tayana Conte .....	115
Apoio Sistêmico na Avaliação de Produtos de Software no Contexto do MEDE-PROS .....	122
Gleyson do Nascimento Gama <sup>1</sup> , Sandro Ronaldo Bezerra Oliveira <sup>1,2</sup> .....	122
<i>RBTtool</i> - Ferramenta de Teste Baseado em Riscos.....	126
Cristine Gusmão <sup>3</sup> , Ellen Souza <sup>1,2</sup> , Júlio Menezes Jr. <sup>1</sup> , Liliane Silva <sup>1</sup> , Nielson Pontes <sup>1</sup> .....	126
WDP-RT-Q: Uma Técnica Automatizada de Perguntas e Respostas para Inspeção de Aplicações Web .....	133
Priscila Fernandes, Tayana Conte .....	133
Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software.....	140

# **Artigos dos Projetos Concluídos Ciclo 2011**

# Programa de análise de qualidade em TI - PraTIQ

Cintia Eitelwein<sup>1</sup>, Felipe Grando Sória<sup>2</sup>, Celia Marli Tokarski Dieguez<sup>3</sup>

Centro Internacional de Tecnologia de Software - CITS

cintia.eitelwein@cits.br, felipe.soria@cits.br, celia.dieguez@cits.br

**Resumo.** Este artigo descreve o conceito e a execução do ciclo 2011 do Programa de Análise de Qualidade em TI - PraTIQ que dissemina o conhecimento de engenharia de software, metodologias, métodos, modelos além de práticas do mercado sem ter um vínculo com um modelo de referência específico.

## 1. Introdução

O PraTIQ(Programa de Análise de Qualidade em TI), foi idealizado pelo CITS(Centro Internacional de Tecnologia de Software) e SEBRAE-PR, para atender empresas integrantes dos APLs(Arranjos Produtivos Locais) no Estado do Paraná. O programa é formado por atividades de capacitação nas áreas de Gestão de Projetos, Gerência de Requisitos e Gestão de Processos, visto que são disciplinas básicas da engenharia de software e disseminam conhecimento agregando valor ao negócio, promovendo a busca por qualidade de produtos e serviços da região. Durante o ciclo de 2011 o programa foi realizado em grupos de empresa, sendo um dos grupos do estado de Santa Catarina.

## 2. Objetivos e Justificativa

O PraTIQ tem como objetivo sensibilizar não só os empresários do mercado de TI do Paraná, mas também seus colaboradores quanto à importância do estabelecimento de um Programa de Melhoria de Processos de Qualidade em Desenvolvimento de Software para a ascensão do negócio. Uma vez que o mercado está cada vez mais exigente no que diz respeito à qualidade e complexidade dos produtos, o intuito do PraTIQ é incentivar a competência e a busca incessante pela melhoria de processos e, em consequência, melhores produtos e serviços, itens fundamentais para a sobrevivência no mercado de TI. Este programa visa micro, pequenas empresas e empresas iniciantes, despertando de forma organizada a introdução de processo de melhoria de qualidade, com custo acessível.

Para o ciclo 2011, o PraTIQ teve como objetivos:

- Desenvolver mais turmas, de forma a consolidar a metodologia no estado do Paraná e em outras localidades;
- Viabilizar a formação de novos grupos, com empresas mais qualificadas, aumentando assim a taxa de sucesso nos resultados de certificações em modelos reconhecidos pelo mercado;
- Dar acesso à micro, pequenas e médias empresas ao estado da arte da engenharia de software, com baixo custo;
- Contribuir para o aumento do número de empresas avaliadas no Paraná, a fim de tornar o estado como referência nacional e internacional neste setor.

## 3. Metodologia e Execução

No ciclo de 2011 o PraTIQ foi executado em dois grupos formados por micro, pequenas e médias empresas, sendo um em Curitiba e outro em Chapecó. O grupo de Curitiba foi composto por sete empresas e o grupo de Chapecó por seis empresas.

Por meio de uma equipe de profissionais credenciados ao CITS, foram realizados cursos de capacitação com grupos de empresas para disseminar os conceitos básicos da engenharia de software, métodos, metodologias e modelos. Após a capacitação foram realizadas, pela mesma equipe do CITS, consultorias individuais nas empresas para identificar a necessidade específica de cada uma e proporcionar um direcionamento em relação à execução de um Programa de Melhoria de Processos de Qualidade em Desenvolvimento de Software.

## 4. Resultados Obtidos

No ciclo 2011 foram capacitadas 27 pessoas distribuídas em 13 empresas em dois estados. Esta capacitação somada à consultoria executada pelo CITS nas empresas proporcionou a disseminação de conceitos da engenharia de software, métodos, metodologias e modelos.

Dentro dos objetivos propostos foram alcançados os seguintes resultados:

- Desenvolver mais turmas, de forma a consolidar a metodologia no estado do Paraná e em outras localidades.

O programa foi realizado com duas turmas de empresas, não aumentando o número de turmas já que no ciclo anterior três turmas realizaram o programa. No ciclo 2011 o programa teve uma das turmas em outro estado, Santa Catarina, na cidade Chapecó, atendendo a expectativa de executar o programa em outras localizações.

- Viabilizar a formação de novos grupos, com empresas mais qualificadas, aumentando assim a taxa de sucesso nos resultados de certificações em modelos reconhecidos pelo mercado.

Treze empresas participaram do programa no ciclo 2011, estas identificaram a importância e a necessidade de um projeto de melhoria de processo alinhado com a estratégia da empresa. Destas empresas sete estão participando do grupo de implementação de um Programa de Melhoria de Processos de Qualidade em Desenvolvimento de Software do modelo MPS.BR com apoio da SOFTEX.

- Dar acesso à micro, pequenas e médias empresas ao estado da arte da engenharia de software, com baixo custo.

Como o PraTIQ é executado em grupo de empresas é possível praticar um custo reduzido. O PraTIQ permite também que o empresário participe de programa de melhoria de processo ciente e confortável para implementá-lo, diminuindo o custo e o risco do projeto de implementação

- Contribuir para o aumento do número de empresas avaliadas no Paraná, a fim de tornar o estado como referência nacional e internacional neste setor.

O PraTIQ permite as empresas executarem o planejamento mais eficiente sobre a implementação de programas de melhorias de processo. Algumas empresas que participaram do programa estão participando de um Programa de Melhoria de Processos de Qualidade em Desenvolvimento de Software do modelo MPS.BR com apoio da SOFTEX e outras sinalizaram a inclusão desta necessidade em sua estratégia organizacional.

## 5. Aplicabilidade dos Resultados

O projeto de implementação melhoria de processo, com o intuito de obter certificação em modelos renomados, como MPS.BR e CMMI proporciona a empresas de software um posicionamento em relação ao mercado nacional e internacional dando condições de competitividade para o mercado global. O PraTIQ aproxima micro, pequenas e médias empresas aos conceitos destes modelos e uma primeira perspectiva do impacto da implementação de um modelo na empresa.

Do ciclo 2011 seis empresas do grupo de Curitiba estão participando do projeto de implementação de melhoria de processo e devem obter o do Nível G de maturidade do modelo MPS.BR em 2013. No grupo de Chapecó composto por seis empresas, uma está participando do projeto de implementação de melhoria de processo e deverá obter o Nível G de maturidade também em 2013.

Na implementação do modelo MPS.BR ficou evidenciado pelos implementadores a facilidade de comunicação e desenvolvimento das atividades devido qualificação e familiaridade com o modelo das empresas que participaram do PraTIQ

## 6. Características Inovadoras

O PraTIQ é uma metodologia inovadora, pois visa um treinamento em melhoria de processos de software, com uma visão mais generalista dos modelos de melhoria existentes, facilitando tanto na base teórica necessária para conseguir a certificação, quanto na parte econômica, a tendência é diminuir os custos com grupos de empresa. Os modelos de referência atuais partem do princípio de que as empresas têm alto poder aquisitivo, restringindo a possibilidade melhoria de processo em micro, pequenas e médias empresas.

## 7. Conclusão e Perspectivas Futuras

O Programa de Análise de Qualidade em TI é executado pelo CITS e SEBRAE, sensibilizando os empresários do mercado de TI do Paraná e Santa Catarina, quanto à importância da implementação de um Programa de Melhoria de Processos de Qualidade em Desenvolvimento de Software. Com a evolução do programa, espera-se a formação de novas turmas e o apoio de outras entidades para apoio e divulgação do programa e atuação em todo território brasileiro e Latino Americano.

## 8. Referências Bibliográficas

[SEI, 20102010] SOFTWARE ENGINEERING INSTITUTE. CMMI for Development (CMMI-DEV), Version 1.3, Technical Report CMU/SEI-2010-TR-033. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2010.

[SOFTEX, 2011a] - ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR – Guia Geral:2011, junho 2011. Disponível em [www.softex.br](http://www.softex.br)

# Segurança de Sistemas da Informação

## (Melhorando a Segurança em Software)

Amândio Balcão Filho, Ana Cervigni Guerra, André Abed Grégio, Ferrucio de Franco Rosa, Edson Corrêa Teracine, Luiz Otávio Duarte, Regina M. Thienne Colombo

Centro de Tecnologia da Informação – CTI Renato Archer  
DSSI – Divisão de Segurança de Sistemas de Informação  
Rod. Dom Pedro I, km 143,6 – 13.069-901 – Campinas – SP – Brazil

[amandio.balcao@cti.gov.br](mailto:amandio.balcao@cti.gov.br) , [ana.guerra@cti.gov.br](mailto:ana.guerra@cti.gov.br) , [ferrucio.rosa@cti.gov.br](mailto:ferrucio.rosa@cti.gov.br),  
[edson.teracine@cti.gov.br](mailto:edson.teracine@cti.gov.br), [lod@gmail.com](mailto:lod@gmail.com) , [regina.thienne@cti.gov.br](mailto:regina.thienne@cti.gov.br)

### 1. Introdução

As aplicações de software se tornaram o alvo preferencial de ataques de criminosos, por isso é preciso garantir que as aplicações sejam projetadas, desenvolvidas e implantadas sem falhas de segurança. Pesquisar e desenvolver novas tecnologias nestas áreas é de suma importância para o país. A Divisão de Segurança de Sistemas de Informação (DSSI) do CTI (Centro de Tecnologia da Informação Renato Archer) desempenha um papel para melhorar a segurança do produto de software no país. Para isto, entre suas principais atribuições estão desenvolver metodologias, procedimentos e ferramentas para melhorar a segurança de sistemas interligados às redes de comunicação, apoiando o aprimoramento da segurança de organizações públicas e privadas e buscando formar pessoal especializado na área, além de disseminar uma cultura de segurança da informação.

No início da utilização de sistemas de software, os sistemas computacionais serviam apenas para cálculos, sem grandes recursos visuais, processamentos e gerenciamento de informações. Entretanto, atualmente os sistemas são complexos envolvendo a interatividade com os usuários onde uma interface amigável pode minimizar dificuldades com usuários leigos no domínio, na aplicação e também nos sistemas com a utilização da rede para a comunicação entre os usuários e servidores dos atuais sistemas de informação. Com relação a estes tipos de sistemas, os usuários especialistas têm a preocupação com a confiança na segurança da informação devido à grande complexidade. Um dos problemas que preocupam muito os CEO's e CIO's no mundo inteiro é também um dos mais complexos de se resolver: como impedir que informações valiosas da organização caiam em mão erradas, ou simplesmente se perdessem. A gestão da segurança da informação é um assunto sério, é estratégico e deve estar entre as preocupações centrais da alta direção da instituição. Regulamentos, normas de conduta e diversos mecanismos de defesa têm sido desenvolvidos nos últimos anos, num cenário em que, cada vez mais, as empresas dependem de seus sistemas de informação e do comportamento da sua força de trabalho, para assegurar que seus dados estejam protegidos.

### 2. Objetivos e Justificativa

Os macros objetivos deste projeto são:

- Conscientizar instituições e organizações sobre a necessidade de melhorar a segurança de software;
- Desenvolver metodologias, procedimentos e ferramentas;
- Disseminar a cultura de segurança da informação.

Para isso, a DSSI atua em linhas de pesquisa e desenvolvimento, tais como:

- Segurança de software e Gestão de segurança da informação
- Segurança em Computação em Nuvem para o Governo Brasileiro

- Coleta e Análise de Artefatos Maliciosos (Malware)
- Computação Forense

O projeto submetido no ciclo de 2011 ao contexto do PBQP-Sw se enquadra na linha de Segurança de software e Gestão de segurança da informação e de *Cloud Computing* na DSSI-CTI.

Em Segurança de software e Gestão de segurança da informação são abordadas metodologias de projeto e desenvolvimento seguro visando melhorar a qualidade de softwares e minorar os riscos de segurança. Procedimentos e ferramentas para o exame e teste de aplicação visam descobrir e eliminar vulnerabilidades.

Ataques bem sucedidos a sistemas computacionais são devidos em boa medida a algum tipo de vulnerabilidade. São justamente vulnerabilidades de software as mais exploradas em ataques e os problemas na codificação são os maiores responsáveis pela maior delas. Portanto, para conseguir minimizar o número de vulnerabilidades presentes em softwares é necessário avaliar e auditar procedimentos de programação segura, auditar o código fonte dos mesmos e avaliar o software executável testando-os sob o ponto de vista de segurança.

Dessa forma, servem, por um lado, no caso da organização desenvolvedora, como um guia para avaliar o seu desempenho e propor melhorias e, do outro lado, para a organização que adquire software, uma forma de avaliar como este é desenvolvido, dando maior confiança na sua qualidade e no cumprimento dos prazos.

Um trabalho realizado neste contexto foi para definir, implantar, manter e evoluir a infraestrutura operacional permanente para o Catálogo Nacional de Produtos de Tecnologia Assistiva (CNPTA), com foco em segurança, usabilidade e acessibilidade da informação.

Uma análise de aspectos de desempenho e segurança dos servidores e dos serviços do software do Portal Nacional de Tecnologia Assistiva e do Catálogo de Ajudas Técnicas foram realizadas por um conjunto específico de testes de segurança, usabilidade e acessibilidade.

A computação em nuvem é uma tecnologia que vem sendo adotada com grande velocidade, ela transforma uma infraestrutura física em um ambiente virtual, o que de fato resulta em diversos benefícios, como a redução de custos com infraestrutura e licenças, alta disponibilidade, escalabilidade e outros. A computação em nuvem está sendo usada atualmente, no entanto, a segurança, interoperabilidade e portabilidade são citadas como as principais barreiras para uma adoção mais ampla. Para superar essas barreiras faz-se necessário um estudo mais aprofundado das implicações da adoção, implantação e manutenção de um ambiente de computação em nuvem do ponto de vista de segurança em órgãos públicos, identificando riscos de segurança em TI e ameaças na adoção desta nova tecnologia e qual o impacto do seu uso nesses órgãos. Também é fundamental identificar possíveis problemas durante todo o processo de implantação de um ambiente de nuvem, e baseado nestes problemas desenvolver pesquisas para a solução ou mitigação destes. O objetivo desta linha de pesquisa, a longo prazo, é o de proporcionar orientação em torno do paradigma da computação em nuvem para acelerar a sua adoção no governo. Ao encurtar o ciclo de adoção permitirá economia de custos e aumento da capacidade de rapidamente criar e implantar aplicações.

### **3. Metodologia de execução**

A metodologia de execução para o trabalho realizado para o Catálogo incluiu diversas abordagens da área DSSI

Visando garantir a disponibilidade de acesso, qualidade e segurança de software foram desenvolvidos e aplicados abordagem para avaliação de software para Web, por meio da experiência com a análise de sites disponíveis, avaliando suas vulnerabilidades e montando uma metodologia específica para esse tipo de avaliação. A literatura consultada [Duarte, 2008], [Barbato, 2009] e a experiência em avaliações de software [Guerra, 2009], [CTI/MCT, 2006a], [CTI/MCT, 2006b] foram pontos importantes para o desenvolvimento do projeto e sucesso do mesmo. Foi avaliada a segurança do produto disponibilizado na Web além da realização de um processo de avaliação utilizando-se a metodologia MEDE-WEB consistindo da aplicação de uma lista de verificação composta por itens de

avaliação, que tratam de requisitos de qualidade de usabilidade e de acessibilidade e adicionalmente a aplicação dos *checklists* disponíveis publicamente [CHECKLISTS, 2010].

O e-Mag é o Modelo de Acessibilidade em Governo Eletrônico que contém as diretrizes ou recomendações que normatizam o desenvolvimento de sítios e portais acessíveis. A validação de tais diretrizes pôde ser realizada, em uma primeira fase, por meio de validadores automáticos, mas eles, por si só, não podem determinar de forma absoluta se um sítio está ou não acessível. Uma validação manual, foi feita pela aplicação de *checklists* por avaliadores humanos, é necessária em uma fase posterior, já que nem todos os problemas de acessibilidade em um sítio podem ser detectados mecanicamente pelos validadores automáticos, ou seja, em muitos aspectos é necessário um julgamento humano.

O software disponível na web foi o catálogo de produtos de Tecnologia Assistiva comercializados no Brasil, tendo por missão oferecer informações a cerca de tais produtos. O Catálogo permite a realização de buscas sobre tais informações que contribuem para uma maior autonomia e qualidade de vida das pessoas deficientes, idosas, entre outras. O catálogo se encontra no endereço: “[www.assistiva.org.br](http://www.assistiva.org.br)”.

Dados do Instituto Brasileiro de Geografia e Estatística – IBGE, de 2010, apontam que 23,91% da população brasileira possuem algum tipo de deficiência, totalizando aproximadamente 45,6 milhões de pessoas. Do ponto de vista tecnológico, suportar o anúncio do website em cadeia nacional requereu a utilização de diversos processos e procedimentos, software e hardware. Sendo que estes últimos foram especialmente configurados para momentos de grande quantidade de acesso em curto espaço de tempo e com o nível de segurança adequado.

O objetivo da avaliação da segurança do catálogo foi o de levantar de forma preliminar a superfície de testes que deve ser utilizada com a finalidade da avaliação de disponibilidade e segurança do site. Assim a avaliação da segurança do catálogo foi realizada com teste de desempenho e disponibilidade. Foram realizados testes de *Stress*, *SQL Injection*, transmissão segura dos dados, testes com relação à hospedagem, para garantir a disponibilidade do servidor ou da aplicação.

Foram realizados acessos simultâneos simulados a fim de verificar o comportamento do serviço/servidor que demonstrou sua capacidade de acessos. A degradação do serviço/servidor também foi avaliada e medida, sendo que nessa degradação do serviço se fez necessário o estudo do processo de reinicialização do sistema. No caso, o estudo apresenta possibilidades automatizadas, não triviais e opção de intervenção humana sendo analisada a quantidade de tempo para esse processo.

#### **4. Resultados obtidos e Considerações finais**

- Com o intuito de analisar o desempenho e a segurança dos servidores que hospedam o Portal Nacional de Tecnologia Assistiva e o Catálogo de Ajudas Técnicas, foi realizada uma série de testes para poder identificar possíveis problemas, com isso foi obtido como resultado: “Desenvolvimento de uma abordagem para avaliação de segurança, usabilidade e acessibilidade de sites”;
- Houve uma transferência de conhecimento e experiência, informalmente, desta metodologia para a empresa desenvolvedora do Portal;
- Elaboração de um Publicação Técnica, em forma de Documento Técnico Interno sobre o resultado da avaliação do Portal;
- Publicação de Trabalhos Acadêmicos:
  - ✓ [Capítulo de Livro] Técnicas para análise dinâmica de código malicioso. Vitor M. Afonso, Dario S. Fernandes Filho, Victor F. Martins, André R. A. Grégio, Rafael D. C. Santos, Mario Jino,

Paulo L. de Geus. Livro de minicursos do XI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSEG). Brasília, DF, BR, novembro/2011;

- ✓ [Artigo] Análise Visual de Comportamento de Código Malicioso. Autores: Alexandre O. C. Baruque, André R. A. Grégio, Paulo L. de Geus. XI SBSeg - Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais. Brasília, DF, BR, novembro/2011;
  - ✓ [Artigo] A hybrid system for analysis and detection of web-based client-side malicious code. Autores: Vitor M. Afonso, André R. A. Grégio, Dario S. Fernandes Filho, Paulo L. de Geus; ICWI'2011 – IADIS International Conference WWW/Internet. Rio de Janeiro, RJ, BR, novembro/2011;
  - ✓ [Artigo] Sistema de coleta, análise e detecção de código malicioso baseado no sistema imunológico humano. Autores: Isabela L. Oliveira, André R. A. Grégio, Adriano M. Cansian. CIAWI'2011 – Conferência IADIS Ibero-Americana WWW/Internet. Rio de Janeiro, RJ, BR, novembro/2011;
  - ✓ [Artigo] Visualization techniques for malware behavior analysis. Autores: André R. A. Grégio, Rafael D. C. Santos. Technologies for Homeland Security and Homeland Defense. Proceedings of SPIE, Vol. 8019. Orlando, FL, USA, abril/2011. ISBN: 9780819485939;
  - ✓ [Artigo] Behavioral analysis of malicious code through network traffic and system call monitoring. Autores: André R. A. Grégio, Dario S. Fernandes Filho, Vitor M. Afonso, Rafael D. C. Santos, Mario Jino, Paulo L. de Geus. Evolutionary and Bio-Inspired Computation. Proceedings of SPIE, Vol. 8059. DOI: 10.1117/12.883457. Orlando, FL, USA, abril/2011. ISBN: 9780819486332;
  - ✓ [Artigo] - "*Evaluation process for software application security*"; Autores: Regina M T Colombo, Ana Cervigni Guerra, Amandio Balcão Filho, Marcelo S Pessôa. 23th International Conference Software & Systems Engineering and their Applications – Paris, France. ICCSEA2011.
- Capacitação de 15 técnicos, entre colaboradores e servidores públicos no curso em “Inteligência e Segurança da Informação” – 40h;
  - Dois doutorandos trabalhando com Segurança de Software e que fizeram a qualificação em 2011;
  - Dois novos servidores em fase inicial de doutorado.

## 5. Aplicabilidade dos Resultados

**Metodologia de avaliação de sites** – Essa metodologia pode ser utilizada em diversos projetos, por exemplo, no plano do governo federal ‘**Viver sem Limite**’ que é o nome dado ao Plano Nacional dos Direitos da Pessoa, onde um conjunto de ações do Governo Federal busca ampliar a integração da pessoa com deficiência na sociedade. ‘Viver sem Limite’ está baseado nos eixos da educação, saúde, inclusão social, tecnologia assistiva e acessibilidade. As ações são executadas em conjunto por 15 órgãos do Governo Federal, sob a coordenação da Secretaria de Direitos Humanos da Presidência da República (SDH/PR) e tem metas a serem implantadas até 2014.

O Plano Nacional ‘Viver Sem Limite’, lançado no dia 17/11/2011, contempla como uma de suas atividades, por intermédio do Ministério da Ciência, Tecnologia e Inovação – MCTI, o ‘Catálogo Nacional de Produtos de Tecnologia Assistiva’ – CNPTA. Cerca de 1200 produtos disponíveis no mercado brasileiro estão reunidos no CNPTA, que está acessível na Internet através da URL <http://assistiva.mct.gov.br>, em uma parceria entre o MCTI e o Instituto de Tecnologia Social – ITSBRASIL.

O CNPTA responde a uma demanda social. As pessoas com deficiência e idosas, assim como suas famílias, os profissionais de reabilitação, as organizações da sociedade civil e os órgãos públicos que prestam serviços para pessoas com deficiência e idosas, precisam de informações que os auxiliem.

O resultado da ação da DSSI possibilitou que o Portal e os servidores apresentassem boa qualidade na disponibilidade do sítio. É importante que sejam realizadas as melhorias sugeridas no relatório de avaliação do Portal a fim de evitar problemas de segurança e desempenho. As melhorias já realizadas foram no software em conjunto com a empresa desenvolvedora do Portal.

Outro exemplo de aplicabilidade deste tipo de resultado está relacionado com o desdobramento da publicação da **Lei da Transparência da Informação – Lei 12527, 2011. A Lei 12.527** de 18 de novembro de 2011 dispõe sobre a garantia de acesso a informações de órgãos públicos, como previsto na Constituição Federal. Esta lei cria uma série de obrigações para os órgãos públicos, com forte impacto nos custos de TI, no desenvolvimento de novas aplicações e soluções.

As competências descritas neste artigo são de aplicabilidade imediata para atender as determinações da referida lei. Tanto as metodologias e técnicas de avaliação das aplicações bem como o uso da computação em nuvem, como infraestrutura para atender os novos requisitos de transparência de dados públicos. Desse modo, as orientações em torno do paradigma da computação em nuvem visa acelerar a sua adoção no governo e encurtar o seu ciclo, proporcionando economia de custos e aumento da capacidade de rapidamente criar e implantar novas aplicações.

## **6. Características Inovadoras**

Este projeto tem como características inovadoras os seguintes aspectos:

**Relevância:** As práticas de pesquisa e desenvolvimento do projeto reúnem elementos que descrevem características de um processo efetivo e resultam em melhores resultados para a comunidade de qualidade e segurança de software que é uma das políticas do governo para melhorar o software brasileiro.

**Abrangência:** É um projeto de âmbito nacional e internacional, sendo que temos parceiros fora do país e serviços/consultoria tanto para o governo brasileiro como para empresas do setor.

**Inovação:** No Brasil as iniciativas ainda são embrionárias na área acadêmica e no mercado para a melhoria de qualidade de software e sistemas com enfoque em segurança.

**Impacto:** No caso de organizações desenvolvedoras de software, esse projeto serve como um guia para avaliar o seu desempenho e propor melhorias e, do outro lado, para organizações que adquirem software, é uma forma de avaliar como este é desenvolvido, dando uma maior confiança na sua qualidade e segurança.

Estas características inovadoras estão presentes nos projetos desenvolvidos neste ciclo para atender as demandas de governo e por conseguinte da academia e do mercado.

## **7. Conclusão e Perspectivas Futuras**

Este projeto atendeu a sua proposta para este ciclo e possibilitou comprovar que:

- Pesquisar e desenvolver novas tecnologias nesta área é de suma importância para o país;
- É preciso que as aplicações sejam projetadas, desenvolvidas e implantadas com requisitos de segurança e avaliadas para garantir desempenho com qualidade;
- Segurança de software e de suas aplicações tornou-se um grande diferencial para o software brasileiro cuja demanda é mundial;
- Segurança da informação é importante em todas as áreas.

## 8. Referências Bibliográficas

- Guerra, A. Cervigni e Colombo, R. M. Thienne (2009) "Tecnologia da Informação: Qualidade de Produto de Software", Editado por PBQP Software.
- Centro de Tecnologia da Informação Renato Archer - CTI/MCT - Tecnologia de Avaliação de Qualidade de Produto de Software, "MEDE-PROS - Método de Avaliação da Qualidade de Produto de Software, versão 2006",
- Centro de Tecnologia da Informação Renato Archer - CTI/MCT - "MEDE-WEB - Método de Avaliação da Qualidade de Websites, versão 2005".
- Ministério do Planejamento, Orçamento e Gestão, Secretaria de Logística e Tecnologia da Informação; Ministério da Educação, Secretaria de Educação Profissional e Tecnológica (2011) "e-Mag Versão 3.0 - Modelo de Acessibilidade em Governo Eletrônico".
- Checklists - Núcleo de Acessibilidade Virtual do IFRS - Campus Bento Gonçalves; Núcleo de Acessibilidade Virtual do IFCE - Campus Fortaleza; Núcleo de Acessibilidade Virtual do IFBaiano - Campus Catu e Campus Guanambi (2010) "Checklist de Acessibilidade Manual para o Desenvolvedor, Checklist de Acessibilidade Manual para Deficientes Visuais, e-Mag Modelo de Acessibilidade em Governo Eletrônico".
- Barbato, L. G. C., Metodologia de Identificação de Vulnerabilidades em Aplicações de Pagamento utilizado Cartões de Crédito – Tese (Doutorado) INPE São José dos Campos, 2009
- Duarte, L. O., Desenvolvimento de um ambiente para análise de códigos-fonte com ênfase em segurança. Dissertação (Mestrado), São José dos Campos: INPE, 2008.

## **SPIN São Paulo 15 anos de contribuição à melhoria da qualidade de software.**

Renato Luiz Della Volpe<sup>1</sup>, Ernesto de Carvalho Bedrikow<sup>2</sup>, Antônio Carlos Tonini<sup>3</sup>, Valdir Cardoso da Silva<sup>4</sup>

<sup>1</sup> SPIN SP – Presidente; ASR Consultoria e Assessoria em Qualidade Ltda.

Rua Doutor Deodato Wertheimer, 1352 - sala 11 – Mogi das Cruzes – SP – Brasil

<sup>2</sup> SPIN SP - Diretor de Infra; Professor das Faculdades Sumaré e Uniesp e Consultor em qualidade e sistemas específicos

<sup>3</sup> Antonio Carlos Tonini Serviços Auxiliares ME

Rua Apiacás, 559 ap 74 – Vila Pompéia – São Paulo – cep 05017-020

4 MV Informática Ltda

Rua Laura 698 apto 51 - Santo André - SP - Brasil

(renatovolpe@asrconsultoria.com.br; ernesto@spinsp.org.br; actonini@uol.com.br; valdircardosodasilva@gmail.com)

### **1. Introdução**

A evolução e a busca contínua de melhorias na gestão organizacional das empresas de tecnologia da informação [MCT-2010 e WAMPS 2011] estão em evidência nas empresas que desenvolvem ou utilizam serviços e produtos de software e correlatos. Como catalisadores destas melhorias, estão presentes diversos fatores como crescimento das organizações, acirrada competitividade no mercado, busca de melhores resultados financeiros e operacionais, e como consequência, estão a preocupação e o entendimento de que resultados somente serão obtidos com o estabelecimento e a melhoria dos processos organizacionais. Felizmente existem diversos esforços e iniciativas governamentais e de entidades privadas para prover as empresas do conhecimento e a capacitação nas novas tendências da gestão da qualidade em TI. Nesse contexto o movimento SPIN – *Software Process Improvement Network* [SEI] possui como principal filosofia o compromisso com a comunidade de atuar, de forma voluntária e sem custos, para o intercâmbio de ideias, informações e apoio mútuo. Este relatório apresenta os resultados, benefícios, lições aprendidas e a experiência obtida pelo grupo SPIN – São Paulo, um projeto que se mantém ativo por 15 anos.

### **2. Objetivos e Justificativa**

O movimento SPIN – Software Process Improvement Network estabelecido desde a década de 1980 pelo SEI – Software Engineering Institute [SEI] se desdobrou por todo o mundo com o objetivo de divulgar as boas práticas e experiências de melhoria da qualidade de software. O SPIN é uma organização de profissionais em determinada wárea geográfica que se dedicam, de forma voluntária, à melhoria de software e engenharia de sistemas.

Participar de um SPIN requer compromisso e lealdade para com a melhoria do estado de software e engenharia de sistemas, bem como coloca o profissional em contato com uma rede de conhecimentos dentro de sua comunidade. É um fórum prático para o intercâmbio de ideias, informações e apoio mútuo. Os SPIN's são compostos de profissionais de todos os setores - indústria, governo, academia, incluindo estudantes.

O SPIN – São Paulo, fundado em 1997, foi o primeiro SPIN estabelecido no Brasil e continua em pleno exercício de suas atividades, incentivando a busca de melhores práticas de mercado da qualidade de software e engenharia de sistemas.

#### **2.1 Objetivos contínuos do SPIN-SP**

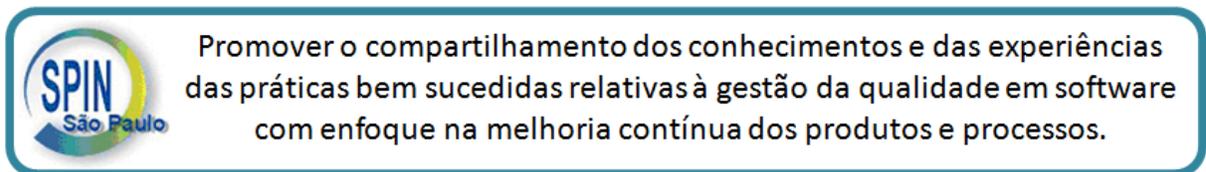
- a) Levantar e apresentar os resultados, benefícios, lições aprendidas e a experiência do SPIN – São Paulo;
- b) Motivar a criação ou continuidade das atividades de outros SPIN's no Brasil;
- c) Prover condições de conscientizar pessoas das áreas de TI e empresários sobre a necessidade estratégica de atuar na melhoria da qualidade do software e de seus produtos e serviços;
- d) Disseminar novos conhecimentos para a comunidade;
- e) Promover a conscientização sobre os benefícios do investimento na melhoria dos processos e produtos de software e
- f) Estabelecer e manter vínculos com grupos e instituições que possuam objetivos similares do SPIN-SP.

## 2.2 Objetivos específicos do ciclo

- a) Apresentar os resultados, benefícios, lições aprendidas e a experiência obtidos nos 15 anos de atuação do grupo SPIN – São Paulo.
- b) Apresentar as motivações, dificuldades, estratégias e a base de dados de conhecimento construída no período de atuação;

## 2.3 Motivações

A principal motivação e justificativa de continuidade da atuação do SPIN-SP está descrita em sua Missão conforme Figura 1.



**Figura 1: Missão do SPIN-SP**

Também como motivação, está inserida na filosofia de qualquer SPIN e que deve ser constantemente lembrada para as atividades do SPIN-SP é uma frase de W. Edwards Deming “*When we cooperate, everybody wins*”. Contudo, esta cooperação no mercado de TI ainda não é fortemente praticada por motivos como concorrência, receio de divulgar práticas e resultados, entre outros motivos. Neste intuito, o SPIN-SP atua no sentido de “quebrar” estas barreiras hoje existentes.

## 3. Metodologia de Execução

O desenvolvimento do “projeto” SPIN-SP e seus resultados são contínuos e suas atividades foram estruturadas e estabelecidas ao longo do período de sua atuação. A estratégia para o projeto do ciclo 2011 é de apresentar como está a estruturação do conhecimento catalogado e de acesso público do SPIN – São Paulo, bem como mostrar a evolução de suas atividades e network desde sua constituição.

Como entregáveis do “projeto” durante o período de atuação desses 15 anos podemos listar, entre outros:

- apresentar o histórico do SPIN - São Paulo que contribui para disseminação do conhecimento e motivação do conceito SPIN ,
- apresentar os métodos e cuidados para implementação de um SPIN,
- prover benefícios para a comunidade de sua atuação com a geração de conhecimento e incentivo ao network entre empresas e profissionais,
- disponibilizar lições aprendidas e o conhecimento acumulado tornando todos os materiais e apresentações de acesso público. Esta Gestão de Conhecimento inclui também: informações e dados obtidos no mercado, divulgação de eventos, divulgação de modelos e métodos e de novas tendências do mercado.
- motivar a constituição de outros SPIN's para atuarem em suas regiões aumentando a abrangência e o movimento SPIN.

## 4. Resultados Obtidos

Entre os resultados obtidos pelo SPIN-SP nestes 15 anos de atividades podem ser destacados:

- a. Desde sua 1ª reunião em abril 1997 com a participação de 27 profissionais observamos uma constância na participação dos eventos do SPIN-SP, com uma média que varia de 60 a 90 pessoas por evento;
- b. Realização até 2011 de 59 reuniões, sendo 3 com palestrantes/ eventos internacionais;
- c. Lista de divulgação com aproximadamente 1500 pessoas onde sempre há o respeito à confidencialidade desta informação em todas as mensagens, não sendo utilizada a lista para outras finalidades que não sejam o network do SPIN-SP. A evolução do network do SPIN-SP pode ser verificado no Gráfico 1;
- d. Geração e manutenção do site do SPIN-SP - <http://www.spinsp.org.br> – ver na Figura 2 que apresenta todos seus campos e onde se pode encontrar os dados da Gestão do Conhecimento anteriormente citada (o SPIN-SP também está referenciado no SEI em [www.sei.cmu.edu/spin/find/index.cfm](http://www.sei.cmu.edu/spin/find/index.cfm) ;
- e. Apoio e Participação a eventos como:

- Participação e Apoio ao CITS - Conferência Internacional de Tecnologia de Software nos anos de 2002 e 2003;
- Patrocínio, Apoio e Palestras em nos eventos SIMPROS - Simpósio Internacional de Melhoria de Processo de Software de 2003 até 2006;
- Apoio na realização do SEPG-LA – 2006
- Apoio e Palestras em todas as Conferências da Qualidade ASR Consultoria nos anos de 2008, 2009, 2010, 2011;

A Tabela 1 mostra que o envolvimento do SPIN-SP é constante nos últimos anos em eventos voltados à Qualidade de Software.

- Promoção e incentivo na formação dos SPIN's de Sorocaba, Ribeirão Preto, Cuiabá, São José dos Campos e Goiânia;
- Apresentações no meio acadêmico como USP, FMU, Universidade São Judas, UNIC, Fundação Vanzolini, Universidade Braz Cubas, PUC-PR, UNIP, SENAC; Colégio Batista Brasileiro, Faculdade de Tecnologia de São Paulo, ESEG, Universidade Mackenzie, Universidade Gama Filho.
- Em termos de apresentações o SPIN-SP:
  - sempre apresenta e busca colocar nas palestras os temas mais atuais para a comunidade. Exemplos destes temas podem ser vistos na Figura 3.
  - possui um acervo de materiais que pode ser observado no Gráfico 2, além de demais materiais, referências, modelos, entre outro materiais disponíveis para a comunidade;
- Elaboração do “Guia para Criar um SPIN” que descreve orientações para o auxílio no estabelecimento de grupos SPIN no Brasil em estados ou regiões que estejam interessadas na melhoria de processos de software e na sua divulgação no âmbito do País. A Figura 3 mostra a capa do “Guia para Criar um SPIN” que está disponível para “download” no “site” do SPIN-SP. Neste Guia colocamos toda a experiência, lições aprendidas, benefícios, cuidados e riscos da realização das atividades do SPIN-SP.



Figura 2: “Site” do SPIN-SP e o conteúdo da Gestão do Conhecimento

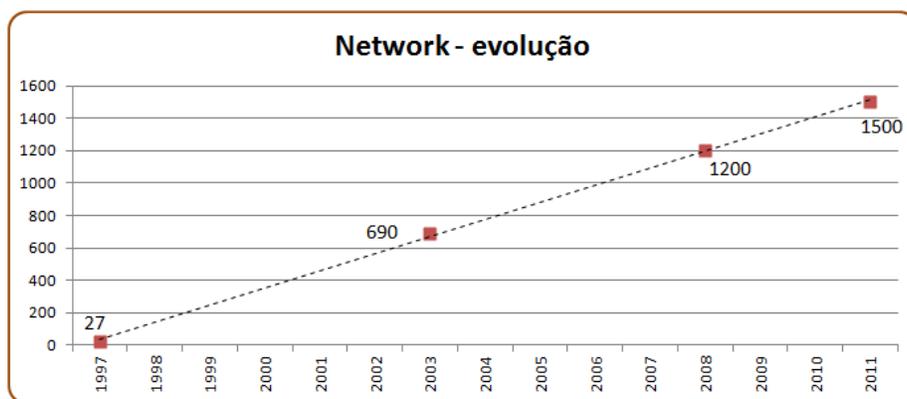


Gráfico 1: Evolução do Network do SPIN-SP

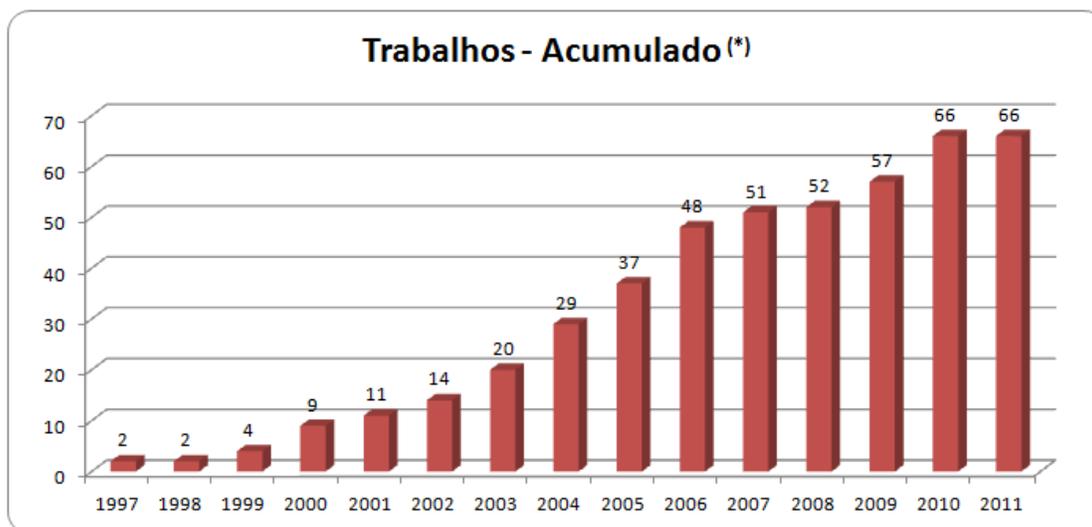


Gráfico 2: Materiais disponíveis no “Site” do SPIN-SP

(\*) nem todos os palestrantes disponibilizaram os materiais principalmente no período de 1997 a 2002

Tabela 1: Participação e Envolvimento do SPIN-em eventos da qualidade

EVENTO	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011
CITS-PR	●	●								
SIMPROS		●	●	●	●					
SEPG-LA					●					
Conferência ASR							●	●	●	●

Período em que não ocorreu o evento no Brasil

Participação/ envolvimento do SPIN-SP

## 5. Aplicabilidade dos Resultados

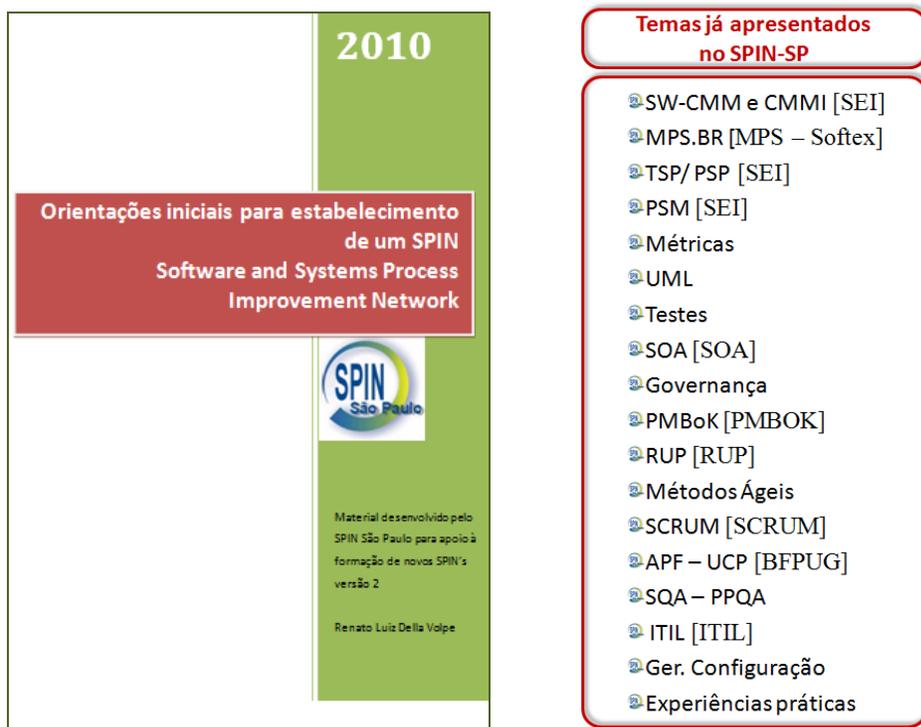
Todos os resultados apresentados principalmente materiais, links existentes no site para modelos, trabalhos, guias entre outros, pela própria filosofia dos grupos SPIN's podem ser utilizados por toda a comunidade sem nenhum tipo de associação ou investimento requerido. Além disso, o “network” existente contribui para que qualquer pessoa

consiga contatar palestrantes e coordenadores do SPIN-SP. “Network” este que é obtido nos encontros, principalmente nos intervalos do café no qual chamamos de a “hora do network” quando os participantes discutem entre si problemas que julgam que são só seus e percebem que outras empresas passam pelas mesmas dificuldades. Desses encontros “informais” surgem inclusive parcerias, estudos, troca de informação, negócios, entre outros.

Com base nos resultados e sua divulgação, todas as pessoas do “network” podem sugerir temas, locais, palestrantes para que seja analisado pela coordenação do SPIN-SP, contribuindo com a continuidade destes resultados.

Com relação à relevância dos resultados, o projeto possui a importância de apresentar um caso de sucesso no Brasil de um movimento mundial e globalmente reconhecido. O impacto dos resultados dos encontros que são realizados é positivo, pois as pessoas que participam, o fazem devido ao interesse em determinado tema que deseja conhecer ou se aprofundar. Apesar de, em uma primeira impressão a participação, de 60 a 90 pessoas por evento/encontro do SPIN-SP, em uma lista de 1500 pessoas, parecer baixo (4% a 6%), este número é bem significativo se forem lavados em conta que vários fatores afetam nesta participação:

- Interesse exclusivo em determinado tema pelos participantes/ inscritos;
- Empresas encaminham/ liberam somente alguns representantes para cada evento mesmo sendo gratuito;
- Dificuldade de agenda dos inscritos;
- Evento gratuito onde nem sempre os inscritos tem sua presença confirmada, entre outros fatores.



**Figura 3: Guia de “Como Criar um SPIN” versão 2 e Exemplos de Temas das Palestras já realizadas.**

## 6. Características Inovadoras

O projeto está alinhado ao movimento mundial do SPIN [SEI] que é um fórum prático para o intercâmbio de ideias, informações e apoio mútuo. O próprio SEI [SEI] esclarece que cada SPIN regional é ligeiramente diferente, com base na visão dos fundadores e as necessidades da comunidade.

O SPIN São Paulo sempre atua com o objetivo de divulgar a seus membros (network) as mais recentes tecnologias e métodos em discussão no meio do desenvolvimento de software. Por vezes existindo apresentações de especialistas com níveis distintos de visões da melhor forma de aplicação destas metodologias o que contribui para que os membros do SPIN – São Paulo possam conhecer e debater junto aos especialistas as formas de adoção e aplicação em suas organizações.

Uma característica importante de ser colocada é que na maioria dos temas apresentados (ver Figura 3), a coordenação SPIN-SP busca palestrantes e experiências práticas que os correlacionassem com os modelos MPS.BR - Melhoria do Processo do Software Brasileiro [SOFTEX 2011] e CMMI-DEV *Capability Maturity Model Integration for Development* [CMMI 2010] (ou anteriormente o SW-CMM). Mesmo recentemente, com o grande interesse em metodologias ágeis, há sempre a preocupação em mapear e descrever formas de sua adaptação a esses dois modelos. A Figura 4 apresenta sucintamente esses dois modelos que qualificam as organizações em seu nível de maturidade. No caso do CMMI a maturidade é crescente do nível 1 para o 5 e no MPS.BR do nível G para A. Para maiores informações recomendamos analisar as Referências Bibliográficas.

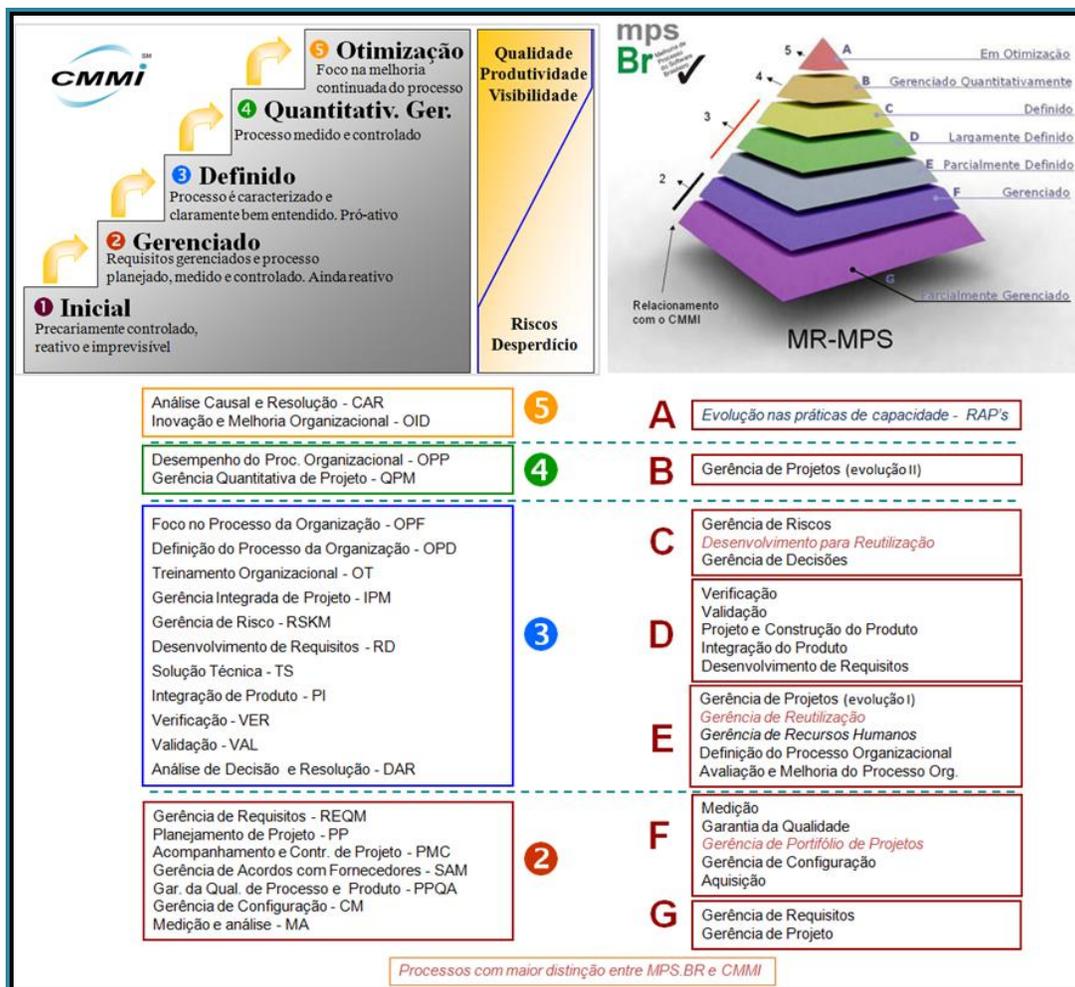


Figura 4: Estrutura básica e processos dos modelos MR-MPS e CMMI-DEV.

## 7. Conclusão e Perspectivas futuras

Ainda existem muitas dificuldades para a realização das atividades do SPIN-SP devido ao fato de serem atividades voluntárias de pessoas que atuam no mercado e que possuem grandes restrições de disponibilidade de agenda, em função de diversas outras responsabilidades e compromissos. Alinhado a isto, o fato do SPIN-SP não possuir nenhum recurso financeiro requer que sejam disponibilizados por empresas ou entidades que apoiam o SPIN-SP, os recursos de logística como local, infraestrutura, *datashow*, disponibilização de café aos participantes, entre outros.

Por outro lado temos positivamente a participação, também de forma voluntária, dos palestrantes que sempre contribuíram imensamente nos encontros realizados com sua experiência, pró-atividade e entendimento da filosofia SPIN. Também não podemos esquecer de citar a contribuição de diversas pessoas que sempre nos ajudam na realização dos encontros seja de forma sistêmica ou pontualmente em alguns eventos. Entidades acadêmicas e empresas que nos ajudam também não podem ser esquecidas.

Citar todos os nomes das pessoas ou entidades/ empresas poderia causar um injustiça caso deixássemos de fazer algumas citações, mas reiteramos que todas estas pessoas e entidades/ empresas estão referenciadas em nosso Banco de Conhecimento que está em nosso “site”.

Contudo, não podemos deixar de citar duas pessoas, além da atual coordenação, que muito contribuíram com os resultados. O primeiro é o Prof. Doutor Atila Belloquim Costa, atualmente na Gnosis IT Knowledge Solutions, que em 1997 atuava na Choose Technologies.e trouxe para o Brasil, após visita ao SEI, o conceito SPIN e organizou e realizou os primeiros encontros do SPIN-SP. O segundo é o Prof. Doutor Mauro de Mesquita Spinola que atua como professor da Universidade de São Paulo e na Fundação Vanzolini e que deu continuidade nas atividades e coordenação do SPIN-SP.

Apesar das dificuldades encontradas podemos considerar que os resultados obtidos e a própria manutenção do SPIN-SP por estes 15 anos demonstram que com esforços de pessoas interessadas e entidades que entendem e contribuem com o movimento SPIN, é possível se obter resultados favoráveis para o mercado de TI. O objetivo da atual coordenação do SPIN-SP é manter o nível de atividades e resultados até então obtidos.

## **8. Referências Bibliográficas**

BFPUG - Brazilian Function Point Users Group - <<http://www.bfpug.com.br/>>

CMMI. Capability Maturity Model® Integration (CMMISM), Version 1.3 CMMI-DEV for Development (CMMI-DEV, V1.3), 2010

ITIL. Official ITIL® Website – ITILV3 - <<http://www.itil-officialsite.com/>>, 2007

MCT. Ministério da Ciência e Tecnologia. Secretaria de Política de Informática e Automação. Qualidade e Produtividade no Setor de Software Brasileiro. Disponível em: <<http://www.mct.gov.br/sepim/>>. Acesso em: 2010

PMBOK - Project Management Body of Knowledge - <<http://www.pmi.org/PMBOK-Guide-and-Standards.aspx>>

RUP - Rational Unified Process (RUP) - <<http://www-01.ibm.com/software/awdtools/rup/>>

SCRUM. InofQ - Agile Community Content on InfoQ - Scrum and XP from the Trenches - <<http://www.infoq.com/agile>>, 2009

SEI – Software Engineering Institute – Carnegie Mellon University Pittsburg <<http://www.sei.cmu.edu/spin/>>

SOA – Service-oriented architecture - <<http://www.soa.org/>>

SOFTEX. Guia Geral:2011 (Junho de 2011, atualizado em Agosto de 2011) – Melhoria de Processo de Software Brasileiro <[http://www.softex.br/portal/mpsbr/\\_guias/default.asp](http://www.softex.br/portal/mpsbr/_guias/default.asp)>

WAMPS 2011 – Workshop Anual MPS - <[http://www.softex.br/mpsbr/\\_wamps/default.asp](http://www.softex.br/mpsbr/_wamps/default.asp)>

# Consolidação do Programa MPS.BR – Melhoria de Processo do Software Brasileiro (2008-2011)<sup>1</sup>

Kival Weber<sup>1</sup>, José Antonio Antonioni<sup>1</sup>, Nelson Franco<sup>1</sup>, Elidiane Barroso<sup>1</sup>, Cleide Silva<sup>1</sup>, André Sotovia<sup>1</sup>

<sup>1</sup>SOFTEX – Associação para Promoção da Excelência do Software Brasileiro

{kival.weber, jaa, nelson, elidiane, cleide, andre}@nac.softex.br

**Abstract.** *This paper describes the main results of the MPS.BR program in the 2008-2011 term, focusing on 2011, according to the 2.01 Project – PBQP Software 2011: MPS.BR Consolidation (2008-2011). It shows that MPS.BR is a successful program both improving software process, in large organizations and SME, and increasing competitiveness of the Software Industry in Brazil.*

**Resumo.** *Este artigo descreve os principais resultados do programa MPS.BR na etapa 2008-2011, com foco em 2011, conforme disposto no Projeto 2.01 – PBQP Software 2011: Consolidação do MPS.BR (2008-2011). O artigo mostra que o MPS.BR é um programa bem sucedido tanto na melhoria dos processos de software, em grandes organizações e em PME, quanto no aumento da competitividade da Indústria de Software no Brasil.*

## 1. Introdução

O programa MPS.BR – Melhoria de Processo do Software Brasileiro, coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), foi criado em 11 de dezembro de 2003 com dois objetivos (desafios): i) um técnico, visando à criação e ao aperfeiçoamento do modelo MPS; ii) outro de negócio, visando à difusão do modelo MPS no mercado. O MPS.BR é um programa bem sucedido, com excelentes resultados alcançados [1].

O modelo MPS está em conformidade com as normas internacionais ISO/IEC 12207 (*Software Life Cycle Process*) e 15504 (*Process Assessment*), é compatível com o CMMI, adota as melhores práticas da engenharia de software e é apropriado tanto às pequenas e médias empresas (PME) quanto às grandes organizações privadas e governamentais.

A Figura 1 mostra os três componentes do modelo MPS: i) modelo de referência MR-MPS; ii) método de avaliação MA-MPS; iii) modelo de negócio MN-MPS. O modelo MPS é documentado em Guias MPS, publicadas em < [www.softex.br/mpsbr](http://www.softex.br/mpsbr) >.

O MPS.BR é um programa de longo prazo, como o CMMI ( ver [www.sei.cmu.edu/cmmi](http://www.sei.cmu.edu/cmmi) ) que começou com o CMM em 1991, com antecedentes desde 1988. Assim, o planejamento plurianual do programa MPS.BR compreende etapas de quatro anos: i) 2004-2007 – Introdução do MPS.BR; ii) 2008-2011 – Consolidação do MPS.BR; iii) 2012-2015 – Expansão do MPS.BR.

A etapa 2004-2007 – Introdução do MPS.BR foi muito bem sucedida e conquistou tanto o prêmio de melhor artigo técnico do SBQS em 2005 [2] quanto o prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software em 2007 [3].

Este artigo descreve os principais resultados do programa MPS.BR na etapa 2008-2011, com foco em 2011, conforme disposto no Projeto 2.01 - PBQP Software 2011: Consolidação do MPS.BR (2008-2011). A seção 2 apresenta os objetivos e a justificativa do projeto no que diz respeito ao ciclo 2011 do PBQP Software. Na seção 3 é descrita sua metodologia de execução. Na seção 4 são apresentados os principais resultados obtidos. A seção 5 mostra a aplicabilidade dos resultados. Na seção 6 são descritas as suas características inovadoras. Na seção 7, além da conclusão do artigo, são apresentadas as perspectivas futuras do MPS.BR.

---

<sup>1</sup> Projeto 2.01 do PBQP Software – Ciclo 2011 (MCTI/SEPIN)

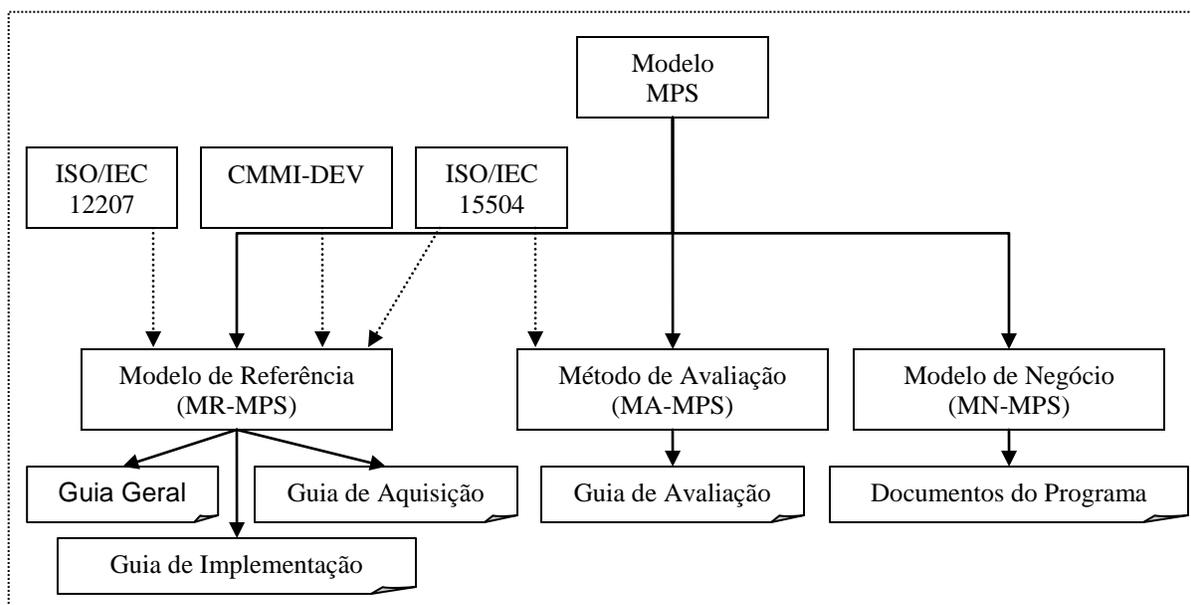


Figura 1 – Componentes do Modelo MPS

## 2. Objetivos e Justificativa

O objetivo do projeto 2.01 – PBQP Software: Ciclo 2011 é a consolidação do programa MPS.BR no período 2008-2011, com foco em 2011, visando à melhoria de processo do software brasileiro e ao aumento da competitividade da Indústria de Software no Brasil. Quanto ao cronograma e estágio de desenvolvimento, este é o quarto de quatro projetos realizados com a mesma denominação nos seguintes ciclos anuais do PBQP Software: 1/4 submetido em 2008, 2/4 resubmetido em 2009, 3/4 resubmetido em 2010 e 4/4 resubmetido em 2011; ou seja, este projeto só foi concluído em 2011 com os resultados alcançados no final da segunda etapa quadrienal do programa MPS.BR.

Na primeira etapa (Introdução do MPS.BR: 2004-2007), os resultados alcançados superaram os resultados esperados [3]. Esta segunda etapa (Consolidação do MPS.BR: 2008-2011) teve três objetivos:

- consolidar o programa MPS.BR, aumentando sua sustentabilidade institucional, operacional e financeira;
- aprimorar o modelo MPS, mantendo tanto sua conformidade com as normas internacionais ISO/IEC 12207 e 15504 quanto sua compatibilidade com o CMMI;
- disseminar o modelo MPS em um número significativo de organizações, tanto em PME (foco principal) quanto em grandes organizações, aumentando a escala de pessoas capacitadas no modelo MPS e de avaliações MPS publicadas.

A motivação para a criação do programa MPS.BR e a justificativa para este projeto do PBQP Software decorreram de estudos realizados no início dos anos 2000 que mostraram [4, 5]: i) ser necessário um esforço significativo para aumentar a maturidade dos processos de software nas empresas brasileiras; ii) que, na última década do milênio passado, 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. Naquela ocasião, no nível mais alto de maturidade, a Índia tinha 32 empresas no nível 5 do CMM, a China uma e o Brasil nenhuma. Naquela época, 214 empresas que desenvolviam software no Brasil tinham certificação ISO 9000 enquanto eram apenas trinta as empresas com certificado SEI/CMU de avaliações CMM. Dentre estas trinta, a maioria subsidiária no país de grandes empresas multinacionais, havia: 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, no final de 2003, a SOFTEX propôs a criação do programa MPS.BR – Melhoria de Processo do Software Brasileiro.

Em atenção à orientação de focalizar este artigo no que diz respeito ao ciclo 2011 do PBQP Software, primeiro serão apresentados os resultados obtidos em 2011; mas, por se tratar de um programa de longo prazo, também serão apresentados os resultados acumulados até agora.

### 3. Metodologia de Execução

Este projeto do PBQP Software foi executado pela estrutura organizacional do programa MPS.BR, com responsabilidades atribuídas tanto ao pessoal da SOFTEX quanto aos colaboradores da Academia, Governo e Indústria (Tripla Hélice) [6]. Esta estrutura organizacional tem a seguinte composição:

- Unidade de Execução do Programa (UEP), coordenada pela SOFTEX, responsável por definir estratégias e gerenciar as atividades do programa.
- Equipe Técnica do Modelo (ETM), composta por membros convidados pela SOFTEX (escolhidos entre profissionais com larga experiência em engenharia de software e melhoria de processos de software) para: i) criação e aprimoramento contínuo do MR-MPS e MA-MPS através dos Guias MPS; ii) capacitação de pessoas por meio de cursos, provas e workshops oficiais MPS.
- Fórum de Credenciamento e Controle (FCC) constituído por representantes da Academia, Governo e Indústria, responsável por: i) emitir parecer que subsidie decisão da SOFTEX sobre o credenciamento de Instituições Implementadoras MPS (II) e Instituições Avaliadoras MPS (IA); ii) monitorar os resultados das II e IA, emitindo parecer propondo à SOFTEX o seu descredenciamento no caso de comprometimento da credibilidade do modelo MPS.
- Conselho de Gestão do Programa (CGP) que se reúne semestralmente para apoiar a SOFTEX no planejamento das atividades anuais e no acompanhamento da execução dessas atividades. O CGP é composto pelas diversas partes interessadas (*stakeholders*) no programa MPS.BR: direção da SOFTEX, membros da UEP, um representante por entidade com convênio em vigor (IA, II, IOGE – Instituições Organizadoras de Grupos de Empresa e ICA – Instituições de Consultoria de Aquisição), membros do FCC e membros da ETM (*senior advisors*, coordenador e vice-coordenador geral, e coordenadores das nove áreas da ETM: Apoio à Internacionalização do Modelo MPS, Aquisição MPS, Avaliação MPS, Comunicação MPS, Cursos MPS, Implementação MPS, MP Serviços, Publicações MPS e WAMPS – Workshop Anual do MPS).

O programa MPS.BR é executado tanto com recursos próprios, oriundos de receitas de serviços MPS, quanto de recursos de terceiros tais como apoios do MCTI/SEPIN, FINEP, SEBRAE e BID/FOMIN.

### 4. Resultados Obtidos

A seguir, são apresentados os principais produtos e resultados obtidos no ciclo 2011 do projeto PBQP Software e os principais resultados acumulados no programa MPS.BR desde o início.

#### 4.1. Produtos gerados que foram disponibilizados para o mercado

O principal produto disponibilizado para o mercado é o significativo número de avaliações oficiais MPS realizadas em empresas desde setembro de 2005, a um custo acessível, e publicadas em ( [www.softex.br/mpsbr](http://www.softex.br/mpsbr) ). No ciclo 2011 do projeto PBQP Software, foram realizadas 70 avaliações MPS (39 no nível G – o mais baixo, 16 no nível F que é equivalente ao nível 2 do CMMI, 4 no nível E, 1 no nível D, 9 no nível C que é equivalente ao nível 3 do CMMI, e 1 no nível A – o mais alto que é equivalente ao nível 5 do CMMI) . Na etapa 2008-2011, foram realizadas 272 avaliações MPS. No total, de 2005 a 2011, foram realizadas 344 avaliações MPS. Deste total de avaliações MPS, cerca de 70% foram realizadas em PME e 30% em grandes organizações – o que mostra que o modelo MPS é apropriado tanto às organizações de grande porte quanto à PME e que o mesmo modelo pode ser usado ao longo do tempo inclusive quando cresce o porte da empresa.

Há mais de 70 avaliadores treinados e autorizados pela SOFTEX para realizar avaliações oficiais MPS. Há 18 II e 12 IA autorizadas pela SOFTEX para prestar, respectivamente, serviços de implementação e avaliação MPS a clientes interessados na adoção do modelo.

Dentre as avaliações MPS realizadas há tanto avaliações complementares quanto avaliações conjuntas com avaliações CMMI, pois os dois modelos são plenamente compatíveis [7].

Além disso, no período 2005-2011, foram realizadas cerca de 150 avaliações CMMI no Brasil que, somadas às 344 avaliações MPS, totalizam cerca de 500 avaliações usando modelos de maturidade equivalentes. Neste período, foram realizadas no Brasil 11 avaliações no nível 5 do CMMI e sete no nível A do modelo MPS, totalizando 18 empresas com avaliações no nível mais alto de maturidade. Isto mostra que cresceu significativamente o uso de normas e modelos especificamente voltadas para a melhoria de processos de software no Brasil, tais como o MPS e o CMMI, e que está sendo revertida a situação constatada em 2003 [5].

#### **4.2. Processos que contribuem para a melhoria da qualidade de software**

As contribuições do programa MPS.BR para a melhoria dos processos de software estão documentadas nos diversos Guias MPS. A primeira versão do Guia Geral MPS foi publicada em 2004 e do Guia de Avaliação MPS em 2005 [2]. No ciclo atual do projeto do PBQP Software foram publicados: Guia Geral:2011, Guia de Aquisição:2011, Guias de Implementação:2011 – partes 1 a 11 e Guia de Avaliação:2011.

#### **4.3. Artigos publicados**

Em 2011, foram publicados os seguintes estudos e artigos (*papers*) sobre o programa MPS.BR e o modelo MPS:

- o estudo 'IMPS 2010: Desempenho das Empresas que Adotaram o Modelo MPS de 2008 a 2010' (Guilherme Horta Travassos e Marcos Kalinowski). – Campinas, SP: SOFTEX, 2011.
- diversos artigos sobre o modelo MPS foram publicados nos Anais do SBQS 2011/ 11º Simpósio Brasileiro de Qualidade de Software, realizado em Curitiba-PR de 06-10/06/2011;
- o artigo 'From Software Engineering Research to Brazilian Software Quality Improvement' (Marcos Kalinowski, Gleison Santos, Rafael Prikladnicki, Ana Regina Rocha, Kival Weber e José Antonio Antonioni) foi publicado nos Anais do CBSOft/SBES 2011/ 25º Simpósio Brasileiro de Engenharia de Software, realizado em São Paulo-SP de 26-30/09/2011 [8];
- diversos artigos sobre o modelo MPS foram publicados nos Anais do WAMPS 2011 – VII Workshop Anual do MPS, realizado em Campinas-SP de 24-28/10/2011.

#### **4.4. Recursos humanos capacitados**

Em 2011, foram capacitados no Brasil novos 364 profissionais através dos cursos oficiais MPS (C1 – Introdução ao MPS.BR, C2 – Implementação MPS, C3 – Avaliação MPS e C4 – Melhoria do Processo de Aquisição de Software), atingindo o total acumulado de 4.779 participantes de cursos oficiais MPS desde o início do programa MPS.BR. No ciclo 2011, foram ofertados cursos oficiais MPS tanto em português no Brasil quanto em espanhol na Colômbia, México e Peru no âmbito do projeto RELAIS (ver seção 4.6). Em 2011, havia 30 instrutores treinados e autorizados para ministrar estes cursos.

#### **4.5. Dissertações e/ou teses geradas**

Não dispomos destes dados uma vez que dissertações e/ou teses sobre o modelo MPS foram e continuam sendo geradas em Universidades de todas as regiões do país; mas pode-se afirmar que este número é significativo. Além disso, é importante destacar a importância da Academia tanto na Equipe Técnica do Modelo (ETM) quanto na disseminação do modelo MPS. Das 18 II credenciadas junto à SOFTEX (que fornecem consultoria de capacitação das empresas para adequação ao modelo de referência MR-MPS), 12 estão diretamente vinculadas a Universidades.

Adicionalmente, das 12 IA credenciadas junto à SOFTEX (que realizam as avaliações que permitem atribuir níveis de maturidade às organizações desenvolvedoras de software), 8 estão diretamente vinculadas a Universidades. Desta forma, a Academia tem um importante papel também no auxílio à Indústria, atuando como ponte entre o estado da arte e o estado da prática da engenharia de software [8].

#### **4.6. Parcerias ou programas de transferência de tecnologia efetuados**

O objetivo geral do projeto BID/FOMIN RELAIS (Rede Latino Americana da Indústria de Software) é promover a melhoria da competitividade da indústria de software na América Latina e Caribe (LAC). O objetivo específico é incorporar os modelos MPS do Brasil e MoProSoft do México nos processos de desenvolvimento de software das PME em países da LAC, que garantam a qualidade de seus produtos e serviços. O nome oficial do projeto RELAIS é 'Mejora de la calidad de los procesos de software elaborados por PyME en Latinoamérica y el Caribe (LAC), a través del empleo de modelos de desarrollo de software (BID/FOMIN RG-M1144)'. Mas, sobretudo, a RELAIS espera consolidar-se como uma 'rede de negócios'.

O projeto RELAIS está sendo executado em 36 meses, do 1º semestre de 2010 ao 1º semestre de 2013, no Brasil (coordenador nacional: SOFTEX), Colômbia (coordenador nacional: ESI Center Sinertic), México (coordenador nacional: CANACINTRA) e Peru (coordenador nacional e executor regional: CCL – Câmara de Comércio de Lima).

Sob coordenação da SOFTEX, foi realizada em 2011 por equipes técnicas do Brasil e México a atividade RELAIS 'Mapeamento e Sistema de Equivalências entre os Modelos MPS e MoProSoft'.

Um processo de "sensibilização" para adoção dos modelos de maturidade e melhoria de processo de software foi realizado em 2011 através de seminários reunindo empresários, consultores e representantes de governo e da academia, em oito cidades de quatro países: Guadalajara e México DF, no México; Arequipa e Lima, no Peru; Campinas e Rio de Janeiro, no Brasil; Medellin e Bogotá, na Colômbia.

Em 2011, iniciou-se a formação de massa crítica de consultores de implementação do modelo MPS na Colômbia, Peru e México, por meio da capacitação presencial de profissionais desses países nos cursos oficiais C1- Introdução ao MPS.BR e C2- Implementação MPS, em espanhol, com posterior aplicação nestes países da prova P2 - Implementação MPS.

Representantes de governo da Colômbia, México e Peru, ligados às atividades de compras públicas, foram capacitados em 2011 no processo de Aquisição de software do modelo MPS, por meio da participação no curso oficial C4 - Melhoria do Processo de Aquisição de Software, em Campinas-SP.

Em 2011, foi criada e testada uma plataforma na Web para operacionalizar "Comunidades de Prática" de interesse da RELAIS. A primeira implantada foi a "CoP/MPS - Comunidade de Prática do MPS" que já está em uso com itens de conhecimento em cinco áreas: Aquisição MPS, Avaliação MPS, Gestão do Programa MPS.BR, Implementação MPS e Organização de Grupos de Empresas MPS. Futuramente novas comunidades de prática poderão ser criadas sobre essa plataforma, como a do modelo mexicano MoProSoft e da própria RELAIS, por exemplo.

No âmbito da RELAIS, sob coordenação da SOFTEX, os cursos oficiais C1 - Introdução ao MPS e C2 - Implementação MPS foram estruturados em 2011 também na modalidade EAD (ensino a distância), para capacitação de profissionais tanto em português no Brasil quanto em espanhol na Colômbia, México e Peru, dentre outros países da América Latina e Caribe (LAC).

O viés de negócios da RELAIS foi iniciado em 2011 por meio da inserção da ferramenta de auto-diagnóstico do PROIMPE (SOFTEX / Área de Mercado) às ações do projeto RELAIS. Foram realizados workshops na Colômbia, nas cidades de Armênia e Bogotá, com profissionais e empresários de diferentes setores econômicos.

Em 2011, a SOFTEX também participou ativamente das discussões iniciais para a institucionalização da rede RELAIS, numa atividade denominada RELAIS 2013, com o objetivo de dar continuidade às ações do projeto RELAIS, de 2013 em diante, após o fim do convênio com o BID/FOMIN.

#### **4.7. Sustentabilidade do Programa**

Por último, mas não menos importante, na execução do MPS.BR a verificação das “condições de sustentabilidade do programa” é analisada segundo três aspectos (isto é feito tanto rotineiramente pela UEP e também pela Diretoria da SOFTEX quanto semestralmente nas Reuniões do CGP – Conselho de Gestão do Programa):

- Sustentabilidade Institucional
- Sustentabilidade Operacional
- Sustentabilidade Financeira

#### **5. Aplicabilidade dos Resultados**

A seguir, descreve-se a relevância, a abrangência e o impacto tanto dos resultados obtidos no ciclo 2011 do projeto PBQP Software quanto dos resultados acumulados no programa MPS.BR desde o início.

##### **5.1. Relevância**

O programa MPS.BR descrito neste projeto atende às políticas públicas, tais como a disposta no Programa de Estímulo ao Setor de Software e Serviços: “Promover melhores práticas – Estimular a adoção das melhores práticas de engenharia de software pelas empresas do setor tanto em seu processo de desenvolvimento quanto no processo de aquisição de software e serviços correlatos, em conformidade com o estado da arte e as normas internacionais de qualidade aplicáveis”.

##### **5.2. Abrangência**

O programa MPS.BR e o modelo MPS têm abrangência nacional, com presença ativa em todas as regiões do Brasil. Em 2011, o modelo MPS começou a ter abrangência internacional através do projeto RELAIS, com foco na Colômbia, México e Peru.

##### **5.3. Impacto**

Em 2011, por ocasião da conclusão da etapa de consolidação do MPS.BR, o impacto deste programa mobilizador em C,T&I atingia: i) milhares de pessoas capacitadas no modelo MPS; ii) centenas de colaboradores da ‘linha de frente’ [6], tanto instrutores MPS quanto pessoal das II, IA, IOGE e ICA; iii) centenas de organizações que adotaram o modelo MPS.

Pesquisas iMPS (Resultados de Desempenho de Organizações que Adotaram o Modelo MPS), realizadas anualmente pelo Grupo de Engenharia de Software Experimental da COPPE/UFRJ e publicadas pela SOFTEX, buscam evidenciar os benefícios para a Indústria. Dentre os resultados da pesquisa iMPS2011 [9], que contou com questionários recebidos de 133 organizações que adotaram o modelo MPS, destaca-se o seguinte: i) a satisfação das empresas com o modelo foi novamente notória, com aproximadamente 97% das empresas se dizendo totalmente ou parcialmente satisfeitas com o modelo MPS; ii) a caracterização das empresas em 2011 permitiu observar correlações positivas entre a maturidade das empresas no modelo MPS e o número de projetos (tanto no país quanto no exterior); iii) na análise de variação de desempenho de 2008 a 2011 identificou-se que empresas que se mantêm persistentes na utilização das práticas de engenharia de software representadas pelos níveis de maturidade do modelo MPS possuem maior número de clientes, desenvolvem maior número de projetos, possuem maior número de funcionários, lidam com projetos de maior tamanho e apresentam menores erros em suas estimativas de prazo, apesar de um ligeiro aumento provocado no tempo médio gasto em seus projetos. Destes resultados da pesquisa iMPS2011 resultaram notícias tais como “SOFTEX: Melhoria do software aumenta faturamento, clientes e projetos” [10].

#### **6. Características Inovadoras**

Nestes oito anos, consolidou-se um programa inovador e um novo modelo de maturidade, inéditos no país, com reconhecimento internacional crescente.

## 7. Conclusão e Perspectivas Futuras

Este artigo descreveu os principais resultados do programa MPS.BR na etapa 2008-2011, com foco em 2001, conforme disposto no Projeto 2.01 - PBQP Software 2011: Consolidação do MPS.BR (2008-2011). O artigo mostrou que o MPS.BR é um programa bem sucedido tanto na melhoria dos processos de software [1], seja em grandes organizações seja em PME, quanto no aumento da competitividade da Indústria de Software no Brasil [9].

Por se tratar de um programa de longo prazo, será submetido um novo projeto ao PBQP Software – Ciclo 2012 intitulado “Expansão do MPS.BR (2012-2015)”, com quatro objetivos:

- garantir a sustentabilidade institucional, operacional e financeira do programa MPS.BR;
- continuar aprimorando o modelo MPS relativo à Melhoria de Processo de Software (MP Sw) e criar o modelo MPS de Melhoria de Processo de Serviços de TI (MP Sv);
- expandir o uso do modelo MPS no Brasil, tanto quantitativamente (mais 300 avaliações MPS de 2012 a 2015) quanto qualitativamente (com as organizações que adotaram o modelo MPS evoluindo continuamente seus níveis de maturidade);
- aumentar a presença internacional do modelo MPS, com foco inicial nos países da América Latina e Caribe (LAC).

## 8. Referências Bibliográficas

[1] Montoni, M.A., Rocha, A.R., Weber, K.C. MPS.BR: A Successful Program for Software Process Improvement in Brazil. *Softw. Process Improve. Pract.* 2009; 14: 289-300. Published online 23 June 2009 in Wiley InterScience ([www.interscience.wiley.com](http://www.interscience.wiley.com)) DOI: 10.1002/spip. 428.

[2] 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: PUCRS, 2005. Prêmio de melhor artigo técnico do SBQS 2005.

[3] Weber, K. MPS.BR – Melhoria de Processo do Software Brasileiro (Dez2003-Dez2007). Projeto 2.25 do PBQP Software – Ciclo 2006. Brasília: MCT/SEPIN, 2007. Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software (1º lugar) em 2007.

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

[5] 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.

[6] Santos, G., Weber, K.C. Lições Aprendidas na Gestão do Programa MPS.BR. In: *MPS.BR : Lições Aprendidas/organizadores: Rocha, A.R.C., Weber, K.C.* Campinas: SOFTEX, 2008.

[7] Rocha, A.R., Rubinstein, A., Magalhães, A.L., Katsurayama, A.E., Duque A., Barbieri Palestino, C., Souza, C., Cerdeiral, C., Teixeira, L., Barros, L. Serranegra Paiva, N. Joint CMMI Level 3 and MPS Level C Appraisal: Lessons Learned and Recommendations. SEIR – Software Engineering Institute Repository <[www.sei.cmu.edu/seir](http://www.sei.cmu.edu/seir)>. Belo Horizonte: Synos Technologies, 2009.

[8] Kalinowski, M., Santos, G., Prikładnicki, R., Rocha, A.R., Weber, K., Antonioni, J.A. From Software Engineering Research to Brazilian Software Quality Improvement. In: *Anais do CBSOft/SBES 2011 – II Congresso Brasileiro de*

Software: Teoria e Prática, trilha 'SBES is 25' / 25º Simpósio Brasileiro de Engenharia de Software. São Paulo: Universidade Presbiteriana Mackenzie, 2011.

[9] Travassos, G.H., Kalinowski, M. iMPS 2011: Resultados de Desempenho das Empresas que Adotaram o Modelo MPS de 2008 a 2011. Campinas: SOFTEX, 2012.

[10] Convergência Digital. SOFTEX: Melhoria do software aumenta faturamento, clientes e projetos – 02/04/2012 – 11h15 < [www.convergenciadigital.uol.com.br](http://www.convergenciadigital.uol.com.br) >.

# Proposta de Modelo de Qualidade de Requisitos de Segurança para Aplicativos de Software na Web

Regina M Thienne Colombo, Amândio Balcão Filho, Ana Cervigni Guerra, Marcelo Schneck de Paula Pessôa.

Centro de Tecnologia da Informação – CTI Renato Archer  
DSSI – Divisão de Segurança de Sistemas de Informação  
Rod. Dom Pedro I, km 143,6 – 13.069-901 – Campinas – SP – Brazil

regina.thienne@cti.gov.br, amandio.balcao@cti.gov.br, ana.guerra@cti.gov.br, mpessoa@usp.br

**Resumo.** As aplicações de software na web se tornaram o alvo preferencial do ataque de curiosos, hackers e até criminosos, por isso é preciso avaliar e assegurar que as aplicações estejam preparadas para tratar estas ameaças e que além dos técnicos especialistas em segurança, a área de gestão e de negócios de TI esteja consciente e decidida à tomada de decisão. O objetivo é estabelecer um modelo de qualidade de requisitos de segurança e um conjunto de medições para compor um processo de avaliação sistemático para os aspectos da segurança do software e que possibilite mostrar para a gerência, o nível de segurança da aplicação de software.

## 1. Introdução

O conhecimento e a experiência sobre o nível de segurança de software precisam ser disponibilizados para os técnicos e gestores de não-segurança de uma maneira objetiva, clara e sem complexidades. Esse conhecimento precisa ser explicitado em forma de documentos tipo relatórios, gráficos ou de outra forma de fácil acesso.

Os especialistas em segurança de software têm experiência e conhecimento tácito algumas vezes sem ter consciência dele. Esse conhecimento e experiência normalmente não são explicitados, mas esses especialistas fazem uso disso o tempo todo. Existem situações onde o conhecimento tácito é necessário tornar-se conhecimento explícito. Na área de segurança, é necessário o desenvolvimento de construtos de apoio à questão de segurança em aplicativos de software [Houmb, 2010]. O levantamento de requisitos de segurança, a definição de métricas e a metodologia de avaliação da segurança de aplicativos de software tenta englobar o conhecimento tácito dos especialistas em segurança juntamente com o conhecimento explícito encontrado na literatura da área.

## 2. Objetivos e Justificativa

Para todo software aplicativo é importante que exista seu modelo de qualidade, para nortear os requisitos de qualidade no desenvolvimento do produto e avaliar seu desempenho segundo estas características de qualidade, juntamente com todas as características específicas da aplicação e do domínio ao qual pertence [Colombo; Guerra, 2009]. Os requisitos críticos tais como aspectos da segurança, os econômicos ou do ambiente, os regulamentos e as leis aplicáveis devem ser integrados no modelo de qualidade. O domínio da aplicação de um software é tão importante como a descrição geral de sua finalidade e seus requisitos de qualidade. Esta proposta tem como objetivo geral propor um modelo de qualidade de segurança, que engloba a especificação de requisitos de segurança de aplicações web e um processo de avaliação que será utilizado na fase de execução e uso da aplicação, o objetivo específico é conhecer e diminuir os riscos de vulnerabilidades em aplicações web, que são desenvolvidos para atender à gestão da instituição de forma integrada, trazendo maior transparência, rapidez e confiabilidade para as informações corporativas.

## 3. Metodologia de execução

No trabalho foi utilizado o método de pesquisa Estudo de Caso. Adicionalmente foi realizada a pesquisa bibliográfica e também, utilizada a pesquisa tipo *survey* do Instituto Ponemon [Ponemon, 2010]. Esta proposta parte de uma questão central de indagação: Como analisar as ameaças e gerir os riscos da segurança de aplicação web?

Essa questão trás como proposição a especificação e avaliação de propriedades e requisitos de segurança que possam ser medidos para conhecer o nível de segurança de aplicativos Web. Assim, a Estruturação da pesquisa está embasada em:

- Um Modelo de Requisitos de Segurança, que dará a estrutura conceitual para medições de Segurança em aplicativos Web;
- O Modelo é direcionado para propriedades de segurança e para vulnerabilidades enquanto as ameaças são centradas em requisitos;
- O Modelo utiliza a abordagem de decomposição de requisitos para obter os componentes básicos mensuráveis (BMCs);
- O Modelo utiliza métodos formais consagrados (AHP) [Saaty, 2003] para possibilitar a ajuda na tomada de decisão;
- A avaliação de requisitos de Segurança será utilizada na fase de operação e uso da aplicação.

Com relação à pesquisa científica nos temas relacionados a este contexto, vê-se na literatura determinados conceitos e teorias que podem ser analisadas e aplicadas. Um objetivo deste trabalho foi estudar como se pode ampliar a avaliação da segurança visando uma melhor obtenção de resultados, seja para especialistas, para a gestão ou para os usuários. Por meio de um referencial teórico específico para o contexto procura-se preencher a lacuna que existe entre o estágio atual e o desejável deste cenário, visando obter mecanismos que permitam melhorar a qualidade da aplicação em relação ao atributo segurança. Também, por meio de um referencial de experiências práticas e iniciativas, são considerados os diversos objetivos relacionados à questão da segurança de aplicações web. Diante deste cenário, o objeto de estudo busca pesquisar a proposição de que o desenvolvimento de uma metodologia sistemática englobando um conjunto de conceitos de segurança reconhecidos internacionalmente e de práticas no mercado, possam se materializar para uma especificação de requisitos de segurança e uma metodologia de avaliação da segurança de aplicações web [Colombo, 2010].

### **3.1. Sobre o Produto de software avaliado**

O SW-WEB é o ambiente computacional que permite a interação dos membros de uma equipe dedicada à produção de um determinado resultado, por meio do qual se pode registrar e trocar informações assim como ter acesso a ferramentas computacionais. O ambiente de trabalho pode ser criado e configurado para cada tipo de resultado produzido pela instituição. O sistema é voltado para a gestão dos resultados de instituições de pesquisa de órgãos federais. Foi realizada uma primeira avaliação, com abordagem exploratória para conhecer o comportamento da aplicação web perante alguns requisitos de segurança. Também foi planejado um conjunto de perguntas para a equipe responsável pelo aplicativo Sw-Web com o propósito de fazer uma análise ampla dos desdobramentos das vulnerabilidades encontradas no sistema. As perguntas são:

- A Vulnerabilidade é nova ou já conhecida por vocês?
- A Gravidade da vulnerabilidade é (Pequena, Média, Alta)?
- A Fonte/Causa do problema é de (Rede, Servidor, Aplicação, Arquitetura)?
- Tem outros desdobramentos de correções no sistema?
- Quem é o Responsável/Função da equipe por corrigir?
- Qual o Prazo estimado para corrigir?
- Que problemas podem gerar ao usuário do sistema?
- Que ações devem ser geradas pela equipe de desenvolvimento?
- Que ações devem gerar aos responsáveis pelo sistema?
- Na especificação de requisitos existia algum requisito para evitar esta vulnerabilidade?

Este estudo de caso serviu para uma simulação e experimento para desenvolver um modelo de qualidade de segurança de aplicação web, um conjunto de medições para avaliar a segurança e conhecer a visão gerencial responsável pela aplicação perante o resultado da avaliação realizada.

### 3.2. Normas, Padrões e Modelos considerados

Os conceitos, modelos, normas e trabalhos relacionados são apresentados a seguir. Com relação às normas, existem alguns órgãos reguladores internacionais em conjunto com a ISO (*International Organization for Standardization*) e nacionais como a ABNT (Associação Brasileira de Normas Técnicas) que estabelecem normas e guias para definir e divulgar as boas práticas de um domínio, que possuam abrangência global podendo ser utilizadas em qualquer parte do mundo nos mesmos moldes e características. À medida que a tecnologia evolui, novas práticas serão incorporadas num processo de melhoria contínua tanto das normas como também das metodologias em uso. Estas exigências e diretrizes em termos de definição e avaliação da qualidade de software e em termos de segurança de software devem ser incorporadas aos processos de desenvolvimento de software para a geração de aplicações de software seguro e estas serem melhoradas continuamente.

### 3.3. A Série ISO/IEC 27000

O Padrão Britânico (*British Standard*) 7799 (BS7799) teve sua origem de um código de prática do governo do Reino Unido (*Department of Trade and Industry - DTI*) de 1993, depois publicado como padrão em 1995 pelo *British Standards Institution* (BSI) e revisado em 1999. Quando foi publicado, inicialmente como um padrão internacional pela ISO em dezembro de 2000, BS7799 parte 1 (BS7799-1) se tornou ISO 17799 [NBR ISO/ IEC 17799, 2005]. Em outubro de 2005, British Standard BS 7799 parte 2 (BS7799-2) foi adotada pela ISO e re-identificado, iniciando a nova série 27000 de padrões internacionais para segurança da informação lançada como norma ISO/IEC 27001 em 2005. No Brasil, a ABNT publica as normas localizadas ABNT NBR ISO/IEC 27002:2005 [NBR ISO/ IEC 17799 -2005] e ABNT NBR ISO/IEC 27001:2006 [NBR ISO/IEC 27000, 2006].

A NBR ISO/IEC 27001 é um padrão para sistema de gestão da segurança da informação (ISMS - *Information Security Management System*). Seu objetivo é ser usado em conjunto com ISO/IEC 17799, o código de práticas para gerência da segurança da informação, o qual lista os objetivos do controle de segurança e recomenda um conjunto de especificações de controles de segurança.

### 3.4. O Modelo *Common Criteria* – Série ISO/IEC 15408

Os Critérios Comuns [Common Criteria, 2011] têm sua origem em 1990 e surgem como resultado da harmonização dos critérios sobre segurança de produtos software já utilizados por diferentes países com o objetivo de que o resultado do processo de avaliação pudesse ser aceitos em múltiplos países. Permitem comparar os resultados entre avaliações de produtos independentes. Para isso, proporcionam um conjunto comum de requisitos funcionais para os produtos de TI. Estes produtos podem ser hardware, software ou firmware. O processo de avaliação estabelece um nível de confiança no grau no qual o produto TI satisfaz a funcionalidade de segurança destes produtos e tem superado as medidas de avaliação aplicadas. São úteis como guia para o desenvolvimento, avaliação ou aquisição de produtos TI que incluam alguma função de segurança.

A série ISO/IEC 15408 – *Information technology – Security techniques – Evaluation criteria for IT security*, surgiu do grupo responsável pelo *Common Criteria* é voltado para a segurança lógica das aplicações e para o desenvolvimento de aplicações seguras.

### 3.5 O Modelo OWASP-ASVS

O projeto OWASP - *Open Web Application Security Project* é formado por uma comunidade aberta que tem o objetivo de possibilitar que organizações desenvolvam, comprem e mantenham aplicações que possam ser confiáveis ([www.owasp.org](http://www.owasp.org)).

Nesta comunidade muitos projetos específicos da área são desenvolvidos sendo que o projeto OWASP-ASVS – *Open Web Application Security Project – Application Security Verification* [OWASP, 2009] que tem o objetivo de definir um conjunto de requisitos de segurança e uma estrutura em níveis de verificação da segurança. O projeto define quatro níveis de aplicação e de verificação de segurança para aplicações web. No nível de aplicação de segurança o ponto central está na análise dos componentes que compõem a camada de aplicação do *Open Systems*

*Interconnection Reference Model (OSI Model)*, dando menos prioridade ao sistema operacional subjacente ou redes conectadas. Cada nível descrito nesse documento inclui um conjunto de requisitos para a verificação da eficácia dos controles de segurança que protegem as aplicações web.

Os requisitos de segurança foram desenvolvidos com preocupação nos seguintes objetivos:

- Usar como uma métrica - Fornecer aos desenvolvedores de aplicativos e proprietários de aplicativos critérios que permitam avaliar o grau de confiança que pode ser colocado em suas aplicações Web,
- Usar como orientação - Fornecer orientações para os desenvolvedores de controle de segurança sobre o que construir em controles de segurança, a fim de satisfazer os requisitos de segurança do aplicativo, e
- Usar durante a aquisição - Fornecer uma base para especificar a verificação de segurança de aplicativos requisitados nos contratos.

Os requisitos de segurança foram projetados para atender os objetivos acima, assegurando a validação de como controles de segurança são concebidos, implementados e utilizados por um aplicativo.

### 3.6 O Modelo ISO/IEC 25010

O modelo de qualidade definido na ISO/IEC 25010 [17] é estruturado de forma hierárquica, ele categoriza qualidade de software em características que são subdivididas em subcaracterísticas e atributos de qualidade como mostra a Figura 4.1. Os atributos são entidades que podem ser medidas de forma direta ou indiretamente.

A ISO/IEC 25010 [26] define um modelo de qualidade de produto de software composto de oito características (que são subdivididas em subcaracterísticas) relacionadas a atributos estáticos do software e atributos dinâmicos do sistema de computador. este modelo define a característica de qualidade Segurança, que é o foco desta pesquisa com as seguintes subcaracterísticas: Confidencialidade, Integridade, Não-repúdio, Responsabilidade e Autenticação.

A Tabela 1 apresenta resumidamente as propriedades de segurança de software apresentadas nos modelos e padrões citados anteriormente.

Tabela1- Correlação dos requisitos de segurança e as referências

PROPRIEDADES E REQUISITOS	ISO/IEC 25010	ISO/IEC 27001	ISO/IEC 15408	OWASP ASVS
Confidencialidade	X	X		
Integridade	X	X		
Não-repúdio	X	X		
Responsabilidade	X	X		
Autenticação	X	X	X	X
Autorização				
Disponibilidade		X		
Confiabilidade		X		

#### 4. Resultados obtidos no ciclo de 2011

- Processo genérico de avaliação de qualidade da segurança de software formalizado.
- Artigo publicado “*Security Evaluation Process for Web Applications*” apresentado no evento internacional – “23rd International Conference on Software & Systems Engineering and their Applications” ICSSEA 2011, 29/11 a 01/12/2011, em Paris – França [Colombo, 2011].
- Exame de Qualificação realizado e aprovado em 06/12/2011, na Universidade de São Paulo, Departamento de Engenharia da Produção.
- Recursos humanos capacitados (bolsista, mestres, doutores) os participantes do projeto em continuo trabalho de discussão e homogeneização das ideias.
- Futuras contribuições acadêmicas e de mercado desta proposta:
  - Definição de um processo sistemático para avaliar segurança
  - Orientação de como obter medidas por meio de um Modelo de segurança
  - Orientação de como extrair métricas para uso da área gerencial
  - Conhecimento e disponibilização do nível de segurança da aplicação web

#### 4.1 Considerações finais

Na prática, a construção de grandes e complexos sistemas de software desprovidos de erros, e vulnerabilidades de segurança, continua a ser uma tarefa difícil. Em primeiro lugar, especificações com os pressupostos explícitos, podem mudar ao longo do projeto, então algo que não era um erro pode se tornar um erro mais tarde. Segundo, especificações formais são raramente escritas na prática. Terceiro, ferramentas de verificação formal usadas na prática para encontrar e corrigir erros, incluindo vulnerabilidades específicas, como estouros de *buffer*, geralmente são ditas sólidas e robustas no comércio. Em quarto lugar, não se pode prever as vulnerabilidades do futuro, ou seja, os erros presentes no software atualmente não podem ter previsão de como serão atacados no futuro.

Desta forma, perante as inúmeras ações para assegurar a segurança de software, esta proposta vem cobrir uma lacuna no contexto de estabelecer uma metodologia sistematizada para avaliar a segurança de aplicações web.

#### 5. Aplicabilidade dos Resultados

Relevância - O relatório da Symantec afirma que em 2008 foram encontrados 1.656.227 novos artefatos maliciosos (malware) o que corresponde a um aumento de 256% em relação aos encontrados no ano de 2007 [Blackbird, 2009]. Portando, devido ao aumento dessas atividades no Brasil e no mundo e ao aumento de potenciais novos alvos, existe uma motivação e preocupação com a segurança de software para traçar um panorama nacional que auxilie os usuários, empresas e instituições a se defender destes artefatos.

Abrangência - Atualmente os sistemas computacionais são complexos envolvendo a interatividade com os usuários, uma interface amigável pode minimizar dificuldades com usuários leigos no domínio, na aplicação e também nos sistemas com a utilização da rede para a comunicação entre os usuários e servidores dos atuais sistemas de informação. Com relação a estes tipos de sistemas, cresce a preocupação com a confiança na segurança da informação para os usuários e especialistas na área.

Impacto dos resultados obtidos - A gestão da segurança a informação é um assunto sério, é estratégico e deve estar entre as preocupações centrais da alta direção de instituições. Regulamentos, normas de conduta e diversos mecanismos de defesa têm sido desenvolvidos nos últimos anos, num cenário em que, cada vez mais, as empresas

dependem de seus sistemas de informação e do comportamento da sua força de trabalho, para assegurar que seus dados estejam protegidos.

Uma parceria a ser citada foi o aluno de especialização CÉLIO CARUSO GOMES do Instituto Tecnológico de Aeronáutica – ITA, Praça Marechal Eduardo Gomes, 50 - Vila das Acácias - São José dos Campos – SP. Esse aluno esteve envolvido no desenvolvimento dos constructos matemáticos para a priorização das características de qualidade de segurança para diferentes categorias de software. [celiocaruso@gmail.com]

## 6. Características Inovadoras

No Brasil e no mundo ainda as iniciativas acadêmicas na área que abrange Qualidade de software com enfoque em Segurança de aplicativos de software são embrionárias e necessitam de maior atenção dos pesquisadores.

## 7. Conclusão e Perspectivas Futuras

Esse processo faz a decomposição de sistemas complexos em sistemas mais simples e menores possibilitando dessa forma estimar propriedades que irão ajudar no seu conhecimento e medição das características de segurança de sistemas de software. O resultado desse processo garante subsídios para gestores priorizar e gerenciar recursos na área de segurança da informação.

Esta proposta de pesquisa já foi avaliada e aprovada. Os próximos passos da proposta serão detalhados e desenvolvidos. Algumas atividades ainda serão realizadas, tais como:

- preparação de um artigo para um periódico internacional;
- preparação da lista de verificação a ser utilizada no processo de avaliação;
- outros estudos de caso estão em definição;
- aplicação dos cálculos no software Super Decisions para o modelo de segurança;
- utilizar a metodologia AHP com ratings para o atributo de segurança - Autenticação.

## 8. Referências Bibliográficas

- Houmb, S.H. at all Eliciting security requirements and tracing them to design: an integration of Common Criteria, heuristics, and UMLsec In Requirements Eng. 15:63-93 DOI 10.1007/s00766-009-0093-9, Springer 2010
- Guerra, A. Cervigni e Colombo, R. M. Thienne (2009) “Tecnologia da Informação: Qualidade de Produto de Software”, Editado por PBQP Software.
- Ponemon Institute, State of Web Application Security, by Imperva & WhiteHat Security, 2010
- Saaty, T.L., et al, The allocation of intangible resources: the analytic hierarchy process and linear programming, Socio-Economic Planning Sciences Volume 37, Issue 3, September 2003, Pages 169–184.
- Common Criteria- <http://www.commoncriteriaportal.org/cc/>, acesso em outubro de 2011.
- OWASP – The Open Web Application Security Project, ASVS – Application Security Verification Standard 2009, em [www.owasp.org](http://www.owasp.org).
- NBR ISO/ IEC 17799 -2005. Tecnologia de Informação – Técnicas de Segurança – Código de prática para a gestão da segurança da informação.
- NBR ISO/IEC 27000 Tecnologia da informação — Técnicas de segurança — Sistemas de gestão de segurança da informação — Requisitos, 2006.
- ISO/IEC 25010 - Software Engineering – System and Software product Quality Requirements and Evaluation (SQuaRE), Quality Model, 2010.
- ISO/IEC 15408-1 - Information technology — Security techniques — Evaluation criteria for IT security — Part 1: Introduction and general model, International Standards Organization, Geneva, 2009.

- Blackbird Joseph, Mo King Low, Teo Adams, David McKinney, Stephen Entwisle, Marika Pauls Laucht, Candid Wueest, Paul Wood, Dan Bleaken, Greg Ahmad, Darren Kemp, and Ashif Samnani. Symantec Global Internet Security Threat Report. Trends for 2008 Volume XIV, Published April 2009.
- Centro de Tecnologia da Informação Renato Archer - CTI/MCT - Tecnologia de Avaliação de Qualidade de Produto de Software, "MEDE-PROS - Método de Avaliação da Qualidade de Produto de Software, versão 2006",
- Colombo R. M. T., Guerra A.C., Balcao A. F., Pessoa M. P, Duarte,L.O. - Avaliação da Qualidade de produtos de Software com enfoque na segurança da Informação - Conferência IADIS Ibero-Americana - Consórcio Doutoral, pag 525, Algarve, Portugal 2010.
- Colombo R. M. T., Guerra A.C., Balcao A. F., Pessoa M. P. "Security Evaluation Process for Web Applications" – "23rd International Conference on Software & Systems Engineering and their Applications" ICSSEA 2011, 29/11 a 01/12/2011, Paris – França.

# Metodologia para Execução de uma Avaliação Integrada

André Villas Boas, Raquel Moreda P. Gutierrez, Silvia R. Lopes, Valmir Martins, Valquíria M. Fagundes

Assessoria da Qualidade – Fundação CPqD Centro de Pesquisa e Desenvolvimento em Telecomunicações  
Rod. Campinas - Mogi-Mirim, Km 118,5 – Campinas – SP – Brasil

{villas, moreda, slopes, vmartins, fagundes}@cpqd.com.br **Abstract.** This article describes a project conducted in CPqD in order to obtain a methodology for carrying out an assessment in an integrated management system, which is compliant to several standards and models.

**Resumo.** Este artigo descreve o projeto realizado no CPqD visando a obtenção de uma metodologia para execução de uma avaliação integrada em um sistema de gestão, o qual é aderente a várias normas e modelos.

## 1. Introdução

O Sistema de Gestão do CPqD abrange processos para desenvolvimento de software, desenvolvimento de produtos, serviços de laboratório, serviços de calibração, serviços de consultoria, certificação de produtos, entre outros. Ele é certificado e atende as normas ISO 9001, ISO 14001, ISO 17025 e o modelo CMMI.

Para que essas certificações sejam mantidas, são realizadas semestralmente, auditorias de manutenção para os serviços aderentes às normas ISO e avaliações periódicas nos projetos de desenvolvimento de software que seguem o modelo CMMI. Estas avaliações e auditorias anualmente envolvem 25 profissionais, tendo um gasto de 2600 horas. Cada auditoria tem uma média de 70 sessões e há por volta de 25 projetos de software por mês.

O desenvolvimento de uma metodologia que integre estas avaliações e auditorias se faz necessário.

## 2. Objetivos e justificativa

A criação de uma metodologia para realização de uma avaliação integrada que seja capaz de atender aos requisitos das Normas ISO e ao modelo CMMI, se faz necessária para minimizar o impacto na rotina de trabalho das áreas avaliadas, melhorar a produtividade na realização desse tipo de serviço e diminuir o custo associado.

Outros objetivos são a melhoria contínua da metodologia, tomando-se como base os resultados obtidos na execução das avaliações; treinamento dos avaliadores e parametrização da ferramenta de avaliação utilizada no CPqD, com os requisitos do modelo CMMI e normas ISO.

## 3. Metodologia de Execução

Iniciou-se com uma análise do modelo CMMI e normas ISO aplicáveis e foram pesquisados artigos e bibliografia relacionados para apoio na definição da metodologia.

Aplicação do PDCA, sendo:

### 1o. ciclo:

Definição de uma metodologia inicial com base na experiência de auditores ISO e CMMI;

Realização de uma avaliação no sistema de gestão baseada na metodologia;

Coleta de lições aprendidas, dados de planejamento e realização de ajustes na metodologia.

### 2o. ciclo:

Planejamento de avaliação com base nos dados do ciclo anterior;

Coleta de lições aprendidas, dados de planejamento e realização de ajustes na metodologia.

### **3o. ciclo:**

Planejamento de avaliação com base nos dados do ciclo anterior;

Coleta de lições aprendidas, dados de planejamento e realização de ajustes na metodologia;

Validação independente da metodologia realizada por especialista na área, visando verificar sua aderência ao CMMI e normas ISO;

### **4o. ciclo:**

Planejamento de avaliação com base nos dados do ciclo anterior;

Validação da metodologia para garantir aderência ao modelo CMMI e Normas ISO em avaliação externa;

Realização de ajustes finais na metodologia para aplicação.

## **4. Resultados obtidos**

Os seguintes resultados foram obtidos:

- Pedido de patente: iniciou-se o estudo para o pedido de patente da metodologia criada;
- Desenvolvimento de uma metodologia: criada uma metodologia para realização de avaliação integrada do sistema de gestão do CPqD, capaz de atender aos requisitos das Normas ISO e ao modelo CMMI. A metodologia possibilitou minimizar o impacto na rotina de trabalho das áreas avaliadas, melhorou a produtividade e objetividade na realização desse tipo de serviço, possibilitou melhor visualização dos problemas do sistema de gestão e reduziu o custo associado em 15%;
- Capacitação de recursos humanos: qualificação de pessoas na metodologia criada;
- Outros resultados: parametrização de uma Ferramenta de Avaliação que serviu de apoio nas avaliações e contribuiu para aumento da produtividade;

## **5. Aplicabilidade dos Resultados**

Criação de facilidades para se alcançar padrões internacionais de qualidade tanto para foco ISO como modelo CMMI, em organizações que executam seus serviços frente a esses modelos, principalmente organizações que possuem outros serviços além de desenvolvimento de software.

Poderá ter impacto em empresas que possuem relacionamento com o CPqD, através da disseminação dos resultados alcançados e influência junto a entidades de apoio.

## **6. Características Inovadoras**

Definiu-se uma metodologia que trata os processos de avaliação de manutenção das normas ISO 9001, ISO 14001, ISO 17025 e modelo CMMI, para uma organização que além de desenvolvimento de software possui outros tipos de serviços. Respeitaram-se as características próprias das normas e modelo, eliminando-se as sobreposições, mantendo-se as individualidades e executando-se a avaliação de forma integrada, ganhando-se maior produtividade e objetividade.

Com a aplicação da metodologia é possível visualizar melhor quais os principais problemas da organização, direcionando de maneira eficiente o esforço para tratamento dos mesmos.

## **7. Conclusão e perspectivas futuras**

A metodologia desenvolvida pode ser aplicada em qualquer empresa cujo sistema de gestão possui processos relativamente estáveis e bem institucionalizados; a disseminação desta metodologia poderá beneficiar estas empresas, que poderão obter maior produtividade e objetividade na avaliação de seus sistemas de gestão.

Espera-se também a obtenção de um registro de patente, havendo já se iniciado o estudo para isto.

## **8. Referências bibliográficas**

Góes, Sergio (2010) “Ação Corretiva e Ação Preventiva”, White Paper.

NBR ISO 9001:2008 – Sistema de Gestão da Qualidade.

NBR ISO 14001:2004 – Sistema de Gestão Ambiental.

NBR ISO 17025:2005 – Requerimentos Gerais para Laboratórios e Ensaio e Calibração

NBR ISO 19011:2002 – Diretrizes para Auditoria de Sistema de Gestão da Qualidade e/ou ambiental.

Pittsburg, PA: Software Engeneering Institute “Capability Maturity Model Integration for Development – CMMI-DEV”, versão 1.2

Schaaf, Kevin (2011) “Planning a Combined CMMI and a AS91000 Appraisal”

# Ontologias e Mapas Conceituais na Transferência de Conhecimento sobre Processo de Software

José Augusto Fabri<sup>1</sup>, André Luís dos Santos Domingues<sup>1</sup>, Alexandre L´Erário<sup>1</sup>, Alessandro Silveira Duarte<sup>1</sup>, Elias Canhadas Genvigir Rodrigo Henrique Cunha Palácios<sup>1</sup>

<sup>1</sup>Coordenadoria de Engenharia da Computação e de Análise e Desenvolvimento de Sistemas – Universidade Tecnológica Federal do Paraná (UTFPR)  
Cornélio Procópio – PR – Brazil

{fabri, andddomingues, alerario, duarte, elias, rodrigo palacios}@utfpr.edu.br

**Abstract.** This paper shows how knowledge management and conceptual maps can be used in the process replication across a production software organization and its affiliates. The model of knowledge management and analysis of the effectiveness are presented in a case study.

**Resumo.** Este artigo apresenta como a gestão de conhecimento e mapas conceituais podem ser utilizados na replicação do processo entre uma organização de produção de software e suas filiais. O modelo de gestão de conhecimento e uma análise de sua eficácia são apresentados em um estudo de caso.

## 1. Introdução

A produção de software no Brasil é altamente deficitária se analisada sob o ponto de vista da balança comercial. Segundo o Ministério do Desenvolvimento, Indústria e Comércio Exterior, em 2009 o país importou cerca de US\$ 3 bilhões e as exportações chegaram perto dos US\$ 500 milhões. Para efeitos comparativos, segundo a Associação das Empresas Brasileiras Produtoras e Exportadoras de Software (BRASSCOM - <http://www.brasscom.org.br/>), países emergentes como a Índia e a Irlanda exportam cerca de oito vezes o que importam.

O índice acima explicitado ocorre devido ao fato do Brasil não possuir padrões de qualidade necessários para atingir o mercado externo. O baixo número de empresas com certificação na qualidade do processo (por exemplo, CMMI, MPS-BR, Moprosoft), a falta de profissionais com formação sólida em engenharia de software e a pesada carga tributária contribuem para os números espelhados. Aliado a estes fatores, as empresas que possuem certo padrão de qualidade têm dificuldade em replicar o conhecimento para parceiras e filiais. Empresas da região metropolitana de Curitiba e do Norte do Paraná passaram por este problema e contrataram uma consultoria que aliou a teoria de gestão de conhecimento (ontologias) e mapas conceituais na melhoria e replicação de seu processo. O objetivo deste trabalho é apresentar, justamente, os procedimentos, técnicas e modelo aplicado (emu ma delas) no referido contexto.

## 2. Objetivos e Justificativas

De posse da contextualização apresentada o foco deste projeto é constituir procedimentos, técnicas e modelos (permeados pela teoria de mapa conceitual) com o objetivo de proporcionar às empresas que possuem o conhecimento sobre processo de software a possibilidade da portabilização e replicação deste conhecimento.

## 3. Metodologia de Execução

A aplicação das teorias relacionadas à gestão de conhecimento e mapas conceituais foi materializada em uma consultoria realizada, no primeiro semestre de 2011, para uma empresa de desenvolvimento de software da região de Curitiba - PR. É importante ressaltar que os autores não possuem autorização formal para divulgar o nome da empresa neste texto. Este fato levou os autores de denominar a referida empresa como ALFA.

### A – Síntese do Problema

A empresa ALFA preocupa-se constantemente com a qualidade de seu processo de produção de software. A ALFA encontra-se em fase de expansão e em 2010 estabeleceu duas novas unidades de produção no interior do estado do Paraná, uma na região norte (com 25 profissionais) e outra na região oeste (com 32 profissionais). A escolha da região levou em consideração as potencialidades de crescimento e a formação de mão de obra qualificada.

Durante a configuração das unidades de produção o grande desafio da ALFA era manter a qualidade e produtividade do processo de software. Para manter o seu padrão de qualidade, a empresa teria que transferir grande parte do conhecimento da matriz (unidade capital) para as filiais (unidade interior).

É importante salientar que matriz não possuía uma gestão de conhecimento efetiva. O processo de software estava materializado textualmente e a disseminação das informações dentro do contexto organizacional era feita verbalmente entre os profissionais, fatos estes que dificultariam a transferência de conhecimento entre matriz e filial. Aliado a estes fatores, a empresa estava produzindo software em sua capacidade máxima e, por este motivo, disponibilizar duas equipes para replicar o conhecimento junto as filiais seria extremamente complicado naquele momento.

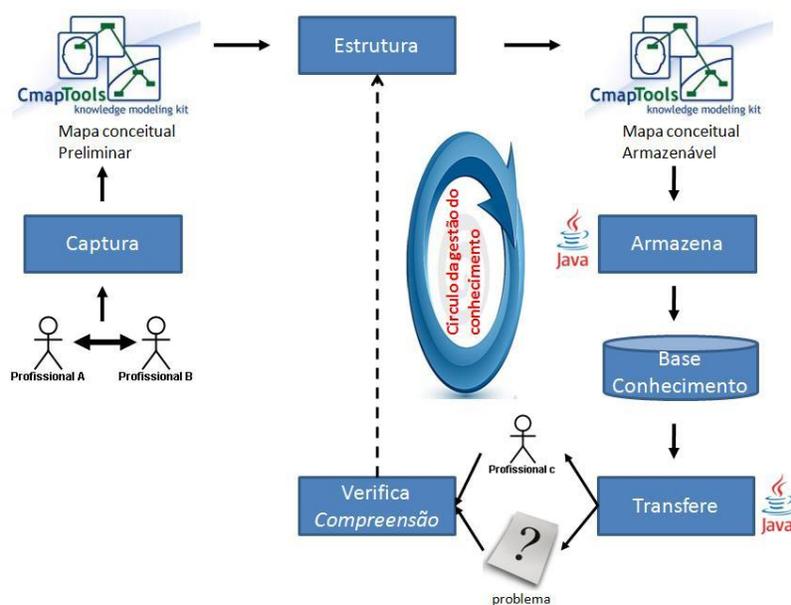
### B – O modelo de gestão de conhecimento adotado pela empresa

Para efetuar a transferência do conhecimento entre matriz e filial foi desenvolvido um modelo (vide Figura 1) que captura, estrutura, armazena, transfere e verifica se a conhecimento transferido foi compreendido.

O modelo apresentado na figura possui um dinamismo prático, ou seja, a execução das atividades, mapeadas pelos retângulos, acontecem durante o desenvolvimento do projeto de software. Para contextualizar tal dinamismo os autores deste trabalho utilizam um dos exemplos colecionados durante a consultoria.

### C – O Dinamismo do Modelo

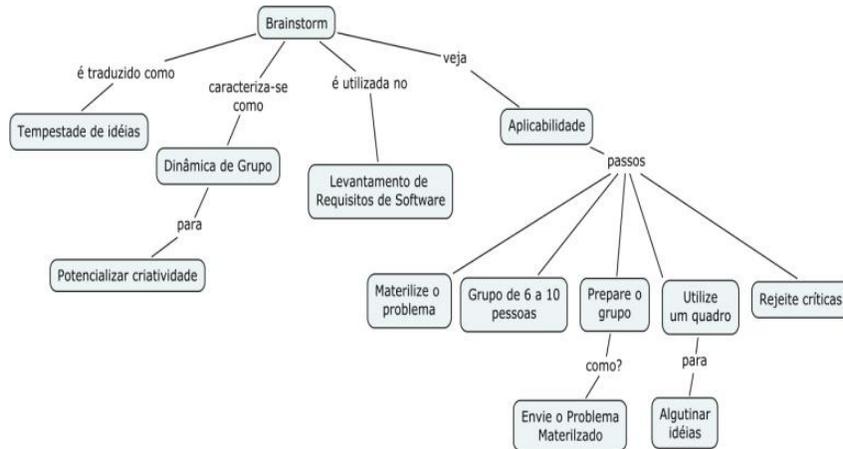
Parta do princípio que dois profissionais (caracterizados no modelo como **A** e **B**) estão trabalhando em um grande projeto de software e uma das formas de contextualizar o projeto passa pela utilização de técnica de *brainstorm*. A referida técnica está caracterizada, textualmente, no processo de software da empresa ALFA<sup>2</sup>. Ambos capturam a referida caracterização e modelam um mapa conceitual (vide Figura 2). Após a captura o mapa é estruturado e aspectos relacionados a formatação, sintática e semântica são verificados pelo engenheiro do conhecimento. De posse do mapa, o engenheiro do conhecimento utiliza uma ferramenta para o seu armazenamento na base de conhecimento. A ferramenta foi desenvolvida pela empresa ALFA, juntamente, com os consultores que participaram do projeto. A arquitetura da ferramenta não será apresentada neste texto.



<sup>2</sup> Caracterização textual da técnica de Brainstorm da empresa ALFA (<http://engenhariasoftware.wordpress.com/2010/12/01/tecnicas-para-desenvolver-um-brainstorm/>)

**Figura 1 – Modelo de gestão de conhecimento gerado para a empresa ALFA**

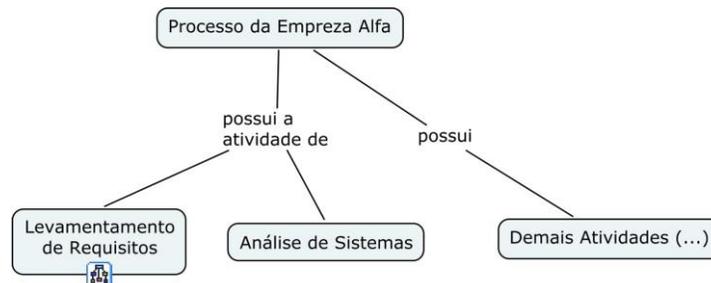
O conhecimento armazenado permite ao profissional imerso em uma das filiais resolver um determinado problema. Neste caso, o **profissional C** deve estruturar um *brainstorm* para contextualizar um projeto de software a ser desenvolvido pela filial. Os artefatos (documentos) gerados com a resolução do problema são analisados pelo engenheiro do conhecimento a partir de um conjunto de critérios de qualidade<sup>3</sup>, se os artefatos não satisfizerem o referido conjunto, o mapa conceitual gerado deve ser reestruturado. A reestruturação dos mapas mantém o círculo da gestão do conhecimento (vide modelo), círculo este responsável pela atualização constante da base.



**Figura 2 – Exemplo de um mapa mental (gerado pelos autores)**

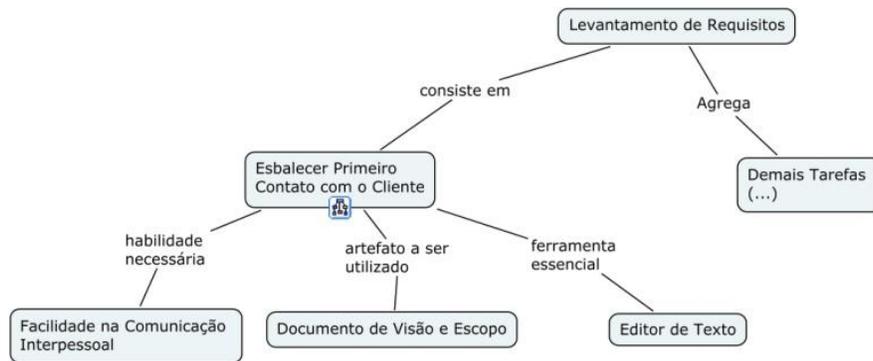
*D – Alguns Mapas Conceituais Gerados e Armazenados durante a Implantação da Filial*

Durante a execução do projeto de implantação do processo nas filiais foram gerados diversos mapas conceituais, alguns apresentados nesta seção. É importante salientar que os mapas auto se explicam e os autores não irão descrevê-los textualmente (vide figuras 3, 4 e 5).

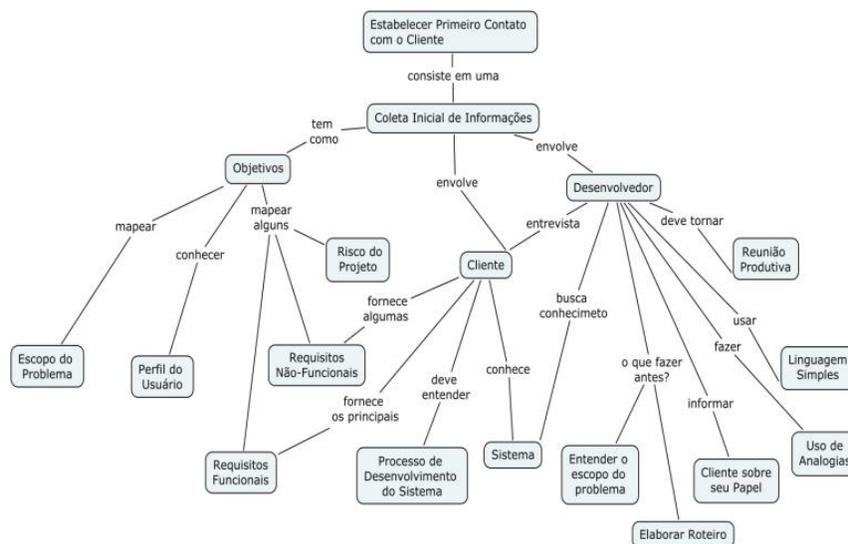


**Figura 3 – Mapa conceitual representando o processo de software da ALFA**

<sup>3</sup> Os critérios não serão apresentados neste texto



**Figura 4 – Mapa conceitual apresentado parte do conhecimento necessário para execução da atividade de Levantamento de Requisitos**



**Figura 5 – Mapa conceitual apresentado parte do conhecimento necessário para execução da tarefa Estabelecer Primeiro Contato com o Cliente**

#### 4. Resultados Obtidos

Os resultados obtidos com a execução (no ciclo 2011) deste projeto foram:

- 1 – Modelo de gestão de conhecimento permeado por mapas conceituais (vide Seção 3 – Figura 1) , este modelo foi publicado e pode ser utilizado por qualquer empresa.
- 2 - Ferramenta para o armazenamento e recuperação dos mapas mentais na base de conhecimento.
- 3 – Artigo científico descrevendo a aplicabilidade do modelo na empresa Alfa (vide referência no quadro 1).

#### Quadro 1 – Referência do artigo publicado com a execução do projeto

FABRI, J. A. ; LERARIO, A. ; DOMINGUES, A. L. S. ; TRINDADE, A. L. ; PESSOA, M. S. P. . Gestão de Conhecimento e Mapas Mepas Conceituas Aplicados a Melhoría de Processo de Software. In: 6a. Conferência Ibérica de Sistemas e Tecnologias de Informação, 2011, Chaves - Portugal. Anais da 6a. Conferência Ibérica de Sistemas e Tecnologias de Informação, 2011. v. II.

## 5. Aplicabilidade dos Resultados

A relação dos resultados obtidos com a aplicação do modelo levando em consideração o objetivo inicial traçado (mapear o conhecimento sobre o processo de software e replicá-lo em duas filiais) pode ser constatado por meio de resultados qualitativos e quantitativos, os itens abaixo apresentam tais resultados:

1 – Qualidade do produto e do processo: Com o desenvolvimento dos mapas conceituais, os profissionais que atuam na matriz, realizaram algumas alterações no processo de produção de software, várias delas promoveram um ganho de qualidade e produtividade, o tempo para o desenvolvimento dos artefatos (produtos) foi reduzido e a qualidade do software e a satisfação do cliente foi maximizada.

2 – A qualidade dos artefatos (produtos) gerados nas filiais também é um ponto a ser destacado com a aplicação do modelo. A cada círculo da gestão do conhecimento (vide Figura 1) realizado, os profissionais recém contratados, internalizavam o processo de software e os níveis produtividade e qualidade aumentavam.

3 – As orientações sintáticas e semânticas aplicados aos mapas foram de fundamental importância para o desenvolvimento das filiais, nesta etapa o papel do engenheiro do conhecimento foi de extrema importância, suas orientações foram eficientes e eficazes para o sucesso pretendido pela empresa ALFA.

4 – A estratégia em utilizar os mapas conceituais está embasada em uma consulta realizada pelos consultores que participaram ativamente no desenvolvimento do processo. Nesta pesquisa, um mapa conceitual foi apresentado a 89 profissionais da área produtiva de software (engenheiros, programadores, analista, testadores, gerentes de projetos), 83% dos profissionais fizeram a leitura do mapa rapidamente e não tiveram problemas sobre a sua compreensão.

5 – Por fim, o número de artefatos rejeitos pelos clientes das filiais foi diminuindo com a evolução qualitativa do conhecimento internalizado pelos profissionais e materializado nos mapas, já no quarto mês de consultoria os artefatos rejeitados foi 80% menor quando comparado ao primeiro mês de operação.

## 6. Característica Inovadores

A transferência de conhecimento entre as empresas do setor produtivo de software caracteriza-se como um dos pontos fortes deste projeto. Disseminar conceitos ligados a ontologias, transferência de tecnologia e conhecimento, reuso de processo/conhecimento é uma ação que merece ser destacada. Os autores que acreditam que a portabilidade dos mapas e facilidade de entendimento, focados na transferência e reuso de processo de software, caracterizam-se como fonte inovadora do projeto.

## 7. Conclusões e Perspectiva Futuras

Este trabalho apresentou a aplicação dos conceitos relacionados à gestão do conhecimento e mapas conceituais na melhoria e replicação de um processo de produção de software. O referencial teórico sobre conhecimento e gestão do conhecimento serviu de embasamento para a composição do modelo de gestão gerado para a empresa ALFA (vide Figura 2). A consulta aos profissionais da área produtiva de software, sobre o entendimento dos mapas conceituais, apresentada na seção anterior, e a sua utilização de forma ampla e irrestrita para a representação do conhecimento determinaram a utilização desta técnica no desenvolvimento deste trabalho.

A configuração do modelo e a materialização dos mapas foram realizadas em uma consultoria no primeiro semestre de 2010. As atividades de captura e armazenamento de conhecimento foram realizadas durante os 3 primeiros meses de trabalho. Já atividade replicação passou por um processo de amadurecimento de 6 meses, pois o círculo da gestão do conhecimento teve que ser percorrido inúmeras vezes. Após 9 meses de trabalho as filiais estavam atendendo o mercado consumidor de software com os mesmos níveis de qualidade encontrados na matriz.

É importante ressaltar o papel do engenheiro do conhecimento, profissional com ampla experiência sobre o processo de produção da empresa e treinado pelos consultores para o desenvolvimento e reestruturação dos mapas conceituais a serem armazenado na base.

Os gestores estratégicos da empresa também fizeram um grande esforço para que os procedimentos estabelecidos na consultoria fossem colocados em prática, este fato também contribuiu para o sucesso do projeto como um todo.

A utilização de mapas conceituais minimizou a utilização de recursos humanos totalmente alocados, ou seja, equipes nas filiais puderam internalizar conhecimento, sem que um recurso humano, lotado na matriz, precisasse fornecer integralmente todo o treinamento necessário para filial.

Salienta-se que existe uma relação estreita entre os mapas conceituais e a teoria ontológica originadas nos pensamentos filosóficos de Aristóteles. Uma ontologia caracteriza-se como a especificação formal e explícita de um conhecimento que necessita ser compartilhado. Nessa definição, formal traduz a ideia de legível para computadores (é possível formalizar semântica e sintaticamente os mapas); especificação explícita relaciona-se a conceitos, propriedades, relações e funções explicitamente definidos (característica perceptível nos mapas); compartilhado significa que todo o conhecimento representado é fruto de um consenso entre os participantes do modelo (fato espelhado no modelo de gestão de conhecimento apresentado neste texto).

Uma nova consultoria utilizando as técnicas e procedimentos apresentados neste trabalho serão aplicadas em 2012 em outra empresa do setor de software, este fato contribuirá para um amadurecimento do modelo que une gestão de conhecimento e mapas conceituais.

## 8. Referências Bibliográfica

- Alias, Maizam; Suradi, Zurinah. "Concept Mapping: A Tool For Creating a Literature Review". in. Proceedings Third International Conference on Concept Mapping. Tallinn, Estonia and Helsinki, Finland Sept. 22-25. 2008.
- Dandolini, Gertrudes Aparecida, et. al. "Mapas Conceptuales, Enseñanza de Lógica y Educación a Distancia". in. Proceedings Fourth International Conference on Concept Mapping. Viña del Mar, Chile Oct. 5-7. 2010.
- Davenport, Thomas H. "Knowledge Management Case Study at Hewlett-Packard Early 1996". in University of Texas at Austin web site – <http://www.mcombs.utexas.edu/kman/hpcase.htm>. Visited in november 2009. USA. 1996.
- Felix, Patrícia P. "Análise Situacional da Gestão de Conhecimento em uma Instituição de Ensino Superior por Meio da Espiral do Conhecimento. Dissertação de Mestrado Apresentada à Faculdade de Engenharia de Produção da Universidade Federal de Santa Catarina.
- Flores, Rafael Pérez. "Mapas Conceptuales y Aprendizaje de Matemáticas". in. Proceedings Second International Conference on Concept Mapping. San José , Costa Rica Sept. 5-8, 2006.
- Fourie, Louis C. H.; Westhuizen, Trunel van der "The Value and Use of Concept Maps in The Alignment of Strategic Intent". in. Proceedings Third International Conference on Concept Mapping. Tallinn, Estonia and Helsinki, Finland Sept. 22-25. 2008.
- Niskanen Vesa A.; Kyrö, Paula. "A Concept-Map Modeling Approach to Business Planning in a Computer Environment". in. Proceedings Third International Conference on Concept Mapping. Tallinn, Estonia and Helsinki, Finland Sept. 22-25. 2008.
- Novak, J. D.; Cañas, A. J. "The origins of the concept mapping tool and the continuing evolution of the tool". In Information Visualization Journal, (5). 2006. pp. 175-184.
- Ramasubramanian, Shivram; Jagadeesan, Gokulakrishnan. "Knowledge Management at Infosys". in IEEE Software. USA: IEEE. may/june 2002.
- Renauld, María Eugenia Venegas. "El Empleo de los Mapas Conceptuales en la Educación Superior Universitaria". in. Proceedings Second International Conference on Concept Mapping. San José , Costa Rica Sept. 5-8, 2006.
- Rivera, Carlos Araya. "El Mapa Conceptual como Herramienta para la Enseñanza em las Ciencias de la Comunicación. in. Proceedings Fourth International Conference on Concept Mapping. Viña del Mar, Chile Oct. 5-7, 2010.
- Russel, Karen G. "What is knowledge management?". in EE – Evaluation Engineering web site – <http://www.evaluationengineering.com/archive/article/1102manag.htm>. USA: Nelson Publishing Inc. Visited in november 2005.
- Tavares, Edval S. "Uma Contribuição para os Processos de Gerência de Projetos através da Gerência do Conhecimento". Tese de doutorado apresentada ao Departamento de Engenharia de Produção da Escola Politécnica da USP. São Paulo. 2004.
- Tavares, R. "Construindo mapas conceituais". Ciências & Cognição, 12(4). 2007. pp. 72-85.
- Terra, José C. C. "Gestão do Conhecimento: Aspectos Conceituais e Estudo Exploratório sobre as Práticas de Empresas Brasileiras". Tese de doutorado apresentada ao Departamento de Engenharia de Produção da Escola Politécnica da USP. São Paulo. 1999.

# Ambiente Integrado – Uma Abordagem Automatizada e Colaborativa para Gestão de Processos do MPS.Br

Simone Vasconcelos Silva, Aline P. V. de Vasconcelos, Jonivan Lisboa Coutinho

Instituto Federal Fluminense (IFF)  
Rua Dr. Siqueira, 273 – Parque Dom Bosco – CEP 28030-130

Campos dos Goytacazes – RJ – Brasil

{simonevs, apires, jlisboa}@iff.edu.br

**Resumo.** Este artigo apresenta uma abordagem que visa a integração de alguns processos do MPS.Br em um ambiente colaborativo de código aberto, de modo que seja atingido um nível de integração não encontrado em outras abordagens estudadas. O objetivo principal é abordar uma metodologia de Ambiente Integrado que apresenta ferramentas para automatizar processos do MPS.Br. O Ambiente Integrado, atualmente possui ferramentas para automatizar os seguintes processos: Gerência de Projetos, Gerência de Requisitos, Gerência de Configuração, Gerência de Portfólio e parte do processo de Validação (nível D) que cobre os Testes Funcionais. Este ambiente foi utilizado para auxiliar o desenvolvimento de software para projetos dos Ministérios da Educação (MEC) e da Comunicação (MC).

## 1. Introdução

No mundo moderno, aspectos como o constante crescimento da demanda por software, sua inegável importância dentro das organizações e seu grau de complexidade cada vez maior, apontam para um cenário onde é fundamental o desenvolvimento de produtos de qualidade. Sendo assim, a busca pela melhoria contínua no processo de desenvolvimento de software constitui um fator crítico de sucesso. De acordo com Pressman (2007), iniciar o desenvolvimento de um software sem um plano de projeto sólido certamente ocasionará um comprometimento no sucesso e na qualidade do produto final.

Nesse contexto, diversos modelos de maturidade para o processo de desenvolvimento de software vêm sendo propostos com base em estudos acadêmicos e observação do mercado produtivo. Dentre eles, destaca-se para o mercado nacional de software o MPS.Br (Melhoria do Processo de Software Brasileiro), que é organizado em níveis incrementais de maturidade, com processos e resultados esperados bem definidos, que podem ser atingidos de forma gradual pelas organizações que adotam sua implementação (SOFTEX, 2011).

Este trabalho apresenta uma metodologia para integração, de forma automatizada através de um ambiente, de alguns dos processos do MPS.Br. A metodologia integra todos os processos do nível G (Gerência de Projetos e Gerência de Requisitos), dois dos processos do nível F (Gerência de Configuração e Gerência de Portfólio) e parcialmente um dos processos do nível D (Testes Funcionais, contemplados pelo processo de Validação). Estes processos apresentam aspectos ligados diretamente à qualidade do produto gerado, tais como: a gestão de equipes de produção de software e recursos por elas utilizados para a geração de produtos; a gestão de documentação de software (modelos, códigos, estratégias de implementação, versões); a garantia de atendimento das funcionalidades esperadas pelos produtos gerados, e conseqüentemente a satisfação dos clientes. Considera-se que a implementação de tais processos em uma organização determina o nível mínimo de maturidade esperado no processo de desenvolvimento, capaz de trazer resultados satisfatórios em relação à documentação de software, gerenciamento de equipes e qualidade do produto.

O ambiente Integrado foi desenvolvido pelo projeto GESTÃO-EPCT (Gerência de Projetos da Educação Profissional e Tecnológica) e QUALI-EPCT (Garantia da Qualidade de Software da Educação Profissional e Tecnológica), onde os participantes são professores, alunos e egressos do Instituto Federal Fluminense (IFF). Estes projetos estão vinculados ao núcleo de Engenharia de Software (NES) do instituto e são patrocinados pela RENAPI (Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais). Foram utilizados como estudos de caso projetos de software mantidos pelo Ministério da Educação (MEC), através da RENAPI e pelo Ministério das Comunicações (MC), através do projeto Formação GESAC.

## 2. Objetivos e Justificativa

A implementação dos processos do MPS.Br citados na seção anterior pode gerar uma quantidade considerável de artefatos de documentação, relativos ao software em si, aos recursos utilizados pela equipe de desenvolvimento e a gestão do projeto como um todo. Embora isso seja fundamental para que a maturidade do processo de desenvolvimento seja atingida, tal fato pode acarretar um esforço para produção e manutenção adequada de todos os documentos produzidos.

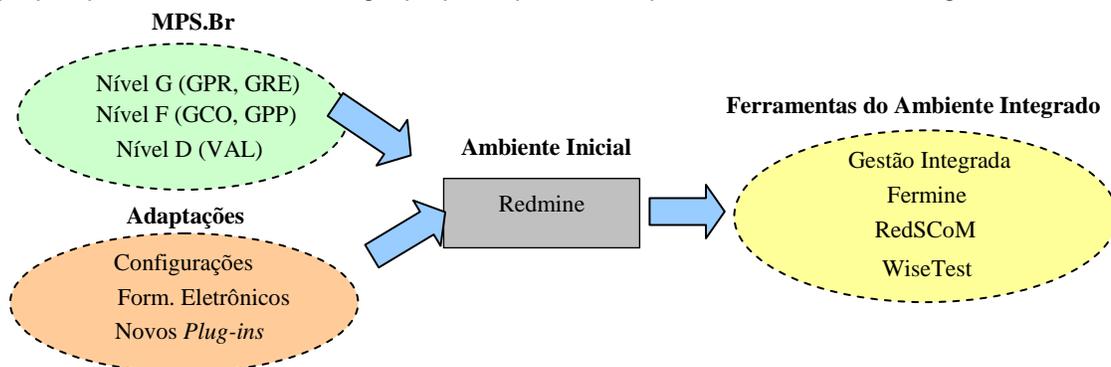
Com o objetivo de minimizar tais esforços, é apresentada neste artigo uma abordagem que integra os processos do MPS.Br em um ambiente único, de modo a possibilitar um gerenciamento integrado e padronizado da documentação produzida durante todo o processo de concepção e implementação de sistemas. Isso facilita, dentre outras coisas, a manutenção dos diversos tipos de documentos e sua utilização por equipes de desenvolvimento que, em muitas situações no mundo atual, encontram-se dispersas geograficamente e temporalmente. Sendo assim, é necessária a adoção de uma metodologia de desenvolvimento colaborativo, de modo a minimizar os impactos negativos que possam acarretar perda da qualidade no produto.

## 3. Metodologia de Execução

O gerenciamento da documentação gerada pelos processos implementados é feito por um conjunto de ferramentas eletrônicas, integradas em um ambiente colaborativo, adaptado com funcionalidades capazes de atender satisfatoriamente os processos citados anteriormente.

As ferramentas desenvolvidas têm como base a plataforma Redmine. Esta plataforma possui funcionalidades relacionadas ao gerenciamento de projetos, portanto através dela é possível cadastrar recursos humanos, cadastrar tarefas associadas as pessoas, gerar cronogramas e diagramas temporais de acompanhamento de execução (Gantt) e também disponibiliza recursos colaborativos como *wiki*, fóruns, repositório de versões de código, armazenamento de arquivos e documentos (REDMINE, 2012).

O Redmine é desenvolvido usando-se o *framework* Ruby on Rails, possuindo como características principais o código aberto, a facilidade de extensão e configurações personalizadas. Estas características facilitaram a implementação das ferramentas apresentadas neste trabalho e para tal foram elaboradas adaptações através de configurações, Implementação de formulários eletrônicos e de *plug-ins* específicos para a obtenção dos resultados esperados nos processos do MPS.Br, que se integram no ambiente juntamente com os demais *plug-ins* nativos da própria plataforma. A metodologia proposta pode ser representada através da Figura 1.



**Figura 1 – Metodologia proposta para o Ambiente Integrado**

Conforme mostra a Figura 1, as adaptações feitas no ambiente Redmine (versão 0.1) originaram um ambiente integrado para o atendimento de processos do MPS.Br, contendo as seguintes ferramentas:

- **Gestão Integrada:** atende à Gerência de Projetos, cujo objetivo é estabelecer e manter planos que definem as atividades, prazos, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções, quando houver desvios significativos no desempenho do mesmo (SILVA et al., 2011);
- **Fermine:** atende à Engenharia de Requisitos, visando buscar técnicas bem definidas para elicitar, especificar, validar, documentar e rastrear requisitos (ALMEIDA et al., 2010), (ALMEIDA et al., 2011);
- **RedSCoM:** atende à Gerência de Configuração: fornece subsídios para identificar e documentar as

características físicas e funcionais de um item de configuração, controlar as alterações nessas características, armazenar e relatar o processamento das modificações e o estágio da implementação e verificar a compatibilidade com os requisitos especificados (CARVALHO et al., 2010);

- WiseTest: atende parcialmente ao processo de Validação através dos Testes Funcionais e possui como principal funcionalidade a geração automática de casos de teste (CARVALHO et al., 2011).

## 4. Resultados Obtidos

### 4.1. Pedidos de patente e/ou patentes registradas

A documentação necessária ao registro de patente da ferramenta Gestão Integrada está pronta para ser enviada e as demais encontram-se em elaboração.

### 4.2. Produtos de software gerados

Os produtos de software gerado estão relacionados as ferramentas que deram origem ao Ambiente Integrado (Figura 1). Basicamente, a utilização destas ferramentas permite a obtenção dos resultados esperados descritos nos processos do MPS.Br, a saber:

a) Resultados do processo de Gerência de Projetos e Gerência de Portfólio (Ferramenta Gestão Integrada):

O escopo do trabalho para o projeto é definido através de um formulário eletrônico para o Termo de Abertura do Projeto (TAP) e um *plug-in* para Estrutura Analítica do Projeto (EAP);

- As tarefas e os produtos de trabalho do projeto são dimensionados utilizando-se métodos adequados (está sendo desenvolvida a funcionalidade para cálculo das estimativas de tamanho e complexidade dos produtos de software através da Análise de Pontos por Função e Análise de Pontos por Caso de Uso);
- O modelo e as fases do ciclo de vida do projeto são definidos no TAP;
- As estimativas de prazos e custos do projeto são calculadas automaticamente através das fórmulas do método EVM (*Earned Value Management*);
- O orçamento é feito através de formulário eletrônico e a ferramenta permite a visualização de um diagrama de Gantt de acordo com diversos níveis da EAP;
- Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados através de formulários eletrônicos para lançamento de riscos e geração do plano de ação;
- A ferramenta possui um cadastro de recursos humanos relacionado com as competências necessárias no projeto, possibilitando a criação da matriz de responsabilidades do projeto, definindo perfis e alocando tarefa de acordo com a EAP, é disponibilizado um formulário eletrônico para solicitação de capacitação, o qual será avaliado pela gerência do projeto;
- A ferramenta permite a documentação do projeto e do produto através do uso de repositórios, documentos e arquivos. Através de um formulário eletrônico é possível estabelecer o plano de comunicação identificando o armazenamento e distribuição das informações;
- O Plano de Projeto é desenvolvido no *wiki* da ferramenta, contendo *links* para todos os documentos do projeto;
- Registros de problemas identificados e das ações corretivas são estabelecidos através de formulários eletrônicos;
- A ferramenta oferece também a possibilidade de lançamento e controle das tarefas através do sistema *kanban* da metodologia *Lean* (método ágil) (POPPENDIECK e POPPENDIECK, 2003).

b) Resultados do processo de Gerência de Requisitos (Ferramenta Fermine):

- O entendimento dos requisitos é obtido junto aos fornecedores através de um formulário eletrônico para descrição dos requisitos;
- Os requisitos de software são aprovados gerando um controle do status do requisito pela ferramenta;
- Através da ferramenta é possível especificar os casos de uso relacionados aos requisitos através da descrição, atores, fluxos principais e alternativos, telas e diagramas. Foi desenvolvido um *plug-in* que proporciona a rastreabilidade entre os requisitos e os artefatos.

c) Resultados do processo de Gerência de Configuração (Ferramenta RedsCon):

- Um sistema de Gerência de Configuração é estabelecido e mantido através de soluções como: repositório integrado ao Redmine e formulário eletrônico para solicitações de manutenção;
- A ferramenta permite a identificação dos itens de configuração (ICs) como parte do Plano de Gerência de Configuração de Software (GCS), que contém: introdução, itens de configuração, *baselines* e infra-estrutura;
- A ferramenta possibilita o planejamento e geração automática de *baselines*, solicitando o *login* e *password* de acesso ao repositório. Após verificar estas informações, é recuperado o conteúdo do *trunk*, pasta onde são encontrados os ICs referentes à versão corrente de desenvolvimento, e, então, uma nova tag é gerada com este conteúdo, através do mecanismo de *tags* do repositório;
- Através da visualização do repositório, a ferramenta permite a verificação dos comentários das operações de *commit*, realização de *diff*, etc.;
- Toda modificação em um IC, quando ele passa a fazer parte de uma *baseline*, deve passar por um controle formal de modificações, em que é gerada uma solicitação de modificação que deverá ser analisada.

d) Resultados da aplicação de Testes Funcionais automatizados (Ferramenta WiseTest):

- Geração de casos de testes a partir dos casos de uso documentados;
- Registro de resultados da execução dos testes;
- Inclusão de artefatos de testes (Plano de Testes) para facilitar o gerenciamento das atividades;
- Versionamento de artefatos da área de testes.

O ambiente integrado permite que (CARVALHO, et al., 2011), (ALMEIDA, et al., 2011):

- RedScoM versione os requisitos cadastrados na Fermine, os planos de projeto cadastrados na Gestão Integrada e os artefatos de testes cadastrados na WiseTest;
- WiseTest crie casos de testes a partir da leitura de casos de uso descritos na Fermine e crie automaticamente tarefas na Gestão Integrada a partir de casos de uso executados com falha;
- Gestão Integrada utilize a funcionalidade de requisitos da Fermine para geração automática de tarefas, onde um requisito aceito pelo gerente do projeto terá o status “em desenvolvimento” e este irá gerar uma tarefa que será associada ao EAP do projeto. Ao concluir a tarefa o status do requisito passará para “atendido”.

#### **4.3. Processos que contribuem para a melhoria da qualidade de software descritos e institucionalizados**

Elaboração de padrões para os documentos de gerência de projetos e gerência de requisitos através dos formulários eletrônicos para os softwares desenvolvidos pela Renapi (Rede Nacional de Pesquisa e Inovação) da SETEC/MEC.

#### 4.4. Artigos publicados

- Gestão Integrada – Uma Ferramenta para Atender aos Processos de Gerência de Projetos e Portfólio do MPS.Br. In: SBQS 2011 – WGPS 2011. Curitiba, 2011;
- Apoio aos Processos de Gerência de Requisitos e Verificação e Validação em um Ambiente Integrado. VII Workshop Anual do MPS, WAMPS 2011. Campinas, 2011;
- Gestão Integrada - Uma Ferramenta para Atender as Áreas da Gerência de Projetos. III Enfepro- Encontro Fluminense de Engenharia de Produção. Rio de Janeiro, 2011;
- Um modelo para avaliar a usabilidade de Sistemas de Suporte à Decisão. IADIS CIAWI – Conferência IADIS Ibero Americana WWW/Internet 2011. Rio de Janeiro, 2011.

#### 4.5. Recursos humanos capacitados (especialistas, mestres, doutores, etc.)

Foram capacitados 4 especialistas em Sistemas de Informação no Instituto Federal Fluminense e 3 mestrados (com trabalhos em andamento) na Universidade Estadual do Norte Fluminense.

#### 4.6. Dissertações e/ou teses geradas

- Modelo Híbrido de Desenvolvimento de Software: uma Abordagem no Processo de Gerência de Projetos - SCRUM X MPS.Br. Monografia de Pós-Graduação em Análise, Projeto e Gerência de Sistemas de Informação. Instituto Federal Fluminense.
- Uma abordagem de integração de métricas estáticas e dinâmicas para apoiar a avaliação da qualidade em modelos de software. Monografia de Pós-Graduação em Análise, Projeto e Gerência de Sistemas de Informação. Instituto Federal Fluminense.
- Avaliação da Usabilidade de um Sistema de Suporte à Decisão: O Caso do Mdecision. Monografia de Pós-Graduação em Análise, Projeto e Gerência de Sistemas de Informação. Instituto Federal Fluminense.
- Uma Aplicação da Gerência de Projetos na Implantação de Ferramentas de EaD no Ensino Presencial. Monografia de Pós-Graduação em Produção e Sistemas. Instituto Federal Fluminense.

#### 4.7. Eventuais parcerias ou programas de transferência de tecnologia efetuados

Parcerias com SETEC/MEC e MC.

#### 4.8. Outros resultados

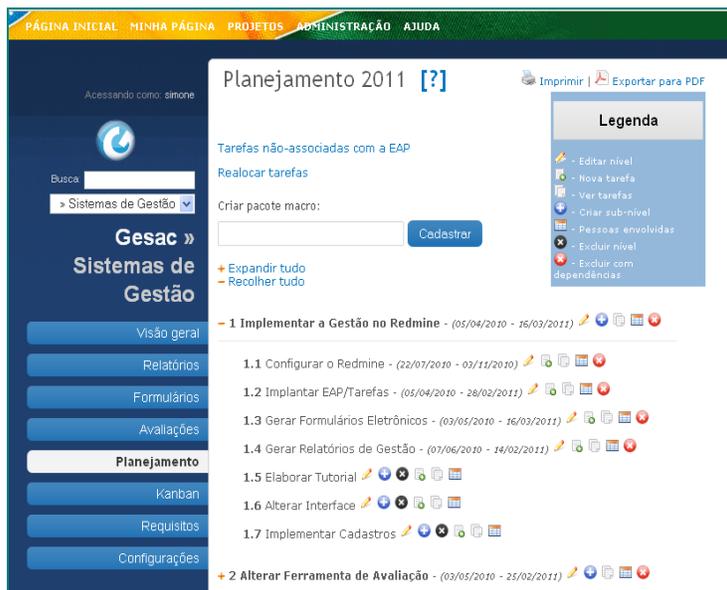
- Tutoriais disponíveis em [www.renapi.gov.br](http://www.renapi.gov.br);
- Treinamentos on-line disponíveis em [www.es.iff.edu.br/eadquali](http://www.es.iff.edu.br/eadquali);
- Utilização das ferramentas Gestão Integrada e Fermine nas aulas do curso superior em Sistemas de Informação do Instituto Federal Fluminense.

#### 5. Aplicabilidade dos Resultados

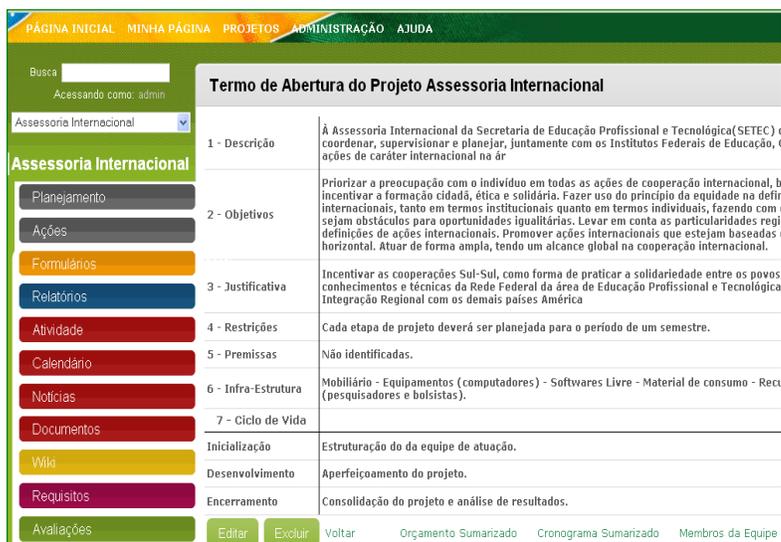
A ferramenta Gestão Integrada é utilizada atualmente pelos seguintes órgãos do Governo Federal: MEC (através dos projetos da RENAPI/SETEC) e MC (através do projeto Formação GESAC). Os projetos da RENAPI são em sua maioria relacionados a Sistemas de Informação que atendam a Educação Profissional e Tecnológica (RENAPI, 2012) e o projeto Formação GESAC trata da Inclusão Digital em todo o país (GESAC, 2012). Em todos os projetos, os *stakeholders* estão distribuídos por diversas localidades, com isso todo planejamento e execução ocorrem de forma colaborativa.

A ferramenta é totalmente adaptável à realidade de cada projeto, onde para os projetos da RENAPI e do GESAC várias funcionalidades específicas surgiram da particularidade de cada um dos projetos e foram totalmente integradas na ferramenta. Uma funcionalidade específica que está sendo bastante utilizada por praticamente todos os projetos e que surgiu a partir das necessidades destes, está relacionada com Avaliações (questionários com perguntas totalmente configuráveis, gráficos estatísticos e relatórios), que no GESAC é utilizada para avaliar a formação oferecida pelos cursos de tutoria da formação à distância e na RENAPI é utilizada para avaliar a implantação dos sistemas nos Institutos Federais. A partir da metodologia de gestão de projetos e portfólio utilizada na ferramenta Gestão Integrada originou-se o Sistema de Gestão dos Institutos Federais, cujo funcionamento é baseado na concepção de ações, objetivos, metas e projetos.

A Figura 2 ilustra o uso da funcionalidade de EAP na ferramenta Gestão Integrada no projeto de Formação GESAC para o subprojeto Sistemas de Gestão e a Figura 3 ilustra a funcionalidade do TAP na ferramenta Gestão Integrada no projeto Assessoria Internacional da SETEC.



**Figura 2. EAP do Subprojeto Sistemas de Gestão do Projeto Formação GESAC**



**Figura 3. TAP do Projeto Assessoria Internacional da SETEC**

As ferramentas Fermine, RedSCoM e WiseTest estão sendo utilizadas atualmente nos sistemas desenvolvidos por projetos da RENAPI, tais como SIGA-EPCT e Gestão-EPCT. A Figura 4 mostra a funcionalidade de rastreabilidade da ferramenta Fermine em relação aos casos de uso no sistema SIGA (Sistema Integrado de Gestão Acadêmica) do projeto SIGA-EPCT.



**Figura 4. Rastreabilidade na ferramenta Fermine para o sistema SIGA**

## 6. Características Inovadoras

Uma característica inovadora deste projeto é a possibilidade de integrar em um único ambiente de forma totalmente colaborativa, ferramentas capazes de automatizar processos do MPS.Br. Tornando desta forma o trabalho da documentação dos softwares mais eficiente, garantindo maior organização, produtividade e qualidade nos processos.

## 7. Perspectivas de Trabalhos Futuros

O Ambiente Integrado encontra-se em melhoria e evolução de suas ferramentas, e estas de suas funcionalidades. Desta forma a perspectiva de trabalhos futuros está relacionada ao desenvolvimento de ferramentas

que atendam a todos os processos do MPS.Br, tornando o Ambiente Integrado capaz de atender a todas as necessidades relacionadas ao processo de desenvolvimento de software.

Atualmente encontra-se na fase final de desenvolvimento uma ferramenta para avaliação do produto desenvolvida de acordo com a metodologia de SILVA (2003). A integração de testes automatizados através de metodologias ágeis na ferramenta WiseTest, assim como a geração de cenários (metodologias ágeis) na ferramenta Fermine.

## **Bibliografia**

- Almeida, G., Ramos, B. *et al.* (2010), Ferramenta de Apoio à Engenharia de Requisitos Integrada a um Ambiente Colaborativo de Código Aberto. Congresso Brasileiro de Software: Teoria e Prática (CBSOFT). Salvador, Ba.
- Almeida, G., Neto, M., Ramos, B., Carvalho, J., Barcelos, M., Vasconcelos, S., (2011), Apoio aos Processos de Gerência de Requisitos e Verificação e Validação em um Ambiente Integrado. Workshop Anual do MPS, WAMPS 2011, Campinas, SP.
- Carvalho, J., Amaral, M., Barcelos, M., Vasconcelos, S., Vasconcelos, A. (2011), Integração da Gerência de Configuração com a Gerência de Projetos e de Requisitos em um Ambiente Colaborativo. Workshop Anual do MPS, WAMPS 2010, Campinas, SP.
- GESAC (2012) – Governo Eletrônico – Serviços de Atendimento ao Cidadão, <http://www.gesac.gov.br>, acessado em 17/02/2012.
- Poppendieck, M., Poppendieck, T., (2003), Lean Software Development: An Agile Toolkit for Software Development Managers. [http://ebookee.org/Lean-Software-Development-An-Agile-Toolkit-for-Software-Development-Managers-Repост-\\_313745.html](http://ebookee.org/Lean-Software-Development-An-Agile-Toolkit-for-Software-Development-Managers-Repост-_313745.html).
- Pressman, R. S. (2007), Engenharia de Software, Makron Books, 6ª edição. São Paulo.
- REDMINE (2012), <http://www.redmine.org>, acessado em 01/06/2011.
- RENAPI (2012) - Portal da Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais, <http://www.renapi.org>, acessado em 01/03/2012.
- Silva, S.V., Lisboa, J. C., Vasconcelos, A. P. V., Barbosa, C. D., Reis, M., Leite, R., Barroso, L. (2011), Gestão Integrada – Uma Ferramenta para Atender aos Processos de Gerência de Projetos e Portfólio do MPS.Br. In: SBQS 2011 – WGPS 2011. Curitiba.
- SOFTEX (2011) – Associação Para Promoção da Excelência do Software Brasileiro. MPS.Br – Guia Geral: 2011. [http://www.softex.br/mpsbr/\\_guias/guias/MPS.BR\\_Guia\\_Geral\\_2011.pdf](http://www.softex.br/mpsbr/_guias/guias/MPS.BR_Guia_Geral_2011.pdf).

## **Consolidação de um Método para Customização de Modelos de Capacidade/Maturidade de Processos de Software**

### **Jean Carlo Rossa Hauck**

Universidade Federal de Santa Catarina (UFSC)  
Programa de Pós-Graduação em Engenharia e  
Gestão do Conhecimento - Florianópolis, SC, Brasil  
jeanhauck@incod.ufsc.br

### **Chrisriane Gresse von Wangenheim**

Universidade Federal de Santa Catarina (UFSC)  
Programa de Pós-Graduação em Ciência da  
Computação Florianópolis, SC, Brasil  
gresse@incod.ufsc.br

Abstract - *Software Process Capability/Maturity Models (SPCMMs) are repositories of best practices for software processes suitable for assessing and/or improving processes in software intensive organizations. Each software development domain, however, presents particular needs, which has led to the tendency of SPCMMs customization for specific domains, which has often been undertaken in an unsystematic way. This paper presents a method for the customization of SPCMMs for specific domains, developed based on standards development, processes modeling and knowledge engineering techniques as well as experiences reported in the literature. Formative evaluations of the method have taken place through case studies and summative evaluation has been conducted through an Expert Panel. The observed results reveal early evidence that the method is suitable for SPCMMs customization.*

**Keywords - Software process improvement, software process assessment, capability/maturity models**

## I. Introdução

Modelos de Maturidade/Capacidade de Processos de Software (SPCMM – *Software Process Capability/Maturity Models*) são repositórios de melhores práticas de processos de software, desenvolvidos com base nas experiências de engenharia e princípios de gestão de processos, organizadas sob o conceito de capacidade/maturidade de processos, adequados para avaliar ou melhorar os processos de uma organização de software **Erro! Fonte de referência não encontrada.** Vários SPCMMs têm sido desenvolvidos pela comunidade de engenharia de software, tais como CMMI-DEV **Erro! Fonte de referência não encontrada.** e ISO/IEC 15504 **Erro! Fonte de referência não encontrada.**, e seu uso para a melhoria e avaliação de processo de software está bem estabelecido na prática. No entanto, como esses modelos genéricos são destinados a atender uma vasta gama de diferentes serviços, processos, tecnologias e produtos de software, sua aplicação prática muitas vezes exige uma customização para o contexto específico **Erro! Fonte de referência não encontrada.** Diversos domínios de desenvolvimento de software têm necessidades específicas de qualidade que precisam ser adequadamente contempladas. Da mesma forma, diversas são as normas específicas que devem ser seguidas no desenvolvimento de software em setores altamente regulados. No domínio da saúde, por exemplo, processos e produtos de software precisam atender normas como AAMI/IEC 62304, FDA, ISO 14971, IEC 60601-1-4, ISO 13485, etc. **Erro! Fonte de referência não encontrada.** A fim de facilitar essa adaptação dos modelos genéricos a estes contextos específicos, tem sido observada uma tendência atual para o desenvolvimento de customizações dos SPCMMs genéricos para domínios específicos **Erro! Fonte de referência não encontrada.** Assim, várias iniciativas têm sido realizadas para se especializar e aperfeiçoar SPCMMs genéricos, adaptando as melhores práticas de melhoria de processos para domínios específicos, tais como: SPICE4SPACE **Erro! Fonte de referência não encontrada.**, OOSPICE**Erro! Fonte de referência não encontrada.**, SMCMM **Erro! Fonte de referência não encontrada.**, etc.

No entanto, a maioria destas iniciativas não tem adotado uma abordagem sistemática para a customização das normas e modelos genéricos para as necessidades dos domínios específicos **Erro! Fonte de referência não encontrada.** De fato, a literatura que detalha como SPCMMs têm desenvolvidos/evoluídos/adaptados é extremamente rara **Erro! Fonte de referência não encontrada.** As organizações normativas, como ISO **Erro! Fonte de referência não encontrada.** ou IEEE **Erro! Fonte de referência não encontrada.**, definem processos genéricos de alto nível para o desenvolvimento e publicação de normas, mas, no entanto, estes processos não descrevem como customizar modelos existentes e não fornecem suporte técnico detalhado para o desenvolvimento específico de SPCMMs.

Considerando, que SPCMMs são repositórios de conhecimento na forma de "melhores práticas" **Erro! Fonte de referência não encontrada. Erro! Fonte de referência não encontrada.**, a Engenharia do Conhecimento (EC) pode fornecer uma contribuição importante no desenvolvimento de SPCMMs, provendo métodos para a obtenção do entendimento das estruturas e processos utilizados por especialistas, no intuito de criar uma melhor integração da tecnologia da informação em suporte ao trabalho intelectual **Erro! Fonte de referência não encontrada.** Atualmente, no entanto, abordagens da EC não têm sido aplicadas com esse objetivo específico.

Apesar de diversas customizações de SPCMMs já terem sido realizadas para domínios específicos **Erro! Fonte de referência não encontrada.**, a pesquisa sobre como executar este tipo de customização de forma sistemática é bastante rara. Um dos poucos trabalhos sendo realizados com este objetivo na área específica de Engenharia de Software, é o framework de método PRO2PI-MFMOD **Erro! Fonte de referência não encontrada.** para o desenvolvimento de SPCMMs, que toma por base as experiências anteriores dos autores no desenvolvimento de diversos modelos. O framework é composto por sete etapas: (i) decisões iniciais, (ii) análise de fontes (de boas práticas), (iii) estratégia de desenvolvimento, (iv) design do modelo utilizando ISO/IEC 15504 como a estrutura geral, (v) desenvolvimento do modelo *draft*, (vi) validação do modelo *draft*, e (vii) consolidação do modelo. Este trabalho representa uma importante pesquisa inicial para se alcançar uma abordagem para a customização de SPCMMs, mas, no entanto, não se constitui em um método, mas em um framework de método sem conter o detalhamento necessário para diretamente realizar a customização.

Existem também poucos trabalhos com este mesmo objetivo na área de modelos de negócio. Neste contexto, Bruin & Rosemann **Erro! Fonte de referência não encontrada.**, propõem uma seqüência de etapas para o desenvolvimento de Modelos de Avaliação de Maturidade, incluindo: (i) a definição do escopo do modelo, (ii) o projeto de um novo modelo, (iii) a população do modelo, usando componentes de domínio, como fonte de necessidades específicas, (iv) teste, (v) implantação, e (vi) a manutenção do modelo. Embora, esse trabalho considere necessidades específicas de domínio, ele não aborda em detalhes a customização para um domínio específico das melhores práticas a partir de SPCMMs genéricos.

Mettler **Erro! Fonte de referência não encontrada.** realiza uma análise mais profunda sobre os fundamentos dos modelos de maturidade do processo, colocando as principais fases descritas em **Erro! Fonte de referência não encontrada.**, sob uma perspectiva de *design science research*. Neste contexto, as fases são comparadas em uma visão de usuário de modelos de maturidade, indicando a necessidade de métodos mais formais e estudos mais aprofundados.

Matook e Indulska **Erro! Fonte de referência não encontrada.** propõem uma abordagem baseada em QFD (*Quality Function Deployment*) para o desenvolvimento de modelos de processos de referência, que incorpora a voz de usuários do modelo e apresenta uma medida para a sua qualidade. A abordagem também fornece um meio para gerenciar o desenvolvimento do modelo de referência, incluindo as seguintes fases: (i) definição do problema, (ii) análise de requisitos, (iii) coleta de informação, (iv) convenções e regras, (v) documentação; (vi) construção e design e, (vii) avaliação. Esta pesquisa apresenta os primeiros passos para o desenvolvimento de um apoio mais sistemático para o desenvolvimento deste tipo de modelos. No entanto, seu foco principal é na construção do modelo, sem nenhuma cobertura prevista de seu uso e evolução. Da mesma forma, não é fornecido um suporte metodológico detalhado para a customização do SPCMMs.

Becker, Knackstedt e Pöppelbuß **Erro! Fonte de referência não encontrada.** propõem um modelo geral de desenvolvimento de Modelos de Maturidade de Processos. Inicialmente os autores propõem oito requisitos para um modelo de referência de processos: (i) comparação com modelos existentes, (ii) desenvolvimento iterativo, (iii) avaliação do modelo, (iv) procedimento multi-metodológico, (v) identificação da relevância do problema, (vi) definição do problema, (vii) resultados publicados e (viii) documentação científica. A partir destes requisitos, diversos modelos obtidos da literatura são analisados. A principal conclusão dos autores é de que existe pouca documentação acerca de como este tipo de modelo é desenvolvido. Os autores propõem então um processo de desenvolvimento de modelos que procura cobrir todos os requisitos definidos. Este processo tem um importante foco na produção da documentação científica referente ao SPCMM a ser desenvolvido, dada a carência de documentação percebida no estudo realizado. Entretanto, o trabalho não aborda a questão da evolução do modelo após a sua publicação.

Uma revisão sistemática de literatura apresentada em **Erro! Fonte de referência não encontrada.** e um *survey* realizado com autores de modelos apresentado em **Erro! Fonte de referência não encontrada.** também indicam que existe pouco suporte sistemático para o desenvolvimento de SPCMMs sendo utilizado na prática, sendo que a maior parte é desenvolvida de maneira *ad hoc* **Erro! Fonte de referência não encontrada.** Também se observa que, na maioria das publicações que se propõem a customizar SPCMMs para um domínio específico, não são informados detalhes sobre como o modelo foi desenvolvido **Erro! Fonte de referência não encontrada.**

Neste contexto, este projeto trata do desenvolvimento de um método para customização de SPCMMs, desenvolvido com base na análise de como customizações de SPCMMs têm sido realizadas, integrando procedimentos de desenvolvimento de normas e também técnicas de EC. Desta forma, a principal contribuição deste trabalho consiste na consolidação de um processo para a customização de SPCMMs genéricos a domínios específicos de desenvolvimento de software.

## II. Objetivos e Justificativa

### A. Objetivo Geral:

Consolidar um método para customização de modelos de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software.

### B. Objetivos específicos

**O1** - Analisar o processo de desenvolvimento de modelos customizados de capacidade/maturidade de processos de software.

**O2** - Identificar as técnicas e ferramentas apropriadas para o desenvolvimento de modelos customizados de capacidade/maturidade de processos de software.

**O3** - Consolidar o método para customização de modelos de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software por meio da implementação das melhorias previstas na sua avaliação anterior.

**O4** - Validar o método desenvolvido por meio de um estudo empírico no desenvolvimento de um modelo de capacidade/maturidade de processos de software para o domínio de convergência digital.

**C. Justificativa:**

Modelos de Capacidade/Maturidade de Processo de Software (*Software Process Capability/Maturity Models - SPCMMs*) são repositórios de melhores práticas utilizados como referência na avaliação e melhoria da qualidade de processos de software. Cada domínio de software, no entanto, apresenta necessidades particulares que têm levado à customização dos SPCMMs.

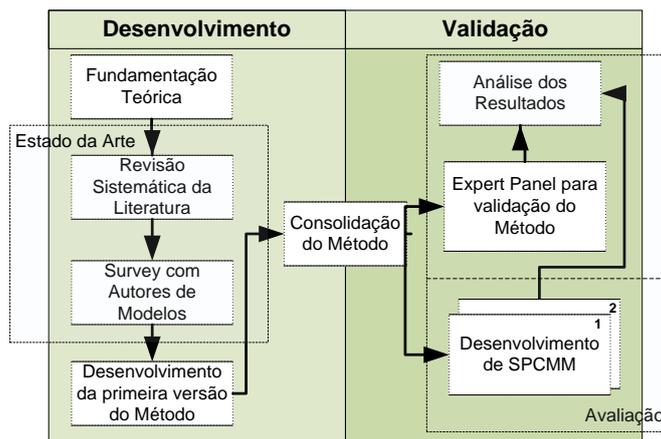
Apesar da sua importância para a engenharia de software, muitas das iniciativas de customização de SPCMMs têm sido realizadas de forma não sistemática, levando a diversas críticas em relação à validade destes modelos. Como não existe atualmente suporte sistemático a este processo, no ano de 2010 foi iniciado o desenvolvimento de um método para customização de SPCMMs para domínios específicos de desenvolvimento de software.

**III. Metodologia de Execução**

O desenvolvimento do método apresentado neste projeto foi realizado sob uma abordagem indutiva seguindo os passos apresentados na figura 1. Iniciou-se o desenvolvimento com uma fundamentação teórica abrangendo os conceitos e abordagens de modelagem de processos. A fim de obter o estado da arte no que diz respeito à forma como SPCMMs específicos de domínio são desenvolvidos, foi realizada uma revisão sistemática da literatura **Erro! Fonte de referência não encontrada.** Como resultado, foram identificados 52 SPCMMs, mas, a maioria deles não apresentava detalhes de como foram desenvolvidos. Para tanto, foi realizada, numa segunda etapa, um *survey* entre os autores dos modelos **Erro! Fonte de referência não encontrada.**, com o objetivo de obter informações adicionais sobre como esses modelos foram desenvolvidos.

Com base nestes resultados, foi desenvolvida a primeira versão do método usando as experiências de desenvolvimento de SPCMMs observadas na literatura **Erro! Fonte de referência não encontrada.** e as descrições fornecidas por autores de modelos **Erro! Fonte de referência não encontrada.**, as fases e etapas do processo de desenvolvimento de normas internacionais da ISO **Erro! Fonte de referência não encontrada.** e processo de desenvolvimento de normas da IEEE **Erro! Fonte de referência não encontrada.**, PRO2PI-MFMOD **Erro! Fonte de referência não encontrada.**, e do framework de processo de desenvolvimento de modelos de maturidade proposto em **Erro! Fonte de referência não encontrada.** Em paralelo ao desenvolvimento do método foram realizados dois estudos de caso em que versões intermediárias do método foram utilizadas para customizar SPCMMs (Modelo SaaS **Erro! Fonte de referência não encontrada.** e Medi SPICE **Erro! Fonte de referência não encontrada.**). O *feedback* destas aplicações foi utilizado para melhorar e completar o método.

Por fim, o método foi validado por meio de um *Expert Panel* onde especialistas em desenvolvimento de SPCMMs foram convidados a avaliar a adequação do método para a customização de SPCMMs.



Etapas no Desenvolvimento do método

**IV. Resultados Obtidos**

Diversos resultados positivos foram obtidos pela realização do presente projeto. Nesta seção são apresentados os principais resultados, conforme ordem e formato definidos na chamada.

**A. Processos que contribuem para a melhoria da qualidade de software descritos e institucionalizado**

O principal resultado deste projeto é um método para customização de SPCMMs elaborado e documentado na forma de um relatório técnico do INCoD<sup>4</sup> disponível em **Erro! Fonte de referência não encontrada..** O relatório técnico descreve como pode ser realizada a customização de um SPCMM e também apresenta a validação e resultados observados na sua avaliação e da sua utilização.

Por entender a customização de SPCMMs como um processo inerente de aquisição de conhecimento, o método é composto por cinco fases (figura 2): (i) Identificação do Conhecimento, (ii) Especificação do Conhecimento, (iii) Refinamento do Conhecimento, (iv) Uso de Conhecimento e (v) Evolução do Conhecimento. Cada fase é composta por um conjunto de atividades que não são necessariamente executados sequencialmente.

Fase 1: Identificação do Conhecimento	Fase 2: Especificação do Conhecimento	Fase 3: Refinamento do Conhecimento	Fase 4: Uso do Conhecimento	Fase 5: Evolução do Conhecimento
1.1 - Familiarizar-se com o Domínio	2.1 - Desenvolver a arquitetura do modelo	3.1 - Avaliar o modelo <i>draft</i>	4.1 - Fornecer suporte ao uso do modelo	5.1 - Administrar as solicitações de mudança
1.2 - Identificar fontes de conhecimento	2.2 - Analisar e integrar modelos relacionados	3.2 - Consolidar o modelo	4.2 - Validar o modelo em uso	5.2 - Confirmação, revisão ou rejeição
1.3 - Definir escopo e objetivos	2.3 - Desenvolver o modelo <i>draft</i> - dimensão de processo	3.3 - Votar o modelo consolidado		
1.4 - Formalizar o grupo de trabalho	2.4 - Desenvolver o modelo <i>draft</i> - dimensão de capacidade/maturidade	3.4 - Aprovar o modelo		
		3.5 - Publicar		

Fases e Atividades da Abordagem

## B. Produtos de software gerados

Durante o desenvolvimento deste método foi também implementada uma plataforma de software para suportar parcialmente a sua execução chamada de MCM – *Maturity & Capability Model Repository*, conforme relatado em **Erro! Fonte de referência não encontrada..** O sistema de software desenvolvido já está disponível em: <http://www.gqs.ufsc.br/mcm>. A principal funcionalidade do sistema MCM consiste em armazenar os SPCMMs que servem de base para o desenvolvimento de outros SPCMMs customizados. Maiores detalhes sobre as funcionalidades desenvolvidas podem ser encontradas na página do sistema ou em **Erro! Fonte de referência não encontrada..**

## C. Artigos publicados

Os seguintes artigos foram publicados como resultado direto do projeto executado:

- *Proposing an ISO/IEC 15504-2 Compliant Method for Process Capability/Maturity Models Customization* - Lecture Notes in Computer Science **Erro! Fonte de referência não encontrada..** Este artigo trata do alinhamento do método à norma ISO/IEC 15504.
- Um Método de Customização de Modelos de Capacidade/Maturidade de Processos de Software para Domínios Específicos - 25th Simpósio Brasileiro de Engenharia de Software (SBES), 2011 **Erro! Fonte de referência não encontrada..** Este artigo trata da descrição em alto-nível do método e a sua validação.
- *The LEGO Maturity & Capability Model Approach* - 5th World Congress on Software Quality 2011 **Erro! Fonte de referência não encontrada..** Este artigo trata da integração do método no escopo da abordagem LEGO de melhoria de processos.
- *FIRST: Common-sense Process Scopes for Starting a Process Improvement Program* - SPICE Conference 2012 **Erro! Fonte de referência não encontrada..** Este artigo apresenta uma proposta de escopo de processos a serem utilizados em alinhamento para a melhoria com o método proposto.
- *Building a Maturity & Capability Model Repository* - PROFES - 12th International Conference on Product Focused Software 2011 **Erro! Fonte de referência não encontrada..** Neste artigo é apresentado o repositório desenvolvido para armazenar os SPCMMs utilizados no método.

## D. Dissertações e/ou teses geradas

<sup>4</sup> Instituto Nacional de Ciência e Tecnologia para Convergência Digital

Como resultado deste projeto, finalizada durante o ciclo ao qual se refere este artigo, foi defendida uma tese de doutorado:

- Um Método de Aquisição de Conhecimento para Customização de Modelos de Capacidade Maturidade de Processos de Software, disponível em: **Erro! Fonte de referência não encontrada.**

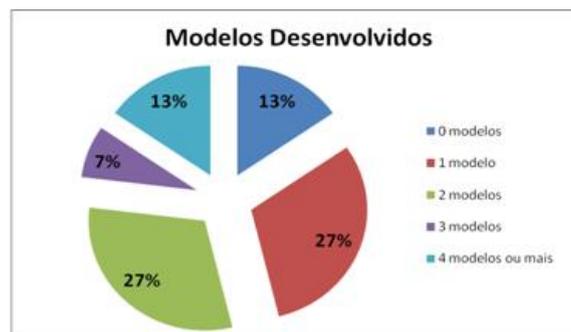
## V. Aplicabilidade dos Resultados

Os resultados do método foram validados por meio de estudos de caso que procuraram caracterizar a aplicabilidade do método em ambientes reais de desenvolvimento de SPCMMs e, por meio de observação e impressões subjetivas coletadas do seu uso prático. Neste sentido, um *Expert Panel* foi realizado procurando avaliar a adequação do método por meio da coleta de opinião de especialistas.

Um *Expert Panel* (painel de especialistas) é uma técnica baseada na opinião de especialistas, bastante utilizada na validação de novos métodos e técnicas **Erro! Fonte de referência não encontrada.** Ele consiste em reunir especialistas de determinada área de conhecimento de forma a obter a sua opinião em relação a algum aspecto do objeto de pesquisa. Para este estudo foram convidados 38 autores de SPCMMs identificados por meio de suas publicações na área, sendo que 15 especialistas participaram do painel, avaliando o método e respondendo o questionário, representando uma taxa de resposta de 39,5%.

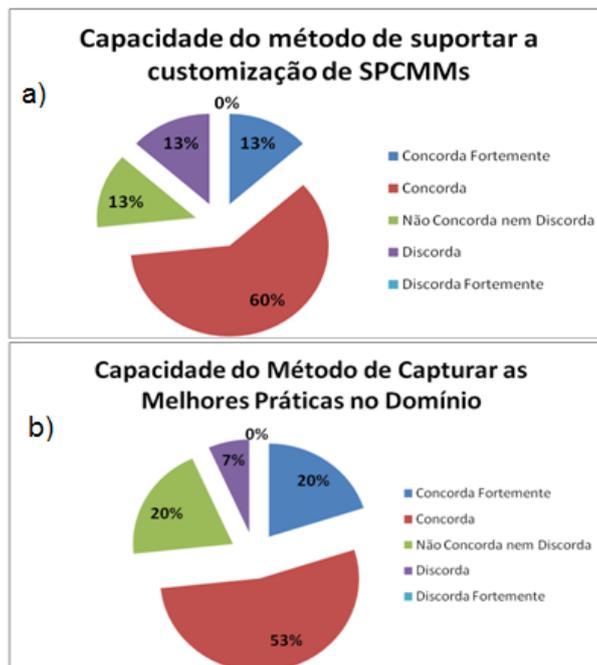
Analisando o perfil dos especialistas que participaram do *Expert Panel*, a maior parte (73% - 11) possui titulação de doutor enquanto 20% (3) possuem título de mestre. Com base nesses dados é possível afirmar que os participantes do painel possuem alta formação acadêmica em sua maioria.

Os participantes do painel foram também questionados acerca de quais SPCMMs eles estiveram envolvidos no desenvolvimento. Como resultado 23 modelos e normas diferentes foram citados pelos participantes, sendo a média geral, considerando modelos que tiveram mais de um participante envolvido, de 2,6 modelos desenvolvidos por participante. Com a exceção de 2 participantes (13%), todos os demais (13 - 87%) estiveram diretamente envolvidos na elaboração de pelo menos um modelo ou norma, sendo que dois participantes (13%) estiveram envolvidos na elaboração de mais de quatro modelos, conforme mostra a Figura 3.



Análise demográfica dos especialistas

Para atender ao objetivo de avaliação, foi coletada a impressão subjetiva dos especialistas sobre a capacidade do método de suportar a customização de SPCMMs por meio de uma afirmação no questionário com perguntas na forma de escala *Likert* de cinco pontos, variando de Concordo Fortemente (5) até Discordo Fortemente (1). Os percentuais das respostas são apresentados na Figura 4-a. Como resultado, 73% (11) concordam que o método fornece o suficiente para possibilitar a customização de SPCMMs e 13% (2) discordam que o método forneça o suficiente. Outros 13% (2) mantêm-se neutros e nenhum especialista discorda fortemente desta afirmação.



Principais resultados do *Expert Panel*

A pergunta mais importante do ponto de vista da utilidade do método na customização de melhores práticas, referindo-se à impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio, foi também coletada por meio de uma afirmação no questionário (pergunta 20) com perguntas na mesma forma de escala *Likert*. Conforme mostra a Figura 4-b, 73% (11) dos especialistas afirmam que o método é capaz de produzir SPCMMs que contenham as melhores práticas de desenvolvimento de software para o domínio, enquanto 7% (1) entendem que o método não tem essa capacidade e 20% (3) permaneceram neutros. Nenhum especialista discorda fortemente desta afirmação, o que levanta indícios de que o método é capaz de gerar SPCMMs que realmente contenham as melhores práticas para o domínio.

Na validação do método várias oportunidades de melhoria foram percebidas, especialmente considerando que esta pesquisa representa um primeiro esforço na direção de sistematizar e detalhar o processo de customização de SPCMMs. Da mesma forma, diversas são as limitações às quais a validação apresentada neste capítulo está sujeita, tais como as limitações típicas dos métodos e técnicas utilizados. Entretanto, após a validação realizada, é possível observar indicações de que o método desenvolvido atende ao seu objetivo na customização de SPCMMs

#### A. Ameaças à validade

Como qualquer resultado de pesquisa, a avaliação do método apresentada nesta seção por meio da realização do *Expert Panel* possui diversas limitações à sua validade.

Uma das principais limitações inerentes à realização de um *Expert Panel* é a pequena amostra utilizada. Isso se deve ao fato de que este tipo de técnica não visa cobrir uma amostra relevante da população-alvo em um estudo em largura do objeto, mas sim, realizar um estudo em profundidade do objeto a partir de um grupo reduzido de especialistas adequadamente qualificados para tal. Entretanto, pode-se afirmar que a amostragem aplicada, apesar de pequena, é bastante focada, pois foram convidados somente especialistas com reconhecida publicação relevante na área e com conhecimento e experiência prática no desenvolvimento de SPCMMs. Ainda assim, a pequena amostra reduz a possibilidade de generalização dos resultados e conclusões assinaladas.

Outra limitação importante da pesquisa realizada no *Expert Panel* diz respeito à taxa de respostas. Dos 38 especialistas convidados, somente 39,5% (15) responderam ao questionário. Esta baixa taxa de respostas pode ser devida a diversos fatores, como o período do ano no qual a pesquisa foi realizada (novembro-dezembro) e considerando o esforço considerável que os especialistas assumiram já que a sua participação na pesquisa necessitou além de responder o questionário a leitura do método na íntegra (79 páginas). Entretanto, uma taxa de retorno média de 55,6%, com variação de +/-19,7% é considerada normal neste tipo de pesquisa **Erro! Fonte de referência não encontrada.**

A amostragem realizada somente com pesquisadores que têm publicação científica relevante também pode limitar as conclusões obtidas. Essa limitação gera clara tendência ao ambiente acadêmico em detrimento da indústria de desenvolvimento de software.

## VI. Características Inovadoras

As pesquisas anteriormente realizadas pelo grupo responsável pela elaboração deste projeto **Erro! Fonte de referência não encontrada.** apontam que, apesar da existência de algumas iniciativas de procedimentos metodológicos para o desenvolvimento de SPCMMs, estas não têm sido seguidas e também não cobrem todas as fases e atividades normalmente necessárias no desenvolvimento de um SPCMM. Também não são abrangidas as especificidades de customização de um SPCMM para domínios específicos de desenvolvimento de software. Neste sentido o método proposto é inovador, pois procura abranger todas as fases e atividades realizadas na customização de SPCMMs de forma a estabelecer melhores práticas de processo de software que adequadamente atendam às necessidades de desenvolvimento de software naquele domínio.

O método desenvolvido neste projeto aborda uma tendência atual de customização de SPCMMs para domínios específicos de desenvolvimento de software, tendência para a qual existe pouco suporte metodológico, conforme abordado na introdução deste artigo.

## VII. Conclusão e Perspectivas Futuras

Este artigo apresenta a consolidação de um Método de Customização de SPCMMs para Domínios Específicos. Os trabalhos relacionados, a estrutura e conteúdo do método são apresentados, bem como a sua validação.

Os primeiros resultados obtidos a partir dos estudos de caso aplicando versões iniciais do método no desenvolvimento de dois SPCMMs para o domínio de SaaS **Erro! Fonte de referência não encontrada.** e Medi SPICE **Erro! Fonte de referência não encontrada.** e os resultados da validação via *Expert Panel*, levantam uma primeira indicação de que o método é adequado para a customização de SPCMMs.

Com o desenvolvimento deste método espera-se apoiar a pesquisa na customização de SPCMMs, contribuindo também no desenvolvimento de SPCMMs de qualidade, que contenham as melhores práticas alinhadas às necessidades de desenvolvimento de software para cada domínio específico.

A versão atual do método é considerada um ponto de partida no sentido de sistematizar a customização de SPCMMs para domínios específicos, que será aperfeiçoado e detalhado com base em sua utilização prática.

### Agradecimentos

Os autores agradecem a todos os participantes do *Expert Panel* pelo esforço empregado. Especialmente agradecemos aos pesquisadores Fergal McCaffery e Luigi Buglione pela participação nessa pesquisa e Timo Varkoi e Sarah Beecham pela sua revisão detalhada do método.

Os autores também agradecem o apoio financeiro do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) e ao Programa Institutos Nacionais de Ciência e Tecnologia (CNPq/MCT).

### Referências Bibliográficas

- Salviano, C. F., Figueiredo, A. M. C. M. (2008) "Unified Basic Concepts for Process Capability Models," in 20th Int Conf on Sw. Eng. and Knowledge Eng. SEKE, San Francisco, USA.
- SEI – Software Engineering Institute (2010) "CMMI for Development, Version 1.3," Carnegie Mellon University/Software Engineering Institute, Carnegie Mellon University, Pittsburgh.
- ISO - International Organization for Standardization / IEC - International Electrotechnical Commission ISO/IEC (2008) "ISO/IEC 15504: Information Technology Process Assessment - Part 1 to 5," International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC), ISO/IEC International Standard.
- Beecham, S., Hall, T., Rainer, A. (2005) "Defining a Requirements Process Improvement Model," *Software Quality Journal*, v. 13, n. 3, Springer.
- McCaffery F., Dorling A., Casey V. (2010) "Medi SPICE: An Update." 10th International Conference on Software Process. Improvement and Capability dEtermination, Pisa, Italy.
- Wangenheim, C. G., Hauck, J. C. R., McCaffery, F., Wangenheim, A. (2010) "Creating Software Process Capability/ Maturity Models," *IEEE Software*, v. 27, n. 4.
- Cass, A., Volcker, C. (2000) "SpiCE for SPACE: A method of Process Assessment for Space Projects," in International SPICE Conference.
- Torgersson, J., Dorling, A. (2002) "Assessing CBD - What's the Difference?," in 28 th Euromicro Conference, Dortmund, Germany.
- April, A., Abran, A., Dumke, R. (2004) "SMCMM Model to Evaluate and Improve the Quality of the Software Maintenance Process," in 8th Euromicro Working Conference on Software Maintenance and Reengineering, IEEE Computer Society.
- Matook, S., Indulska (2009) "Improving the quality of process reference models: A quality function deployment-based approach," *Decision Support Systems*, vol. 47.
- ISO International Organization for Standardization. (2010) How ISO develops standards. [Online]. [http://www.iso.org/iso/how\\_iso\\_develops\\_standards](http://www.iso.org/iso/how_iso_develops_standards)
- IEEE Computer Society (2010) IEEE Standards Development Process. [Online]. <http://standards.ieee.org/resources/development/index.html>

- Goodman, B. D., Goldman, S. N. (2007) "Freeing creativity by understanding the role of best practices," 19th IEEE International Engineering Management Conference, Austin
- Graupner S., Motaahari-Nezhad H. R., Singhal S., Basu, S. (2009) "Making processes from best practice frameworks actionable," 13th Enterprise Distributed Object Computing Conference Workshops, Auckland.
- Schreiber, G., Akkermans, H., Shadbolt, W. V., Vielinga, B. (2000) "Knowledge Engineering and Management – The CommonKADS Methodology," USA: The MIT Press.
- Bruin T., Rosemann, M. (2005) "Understanding the main phases of developing a maturity assessment model," in 16th Australasian Conference on Information Systems, Sydney, Australia.
- Salviano, C. F., Zoucas, A., Silva, J. V. L., Alves, A. M., Wangenheim, C. G., Thiry, M. (2009) "A Method Framework for Engineering Process Capability Models," in 16th European Systems and Software Process Improvement and Innovation, Alcalá, Spain.
- Mettler, T. (2009) "A Design Science Research Perspective on Maturity Models in Information Systems," Universität St. Gallen, St. Gallen, Switzerland, Technical Report BE IWI/HNE/03.
- Matook S., Indulska, M. (2009) "Improving the Quality of Process Reference Models: A Quality Function Deployment-Based Approach". *Decision Support Systems*, v. 47.
- Becker J., Knackstedt R., Pöppelbuß J. (2009) "Developing Maturity Models for IT Management - A Procedure Model and its Application". *Journal Business & Information Systems Engineering*, Gabler Verlag, v. 1, n. 3
- Wangenheim, C. G., Hauck, J. C. R., Salviano, C. F., Wangenheim, A. (2010) "Systematic Literature Review of Software Process Capability/Maturity Models," in Spice Conference 2010, Pisa, Italy.
- Schreiber G., Wieling B. J. (1998) "Knowledge Model Construction," in 11th Workshop on Knowledge Acquisition, Modeling and Management, Voyager Inn, Banff, Alberta, Canada.
- Cancian, M.; Hauck J. C. R.; Wangenheim, C. G.; Rabelo R. (2010) "Discovering Software Process and Product Quality Criteria in Software as a Service," *Lecture Notes in Computer Science*, v. 6156.
- Medi SPICE (2011) "Portal do projeto Medi SPICE," [Online] <http://medispice.ning.com>
- Lakatos E. M., Marconi M. A. (1995) "Metodologia do Trabalho Científico". São Paulo: Atlas.
- Acuña A. A., Ferre X., López M., Mate L. (2000) "The Software Process: Modeling, Evaluation and Improvement". *Handbook of Software Engineering and Knowledge Engineering*. Argentina: World Scientific Publishing Company.
- Hua, J. (2008) "Study on Knowledge Acquisition Techniques". 2nd International Symposium on Intelligent Information Technology Application, Shanghai.
- Hauck J. C. R., Wangenheim C. G., Wangenheim, A. (2011) "A Knowledge Engineering Based Method for SPCMMs Customization". *Relatório Técnico RT GQS 1 0*. [online] [http://www.inf.ufsc.br/~jeanhauck/method/RT\\_GQS\\_01\\_SPCMMs\\_Dev\\_Method\\_v\\_1\\_0.pdf](http://www.inf.ufsc.br/~jeanhauck/method/RT_GQS_01_SPCMMs_Dev_Method_v_1_0.pdf)
- Hauck J. C. R., Wangenheim C. G., McCaffery F., Buglione L. (2011) "Proposing an ISO/IEC 15504-2 Compliant Method for Process Capability/Maturity Models Customization". 12<sup>th</sup> International Conference on Product Focused Software Development and Process Improvement. Torre Canne, Itália (aceito para publicação).
- Basili V. R., Caldiera G., Rombach H. D. (1994) "The Goal Question Metric Approach". *Encyclopedia of Software Engineering*, v. 1, John Wiley & Sons.
- Beecham, S. Hall, T., Rainer, A. (2003) "Building a Requirements Process Improvement Model," Faculty of Engineering and Information Sciences, University of Hertfordshire, Hertfordshire, Technical Report 378.
- GAO - United States General Accounting Office. (1990) "Case Study Evaluations". Technical Report GAO/PEMD-91-10.1.9. Program Evaluation and Methodology Division [online] [http://www.gao.gov/special.pubs/10\\_1\\_9.pdf](http://www.gao.gov/special.pubs/10_1_9.pdf)
- Cancian, M. "Process Reference Model for SaaS" Universidade Federal de Santa Catarina - Florianópolis/Brasil. [online] <http://www.gsigma.ufsc.br/~cancian/guide/>
- European Council. (2007) "Council Directive 2007/47/EC (Amendment)." Official Journal of The European Union, Luxembourg [online] <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2007:247:0021:0055:en:PDF>
- Dyba, T. (2000) "An instrument for measuring the key factors of success in software process improvement," *Empirical Software Engineering*, v. 5, n. 4.
- Emam, E. K., Madhavji, N. H. (1996) "Does organizational maturity improve quality?" *IEEE Software*, v. 13, n. 5.
- Powell, C. (2003) "The Delphi technique: myths and realities," *Journal of Advanced Nursing*, v. 41, n. 4, Blackwell Publishing.
- Cook D. A., Beckman T. D. (2006) "Current Concepts in Validity and Reliability for Psychometric Instruments: Theory and Application," *The American Journal of Medicine* v. 119, n. 166.
- Gliem J. A., Gliem R. R. (2003) "Calculating, Interpreting, and Reporting Cronbach's Alpha Reliability Coefficient for Likert-Type Scales". In *Midwest Research-to-Practice Conference in Adult, Continuing, and Community Education*.
- Wessa, P. (2011) "Cronbach alpha (v1.0.1) in Free Statistics Software (v1.1.23-r6)". Office for Research Development and Education. [online] [http://www.wessa.net/rwasp\\_cronbach.wasp/](http://www.wessa.net/rwasp_cronbach.wasp/)
- Goldenson, D. R., Emam, K. E., Herbsleb, J., Deephouse, C. (1999) *Empirical Studies of Software Process Assessment Methods*. In *Elements of Software Process Assessment & Improvement*, Khaled El Emam & Nazim H. Madhavji (Eds), Wiley IEEE.
- Cook D. A., Beckman T. D. (2006) "Current Concepts in Validity and Reliability for Psychometric Instruments: Theory and Application". *The American Journal of Medicine* v. 119, n. 166.
- Litwin, M. (1995) "How to Measure Survey Reliability and Validity". Sage Publications.
- Baruch, Y. (1999) "Response Rate in Academic Studies – A Comparative Analysis," *Human Relations*, v. 52, n. 4.
- Bruin T., Rosemann, M. (2007) "Using the Delphi Technique to Identify BPM Capability Areas," 18th Australasian Conference on Information Systems.
- Hauck J. C. R., Wangenheim C. G., Wangenheim A. (2011) "Método de Aquisição de Conhecimento para Customização de Modelos de Capacidade/Maturidade de Processos de Software" *Relatório Técnico INCoD –*

Nº 002/2011 – P – GQS. Disponível em: [http://www.incod.ufsc.br/wp-content/uploads/2011/06/INCoD\\_Relatorio\\_Tecnico\\_11.002.P.GQS\\_.pdf](http://www.incod.ufsc.br/wp-content/uploads/2011/06/INCoD_Relatorio_Tecnico_11.002.P.GQS_.pdf)

- Hauck, J. C. R., Wangenheim C. G., McCaffery F.; Buglione L. (2011) "Proposing an ISO/IEC 15504-2 Compliant Method for Process Capability/Maturity Models Customization". Lecture Notes in Computer Science, v. 6759, p. 44-58.
- Hauck J. C. R., Wangenheim, C. G. (2011) "Um Método de Customização de Modelos de Capacidade/Maturidade de Processos de Software para Domínios Específicos". 25th Brazilian Symposium on Software Engineering (SBES), 2011, São Paulo. Anais do 25th Brazilian Symposium on Software Engineering (SBES), 2011. p. 1-10
- Buglione L., Wangenheim C. G., Hauck J. C. R. McCaffery F. (2011)"The LEGO Maturity & Capability Model Approach". 5th World Congress on Software Quality, 2011, Shanghai. Proceedings, 2011. v. 1. p. 10-18.
- Buglione L., McCaffery F., Hauck J. C. R., Wangenheim C. G. (2011) FIRST: Common-sense Process Scopes for Starting a Process Improvement Program. SPICE Conference, 2012, Palma de Mallorca. Anais, 2012. v. 1. p. 10-18.
- Wangenheim C. G., Hauck J. C. R., Buglione L., McCaffery F. Lacerda T., Cruz R. F. V. (2011) "Building a Maturity & Capability Model Repository". PROFES - 12th International Conference on Product Focused Software. Development and Process Improvement, 2011, Torre Canne. Proceedings, 2011. v. 1. p. 10-18.
- Hauck J. C. R. "Um Método De Aquisição De Conhecimento Para Customização De Modelos De Capacidade/Maturidade De Processos De Software". Program de Pós-Graduação em Engenharia e Gestão do Conhecimento. Universidade Federal de Santa Catarina ,Tese, 2011. Disponível em: <http://btd.egc.ufsc.br/?p=931>

## AgileCCPM – Uma integração entre Scrum e Corrente Crítica

Daniel Tadeu Martínez C. Branco<sup>1,2</sup>, Rafael Prikladnicki<sup>3</sup>, Tayana Conte<sup>1</sup>

<sup>1</sup>USES - Grupo de Usabilidade e Engenharia de Software  
Instituto de Computação – Icomp (UFAM)  
Manaus, AM – Brasil

<sup>2</sup>Processamento de Dados Amazonas S.A. (PRODAM)  
Manaus, AM – Brasil

<sup>3</sup>Faculdade de Informática (FACIN) - PUCRS  
Porto Alegre, RS – Brasil

dmartinez@prodam.am.gov.br, rafaelp@pucrs.br, tayana@icomp.ufam.edu.br

**Abstract.** *Regular deliveries of value to customer and use of optimized schedules are essential in software development and can directly influence the outcome of an organization. The framework of Scrum has been used for, among other benefits, delivering value to the customer as soon as possible. In parallel, the theory of constraints, through the method of Critical Chain, has been employed in the pursuit of optimization of project schedules. This research proposes a method to integrate Scrum with the method of Critical Chain, aimed at advancing the product requirements and encourage the development of innovations.*

**Resumo.** *Entregas constantes de valor ao cliente e o uso de cronogramas otimizados são essenciais no desenvolvimento de software e podem influenciar diretamente no resultado de uma organização. O framework do Scrum tem sido utilizado para, entre outros benefícios, entregar valor mais cedo para o cliente. Em paralelo, a teoria das restrições, através do método da Corrente Crítica, tem sido empregada na busca da otimização de cronogramas de projeto. Esta pesquisa propõe um método para integração de Scrum com o método da Corrente Crítica, visando a antecipação de requisitos e o incentivo ao desenvolvimento de inovações.*

### 1. Introdução

Quando consideramos a necessidade de entrega rápida de valor aos usuários, as metodologias ágeis apresentam-se como uma solução pelo foco voltado para velocidade, adaptabilidade e qualidade [HIGHSMITH, 2010]. Estas variáveis podem ser implementadas através de ciclos iterativos de desenvolvimento, permitindo que o software seja entregue de maneira gradativa e o aprendizado da equipe seja construído, possibilitando a inovação [DYBA, 2010]. Estas metodologias, mesmo sendo adaptativas, possuem etapas de planejamento que permitem avaliar e estimar os requisitos a serem desenvolvidos [COHN, 2010].

As estimativas em projetos estão sujeitas a riscos durante a execução [COHN, 2010], induzindo a equipe a incluir margens de segurança e a assumir os piores casos como estimativas reais. Um dos problemas da utilização destas margens é que são atribuídas para cada atividade e não para o projeto como um todo, impactando no prazo final do projeto [LEACH, 2005]. A má utilização destas margens de segurança pode ser prejudicial para o andamento do projeto e, em equipes ágeis que possuem potencial de inovação [DYBA, 2010], para o não desenvolvimento de idéias inovadoras.

Em projetos ágeis de software, a identificação das margens de segurança pode permitir que este tempo seja mais bem aproveitado, por exemplo, na antecipação de requisitos do produto ou no desenvolvimento de inovações propostas pelo grupo. O desenvolvimento antecipado de requisitos pode permitir que um produto mais completo esteja disponível no mercado. De forma complementar, o desenvolvimento de inovação pode permitir o aumento no valor agregado ao produto, além de promover a motivação da equipe de desenvolvimento [SHARP *et al.*, 2009].

Este projeto propõe um método para integração da Corrente Crítica [GOLDRATT, 2006] com *Scrum* [COHN, 2009]. Como efeito da integração, espera-se otimizar os planejamentos ágeis, mediante a eliminação das margens de segurança das atividades. Ao longo do projeto houve uma mudança no seu escopo, o que inviabilizou a realização de todas as etapas previstas e o alcance aos objetivos integrais definidos.

O restante deste artigo está estruturado da seguinte forma. A seção 2 expõe os objetivos e a justificativa deste projeto. A seção 3 apresenta a metodologia utilizada para sua execução. A seção 4 discorre sobre os resultados encontrados. A seção 5 apresenta a aplicabilidade dos resultados. A seção 6 exhibe as características inovadoras desta pesquisa. Por fim, a seção 7 conclui o artigo.

## 2. Objetivos e Justificativa

Se por um lado o uso de metodologias ágeis favorece a adaptabilidade e o incremento de inovações [DYBA, 2010], por outro lado o gerenciamento de projetos permite maior previsibilidade do projeto, uma vez que seu cenário é antecipadamente definido. Por exemplo, o gerenciamento de tempo em projetos prevê variáveis como duração de atividades, recursos necessários e técnicas de compressão. Técnicas de compressão são utilizadas para antecipar a data de conclusão de uma fase e/ou projeto sem que existam alterações no seu escopo [PMI, 2008]. Um dos métodos utilizados para otimizar cronogramas é o método da Corrente Crítica [GOLDRATT, 2006].

O projeto tem como objetivo a definição de um processo de desenvolvimento de software que integre Scrum com o método da Corrente Crítica, permitindo a otimização de cronogramas e criação de uma estrutura que favoreça o desenvolvimento de inovações. Este processo, com papéis e atividades definidos, deverá ser avaliado experimentalmente, para que possa ser verificada a validade da proposta em um contexto de desenvolvimento de software.

## 3. Metodologia de execução

Para propor o método de integração da Corrente Crítica com o *Scrum*, a metodologia seguida por esta pesquisa tem por base uma abordagem baseada em evidências, como as propostas por Mafra *et al.* (2006) e Shull *et al.* (2001). Inicialmente são executados estudos secundários que fornecem a base para a definição da nova tecnologia. Após a proposta inicial da tecnologia, são realizados estudos experimentais para sua avaliação e evolução, como pode ser visto na Figura 1.

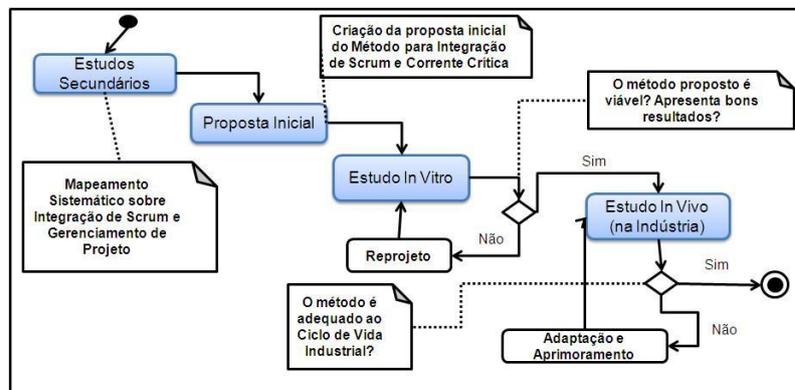


Figura 2 - Metodologia da Pesquisa baseada em Mafra *et al.* (2006)

#### 4 Resultados obtidos Estudos Secundários

Na primeira etapa da pesquisa foi realizado um mapeamento sistemático, conforme discutido em (Branco, 2011). Este estudo secundário permitiu identificar a ausência de abordagens experimentais que considerem a otimização de cronogramas ágeis, indicando o caráter inovador na integração entre a Corrente Crítica e Scrum.

##### Modelo conceitual do processo *AgileCCPM*

O método *AgileCCPM* propõe que a Corrente Crítica seja aplicada dentro de um *sprint*, uma vez que o *Scrum* prevê etapas de planejamento e decomposição das *user stories*. Decompostos os requisitos dentro do *sprint* deverão ser identificadas as margens de segurança para a formação dos *buffers*.

Além do *buffer* de cronograma, o *AgileCCPM* utiliza o conceito de *Feature Buffer* [COHN, 2010] em que requisitos são priorizados em caso de antecipação de atividades. A partir dos requisitos previstos pelo usuário, ao longo do projeto surgem possibilidades de inovação que podem agregar valor ao projeto. Com base nesta premissa, o *AgileCCPM* cria um *Innovation Backlog*, o qual também será considerado no *Feature Buffer*, caso, graças ao adiantamento do cronograma, haja tempo para o desenvolvimento de uma inovação no *sprint*. A Figura 2 representa o modelo conceitual do *AgileCCPM*.

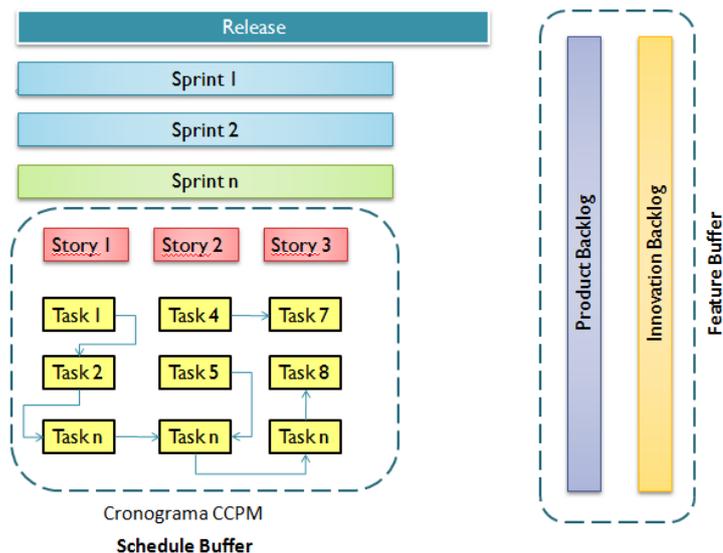
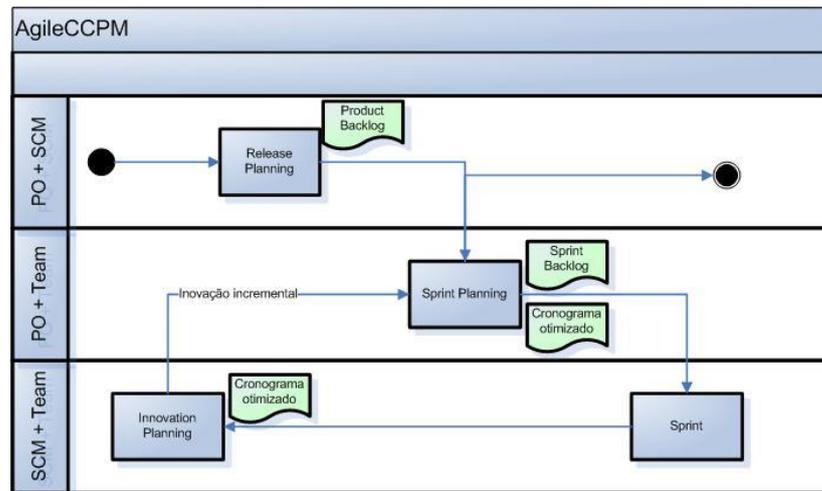


Figura 3 - *AgileCCPM*: Modelo conceitual

A utilização do *Feature Buffer* baseado em uma lista de inovações e de requisitos da aplicação (*Innovation Backlog* e *Product Backlog*, respectivamente) permitirá que o *Schedule Buffer* possa ser utilizado ou para antecipação do projeto ou para o desenvolvimento de inovações no produto. Assim, uma fila (*Product Backlog*) garantirá que o produto seja entregue e possa garantir competitividade no mercado, enquanto a outra (*Innovation Backlog*) pode permitir o destaque no mercado e uma plena diferenciação do produto frente à concorrência. A Figura 3 descreve o processo *AgileCCPM* considerando os papéis previstos no Scrum (*Scrum Master*, *Product Owner* e time).



**Figura 4 - AgileCCPM: Processo**

### **Estudo Experimental para o processo AgileCCPM**

Visando validar experimentalmente o processo proposto, realizou-se um estudo *in vitro* com 19 alunos de graduação do curso de Ciência da Computação da Universidade Federal do Amazonas. Cada time era composto por quatro ou cinco pessoas, desempenhando os seguintes papéis: um Scrum Master e dois ou três membros, acompanhados por um pesquisador desempenhando o papel de Product Owner.

Dada a limitação de tempo, o estudo ocorreu ao longo de duas aulas totalizando 4 *sprints*. O escopo desta atividade foi definido previamente visando o desenvolvimento de *mockups* (protótipos de interface) e descrições de caso de uso, não contemplando o desenvolvimento de software em si.

Durante o experimento, os participantes foram divididos em 3 equipes utilizando o processo Scrum e 3 equipes com o processo *AgileCCPM*. Como resultado não foi identificada diferença significativa que denotasse a validade do processo proposto sobre o Scrum. Entretanto, foi possível identificar a presença de variáveis psicossociais afetando os resultados, o que motivou a mudança do escopo do projeto.

#### **Artigo publicado**

- O artigo “Mapeamento Sistemático sobre integração entre Scrum e gerenciamento de projeto tradicional” (BRANCO *et al.*, 2011) foi apresentado no IV Workshop de Gerenciamento de Projetos de Software (evento paralelo ao X Simpósio Brasileiro de Qualidade de Software), realizado em Curitiba, PR.

#### **4. Aplicabilidade dos resultados: Benefícios para a Indústria e Academia**

Mesmo tendo sido alterado o escopo do projeto foi possível identificar uma lacuna experimental no que diz respeito na integração entre Scrum e a Corrente Crítica. Apesar da existência desta integração no mercado, não existem indícios suficientes que indiquem a efetividade deste processo. Percebe-se então a necessidade de trabalhos adicionais que validem seu uso e permitam a utilização em larga escala na indústria, permitindo a integração entre adaptabilidade e previsibilidade.

#### **5. Características inovadoras**

Dentro do contexto acadêmico, dada a inexistência de estudos experimentais, o presente projeto possui caráter inovador devido a proposta de integração do Scrum com o Gerenciamento de Projetos. Além disso, as propostas existentes no mercado não consideram a utilização do *buffer* do projeto para o desenvolvimento da inovação dentro de equipes de software.

## 6. Conclusões e trabalhos futuros

O *AgileCCPM* é um processo que busca, em projetos de software, antecipar entregas e criar um ambiente que permita o desenvolvimento de inovações através da integração entre Scrum e Corrente Crítica. Estes objetivos, quando considerados no mercado, podem garantir vantagens competitivas frente a outros produtos. Mesmo com as etapas não tendo sido cumpridas em sua totalidade, como trabalho futuro o processo proposto deve ser refinado e, posteriormente, validado em experimentos *in vitro* e *in vivo*. Dessa maneira, pode ser dada uma contribuição acadêmica efetiva e permitir o uso adequado na indústria.

### Referências Bibliográficas

- Branco, D., Prikladnicki, R., Conte, T. (2011). Mapeamento sistemático sobre integração entre Scrum e Gerenciamento de Projetos Tradicional. IV Workshop de Gerenciamento de Projetos de Software .
- Dyba, T. (2010) The role of improvisation in Software Development. Apresentação realizada no IX Simpósio Brasileiro de Qualidade de Software.
- Highsmith, J. (2010) Agile Project Management – Creative Innovative Products. The Agile Software Development Series. Second Edition. Addison Wesley.
- Cohn, M. (2009) Succeeding with Agile: Software Development Using Scrum. Addison-Wesley.
- Cohn, M. (2010) Agile Estimating and Planning. Robert C. Martin Series. Prentice Hall.
- Goldratt, E.M. (2006) Corrente crítica. Editora Nobel.
- Leach, L. (2005) Critical Chain Project Management, Second Edition. Effective Project Management Series. Artech House.
- Mafra, S., Barcelos, R., Travassos, G. H. (2006) Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software. Anais do XX SBES.
- Project Management Institute (PMI). (2008) Project Management Body of Knowledge.
- Sharp, H., Baddoo, N., Beecham, S., Hall, T., Robinson, H. (2009) Models of motivation in software engineering. Information and Software Technology vol 51, 219–233.
- Shull, F., Carver, J., Travassos, G. H. (2001) An empirical methodology for introducing software processes. ACM SIGSOFT Software Engineering Notes, vol 26 (5), pp. 288-296.

## Institucionalizar Segurança no RUP-BNB

Francisco José Barreto Nunes, Joaquim Manoel Pitombeira de Oliveira

Banco do Nordeste do Brasil (BNB)  
Av. Pedro Ramalho, nº 5.700 – 60.743-902 – Fortaleza – CE – Brasil

{franzenunes,joaquimpo}@bnb.gov.br

**Abstract.** *Application security becomes the greater problem to be solved. An effective solution would be to approach software security actions focused on the improvement of development processes, considering the integration of security best practices from the beginning of a product's life cycle. Banking corporations, such as Banco do Nordeste (BNB), once being dependent on quality and reliable software, besides receiving regulatory inspection, need to institutionalize software security in the context of corporate software development processes. This paper proposes presenting not only the institutionalization of security practices in Requirements discipline of the development process RUP-BNB but also the project final results.*

**Resumo.** *A segurança de aplicações torna-se o grande problema a ser resolvido. Uma solução eficaz seria estruturar ações de segurança de software com foco na melhoria de processos de desenvolvimento, considerando a integração de melhores práticas de segurança desde o início do ciclo de vida do produto. Instituições bancárias, como o Banco do Nordeste (BNB), por dependerem de produtos de software com qualidade e confiáveis, além de se submeter à fiscalização de órgãos reguladores, precisam institucionalizar a segurança de software nos processos corporativos de desenvolvimento de software. O objetivo deste artigo é apresentar além da institucionalização de práticas de segurança no contexto da disciplina Requisitos do processo de desenvolvimento RUP-BNB, os resultados obtidos com o projeto piloto dessa institucionalização.*

### 1. Introdução

Instituições bancárias sofrem constantes fiscalizações por órgãos reguladores, sendo obrigadas a manter conformidade com leis e resoluções diversas, como a Resolução nº 3380 do Banco Central do Brasil [BCB 2006]. Diante de uma realidade de regulamentação rígida e de um cenário de fraudes e roubos onde informações precisam ser protegidas para reduzir riscos operacionais, órgãos fiscalizadores começam a destacar a importância de elevar o nível de proteção das informações manipuladas pelos aplicativos.

Não obstante, consultorias especializadas e profissionais de segurança sinalizam como oportuna e necessária a reorganização de práticas de desenvolvimento visando atender requisitos de segurança; o objetivo é produzir, conseqüentemente, aplicativos mais seguros. Além disso, estatísticas, como CERT.BR (2011), ou recentes pesquisas, como *Verizon Data Breach Investigations Report* (2011) (Figura 1), destacam a crescente ameaça à segurança de aplicações em virtude da exploração de vulnerabilidades encontradas em produtos de *software*.

Figure 28. Categories of affected assets by percent of breaches and percent of records

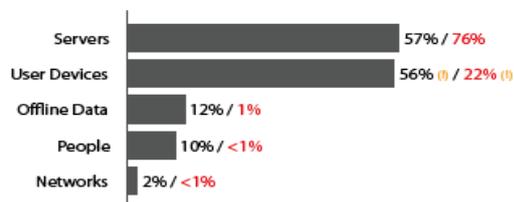


Figura 1. Verizon Data Breach Investigations Report

Isto é, o foco dos ataques ao software migrou da camada de rede para a camada de aplicação, constituída, por muitas vezes, de *software* inseguro, além de representar a menor distância entre o atacante e as informações corporativas. Os seguintes fatores podem contribuir para o número crescente de vulnerabilidades de *software* [McGraw 2006]:

- Maior interligação entre cada *software* e deles com a *Internet*;
- Produtos de *software* grandes, complexos e distribuídos;
- Necessidade de atender, rapidamente, às demandas de mercado (*time to market*).

Diante destas constatações, ratifica-se a importância, para o Banco do Nordeste, em investir na modelagem de segurança para *software*, seguindo proposta cujo objetivo, escopo e ações estejam alinhados com os objetivos estratégicos do Banco e considerando o tripé tecnologia, processos e pessoas para obter resultados eficazes.

## **2. Objetivos e Justificativa**

Percebe-se que o Banco do Nordeste (BNB), por ser responsável pela segurança das informações de clientes e de transações bancárias, beneficiar-se-á financeiramente além da melhoria da imagem por meio de ações inovadoras de segurança de *software*.

O projeto objetiva melhorar a segurança dos aplicativos a serem desenvolvidos no BNB por meio de ações de segurança realizadas na disciplina “Requisitos” do processo de desenvolvimento da metodologia RUP-BNB, preenchendo lacunas e trazendo inovação às iniciativas de segurança existentes. A partir de cada ação de segurança institucionalizada, pretende-se tratar e solucionar deficiências em aspectos da segurança de *software* dentro dessa disciplina, especificamente na identificação de necessidades e requisitos de segurança. Essa ação resulta em aplicativos mais seguros que ensejam menos falhas e interrupções e, por conseguinte, com melhor qualidade, satisfazendo as necessidades por *software* confiável.

## **3. Metodologia de Execução**

Para institucionalizar a segurança no RUP-BNB, a fim de obter os benefícios esperados com a melhoria do processo, o projeto foi dividido em três etapas de institucionalização.

### **3.1. Utilização do Processo RUP-BNB**

O processo de desenvolvimento de *software* do Banco do Nordeste, chamada RUP-BNB é baseado, principalmente, nos conceitos do *Capability Maturity Model Integration* (CMMI) (2006) e no RUP - *Rational Unified Process* (2011).

O processo é mantido como uma instância customizada do RUP para o BNB, disponível na Intranet e representa uma adaptação destes conceitos para as peculiaridades do Banco.

### **3.2. Primeira Etapa da Institucionalização**

A institucionalização iniciou com a organização de programa educativo sobre segurança de *software* para a conscientização de gestores e técnicos.

Preparou-se curso introdutório de segurança de *software* disponibilizado na Intranet do Banco com 24 horas/aula de duração. Nessa capacitação são apresentados os seguintes tópicos: Vulnerabilidade de *software*, Modelagem de ameaças, Controle de acesso, Testes de segurança e Requisitos de segurança.

A partir das informações coletadas e segundo necessidades que incorreriam demandas mais críticas de segurança e que precisariam ser tratadas prioritariamente, decidiu-se iniciar o processo de institucionalização da segurança pela disciplina Requisitos.

Inicialmente, definiram-se as atribuições e responsabilidades entre envolvidos com o projeto piloto e, em seguida, consensuou-se adotar abordagem de segurança, orientada pelo risco, que avaliasse vulnerabilidades, ameaças e impactos, bem como considerasse legislação, políticas, padrões e normas vigentes. A orientação pelo risco, ou gerência de risco, almeja à utilização eficaz dos recursos, com foco naquilo que é mais crítico. A principal dificuldade era identificar as necessidades de segurança e transformá-las em requisitos.

### **3.3. Segunda Etapa da Institucionalização**

Com a definição do escopo inicial do projeto, coletaram-se informações sobre Segurança de *Software* que compuseram cada ação a ser estruturada no processo. A principal fonte de informação foi a dissertação de mestrado “PASS - Processo de Apoio à Segurança de *Software*” [Nunes 2007]. Todavia, conteúdos dos livros [McGraw 2006], [Howard and LeBlanc 2002] e [Howard and Lipner 2006] também foram utilizados, além de consultas a boas práticas de mercado, normas nacionais e internacionais, como a ISO/IEC 15408 (2005). Este conjunto de novas ações de segurança para a disciplina Requisitos foi organizado e definido para validação da área de desenvolvimento de *software* a fim de compor a estrutura do RUP-BNB.

Entre as atividades e tarefas das disciplinas Modelagem de Negócio e Requisitos do RUP-BNB, identificaram-se aquelas que poderiam interagir com as ações de segurança propostas e suportá-las com artefatos de entrada. Considerou-se que as ações de segurança também poderiam utilizar como entrada alguns documentos

do projeto e ou do processo de negócio a ser suportado pelo sistema. Por conseguinte, cada ação de segurança também considerou, em seu relacionamento, as funções, as entradas e saídas.

A organização das ações demandou a elaboração de artefatos de apoio e, ainda, a adaptação de artefatos já existentes no RUP-BNB que passaram a considerar a segurança. Registrem-se as mudanças realizadas no documento “Especificação Suplementar” (Figura 2). Com as novas ações, elevou-se a qualidade dos aplicativos desenvolvidos ao prevenir que necessidades de segurança sejam esquecidas e que falhas de segurança sejam inseridas inadvertidamente.

Neste contexto, foi organizado o guia de modelagem para casos de abuso [McGraw 2006], incluindo exemplo completo de especificação, onde foi proposto caso de abuso utilizando dados reais de um projeto. As opiniões indicaram que o caso de abuso impactaria, sobremaneira, a produtividade das equipes, pois ainda não havia cultura organizacional e conhecimento necessário para a eficaz elaboração desses documentos.

<b>3. Confiabilidade</b>	<b>2</b>
3.1 Maturidade	2
3.2 Tolerância a Falhas	2
3.3 Recuperabilidade	2
3.4 Políticas de Backup	2
3.5 Disponibilidade	2
<b>4. Desempenho</b>	<b>2</b>
4.1 Tempo de Resposta	2
4.2 Eficiência	2
4.3 Capacidade	2
<b>5. Suportabilidade</b>	<b>2</b>
5.1 Testabilidade	2
5.2 Manutenibilidade	2
5.3 Extensibilidade	2
5.4 Adaptabilidade	2
5.5 Compatibilidade	2
5.6 Configurabilidade	2
5.7 Instabilidade	2
5.8 Localizabilidade	2
5.9 Padrão de Codificação	2
<b>6. Segurança</b>	<b>2</b>
6.1 Identificação e Autenticação	2
6.2 Controle de acesso	2
6.3 Integridade de dados	2
6.4 Gerenciamento de exceções	2
6.5 Controle de sessão	2
6.6 Canais de comunicação	2
6.7 Auditoria	2
6.8 Gerência de Configuração	2
6.9 Uso do correio eletrônico pelo aplicativo	2
6.10 Outros	2
<b>7. Restrições de Design</b>	<b>2</b>
7.1 <Restrição de Design 1>	2

Figura 2. Índice do documento “Especificação Suplementar”

Por conseguinte, foi estruturado documento de apresentação, das principais vulnerabilidades e ameaças das aplicações que funcionou, ainda, como ferramenta de conscientização e integração de equipes.

Dessa forma, paralelamente à estruturação das ações, foi organizada a Diretriz “Guia de Orientação de Segurança de Software - Falhas de Segurança em Aplicativos”, atingindo analistas, arquitetos e testadores cujo objetivo é explicar as principais vulnerabilidades de segurança de software existentes atualmente. Este documento foi criado a partir da identificação e organização de conteúdo sobre vulnerabilidades e ameaças de segurança de software, incluindo OWASP (2012), MITRE (2010) e WEBAPPSEC (2010).

Após vários encontros de trabalho sobre o conteúdo e apresentação das ações de requisitos de segurança na disciplina Requisitos, decidiu-se inseri-las na tarefa “Desenvolver especificações suplementares” cujos nomes e descrição são identificados na Tabela 1.

### 3.4. Terceira Etapa da Institucionalização

Com a validação da área de desenvolvimento de software, das ações e das mudanças de artefatos, foi planejado o projeto piloto cujo objetivo foi avaliar o impacto no projeto da execução dessas novas ações, com o consequente aperfeiçoamento da nova estrutura do RUP-BNB.

Tabela 1. Ações de segurança para disciplina Requisitos do RUP-BNB

Ação	Breve descrição
Identificar Necessidade de Segurança	As considerações relativas às necessidades de segurança devem perpassar as necessidades e potenciais funcionalidades do aplicativo, incluindo aspectos de usabilidade, performance, entre outros, bem como, aspectos de persistência, apresentação e regras de negócio. Os serviços afetados por incidentes, impactando confidencialidade, integridade e disponibilidade, entre outros, podem ser identificados e categorizados pelos direitos de nível de acesso do usuário.
Capturar Requisitos de Segurança	Categorizar requisitos de segurança pode envolver: a) Pesquisar técnicas de proteção de sistema e de prevenção de ameaça b) Determinar quais técnicas podem ser aplicadas à arquitetura do aplicativo Uma premissa é que os analistas verifiquem o documento de referência, contendo propostas de requisitos de segurança, na diretriz “Requisitos de Segurança” do RUP-BNB a fim de identificar requisitos adicionais.

Esclareceu-se que não haveria grande impacto na condução do projeto, pois haveria pequenas mudanças na forma de trabalho sem atrasos no cronograma, uma vez que o “engenheiro de segurança de *software*” responsabilizar-se-ia pelas ações de segurança até a completa assimilação pela equipe técnica. Em outras palavras, caberia aos envolvidos apenas esclarecer dúvidas, fazer sugestões e questionamentos, além de validar as contribuições de segurança.

Houve reuniões antecedentes ao início do piloto das quais participaram gerente do projeto, analista e representantes da unidade de negócio e sendo explanados a motivação, escopo, benefícios e resultados esperados.

O piloto iniciou a partir do projeto de desenvolvimento do sistema *Web* de Gerenciamento da Cobrança Extrajudicial Terceirizada cujo objetivo é possibilitar que o processo de cobrança extrajudicial, exercido por empresa contratada, ocorra de forma eficiente e segura, atendendo às necessidades do cliente no que concerne ao controle das operações que estão em cobrança, consolidação dos dados que serão enviados à empresa terceirizada, a integração com outros sistemas e a emissão de relatórios de acompanhamento e gerenciais.

A primeira atividade do piloto foi organizar informações sobre o novo sistema a partir da documentação do projeto e de artefatos, como: Fluxo do Processo, Regras de Negócio e Termo de Abertura.

A partir destas informações, realizou-se *brainstorming* para identificação de pontos mais críticos com intuito de subsidiar a ação “Identificar Necessidades de Segurança”. Além disso, adicionalmente, foram apontados os principais impactos para cada ponto crítico, como: imagem, confidencialidade, ou resultado financeiro. Cobrar pessoas erradas ou cobrar com valor errado poderia comprometer a imagem do Banco, por exemplo.

O *brainstorming* suscitou questões importantes sobre riscos inerentes ao projeto, como: segurança nos serviços de Rede de Valor Agregado, bem como a falta de tempestividade da troca de informações, podendo resultar em cobranças indevidas e permanência das restrições externas em nome do cliente após a regularização das suas dívidas.

A definição das responsabilidades pelas ações de segurança entre os envolvidos externos e internos e a identificação de resultados satisfatórios na atual forma de verificar e validar requisitos não funcionais foram outros assuntos surgidos durante o *brainstorming*.

Logo, as necessidades de segurança foram identificadas e inseridas no documento de Visão, sendo as factíveis transformadas em requisitos de segurança, como parte da ação “Capturar Requisitos de Segurança” (Tabela 2). A diretriz “Requisitos de Segurança” e o “Guia de Orientação de Segurança de *Software*” também contribuíram para a descrição de alguns requisitos de segurança.

**Tabela 2. Exemplos de requisitos de segurança**

Item	Breve descrição
6.2 Controle de acesso	O controle de acesso às funcionalidades do sistema ocorre conforme o nível de sigilo das informações manipuladas pelo sistema, que é definida pelo gestor principal da área de negócio, a partir de critérios de criticidade da informação.
6.3 Integridade de dados	Controles de entrada de dados por usuários e ou interface utilizarão validações para os tipos de dados, ou expressões regulares, entre outras técnicas para evitar a entrada de dados inconsistentes ou scripts com comandos invasores.
6.5 Controle de sessão	O sistema limitará o número de sessões concorrentes e fechará a interface com o usuário e ou outro sistema, obrigando nova autorização, quando os seguintes tipos de situações ocorrerem: Perda de integridade de informações de controle de acesso; Falha na comunicação com o servidor; e Tempo limite sem atividade expirado.

Foi importante a frequente participação de usuários nas discussões de segurança, pois evidenciou o compromisso da equipe técnica com a qualidade e segurança do futuro sistema, além de permitir o constante alinhamento das percepções sobre funcionalidades e a compreensão de novas necessidades.

Ponto importante no artefato “Especificação Suplementar”, seção “Confiabilidade”, é o item “Tolerância a Falhas” (Figura 2). Sobre este tópico, foram elencadas as funcionalidades críticas do sistema que precisariam continuar ativas mesmo em cenários de falha. Contudo, foi decidido que os controles para manter em operação estas funcionalidades seriam externos ao funcionamento do *software* e garantidos pela equipe de arquitetura, com suporte de infraestrutura.

#### 4. Resultados Obtidos

Com o desenvolvimento de um sistema em ambiente Web, utilizado como projeto piloto da institucionalização da segurança na disciplina Requisitos do RUP-BNB, por meio da aplicação de novas ações, utilização de diretrizes de segurança de *software* e artefatos adaptados, nas fases de Iniciação e Elaboração, foram ratificadas as melhorias propostas no processo de *software* adaptado com práticas de segurança. Essas melhorias são percebidas com a conclusão eficaz do escopo do projeto e atendimento aos objetivos definidos, além de atender recomendações de órgãos reguladores. Assim, entre os resultados obtidos destacam-se:

- Proposição de melhoria do processo de desenvolvimento RUP-BNB pela institucionalização de ações de segurança para a disciplina Requisitos.
- Atualização e elaboração de artefatos para institucionalização do processo especializado, quais sejam:
  - Diretriz Requisitos de Segurança: Artefato de apoio à identificação de requisitos de segurança;
  - Guia de Orientação de Segurança de *Software* - Falhas de Segurança em Aplicativos: Artefato de apoio para identificar vulnerabilidades e ameaças em aplicativos.
- Aperfeiçoamento de artefatos “Visão” e “Especificação Suplementar” com a inclusão de sessões de segurança para apoiar a execução do processo.
- Customização da tarefa “Identificar Padrões de Segurança” e atualização de abordagens relacionadas à disciplina Teste.
- Realização de projeto piloto para validar as ações de segurança, contemplando: Identificação de necessidades de segurança, Formalização de requisitos de segurança, Ampla discussão de assuntos inerentes à segurança, Análise de resultados para avaliar melhorias necessárias no processo proposto, e Sinalização de novos projetos e perspectiva de institucionalizar segurança na disciplina Teste.
- Disseminação da cultura de segurança de *software* a partir das seguintes iniciativas: Curso sobre introdução à Segurança de *Software*, Realização de palestra sobre Segurança de *Software*, e Condução de reuniões de trabalho onde foi tratado o tema de segurança em relação às necessidades das aplicações.

## 5. Aplicabilidade dos Resultados

O alcance desse projeto é, inicialmente, de âmbito interno ao Banco do Nordeste e a proposta poderá ser utilizada em todos os projetos de desenvolvimento de software com impacto direto na segurança dos aplicativos. Não obstante, as ideias, experiência e resultados adquiridos poderiam não apenas ser compartilhados com pessoas e instituições interessadas, como também organizadas para divulgação em artigos para revistas especializadas e congressos nacionais e internacionais.

## 6. Características Inovadoras

Houve melhoria no processo de desenvolvimento de aplicativos no que concerne à produção de software mais seguro a partir do uso dos resultados deste trabalho, que considera trabalhos acadêmicos, normas internacionais e melhores práticas de mercado. Portanto, esse projeto inova em propor e validar ações de segurança conduzidas na disciplina Requisitos do processo de desenvolvimento de software baseado na metodologia RUP, que pode vir a ser adaptado para outras corporações, de acordo com suas necessidades.

## 7. Conclusão e Perspectivas Futuras

Em geral, a contribuição de segurança na disciplina Requisitos do RUP-BNB foi amplamente aceita e considerada eficaz e adequada para as características dos profissionais e projetos do Banco do Nordeste, alinhando-se com a atual estratégia corporativa.

Pode-se considerar esse projeto como evolutivo e renovador, com a evolução caracterizada pelas melhorias em artefatos, atividades, e no próprio processo que objetivam aperfeiçoar a segurança de aplicativos, e, por conseguinte, sua qualidade. A renovação é representada pela alteração na forma de pensar e de entender segurança no contexto de ciclo de vida de software, implicando em mudança cultural de técnicos e executivos de negócio.

A experiência prática adquirida na execução do piloto elevou a percepção de todos sobre a importância de tratar a segurança dentro do ciclo de vida de *software*, em especial no seu processo de desenvolvimento. Permitiu, ainda, ampla discussão e associação de assuntos que, direta ou indiretamente, repercutiam com a confidencialidade, integridade e disponibilidade das informações, como a sugestão de se inserir matriz de responsabilidades, conhecida como “*RACI chart*”, no documento de Especificação Suplementar, a fim de evitar vulnerabilidades na segregação de funções.

Portanto, conclui-se que a segurança no RUP-BNB provou-se institucionalizável, uma vez que atingiu os objetivos e resultados esperados. Não obstante, apesar da conclusão do projeto piloto ocorrer com a formalização dos requisitos de segurança, julgou-se coerente e necessário continuar apoiando a equipe técnica na transformação dos requisitos em controles e na verificação e validação destes.

Como proposta de trabalho futuro, de forma a consolidar e aperfeiçoar as atuais ações de requisitos de segurança no RUP-BNB, será prestada assessoria continuada na execução de outros projetos, orientando e prestando o suporte necessário até que as equipes estejam capacitadas a conduzirem, de forma autônoma, estas ações.

Por conseguinte, ao passo que diferentes necessidades e requisitos de segurança sejam identificados no mercado e em novos projetos, será possível definir um padrão mínimo (*baseline*), suportado pela gestão do conhecimento, como direcionador de resultados mais efetivos.

Finalmente, utilizando-se do novo “Plano de Testes” e das abordagens “Teste de Segurança e de Controle de Acesso” e “Teste de Tolerância a Falhas e de Recuperação”, espera-se uma condução premente de nova iniciativa para institucionalizar segurança na disciplina Teste do processo RUP-BNB, ensejando novas melhorias no processo.

## 8. Referências Bibliográficas

BCB (2006), Resolução nº 3380 do Banco Central do Brasil. Disponível em: [www.bcb.gov.br](http://www.bcb.gov.br).

CERT.BR. (2011), Estatísticas. Disponível em: <http://www.cert.br/stats/incidentes/>.

CMMI-DEV, CMMI® for Development, V1.2 model, CMU/SEI-2006-TR-008. Software Engineering Institute, 2006.

Howard, M. and LeBlanc, D. (2002), *Writing Secure Code*, Microsoft Press, 2<sup>nd</sup> edition.

Howard, M. and Lipner, S. (2006), *The Security Development Lifecycle*, Microsoft Press, 1<sup>st</sup> edition.

- ISO/IEC 15408. (2005) Information technology – Security techniques – Evaluation criteria for IT security.
- McGraw, G. (2006), Software security: building security in, Addison Wesley, 1st edition.
- MITRE. (2010), Making Security Measurable. Disponível em: [measurablesecurity.mitre.org](http://measurablesecurity.mitre.org).
- Nunes, F. PASS: Processo de Apoio à Segurança de Software. Orientador: Arnaldo Dias Belchior. Dissertação de Mestrado. UNIFOR – Fortaleza, Brasil, 2007.
- OWASP. (2010), OWASP Top 10 Project. Disponível em: <http://www.owasp.org>.
- RUP, IBM - Rational Unified Process ®. IBM Corporation. Copyright © 1987 – 2011.
- WEBAPPSEC. (2010), Web Application Security Consortium – Threat Classification. Disponível em: [www.webappsec.org](http://www.webappsec.org).
- VERIZON. (2011), Data Breach Investigations Report. Disponível em: [http://www.verizonbusiness.com/resources/reports/rp\\_data-breach-investigations-report-2011\\_en\\_xg.pdf](http://www.verizonbusiness.com/resources/reports/rp_data-breach-investigations-report-2011_en_xg.pdf)

# Um *framework* para implantação de gerenciamento de configuração baseado no MPS.BR e RUP

Marcelo Hideki Yamaguti, Helder Lino Moura Cardoso, Eder John Delgado de Pina  
Faculdade de Informática – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)  
Av. Ipiranga, 6681 - Prédio 32, CEP: 90619-900 - Porto Alegre – RS

yamaguti@pucrs.br, helder.caboverde@gmail.com, brotherjohn55@hotmail.com

**Abstract.** *In this paper we present the definition of a framework for configuration management using the MPS.BR reference model, by adapting the process proposed by RUP. We also present some analysis of free tools that may be used in the proposed process.*

**Resumo.** *Este trabalho visa apresentar a definição de um framework para o gerenciamento de configuração usando o modelo de referência do MPS.BR, pela adaptação do processo proposto pelo RUP. Também são analisadas algumas ferramentas livres que poderão ser utilizadas no processo proposto.*

## 1 Introdução

No desenvolvimento de software as empresas de pequeno e médio porte apresentam algumas dificuldades para garantir a qualidade dos seus processos devido aos custos, modelos de qualidade complexos e falta de um processo definido alinhado com um modelo de qualidade.

Um desses processos que acabam não sendo implantados ou implantados com pouca qualidade é o gerenciamento de configuração. Nesse contexto, este trabalho visa oferecer às pequenas e médias empresas um *framework* que satisfaz as suas necessidades. Para isso, esse trabalho usa o RUP para pequenos projetos como um processo já definido, apresentando uma análise da conformidade da gerência de configuração do mesmo para com a gerência de configuração do modelo de qualidade MPS, propondo ajustes no RUP com os resultados obtidos. Para auxiliar nesse processo de gerenciamento de configuração, partindo dos ajustes feitos, algumas ferramentas livres serão analisadas e indicadas uma vez que, para pequenas e médias empresas comprar licença de um software proprietário eleva consideravelmente os custos do empreendimento. Ao final será realizado um estudo de caso, para analisar em campo a eficiência desse *framework* e fazer um relatório dos resultados obtidos e fazer os ajustes caso necessário.

## 2. Objetivos e Justificativa

O objetivo deste projeto é apresentar um *framework* para o gerenciamento de configuração, com um processo baseado no RUP para pequenos projetos e aderente ao MPS.BR.

Este *framework* também identificará alguns softwares livres adequados ao suporte do processo.

Como objetivos específicos do projeto tem-se:

- Adaptação do processo de gerência de configuração do RUP para aderência com o MPS.BR;
  - Análise ferramentas livres que são utilizadas na Gerência de Configuração, para a composição do *framework*;
  - Execução de um estudo de caso com o *framework* proposto;
  - Análise dos resultados do estudo de caso realizado;
  - Consolidação de um *framework* completo;
  - Geração de um Trabalho de Conclusão de Curso de graduação.

A justificativa do trabalho baseia-se no fato que mesmo pequenas e médias empresas com limitação de recursos têm buscado melhorar os seus processos de desenvolvimento de software para garantir a qualidade de seus produtos. Uma importante área de processo é o gerenciamento de configuração, pois mudanças são quase inevitáveis ao longo do ciclo de vida dos projetos, e a sua inexistência gera confusão no desenvolvimento.

Neste contexto, é interessante a definição de um processo aderente a um modelo de qualidade e o uso de tecnologias com custos viáveis. O MPS.BR define um modelo de qualidade nacional adequado a pequenas e médias empresas. O RUP é um processo de desenvolvimento e possui uma adaptação para pequenos projetos. Além disso,

existem diversas ferramentas livres que podem ser adaptadas e empregadas para o suporte ao gerenciamento de configuração.

### 3. Metodologia de Execução

O projeto foi desenvolvido em seis etapas:

1. Fundamentação teórica: contextualização do trabalho, aprofundamento de conceitos relativos ao gerenciamento de configuração em projetos.
2. Pesquisa de trabalhos similares: busca de trabalhos anteriores que abordassem o tema de gerenciamento de configuração em projetos.
3. Análise de aderência do processo RUP para pequenos projetos em relação ao processo GCO definido no MR.MPS.
4. Adaptação do processo RUP para suporte ao processo GCO do MR.MPS.
5. Estudo de ferramentas livres para apoio ao processo definido.
6. Realização de um experimento de uso e análise do *framework* proposto.

### 4. Resultados Obtidos

- Processo: como resultado do projeto foi gerado um processo de gerenciamento de configuração de software aplicável a pequenas e médias empresas com uso de ferramentas de suporte livre.
- Trabalho acadêmico: o *framework* foi desenvolvido durante um trabalho de conclusão de Curso de Graduação de Bacharelado em Sistemas de Informação na Faculdade de Informática da Pontifícia Universidade Católica do Rio Grande do Sul.

### 5. Aplicabilidade dos Resultados

O uso do *framework* permite que pequenas e médias empresas possam implantar um processo de gerenciamento de configuração em seus projetos, que seja aderente ao processo GCO do MR.MPS, com uso de ferramentas livres.

### 6. Características Inovadoras

Existem outras iniciativas de definição de processos de gerenciamento de configuração, entretanto, este projeto visou definir um *framework* composto pela definição de um processo (adaptado do processo RUP e aderente ao MR.MPS) e um conjunto de ferramentas livres que podem ser utilizados para dar suporte ao processo, em pequenas e médias empresas.

### 7. Conclusão e Perspectivas Futuras

O desenvolvimento deste projeto durante o ano de 2011, envolveu inicialmente estudos e pesquisas sobre gerenciamento de configuração, MPS.BR e RUP para pequenos projetos; que permitiu uma análise de conformidade do gerenciamento de configuração do MR.MPS no RUP para pequenos projetos.

Com a análise de conformidade foram propostas ajustes nas atividades do processo do RUP para pequenos projetos, em que a principal alteração foi a criação da atividade de auditoria, uma vez que, o RUP não possui a mesma. Com os resultados desses ajustes, foi desenvolvido o processo definido para pequenos projetos adaptado, com todas as tarefas, papéis e artefatos. Esse processo proposto ajuda a diminuir o grau de dificuldade que pequenas e médias empresas se deparam no que se refere à necessidade de ter um processo definido para estrutura deste tipo de empresa alinhado com um modelo de qualidade brasileiro.

Como parte complementar do *framework*, houve o levantamento, seleção e análise das ferramentas livres, seguido de combinações das mesmas. Provendo assim as alternativas de ferramentas que podem ser utilizadas pelas pequenas e médias empresas no gerenciamento de configuração.

Para a análise de viabilidade do *framework*, foi realizado o estudo de caso com 3 equipes (4 alunos cada), durante todo o segundo semestre de 2011.

Ao final da construção deste *framework* foi constatado que esse trabalho pode ser continuado, com uma análise mais focada em ferramentas livres que realizam o gerenciamento de configuração de forma distribuída que tem se mostrado uma tendência do mercado. Futuramente pode ser realizado novo experimento, a publicação do processo utilizando uma ferramenta de edição e publicação de processos (como por exemplo, o EPF Composer), a automatização do processo proposto com a construção de uma ferramenta e uma análise de aderência deste *framework* com alguns tipos de projetos de desenvolvimento de software que pequenas e médias empresas realizam.

## 8. Referências Bibliográficas

Barros, 2009 - Manual do Usuário. Disponível em:<

[http://www.spider.ufpa.br/projetos/spider\\_cl/spider\\_cl\\_manual.pdf](http://www.spider.ufpa.br/projetos/spider_cl/spider_cl_manual.pdf)>. Acesso em: 09 de Dezembro de 2011.

Ben Collins-Sussman, 2007 - Version Control with Subversion, 2007. Disponível em:< <http://svnbook.red-bean.com/en/1.7/index.html>>. Acesso em: 09 de Dezembro de 2011.

CollabNet, 2009. Disponível em: <<http://scarab.tigris.org/>>. Acesso em: 09 de Dezembro de 2011.

David J. Murphy, 2007- Managing Software Development with Trac and Subversion, 2007 - packt publishing. Disponível em:< [http://www.amazon.com/Managing-Software-Development-Trac-Subversion/dp/1847191665#reader\\_B005D6QDNE](http://www.amazon.com/Managing-Software-Development-Trac-Subversion/dp/1847191665#reader_B005D6QDNE)>/>. Acesso em: 09 de Dezembro de 2011.

Free Software Foundation, 2008. Disponível em: < <http://www.fsfla.org/svnwiki/about/what-is-free-software.pt.html>>. Acesso em: 09 de Dezembro de 2011.

GIT – ferramenta livre. Disponível em: <<http://git-scm.com/>>. Acessado em 09 de Dezembro de 2011.

GNU, 2011. Disponível em: <<http://nongnu.org/cvs/>>. Acesso em: 09 de Dezembro de 2011.

IBM RUP - <http://www-01.ibm.com/software/awdtools/rup/>; Acessado no dia: 09 de Dezembro de 2011.

IBM RUP – Small Projects, 2011. Disponível em: <<http://www.wthree.com/rup/smallprojects/>>. 09 de Dezembro de 2011.

Kenzaburo Ito, 2002 - "Mantis User Documentation". Disponível em:< <http://open.affelio.jp/modules/mantis/doc/documentation.html>>. Acesso em: 09 de Dezembro de 2011.

Mantis, Ferramenta - Disponível em: <http://www.mantisbt.org/>; Último acesso no dia 09 de Dezembro de 2011.

NCE, 2007. Disponível em:

<<http://labase.nce.ufrj.br/cursos/fdbc/aulas/aula1/RUPx20x2dx20Ciclox20dex20vidax20dox20desenvolvimentox20dex20umx20sistemax2e.html>>. Acesso em: 09 de Dezembro de 2011.

NOGUEIRA, Marcelo. Gestão de configuração software: uma visão estabelecida pelas normas e modelos de qualidade. Disponível em: <[http://www.noginfo.com.br/arquivos/CC\\_ESOF\\_II\\_02.pdf](http://www.noginfo.com.br/arquivos/CC_ESOF_II_02.pdf) >. Acesso em: 09 de Dezembro de 2011.

SILVA, Carlos et al. Uso da Gestão de Configuração de Software pelas Organizações em Busca de Certificação. Disponível em: < <http://www.iem.unifei.edu.br/sanches/Pesquisa/Artigos%20publicados/a52.PDF> >. Acesso em: 09 de Dezem  
SILVA, Josenildo et al. Aderência de um Processo Pesado(RUP) a um Modelo de Qualidade (MPS.BR), Recife, 2007).

SUBVERSION - ferramenta livre. Disponível em: < <http://subversion.tigris.org/>>. Acessado em 09 de Dezembro de 2011.

SOFTEX - Guia Geral 2009. Disponível em : [http://www.softex.br/mpsbr/\\_guias/default.asp](http://www.softex.br/mpsbr/_guias/default.asp); Acesso em: 09 de Dezembro de 2011.

SOFTEX - Guia de Implementação – Parte 2: Nível F:2011 (Julho de 2011). Disponível em : [http://www.softex.br/mpsbr/\\_guias/default.asp](http://www.softex.br/mpsbr/_guias/default.asp); Acesso em: 09 de Dezembro de 2011

SOMMERVILLE, Ian. Engenharia de software. 8. Ed. São Paulo: 2007.

PEREIRA, Anna Carolina, implantando gerência de configuração com o apoio da ferramenta subversion (svn) para atingir os resultados esperados do processo GCO do nível F do MPS.BR. Disponível em: <[http://subversion.assembla.com/svn/gco/livros/Anna\\_Carolina\\_Pereira.pdf](http://subversion.assembla.com/svn/gco/livros/Anna_Carolina_Pereira.pdf)>. Acesso em: 09 de Dezembro de 2011.

PRESSMAN, Rogers S. Engenharia de Software, São Paulo, 1995

Pro GIT – Scott Chacon. Disponível em: <http://progit.org/book>. Acesso em: 09 de Dezembro de 2011.

Scott Chacon, 2009 - PRO GIT. Disponível em:< <http://progit.org/book/>>. Acesso em: 09 de Dezembro de 2011.

TECPRO IT - <http://www.tecproit.com.br/consultoria-rup.html>; Acesso em: 09 de Dezembro de 2011.

TRAC – ferramenta livre. Disponível em: < <http://trac.edgewall.org/>>. Acesso em: 09 de Dezembro de 2011.

YOSHIDOME, Ewelton et al. Uma proposta de apoio sistematizado à gerência de configuração do MPS.BR utilizando ferramentas de software livre, Belém, 2009.

## Ferramenta de Desenvolvimento Quellon Studio

Carlos José Pereira

Fácil Informática Ltda

Rua Gustavo Budag, 407 – 89.036-501 – Blumenau, SC – Brasil

pereira@facil.com.br

**Abstract.** The systems developed by Quellon can be used by various browsers like IE9, Chrome, Firefox, Safari and others. The systems are compatible with major database market as Oracle and MS-SQL Server. The tool has all the treatment of groups of users and their profiles, report generator and segregation of services. The systems are born with different functionalities needed, which ensures great productivity and agility. The interface is simple and modern and the screens can be automatic or built by programmers when necessary. Prepared for large volumes of information and lots of hits.

**Resumo.** Os sistemas desenvolvidos com a Quellon podem ser utilizados por diversos browsers como IE9, Chrome, Firefox, Safari e outros. Os sistemas são compatíveis com os principais bancos de dados do mercado, como Oracle e MS-SQL Server. A ferramenta dispõe de todo o tratamento de grupos de usuários e seus perfis, gerador de relatórios e segregação de serviços. Os sistemas nascem com diversas funcionalidades necessárias já prontas, o que garante muita produtividade e agilidade. A interface é simples e moderna e as telas podem ser automáticas ou construídas pelos programadores quando necessário. Preparada para grandes volumes de informações e grande quantidade de acessos.

### 1. Introdução

O desenvolvimento de softwares aplicativos em grande escala, para venda em todo o Brasil, América Latina ou outros países, traz consigo enormes desafios para as empresas de software.

#### 1.1. Idioma

Este problema não afeta produtos que foram projetados para serem comercializados apenas em um país. Mas é muito comum atualmente, com a difusão da internet e a necessidade de expansão mercadológica, que as empresas pensem em ampliar seus mercados, fazendo produtos que atendam vários países.

O desenvolvedor deve permitir que seus produtos possam ser multi-idioma. Que os usuários ao acessarem as aplicações tenham as telas traduzidas para o idioma nativo, facilitando o entendimento e a utilização.

#### 1.2. Mobilidade

Os aplicativos devem estar disponíveis para acesso em diversos dispositivos, mormente aqueles móveis como Tablets, Smartphones e diversos browsers. A compatibilização da aplicação com os diversos modelos é tarefa bastante difícil, exigindo equipes de técnicos focados apenas no tratamento de interfaces.

#### 1.3. Personalizações

As aplicações precisam ser flexíveis e se adaptarem rapidamente às preferências e características dos clientes. O desenvolvedor precisa lidar com produtividade, baixo custo de adaptação e principalmente com o controle dessas mesmas alterações diferenciadas para centenas e às vezes milhares de clientes. Uma tarefa quase impossível se não houver um método adequado, um processo construído especialmente para atender a essas particularidades, principalmente nas trocas de versões do produto. É impossível a manutenção de centenas de fontes. A aplicação precisa ser construída de forma a atender esse quesito com transparência para os desenvolvedores.

#### 1.4. Produtividade

Rapidez nos desenvolvimentos é fundamental, principalmente nas personalizações quando necessárias.

#### 1.5. Segurança

Produtos web exigem segurança redobrada. Facilidade na criação de perfis com permissões e restrições, opção de instalações em quatro camadas, restrições de acessos indevidos através de comandos indevidos, são algumas das exigências básicas aos novos produtos.

## 1.6. Relatórios e Gráficos

O usuário precisa ter ampla flexibilidade na emissão de relatórios e configurações de diversos tipos de gráficos para indicadores de desempenho. Um gerador de relatórios é item importante para qualquer sistema, com saídas PDF, Office e diversos tipos de gráficos.

## 1.7. Integrações

Aplicações devem se ligar com os outros sistemas através de conectores como os Web Services e outras formas de acesso diretos a bancos de dados diferenciados.

## 1.8. Multi-Banco

Os principais bancos de dados de mercado devem ser suportados. Para facilitar essas questões, a Quellon do Brasil Sistemas S.A. desenvolveu uma ferramenta para auxiliar no desenvolvimento de aplicações. Todos os pontos citados acima foram abordados. Os desenvolvedores conseguem fazer sistemas mais rapidamente, com menos erros, mais potentes, mais rápidos e principalmente feitos para serem utilizados globalmente.

## 2. Objetivos e Justificativa

### 2.1. Sistemas são Dinâmicos

O ideal seria que um sistema/programa aplicativo uma vez desenvolvido não sofresse mais alterações. Ou que sofresse somente intervenções para correções de erros ou de desvios de suas funcionalidades em relação ao projeto original. Um sistema/programa quanto mais utilizado e quanto menos alterado melhor.

Na vida real os sistemas/programas aplicativos são extremamente dinâmicos. Necessitam de modificações constantes por:

- Correções de erros ou de desvios em relação ao projetado;
- Atos governamentais. Federais, Estaduais e Municipais;
- Modificações nas regras do negócio;
- Alterações exigidas por auditorias;
- Mudanças tecnológicas;
- Melhorias na interação com o usuário;
- Aperfeiçoamento constante em relação à segurança das informações;
- Adaptações a diversos segmentos de atividades;
- Inadequação das soluções previstas no projeto original;
- Mudanças de expectativa de novas diretorias, necessidade de novos resultados;
- Outras.

O que importa é que um sistema não é estático. É muito dinâmico.

Em um sistema de mercado, utilizado por muitas empresas, são milhares de novas linhas de código complexas inseridas todos os anos e que devem atuar em consonância com códigos estabelecidos ou mesmo substituí-los, criando novas ligações e possibilidades. As bases de dados recebem constantemente espaços para novas informações, para todas as situações, ou para servirem somente em determinados aspectos. Telas são alteradas para espelharem as mudanças e a necessidade de novas parametrizações é inerente.

### 2.2. Atualizar o sistema - grande desafio para produtos altamente personalizados

Quando o cliente faz uso das vantagens de um sistema altamente personalizado, está criando particularidades no sistema que são somente suas. Inexistentes em sistemas de outros clientes que utilizem o mesmo programa. E é, na troca de versões, que se apresenta uma situação bastante peculiar. Mesclar novas versões que foram desenvolvidas pelo fornecedor, com novidades, melhorias, novas funcionalidades, mudança de funcionalidades existentes, enfim centenas de modificações, com a versão do cliente, que também está muito personalizada, configurada e adaptada e pode ter sofrido inúmeras transformações. Ou na melhor das hipóteses, não sofreu grandes modificações, mas está parada no tempo, enquanto o produto do fornecedor alterou-se bastante.

### 2.3. Quellon - preparada para respeitar personalizações

Existe um investimento considerável em ferramental que permite resolver o problema complexo: mesclar uma versão, ou versões mais recentes, em uma versão altamente personalizada para determinado cliente ou versão muito antiga.

A Quellon faz parte dessa nova geração de ferramentas, que permite todos os níveis de parametrizações, configurações e customizações. O programa atualizador da Quellon enfrenta todas as “intempéries” de uma troca de versão:

- Permite a atualização acumulada de versões. Ou seja, se o usuário deixou de fazer a atualização de algumas versões, basta atualizarem a última e estará atualizando todas.
- Inclui na versão do cliente as novidades, melhorias e correções de erros de todas as versões que estão sendo atualizadas. São centenas de modificações em diversos pontos.
- Mantém as personalizações que foram feitas para o cliente. Legendas, mudanças de telas, parametrizações, customizações, novas informações, etc.
- Informa sobre casos que necessitem de intervenção manual. Exemplo: o cliente solicitou a exclusão de uma tela, pois achava desnecessária, mas novas versões alteram essa tela incluindo novas informações. As novas informações devem ser transferidas para outros locais, caso o cliente deseje utilizá-las ou então devem ser desativadas como as demais da tela antiga.

### **3. Metodologia de Execução**

Partindo-se das premissas colocadas na introdução, os sócios de três empresas tradicionais de software de Blumenau, Hbsis, Ellevo e Fácil Informática reuniram capital e criaram um empresa sociedade anônima, Quellon do Brasil Sistemas S.A., para desenvolver uma ferramental para desenvolvimento de sistemas web.

Um grupo de programadores contratados desenvolveu a ferramenta e as empresas passaram a aplicar em seus próprios sistemas com bastante êxito e oferecida para outras empresas de software brasileiras.

### **4. Resultados Obtidos**

A Fácil Informática desenvolveu um produto denominado ESPAIDER para a área jurídica e tem atualmente centenas de clientes em todo o Brasil, agora com expansão para a América Latina. ([www.facil.com.br](http://www.facil.com.br)).

A ferramenta comportou-se como o esperado, melhorando a produtividade, facilitando o desenvolvimento e atualização de sistemas, bem como, cumprindo perfeitamente o compromisso da mobilidade e resultados para os clientes.

### **5. Aplicabilidade dos Resultados**

A Fácil Informática utiliza a ferramenta de desenvolvimento Quellon com bastante sucesso. ([www.facil.com.br](http://www.facil.com.br)).

Centenas de clientes já utilizam sistemas com a tecnologia Quellon. Grande parte do sucesso da empresa se deve à utilização dessa tecnologia, que permite sistemas flexíveis, móveis e perfeitamente adaptados às características dos clientes.

### **6. Características Inovadoras**

Ferramentas de desenvolvimento existem em todo o mundo. Embora sejam poucos fornecedores, dada a complexidade do tema e o mercado diferenciado que são as empresas. As características mais inovadoras da Quellon são:

- Uma das características mais inovadoras da ferramenta é a não geração de códigos fontes, o que facilita a construção e manutenção de sistemas. Os códigos ficam encapsulados e não necessitam de tratamento pelo desenvolvedor.
- Construída para os novos sistemas web. É uma das primeiras ferramentas mundiais para desenvolvimento de sistemas na plataforma web.
- É multi-browser. Os principais browsers do mercado como o Internet Explorer 9.0, o Chrome 15, o Firefox e o Safari, podem ser utilizados. E já no segundo semestre de 2012 estará disponível também para tablets.
- Pode ser utilizada com os bancos de dados Oracle e MS-SQL Server e os comandos para os desenvolvedores são transparentes.
- É a única da América Latina que se preocupa com o aspecto de atualização das versões dos sistemas e manutenção das customizações realizadas para os clientes.
- A ferramenta já dispõe de gerador de relatórios agregado.
- Os sistemas desenvolvidos com a Quellon podem ser instalados em até 4 camadas e os serviços distribuídos em diversos servidores melhora a performance.

## **7. Conclusão e Perspectivas Futuras**

A ferramenta de desenvolvimento Quellon se mostrou excelente para o desenvolvimento de aplicações web. O sucesso das empresas que utilizam comprova esse fato.

A empresa Quellon tem crescido ao longo do tempo e tem como meta expandir os negócios no Brasil e na América Latina oferecendo para empresas de software essa ferramenta inovadora e também para empresas que tenham que passar seus sistemas legados para plataformas mais modernas.

## **8. Referências Bibliográficas**

Eberspächer, H. e Forbellone, A. (1993) “Lógica de Programação – A Construção de Algoritmos e Estrutura de Dados”.

FÁCIL informática. Disponível em: <<http://www.facil.com.br>> Acesso em: 10 mar. 2012.

HBSIS. Disponível em: <<http://www.hbsis.com.br>> Acesso em: 15 mar. 2012. Hock, Dee. (2006), Nascimento da Era Caórdica, 10ª edição.

Jackson, Michael. (1983), System Development, s. ed.

Kimball, R. e Merz, R. (2000) “Data Webhouse: Construindo o data warehouse para a Web”.

Larocca, Danielle. (2000), SAP R/3 – Ferramentas de Relatório, s.ed.

QUELLON sistemas do brasil sa. Disponível em: <<http://www.uellon.com.br>> Acesso em: 07 fev. 2012. Wirth, Niklaus. (1978), Programação Sistemática, s. ed.

# O processo de Inovação Centrada no Ser Humano: boas práticas de desenvolvimento de software utilizadas no projeto Portal do Aluno

Gabriel C. Cardoso<sup>1</sup>, Ivan L. Martins<sup>2</sup>

<sup>1</sup>Mestrado em Design e Expressão Gráfica – Universidade Federal de Santa Catarina (UFSC) – Florianópolis, SC

<sup>2</sup>Centro de Convergência Digital – Fundação Centros de Referência em Tecnologias Inovadoras (CERTI) – Florianópolis, SC

gabrielc.cardoso@gmail.com, linhares@gmail.com

**Abstract.** *This paper is about the software development process adopted by the Digital Convergence Center of the CERTI Foundation, Florianópolis, Santa Catarina, Brazil. It describes the Human-Centered Innovation framework, which combines good principles of Software Engineering, Agile Software Development and User-Centered Design. It shows, using the Student Portal project as an application, some techniques that make up the framework and how its use has contributed to increasing productivity and software quality.*

**Resumo.** *Este artigo aborda o processo de desenvolvimento de software utilizado pelo time de software do Centro de Convergência Digital da Fundação CERTI, Florianópolis, Santa Catarina. Descreve o framework Inovação Centrada no Ser Humano, que une bons princípios de Engenharia de Software, Desenvolvimento Ágil de Software e Design Centrado no Usuário. Ilustra, utilizando o projeto Portal do Aluno, algumas técnicas que compõem o framework e como o seu uso contribuiu para o aumento da produtividade e da qualidade do software desenvolvido.*

## 1. Introdução

O volume de software desenvolvido e disponível para as pessoas cresceu de maneira significativa na última década. Junto a esse aumento, outras mudanças relevantes aconteceram na Engenharia de Software, como a crescente utilização do software como serviço (SaaS, *software as a service*) e da computação na nuvem (*cloud computing*). Com diferentes opções disponíveis de software que apresentam funcionalidades similares, os usuários agora têm a possibilidade de escolher qual utilizar. Agora eles não procuram apenas funcionalidade e usabilidade. O que importa é a experiência proporcionada pela tecnologia (NORMAN, 2011).

Projetos de software costumam falhar em duas frentes: a primeira diz respeito aos processos internos. Se estes não forem bem estruturados, o software pode levar mais tempo que o estimado, consumir mais recursos que o orçado, focar excessivamente nos aspectos tecnológicos -- em detrimento da satisfação dos usuários --, ou responder a mudanças exigidas pelo mercado de uma maneira mais lenta que o necessário. A segunda frente diz respeito ao resultado do projeto. O software entregue pode apresentar uma qualidade inferior à esperada pelos usuários, pode ser complexo demais, difícil de usar ou mesmo rude com eles, o que pode levar ao fracasso do produto.

Frente a esse contexto, Cooper, Reimann e Cronin (2007) apontam que os produtos digitais interativos necessitam de melhores processos. É sabido que processos internos podem ser melhorados por meio da adoção de boas práticas de Engenharia de Software. Questões referentes a recurso empregado e adaptação podem ser melhoradas com o uso de Métodos Ágeis de Desenvolvimento de Software. E para entregar software de melhor qualidade que proporciona uma experiência de uso positiva, recomenda-se o uso da abordagem de Design Centrado no Usuário (GARRETT, 2011).

Partindo das recomendações acima apresentadas, o time de software do Centro de Convergência Digital (CCD) da Fundação CERTI vem aperfeiçoando, durante os últimos três anos, o processo de desenvolvimento utilizado, buscando adotar a filosofia do Desenvolvimento Ágil, boas práticas de Engenharia de Software e os princípios do Design Centrado no Usuário. Como resultado, chegou-se à abordagem *Inovação Centrada no Ser Humano*, aqui apresentada por exemplos do projeto Portal do Aluno.

## 2. Objetivos e Justificativa

É visível o destaque que os métodos ágeis alcançaram em âmbito mundial nos últimos anos. Em uma pesquisa realizada em 2011 com mais de 6000 pessoas sobre o panorama mundial do desenvolvimento ágil, os entrevistados apontaram os benefícios da adoção de métodos ágeis. Três quartos (75%) apontaram um aumento de produtividade na companhia e 84% relataram como maior benefício a capacidade de modificar as prioridades do consumidor (VERSIONONE, 2011).

Assim como métodos ágeis trazem benefícios, a aplicação de bons princípios de Engenharia de Software contribui para a qualidade do software desenvolvido. A escrita de testes unitários, o uso de testes automatizados e de processos de integração contínua -- que verificam tais testes -- garantem que o código escrito não apresente problemas, evitando a frustração do usuário e experiências negativas advindas do uso do software.

A experiência do usuário se tornou, na última década, um tópico de grande interesse no campo da Interação Humano-Computador (IHC). Segundo Garrett (2011), o projeto de produtos que buscam oferecer uma boa experiência do usuário se dá por meio do Design Centrado no Usuário (DCU). O uso de tal abordagem objetiva tornar os sistemas utilizáveis e úteis, e “[...] aumenta a eficácia e a eficiência, aprimora o bem-estar do ser humano, a satisfação do usuário, a acessibilidade e a sustentabilidade; e neutraliza possíveis efeitos adversos do seu uso na saúde, na segurança e no desempenho.” (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2011, p. 3).

Considerando tais evidências, o objetivo do projeto foi construir um *framework* de desenvolvimento de software que maximizasse as chances de sucesso do software desenvolvido. Esse deveria surgir como resultado do desenvolvimento do projeto Portal do Aluno, fazendo uso do mesmo para testar diferentes técnicas e processos. Assim, o desenvolvimento do *framework* e do projeto buscou:

*Promover uma melhor integração das competências multidisciplinares*, por meio do estabelecimento de canais de comunicação mais efetivos entre as competências;

*Capacitar o time em novas técnicas e abordagens*, pelo estímulo à participação em treinamentos e apresentação de seminários entre os membros do time.

*Exercitar a reflexão sobre os processos adotados*, pela discussão do que deu certo e o que deu errado a cada intervalo de projeto.

## 3. Metodologia de Execução

No ano de 2009, o time de software do CCD já utilizava o Scrum, um *framework* para o desenvolvimento de produtos complexos que faz uso de uma abordagem iterativa e incremental, com o objetivo de aumentar a previsibilidade e controlar os riscos do projeto (SCHWABER; SHUTERLAND, 2009). Seu funcionamento já havia sido incorporado pelo time, se fazendo necessárias melhorias da Engenharia de Software. Chegou-se ao entendimento de que a qualidade do desenvolvimento deveria reforçar:

*Testes unitários*. A garantia inicial de que o código desenvolvido continua fazendo o que foi projetado para fazer (BECK, 2003). Buscou-se trabalhar com *Test Driven Development* (TDD) e atingir coberturas superiores a 70% do código. Também se procurou utilizar um padrão uniforme de formatação de código, que permitisse uma leitura uniforme em todo o desenvolvimento;

*Testes automatizados de integração e aceitação*. Garantem que as funcionalidades planejadas foram implementadas (CRISPIN, 2009). São testes de mais alto nível, em geral desenvolvidos por *testers* e que usam um ambiente próximo do ambiente final. Além dos testes automatizados, deveriam ser realizados testes manuais, tanto de regressão quanto exploratórios, para garantir que não existissem problemas;

*Construção (build) automatizada e integração contínua*. Processo automatizado de *build* do software que permite a integração contínua, processo que coleta automaticamente alterações feitas no software, executa todo o processo de *build*, todos os testes unitários, todos os testes de integração/aceitação e faz a estática do código e geração dos respectivos relatórios.

Percebendo os avanços significativos nos processos de software e uma carência com relação à interface e à usabilidade dos produtos, o time de software sentiu a necessidade de investir na experiência do usuário. Para tanto, designers e pesquisadores começaram a adotar princípios do Design Centrado no Usuário que, segundo a Associação Brasileira de Normas Técnicas (2011, p. 8), seguem os seguintes princípios:

- a) o projeto é baseado em um entendimento explícito de usuários, tarefas e ambientes;
- b) os usuários são envolvidos em todo o projeto e desenvolvimento;
- c) o projeto é conduzido e refinado por uma avaliação centrada no usuário;
- d) o processo é iterativo;
- e) o projeto aborda a experiência do usuário como um todo;
- f) a equipe de projeto inclui competências e perspectivas multidisciplinares.

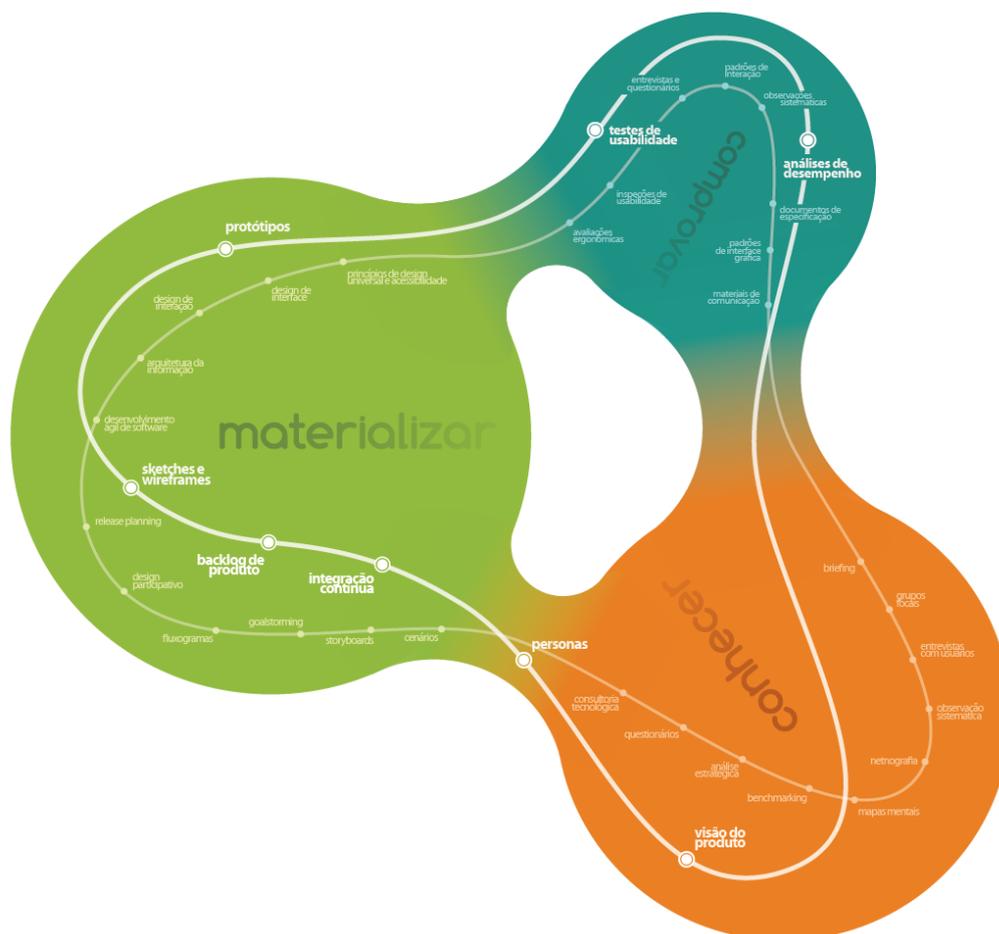
A adoção da abordagem de DCU trouxe desafios ao time, que deveria equacionar as preocupações com a tecnologia junto com as preocupações com os usuários. Contudo, o uso da nova abordagem começou a gerar bons resultados. Para Colfelt (2010, não paginado, tradução nossa), “o DCU requer iterações – fazer design, testar com usuários, refinar, testar com usuários novamente, refinar... repetir até que fique certo. É dessa forma que o Ágil e o DCU podem trabalhar brilhantemente juntos”.

#### **4. Resultados Obtidos**

Os esforços empregados na melhoria contínua do processo de desenvolvimento de software resultaram no *framework Inovação Centrada no Ser Humano*, contribuíram na melhoria da qualidade e produtividade do projeto Portal do Aluno, dentre outros resultados, apresentados a seguir.

##### **4.1. O *framework* Inovação Centrada no Ser Humano**

O *framework Inovação Centrada no Ser Humano* (Figura 1) consiste numa abordagem de desenvolvimento que envolve princípios do Design Centrado no Usuário, do Desenvolvimento Ágil de Software e boas práticas de Engenharia de Software. Ele é resultado da sistematização desses princípios e é composto por três etapas -- conhecer, materializar e comprovar -- que envolvem as técnicas citadas nas tabelas 1, 2 e 3.



**Figura 1 - Diagrama que sintetiza o framework Inovação Centrada no Ser Humano.**

A etapa **conhecer** está relacionada ao entendimento dos objetivos do produto, das necessidades dos usuários e do contexto de mercado. Algumas técnicas dessa etapa são explicitadas na Tabela 1. O resultado dessa etapa expressa o que gostaríamos que o software tivesse, a lente do desejo. Esta é filtrada pela praticabilidade (é possível técnica e organizacionalmente?) e pela viabilidade (é viável financeiramente?) (IDEO, 2011).

**Tabela 1. Principais técnicas da etapa conhecer.**

Técnica	Descrição
Entrevista com usuários	Conversa com um usuário potencial onde se busca compreender motivações, objetivos e frustrações.
Visão de produto	Técnica de construção colaborativa sobre o entendimento do que é o produto, características, concorrentes, recursos, diferencial e público-alvo.
Personas (Figura 3)	Modelos que representam como os usuários se comportam, o que eles pensam, o que querem fazer e porque. São baseadas em observações reais.

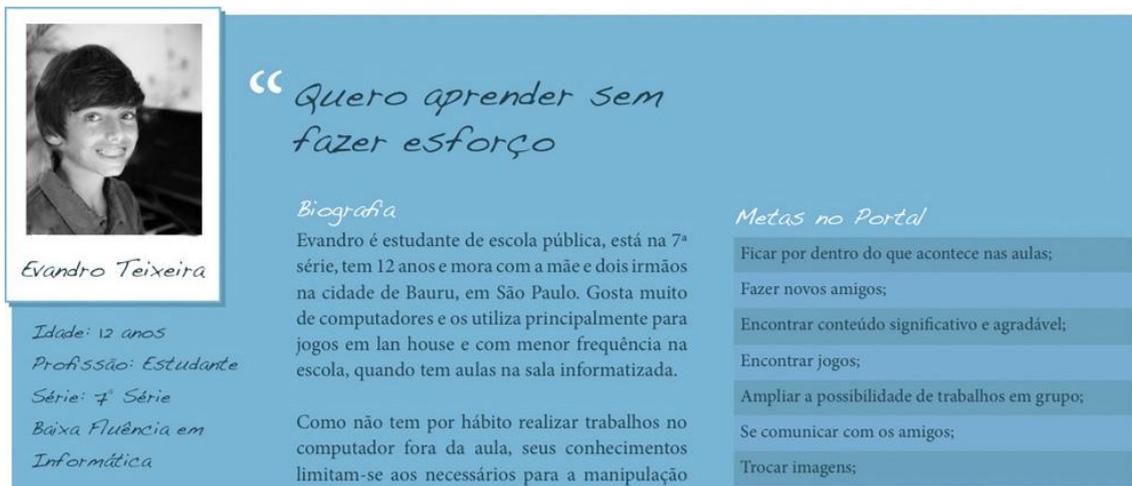


Figura 2 - A Fragmento de persona desenvolvida para o projeto Portal do Aluno.

A fase **materializar** traz técnicas de criação e para a implementação das ideias, apresentadas na Tabela 2.

Tabela 2. Principais técnicas da etapa materializar.

Técnica	Descrição
Cenários	Narrativas que descrevem a interação ideal do usuário com o dispositivo.
Backlog de produto	Uma lista de funcionalidades, tecnologias, correções e melhorias necessárias para desenvolver e lançar o produto com sucesso.
Sketches e wireframes	Sketches são esboços rápidos para comunicar ideias. Wireframes servem para especificar os elementos de conteúdo, navegação e interação de cada tela.
Protótipos (Figura 3)	Versões iniciais de um sistema, com o objetivo de verificar e avaliar seu uso junto ao usuário, antes do efetivo lançamento do produto.
Integração contínua	Processos contínuos que visam melhorar a qualidade ao aplicar o controle de qualidade de maneira frequente e em pequenos pedaços.



Figura 3. Protótipos de telas desenvolvidas para o Portal do Aluno.

A etapa **comprovar** traz técnicas de avaliação da qualidade do software desenvolvido, tanto do ponto de vista tecnológico como do uso (Tabela 3).

Tabela 3. Principais técnicas da etapa comprovar.

Técnica	Descrição
Avaliações ergonômicas	Avalia a qualidade de um dispositivo em adaptar-se a seu operador e às tarefas por ele realizadas.
Testes de usabilidade	Conjunto de métodos e técnicas que tem o objetivo de verificar e avaliar a facilidade que o software ou website possui de ser compreendido e utilizado.
Análises de desempenho	Ferramentas para a avaliação e mensuração da performance de soluções já desenvolvidas.

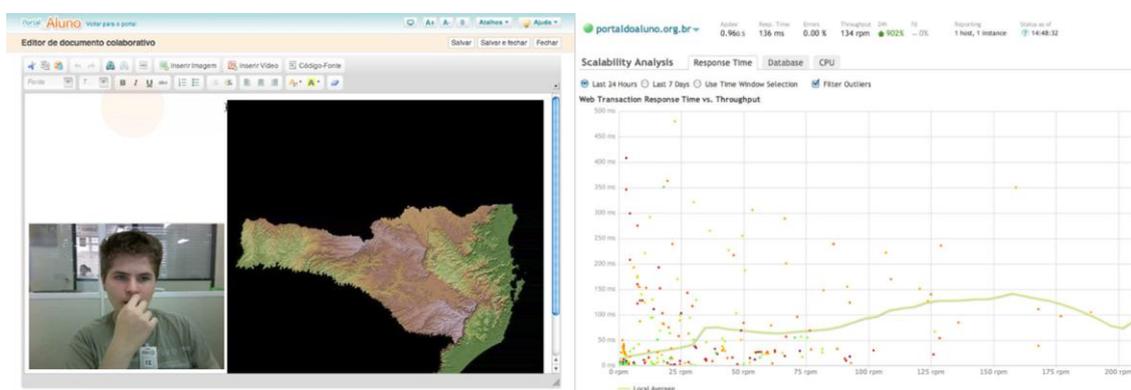


Figura 4. Teste de usabilidade (à esq.) e análise de desempenho (à dir.) realizados no Portal do Aluno.

#### 4.2. Resultados advindos do *framework*

Um claro resultado da aplicação do *framework Inovação Centrada no Ser Humano* pode ser visto no projeto Portal do Aluno. O uso do *Scrum* permitiu que o cliente pudesse acompanhar a evolução do projeto, podendo priorizar o *backlog* ao longo das iterações. Ao trabalhar dessa forma, os usuários-piloto também puderam acompanhar novas funcionalidades e melhorias implementadas, fornecendo *feedback* em curtos intervalos de tempo. Por meio de pesquisas realizadas no Piloto do Portal do Aluno, foi possível observar e ouvir mais de 2000 alunos e professores de 13 escolas de 7 municípios espalhados pelo Brasil. Tais *feedbacks* eram incorporados ao *backlog* como itens de usabilidade a serem trabalhados e priorizados de acordo com sua importância.

Além do projeto Portal do Aluno, outros projetos se beneficiaram do *framework* e o conhecimento adquirido possibilitou que o processo fosse apresentado a visitantes da Fundação CERTI e também em eventos externos, a saber:

- Rede social educacional Portal do Aluno, desenvolvida para o Ministério da Educação (MEC). Disponível em <<http://www.portaldoaluno.org.br>>.
- Ferramenta que permite avaliar o nível de eficiência energética em edificações comerciais, desenvolvida para o Laboratório de Eficiência Energética em Edificações da UFSC (LabEEE - UFSC). Disponível em <<http://www.s3e.ufsc.br>>.

- Palestra sobre Inovação Centrada no Ser Humano realizada para os concorrentes do programa Sinapse da Inovação<sup>5</sup>, edição 2012;
- Artigo *Da pesquisa à apresentação da informação em interfaces digitais: o estudo do exemplo de um ambiente online de aprendizagem colaborativa*, apresentado no 5º Congresso Internacional de Design da Informação, ocorrido em Agosto de 2011;
- Artigo [Desenvolvimento Ágil de um produto Web sob a ótica do Design da Experiência do Usuário](#), apresentado no 9º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, ocorrido em outubro de 2010;
- Palestra sobre o design da experiência do usuário no Portal do Aluno, para estudantes do curso de graduação em Design Gráfico da Universidade Federal de Santa Catarina em novembro de 2009.

## 5. Aplicabilidade dos Resultados

O *framework* *Inovação Centrada no Ser Humano* foi projetado para ser utilizado de maneira modular. Dessa forma, cada projeto pode utilizar as etapas e ferramentas necessárias, dependendo da complexidade do projeto e dos recursos disponíveis.

O objetivo foi alcançado pois se chegou a uma sistematização de filosofias e técnicas que auxiliam o processo de inovação e desenvolvimento de software calcado na produtividade e qualidade tanto em aspectos tecnológicos como em relação à experiência do usuário. Tal sistematização deixa disponível um *framework* que funciona para ser utilizado em projetos futuros, minimizando os riscos do projeto advindos do uso de técnicas desconhecidas ou cujo resultado não é sabido.

## 6. Características Inovadoras

A mais relevante característica inovadora é a união harmônica entre design e engenharia, não explicitada de nas abordagens de desenvolvimento ágil de software e de design centrado no usuário. Dessa forma, uma maneira de trabalhar essas duas abordagens em conjunto traz uma maneira inovadora de pensar o processo de desenvolvimento de software.

Outra característica inovadora é a flexibilidade do *framework*, pensado para projetos de software mas também utilizado em projetos de hardware. Assim, não consiste em uma metodologia engessada que deve ser seguida à risca, mas sim em módulos que funcionam em conjunto ou separadamente. Por fim, outra característica inovadora foi a construção coletiva do instrumento, resultante da reflexão de diversos projetos e da contribuição de competências multidisciplinares do time de software.

## 7. Conclusões e Perspectivas Futuras

Ao observar o processo de desenvolvimento de software utilizado antes do projeto Portal do Aluno e compará-lo com o processo de *Inovação Centrada no Ser Humano*, é possível notar um grande avanço com relação à organização das etapas de projeto e uso de diversas técnicas para variados objetivos. Atualmente o CCD conta com um processo sistematizado que não era disponível anteriormente, o que representa uma ferramenta para maximizar a produtividade e a qualidade do software desenvolvido por meio do aprendizado do que funcionou e o que não funcionou em projetos anteriores.

A metodologia conta com embasamento teórico -- que justifica a adoção dos processos e técnicas -- testado na prática e aprovado, constituindo um instrumento poderoso e pragmático para o desenvolvimento de software customizado. Atualmente o *framework* é considerado um resultado positivo dentro da organização. Como perspectivas futuras, pode-se afirmar que o instrumento não deve ficar estagnado, e sim ser continuamente questionado e adaptado de acordo com novas experiências e mudanças de cenários que se constituem no desenvolvimento de software.

---

<sup>5</sup> Programa que recebe inscrições de ideias inovadoras e apóia financeiramente as 50 melhores para que sejam implementadas. Disponível em <<http://www.sinapsedainovacao.com.br>>. Acesso em 12 mar. 2012.

## 8. Referências Bibliográficas

- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **CEE 126**: ergonomia da interação humano-sistema – Parte 210: Projeto centrado no ser humano para sistemas interativos. [S.l.], 2011. Disponível em: <[http://www.faberludens.com.br/files/ABNT\\_NBR\\_ISO\\_9241-210\\_2011.pdf](http://www.faberludens.com.br/files/ABNT_NBR_ISO_9241-210_2011.pdf)> Acesso em: 22 dez. 2011.
- BECK, Kent. **Test-Driven Development By Example**. Addison-Wesley, 2003.
- CRISPIN, Lisa; GREGORY, Janet. **Agile Testing**: A Practical Guide for Testers and Agile Teams. Addison-Wesley, 2009.
- COOPER, Alan; REIMANN, Robert; CRONIN, David. **About Face 3**: The Essentials of Interaction Design. Indianapolis: Wiley Publishing, 2007.
- COLFELT, Anthony. Bringing User Centered Design to the Agile Environment. **Boxes And Arrows**, 2010. Disponível em: <<http://www.boxesandarrows.com/view/bringing-user>>. Acesso em 05 mar. 2012.
- GARRETT, Jesse J. **The Elements of User Experience**: User-centered Design for the Web and Beyond. 2nd ed. Berkeley: New Riders, 2011.
- IDEO. **Human Centered Design**: Kit de Ferramentas. 2ª edição, 2009. Disponível em: <<http://www.ideo.com/work/human-centered-design-toolkit>>. Acesso em: 8 mar. 2012.
- NORMAN, Donald. Comentário em “**User Experience and Experience Design**” por HASSENZAHN, Marc. 2011. Disponível em: <[http://www.interaction-design.org/encyclopedia/user\\_experience\\_and\\_experience\\_design.html#donald+a.+norman](http://www.interaction-design.org/encyclopedia/user_experience_and_experience_design.html#donald+a.+norman)>. Acesso em: 24 nov. 2011.
- VERSIONONE. **The State of Agile Development**. 2011. Disponível em <[http://www.versionone.com/pdf/2011\\_State\\_of\\_Agile\\_Development\\_Survey\\_Results.pdf](http://www.versionone.com/pdf/2011_State_of_Agile_Development_Survey_Results.pdf)>. Acesso em: 10 mar. 2012.
- SCHWABER, Ken; SHUTERLAND, Jeff. **Guia do Scrum**. 2011. Disponível em <<http://www.scrum.org/storage/Scrum%20Guide%202011%20-%20PTBR.pdf>>. Acesso em: 5 mar. 2012.

## BRISA QPM – Qualimobile Process Management

Kátia Argolo de Castro, André Luís Gobbi Sanches, Johnny Souza, Thamer de Jesus Hatem

BRISA – Sociedade para o Desenvolvimento da Tecnologia da Informação – SCS, Quadra 07 Bloco A, nº 100, Ed. Torre do Pátio Brasil, Sala 822/824 - 70.307-902 – Brasília – DF - Brazil

{katia.castro, andre.sanches, johnny.souza, thamer.hatem}@brisa.org.br

**Abstract.** *This paper describes BRISA's QPM - Qualimobile Process Management, which is a software solution for managing software development projects for mobile devices. This paper analyses the main traits of this specific market, and presents Qualimobile, which is developed from the ground having this market in sight. It also details the features of Qualimobile's first production version, which is BRISA's result for PBQP's Cycle 2011. It provides useful information for anyone which aims to develop for mobile devices.*

**Resumo.** *Este artigo descreve o BRISA QPM – Qualimobile Process Management, que é uma solução de software para gerenciar projetos de desenvolvimento de software voltados para dispositivos móveis. O artigo analisa as principais características desse mercado, e apresenta o QPM, que foi desenvolvido desde concepção para ser voltado a esse mercado. O artigo também detalha as funcionalidades da primeira versão de produção do Qualimobile, que é o resultado apresentado pela BRISA no Ciclo 2011 do PBQP. Ele fornece informações úteis para qualquer pessoa interessada no desenvolvimento de software para dispositivos móveis.*

### 1. Introdução

A forte expansão do mercado de dispositivos móveis, sobretudo de telefones celulares e tablets, acabou por criar uma grande demanda por aplicativos e software básico. Pela necessidade de produção de softwares com alto nível de qualidade e pela importância do fator “*time to market*” nesse nicho de mercado, surge a necessidade da consolidação de boas práticas de desenvolvimento de software para dispositivos móveis e a automação dos processos para incorporar o conhecimento prévio e potencializar o sucesso dos projetos.

Durante a execução de vários projetos de desenvolvimento e testes de produtos para dispositivos móveis, a BRISA percebeu que uma ferramenta que gerenciasse alguns aspectos específicos, enquadrando-se exatamente no fluxo de trabalho desses projetos, seria altamente benéfica. Essa ferramenta não apenas melhoraria a qualidade do produto final, como também traria ganhos de produtividade, uma vez que gerenciar todos esses aspectos sem uma ferramenta adequada é um processo muito custoso para as equipes de especialistas da organização. Entretanto, não foi possível identificar uma ferramenta disponível no mercado que suprisse as necessidades e se adaptasse a esse fluxo de trabalho. Mesmo as ferramentas mais elaboradas têm enfoques diferentes, e não são direcionadas à gestão de qualidade de software para dispositivos móveis. Analisando os vácuos deixados por essas ferramentas tradicionais, as equipes de especialistas da organização conceberam o projeto de uma ferramenta orientada às necessidades dessa área, o BRISA QPM - Qualimobile Process Management.

O desenvolvimento da primeira versão dessa ferramenta constitui o projeto apresentado pela BRISA para o Ciclo 2011 do PBQP. Esse artigo descreve o software desenvolvido, sua aplicabilidade e características inovadoras. O artigo também contém uma descrição detalhada das características especiais desse segmento de mercado, de acordo com práticas de desenvolvimento empregadas pelas maiores empresas da área.

### 2. Objetivos e Justificativa

- 1) Há mais de cinco anos a BRISA atua nesse segmento de mercado, tanto na gestão da qualidade quanto no desenvolvimento de software básico e aplicativos para dispositivos móveis. Com essa experiência, construiu uma solução de gestão desse processo - abrangendo o desenvolvimento e testes até a liberação do produto, que buscou atender aos principais fatores críticos de sucesso do segmento:
- 2) *Time to Market*: o ritmo de inovação é muito alto, e o tempo de desenvolvimento deve ser otimizado. Meses ou mesmo semanas são suficientes para que um novo modelo já esteja ultrapassado.
- 3) Processo de homologação: dispositivos móveis são geralmente comercializados pelas operadoras de telefonia celular. O aparelho precisa ser adaptado e testado para operar corretamente com todas as operadoras para as quais será comercializado. Esse processo de localização e internacionalização envolve,

sobretudo, garantir a qualidade da transmissão de dados, mas também contornar eventuais inconformidades da operadora aos protocolos e padrões, e suporte adequado aos aplicativos que a operadora fornece em seu *SIM card*. Esse processo é dispendioso e repetitivo, mas deve ser seguido para cada novo modelo, de forma a obter a homologação da operadora.

- 4) Respeito a prazos: dispositivos móveis são desenvolvidos por equipes de centenas de pessoas, distribuídas em vários países. Os testes de campo são planejados de forma a testar em uma única viagem vários telefones celulares, para reduzir o impacto dos deslocamentos. O atraso na entrega de uma versão para testes impacta significativamente em todo o cronograma de testes, podendo gerar atrasos, inclusive, em outros modelos. Dada a importância do *Time to Market* nessa área, até mesmo o atraso de versões intermediárias para testes pode causar grandes danos comerciais. Esse fator é agravado pela sazonalidade: grande parte das vendas ocorre em datas próximas a feriados importantes. Como os lançamentos são planejados para logo antes dessas datas, para oferecer os recursos mais recentes, há pouco espaço de manobra no caso de atrasos.
- 5) Dificuldade de atualização: a grande maioria dos dispositivos móveis comercializados não oferece atualização automática de firmware, que é geralmente atualizado apenas em assistências técnicas, quando o equipamento é enviado para conserto. Portanto, a qualidade do software no lançamento é crucial, e define de fato qual será a experiência do usuário, mesmo anos após a compra. Esse fator torna a gestão da qualidade ainda mais crítica nesse mercado.
- 6) Testes de campo: Dispositivos móveis precisam ser testados nos países em que serão comercializados, conectados às operadoras com as quais serão utilizados pelo usuário. Por isso, é muito difícil reproduzir em laboratório o ambiente exato em que o dispositivo será utilizado. Consequentemente, os testes de campo são feitos em etapas do processo de controle de qualidade. Caso *bugs* ou incompatibilidades com redes específicas sejam gerados por alterações posteriores aos testes de campo, eles dificilmente serão identificados. Por isso, durante o processo de qualidade, é preciso adotar uma postura mais cuidadosa e conservadora, procurando reduzir o impacto das alterações ao mínimo necessário. Para que uma equipe tenha tal postura, geralmente investe-se em treinamento e pratica-se a revisão, inspeção e testes extensivos de todas as alterações, especialmente nas fases finais do projeto.
- 7) Rastreabilidade de decisões: os dispositivos móveis geralmente são projetados mundialmente; mas são customizados para cada país e para cada operadora. Geralmente o desenvolvimento é feito por equipes sediadas em país diferente do qual é testado, ligado a outro braço da empresa. Para cada versão, é preciso conciliar informações fornecidas por:
  - a) Projetistas da versão mundial;
  - b) Testadores de interface, que testam os recursos de interação com o usuário, não necessariamente no país em que será comercializado;
  - c) Testadores de campo, que levam em conta os fatores culturais de cada país, e incompatibilidade com serviços utilizados nele;
  - d) Exigências da operadora, que é geralmente a revendedora do equipamento;
  - e) Legislações regionais (por exemplo, em alguns países os aparelhos celulares precisam ser desbloqueados e aceitar qualquer *SIM card*).

Quando especificações de diferentes fontes conflitam, é preciso rastrear porque cada decisão foi tomada, e identificar se determinado comportamento é proposital ou um *bug*. Mesmo pequenas alterações, como mensagens apresentadas ao usuário, devem ser rastreáveis.

Projetos relacionados: geralmente há um projeto por modelo para cada operadora. Consequentemente, há sempre um elevado número de projetos sendo desenvolvidos em paralelo, com prazos e requisitos diferentes, mas que compartilham grande parte do código-fonte. Portanto, a interação entre os projetos é alta – compartilham partes da equipe, e melhorias feitas em um projeto devem ser propagadas aos demais projetos do mesmo modelo – a não ser que conflitem com algum requisito explícito do projeto em questão.

Entretanto, para cada projeto há recursos adicionais a serem testados. Da mesma forma, ao adaptar um modelo a uma operadora, é provável que a adaptação seja semelhante à utilizada em outros projetos para a

mesma operadora. Porém, é preciso testar se a adaptação é adequada para o novo modelo e se não conflita com outras funcionalidades.

Problemas semelhantes ou repetidos: para melhorar o *Time to Market*, é comum que todos os projetos de um mesmo modelo sejam testados em paralelo. Consequentemente, é comum que as mesmas alterações sejam solicitadas em vários desses projetos, possivelmente até em todos. Entretanto, não é possível descartar os problemas duplicados: embora a atividade de desenvolvimento seja a mesma, um problema reportado indica que aquela alteração deve sempre ser testada naquele projeto e também que, caso algum problema contradiga a alteração, que se deve tratar o conflito. Como já citado anteriormente, para esses casos é preciso ter ótima rastreabilidade. Além disso, a semelhança entre projetos torna bastante interessante o emprego de estimativas por histórico, conforme descrito por McConnell (2006).

Alterações constantes de prioridades – quando um novo modelo surge, é comum criar projetos para todas as operadoras que possam se interessar – mesmo as que ainda não demonstraram interesse no modelo (novamente, isso é para acelerar o *Time to Market*). Entretanto, os avanços nas negociações resultam em mudanças de prioridades. Repentinamente, projetos urgentes são postergados por meses, e projetos sem prioridade alguma se tornam urgentes. Isso acarreta a alteração da agenda de atividades de vários grupos (testadores, desenvolvedores, equipes em países distintos). Por isso, é fundamental que seja possível calcular automaticamente as prioridades de cada tarefa, com bases em todos os projetos que são afetados (no plural por causa dos problemas duplicados). Para cada tarefa que se tornou urgente, é preciso identificar quais são os profissionais que tem o perfil necessário e cuja atividade atual pode ser postergada, visando maximizar a chance de cumprir todos os prazos. Essa informação precisa ser obtida rapidamente, justamente por serem casos de urgência.

Previsibilidade – as datas previstas para lançamento de versões para sistemas móveis devem ser altamente confiáveis. Considerando que em cada projeto várias equipes, posicionadas em países e fusos horários distintos, precisam interagir, o atraso em uma data de lançamento obriga várias equipes a redefinirem suas agendas, resultando em alterações nos prazos e prioridades de outros projetos. Consequentemente, caso um projeto tenha atrasos, é conveniente prever esses atrasos com antecedência, para que as equipes tenham mais espaço de manobra para contornar o atraso. Por conseguinte, a precisão de estimativas é particularmente valorizada.

Ciclos curtos - os ciclos de desenvolvimento e teste costumam ser bastante curtos, não raro durando 1 semana ou menos. Os projetos também são curtos, durando no máximo poucos meses. Consequentemente, os processos de criação e encerramento de projetos devem ser ágeis, e não deve tomar da equipe mais tempo do que o necessário.

Gestão do Conhecimento: o desenvolvimento de software para dispositivos móveis exige o conhecimento das principais plataformas utilizadas, como: Infineon, Qualcomm e Mediatek, Android, iOS, Windows Phone e J2ME para aplicativos. Em muitos casos também é preciso conhecimento dos protocolos utilizados, especialmente GSM (definidos pela 3GPP) e os definidos pela Open Mobile Alliance (OMA). Profissionais com experiência nessa área são muito raros, especialmente no Brasil, e com frequência é preciso capacitar os profissionais após a contratação. Mas, dada a alta rotatividade de profissionais de TI no Brasil, é fundamental gerenciar o conhecimento para reduzir o impacto do *turnover* na qualidade e produtividade dos projetos. Para cada tarefa, é crucial identificar rapidamente quais profissionais já executaram tarefas similares. Quando há mudanças em prazos e prioridades dos projetos, é preciso rapidamente realocar toda a equipe, considerando a melhor relação entre a experiência de cada profissional e a prioridade das tarefas que serão temporariamente interrompidas. Conforme discutido acima, tais mudanças são frequentes e podem ocorrer várias vezes em um mesmo dia. Portanto o processo de realocação deve ser leve e facilitado por ferramentas. Da mesma forma, é preciso documentar todas as soluções encontradas, especialmente para problemas recorrentes em mais de um modelo.

Algumas dessas características também podem ser encontradas em projetos que não se destinam a dispositivos móveis. Entretanto, seu conjunto e a prioridade que recebem em relação a outros fatores é o que define a gestão de qualidade para essa área.

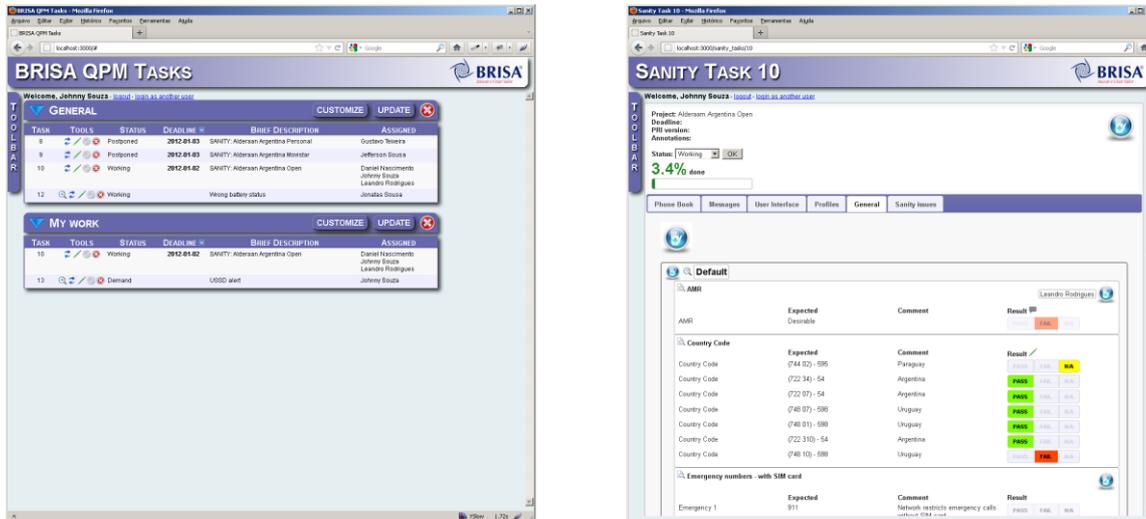
### **3. Metodologia de Execução**

No início do ciclo, a BRISA já havia concluído as etapas de análise, projeto e especificação do QPM. Durante o Ciclo 2011, a solução foi implementada utilizando Ruby on Rails e PostgreSQL. O desenvolvimento seguiu a metodologia de desenvolvimento de software da BRISA, baseada nas melhores práticas do CMMI e do PMBoK para gestão de projetos.

### **4. Resultados Obtidos**

- **Produtos de software gerados**

Durante o Ciclo 2011, a solução BRISA QPM foi implementada, de acordo com a especificação elaborada antes do início do Ciclo. A Figura 5 apresenta duas *screenshots* da versão atual do sistema.



**Figura 5 - Telas do QPM: Gestão de Tarefas (esquerda) e Execução de Testes (direita)**

As principais funcionalidades oferecidas pela versão atual são:

1. Gestão de Tarefas com acompanhamento em tempo real da equipe: A interface do QPM é completamente baseada em AJAX, provendo a todos os usuários informações constantemente atualizadas sobre o andamento de todos os projetos. A tela de gestão de tarefas consolida todas as informações necessárias para rapidamente reprogramar as prioridades de todos os projetos, e rapidamente determinar as realocações de recursos que são necessárias para cumprir os novos prazos.
2. Acompanhamento em tempo real dos testes, com notificação imediata dos desenvolvedores: Durante a execução de qualquer suíte de testes, qualquer teste que falhe gera imediatamente uma nova tarefa no sistema, a ser alocada para algum desenvolvedor. Isso permite que as correções sejam feitas durante a própria execução dos testes, acelerando a correção, e disponibilizando a versão corrigida do software o quanto antes possível. Essa funcionalidade é indispensável para atender a rigorosa exigência de *Time to Market* do mercado em questão. Quando um teste é marcado como "FAIL", o analista de qualidade precisa fornecer uma descrição do problema encontrado, que é imediatamente reportado como um *issue* para os desenvolvedores.
3. Criação automática de testes e regressões com base em históricos: O QPM permite gerar suítes em testes com base em combinações das suítes de testes anteriores. Por exemplo, todos os projetos para um determinado país precisam ser adaptados para utilizar o SCP (Service Control Point) especial do país. Ao criar um projeto para esse país, o sistema automaticamente já customiza a suíte de testes para incluir o conjunto de testes que verifica essa funcionalidade. E caso algum teste falhe no lançamento de um release, o QPM gera automaticamente a suíte de teste de regressão, que contém apenas os testes que falharam.
4. Integração parcial com gestão de requisitos: Para cada projeto, o sistema gera automaticamente a suíte de testes para cada release com base nos requisitos cadastrados pelo gestor do projeto, e também os *issues* já reportados para o projeto em questão.
5. Interface altamente configurável pelo usuário: Permite definir o layout da tela principal de forma específica para cada usuário, permitindo visualizar em uma única tela todas as informações relevantes às atividades do usuário em questão, combinando informações de vários projetos. Por exemplo, um analista vê as atividades alocadas a ele e os prazos correspondentes. Um líder de equipe vê, além das suas atividades, as de seus subordinados (com alertas em caso de atraso). Já os gerentes de projetos podem configurar suas telas para visualizar todas as pendências para os releases de todos os seus projetos. A ferramenta permite criar várias listagens com filtros complexos e ordenação customizada, o que é especialmente útil para profissionais que lidam com vários projetos ou assumem vários papéis distintos.

## **5. Aplicabilidade dos Resultados**

### **5.1 Relevância**

As empresas brasileiras estão começando a se inserir no mercado de dispositivos móveis, com lançamentos nacionais de celulares e *tablets*. Entretanto, para enfrentar a forte competição nesse mercado altamente promissor, tanto no Brasil quanto no mercado mundial, a indústria brasileira precisa adotar os padrões internacionais de qualidade, de eficiência e de "*time to market*". Nesse contexto, a ferramenta proposta destaca-se como facilitadora e disseminadora das melhores práticas de desenvolvimento de software para dispositivos móveis.

### **5.2 Abrangência**

O Brasil, por ser um país geograficamente estratégico, com fronteiras com diversos países da América do Sul, possui grande potencial para instalação de equipes de desenvolvimento e testes de campo. Nesse ponto, o QPM oferece uma importante contribuição no que diz respeito à aderência aos padrões de qualidade internacionais possibilitando o processo de off-shore para diversos países.

### **5.3 Impacto**

A implantação do BRISA QPM oferece melhoria da qualidade e da produtividade no processo de desenvolvimento de software embarcado para dispositivos móveis. Ao permitir controlar de forma integrada todo o processo, aumenta significativamente a probabilidade de sucesso dos projetos de software para dispositivos móveis, com entregas dentro do prazo, custo e qualidade desejada. As organizações que adotarem o BRISA QPM obterão vantagem competitiva e a disseminação em suas equipes da cultura de melhoria contínua de seus processos por meio dos mecanismos oferecidos.

## **6. Características Inovadoras**

O BRISA QPM diferencia-se por integrar regras de negócio do nicho e automatizar processos críticos relevantes. Para exemplificar, podemos citar:

1. Gestão de alterações nos prazos, conforme o avanço das negociações comerciais;
2. Gestão das equipes de desenvolvimento, que são compartilhadas por vários projetos, em função de alterações frequentes das prioridades dos projetos;
3. Gestão do conhecimento viabilizando a realocação das equipes com base nas novas prioridades minimizando o impacto. A ferramenta proposta fará esses cálculos automaticamente. Com isso, se ganha agilidade, especialmente em equipes de maior porte, e obtém-se vantagem competitiva, por aumentar a probabilidade do software ser entregue no prazo, respeitando-se o controle de qualidade.

## **7. Conclusão e Perspectivas Futuras**

Durante o Ciclo 2011 a BRISA desenvolveu a primeira versão de produção do BRISA QPM – Qualimobile Process Management. A aplicação foi construída como uma ferramenta facilitadora para o desenvolvimento e controle de qualidade de software baseado num processo criado e validado dentro da BRISA.

Neste contexto, ele já está disponível como ferramenta para automação dos processos na BRISA e terá sua utilização avaliada para verificar a melhoria introduzida na eficiência, eficácia e economicidade do processo de desenvolvimento de software. Além disto, o QPM poderá ser oferecido ao mercado como uma ferramenta para acelerar a maturação de novos laboratórios de aplicações embarcadas no Brasil. No processo de oferta do QPM ao mercado poderão surgir demandas que levem ao desenvolvimento de versões que possam se adaptar a nichos específicos de desenvolvimento e controle de qualidade de software.

## **8. Referências Bibliográficas**

McConnell, S. (2006) "Software Estimation: Demystifying the Black Art", Microsoft Press, 1ª edição.

# Spider-Reuse: Ferramenta Aplicada à Gerência de Reutilização de Ativos de Software aderente às recomendações dos Modelos/Normas de Qualidade

Fabrcio Medeiros Alho<sup>2</sup>, Kleverton Oliveira Macedo<sup>1</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1,2</sup>

<sup>1</sup>Faculdade de Computação – Instituto de Ciências Exatas e Naturais (ICEN) – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 01, Belém –PA – Brasil

<sup>2</sup>Programa de Pós-Graduação em Ciência da Computação (PPGCC) – Universidade Federal do Pará (UFPA) - Rua Augusto Corrêa, 01 – Guamá – Belém - PA - Brasil

fabrcioalho@globocom, kleverton.macedo@gmail.com, srbo@ufpa.br

**Abstract.** *Spider-Reuse is a free software tool to support a reuse boarding based on MR-MPS model and ISO/IEC12207 and IEEE1517 norms, materialized from a referential model that contemplates activities, profiles and work products applied to the Reuse Management. This paper presents the framework that contemplates this boarding, the systemize workflow of the tool, as well as its intention, the functioning context and the results gotten with the use of this product.*

**Resumo.** *Spider-Reuse é uma ferramenta de software livre para apoio a uma abordagem de reutilização baseada no modelo MR-MPS e nas normas ISO/IEC12207 e IEEE1517, materializada a partir de um modelo referencial que contempla atividades, papéis e artefatos aplicados à gerência de reutilização. Este artigo apresenta o framework que contempla a abordagem proposta, o fluxo sistematizado da ferramenta, bem como o seu propósito, o contexto de funcionamento e os resultados obtidos com a utilização deste produto.*

## 1. Introdução

O desenvolvimento de um software é algo que tem evoluído de uma estrutura monolítica, centralizada e estática para uma estrutura modular, distribuída e dinâmica, tanto no nível de processo quanto no de produto. Neste contexto, o reuso de software vem sendo discutido, estudado e implantado nas empresas que buscam adaptar-se a uma realidade mais qualitativa e competitiva.

Segundo Murta (2008), a reutilização aplicada ao domínio de software parte do pressuposto de que grande parte dos sistemas desenvolvidos não é totalmente nova, eles apenas representam variações de sistemas desenvolvidos anteriormente. Isso faz com que muitas organizações desenvolvam sistemas baseados em determinadas linhas de negócio, denominadas domínio de aplicação.

Para Frakes (2005), o propósito da reutilização envolve melhorar a qualidade do produto e a produtividade durante o seu desenvolvimento. A qualidade do produto é assegurada através da utilização de ativos que já foram previamente testados, aprovados e reutilizados em outros projetos. A reutilização destes ativos faz com que a produtividade aumente, diminuindo o tempo de atendimento ao mercado (*time-to-market*) e aumentando a competitividade da organização. Para atingir estes objetivos é necessário definir uma estratégia que torne a reutilização sistemática e parte das atividades cotidianas da organização, baseando-se em modelos de maturidade de software consolidados e que sejam compatíveis com normas técnicas existentes.

Neste contexto, boas práticas para reutilização de software são encontradas em modelos e normas de qualidade, como o MPS.BR (SOFTEX, 2011a e 2011b) e nas normas ISO/IEC12207 (ISO/IEC, 2008) e IEEE1517 (IEEE, 1999), que norteiam a implantação e sistematização de um processo de gerência de reutilização nas organizações. Para tanto, o sucesso desta implantação e sistematização também depende do uso de ferramentas de software que apoiem as atividades desenvolvidas pelas empresas na aplicação do reuso de software, tornando-as aderentes aos modelos e normas de qualidade citados.

Este trabalho objetiva apresentar a Spider-Reuse, uma solução de software livre para apoio a uma abordagem de reutilização baseada no modelo MR-MPS (Modelo de Referência do MPS.BR) e nas normas ISO/IEC12207 e IEEE1517. A Seção 2 apresenta os objetivos e justificativas do referido projeto. A Seção 3 apresenta a metodologia de execução de desenvolvimento do projeto. A Seção 4 apresenta os resultados obtidos no ciclo atual do projeto. Na Seção 5 será apresentada a aplicabilidade dos resultados obtidos. A Seção 6 apresenta características inovadoras incorporadas ao projeto. E por fim, na Seção 7, são apresentadas as considerações finais.

## 2. Objetivos e Justificativas

Segundo Pietro-Díaz (1991), o objetivo principal da reutilização de artefatos de software é reduzir o tempo e o esforço necessários para construir sistemas. Além disso, a reutilização proporciona benefícios como confiabilidade, facilidade na manutenção, entre outros.

Além das vantagens inerentes à reutilização de software, é notória que a demanda por profissionais e empresas qualificadas na área de desenvolvimento de software tem levado a uma busca globalizada e crescente pelos modelos/normas de qualidade, visando uma maior maturidade nos processos de software instituídos. A criação e a evolução desta maturidade demandam um esforço significativo entre os envolvidos, assim como uma escalada difícil na curva de aprendizado que normalmente apresenta-se, principalmente em processos de software que, apesar de possuírem uma base em modelos de maturidade consolidados e que sejam compatíveis com normas técnicas existentes, não estão consideravelmente disseminados entre as empresas que desenvolvem softwares, diminuindo o acervo de lições aprendidas na implantação e manutenção destes processos.

Existem vários modelos e normas de qualidade que caracterizam e definem a reutilização de software, como o modelo MR-MPS que define um processo de gerência de reutilização no nível de maturidade E (parcialmente definido), cujo propósito envolve gerenciar o ciclo de vida de ativos reutilizáveis. A norma ISO/IEC12207 estabelece uma estrutura comum para os processos de ciclo de vida de software, com terminologia bem definida, que pode ser referenciada pela indústria de software. A norma IEEE1517 é uma extensão da norma ISO/IEC12207 e define procedimentos, atividades, e as tarefas que permitem a identificação, a construção, a manutenção, e a gerência de ativos reutilizáveis.

Por meio de um mapeamento entre o modelo MR-MPS, a norma ISO/IEC12207 e a norma IEEE1517, foi elaborada uma abordagem que auxilia as empresas interessadas na sistematização do processo de gerência de reutilização, contribuindo na aplicação da maturidade para este processo, disponível em [www.spider.ufpa.br/projetos/spider\\_reuse/mapeamento\\_reuso.zip](http://www.spider.ufpa.br/projetos/spider_reuse/mapeamento_reuso.zip). O mapeamento realizado permitiu a identificação e o tratamento de *gaps* existentes entre o modelo e as normas estudadas. A ferramenta Spider-Reuse tem o propósito de apoiar a utilização da referida abordagem, a partir de um fluxo de atividades que contempla as boas práticas resultantes do mapeamento realizado. O fluxo encontra-se, de maneira macro, apresentado na Figura 1, descrito usando as notações do BPMN – *Business Process Modeling Notation*. Um melhor entendimento do fluxo pode ser obtido em [www.spider.ufpa.br/projetos/spider\\_reuse/framework\\_reuso.zip](http://www.spider.ufpa.br/projetos/spider_reuse/framework_reuso.zip).

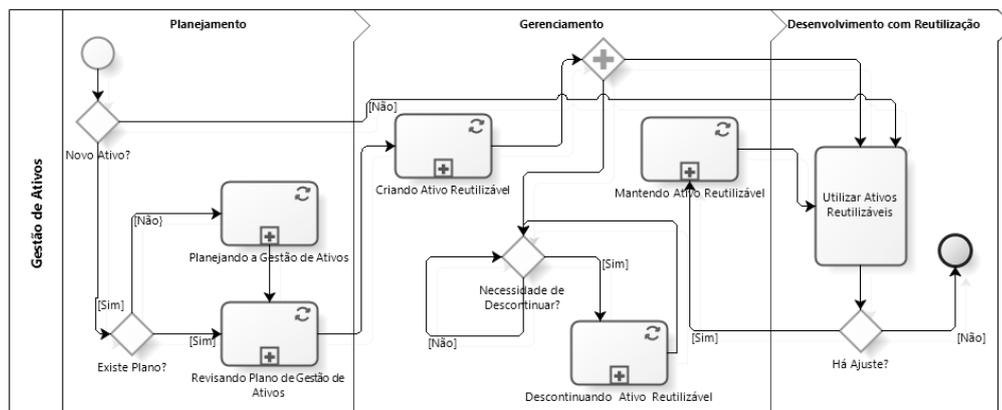


Figura 1 – Macro Fluxo da Abordagem de Reutilização

O grande diferencial da proposta desenvolvida por este projeto em relação aos trabalhos estudados durante este ciclo é o fato do *framework* de processo proposto ter sido definido levando em consideração o estudo de modelos e normas que tratam do reuso de ativos de software, que não estão totalmente alinhados, e como existem diferentes empresas interessadas em instanciar esses diferentes modelos e normas, a adoção isolada de apenas um modelo e norma é inviabilizada.

O projeto, também, não tentou apenas utilizar uma combinação de boas práticas e recomendações entre o modelo MR-MPS e as normas ISO/IEC 12207 e IEEE 1517, mas também demonstrar a equivalência existente entre o modelo e as normas e como atingir esta equivalência, descrevendo os passos necessários para tal.

Por fim, foi desenvolvida, ainda, uma ferramenta de apoio ao processo de gestão de reuso de ativos de software, a Spider-Reuse, que apóia não só a implementação e execução do processo, como também o seu aprendizado por parte dos envolvidos.

## 3. A Metodologia de Execução

Esta seção descreve a metodologia empregada para o desenvolvimento deste trabalho. A realização do trabalho foi dividida nas seguintes etapas:

### **1. Etapa de Estudo Inicial**

- Estudo geral de trabalhos na área de Engenharia de Software, que forneceu uma visão sobre reuso de artefatos de software e das limitações existentes;
- Estudo geral de modelos, normas e guias para processos de software;
- Estudo aprofundado de trabalhos na área de reutilização de software que serviram de fundamentação para a elaboração do estado da arte.

### **2. Etapa de Análise dos Modelos de Qualidade**

- Estudo aprofundado sobre o modelo MR-MPS e das normas ISO/IEC 12207 e IEEE 1517, para obter o entendimento sobre as boas práticas e recomendações dos mesmos;
- Desenvolvimento do mapeamento entre as boas práticas propostas em cada um dos modelos.

### **3. Etapa de Análise do Mapeamento**

- A partir da análise do mapeamento, foi possível a elaboração de um framework de processo para atender as atividades de reuso de artefatos de software;
- Avaliar o framework proposto com o auxílio de especialista na área de reutilização de software;
- Realizar uma análise de aderência das atividades do framework proposto ao mapeamento realizado, através da execução de um questionário contendo perguntas sobre o framework;
- Avaliação das sugestões propostas pelos especialistas participantes da avaliação.

### **4. Etapa de Especificação e Construção da Ferramenta Sistematizada**

- Definição dos requisitos da ferramenta Spider-Reuse, com base no estudo do framework proposto;
- Priorização e validação dos requisitos definidos junto a um consultor experiente certificado pela SOFTEX;
- Elaboração da estrutura arquitetural da Spider-Reuse;
- Implementação da Spider-Reuse a partir dos requisitos elicitados e realização de testes.

### **5. Etapa de Validação da Ferramenta Spider-Reuse**

- Execução de uma experimentação de uso da ferramenta Spider-Reuse em um cenário que simule o cenário real em uma organização adquirente;
- Execução de um questionário contendo perguntas a ferramenta proposta e a sua utilização durante o experimento;
- Análise quantitativa e qualitativa das respostas coletadas pelo questionário;
- Análise do aprendizado obtido com uso da ferramenta através das respostas obtidas com o questionário.

Segundo Silva e Menezes (2001), existem várias formas de se classificar a pesquisa realizada, com base na literatura especializada. Assim, neste contexto, pode-se caracterizar a pesquisa realizada neste trabalho como sendo:

- Quanto à natureza: pesquisa aplicada, por objetivar a geração de conhecimentos para a aplicação prática dirigidos à solução de problemas específicos;
- Quanto à abordagem do problema: uso tanto de pesquisa quantitativa, quanto de pesquisa qualitativa, pois em determinados momentos há necessidade de se traduzir em números, opiniões e informações obtidas com o uso de questionários e em outros momentos, como durante o mapeamento, o pesquisador tende a analisar os dados de maneira indutiva;
- Quanto aos objetivos: pesquisa exploratória e descritiva, proporcionando um maior entendimento do problema, tornando-o mais explícito, envolvendo levantamento bibliográfico, entrevistas com pessoas

com experiência prática, como durante a participação do consultor certificado pela SOFTEX, e por utilizar questionários como forma de verificar as características de uma população;

- Quanto aos procedimentos técnicos – pesquisa bibliográfica, pois a mesma foi elaborada a partir de materiais publicados como artigos de periódicos e eventos, livros e materiais disponibilizados na Internet.

#### 4. Resultados Obtidos

Além do mapeamento entre o modelo MR-MPS, a norma ISO/IEC12207 e a norma IEEE1517, e do *framework* que apresenta uma abordagem para a gerência de reutilização de ativos de software, descritos na seção 2, outros resultados significativos foram gerados neste projeto, conforme segue:

##### 4.1. A Ferramenta Spider-Reuse

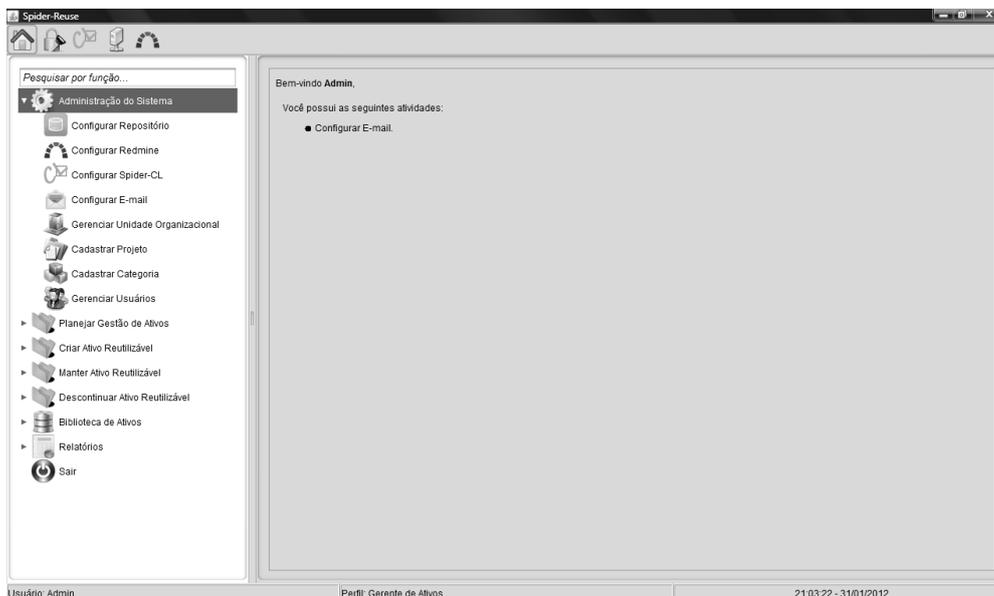
A Spider-Reuse é uma ferramenta de licença GPL – *General Public License* (GNU, 2010) voltada especificamente para apoiar a abordagem apresentada na seção 2, o qual traz uma solução para sistematizar o processo de gestão de ativos de software em uma organização. Esta ferramenta é parte integrante do Projeto SPIDER (Oliveira *et al.*, 2010) ([www.spider.ufpa.br](http://www.spider.ufpa.br)), o qual tem como objetivo criar um *suite* de aplicativos de software livre aderente ao MPS-BR para minimizar os custos de implementação deste modelo.

A Spider-Reuse é uma ferramenta de apoio para instituições que implementam o modelo MR-MPS, é também voltada para empresas que já o adotaram, mas desejam facilitar e aprimorar a gestão de ativos. O seu desenvolvimento foi pautado no uso de ferramentas de software livre, tais como: Sistema Operacional Ubuntu 9.10, Eclipse IDE 3.2, Java 6.0 e banco de dados MySQL 5.0. Encontra-se disponível para download no site do projeto SPIDER, e é:

- **Baseada em Modelos e Guias** – é aderente aos resultados esperados do processo de Gerência de Reutilização do MR-MPS, sendo também aderente às atividades da norma ISO/IEC 12207 e das tarefas da IEEE 1517;
- **Fornecer evidências** – emite relatórios técnicos, gerenciais e os planos envolvidos no reúso de ativos de software. Além disto, possibilita a centralização dos artefatos que evidenciam o ciclo de vida de ativos reutilizáveis. Desta forma, a organização pode fazer uso dos mesmos para que, em uma possível avaliação oficial do MPS.BR ou outro modelo adotado pela organização, os artefatos possam ser utilizados como indicadores de que a organização está implementando o processo de gerência de reutilização de software;
- **Integração com ferramentas** – a ferramenta Spider-Reuse é integrada a ferramentas livres bem estabelecidas no mercado, como exemplo, o Redmine e o SVN. Pretende-se, com isso, que o impacto da utilização da Spider-Reuse em uma organização seja minimizado, pois existe uma boa chance desta organização já ter trabalhado com uma dessas ferramentas, diminuindo, assim, o esforço despendido com o aprendizado necessário para utilizar a ferramenta Spider-Reuse;
- **Possui a customização de critérios objetivos** – com o auxílio da ferramenta Spider-CL, é permitido à organização adquirente criar seus próprios critérios objetivos que serão usados nas atividades que necessitam deste nível de imparcialidade, não estando a organização “presa” a um pré-determinado conjunto de critérios estabelecidos na ferramenta.

A ferramenta possui dois módulos funcionais, divididos em: **Administração**, que abrange a configuração das integrações necessárias, os cadastros básicos como unidade organizacional, categorias, subcategorias e palavras-chaves, usuários, perfis e os controles de acesso ao sistema; o módulo de **Execução**, que envolve do planejamento da gestão do reúso de ativos ao desenvolvimento com reutilização, organizado nas 3 (três) fases propostas pelo *framework* do processo.

A Figura 3 apresenta a tela inicial da ferramenta Spider-Reuse, possuindo um menu superior agrupando funções de configurações de integração com outras ferramentas (credenciais de acesso as ferramentas Redmine, Spider-CL e Repositório de Ativos), um menu lateral que permite a navegação das funcionalidades de administração e de execução.



**Figura 3 – Tela da Ferramenta Spider-Reuse**

## 4.2 Trabalhos Publicados

Ao longo do desenvolvimento deste projeto, algumas publicações de trabalho foram realizadas, conforme lista a seguir:

- **WTDQS 2010 (SBQS 2010):** neste artigo foi apresentada a proposta deste projeto e os resultados esperados;
- **WSL 2011:** neste artigo foi apresentado a ferramenta Spider-Reuse, a metodologia de desenvolvimento utilizada e o contexto das funcionalidades da ferramenta;
- O trabalho deu origem a uma Dissertação de Mestrado do Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Pará, a ser avaliada pela banca examinadora em 2012;

## 4.3 Recursos Humanos Capacitados

O projeto SPIDER objetiva a criação de um suíte de ferramentas de software livre com características adequadas para possibilitar a criação de produtos de trabalho (artefatos que evidenciam a implementação do programa da qualidade organizacional). A motivação para a criação do SPIDER está no fato de que os programas de melhoria da qualidade normalmente definem produtos de trabalho que podem ser melhor gerenciados por meio de ferramentas de software. Para tanto, o SPIDER deve prover ferramentas capazes de apoiar a implementação de programas de melhoria da qualidade organizacional e integrá-las. Uma dessas ferramentas está direcionada para atender à gerência reutilização de software e está sendo tratada neste projeto.

Durante o desenvolvimento deste projeto, foram capacitados: um aluno de Iniciação Científica da UFPA, que participou do desenvolvimento da ferramenta Spider-Reuse; um aluno de Mestrado, que participou da concepção, elaboração e construção do trabalho realizado; e 10 membros do grupo SPIDER, que participaram da avaliação da ferramenta por meio de uma experimentação.

## 5. Aplicabilidade dos Resultados

A relevância dos resultados gerados neste ciclo do projeto é significativa e positiva, pois além de gerar um ferramental de apoio ao processo de gestão de reutilização de ativos de software aderente a modelos e normas de qualidade, que pode ser utilizado por organizações interessadas, a Spider-Reuse também pode ser utilizada pela academia para apoiar no aprendizado do referido processo. Neste sentido, o apoio ao aprendizado foi verificado durante a experimentação da ferramenta, realizada com membros do grupo SPIDER.

Enquanto a abrangência, o referido projeto culminou na criação de uma abordagem para a reutilização de ativos de software, baseada em um *framework* apoiado em um ferramental que possibilita a implantação da gestão de reuso de ativos de software no âmbito organizacional.

Além disso, encontra-se em implementação em 1 empresa de desenvolvimento de software parceira do projeto SPIDER (*Software Process Improvement: DEvelopment and Research*) o uso do *framework* de processo e a

ferramenta Spider-Reuse no contexto da reutilização de software. Posteriormente, pretende-se produzir um artigo relatando a experiência obtida com estas adoções na indústria de software.

## 6. Características Inovadoras

A característica inovadora do projeto é materializada no mapeamento gerado entre o modelo MR-MPS e as normas ISO/IEC 12207 e IEEE 1517, que possibilitou a criação da abordagem aplicada à gerência de reutilização de ativos de software, que culminou no desenvolvimento da ferramenta Spider-Reuse, que apóia a referida abordagem.

## 7. Considerações Finais

A Spider-Reuse tem o intuito de apoiar a gestão de reuso dos ativos de software nas organizações. Desta forma, ela visa ser um referencial na sistematização da gestão de ativos de software.

Os trabalhos futuros estão permeados no relato da utilização da ferramenta em uma organização parceira do Projeto SPIDER para apoio à implementação do MPS.BR, e também na utilização desta ferramenta no apoio ao aprendizado do processo de gerência de reutilização de software, no Programa de Pós-Graduação da Universidade Federal do Pará.

## 8. Agradecimentos

Este trabalho recebeu apoio financeiro da CAPES a partir da concessão de bolsa de Mestrado institucional. Este projeto é parte do Projeto SPIDER-UFPA ([www.spider.ufpa.br](http://www.spider.ufpa.br)).

## Referências

- Frakes, W. B., Kang, K. (2005) "Software Reuse Research: Status and Future". IEEE Transactions on Software Engineering, v. 31, n. 7, pp. 529-536.
- GNU Project. "General Public License". Disponível em: <http://www.gnu.org/licenses/gpl.html>. Último acesso em 25/10/2010.
- IEEE - the Institute of Electrical and Electronics Engineers (1999) "Std 1517-1999 IEEE Standard for Information Technology—Software Life Cycle Processes—Reuse Processes -Description". IHS Standards. USA.
- ISO/IEC - the International Standardization Organization and the International Electrotechnical Commission (2008) "ISO/IEC 12207 - Information Technology – Software Life Cycle Processes". Geneva: ISO.
- Murta, L., Filho, R., Katsurayama, A., Santos, G., Rocha, A. (2008) "A Experiência na Implantação do Processo de Gerência de Reutilização no Laboratório de Engenharia de Software da COPPE/UFRJ". In: Simpósio Brasileiro de Qualidade de Software, Florianópolis, SC. VII, pp. 279-294.
- Oliveira, S. R. B. et al. (2010) "SPIDER - Um Suite de Ferramentas de Software Livre de Apoio à Implementação do modelo MPS.BR". Anais do VIII Encontro Anual de Computação – ENACOMP 2010, Catalão-GO.
- Prieto-Díaz, R. (1991) "Implementing faceted classification for software reuse". Communications of the ACM, Volume 34 Issue 5.
- SILVA, E. L. e MENEZES, E. M. Metodologia da Pesquisa e Elaboração da Dissertação. 3. ed. rev. Atual – Laboratório de Ensino a Distância da UFSC. Florianópolis, Brasil, 2001.
- SOFTEX - Associação para a Excelência do Software Brasileiro (2011a) "MPS.BR – Guia Geral:2011".
- SOFTEX - Associação para a Excelência do Software Brasileiro (2011b) "MPS.BR – Guia de Implementação Parte 3: 2011".

# Apoio Sistematizado de Software Livre para Implementação do Processo de Desenvolvimento de Requisitos do MPS.BR

Ewelton Yoshio Chiba Yoshidome<sup>1</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação – Universidade Federal do Pará (UFPA) - Rua Augusto Corrêa, 1 – Guamá – 66075-110 – Belém – PA – Brasil

renansalesbarros@gmail.com, [srbo@ufpa.br](mailto:srbo@ufpa.br)

**Abstract.** This paper presents the project of a methodology of using a set of tools to cover the Requirement Development process objectives, describing its goals, results, innovations and contribution.

**Resumo.** Este artigo apresenta o projeto da metodologia de uso de um conjunto de ferramentas para contemplar o processo de Desenvolvimento de Requisitos (DRE), descrevendo seus objetivos, resultados obtidos, inovações e contribuições para a área de Engenharia de Software.

## 1. Introdução

A alta qualidade de produtos e serviços é um dos grandes diferenciais na competitividade do mercado nacional e internacional, por esse motivo necessitou-se que empresas brasileiras desenvolvedoras de software oferecessem mecanismos para garantir a qualidade em seus produtos. A partir deste cenário, foi desenvolvido pela SOFTEX (Associação para Promoção da Excelência do Software Brasileiro) o programa MPS.Br (Melhoria do Processo de Software Brasileiro). O MPS.Br tem como objetivo sugerir boas práticas de engenharia de software para a adequação de processos organizacionais direcionados para o desenvolvimento de software. A implementação e avaliação deste modelo é orientado, principalmente, a micro, pequenas e médias empresas brasileiras a custos acessíveis. Este programa divulga o modelo de referência MR-MPS [Softex, 2011].

Entre os processos presentes no MR-MPS está o processo de Desenvolvimento de Requisitos (DRE), que tem como propósito definir os requisitos do cliente, do produto e dos componentes do produto [Softex, 2011; SEI, 2010]. Essas características inerentes ao processo de Desenvolvimento de Requisitos são melhores gerenciadas quando se sistematiza o processo a partir da utilização de ferramentas, pois esta prática pode implicar na redução de esforço e tempo, devido a diminuição da necessidade de documentação, agilizando o processo como um todo.

Neste contexto, surgiu o Projeto SPIDER (*Software Process Improvement - DEvelopment and REsearch*) o qual tem como objetivo de desenvolver um SUITE de ferramentas livres aderentes aos resultados esperados do MPS.BR. O propósito em utilizar ferramentas livres consiste na diminuição dos custos de implantação do processo, devido a gratuidade no seu uso e distribuição [Oliveira *et al.*, 2011]. Um resultado esperado “é um resultado observável do sucesso do alcance do propósito do processo” [ISO/IEC, 2003]

Este artigo encontra-se estruturado em sete seções. A seção 2 apresenta os objetivos e a justificativa para a realização do projeto. Na seção 3 é apresentada a forma que foi realizada a execução da metodologia. A seção 4 mostra os resultados que foram obtidos no decorrer do projeto. A seção 5 exemplifica a aplicabilidade dos resultados obtidos com a execução do Projeto. A seção 6 mostra as características que tornam o projeto inovador. E, finalmente, a seção 7 apresenta as conclusões e as perspectivas futuras para o projeto.

## 2. Objetivos e Justificativa

O objetivo deste trabalho é propor uma metodologia de uso para um conjunto de ferramentas de software livre, contendo práticas que sejam capazes de contemplar os objetivos definidos no processo de Desenvolvimento de Requisitos (DRE) dos modelos de qualidade como CMMI e MPS.BR. Neste trabalho entende-se metodologia como sendo a codificação de um conjunto de práticas recomendadas (por exemplo, os serviços/operações providos a partir do uso de ferramentas), às vezes acompanhada de material de treinamento, programas de educação formal, planilhas, diagramas, tomando como parte um método (processo com uma série de passos, para construir um software) [Pressman, 2006].

A metodologia proposta nesta pesquisa atua como uma extensão à metodologia proposta por Cardias Junior *et al.* (2010), no qual seu trabalho descreve uma metodologia de uso de ferramentas livres para o apoio do Processo

de Gerência de Requisitos do MPS.BR. Pretende-se, a partir desta extensão, utilizar um *kit* de ferramentas em comum que sejam capazes de contemplar os processos de gerência e desenvolvimento de requisitos.

### **3. Metodologia de Execução**

Na primeira etapa de pesquisa foi realizado o planejamento do projeto junto ao coordenador do projeto, que possui experiência em implementação e avaliação no modelo MR-MPS, sendo certificado oficialmente pela SOFTEX. Depois, foi realizada uma pesquisa bibliográfica buscando conhecer os principais conceitos e objetivos abordados pelo processo de Desenvolvimento de Requisitos. Em seguida, foi realizado um levantamento de ferramentas livres que possuíam funcionalidades capazes de contemplar as práticas sugeridas pelos modelos de qualidade. Nesta fase, percebeu-se que o conjunto de ferramentas sugeridos por Cardias Junior *et. al.* (2010) contemplavam grande parte das práticas sugeridas no processo de DRE, o que permitiu estender a idéia proposta em seu trabalho.

Depois de definido o conjunto de ferramentas, iniciou-se a fase de uso das ferramentas de acordo com as boas práticas indicadas pelos modelos CMMI e MR-MPS, analisando pontos fortes, pontos fracos e melhorias a serem contempladas no ferramental de apoio. Com base neste resultado, passou-se à fase de elaboração da metodologia adequando-se as práticas de Gerência de Requisitos sugeridas por Cardias Junior *et. al.* (2010). Durante este período, foram realizadas reuniões semanais com o coordenador para realizar a análise e o acompanhamento do projeto.

### **4. Resultados Obtidos**

O estudo realizado e sua aplicação, trouxe um conjunto de resultados e lições aprendidas:

- O levantamento de um conjunto de ferramentas de software livre para apoio ao processo de DRE, inclusive identificando as possíveis ferramentas candidatas à integração no *suite* de ferramentas a ser proposto no Projeto SPIDER;
- Capacitação e formação de recursos humanos provenientes da certificação do curso de introdução do MPS.BR, e de membros do grupo de pesquisa SPIDER;
- Desenvolvimento de uma metodologia para apoiar a implementação do processo de DRE aderentes à programas de melhoria (MPS.BR, CMMI) por meio do uso de ferramentas de software livre;
- Disseminação do conhecimento apresentado na metodologia em 2 empresas parceiras do projeto SPIDER no contexto de implementação de programas de melhoria da qualidade do processo organizacional. Estas empresas, atualmente, desenvolvem suas atividades de DRE usando o ferramental proposto nesta pesquisa;
- Publicação dos resultados obtidos nesta pesquisa numa conferência internacional realizada na Argentina, o XV Workshop de Engenharia de Requisitos (WER2012);

### **5. Aplicabilidade dos Resultados**

O levantamento e as propostas de solução para a utilização de ferramentas de software livre, permitiram uma melhor capacitação dos recursos humanos na área de qualidade de software. Como consequência, houve aumento da difusão da importância e viabilidade de implantação da Qualidade de Software na região, tanto no meio acadêmico quanto na indústria de software local. Além disso, o Projeto SPIDER está atuando em projetos de consultoria de melhoria de processo nas organizações locais.

Por este projeto estender a metodologia de Gerência de Requisitos [Cardias Junio *et. al.*, 2010], muitas boas práticas de Desenvolvimento de Requisitos poderão ser agregados durante uma implementação de nível G, no que diz respeito a coleta de requisitos. Prova disso é a implementação desta prática em uma empresa de Belém de pequeno porte com interesse na implementação e avaliação do Nível G do MPS.BR, onde práticas de DRE já estão sendo implementadas em conjunto com práticas de GRE, gerando resultados eficazes na execução do processo organizacional.

### **6. Características Inovadoras**

Este projeto destaca-se pelo fato de utilizar um conjunto de ferramentas de software livre para o apoio à implementação do Processo de Desenvolvimento de Requisitos aderentes a programas de melhoria como MPS.BR e CMMI, aliadas a uma metodologia de uso totalmente aderente a estes modelos.

Os procedimentos sugeridos na metodologia desta pesquisa é adaptável para organizações que já utilizam ferramentas para realizar o desenvolvimento de requisitos. Desta forma, não havendo a necessidade de definir uma nova solução de software. Outro aspecto importante é o fato da metodologia basear-se em um conjunto de ferramentas de software livre, o que pode implicar na redução de custo na aquisição e uso das ferramentas ao longo da implementação de programas de melhoria da qualidade do processo organizacional.

## 7. Conclusões e Perspectivas Futuras

As ferramentas analisadas neste trabalho possibilitam a implementação sistêmica do processo de Desenvolvimento de Requisitos, em conformidade com as práticas sugeridas pelo MR-MPS e CMMI-DEV, de forma a estender a metodologia de apoio à Gerência de Requisitos, anteriormente proposta em Cardias Junior *et al.* (2010).

O conjunto de ferramentas proposto neste projeto é capaz de implementar as práticas sugeridas pelo processo de Desenvolvimento de Requisitos dos modelos MR-MPS e CMMI-DEV, desde que seja utilizado de forma planejada, conforme a metodologia apresentada. Também é importante ressaltar que a metodologia apresentada constitui boas práticas de uso das ferramentas para atender às sugestões do MR-MPS e CMMI-DEV sem, no entanto, estabelecer um processo de Desenvolvimento de Requisitos.

Como trabalhos futuros, pretende-se realizar a integração das ferramentas propostas no escopo do Projeto SPIDER para o desenvolvimento de um *Suite* de ferramentas de software livre para apoio aos programas MPS.BR e CMMI, além de um relato do estudo de caso da metodologia, já em execução em 2 empresas parceiras do projeto SPIDER, realizando implementações em ambientes de desenvolvimento das empresas/instituições de desenvolvimento de software para a análise dos pontos fortes, fracos e oportunidades de melhoria.

## 8. Agradecimentos

Este trabalho recebeu apoio financeiro do CNPq a partir da concessão de bolsa de Mestrado institucional. Este projeto é parte do Projeto SPIDER-UFPA ([www.spider.ufpa.br](http://www.spider.ufpa.br)).

## Referências Bibliográficas

- ISO/IEC - International Organization for Standardization / International Electrotechnical Commission. (2003) "ISO/IEC 15504-2: Information Technology - Process Assessment – Part 2 – Performing an Assessment". Geneve.
- Oliveira, S. R. B. *et al.* (2011) "SPIDER – Uma Proposta de Solução Sistêmica de um SUITE de Ferramentas de Software Livre de Apoio à Implementação do Modelo MPS.BR". Revista do Programa Brasileiro da Qualidade e Produtividade em Software. PBQP Software. SEPIN-MCT.
- Pressman, R. S. Engenharia de software. Tradução Rosângela Velloso Penteado. São Paulo: McGraw-Hill, 2006.
- Cardias Junior, A. B. et al. Uma Análise Avaliativa de Ferramentas de Software Livre no Contexto da Implementação do Processo de Gerência de Requisitos do MPS.BR. Anais do WER10 - Workshop em Engenharia de Requisitos. Cuenca, Equador, 2010.
- Associação para Promoção da Excelência do Software Brasileiro (2011) "MPS.BR – Guia Geral: 2011". Disponível em: [www.softex.br/mpsbr](http://www.softex.br/mpsbr).
- SEI – Software Engineering Institute.(2006) "CMMI for Development".Pittsburgh. Disponível em: <<http://www.sei.cmu.edu/reports/06tr008.pdf>>. Acesso: 20 janeiro. 2012.

# Uma Proposta de Processo para Aquisição de Software e Serviços Correlatos Baseada no CMMI-ACQ e no Guia de Implementação do MPS.BR

Julio Cezar Costa Furtado<sup>1</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação (PPGCC) – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 01 – Guamá – Belém - PA - Brasil

julio.furtado@gmail.com, srbo@ufpa.br

**Abstract.** *This paper aims to contribute to a proposal to support the acquisition process, consisting of a process framework and a tool supporting the implementation and execution of the Acquisition process. To achieve these results, a mapping between the MPS.BR Acquisition Guide and the CMMI-ACQ was conducted to identify equivalences between them. The results are part of the SPIDER project, a suite of free software tools to support the implementation of the MPS.BR program processes.*

**Resumo.** *Este trabalho visa contribuir com uma proposta de apoio ao processo de aquisição, composta por um framework de processo e uma ferramenta para apoio à implementação e execução do processo de Aquisição. Para atingir tais resultados, um mapeamento entre as recomendações do Guia de Aquisição do MPS.BR e do CMMI-ACQ foi realizado, visando identificar equivalências entre eles. Os resultados obtidos fazem parte do projeto SPIDER, um suíte de ferramentas de software livre para apoiar a implementação dos processos do programa MPS.BR.*

## 1. Introdução

Muitas organizações têm necessidades específicas que só conseguem ser atendidas a partir do desenvolvimento direto de um produto que atenda a estas necessidades, sendo, muitas vezes, a solução mais viável a terceirização do desenvolvimento do produto (Guerra e Alves, 2004).

A terceirização de produtos e serviços de software é uma tarefa complexa e necessária para estas organizações. Considerando as dificuldades encontradas quanto ao atendimento das exigências para o desenvolvimento dos requisitos de sistemas e visando diminuir os riscos quanto ao cumprimento de estimativas de prazo e custos para seus projetos de aquisição, as organizações adquirentes têm optado pela adoção de processos de aquisição como forma de alcançar os objetivos traçados para seus projetos. Assim, o adquirente tem um maior acompanhamento sobre este contrato de compra, da preparação das necessidades da aquisição, passando pela seleção do fornecedor até a entrega do produto ou serviço.

Neste contexto, existem diversos modelos, guias e normas que visam oferecer as boas práticas para o processo de Aquisição de Software e Serviços Correlatos, como: a norma ISO/IEC 12207 (ISO, 2008), o CMMI-ACQ – *Capability Maturity Model Integration for Acquisition* (SEI, 2010b) e o Guia de Aquisição do MPS.BR (Melhoria do Processo de Software Brasileiro) (SOFTEX, 2011).

O propósito do processo de aquisição é adquirir o S&SC – Software ou Serviço Correlato, que atenda as necessidades expressas pelo cliente e assegure a qualidade do produto de software. Para o processo de aquisição de software, a qualidade é justamente esta adequação do software ou serviço adquirido às necessidades da empresa adquirente, sejam estas necessidades funcionais, não-funcionais ou mesmo gerenciais, como custo e prazo de implantação (Lima, 2004).

Assim, desenvolver software de qualidade assegurada, com elevada produtividade, dentro do prazo estabelecido e sem necessitar de mais recursos do que os alocados tem sido o grande desafio da Engenharia de Software (Fiorini, 1998). Esse desafio é ainda maior nos casos de aquisição de produtos e serviços de software de empresas externas, pois o não cumprimento de um acordo legal pode levar a uma situação litigiosa (Jones, 2007).

Neste contexto, o projeto no qual a experimentação está inserida visa o mapeamento de equivalências entre as atividades do modelo CMMI-ACQ e o Guia de Aquisição do MPS.BR, identificando, desta forma, as melhores práticas definidas para o processo de aquisição de software e serviços, e, com base neste estudo, compor o *framework* de processo que foi desenvolvido no projeto. O conceito de *framework*, no contexto do projeto, retrata a customização de um processo para seguir um ou mais modelos [8].

Por fim, o projeto ainda propôs a ferramenta Spider-ACQ, que tem por objetivo apoiar a implementação e execução das boas práticas constantes no processo de aquisição tendo como base o CMMI-ACQ e/ou o Guia de Aquisição do MPS.BR.

Este artigo encontra-se estruturado em sete seções. A seção 2 apresenta os objetivos e a justificativa para a realização do projeto. Na seção 3 é apresentada a forma que foi realizada a execução da metodologia. A seção 4

mostra os resultados que foram obtidos no decorrer do projeto. A seção 5 exemplifica a aplicabilidade dos resultados obtidos com a execução do Projeto. A seção 6 mostra as características que tornam o projeto inovador. E, finalmente, a seção 7 apresenta as conclusões e as perspectivas futuras para o projeto.

## 2. Objetivos e Justificativa

Em 2003, a Qualidade tornou-se uma das prioridades da SOFTEX (Associação para Promoção e Excelência do Software Brasileiro), elencada como um dos seus Projetos Estruturantes. Desde dezembro de 2003, sete renomadas instituições brasileiras, com competências complementares na melhoria de processos de software em empresas, participam do projeto MPS.BR (Melhoria de Processos de Software Brasileiro): a Sociedade SOFTEX; três instituições de ensino, pesquisa e centros tecnológicos (COPPE/UFRJ, CESAR, CenPRA); uma sociedade de economia mista (CELEPAR); e duas organizações não-governamentais integrantes do Programa SOFTEX (RIOSOFT e Sociedade Núcleo SOFTEX de Campinas).

De uma forma geral, o MPS.BR descreve o que deve ser feito para melhoria gradual de processos, definindo níveis de maturidade que são organizados por processos que possuem objetivos alcançados por resultados esperados os quais são evidenciados em produtos de trabalho. Para tanto, faz-se necessária a utilização de ferramentas de software para possibilitar a implantação do modelo MPS nas organizações. Neste sentido, o número de empresas que adotaram o modelo MPS e que também adotaram outros modelos de referência é significativo, principalmente no conjunto de empresas de maior maturidade (Travassos e Kalinowsky, 2008).

Neste cenário, a introdução da aquisição de S&SC como parte do MPS.BR, tem como finalidade orientar as organizações que adquirem S&SC, por meio de um processo de aquisição onde são descritas as atividades e tarefas fundamentais para a garantia da qualidade do contrato e respectivos produtos e serviços entregues pelo fornecedor (SOFTEX, 2011). Um guia que também fornece orientações sobre as tarefas previstas para se atingir os objetivos e obter os resultados previstos para cada atividade, assim como os produtos requeridos para executar cada tarefa prevista na atividade e os produtos gerados pela execução da atividade (SOFTEX, 2011).

Internacionalmente tem-se o esforço do SEI (*Software Engineering Institute*) com o CMMI (*Capability Maturity Model Integration*), no qual o MPS.BR baseia algumas de suas decisões. O CMMI é uma abordagem para melhoria de processos que provê para as organizações elementos essenciais para melhoria dos processos organizacionais. O modelo ajuda a integrar funções que tradicionalmente são separadas, definir os objetivos da melhoria dos processos e prioridades. Provê informações para a qualidade de processos e um ponto de referência para avaliar o estado atual dos processos da organização (SEI, 2010a).

Assim, o CMMI-ACQ é um modelo de extrema importância e relevância, não apenas pelo seu reconhecimento internacional, mas também por incorporar os trabalhos e boas práticas de diversas organizações pelo mundo, de tal forma a adaptar e melhorar o desenvolvimento do modelo para o uso em organizações adquirentes (SEI, 2010b).

Pode-se considerar, ainda, que as iniciativas de melhoria de processos de software multi-modelos quando não apoiadas e coordenadas apropriadamente podem acarretar em custo e esforço adicionais, aumentando o risco de ineficiências e redundâncias (Baldassarre *et al.*, 2010) e que as lacunas entre os modelos acarretam em problemas operacionais e redução da produtividade (Ferreira *et al.*, 2010). Neste contexto, dá-se a importância deste trabalho em apontar as dificuldades relacionadas à interpretação dos modelos de referência e que recomendações possam ser providas às organizações adquirentes, potencializando resultados esperados e, até mesmo, garantindo sua permanência no mercado.

Desta forma, o alinhamento entre as atividades do modelo CMMI-ACQ e o Guia de Aquisição do MPS.BR visa identificar as melhores práticas definidas para o processo de aquisição de software e serviços, para, assim, compor o *framework* de processo que foi desenvolvido no projeto, beneficiando as organizações interessadas nas recomendações deste modelo e guia, tendo em vista que ao implantar o processo proposto estará, também, mesmo que limitadamente, implantando os processos tomados como base, sem ser necessário um profundo conhecimento nos mesmos.

O projeto teve como principal foco o estudo das orientações fornecidas pelo Guia de Aquisição do MPS.BR, junto com o CMMI-ACQ, e com base deste estudo, a elaboração de um *framework* de processo para aquisição de S&SC que incorpore estas recomendações e, junto a isto, pretendeu-se desenvolver e prover o estabelecimento de uma ferramenta CASE (*Computer-aided Software Engineering*), de software livre devido a disponibilidade do código fonte para organizações adquirentes que planejam modificar a ferramenta para melhor atender suas necessidades, que sistematize o *framework* gerado pelo estudo.

Sendo assim, o objetivo geral do trabalho é a definição de um *framework* de processo para aquisição de S&SC que tome como base as boas práticas e recomendações propostas pelo Guia de Aquisição do MPS.BR e CMMI-ACQ, sendo aderente à estes, de forma a apoiar as empresas nacionais existentes ou em criação que desenvolvam contratos de aquisição, a partir da implantação deste *framework* de processo.

Para o atendimento de tal objetivo geral, os objetivos específicos a seguir foram contemplados:

1. Apresentar um mapeamento entre o Guia de Aquisição do MPS.BR e o CMMI-ACQ, para relacionar as boas práticas presentes em cada modelo, visando garantir a completude do processo de Aquisição;
2. Desenvolver o *framework* de processo com base no mapeamento;
3. Avaliar a aderência do *framework* de processo ao mapeamento realizado;
4. Conceber, elaborar e construir uma ferramenta de software livre que sistematize as atividades definidas pelo *framework* de processo gerado pelo estudo;
5. Analisar a aderência da ferramenta em relação ao *framework* de processo;
6. Analisar o apoio à aprendizagem de processo de software, especificamente sobre a aquisição, provido pela ferramenta, realizando, também, um experimento de uso da ferramenta proposta com foco em avaliar e contribuir com o aprimoramento da mesma por meio da aplicação de um estudo controlado.

### 3. Metodologia de Execução

A metodologia foi composta de cinco fases. A primeira fase foi composta de um estudo do estado da arte sobre os processos de aquisição de software, o que contemplar e quais dificuldades, assim como quais modelos a seguir.

A segunda fase realizou uma revisão teórica sobre os modelos e as normas para aquisição de software adotados para o estudo (Guia de Aquisição do MPS.BR, CMMI-ACQ). E com base nesta revisão, foi realizado um estudo comparativo entre os mesmos para que o mapeamento fosse viabilizado.

O mapeamento considera a equivalência entre os resultados obtidos nos diferentes modelos. Para tal, foram considerados os seguintes itens de cada modelo: (a) tarefas previstas no Guia de Aquisição do MPS.BR; e (b) práticas específicas do CMMI-ACQ. Para cada item relacionado, há uma justificativa, assim como, a relevância do item para a definição do *framework* de processo proposto no trabalho.

A terceira fase iniciou com a definição da proposta do *framework* de processo aderente ao Guia de Aquisição do MPS.BR e ao CMMI-ACQ. O *framework* de processo foi exposto para a avaliação a alguns especialistas da área, a partir de um questionário avaliativo da proposta, visando-se a validação do mesmo e possíveis contribuições técnicas.

Nesta fase, os resultados coletados foram avaliados e priorizados para a incorporação pelo projeto dos possíveis indicadores de pontos fracos e as oportunidades de melhoria.

Com o modelo de processo avaliado, iniciou-se a quarta fase do projeto, onde se deu o desenvolvimento uma ferramenta de software que apóia a sistematização das atividades definidas pelo *framework* de processo. Este forneceu os requisitos para a concepção e elaboração do projeto da ferramenta a partir de diagramas de UML – *Unified Modeling Language*, modelo e dicionário de dados, e as interfaces de comunicação entre os componentes. Ao término da codificação, junto a *release* estável, foi desenvolvido um manual de uso para as organizações interessadas.

Por fim, a quinta fase, tratou-se da avaliação da ferramenta desenvolvida, onde uma experimentação de uso foi realizada com alunos dos cursos de Graduação e Pós-Graduação de Ciência da Computação e Sistema de Informação, ambos da Universidade Federal do Pará. Com isso, pretendeu-se avaliar o apoio ao aprendizado de processos, mais especificamente o de Aquisição, provido pela ferramenta.

### 4. Resultados Obtidos

A seguir são apresentadas algumas contribuições obtidas durante o desenvolvimento deste trabalho:

- Spider-ACQ (Produto de Software) – a ferramenta, é importante por apoiar o processo de aquisição, a partir da sistematização do *framework*. Além de ser um diferencial para as organizações que a adotarem, pois se trata de uma maneira sistematizada de gerenciar seus projetos de aquisição e manter os ativos do projeto centralizados em uma única base, podendo servir de indicadores diretos durante a execução de uma avaliação em uma organização, que, por exemplo, esteja interessada em obter uma avaliação de maturidade pelo programa MPS.BR. Vale ressaltar, ainda, o apoio ao aprendizado dos alunos beneficiados pelo uso da ferramenta;
- *Framework* Spider-ACQ (Processo) – o *framework* de processo é importante pois leva em consideração as boas práticas de dois importantes modelos para processos de software na sua concepção. Assim, tornou-se um *framework* mais fácil de ser aceito, na medida em que ele é baseado em modelos amplamente difundidos e usados. Ressaltando que esta aceitação ainda é embasada pela avaliação realizada pelos especialistas na área de aquisição;

- Artigos Produzidos – uma versão inicial do mapeamento foi publicada no WAMPS – Workshop Anual do MPS.BR, realizada em outubro de 2009 (Furtado e Oliveira, 2009);
- O trabalho também foi exposto durante o WTDQS – Workshop de Teses e Dissertações em Qualidade de Software, dentro do contexto do SBQS 2010 – Simpósio Brasileiro de Qualidade de Software, realizado em junho de 2010 (Furtado e Oliveira, 2010);
- Em 2011, a metodologia de implementação do *framework*, utilizando-se da ferramenta desenvolvida, foi publicada no WGPS – Workshop de Gerência de Projetos de Software, dentro do contexto do SBQS 2011, realizado em junho do mesmo ano (Furtado e Oliveira, 2011a);
- Um artigo descrevendo a ferramenta foi aceito no CBSOFT 2011 – Congresso Brasileiro de Software, dentro da trilha de ferramentas, realizado em setembro de 2011 (Furtado e Oliveira, 2011b);
- Foi submetido para avaliação na Revista RITA – Revista de Informática Teórica e Aplicada do Instituto de Informática da Universidade Federal do Rio Grande do Sul, o artigo Experimentação de Uso da Ferramenta Spider-ACQ: Um Relato de Experiência;
- O trabalho deu origem a uma Dissertação de Mestrado do Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Pará, aprovada pela banca examinadora em outubro de 2011;
- Um dos membros deste projeto participou de capacitação do Curso de Melhoria do Processo de Aquisição de Software (C4- MPS.BR), promovido pela SOFTEX;
- Além disso, encontra-se em implementação em 2 empresas de desenvolvimento de software parceiras do projeto SPIDER (*Software Process Improvement: DEvelopment and REsearch*) o uso do *framework* de processo e a ferramenta Spider-ACQ no contexto da aquisição de software e serviços correlatos. Posteriormente, pretende-se produzir um artigo relatando a experiência obtida com estas adoções na indústria de software.

## 5. Aplicabilidade dos Resultados

O foco deste projeto está em atender as necessidades de empresas adquirentes de software que não possuem um processo bem estabelecido para a tarefa de gerir seus contratos. Pretende-se sanar esta necessidade através da definição de um processo padrão embasado nas principais recomendações internacionais e nacionais (CMMI-ACQ, Guia de Aquisição do MPS.BR) junto com uma ferramenta de apoio para o processo desenvolvido. Ferramenta esta, que também servirá de apoio a instituições privadas que se utilizem ou do CMMI-ACQ ou do Guia de Aquisição do MPS.BR, tendo em vista que o processo proposto por este projeto, a qual a ferramenta provê apoio, é aderente às recomendações destes modelos e normas.

## 6. Características Inovadoras

O projeto destaca-se em inovação por procurar o alinhamento entre as atividades do modelo CMMI-ACQ e do Guia de Aquisição do MPS.BR, visando identificar as melhores práticas elencadas para o processo de aquisição de software e serviços, para assim, compor o processo que será desenvolvido no projeto, beneficiando as organizações interessadas nas recomendações deste modelo e guia, tendo em vista que ao implantar o processo proposto estará também, de certa forma, implantando os processos tomados como base, sem ser necessário um profundo conhecimento nos mesmos. Destaca-se em buscar um apoio ferramental para a execução de processos de aquisição, por meio de uma ferramenta de software livre devido a disponibilidade do código fonte para organizações adquirentes, provendo customização.

## 7. Conclusão e Perspectivas Futuras

O trabalho realizado focou nas boas práticas e recomendações de modelos bem aceitos pela comunidade, sendo o CMMI-ACQ e o Guia de Aquisição do MPS.BR, na medida em que se teve como meta a construção de um *framework* de processo e de uma ferramenta que pudessem ser utilizados em organizações privadas, adquirentes de suas necessidades, promovendo um trabalho que pode minimizar tempo de implementação e de aprendizagem, além de apoiar a gerência dos projetos, visto que o gerenciamento ficaria centralizado na ferramenta proposta.

Como perspectivas futuras espera-se, ainda, uma melhor utilização da ferramenta, onde seria importante que a ferramenta Spider-ACQ fosse integrada a uma ferramenta de Gerência de Configuração, responsável por manter e disponibilizar as várias versões dos produtos de trabalho desenvolvidos durante um projeto de aquisição em uma organização adquirente. A partir desta integração, os produtos de trabalhos armazenados na ferramenta de Gerência de Configuração estarão disponíveis diretamente na Spider-ACQ para que os usuários possam utilizá-los, sem precisar de outra ferramenta para visualizar o documento.

Atender as oportunidades de automatização presentes no contexto da ferramenta, possibilitando que algumas tarefas mais propícias possam ser realizadas sem o esforço do usuário do sistema como por exemplo, a possibilidade de classificar em posições os fornecedores, facilitando assim a avaliação por parte do usuário. Além disto, pretende-se ainda melhorar a comunicação que a ferramenta possa vir a oferecer entre o adquirente e o fornecedor, possibilitando que os produtos de trabalho e os relatórios de andamento possam ser trocados através da Spider-ACQ.

Além das integrações da ferramenta Spider-ACQ com outras ferramentas, pode ser sugerida a manutenção evolutiva da mesma, existindo, por exemplo, a possibilidade de se transformar a ferramenta em uma ferramenta de uso *web* para favorecer o uso do ferramental de apoio em ambientes distribuídos.

A metodologia de uso do *framework* e, conseqüentemente, da ferramenta Spider-ACQ, evidencia uma enorme quantidade de atividades onde é necessário utilizar-se de critérios objetivos, através da aplicação de *checklists*, como forma de selecionar, revisar e avaliar artefatos. Assim, neste contexto, seria interessante a realização de uma pesquisa que vise descobrir quais seriam os critérios mais relevantes às atividades do *framework* e fornecer estas orientações às organizações interessadas em implementar a proposta deste trabalho.

## 8. Agradecimentos

Este trabalho recebeu apoio financeiro do CNPq a partir do Edital MCT/CNPq Nº 70/2008. Este projeto é parte do Projeto SPIDER-UFPA ([www.spider.ufpa.br](http://www.spider.ufpa.br)).

## Referências Bibliográficas

- BALDASSARRE, M.T., CAIVANO, D., PINO, F.J., PIATTINI, M., VISAGGIO, G.. A Strategy for Painless Harmonization of Quality Standards: A Real Case. PROFES 2010, LNCS 6156, pp 395-408, 2010.
- FERREIRA, A.L., MACHADO, R.J., PAULK, M.C. Size and Complexity Attributes for Multimodel Improvement Framework Taxonomy, 36th EUROMICRO Conference on Software Engineering and Advanced Applications, 2010.
- FIORINI, S. T. et al. Engenharia de Software com CMM. Brasport, 1998.
- FURTADO, J. C. e OLIVEIRA, S. R. Uma Proposta de Processo para Aquisição de Software e Serviços Correlatos Baseado no CMMI-ACQ, no Guia de Aquisição do MPS.BR e na Instrução Normativa Nº 4. IX Simpósio Brasileiro de Qualidade de Software, VIII Workshop de Teses e Dissertações em Qualidade de Software - WTDQS, 2010, Belém. Anais do IX Simpósio Brasileiro de Qualidade de Software, 2010.
- FURTADO, J. C. e OLIVEIRA, S. R. Uma Proposta de Mapeamento dos Processos Existentes no Guia de Aquisição do MPS.BR e no CMMI-ACQ. V Workshop Anual do MPS.BR (WAMPS), 2009, Campinas - SP. Anais do WAMPS 2009, 2009. v. 5. p. 164-175.
- FURTADO, J. C. e OLIVEIRA, S. R. Uma Implementação de Práticas de Gerência de Projetos de Aquisição Utilizando Ferramentas de Software Livre. X SBQS - VI WGPS, 2011, Curitiba - PR. Anais do X Simpósio Brasileiro de Qualidade de Software / VI Workshop de Gerenciamento de Projetos de Software, 2011a.
- FURTADO, J. C. e OLIVEIRA, S. R. Apoio Ferramental à Implementação do Processo de Aquisição de S&SC constante em Modelos de Qualidade. Congresso Brasileiro de Software: Teoria e Prática (CBSOFT 2011) - Sessão de ferramentas, 2011, São Paulo - SP, 2011b.
- GUERRA, A. e ALVES, A. Aquisição de Produtos e Serviços de Software. Rio de Janeiro: Campus, 2004.
- ISO/IEC - International Organization for Standardization/ The International Electrotechnical Commission. ISO/IEC 12207 Systems and software engineering– Software life cycle processes. Geneve, 2008.
- JONES, C. Conflict and Litigation Between Software Clients and Developers. Software Productivity Research LLC, 2007.
- LIMA, PAULO. Proposta de um Modelo Simplificado de Aquisição de Software para Pequenas Empresas. Dissertação de Mestrado. São Paulo. Escola Politécnica da Universidade de São Paulo, 2004.
- SEI – Software Engineering Institute. Capability Maturity Model Integration for Development – CMMI-Dev. Versão 1.3. Carnegie Mellon. 2010a.
- SEI. Capability Maturity Model Integration (CMMI) for Acquisition, Version 1.3. Carnegie Mellon, USA, 2010b.
- SOFTEX. MPS.BR - Guia de Aquisição 2011, 2011.

SOUZA, J. e OLIVEIRA, S. Uma Proposta de Apoio Sistemico à Avaliação de Processos com Base no MA-MPS, SCAMPI e ISO/IEC 15504. Workshop de Teses e Dissertações em Qualidade de Software – WTDQS 2010, Belém, 2010.

TRAVASSOS, G.H, KALINOWSKI, M., iMPS - Resultados de Desempenho de organizações que adotaram o modelo MPS. SOFTEX, 2008.

## Avaliação de Usabilidade de Aplicações em Dispositivos Móveis

Bruno Bonifácio, Mario J. S. C. Filho, Tayana Conte

USES - Grupo de Usabilidade e Engenharia de Software  
Instituto de Computação - IComp  
Universidade Federal do Amazonas (UFAM) – Manaus, AM - Brasil  
{brunnoboni,tayana}@icomp.ufam.edu.br, mfilho@gmail.com

**Abstract.** *The evolutionary context in which mobile applications are developed, have made the usability evaluation of these applications an important research area. However, usability evaluations are often avoided by developers or companies due to their lack of experience in the field. For this reason, we present the development of a new usability inspection technique that are specific for evaluating Web mobile applications, the UBICUA (Usability-Based Inspection CUsTomizable Approach). In our solution, we propose a customizable approach that is based on the inspector knowledge, resulting in greater usability defect identification. This article discusses the main findings of experimental studies to definition and evaluation of UBICUA. The aim is to encourage the adoption of a new usability evaluation technology by software engineers in the software development process.*

**Resumo.** *O contexto evolutivo no qual aplicações móveis são desenvolvidas tem tornado a avaliação de usabilidade dessas aplicações um importante campo de pesquisa. No entanto, as avaliações de usabilidade são normalmente evitadas pelos desenvolvedores ou empresas, devido à sua falta de conhecimento na área. Por isso, este artigo apresenta uma nova técnica de inspeção de usabilidade, específica para avaliação de aplicações Web móvel, a UBICUA (Usability-Based Inspection CUsTomizable Approach). Esta nova abordagem é customizada pelo conhecimento do inspetor visando uma maior identificação de problemas de usabilidade. Este artigo discute os principais resultados dos estudos experimentais para definição e avaliação da UBICUA. O objetivo é encorajar a adoção de novas tecnologias de avaliação de usabilidade por engenheiros de software dentro do processo de desenvolvimento dessas aplicações.*

### Introdução

O avanço tecnológico e o sucesso dos serviços disponibilizados através da *Web* têm influenciado a busca por diferentes alternativas de acesso às informações pelos usuários, fazendo este acesso evoluir para o contexto móvel [LUNA *et al.* 2008]. Porém, apesar da crescente demanda de usuários que buscam os dispositivos móveis, devido às vantagens de acesso às informações em qualquer lugar e a qualquer momento, o desenvolvimento de aplicações *Web* móvel não é tarefa simples [FLORENCE *et al.* 2009]. Isto porque aplicações *Web* em ambientes móveis possuem diferenças determinadas principalmente por algumas características, como: as características específicas dos dispositivos, as limitações do ambiente de comunicação sem fio e o contexto dinâmico de uso [WEISS, 2005]. Estas características podem influenciar no design de criação de aplicações *Web* móvel. Por essa razão, a usabilidade é um fator muito importante em aplicações para dispositivos móveis, pois pode melhorar a qualidade de uso e minimizar a dificuldade de interação dos usuários finais.

Nesse contexto, técnicas de avaliação de usabilidade específicas para aplicações *Web* móvel podem beneficiar o processo de desenvolvimento de aplicações em tais dispositivos. Assim, várias técnicas específicas têm sido propostas. No entanto, estudos comparativos mostram que as metodologias de avaliação de usabilidade para este tipo de aplicação não estão consolidadas [BETIOL, 2005; BASTIEN *et al.* 2007], uma vez que as características diferenciais das aplicações móveis fazem com que novos aspectos devam ser considerados em avaliações de usabilidade.

Este fato motivou a elaboração de uma nova técnica de inspeção, chamada UBICUA (*Usability-Based Inspection CUsTomizable Approach*), específica para avaliação de usabilidade de aplicações *Web* em dispositivos móveis. Esta técnica é customizada através de critérios específicos, como o conhecimento do inspetor, para que possa ser empregada pelos próprios engenheiros de software durante o processo de desenvolvimento. O objetivo é que com esta customização possa-se identificar maior quantidade de defeitos se comparado às abordagens

existentes. Este artigo discute os principais resultados dos estudos experimentais para definição e avaliação da UBUICUA.

O restante deste artigo está estruturado da seguinte forma. A seção 2 expõe os objetivos e a justificativa deste projeto. A seção 3 apresenta a metodologia utilizada para sua execução. A seção 4 discorre sobre os resultados encontrados. A seção 5 apresenta a aplicabilidade dos resultados. A seção 6 exhibe as características inovadoras desta pesquisa. Por fim, a seção 7 conclui o artigo.

### Objetivos e Justificativa

Segundo BETIOL (2005), além da utilidade, disponibilidade e custo, um dos principais elementos para a aceitação do usuário, no que diz respeito a aplicações móveis, refere-se à usabilidade das interfaces e os dispositivos utilizados. Apesar da evolução tecnológica que dos dispositivos móveis, muitos usuários ainda encontram dificuldades na utilização de aplicações móveis [ANATEL, 2011]. Por isso, a usabilidade das aplicações móveis é um fator de grande importância para o diferencial competitivo e sucesso do produto.

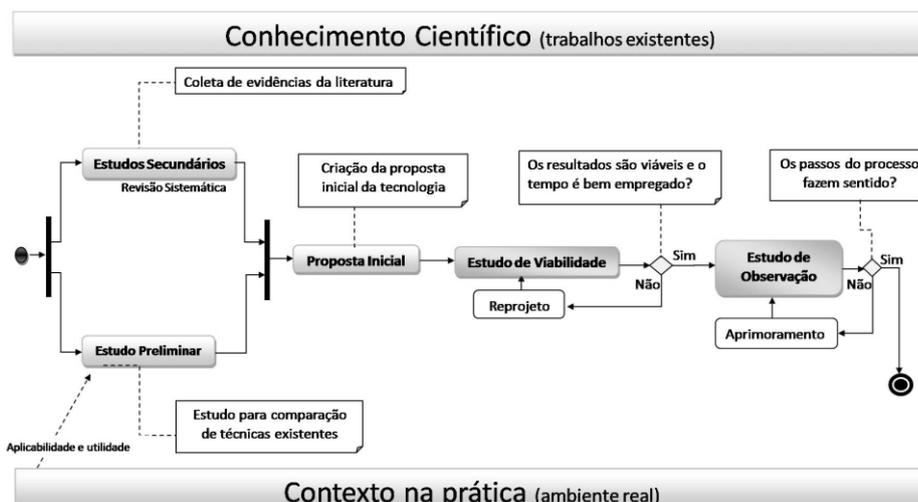
Deste modo, a questão de pesquisa que norteia este trabalho consiste na possibilidade de definição de tecnologia para inspeção, específica para avaliação de usabilidade de aplicações móveis, que apóie a obtenção de maior qualidade nas aplicações Web para dispositivos móveis.

Neste sentido, o objetivo principal deste artigo consiste em apresentar os resultados dos estudos experimentais para a definição e avaliação técnica UBUICUA, utilizando uma abordagem metodológica baseada em experimentação. O propósito final é que a técnica possa ser utilizada por todos os envolvidos nos projetos de desenvolvimento de aplicações móveis, desde os desenvolvedores até os especialistas em usabilidade. E com isso, difundir o conhecimento sobre estudos experimentais para apoiar a transferência segura de novas tecnologias de software para indústria, maximizando os benefícios da adoção destas tecnologias de Engenharia de Software.

### Metodologia de execução

A metodologia utilizada nesta pesquisa, conforme a Figura 1, é uma extensão da abordagem proposta por SHULL *et al.* (2001) e MAFRA (2006) para a introdução de tecnologias de software na indústria de forma segura. A abordagem consiste na condução de estudos experimentais como forma de determinar o que funciona ou não durante a avaliação da tecnologia proposta, desde sua definição até sua transferência para indústria.

A abordagem desta pesquisa possui cinco etapas: 1) Execução de estudos secundários: para identificar, avaliar e interpretar todos os resultados relevantes de acordo com a questão de pesquisa desejada; 2) Estudo Preliminar: para geração de indícios que possam auxiliar na definição da nova tecnologia, complementando assim os dados obtidos nos estudos secundários; 3) Criação de uma versão inicial da nova tecnologia, utilizando a base de conhecimento adquirida nos passos anteriores; 4) Execução de estudo de viabilidade: para caracterizar a proposta e verificar sua viabilidade de uso; 5) Execução de estudo de observação: onde foi feito um estudo comparativo entre técnicas de inspeção de usabilidade específicas para aprofundar a compreensão de como a proposta é aplicada, possibilitando seu aprimoramento.



**Figura 1 - Visão geral da metodologia experimental adotada, baseada nas propostas por de Shull et al. (2001) e Mafra et al. (2006).**

## **Resultados obtidos**

### **Estudos Preliminares para a Definição da UBICUA**

Esta primeira fase é detalhadamente discutida em [BONIFÁCIO et al., 2010a], deste modo apresentaremos somente os principais resultados. O primeiro estudo experimental apresenta os trabalhos identificados através de um estudo baseado em revisão sistemática de literatura. O objetivo deste estudo foi identificar características das abordagens da literatura técnica, para que a definição da nova tecnologia pudesse estar baseada em evidências da literatura. Pelo rigor e confiança nos resultados, em comparação a revisões informais de literatura, foi decidido à utilização de uma abordagem de revisão sistemática de literatura chamada mapeamento sistemático (MS) [KITCHENHAM et al. 2010].

O mapeamento sistemático, conduzido nesta pesquisa, contribuiu para a escolha de uma abordagem baseada em inspeção de usabilidade, devido ao baixo custo que este tipo de avaliação possui. Além disso, este estudo contribuiu para caracterização de fatores de usabilidade relevantes para computação móvel, que podem influenciar no contexto de uso, tais como:

- **Perfil do Usuário:** características pessoais do usuário que podem influenciar na interação, por exemplo, flexibilidade, presteza e experiência;
- **Características do Dispositivo:** características específicas dos dispositivos que podem afetar na sua usabilidade total como: plataforma, a interface da aplicação e o tempo de inicialização e;
- **Características do Canal de Comunicação:** o ambiente de comunicação sem fio e as condições que podem afetar a usabilidade da aplicação.

Outra característica importante utilizada para definição de novas propostas de avaliação deste porte, corroborada com a revisão, foi à utilização de métodos-base, adaptados para o contexto móvel. Para apoiar a escolha do método-base mais adequado, foi realizado um estudo preliminar com técnicas de avaliação de usabilidade específicas para aplicações *Web*. O objetivo deste estudo foi identificar, através de estudo comparativo, qual técnica melhor se adaptava para o paradigma móvel e que pudesse ser utilizada como método-base na definição da UBICUA.

A partir dos resultados deste estudo, descritos mais detalhadamente em BONIFÁCIO et al. (2010b), foi possível perceber as características de cada técnica considerando indicadores de eficiência e eficácia na detecção de defeitos de usabilidade. Com base nos resultados obtidos, optou-se por utilizar o método de Avaliação Heurística. A AH utiliza diretrizes que indicam o que deve ser avaliado na interface da aplicação, podendo aumentar o equilíbrio entre o tempo gasto e o número de defeitos de usabilidade encontrados e com isto diminuir o tempo gasto de inspeção.

Os resultados do Mapeamento sistemático em conjunto com o estudo preliminar, auxiliaram na definição da nova técnica de avaliação de usabilidade, chamada UBICUA (*Usability-Based Inspection CUsomizable Approach*).

### **A técnica UBICUA: Usability-Based Inspection CUsomizable Approach**

O propósito da técnica UBICUA é que esta possa ser empregada pelos próprios engenheiros de software durante o processo de desenvolvimento, considerando fatores que possam afetar o contexto de uso e a usabilidade da aplicação. A partir dos resultados obtidos no MS e no estudo preliminar, foi elaborada a nova técnica na qual é testada a adoção de fatores de usabilidade relevantes em computação móvel, identificados no MS, através de itens de verificações customizáveis de acordo com tais fatores. A proposta da técnica é detalhar cada item de verificação com base em cada fator de usabilidade.

Em cada fator de usabilidade, a técnica UBICUA utiliza itens de verificação que apontam um fator que deve ser avaliado e relaciona cada item de verificação com as heurísticas de NIELSEN (1994). O objetivo desta relação

com as heurísticas é para detalhar melhor o que está sendo avaliado, através de itens de verificação incluídos em todas as heurísticas.

Para cada item de verificação na técnica UBICUA, foram listados itens que apontam o que deve ser avaliado em relação ao fator de usabilidade referente. Um exemplo disso é apresentado na Figura 2, onde o mesmo item de verificação é detalhado de forma diferente, com base no fator de usabilidade que está em foco.

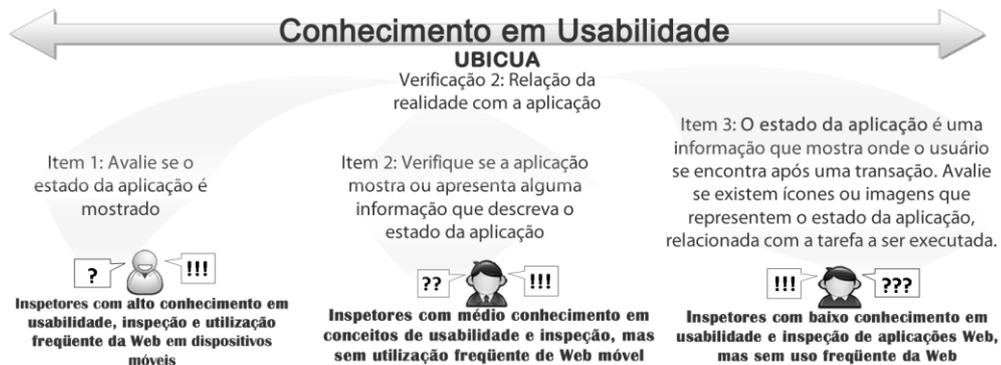
Além dos fatores de usabilidade, a técnica UBICUA é customizada dependendo do nível de conhecimento em avaliações de usabilidade. Nem todo engenheiro de software possui conhecimento em avaliação de usabilidade, por isso pode ser necessário dar mais direcionamento aos mesmos durante a inspeção.



**Figura 2. Itens de verificação modificados de acordo com foco em cada fator.**

Desta forma, dependendo do inspetor, a técnica pode apresentar um maior ou menor detalhamento dos itens para auxiliar na detecção de defeitos de usabilidade. Um exemplo disto é mostrado na Figura 3, que mostra a UBICUA customizada de acordo com o nível de conhecimento de cada inspetor. Assim, novos itens são adicionados para facilitar a busca por defeitos na aplicação. Para inspetores com alto conhecimento é apresentado somente o “item 1 da verificação 2”. Para inspetores com médio conhecimento, a técnica mostra além do “item 1” um outro item detalhando o que está sendo avaliado na aplicação, e o mesmo é feito para inspetores de baixo conhecimento.

**Figura 3. Extrato da UBICUA e a customização para o mesmo item de acordo com o conhecimento do inspetor.**



Para essa customização, o conhecimento do inspetor foi dividido de acordo a freqüência de uso de serviços Web. Dessa forma, foram considerados: o conhecimento em usabilidade e em inspeção de software, além do nível de experiência de uso (freqüência de utilização) de Web móvel, para cada inspetor.

### Estudos Experimentais para Avaliação da UBICUA

Para apoiar a definição e o aprimoramento da técnica UBICUA, foram feitos até o momento dois estudos experimentais. No primeiro, descrito em BONIFÁCIO *et al.* (2011), buscou avaliar a viabilidade da UBICUA em comparação com o método-base a Avaliação Heurística (AH). O segundo estudo, descrito em BONIFÁCIO *et al.* (2012), foi conduzido através de um estudo de observação comparando a UBICUA com outra técnica específica para avaliação de usabilidade de aplicações Web móvel, chamada Mobile Heuristic (MH).

Os resultados dos estudos experimentais apontaram indícios da viabilidade da técnica UBICUA para detecção de defeitos de usabilidade. No primeiro estudo a UBICUA obteve eficiência média de 5,11 defeitos por hora, e eficácia individual média de 20,6% em relação à AH, que obteve como indicador de eficiência de 3,62 e eficácia de 10%. E quando comparada a outro método específico para avaliar aplicações *Web* móvel, a MH, no segundo estudo a UBICUA obteve um indicador de eficiência de 11,29 defeitos por hora, e eficácia individual média de 21,92%. Este percentual é acima dos resultados obtidos da MH que obteve como indicador de eficiência de 7,38 e eficácia de 12,71%. De acordo com esses resultados, é possível ter indícios de que o uso da UBICUA, para inspeção de aplicações *Web* em dispositivos móveis é viável para a detecção de problemas de usabilidade.

#### **Artigos publicados**

- O artigo “Aplicando Técnicas de Inspeção de Usabilidade de Aplicações para Avaliar Aplicações Móveis” [BONIFÁCIO *et al.*, 2010b] foi apresentado no IX Simpósio de Fatores Humanos em Sistemas Computacionais (IHC 2010), realizado em Belo Horizonte, MG.
- O Pôster acadêmico “Avaliação de Usabilidade de Aplicações em Dispositivos Móveis” [BONIFÁCIO *et al.*, 2010c] foi apresentado no IX Simpósio de Fatores Humanos em Sistemas Computacionais (IHC 2010), realizado em Belo Horizonte, MG.
- O artigo “UBICUA: A Customizable Usability Inspection Approach for Web Mobile Applications” [BONIFÁCIO *et al.*, 2011] foi apresentado no IADIS International Conference (WWW/INTERNET), realizado no Rio de Janeiro, RJ.
- O artigo “Usabilidade de Aplicações *Web* Móvel: Avaliando uma Nova Abordagem de Inspeção através de Estudos Experimentais” [BONIFÁCIO *et al.*, 2012] foi aceito para apresentação no XV Congresso Ibero-Americano em Engenharia de Software (CIBSE 2012), a ser realizado em Buenos Aires, Argentina.

#### **Outros resultados: Recurso Humano Capacitado**

Projeto de Iniciação Científica de Andreza Morgana F. V. de Castro: Bacharel em Engenharia da Computação. Universidade Federal do Amazonas - UFAM

Projeto Final de Conclusão de Curso de Mario J. S. C. Filho: Bacharel em Ciência da Computação. Universidade Federal do Amazonas – UFAM

Resultados deste projeto fazem parte da pesquisa de dissertação de Bruno Araújo Bonifácio: Mestrando em Informática. Universidade Federal do Amazonas – UFAM

#### **Aplicabilidade dos resultados: Benefícios para a Indústria e Academia**

Embora a pequena amostra nos estudos seja uma limitação desta pesquisa, foram utilizados em cada estudo métodos estatísticos e baseados em análise qualitativa visando aumentar a validade dos resultados obtidos. Além disso, outro benefício com esta pesquisa consiste em incentivar uso de avaliações de usabilidade por profissionais da indústria, dentro do processo de desenvolvimento, e com isso obter maior aumento de uso de aplicações móveis.

Além dos benefícios para a indústria, vale ressaltar os benefícios acadêmicos deste estudo. Este projeto contribuiu para o “corpo de conhecimento” sobre avaliação de usabilidade de aplicações *Web* em dispositivos móveis, onde foram identificadas tendências, práticas e oportunidades de pesquisa que podem ser utilizadas para avaliar este tipo de aplicação, disponível em BONIFÁCIO *et al.* (2010a). Além disso, outra contribuição está relacionada a estudos experimentais com o propósito de disseminar o conhecimento sobre sua utilização para avaliar e evoluir uma nova tecnologia em Engenharia de Software e IHC [BONIFÁCIO *et al.* 2010b; BONIFÁCIO *et al.* 2011].

#### **Características inovadoras**

Como característica inovadora deste projeto, temos a elaboração de diretrizes e recursos de apoio à realização de inspeções de usabilidade em ambientes acadêmicos: descrição de atividades a serem realizadas para a realização de inspeção de usabilidade, realizar uma avaliação de usabilidade de aplicação *Web* e planilhas para relato de discrepâncias.

Além disso, os resultados qualitativos foram úteis para ratificar a customização da UBICUA, uma vez que os resultados geraram indícios de que a customização da técnica, de acordo com o conhecimento do inspetor, é que pode aumentar a identificação de problemas de usabilidade em relação às abordagens existentes. E com isso incentivar o uso de técnicas de avaliação de usabilidade por todos os envolvidos nos projetos de desenvolvimento de aplicações móveis, desde os desenvolvedores até os especialistas em usabilidade.

### **Conclusões e trabalhos futuros**

Este artigo a técnica UBICUA, uma técnica específica para avaliação aplicações *Web* em dispositivos móveis e apresentou uma série de estudos para definição da técnica. Esta nova técnica está sendo avaliada através de uma metodologia que utiliza os resultados de estudos experimentais, visando aumentar a eficiência e eficácia na detecção de defeitos de usabilidade comparada às abordagens existentes.

Como trabalho futuro pretende-se, ainda: (1) melhorar a técnica através de uma análise detalhada da influência de cada verificação na lista final de defeitos detectados; (2) investigação sobre a influência de cada fator de usabilidade, por exemplo, se a eficiência e a eficácia podem ser melhoradas se cada inspetor se concentra em apenas um fator e; (3) a replicação da experiência na indústria.

Ao apresentar os estudos experimentais, realizados para avaliação da UBICUA, almeja-se contribuir com o compartilhamento do conhecimento científico. Espera-se com isso, incentivar a indústria de software *Web* móvel a realizar avaliações de usabilidade com maior frequência no desenvolvimento deste tipo de aplicação.

### **Agradecimentos**

Os autores agradecem pelo apoio financeiro do CNPq sob os processos 556944/2009-7 e 575808/2008-0.

### **Referências Bibliográficas**

- ANATEL 2011 - Agência Nacional de Telecomunicação (Gov.). Disponível em: <<http://www.anatel.gov.br/Portal/exibirPortalNoticias.do?acao=carregaNoticia&codigo=20824>>. Acesso em maio de 2011.
- BASTIEN, J.M.C., 2010a, "Usability testing: a review of some methodological and technical aspects of the method", *International Journal of Medical Informatics*, v. 18, pp. e18-e23.
- BETIOL, A.D.A., 2005, "Usability Testing of Mobile Devices: A Comparison of Three Approaches". In: *Human-Computer Interaction - INTERACT 2005*, v. v. 3585, pp. 470-481, Rome - Italy.
- BONIFÁCIO, B., CRUZ, M.J., CONTE, T., 2010a, Extração e análise de características para avaliação de usabilidade em aplicações móveis baseadas em evidências da literatura, Grupo de Usabilidade e Engenharia de Software (USES-UFAM), Relatório Técnico UsES-RT-003-2010, disponível em: <[www.dcc.ufam.edu.br/uses/index.php/publicacoes/cat\\_view/69-relatorios-tecnicos](http://www.dcc.ufam.edu.br/uses/index.php/publicacoes/cat_view/69-relatorios-tecnicos)> acesso em: fevereiro de 2012.
- BONIFÁCIO, B., SANTOS, D.V., ARAÚJO, C., VIEIRA, S., CONTE, T., 2010b, "Aplicando Técnicas de Inspeção de Usabilidade para Avaliar Aplicações Móveis". In: *IX Simpósio de Fatores Humanos em Sistemas Computacionais (IHC 2010)*, v. V. 1., pp. 189-192., Belo Horizonte - MG, Brasil.
- BONIFÁCIO, B., OLIVEIRA, H.A.B.F.D., CONTE, T., 2010c, "Avaliação de Usabilidade de Aplicações em Dispositivos Móveis". In: *IX Simpósio de Fatores Humanos em Sistemas Computacionais (IHC 2010)*, v. 1, pp. 269-270, Belo Horizonte, MG, Brasil.
- BONIFÁCIO, B., FERNANDES, P., OLIVEIRA, H.A.B.F.D., CONTE, T., 2011, "UBICUA: A Customizable Usability Inspection Approach for *Web* Mobile Applications", *IADIS*, Rio de Janeiro, RJ - Brasil.
- BONIFÁCIO, B. FERNANDES, P., SANTOS, F., OLIVEIRA, H.A.B.F.D., CONTE, T., 2012, "Usabilidade de Aplicações *Web* Móvel: Avaliando uma Nova Abordagem de Inspeção através de Estudos Experimentais", *XV Congresso Ibero-Americano em Engenharia de Software (CIBSE 2012)*, Buenos Aires, Argentina (Aceito para publicação).
- FLORENCE, B.-F., JENNY, F., HEINRICH, H., 2009, "Evaluation of User Interface Design and Input Methods for Applications on Mobile Touch Screen Devices", *Springer-Verlag*, v. 30, pp. 243 – 249, Uppsala, Sweden.
- KITCHENHAM, B.A., BUDGEN, D., PEARL BRERETON, O., 2010, "Using mapping studies as the basis for further research - A participant-observer case study", *Journal of Information and Software Technology*, v. 53, n. 6, pp. 638-651.

- LUNA, E.R., PANACH, J.I., GRIGERA, J.N., ROSSI, G., PASTOR, O., 2010, "Incorporating usability requirements in a test/model-driven *Web* engineering approach", *Journal of Web Engineering* v. 9, n. 2, pp. 132-156.
- MAFRA, S.N., BARCELOS R.F., TRAVASSOS G.H.L, 2006 "Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software." In: XX SBES, pp. 394 – 410, Florianópolis, SC, Brasil.
- NIELSEN, J., 1994, "Heuristic evaluation." In: NIELSEN, J., MACK, R.L. (eds), *Usability Inspection Methods*, John Wiley and Sons
- SHULL, F., CARVER, J., TRAVASSOS, G.H. 2001, "An empirical methodology for introducing software processes" In: *ACM SIGSOFT Software Engineering Notes* v. 26, p. 288-296
- WEISS, S., 2005, "Handheld usability: design, prototyping and usability testing for mobile phones", *ACM*, v. 12, pp. 367-369, Salzburg, Austria.

# Apoio Sistêmico na Avaliação de Produtos de Software no Contexto do MEDE-PROS

Gleyson do Nascimento Gama<sup>1</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1,2</sup>

<sup>1</sup>Faculdade de Computação – Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 01 – Guamá – Belém - PA – Brasil

<sup>2</sup>Programa de Pós-Graduação em Ciência da Computação – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 01 – Guamá – Belém - PA - Brasi

gleysongama@gmail.com, srbo@ufpa.br

**Abstract.** *The software quality is a research field of software engineering aimed at developing studies to contribute to the software development improvement released in the market. To measure the software quality, it should be submitted a quality assessment according to quality models and standards. The MEDE-PROS is a quality model for evaluating software products that, in its current phase, is used to perform quality assessments manually. This paper presents a Spider-PQ, a software system that aims to help software product evaluators in the preparation, execution and reporting of an MEDE-PROS evaluation.*

**Resumo.** *A qualidade software é um campo de pesquisa da engenharia de software destinada a desenvolver estudos que contribuam para a melhoria do desenvolvimento dos softwares lançados no mercado. Para medir a qualidade de um software, este deve ser submetido a uma avaliação de qualidade de acordo com normas e modelos de qualidade. O MEDE-PROS é um modelo de qualidade para avaliar produtos de software que, em sua fase atual, é utilizado para realizar avaliações de qualidade de forma manual. Este trabalho apresenta a Spider-PQ, um sistema de software que visa auxiliar avaliadores de produto de software no processo de preparação, execução e geração de relatório de uma avaliação MEDE-PROS.*

## 1. Introdução

Existem diversas formas de se verificar o nível de qualidade apresentada por um software, a nível internacional é importante destacar as normas de qualidade especificadas pelas *International Organization Standardization* – ISO e a *International Electrotechnical Commission* – IEC que apresentam características importantíssimas, como confiabilidade, eficiência e manutenibilidade. A nível nacional, e não menos importante, o MEDE-PROS é um modelo de qualidade que foi desenvolvido para avaliar a qualidade de produto de software, fornecendo resultados qualitativos e quantitativos sobre a qualidade desse produto, sendo que esses resultados sejam compreensíveis, aceitáveis a quaisquer das partes interessadas, e confiáveis (Colombo e Guerra, 2009b).

No contexto da qualidade software, surgiu o Projeto SPIDER (*Software Process Improvement: DEvelopment and Research*), que objetiva apresentar alternativas viáveis com características adequadas para possibilitar a criação de produtos de trabalhos derivados dos resultados esperados descritos nos objetivos do MPS.Br – Melhoria do Processo de Software Brasileiro (Oliveira *et al.*, 2011).

Como sub-projeto associado aos produtos de trabalho desenvolvidos pelo Projeto SPIDER, a pesquisa desenvolvida neste trabalho durante o ciclo 2011, atuou na análise e provimento de melhoria no processo de cadastro e execução de avaliação utilizado pelo MEDE-PROS. Atualmente uma avaliação de qualidade realizada pelo modelo de qualidade é realizado por meio do preenchimento de formulários em papel. Com a utilização de uma ferramenta sistematizada, espera-se que as avaliações sejam realizadas on-line e que os resultados possuam um repositório para as avaliações de produtos de software.

Este artigo encontra-se estruturado em sete seções. A seção 2 apresenta os objetivos e a justificativa para a realização do projeto. Na seção 3 é apresentada a forma que foi realizada a execução da metodologia. A seção 4 mostra os resultados que foram obtidos no decorrer do projeto. A seção 5 exemplifica a aplicabilidade dos resultados obtidos com a execução do Projeto. A seção 6 mostra as características que tornam o projeto inovador. E, finalmente, a seção 7 apresenta as conclusões e as perspectivas futuras para o projeto.

## 2. Objetivos e Justificativa

Pressman (2006) afirma que os usuários estão mais interessados no uso do software, na sua funcionalidade, no desempenho e nos efeitos que o uso possa produzir na sua empresa ou organização, em outras palavras, cliente valoriza que o software responda às suas necessidades. Em um conceito técnico o usuário deseja que o software tenha qualidade.

O MEDE-PROS possui uma metodologia de avaliação que está estruturada em uma lista de verificação. Com essa lista, o avaliador executa sua avaliação de qualidade sobre um produto de software e ao final de avaliação é disponibilizado para as partes interessadas o relatório da avaliação contendo informações de aspecto positivo e informações de aspectos a serem revistos pelo produtor do software (Colombo e Guerra, 2009a). Atualmente, essa prática de avaliação é realizada de forma manual, onde o avaliador preenche os campos da lista de verificação por meio de caneta e os dados da avaliação são arquivados em papel.

Neste contexto, este projeto teve o objetivo de propor um sistema que se adéque ao modelo de avaliação da qualidade do produto de software apresentado no MEDEPROS. De acordo com esse cenário, o projeto de pesquisa desenvolveu durante o ciclo 2011 o sistema de software Spider-PQ, que foi concebido objetivando auxiliar o avaliador de produto de software no processo de avaliação, assim como de proporcionar para a entidade gestora da avaliação uma base de dados contendo informações referentes às avaliações realizadas (Gama e Oliveira, 2011).

## 3. Metodologia de Execução

O projeto foi executado pelo coordenador (certificado pelo CTI no modelo MEDE-PROS e vinculado ao LTS – Laboratório de Tecnologia de Software da UFPA credenciado pelo CTI para realizar avaliações usando o MEDE-PROS), em conjunto com o pesquisador alocado nas dependências da Faculdade de Computação, do Instituto de Ciências exatas e Naturais (ICEN) da UFPA, onde, a partir de um planejamento definido inicialmente, foram agendadas reuniões periódicas, para alinhar avanços nas diversas etapas do projeto, desde a análise do modelo de avaliação MEDE-PROS até a especificação da ferramenta proposta.

A cada reunião o coordenador atribuiu para o pesquisador tarefas relacionadas a cada etapa do projeto, baseado em um planejamento inicial do projeto elaborado em conjunto com o pesquisador. Para melhor compreensão a Figura 1 apresenta o modelo de desenvolvimento iterativo para concepção da Spider-PQ.

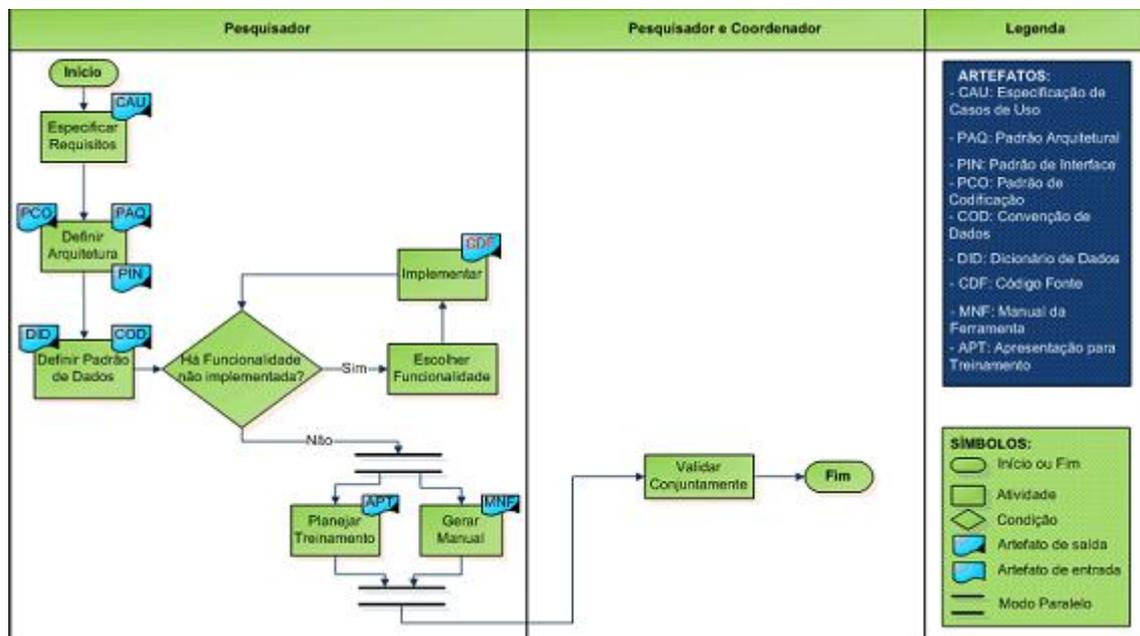


Figura 6. Fluxo de desenvolvimento da ferramenta Spider-PQ.

#### 4. Resultados Obtidos

Durante o ciclo 2011, o projeto alcançou como resultados:

- a concepção, elaboração e construção da Spider-PQ, ferramenta de software livre de apoio à avaliação da qualidade de produtos de software de acordo com o modelo de avaliação MEDE-PROS, contemplando atividades como:
  - gestão dos critérios e *checklists* de avaliação constantes nas diferentes versões do modelo MEDE-PROS;
  - gestão de usuários e produtos a serem avaliados;
  - planejamento, controle, monitoração e execução de avaliação do produto, seguindo o ciclo metodológico proposto pelo modelo MEDE-PROS;
  - geração de relatórios sobre o planejamento, *checklist* e resultado da avaliação;
- elaboração de um Manual do Usuário da ferramenta Spider-PQ;
- capacitação e melhoria contínua dos conhecimentos teóricos e práticos dos estudantes dos programas de graduação e pós-graduação da UFPA nas áreas de Engenharia e Qualidade de Software;
- apoio para que as empresas da região que desenvolvam atividades inovadoras em termos tecnológicos de impacto comercial ou social com qualidade, a partir da implantação da ferramenta desenvolvida e da metodologia de uso definida;
- participação efetiva da comunidade acadêmica para provimento de melhoria dos processos e produtos organizacionais, e a disseminação de programas de melhoria de qualidade de software nas empresas locais.

Ainda durante o ciclo 2011, o desenvolvimento do projeto rendeu a defesa do Trabalho de Conclusão de Curso em Bacharelado em Ciência da Computação da UFPA do pesquisador juntamente com a orientação do coordenador do Projeto SPIDER, e a publicação de 2 artigos científicos nos seguintes eventos nacionais: IX Encontro Anual de Computação (ENACOMP 2011); e I Simpósio de Informática e Geotecnologia de Santarém (SIGES 2011).

#### 5. Aplicabilidade dos Resultados

Como foi apresentado na seção 1 deste trabalho, o sistema Spider-PQ é um software desenvolvido para ter seu funcionamento sobre a plataforma *web*, garantindo, assim, as seguintes aplicabilidades práticas para as partes interessadas em uma avaliação:

- Mobilidade: a ferramenta pode estar disponível de qualquer lugar, necessitando apenas da Internet e de um navegador *web*;
- Investimento reduzido: baixo custo de instalação e manutenção de versões, ou seja, não há a necessidade que a ferramenta seja instalada no computador, facilitando, assim, mudanças de versões da ferramenta;
- Gerenciamento centralizado: o servidor do sistema *web* encarrega-se de promover todo o processamento das requisições vindas dos computadores clientes;
- Liberdade: como consequência de um gerenciamento centralizado as aplicações podem se tornar a cada nova versão um sistema mais flexível, segura e de alto desempenho.

De acordo com o cenário apresentado, o foco do projeto esteve associado ao atendimento das atividades relacionadas à melhoria da qualidade dos produtos de software desenvolvidos no mercado. Empresas desenvolvedoras de software, em nível de abrangência federal, estadual e municipal serão contempladas visto que o projeto se trata do uso de ferramentas de software livre.

## 6. Características Inovadoras

O projeto destaca-se em construir soluções de software livre para apoio ao processo de avaliação da qualidade de produtos de software submetidas ao MEDE-PROS.

A confecção da ferramenta foi possível, pois foram consideradas no escopo do projeto apenas soluções tecnológicas de software livre. A utilização de ferramentnas livres possibilita a redução dos custos de aquisição ferramental para as organizações e flexibilizando de forma considerável o tempo das prestações de serviços por empresas de consultoria neste contexto.

## 7. Conclusão e Perspectivas Futuras

A Figura 1 apresenta o fluxo de desenvolvimento da Spider-PQ, no entanto as etapas de “Planejar Treinamento” e “Validar Conjuntamente” não foram concluídos devido o prazo exíguo que os membros do projeto tiveram para finalizar os trabalhos ao longo do ciclo 2011.

A versão atual Spider-PQ é desprovida de um mecanismo que realize o controle de versão das avaliações cadastradas. A implantação de um mecanismo de controle de versão torna o uso da ferramenta mais direcionada com o modelo MEDE-PROS, pois o mesmo possui diferentes versões que são lançadas à medida que surgem alterações nas normas de qualidade das quais o MEDE-PROS é dependente, ou quando novos resultados de pesquisa resultam em itens a serem incorporados ou removidos da lista de verificação do modelo MEDE-PROS. Sendo assim, as atividades relacionadas ao desenvolvimento da Spider-PQ terão sua continuidade dentro dependências do Projeto SPIDER.

Além disso, entende-se como atividade futura a disponibilização para os membros do CTI mantenedores do modelo MEDE-PROS para avaliação da sistematização proposta na ferramenta no que tange a adequação e efetividade da metodologia de avaliação. Contatos já foram realizados com estes membros e os trabalhos de validação do ferramental encontram-se em planejamento.

## Referências Bibliográficas

- Colombo, R. M. T., Guerra, A. C. (2009a) “Tecnologia da Informação: qualidade de produto de software”. Brasília: PBQP, 2009. 429p.
- Colombo, R. M. T., Guerra, A. C. (2009b) “Qualidade de Produto de Software”. Brasília : Ministério da Ciência e Tecnologia, 2009. Disponível em: <<http://www.mct.gov.br/index.php/content/view/306537.html>> Acesso em: 19 de março de 2012.
- Gama, Gleyson do e Oliveira, Sandro Ronaldo Bezerra. (2011). “Spider-PQ: Uma Abordagem Ferramental de Apoio à Avaliação de Produtos usando o MEDE-PROS”. Simpósio de Informática e Geotecnologia de Santarem – SIGES 2011.
- Oliveira, S. R. B. *et al.* (2011) "SPIDER – Uma Proposta de Solução Sistêmica de um SUITE de Ferramentas de Software Livre de Apoio à Implementação do Modelo MPS.BR". Revista do Programa Brasileiro da Qualidade e Produtividade em Software, SEPIN-MCT. 2ª Edição. Brasília-DF. Disponível em: [http://www.mct.gov.br/upd\\_blob/0216/216673.pdf](http://www.mct.gov.br/upd_blob/0216/216673.pdf).
- Pressman, Roger S. (2006). “Engenharia de Software”. São Paulo: McGrawHill, 6ªEd.

## **RBTtool - Ferramenta de Teste Baseado em Riscos**

Cristine Gusmão<sup>3</sup>, Ellen Souza<sup>1,2</sup>, Júlio Menezes Jr.<sup>1</sup>, Liliane Silva<sup>1</sup>, Nielson Pontes<sup>1</sup>

<sup>1</sup>Centro de Informática - Universidade Federal de Pernambuco (UFPE) Recife - PE - Brasil  
{eprs,jvmj,lss4,njsj}@cin.ufpe.br

<sup>2</sup>Bach. Sistemas de Informação – Universidade Federal Rural de Pernambuco (URFPE) Unidade Acadêmica de Serra Talhada – PE – Brasil  
[eprs@uast.ufrpe.br](mailto:eprs@uast.ufrpe.br)

<sup>3</sup>Núcleo de Telessaúde – Universidade Federal de Pernambuco (UFPE) Recife – PE – Brasil  
cristine.gusmao@nutes.ufpe.br

**Abstract.** *Risk-based Testing (RBT) is an approach that allows the identification of software features that are more likely to fail. Although studies prove the effectiveness of RBT approach, test engineers still find it difficult to apply the approach in practice and this is due mainly to lack of support tools. In this context, this paper proposes the RBTTool, an open source CASE tool available in both desktop and web versions, to support the RBT approach, in order to provide higher quality and productivity in the testing activity, by automating repetitive tasks, therefore reducing the probability of inconsistent results.*

**Resumo.** *A abordagem de teste baseada em riscos ou Risk-based Testing (RBT) permite identificar as funcionalidades do software que possuem maior probabilidade de apresentar falhas. Apesar de estudos comprovarem a eficiência da abordagem RBT, os engenheiros de teste ainda encontram dificuldades em aplicá-la na prática e isto se deve, principalmente, a ausência de ferramentas de apoio. Nesse contexto, este trabalho propõe a RBTTool, uma ferramenta CASE, de código aberto, disponível nas versões desktop e web, de apoio ao RBT, com o objetivo fornecer uma maior qualidade e produtividade na atividade de teste, através da automação de tarefas repetitivas, diminuindo a possibilidade de inconsistência nos resultados.*

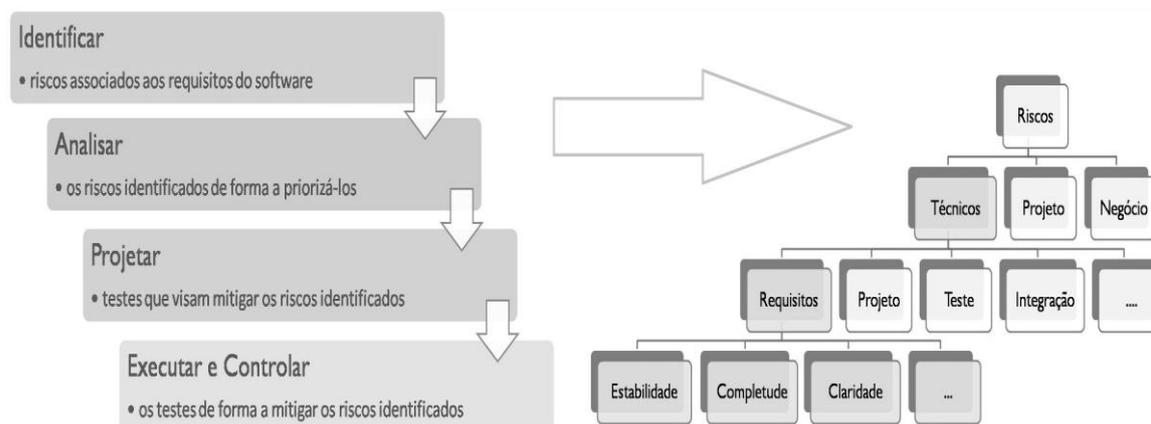
### **1. Introdução**

As organizações tem crescentemente despertado para a importância da atividade de teste de software como forma de melhorar a qualidade dos seus produtos e manterem-se competitivas no mercado. Também, a complexidade das tecnologias utilizadas e dos produtos de software produzidos tem aumentado, tornando-se indispensável a utilização de processos, métodos, técnicas e ferramentas que permitam a realização de testes de software de maneira sistematizada e com fundamentação teórica [Delamaro et al., 2007].

Contudo, o teste demanda muito esforço e recursos, principalmente financeiros, chegando a custar até cinquenta por cento do valor inicial de um software em desenvolvimento [Kaner et al., 1999]. Neste sentido, a abordagem de teste baseada em análise de riscos, também conhecido como *Risk-based Testing* (RBT) tem sido fortemente debatida e estudada como uma proposta promissora para reduzir os custos e esforço relacionados ao teste de software e antecipar a descoberta de defeitos, melhorando a qualidade do produto final.

Por meio da avaliação dos riscos associados aos requisitos funcionais e não- funcionais, a abordagem RBT indica as funcionalidades do software que possuem maior probabilidade de apresentar falhas e, conseqüentemente, precisam ser melhor testadas. Assim, o processo de teste de software é melhorado a partir da adoção de atividades do gerenciamento de riscos, permitindo a priorização de esforços e alocação de recursos para as funcionalidades do software que necessitam ser testadas mais cuidadosamente.

A Figura 1 apresenta as atividades do gerenciamento de riscos e de testes que compõem a abordagem RBT, além da classificação dos riscos técnicos, da classe de requisitos tratados pela a abordagem. A classificação utilizada está baseada na classificação proposta pelo SEI (2006).



**Figura 1. Atividades e riscos técnicos de requisitos tratados pela abordagem RBT**

Apesar de estudos [Chen, 2002; Amland, 1999; Souza et al., 2009] comprovarem a eficiência e facilidade de uso da abordagem RBT, os engenheiros de teste ainda encontram dificuldade em aplicá-la na prática [Goldsmith, 2006], e isto se deve ao fato de que as atividades de gerenciamento de riscos são abstratas, necessitam de treinamento para sua aplicação [SEI, 2006], e não são de conhecimento da maioria dos engenheiros de teste. Além disso, os riscos necessitam ser identificados, analisados e controlados manualmente para cada um dos requisitos do software, tornando a abordagem RBT inviável para produtos de software com um número elevado de funcionalidades.

Nesse contexto, apresentamos a *RBTool*, uma ferramenta CASE, de código aberto, disponível nas versões *desktop* e *web*, para o Teste de Software baseado em Riscos. Assim como as demais ferramentas CASE, a *RBTool* tem como objetivo fornecer uma maior qualidade e produtividade na atividade de teste, através da automação de tarefas repetitivas, diminuindo a possibilidade de inconsistência nos resultados.

A organização deste artigo segue da seguinte forma: após esta seção introdutória, onde é feita uma explanação geral sobre a problemática que motiva a construção da *RBTool*, são apresentados os objetivos principal e específicos (Seção 2), a metodologia de execução do projeto (Seção 3), os principais resultados obtidos (Seção 4), bem como a aplicabilidade dos mesmos (Seção 5). Na seqüência, são apresentadas as características inovadoras do projeto (Seção 6) e as conclusões e trabalhos futuros (seção 7). Por fim, as principais telas do sistema são apresentadas (Apêndice).

## 2. Objetivos e Justificativas

Para o ciclo atual, o **objetivo principal** desse projeto consistiu no desenvolvimento da versão *web* para *RBTool*, incluindo todas as funcionalidades da versão *desktop*, além de novas funcionalidades. Contudo, **objetivos específicos** também são apresentados:

- Elaborar estudo de novos métodos de identificação, análise e controle de fatores de riscos para o teste de software;
- Incluir novas funcionalidades à ferramenta relacionadas com os métodos de identificação, análise e controle de fatores de riscos estudados no objetivo anterior;
- Permitir um controle mais eficiente dos riscos identificados, através da elaboração de gráficos e imagens;
- Possibilitar o gerenciamento dos casos de teste planejados e executados, controlando os riscos que foram mitigados e
- Divulgar os métodos e técnicas propostos, bem como a *RBTool*, através da escrita de artigos científicos.

A principal **justificativa** para o projeto é o desenvolvimento de uma versão completa da ferramenta *RBTool*, possibilitando sua execução em diferentes ambientes (*desktop* e *web*), visando fornecer maior qualidade e produtividade na atividade de teste de software, mais precisamente na atividade de teste baseado em riscos, através da automação de tarefas repetitivas, diminuindo a possibilidade de inconsistência e erros nos resultados, favorecendo uma execução sistemática e contínua.

A proposta da *RBTool* é bastante pertinente pelo fato de ainda existirem lacunas no uso de ferramentas para automação de casos de teste baseado em riscos. É importante salientar que, na literatura, não foram

encontradas referências sobre ferramentas de apoio à abordagem RBT. Apenas Jørgensen (2005) desenvolveu um protótipo contendo poucas funcionalidades, não contemplando as atividades fundamentais da abordagem tais como: a identificação e análise de fatores de riscos associados aos requisitos do software e o planejamento e projeto de teste baseado em riscos. Além disso, o protótipo desenvolvido foi descontinuado e visava apenas fornecer apoio a uma abordagem RBT específica, proposta pelo próprio autor.

### 3. Metodologia de Execução

A gestão do projeto está baseada nas práticas do Guia PMBOK e do método ágil de gerenciamento de projeto SCRUM. Para a realização do projeto, as atividades a serem executadas foram divididas em seis fases, descrita a seguir:

**Fase 1 - preparação do projeto:** definição da equipe do projeto selecionando bolsistas a partir de experiências e habilidades. Alinhamento da equipe às expectativas do projeto apresentando cronograma e responsabilidades;

**Fase 2 - estudo sobre o tema:** revisão bibliográfica sobre a área de testes baseados em riscos, abordagens e ferramentas existentes. Realização de estudo comparativo de ferramentas de testes baseado em riscos disponíveis e pesquisa dos conceitos fundamentais de gerenciamento de riscos e teste de software. Além de estudo das tecnologias e ambiente de desenvolvimento;

**Fase 3 - levantamento e revisão dos requisitos e modelagem:** revisão dos requisitos existentes na primeira versão da *RBTTTool* visando realizar algumas melhorias. Levantamento de novos requisitos para modelagem e implementação da versão web da *RBTTTool*;

**Fase 4 - construção da ferramenta:** implementação da ferramenta de acordo com os requisitos levantados e revisados;

**Fase 5 - avaliação da ferramenta:** avaliação da ferramenta através de estudos experimentais e prova de conceito. Criação de manuais de utilização da ferramenta;

**Fase 6 - divulgação dos resultados:** compreende a escrita e apresentação de artigos em congressos científicos.

### 4. Resultados Obtidos

Nesta seção, são apresentados, de forma sucinta, os produtos e resultados relevantes obtidos no ciclo atual de desenvolvimento do projeto.

A primeira contribuição do projeto refere-se ao **produto de software gerado**, a ferramenta *RBTTTool*, na versão *web*, de código aberto, disponível para o mercado com dois grandes módulos: o módulo de apoio à gerência de riscos para o teste de software, que compreende as atividades de identificação, análise e controle dos fatores de riscos associados aos requisitos do software e (ii) o módulo de apoio ao gerenciamento do teste de software baseado em riscos que compreende as atividades de planejamento, projeto, execução e avaliação dos testes. As telas das principais funcionalidades são apresentadas no Apêndice.

Conforme apresentado nos objetivos específicos, novos **métodos e algoritmos** também foram estudados e desenvolvidos com o propósito de aprimorar a abordagem RBT e a ferramenta *RBTTTool*. A primeira contribuição corresponde a definição de indicadores para avaliação de riscos em projetos de software [Menezes Jr., 2012] e a segunda contribuição corresponde a proposta de técnica para identificação de riscos específica para avaliação de requisitos em projetos de software [Silva, 2012]. Ambos os trabalhos foram desenvolvidos durante o mestrado em Ciência da Computação, na Pós-graduação do Centro de Informática da UFPE (CIn/UFPE).

A primeira versão da *RBTTTool* foi proposta por Oliveira (2008), sendo melhorada posteriormente por Souza (2008), Menezes Jr, (2009) e Silva (2009), resultando nas seguintes **publicações de artigos**: Menezes Jr. et al. (2009), Souza et al. (2009), Menezes Jr. et al. (2010) e Souza et al. (2010).

A versão *web* da *RBTTTool* está em fase de avaliação e os métodos e algoritmos propostos neste ciclo resultaram nas seguintes **publicações de artigos**: Menezes Jr. et al. (2011) e Menezes Jr. et al. (2012).

Com relação aos **recursos humanos capacitados**, neste ciclo, a realização deste projeto possibilitou a formação de dois mestres na pós-graduação em Ciência da Computação do CIn/UFPE, além de um trabalho de mestrado em fase de conclusão e dois trabalhos de doutorado em andamento, também no CIn/UFPE, resultando em duas **dissertações de mestrado** defendidas nos anos de 2011 e 2012.

Além disso, o projeto contou com uma **parceria** com programa de pós-graduação em Engenharia da

Computação da Universidade de Pernambuco, na disciplina de Aplicações em Engenharia de Software, onde um grupo de alunos da disciplina desenvolveu um módulo da ferramenta *RBTTTool*, responsável pela leitura automática dos requisitos a partir de um documento de requisitos que segue um formato pré-estabelecido.

## 5. Aplicabilidade dos Resultados

A *RBTTTool web* tem aplicação imediata, podendo ser utilizada em projetos de desenvolvimento de software acadêmicos e/ou comerciais, de pequeno e médio porte que adotem a abordagem RBT para a realização de teste de software. Estudos [Chen,

2002; Amland, 1999; Souza et al., 2009; Souza et al., 2010] tem comprovado que tanto a abordagem, quanto a ferramenta permitem priorizar os recursos disponíveis para as funcionalidades que necessitam ser melhor testadas, ou seja para as funcionalidades que possuem maior exposição aos riscos.

## 6. Características Inovadoras

A *RBTTTool* dá suporte a uma abordagem de teste iterativa, onde em cada iteração são definidos os requisitos que serão testados. Os requisitos são selecionados com base no valor da exposição ao risco, calculado durante a análise de riscos, e os casos de teste são gerados com o objetivo de mitigar os riscos identificados em cada requisito. A ferramenta destaca a importância da atividade de gerenciamento de riscos aplicada ao processo de teste de software, favorecendo a priorização de esforços e alocação de recursos para os componentes de software que necessitam ser testados, mais cuidadosamente a partir da identificação, análise e controle dos riscos técnicos associados ao requisito do software.

## 7. Conclusão e Perspectivas Futuras

A *RBTTTool* é uma ferramenta inovadora, com aplicabilidade prática em projetos de desenvolvimento de software acadêmicos e/ou comerciais de apoio a abordagem RBT. A versão *web*, proposta neste ciclo, permite maior facilidade de uso e instalação da ferramenta, possibilitando uma maior produtividade e qualidade nos resultados. Os estudos de casos realizados e trabalhos publicados confirmam a eficiência no uso ferramenta em conjunto com a abordagem para o teste de software. Novas técnicas estudadas estão sendo continuamente integradas à ferramenta. Como trabalhos futuros, além da realização de estudos de casos, pretende-se utilizar técnicas de aprendizagem de máquina com o propósito de automatizar ou semi-automatizar tarefas da abordagem que demandam muito esforço e/ou são repetitivas.

## Referências

- Amland, S. (1999), Risk Based Testing and Metrics: Risk analysis fundamentals and metrics for software testing including a financial application case study. In: 5<sup>th</sup> International Conference EuroSTAR'99, Espanha.
- Chen, Y. (2002), Specification-based Regression Testing Measurement with Risk Analysis. Dissertação de Mestrado, University of Ottawa, Canada.
- Delamaro, M., Maldonado, J., Jino M. (2007), Introdução ao Teste de Software, Elsevier, 1<sup>a</sup> edição.
- Goldsmith, R. (2006), Early and Effective: The Perks of Risk-based Testing. Software Test & Performance Magazine, Volume 3, p.24-30.
- Jørgensen L. K. U. (2005), A software tool for risk-based testing. Dissertação de mestrado, Norwegian University of Science and Technology, Noruega.
- Kaner, C., Falk, J., Nguyen, H. Q. (1999), Testing computer software, Wiley, 2<sup>a</sup> edição. Oliveira, K. (2008), *RBTTTool: Uma Ferramenta para Identificação de Riscos no Teste de Software*. Trabalho de Graduação em Engenharia da Computação, Universidade de Pernambuco, Brasil.
- Menezes Jr, J. V. (2009), *RBTTTool: Proposta de Melhorias e Evolução na Automação das Atividades de Teste baseado em Riscos*. Trabalho de Graduação em Engenharia da Computação, Universidade de Pernambuco, Brasil.
- Menezes Jr, J. V., Gusmão, C. M. G., Mendes, E., Souza, E. P. R. (2009) *RBTTTool - Uma Ferramenta de Apoio à Abordagem de Teste de Software baseado em Riscos*. In: Simpósio Brasileiro de Engenharia de Software (SBES), Sessão de Ferramentas, p. 7-12, Fortaleza, Brasil.
- Menezes Jr, J. V., Moura, H. P., Gusmão, C. M. G. (2010), Definição de Indicadores para Avaliação de Riscos em Projetos de Software. In: 15<sup>o</sup> Workshop de Teses e Dissertações em Engenharia de Software. Congresso Brasileiro de Software: Teoria e Prática (CBSOFT), Salvador, Brasil.
- Menezes Jr, J. V., Gusmão, C. M. G., Moura, H. P. (2011), Towards Indicators for Risk Assessment in Software

- Projects. In: 23<sup>rd</sup> International Conference on Software & Systems Engineering and their Applications (ICSSEA), Paris, França.
- Menezes Jr, J. V., Gusmão, C. M. G., Moura, H. P. (2012), Indicators and Metrics for Risk Assessment in Software Projects: A Mapping Study. IX Experimental Software Engineering Latin American Workshop (ESELAW), Buenos Aires, Argentina.
- Menezes Jr., J. V. (2012), Indicators for Risk Assessment in Software Development Projects. Dissertação de Mestrado em Ciência da Computação. Universidade Federal de Pernambuco, Brasil.
- SEI - Software Engineering Institute (2006), Risk Management, <http://www.sei.cmu.edu/risk/>, Janeiro.
- Silva, L. (2009), Processo de Avaliação de Ferramenta de Apoio à Abordagem de Testes Baseados em Riscos do Ponto de Vista de Especialista em Gerência de Riscos. Trabalho de Graduação em Engenharia da Computação, Universidade de Pernambuco, Brasil.
- Silva, L. S. (2012), Risk Identification Technique for Requirements Assessment. Dissertação de Mestrado em Ciência da Computação. Universidade Federal de Pernambuco, Brasil.
- Souza, E. (2008), *RBTPProcess*: Modelo de Processo de Teste de Software baseado em Riscos, Dissertação de Mestrado em Engenharia da Computação, Universidade de Pernambuco, Brasil.
- Souza, E., Gusmão, C., Alves, K., Venâncio, J., Melo, R. (2009), Measurement and Control for Risk-based Test Cases and Activities. In: 10<sup>th</sup> IEEE Latin American Test Workshop (LATW), Búzios, Rio de Janeiro, Brasil.
- Souza, E. P. R., Gusmão, C. M. G., VENÂNCIO, J. (2010), Risk-Based Testing: A Case Study. In: 7<sup>th</sup> International Conference on Information Technology: New Generations (ITNG), p. 1032-1037, Las Vegas, USA.

## Apêndice

Nesta seção, são apresentadas as principais telas da ferramenta *RBTTool*, versão *web*. Após efetuar o *login*, o usuário deve selecionar ou cadastrar um projeto de teste de software baseado em riscos, como mostrado na Figura 2. Em seguida, o usuário cadastra os requisitos do software a ser testado ou seleciona a função de importação dos requisitos para a ferramenta a partir de um documento de requisitos com formato específico, conforme mostrado na Figura 3.

The screenshot shows the 'RBT Tool' web interface. The user is logged in as 'Usuário de Teste'. The main menu includes 'Home', 'Admin', 'Manage risk', 'Manage test', 'Logout', and 'Project: Projeto 1'. The left sidebar contains navigation options: 'Manage project', 'Manage requirement', 'Manage metric', 'Manage question', 'Manage questionnaire', 'Manage risks', and 'Manage answer'. The main content area is titled 'Insert project' and contains the following form fields: 'Name' (with a text input containing 'Projeto 1'), 'Start date' (with a date picker), 'Due date' (with a date picker), 'Main description' (with a large text area), and 'Requirement baseline' (with a text input). Below the form is a 'Manage resources' dialog box showing 'No results found' and 'Insert'/'Cancel' buttons.

Figura 2. Tela de cadastro de projeto

The screenshot shows the 'RBT Tool' web interface. The user is logged in as 'Usuário de Teste'. The main menu includes 'Home', 'Admin', 'Manage risk', 'Manage test', 'Logout', and 'Project: Project test'. The left sidebar contains navigation options: 'Manage project', 'Manage requirement', 'Manage metric', 'Manage question', 'Manage questionnaire', 'Manage risks', and 'Manage answer'. The main content area is titled 'Manage requirement' and displays a table of requirements.

Name	Version	Risk Exposure	Probability	Impact	Options
Gerenciar Projeto de Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Requisito para Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Métricas para Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Equações de Exposição ao Risco para Teste	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Equações de Impacto para Teste	1.0 ( 10/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Equações de Probabilidade para Teste	1.0 ( 10/00/2008 )	0.0	0.0	0.0	[Icons]
Analisar Riscos para Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Gerenciar Riscos para Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]
Identificar Riscos	1.0 ( 13/04/2009 )	0.0	0.0	0.0	[Icons]
Gerenciar Listas de Riscos para Teste baseado em Riscos	1.0 ( 00/00/2008 )	0.0	0.0	0.0	[Icons]

Figura 3. Tela com a lista de requisitos

Depois de cadastrados os requisitos, o usuário inclui as questões que farão parte do questionário baseado em taxonomia de riscos, utilizado na identificação dos riscos, conforme mostra a Figura 4 e a Figura 5.

Em seguida, para cada requisito, são identificados (Figura 6) e analisados riscos (Figura 7) utilizados para a identificação das funcionalidades que possuem maior probabilidade de apresentar falhas e, conseqüentemente, necessitam ser mais bem testadas.

The screenshot shows the 'RBT Tool' web interface. The user is logged in as 'Usuário de Teste'. The main menu includes 'Home', 'Admin', 'Manage risk', 'Manage test', 'Logout', and 'Project: Project test'. The left sidebar contains navigation options: 'Manage project', 'Manage requirement', 'Manage metric', 'Manage question', 'Manage questionnaire', 'Manage risks', and 'Manage answer'. The main content area is titled 'Insert question' and contains the following form fields: 'description\*' (with a text input containing 'Nesta funcionalidade'), 'Risk occurs when the answer is:.' (with a dropdown menu set to 'Yes'), and 'Insert'/'Cancel' buttons.

Figura 4. Tela de cadastro de questões

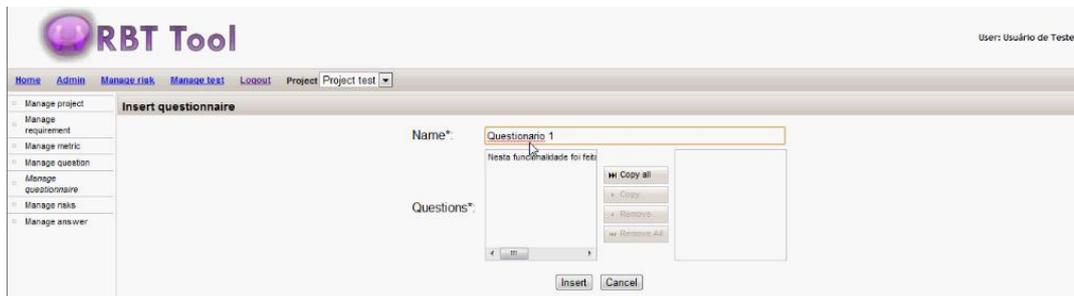


Figura 5. Tela de cadastro de questionário



Figura 6. Tela de identificação de riscos

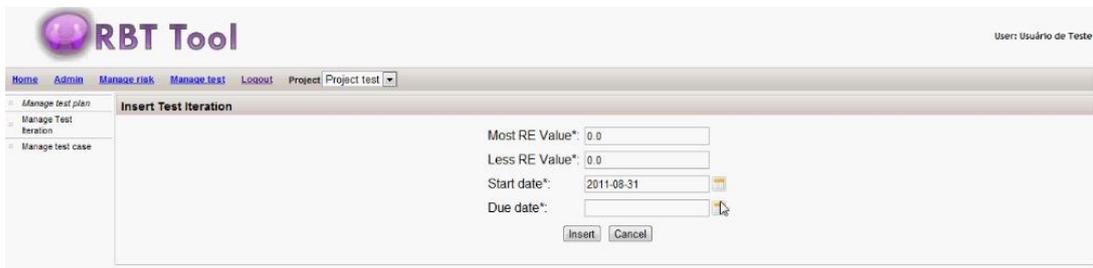


Figura 7. Tela de análise de riscos

# WDP-RT-Q: Uma Técnica Automatizada de Perguntas e Respostas para Inspeção de Aplicações Web

Priscila Fernandes, Tayana Conte

USES - Grupo de Usabilidade e Engenharia de Software  
Instituto de Computação - IComp  
Universidade Federal do Amazonas (UFAM) – Manaus, AM - Brasil  
{priscila.fernandes, tayana}@icomp.ufam.edu.br

**Abstract.** *Given the growth in the usage of Web Applications, the usability of Web applications has become a key success factor. Several technologies have been developed to evaluate and improve this quality factor. However, the usability inspections results still depend on the inspector experience. This research aims to develop a new Web usability inspection approach, called WE-QT (Web Evaluation – Question Technique), a question based technique that aims to reduce the difficulties of inspectors with little knowledge of usability.*

**Resumo.** *Com o crescimento do uso das aplicações Web, a usabilidade dessas aplicações têm se tornado um fator chave de sucesso. Diversas tecnologias têm sido elaboradas para avaliar e aprimorar esse fator de qualidade. No entanto, o resultado de inspeções de usabilidade ainda é diretamente influenciado pela experiência do inspetor. Essa pesquisa objetiva desenvolver uma nova abordagem de inspeção de usabilidade de aplicações Web, a WE-QT (Web Evaluation Question Technique), técnica baseada em perguntas para minimizar as dificuldades de inspetores com pouco conhecimento em usabilidade.*

## Introdução

O aumento dos recursos providos pela *Web* vem permitindo o uso cada vez mais intenso dessas aplicações na sociedade [Kappel *et al.* 2006]. Este crescimento se deve a duas características: a rápida evolução da *Web* e a usabilidade das aplicações [Luna *et al.* 2010]. A usabilidade é um dos fatores de qualidade mais importantes para aplicações *Web* [Offut 2002], sendo que a aceitabilidade das aplicações depende diretamente de sua usabilidade [Matera *et al.* 2006].

Aplicações *Web* com baixa usabilidade serão rapidamente substituídas por outras mais usáveis, assim que o público alvo souber de sua existência [Mendes *et al.* 2006]. Porém, ao utilizar estas aplicações, muitas vezes os usuários se deparam com erros ao executar tarefas, causados pela interface não intuitiva da aplicação [Gomes *et al.* 2009]. Por isso, melhorar a usabilidade de aplicações *Web* pode minimizar substancialmente a dificuldade de interação dos usuários e melhorar a qualidade dessas aplicações [Bonifácio *et al.* 2010].

O desafio de desenvolver aplicações *Web* com interface mais simples e fácil de usar tem tornado a avaliação de usabilidade um importante campo de pesquisa. Em vista desse cenário, uma variedade de métodos, técnicas e ferramentas para tratar de questões de usabilidade *Web* têm sido propostas [Insfran e Fernandez 2008]. No entanto, dentro do processo de desenvolvimento destas aplicações, avaliações de usabilidade são normalmente evitadas pelos desenvolvedores ou empresas, devido à sua falta de conhecimento nas tecnologias de avaliação.

O foco da presente pesquisa consiste na elaboração de uma tecnologia para avaliação de usabilidade específica para aplicações *Web*, que possa ser utilizada por profissionais com pouco conhecimento em usabilidade. O objetivo é auxiliar o processo de desenvolvimento de aplicações neste paradigma, possibilitando identificar e corrigir um maior número de defeitos de usabilidade; assim como incentivar a utilização de técnicas de avaliação de usabilidade por desenvolvedores e gerentes de projetos envolvidos no desenvolvimento de aplicações *Web*. Espera-se com isso contribuir para aumentar a qualidade das aplicações *Web*, e assim prover maior facilidade de interação aos usuários ao utilizarem tais aplicações. A técnica proposta, inicialmente chamada de “WDP-RT-Q: Uma Técnica Automatizada de Perguntas e Respostas para Inspeção de Aplicações Web”, teve o nome reformulado por motivo de simplificação, sendo renomeada de “WE-QT (Web Evaluation Question Technique)”.

O restante deste artigo está estruturado da seguinte forma. A seção 2 expõe os objetivos e a justificativa desta pesquisa. A seção 3 apresenta a metodologia utilizada neste trabalho. A seção 4 apresenta os resultados

obtidos. A seção 5 discorre sobre a aplicabilidade dos resultados. A seção 6 exhibe as características inovadoras desta pesquisa. Por fim, a seção 7 conclui o artigo e apresenta os trabalhos futuros.

### Objetivos e Justificativa

Embora muitos desenvolvedores se preocupem com a usabilidade das aplicações durante o processo de desenvolvimento, muitas aplicações ainda não atendem a expectativa de seus usuários [Offutt 2002]. Por isso, investimentos em usabilidade podem garantir maior qualidade das aplicações e menor custo de desenvolvimento, através da adoção de avaliação de usabilidade durante o processo de desenvolvimento. No cenário de desenvolvimento de aplicações de software convencionais, é possível constatar aumentos substanciais de qualidade e redução de defeitos devido à adoção de inspeções e outras técnicas de revisão [Travassos *et al.* 1999; Leite *et al.* 2005].

Por esse motivo, várias técnicas específicas para aplicações *Web* têm sido propostas, mas estas não estão sendo aplicadas pelas empresas de desenvolvimento de software *Web*. Isto se deve, possivelmente, ao desconhecimento das novas abordagens, limitação no orçamento, assim como a dificuldade dos desenvolvedores em utilizar as abordagens propostas.

Deste modo, a questão de pesquisa que norteia este trabalho consiste na possibilidade de definição de uma técnica para inspeção específica para aplicações *Web*, que possa ser facilmente utilizada por inspetores com pouco conhecimento em usabilidade e inspeção, apoiando a obtenção de maior qualidade nas aplicações *Web*.

O objetivo principal deste artigo consiste em apresentar os resultados dos estudos experimentais para a avaliação técnica WE-QT, utilizando uma abordagem metodológica baseada em experimentação. O propósito final é que a técnica possa ser utilizada, com esforço reduzido, por todos os envolvidos nos projetos de desenvolvimento de aplicações *Web*. E com isso, difundir o conhecimento sobre estudos experimentais para apoiar a transferência segura da nova tecnologia de software para indústria.

### Metodologia de execução

A metodologia utilizada nesta pesquisa (Figura 1) é uma adaptação da abordagem proposta por Shull *et al.* (2001). Esta metodologia utiliza estudos experimentais para a introdução de tecnologias de software na indústria, como forma de determinar o que funciona ou não durante a avaliação da tecnologia proposta desde sua definição até sua transferência para indústria.

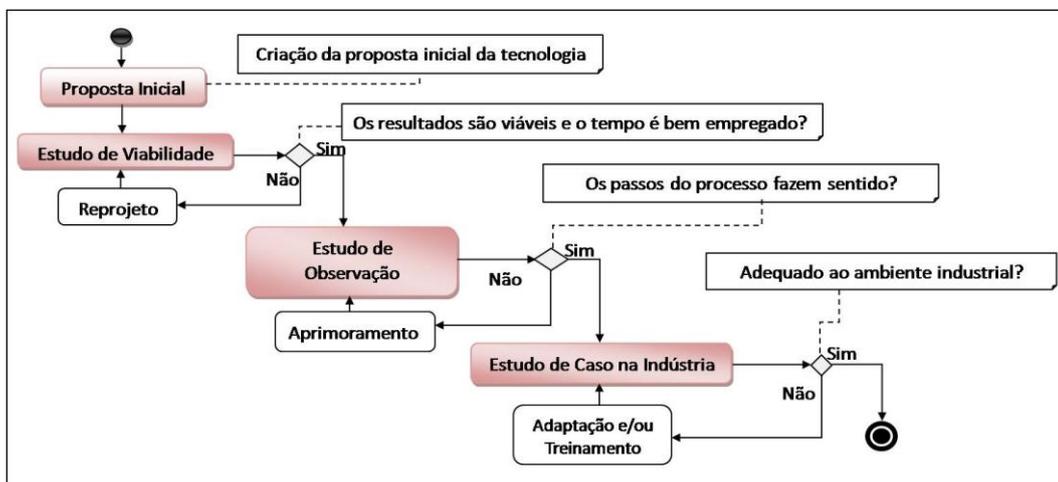


Figura 1: Visão geral da metodologia utilizada na pesquisa, adaptada de Shull *et al.* (2001).

A abordagem desta pesquisa possui três etapas: (1) Execução de estudo de viabilidade: para caracterizar a técnica proposta e verificar sua viabilidade de uso; (2) Execução de estudo de observação: para aprofundar a compreensão de como a técnica é aplicada, possibilitando seu aprimoramento; (3) Estudo de caso em um ciclo de vida de desenvolvimento real: com o propósito de caracterizar a adequação da técnica aos ciclos de vida de desenvolvimento *Web*. Foram realizados até o momento dois estudos experimentais. O primeiro estudo foi feito para avaliar a viabilidade da WE-QT e o segundo para entender como os inspetores aplicam a WE-QT.

### 4. Resultados Obtidos

#### 4.1. Proposta Inicial da WE-QT (Web Evaluation - Question Technique)

Com o objetivo de tornar mais fácil a aplicação por inspetores novatos, sem perda de eficácia na detecção de defeitos, a WDP-RT foi evoluída para outro tipo de técnica de inspeção: a técnica baseada em perguntas WE-QT [Fernandes *et al.* 2011]. A WE-QT consiste em uma série de perguntas que guiam o inspetor a encontrar os problemas de usabilidade. A abordagem mais simples da WE-QT visa reduzir as dificuldades de inspeção dos inspetores com pouco conhecimento sobre avaliação de usabilidade.

A WE-QT consiste em uma série de perguntas que guiam o inspetor a encontrar os problemas de usabilidade. A abordagem mais simples da WE-QT visa reduzir as dificuldades de inspeção dos inspetores com pouco conhecimento sobre avaliação de usabilidade. Não é necessário treinamento na técnica WE-QT antes de utilizá-la.

A WE-QT oculta os conceitos das perspectivas *Web*, usados na WDP-RT, além de qualquer informação que não seja necessária em um determinado momento da inspeção. O objetivo é não confundir o inspetor com informações irrelevantes para alcançar determinado objetivo.

As perguntas da WE-QT foram elaboradas a partir das instruções da WDP-RT extraindo-se o objetivo principal de cada instrução e os transformando em perguntas. As perguntas são divididas em Perguntas de Decisão (PD) e Sub-Perguntas (SP). As PDs são responsáveis por fazer a verificação da existência de elementos a serem avaliados; enquanto as SPs são encarregadas da avaliação desses elementos. O item (a) da Figura 2 ilustra o mapeamento de uma instrução da WDP-RT em uma PD da WE-QT.

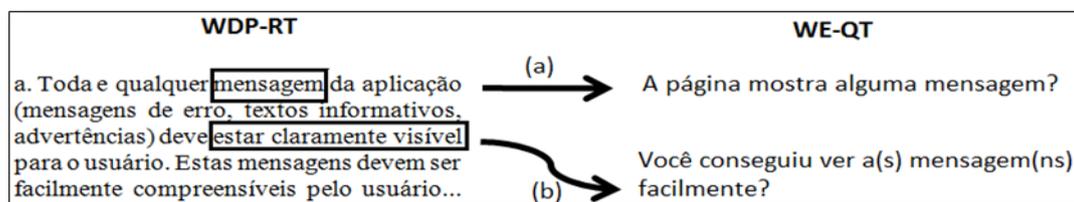


Figura 2: Formação de perguntas: (a) Pergunta de Decisão; (b) Sub-Pergunta

Uma vez formada a PD de uma instrução, extraiu-se os objetivos principais que deverão ser verificados referentes ao objeto principal da PD, formando-se uma SP; como mostra o item (b) da Figura 2.

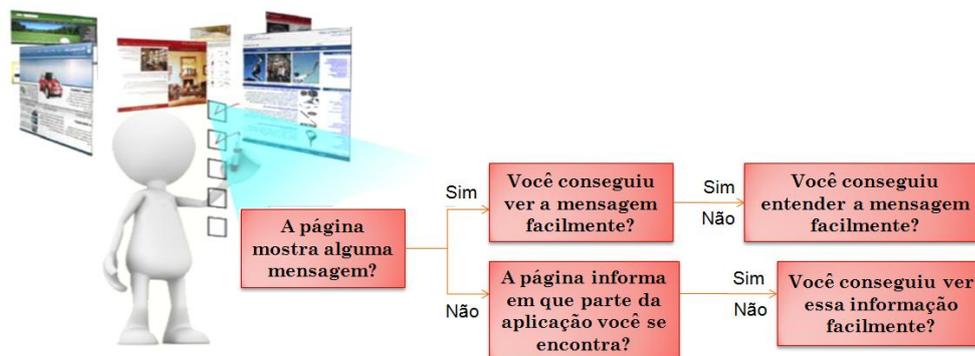


Figura 3: Ilustração do mapeamento das perguntas da WE-QT.

As PDs definem, dependendo da resposta do usuário, se as SPs serão ou não apresentadas ao inspetor, como ilustra a Figura 3. Esse processo é automatizado pela ferramenta de apoio Interactive WE, que apresenta as perguntas da WE-QT ao inspetor na ordem correta, além de apoiar o registro dos problemas de usabilidade detectados [Rivero 2010].

#### 4.2. Estudo de Viabilidade

O estudo de viabilidade, primeira etapa da metodologia, foi realizado em Setembro de 2010 com 12 alunos de graduação do curso de Ciência da Computação da UFAM. Para este estudo foram considerados dois indicadores quantitativos: eficácia (razão entre o número de defeitos detectados e o total de defeitos existentes) e eficiência (razão entre o número de defeitos detectados e o tempo gasto na inspeção).

A ideia preliminar foi comparar a técnica WE-QT com a WDP-RT em termos de eficácia e eficiência, mas foi detectado um problema que poderia ameaçar a validade deste estudo ao tentar comparar os dois tratamentos diretamente: a WDP-RT exige treinamento, enquanto a WE-QT precisava ser aplicada sem treinamento para

possibilitar a avaliação de sua viabilidade. Como fornecer treinamento para um único grupo causaria um desbalanceamento entre os participantes, optou-se por usar a WE-QT como único tratamento. Por isso, este estudo de viabilidade não teve como objetivo principal realizar uma comparação, mas avaliar se a WE-QT apresenta um resultado viável em termos de possibilitar inspetores com pouco conhecimento em usabilidade a encontrar defeitos na execução de sua primeira avaliação de usabilidade.

Os resultados quantitativos obtidos, descritos detalhadamente em Fernandes *et al.* (2011), foram os seguintes: a WE-QT possui uma eficiência de 13,09 defeitos por hora, e eficácia individual média de 29,37%. Usamos os resultados deste mesmo indicador em outros estudos como base para possibilitar avaliar se o resultado alcançado por este indicador está adequado. O percentual de eficácia neste estudo é equivalente ao percentual apresentado pelos inspetores que usaram a técnica WDP-RT (29%) e acima dos resultados apresentados pelos inspetores que usaram a técnica WDP (13%) em estudo apresentado em [Gomes *et al.* 2009]. Como a aplicação *Web* utilizada neste estudo não é a mesma utilizada em [Gomes *et al.* 2009], não foi feita uma comparação da eficiência. De acordo com esses resultados, é possível ter indícios de que o uso da WE-QT como técnica de inspeção para inspetores com pouco conhecimento em usabilidade é viável para a detecção de defeitos. A análise qualitativa também possibilitou a melhoria da técnica.

### 4.3. Evoluindo a WE-QT

Os resultados obtidos no estudo de viabilidade nos permitiram revisar e evoluir a técnica proposta para a sua segunda versão. Após análise detalhada da técnica, verificou-se que: (1) as perguntas relacionadas ao mesmo objeto de avaliação podiam ser agrupadas, ilustrado na Figura 4; (2) as perguntas que avaliam as mesmas características dos objetos de avaliação (perguntas redundantes) poderiam ser descartadas; (3) determinadas perguntas não precisavam ser respondidas por página; e (4) as perguntas careciam de mais detalhamento de certos termos.

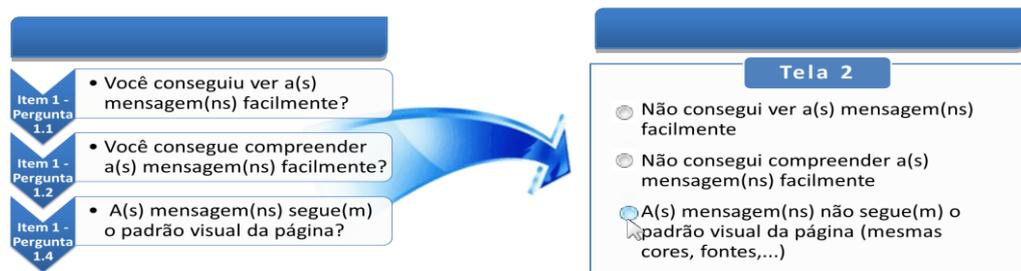


Figura 4: Ilustração de melhorias feitas durante a evolução da técnica WE-QT para a versão 2.

#### 4.3.1. Estudo de Observação

Uma vez verificada a viabilidade da WE-QT, foi elaborado um estudo de observação para verificar de que maneira a técnica é aplicada e responder a pergunta “Os passos do processo fazem sentido?”, seguindo a metodologia adotada nesta pesquisa. Dessa forma, foi realizado um estudo experimental com o propósito de avaliar a nova versão da WE-QT. Os resultados foram importantes para verificar se a técnica auxilia a detecção de defeitos por inspetores novatos de maneira eficaz e eficiente. O estudo foi realizado em outubro de 2011 com 7 alunos de pós-graduação da UFAM. A inspeção foi realizada individualmente com cada participante, de forma a obter a maior quantidade de informações possível.

Neste estudo os resultados quantitativos foram: 34,29% de eficácia e 31,91 de eficiência, indicando uma melhoria na técnica em relação a primeira versão. O objetivo principal deste estudo de observação foi compreender como os inspetores aplicam a técnica WE-QT, entretanto a análise dos dados quantitativos constitui uma oportunidade para avaliar o desempenho da técnica.

Através deste estudo foi possível obter um conjunto de informações detalhadas em relação à aplicação da técnica pelos inspetores. Isso permitiu não só identificar mais detalhadamente pontos fracos na técnica; mas a maneira que ela é utilizada para realizar a detecção dos defeitos de usabilidade e comportamentos do usuário ao utilizá-la.

Em relação as melhorias a serem feitas na WE-QT, foi possível observar pontos como dificuldade de entender certas perguntas e afirmações; dificuldade em realizar a avaliação em relação às afirmações; e dificuldade de compreender e realizar o processo de inspeção. Apesar dos resultados quantitativos terem sido positivos, os dados observacionais mostram que a técnica não é aplicada da maneira correta, sendo necessário aprimoramento da WE-QT e realização de um novo estudo de observação.

### 4.4. Trabalhos publicados

- Artigo “Avaliando uma nova Abordagem para Inspeção de Usabilidade através de Análise Quantitativa e Qualitativa” apresentado no VIII Workshop Latino Americano de Engenharia de Software Experimental (ESELAW 2011), Rio de Janeiro – RJ.
- Pôster “WE-QT Tool: Uma Ferramenta de Apoio a Inspeção de Usabilidade de Aplicações *Web*” apresentado no X Simpósio Brasileiro de Fatores Humanos em Sistemas Computacionais (IHC 2011), Porto de Galinhas – PE.
- Artigo “Improving a Web Usability Inspection Technique through an Observational Study” submetido para o 24th International Conference on Software Engineering and Knowledge Engineering (SEKE 2012), São Francisco, USA.

#### **4.5. Outros resultados: Recurso Humano Capacitado**

Projeto de Iniciação Científica de Lennon Correa Chaves: Bacharel em Engenharia da Computação. Universidade Federal do Amazonas – UFAM.

Resultados deste projeto fazem parte da pesquisa de projeto final e da dissertação de Priscila Silva Fernandes: Mestranda em Informática. Universidade Federal do Amazonas – UFAM

#### **5. Aplicabilidade dos resultados: Benefícios para a Indústria e Academia**

O procedimento comum em estudos de viabilidade é comparar duas tecnologias. A idéia preliminar foi comparar a técnica WE-QT com suas técnicas base. No entanto, foi detectado um problema que poderia ameaçar a validade deste estudo ao se tentar comparar os dois tratamentos diretamente: as técnicas base exigem treinamento, enquanto a WE-QT precisava ser aplicada sem treinamento para possibilitar a avaliação de sua viabilidade. Como fornecer treinamento para um único grupo causaria um desbalanceamento entre os participantes, optou-se por usar a WE-QT como único tratamento [Fernandes *et al.* 2011]. Apesar de não ter sido realizada uma comparação da WE-QT com as técnicas base, os resultados mostram o indicativo de viabilidade da técnica proposta.

Além disso, outro benefício com esta pesquisa consiste em incentivar uso de avaliações de usabilidade por profissionais da indústria, dentro do processo de desenvolvimento, e com isso aumentar a qualidade de aplicações *Web*.

Além dos benefícios para a indústria, os benefícios acadêmicos deste estudo foram contribuiu para a geração de conhecimento sobre avaliação de usabilidade em aplicações *Web*. Além disso, outra contribuição está relacionada a estudos experimentais com o propósito de disseminar o conhecimento sobre sua utilização para avaliar e evoluir uma nova tecnologia nas áreas de Engenharia de Software e Interação Humano-Computador.

#### **6. Características inovadoras**

Como característica inovadora deste projeto temos a elaboração de uma técnica voltada especificamente para inspetores novatos, que possuem pouco ou nenhum conhecimento em usabilidade e inspeção. Além disso, a ideia é que ao utilizar a WE-QT, os inspetores possam adquirir conhecimentos sobre usabilidade, utilizando uma abordagem adaptativa que facilita o processo de detecção de defeitos, através do fluxo de inspeção da WE-QT adaptado a partir das respostas providas pelos inspetores.

Também se pode considerar a geração de indícios de que a abordagem da técnica pode aumentar a identificação de problemas de usabilidade por inspetores novatos. E com isso incentivar o uso de técnicas de avaliação de usabilidade por todos os envolvidos nos projetos de desenvolvimento de aplicações *Web*.

#### **7. Conclusões e trabalhos futuros**

Neste artigo, a técnica WE-QT (*Web Evaluation – Question Technique*) foi apresentada. Esta técnica é específica para inspeção de usabilidade de aplicações *Web* por inspetores com pouco ou nenhum conhecimento em usabilidade e inspeção. Esta técnica está sendo avaliada através de uma metodologia que utiliza estudos experimentais para a evolução da tecnologia proposta e realizar sua transferência segura para indústria. Foram apresentados também os estudos experimentais realizados para avaliação da técnica.

Como trabalho futuro pretende-se, ainda: (1) melhorar a técnica através de uma análise detalhada de cada pergunta/afirmação na identificação da lista final de defeitos detectados; (2) realizar um estudo comparativo entre

WE-QT e uma abordagem de inspeção de usabilidade existente, juntamente com análise estatística e (3) a replicação da experiência na indústria.

Ao apresentar os estudos experimentais, realizados para avaliação da WE-QT, almeja-se contribuir com o compartilhamento do conhecimento científico. Espera-se com isso, incentivar a indústria de software *Web* a realizar avaliações de usabilidade com maior frequência durante o desenvolvimento.

### Referências Bibliográficas

- Bonifácio B., Viana D, Vieira S., Araújo C., Conte T., 2010, "Aplicando Técnicas de Inspeção de Usabilidade para Avaliar Aplicações em Dispositivos Móveis", Simpósio de Fatores Humanos em Sistemas Computacionais, v. 1, pp. 189-192.
- Conte, T., Massolar, J., Mendes, E., *et al.*, 2009a, "Web Usability Inspection Technique Based on Design Perspectives", IET Software Journal, v. 3, n. 2, pp. 106-123.
- Conte, T., Vaz, V., Massolar, J., *et al.*, 2009b, "Improving a Web Usability Inspection Technique using Qualitative and Quantitative Data from an Observational Study". XXIII Simpósio Brasileiro de Engenharia de Software, pp. 227 - 235.
- Fernandes, P., Rivero, L., Bonifácio, B., Santos, D., Conte, T., 2011, "Avaliando uma nova Abordagem para Inspeção de Usabilidade através de Análise Quantitativa e Qualitativa". VIII Workshop Latino Americano de Engenharia de Software Experimental, v. 1, p. 67-76.
- Gomes, M., Santos, D. V., Chaves, L., Castro, A., Vaz, V. T., Soares, A., Travassos, G. H., Conte, T., 2009. "WDP-RT: Uma técnica de leitura para inspeção de usabilidade de aplicações Web". VI Experimental Software Engineering Latin American Workshop (ESELAW 2009), v. 1, pp. 124-133.
- Gomes, M., Santos, F., Santos, D. V., Travassos, G. H., Conte, T., 2010, "Evoluindo uma Técnica de Avaliação de Usabilidade através de Estudos In Vitro e In Vivo". IX Simpósio Brasileiro de Qualidade de Software. pp. 229-244.
- Infran, E., Fernandez, A., 2008, "A Systematic Review of Usability Evaluation in Web Development ". II International Workshop on Web Usability and Accessibility (IWWUA 2008), v. LNCS 5176 - WISE 2008 Workshops, pp. 81-91.
- Kappel, G., Pröll, B., Reich, S., Retschitzegger, W., 2006. "An Introduction to Web Engineering". Kappel *et al.* (eds), Web Engineering: The Discipline of Systematic Development of Web Applications, John Wiley \ & Sons.
- Luna, E.R., Panach, J.I., Grigera, J.N., Rossi, G., Pastor, O., 2010, "Incorporating usability requirements in a test/model-driven web engineering approach", Journal of Web Engineering, v. 9, n. 2, pp. 132-156.
- Matera, M., Rizzo, F., Carughi, G., Mendes, E., Mosley, N., 2006, "Web Usability: Principles and Evaluation Methods Web Engineering", Springer Berlin Heidelberg.
- Mendes, E., Mosley, N. and Counsell, S., 2006, "The need for Web Engineer: an introduction", Web Engineering, Springer, pp.1-27.
- Nielsen, J., 1994, "Heuristic evaluation", In Nielsen, J., and Mack, R.L. (Eds.), "Usability Inspection Methods" (John Wiley & Sons, 1994).
- Offutt, J., 2002, "Quality Attributes of Web Software Applications", IEEE Software Journal, v. 19, n. 2, pp. 25-32.
- Rivero, L., 2010, "Interactive WE: Um assistente de apoio ao processo de inspeção de usabilidade", Monografia, disponível em: [www.dcc.ufam.edu.br/uses/](http://www.dcc.ufam.edu.br/uses/).
- Shull, F., Carver, J., Travassos, G. H., 2001. "An empirical methodology for introducing software processes". ACM SIGSOFT Software Engineering Notes, v. 26, n. 5, pp. 288-296.

# **Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software**

## **Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software**

### **Vencedores Anuais**

Encerradas as etapas de cada Ciclo Anual de Projetos do PBQP Software, com a efetivação do processo de avaliação dos projetos concluídos durante o período vigente, os resultados são divulgados amplamente apontando os indicados para premiação.

A seguir, a relação completa dos projetos já laureados, com seus objetivos e dados básicos sobre a coordenação, entidades e responsáveis.

### **Vencedores do Ciclo 2011**

#### **1º Lugar:**

#### **BRISA QPM – Qualimobile Process Management**

**Kátia Argolo de Castro, André Luís Gobbi Sanches, Johnny Souza, Thamer de Jesus Hatem - BRISA/DF**

E-mail:katia.castro@brisa.org.br, andre.sanches@brisa.org.br, henrique.conti@brisa.org.br, Thamer.deJesusHatem@brisa.org.br, johnny.souza@brisa.org.br

Objetivos : Implementar uma ferramenta de controle de qualidade voltada para o desenvolvimento de software para dispositivos móveis, especialmente na camada do sistema operacional, aproveitando a larga experiência da Brisa na área, adquirida após anos de atuação em desenvolvimento e controle de qualidade.

#### **2º Lugar:**

#### **Projeto Spider - Software Process Improvement: DEvelopment and Research**

**Sandro Ronaldo Bezerra Oliveira, Ewelton Yoshio Chiba Yoshidome, Fabrício Medeiros Alho, Kleverton e Julio Cezar Costa Furtado**

**Universidade Federal do Pará - UFPA/PA**

E-mail:ewelton.yoshio@gmail.com, fabricioalho@unama.br, julio.furtado@gmail.com, [srbo@ufpa.br](mailto:srbo@ufpa.br)

Objetivos: Por meio de um mapeamento entre o modelo MR-MPS, a norma ISO/IEC12207 e a norma IEEE1517, foi elaborada uma abordagem que auxilia as empresas interessadas na sistematização do processo de gerência de reutilização, contribuindo na aplicação da maturidade para este processo, disponível em [www.spider.ufpa.br/projetos/spider\\_reuse/mapeamento\\_reuso.zip](http://www.spider.ufpa.br/projetos/spider_reuse/mapeamento_reuso.zip). O mapeamento realizado permitiu a identificação e o tratamento de gaps existentes entre o modelo e as normas estudadas. A ferramenta Spider-Reuse tem o propósito de apoiar a utilização da referida abordagem, a partir de um fluxo de atividades que contempla as boas práticas resultantes do mapeamento realizado. O fluxo encontra-se, de maneira macro, apresentado na Figura 1, descrito usando as notações do BPMN – Business Process Modeling Notation. Um melhor entendimento do fluxo pode ser obtido em [www.spider.ufpa.br/projetos/spider\\_reuse/framework\\_reuso.zip](http://www.spider.ufpa.br/projetos/spider_reuse/framework_reuso.zip).

#### **3º Lugar:**

#### **O Processo de Inovação Centrada no Ser Humano: Boas Práticas de Desenvolvimento de Software Utilizadas no Projeto Portal do Aluno**

**Gustavo Andrade Maestri, Gabriel C. Cardoso e Ivan Linhares Martins**

**Centro de Convergência Digital – Fundação Centros de Referência em Tecnologias Inovadoras (CERTI) – Florianópolis, SC Universidade Federal de Santa Catarina (UFSC)**

E-mail: gabrielc.cardoso@gmail.com, gaa@certi.org.br

Objetivos : A proposta do projeto é a criação de um ambiente online de aprendizagem colaborativa, sendo destinado principalmente a alunos e professores do ensino médio e fundamental. Dentre seus objetivos estão:

- \* Criação de um ambiente online em formato de rede social, para formação de comunidades e compartilhamento de informação e recursos educacionais;
- \* Criação de um biblioteca digital que agregue tanto bases educacionais já existentes, quanto conteúdos produzidos pela comunidade;

\* Desenvolvimento de competências em infraestrutura computacional de larga escala (suporte a centenas de milhares de usuários) e métodos de engenharia de software para simulação destes cenários;

\* Desenvolvimento de métodos de pesquisa para permitir o Design Centrado na experiência do Usuário (DCU);

No Ciclo 2011 do PBQP Software além da premiação de projetos, o Comitê de Avaliadores resolveu reconhecer o mérito diferenciado do projeto Consolidação do Programa MPS.BR - Melhoria de Processo do Software Brasileiro (2008-2011), cujos resultados, impactos, porte, relevância e abrangência indicaram a necessidade de conferir uma Premiação Especial.

## Premiação Especial

### Consolidação do Programa MPS.BR - Melhoria de Processo do Software Brasileiro (2008-2011)

Kival Weber, José Antonio Antonioni, Nelson Franco, Elidiane Barroso, Cleide Silva, André Sotovia

SOFTEX - Associação para Promoção da Excelência do Software Brasileiro

E-mail: [jaa@nac.softex.br](mailto:jaa@nac.softex.br), [cleide@nac.softex.br](mailto:cleide@nac.softex.br), [elidiane@nac.softex.br](mailto:elidiane@nac.softex.br), [nelson@nac.softex.br](mailto:nelson@nac.softex.br),  
[kival.weber@nac.softex.br](mailto:kival.weber@nac.softex.br)

Objetivos: Consolidar um método para customização de modelos de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software.

01 - Analisar o processo de desenvolvimento de modelos customizados de capacidade/maturidade de processos de software.

02 - Identificar as técnicas e ferramentas apropriadas para o desenvolvimento de modelos customizados de capacidade/maturidade de processos de software.

03 - Consolidar o método para customização de modelos de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software por meio da implementação das melhorias previstas na sua avaliação anterior.

04 - Validar o método desenvolvido por meio de um estudo empírico no desenvolvimento de um modelo de capacidade/maturidade de processos de software para o domínio de convergência digital.

Comissão de Avaliação do Grupo de Projetos do PBQP Software Ciclo 2010

<ul style="list-style-type: none"><li>• Edmeia Leonor Pereira de Andrade - EMBRAPA</li><li>• Teresinha Falabella - Eletronorte</li><li>• Edna Pereira Fernandes – Receita Federal do Brasil</li><li>• Carlos Lombardi - Consultor</li><li>• Salvador Alves de Melo Junior - Consultor</li><li>• Cristiano Lehrer - Consultor</li><li>• Francisco Vasconcellos - Consultor</li><li>• Carlos Nassur - Câmara dos Deputados</li><li>• Luiz Carlos de A. Oliveira - CELEPAR</li><li>•</li></ul>	<ul style="list-style-type: none"><li>• Alexandre Vasconcellos - UFPE</li><li>• Cristina Angela Filipak Machado - CELEPAR</li><li>• Sarah Kohan - Fundação Vanzolin</li><li>• André Villas-Boas - CPqD</li><li>• Jucele França Vasconcellos - Consultor</li><li>• Ana Liddy Cenni de Castro Magalhães -</li><li>• Rafael Prikladnicki -PUCRS</li><li>• Maria de Fátima Mattiello Francisco – INPE</li></ul>
---	---

## Vencedores do Ciclo 2010

### 1º Lugar:

**Curso de Residência em Arquitetura de Software: Uma Parceria UFPA, UNAMA e CESUPA com AmazonCorp, CTIC-UFPA e PRODEPA**

**Entidades Coordenadoras: 1Laboratório de Engenharia de Software - Universidade Federal do Pará (UFPA)2, Instituto Federal do Pará (IFPA)3**

**Responsáveis: Rodrigo Quites Reis1, Carla Alessandra Lima Reis1, Claudio Roberto de Lima Martins2, Marcos Venícios Araújo3**

E-mail: quites@ufpa.br , clima@ufpa.br, cmartins@ifpa.edu.br, maraujo@cesupa.br

Objetivo: Formar profissionais/especialistas com forte embasamento teórico e experiência prática como residentes na área de Arquitetura de Software.

### 2º Lugar:

**Uma Investigação sobre os Fatores Críticos de Sucesso em Iniciativas de Melhoria de Processos de Software**

**Entidade Coordenadora: Universidade Federal do Rio de Janeiro/ Coordenação de Programa de Pós-Graduação em Engenharia - UFRJ/COPPE**

**Responsáveis: Mariano Angel Montoni e Ana Regina Rocha**

E-mail: {mmontoni,darocha}@cos.ufrj.br

Objetivo: O objetivo deste projeto é desenvolver uma abordagem para condução de iniciativas de melhoria de processos de software com o propósito de permitir que organizações de consultoria em melhoria de processos de software estabeleçam estratégias adequadas para conduzir iniciativas de melhoria, bem como permitir a execução e acompanhamento dessas estratégias.

### 3º Lugar:

**Agile Brazil 2010 - Conferência Brasileira sobre Métodos Ágeis de Desenvolvimento de Software**

**Entidade Coordenadora: Pontifícia Universidade Católica do Rio Grande do Sul - PUCRS**

**Responsável: Rafael Prikladnicki**

E-mail: rafaelp@pucrs.br

Objetivo: Realização da Conferência Brasileira sobre Métodos Ágeis de Desenvolvimento de Software (Agile Brazil 2010). A Agile Brazil é uma conferência nacional sem fins lucrativos organizada por representantes das principais comunidades ágeis brasileiras. O evento tem como objetivo promover a comunicação e a colaboração entre seus integrantes visando à disseminação coordenada da cultura Ágil por todo o país.

## Comissão de Avaliação do Grupo de Projetos do PBQP Software Ciclo 2010

<ul style="list-style-type: none"><li>• Ana Cervigni Guerra - CTI</li><li>• Ana Liddy Cenni de Castro Magalhães - QualityFocus</li><li>• André Henrique de Siqueira - BCB</li><li>• André Villas-Boas - CPqD</li><li>• Carlos Lombardi - Consultor</li><li>• Clênio Figueiredo Salviano - CTI</li><li>• Cristiano Lehrer - MS</li><li>• Cristina Ângela Filipak Machado - CELEPAR</li><li>• Edmeia Leonor Pereira de Andrade - EMBRAPA</li><li>• Edna Pereira Fernandes - Ministério da Fazenda/SRF</li><li>•</li></ul>	<ul style="list-style-type: none"><li>• Francisco José Silveira de Vasconcellos - Consultor</li><li>• Jucele França Vasconcellos - Consultora</li><li>• Luiz Carlos de Almeida Oliveira - CELEPAR</li><li>• Maria de Fátima Mattiello Francisco - INPE</li><li>• Regina Maria Thienne Colombo - CTI</li><li>• Sarah Kohan - FCAV</li><li>• Salvador Alves de Melo Junior - Consultor</li><li>• Sonia Thereza Maintinguer - CPqD</li><li>• Teresa Maria de Medeiros Maciel - UFRPE</li><li>• Teresinha Morais Falabella de Castro – Eletronorte</li></ul>
---	--

## Vencedores do Ciclo 2009

### 1º Lugar:

#### Processo de Contratação de Software e Serviços Correlatos para Entes Governamentais

**Entidades Coordenadoras:** Tribunal de Contas da União (TCU); Empresa Brasileira de Pesquisa Agropecuária (Embrapa); Universidade de Brasília (UnB). - Faculdade FGA/Campus Gama

**Responsáveis:** Cláudio Silva da Cruz, Edméia Leonor Pereira de Andrade, Rejane Maria da Costa Figueiredo

E-mail: cscruz@cscruz.org, edmeia.andrade@embrapa.br, rejanecosta@unb.br

Objetivo: Propor uma versão do MPS.BR-Guia de Aquisição para entes da Administração Pública Federal, compatibilizando o processo de aquisição de software e serviços correlatos com a legislação pública brasileira para contratações de serviços de tecnologia da informação (TI) e a respectiva jurisprudência.

### 2º Lugar:

#### FireScrum: Ferramenta Open-Source

**Entidades Coordenadoras:** Universidade Federal Rural de Pernambuco - UFRPE; Universidade Federal de Pernambuco - UFPE; Centro de Estudos e Sistemas Avançados do Recife – CESAR

**Responsáveis:** Eric Cavalcanti, Teresa M. de Medeiros Maciel e Jones Albuquerque

E-mail: ecavalcanti@gmail.com; tmmaciel@gmail e jas@gmail.com

Objetivo: Definição, especificação e desenvolvimento de uma ferramenta, FireScrum, open source proposta para servir de suporte ao Scrum em equipes remotas. O objetivo principal da ferramenta é proporcionar o uso do Scrum por equipes remotas evitando a perda de histórico, propiciando o levantamento de métricas sem demandar esforço excessivo do time e, principalmente, facilitando sua adoção sem burocratizar o processo.

### 3º Lugar:

#### WDP-RT: Uma Técnica de Leitura para Inspeção de Usabilidade de Aplicações Web

**Entidade Coordenadora:** Universidade Federal do Amazonas - UFAM

**Responsáveis:** Tayana Conte e Marcos Antonio da Silva Gomes

E-mail: tayanaconte@gmail.com e marcos.sgomes@gmail.com

Objetivo: Desenvolver uma técnica de leitura para inspeção de software Web, através da extensão da técnica WDP (Web Design Perspectives-based Inspection Technique) (CONTE et al, 2007). Propor melhorias ao método, através de estudos experimentais. Referências: Conte T., Massolar, J., Mendes, E., Travassos, G. (2007) Web Usability Inspection Technique Based on Design Perspectives. SBES 2007, João Pessoa, PB, Brasil.

#### Comissão de Avaliação do Grupo de Projetos do PBQP Software Ciclo 2009

<ul style="list-style-type: none"><li>Ana Cervigni Guerra - CTI</li><li>Ana Liddy Cenni de Castro Magalhães - QualityFocus</li><li>Alexandre Marcos Lins de Vasconcelos - UFPE</li><li>André Villas-Boas - CPqD</li><li>Carlos Lombardi - Consultor</li><li>Cristiano Lehrer - ICODES</li><li>Cristina Ângela Filipak Machado - CELEPAR</li><li>Edna Pereira Fernandes - Ministério da Fazenda/SRF</li></ul>	<ul style="list-style-type: none"><li>Francisco José Silveira de Vasconcellos - SAE/PR</li><li>Jucele França de A. Vasconcellos - UFRJ/COPPE</li><li>Maria de Fátima Mattiello Francisco - INPE</li><li>Regina Maria Thienne Colombo - CTI</li><li>Samy Lima Assi - FabriQ</li><li>Sarah Kohan - FCAV</li><li>Sonia Thereza Maintinguer - CPqD</li></ul>
--	--

## Vencedores do Ciclo 2008

### 1º Lugar:

**“FAPS - Ferramenta de Suporte à Avaliação de Processos de Software Alinhada com os Modelos CMMI e MPS.BR”**

**Entidades Coordenadoras: Universidade do Vale do Itajaí (UNIVALI) e Incremental Tecnologia em Informática**

**Responsáveis: Marcelo Thiry, C. G. Gresse von Wangenheim, Alessandra Zoucas, Leonardo Reis Tristão, José Luiz da Silveira**

E-mail: {marcello.thiry, gresse, alessandra.zoucas, leutristao, joseluizsilveira}@gmail.com

Objetivo: Este projeto tem por objetivo desenvolver uma ferramenta de software baseada nos métodos de avaliação SCAMPI e MA-MPS, que auxilie a equipe avaliadora durante o processo de avaliação (Gerenciamento, Contextualização e Execução), suportando as atividades da avaliação. O processo de avaliação é intelectualmente complexo e requer alta experiência dos avaliadores especialmente no mapeamento dos processos e níveis de capacidade, na identificação dos riscos e sugestões de melhoria. Assim, o objetivo deste trabalho não é automatizar completamente o processo de avaliação, mas prover um suporte para a gerência dos documentos relacionados e semi-automatizar procedimentos quando possível.

### 2º Lugar:

**“Um Framework para Avaliação da Implantação do MR MPS em Grupos Cooperados de Empresas”**

**Entidades Coordenadoras: Faculdade de Informática da Pontifícia Universidade Católica do Rio Grande do Sul - PUCRS**

**Responsável: Rafael Prikladnicki**

E-mail: rafael@inf.pucrs.br

Objetivo: O foco deste projeto é criação de um framework de avaliação da implantação do Modelo de Negócio Cooperado (MNC-MPS.BR) em grupos de empresas, de forma a melhorar a execução de projetos cooperados em um contexto nacional, permitindo avaliações sistematizadas e comparativas, algo que não tem sido feito atualmente.

### 3º Lugar:

**“FROISPI - Framework Return on Investment of Software Process Improvement”**

**Entidades Coordenadoras: Centro de Informática da UFPE e Fundação Centro de Análise, Pesquisa e Inovação Tecnológica - Instituto de Ensino Superior (FUCAPI-CESF)**

**Responsáveis: Paulino W. P. Viana, Alexandre M. L. de Vasconcelos**

E-mail: wagner\_palheta@fucapi.br

Objetivo: Propor um Framework constituído por fases, baseado nos conceitos de Business Process Management, Metodologia ROI e DMAIC, baseando-se nos indicadores utilizados por David Rico em ROI of SPI e uma seleção de medições utilizadas para Melhorias de Processo de Software

## Comissão de Avaliação do Grupo de Projetos do PBQP Software Ciclo 2008

<ul style="list-style-type: none"><li>• André Villas-Boas - CPqD</li><li>• Carlos Lombardi - Consultor</li><li>• Cristiano Lehrer - ICODES</li><li>• Cristina Ângela Filipak Machado - CELEPAR</li><li>• Edna Pereira Fernandes - Ministério da Fazenda/SRF</li><li>• Gisele Rodrigues de Mesquita Ferreira - Banco Central</li><li>• José Eduardo Albuquerque Medeiros - MCT/SEPIN</li><li>• Jucele França de A. Vasconcellos - UFRJ/COPPE</li></ul>	<ul style="list-style-type: none"><li>• Luiz Carlos de Almeida Oliveira – CELEPAR</li><li>• Maria de Fátima Mattiello Francisco - INPE</li><li>• Salvador Alves de Melo Junior - Consultor</li><li>• Samy Lima Assi - FabriQ</li><li>• Sonia Thereza Maintinguer - CPqD</li><li>• Teresa Maria de Medeiros Maciel - C.E.S.A.R</li><li>• Teresinha Morais Falabella de Castro – Eletronorte</li></ul>
---	--

## Vencedores do Ciclo 2007

Dos 60 projetos integrantes desse Ciclo foram relatados 54% dos projetos, sendo que 22 foram concluídos (37%) e encaminhados para avaliação final com seus respectivos artigos.

Este processo de avaliação foi finalizado em Brasília/DF, nos dias 14 e 15 de maio de 2008, durante o Workshop da Qualidade e Produtividade em Software - WQPS.

A seguir, o resultado da classificação final:

### 1º Lugar:

**“Plataforma de Tecnologia da Informação e Comunicação de Santa Catarina - PLATIC”**

**Entidades Coordenadoras: Instituto Euvaldo Lodi de Santa Catarina - IEL/SC e pela Universidade Regional de Blumenau - FURB**

**Responsáveis: Carlos Eduardo Negrão Bizzotto, Eliza Coral e Valéria Arriero Pereira**

E-mail: carlos\_bizzotto@uol.com.br, elizaiel@yahoo.com.br, varriero@gmail.com

Objetivo: Desenvolver e disponibilizar um conjunto de ferramentas que melhorem a qualidade e a produtividade, ampliando a competitividade das empresas. Essa competitividade será obtida por meio da melhoria do processo de desenvolvimento, dos produtos e serviços, da gestão do negócio, além da capacitação de pessoas. O projeto integra a academia, as empresas e o governo, garantindo que os conhecimentos gerados sejam aplicados nas empresas, contribuindo para que elas possam competir no mercado mundial.

### 2º Lugar:

**“IACS - Identificação Automática de Componentes de Software”**

**Entidades Coordenadoras: Laboratório de Inovação– DigitalAssets / Ci&T e Universidade Estadual de Campinas - UNICAMP.**

**Responsáveis: Marcílio Oliveira, José Vahl, Kleber Bacili.**

E-mail: marcilio.oliveira@digitalassets.com.br; jose.vahl@digitalassets.com.br; kleber.bacili@digitalassets.com.br.

Objetivo: 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. Está sendo implementada uma ferramenta que utiliza os mecanismos que identifiquem automaticamente o agrupamento de artefatos relacionados sugerindo ativos de software (componentes, serviços etc.) candidatos à reutilização.

### 3º Lugar:

**“Avaliação de Riscos Aplicada à Qualidade em Desenvolvimento de Software”**

**Entidades Coordenadoras: Módulo Security e pela Prime Up Soluções em TI**

**Responsáveis: Alberto Bastos, Gustavo Carvalho, Leandro Daflon, Rafael Espinha e João Condack.**

E-mail: abastos@modulo.com.br, gustavo.carvalho@primeup.com.br, daflon@primeup.com.br, rafael.espinha@primeup.com.br

Objetivo: Este projeto gera uma solução para auxiliar empresas contratantes e desenvolvedoras de software, a avaliar, de modo automático e padronizado, pessoas, fornecedores e processos com base nas melhores práticas de engenharia de software (CMMI e MPS.Br). O software contribuirá com a difusão e gestão de conceitos de Qualidade tendo como base Regulamentações e Normas técnicas, otimizando investimentos e reduzindo custos na realização de avaliação de projetos de desenvolvimento de software.

## Comissão de Avaliação do Grupo de Projetos do PBQP-Software Ciclo 2007

<ul style="list-style-type: none"><li>Alexandre Marcos Lins de Vasconcelos - UFPE</li><li>Ana Cervigni Guerra - CenPRA</li><li>Ana Liddy Cenni de Castro Magalhães - Consultor</li><li>André Henrique de Siqueira - CEF</li><li>André Villas-Boas - CPqD</li><li>Carlos Henrique R. Cardoso - INATEL</li><li>Carlos Lombardi - Consultor</li><li>Carlos Mathias Mota Vargas - Consultor</li><li>Clênio Figueiredo Salviano - CenPRA</li><li>Cristiano Lehrer - ICODES</li><li>Cristina Ângela Filipak Machado - CELEPAR</li><li>Denise Maria Rodrigues Carneiro - UNIFOR</li><li>Edmeia Leonor Pereira de Andrade - Ministério da Agricultura</li></ul>	<ul style="list-style-type: none"><li>Edna Pereira Fernandes - Ministério da Fazenda/SRF</li><li>José Eduardo Albuquerque Medeiros - MCT/SEPIN</li><li>Jucele França de A. Vasconcellos - UFRJ/COPPE</li><li>Karina Rodrigues Pereira - C.E.S.A.R</li><li>Luiz Carlos de Almeida Oliveira – CELEPAR</li><li>Maria de Fátima Mattiello Francisco - INPE</li><li>Regina Thienne Colombo - CenPRA</li><li>Renata Endriss - SOFTEX/Recife</li><li>Salvador Alves de Melo Junior - Consultor</li><li>Samy Lima Assi - FabriQ</li><li>Sonia Thereza Maintinguer - CPqD</li><li>Teresa Maria de Medeiros Maciel - C.E.S.A.R</li><li>Teresinha Morais Falabella de Castro – Eletronorte</li></ul>
---	---

## Vencedores do Ciclo 2006

Dos 99 projetos integrantes desse Ciclo foram relatados 95% dos projetos, sendo que 47 foram concluídos (48%) e encaminhados para avaliação final.

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 14 e 15 de maio de 2007, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

### 1º Lugar:

**“MPS.BR - Melhoria de Processo do Software Brasileiro”**

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

**Responsáveis: Kival Chaves Weber e Eratóstenes Araújo**

E-mail: kival.weber}@nac.softex.br, era@nac.softex.br

Objetivo: Criação de um novo modelo para melhoria de processo de software - composto de um Modelo de Referência (MRMPS), 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).

### 2º Lugar:

**“Laboratório de Engenharia de Software e Inteligência Artificial: Construção do ambiente WebAPSEE”**

**Entidade Coordenadora: Laboratório de Engenharia de Software da Universidade Federal do Pará - UFPA**

**Responsáveis: Carla Alessandra Lima Reis e Rodrigo Quites Reis**

E-mail: clima@ufpa.br, quites@computer.org

Objetivo: Consolidar pesquisa em Tecnologia do Processo de Software, em especial a experiência na construção do ambiente WebAPSEE, para gerência de processos de software, procurando avançar no estado da arte dessa tecnologia através da integração com técnicas de inteligência artificial.

### 3º Lugar: (empate)

**“RCCS - Rede de Compartilhamento de Componentes de Software”**

**Entidades Coordenadoras: Laboratório de Inovação em Software - Ci&T Labs / UNICAMP e Instituto de Computação - Universidade Estadual de Campinas**

**Responsáveis: Kleber Bacili, Ricardo Anido, Marcílio Oliveira**

E-mail: krbacili@cit.com.br, ranido@ic.unicamp.br, marcilio@cit.com.br

Objetivo: O principal objetivo deste projeto é a criação de uma rede pública de distribuição e compartilhamento de componentes de software através de uma arquitetura baseada em serviços e da utilização dos padrões Web Services.

**“Qualidade do Software Embarcado em Aplicações Espaciais”**

**Entidades Coordenadoras: Instituto Nacional de Pesquisas Espaciais - INPE e DBA Engenharia de Sistemas Ltda**

**Responsáveis: Fátima Mattiello Francisco e Ricardo Costa**

E-mail: fatima@dss.inpe.br, rcosta@dba.com.br

Objetivo: Transferir para a indústria brasileira do setor de software os conhecimentos adquiridos no INPE no desenvolvimento de software crítico para a área espacial, capacitando a indústria no uso de ambientes e técnicas de validação adotadas na integração e testes dos software embarcados em cargas úteis de missões de satélites científicos.

## Comissão de Avaliação do Grupo de Projetos do PBQP-Software Ciclo 2006

<ul style="list-style-type: none"><li>Alexandre Marcos Lins de Vasconcelos - UFPE</li><li>Ana Cervigni Guerra - CenPRA</li><li>Ana Liddy Cenni de Castro Magalhães - Consultor</li><li>André Henrique de Siqueira - CEF</li><li>André Villas-Boas - CPqD</li><li>Carlos Henrique R. Cardoso - INATEL</li><li>Carlos Lombardi - Consultor</li><li>Carlos Mathias Mota Vargas - Consultor</li><li>Clênio Figueiredo Salviano - CenPRA</li><li>Cristiano Lehrer - ICODS</li><li>Cristina Ângela Filipak Machado - CELEPAR</li><li>Denise Maria Rodrigues Carneiro - UNIFOR</li><li>Edmeia Leonor Pereira de Andrade - Ministério da Agricultura</li><li>Edna Pereira Fernandes - Ministério da Fazenda/SRF</li></ul>	<ul style="list-style-type: none"><li>Francisco José Silveira de Vasconcelos - Marinha do Brasil</li><li>José Salustiano Fagundes - HIRIX</li><li>Jucele França de A. Vasconcelos - UFRJ/COPPE</li><li>Karina Rodrigues Pereira - C.E.S.A.R</li><li>Luiz Carlos de Almeida Oliveira - CELEPAR</li><li>Marcelo Alves de Barros - UFPB</li><li>Regina Thienne Colombo - CenPRA</li><li>Renata Endriss - SOFTEX/Recife</li><li>Salvador Alves de Melo Junior - Consultor</li><li>Samy Lima Assi - FabriQ</li><li>Sonia Thereza Maintinguer - CPqD</li><li>Teresa Maria de Medeiros Maciel - C.E.S.A.R</li><li>Teresinha Morais Falabella de Castro - Eletronorte</li></ul>
---	---

## Vencedores do Ciclo 2005

Dos 142 projetos integrantes desse Ciclo foram relatados 92% dos projetos, sendo que 50 foram concluídos (35%) e encaminhados para avaliação final 49 projetos cujos artigos foram apresentados.

Este processo de avaliação foi finalizado em Brasília/DF, nos dias 27 e 28 de abril de 2006, durante o Workshop da Qualidade e Produtividade em Software - WQPS.

A seguir, o resultado da classificação final:

### 1º Lugar:

#### “Residência em Software”

**Entidades Coordenadoras: Quali Software Processes, Universidade Federal de Pernambuco - UFPE e Motorola Industrial Ltda**

**Responsáveis: Augusto Cezar Alves Sampaio e José Mário Lima**

E-mail: acas@cin.ufpe.br, augusto@qualiti.com, jose.lima@motorola.com

Objetivo: Implantar um modelo inovador de formação em Engenharia de Software seguindo a filosofia da Residência Médica. Na Residência em Software o papel dos hospitais é desempenhado pelas fábricas de software e uma instituição de ensino oferece a formação teórica.

O modelo proposto pode ser utilizado tanto como uma formação complementar a cursos de graduação em computação e áreas afins quanto para reciclar profissionais de mercado nas novas tecnologias, conceitos e paradigmas da engenharia de software.

### 2º Lugar:

#### “Revista ProQualiti: Qualidade na Produção de Software”

**Entidades Coordenadoras: Universidade Federal de Lavras - UFLA e SWQuality Consultoria e Sistemas Ltda**

**Responsáveis: Ana Cristina Rouiller e Geovane Nogueira Lima**

E-mail: acr@comp.ufla.br, geovane@swquality.com.br

Objetivo: Criar uma edição semestral da revista InfoComp especializada em Qualidade de Software. Esta revista, publicada em meio impresso pela Editora da UFLA, deve apresentar a priori dois tipos de trabalhos: artigos científicos e relatos de experiência. É possível a inclusão de artigos de cunho didático, tais como tutoriais e revisões bibliográficas.

### 3º Lugar:

#### “PRO2PI: Perfis de Capacidade de Processo para Melhoria de Processo”

**Entidades Coordenadoras: Centro de Pesquisas Renato Archer - CenPRA, Universidade Estadual de Campinas/Faculdade de Engenharia Elétrica e de Computação - Unicamp/FEE**

**Responsáveis: Clênio Figueiredo Salviano e Mario Jino**

E-mail: clenio.salviano@cenpra.gov.br, jino@dca.fee.unicamp.br

Objetivo: Consolidar uma metodologia para orientar a melhoria de processo de software, com referência a múltiplos modelos de capacidade de processo, baseada no conceito de perfis de capacidade de processo, que sejam, dinâmicos e específicos. Esta metodologia vem sendo desenvolvida e aplicada nos últimos seis anos e foi recentemente nomeada PRO2PI. Os modelos utilizados têm sido: ISO/IEC 15504, SW-CMM, CMMI-SE/SW, MR-MPS e outros.

## Comissão de Avaliação do Grupo de Projetos do PBQP-Software Ciclo 2005

<ul style="list-style-type: none"><li>Alexandre Marcos Lins de Vasconcelos - UFPE</li><li>Ana Cervigni Guerra - CenPRA</li><li>Ana Liddy Cenni de Castro Magalhães - Consultor</li><li>André Henrique de Siqueira - CEF</li><li>André Villas-Boas - CPqD</li><li>Carlos Henrique R. Cardoso - INATEL</li><li>Carlos Lombardi - Consultor</li><li>Carlos Mathias Mota Vargas - Consultor</li><li>Clênio Figueiredo Salviano - CenPRA</li><li>Cristiano Lehrer - ICODES</li><li>Cristina Ângela Filipak Machado - CELEPAR</li><li>Denise Maria Rodrigues Carneiro - UNIFOR</li><li>Edmeia Leonor Pereira de Andrade - Ministério da Agricultura</li></ul>	<ul style="list-style-type: none"><li>Edna Pereira Fernandes - Ministério da Fazenda/SRF</li><li>Francisco José Silveira de Vasconcelos - Marinha do Brasil</li><li>José Salustiano Fagundes - SUCESU/DF</li><li>Jucele França de A. Vasconcelos - UFRJ/COPPE</li><li>Karina Rodrigues Pereira - C.E.S.A.R</li><li>Luiz Carlos de Almeida Oliveira - CELEPAR</li><li>Paulo Sérgio Bomfim – MCT/SPOA/CGTI</li><li>Regina Thienne Colombo - CenPRA</li><li>Renata Endriss - SOFTEX/Recife</li><li>Salvador Alves de Melo Junior – MCT/SEPIN</li><li>Samy Lima Assi - FabriQ</li><li>Teresa Maria de Medeiros Maciel - C.E.S.A.R</li><li>Teresinha Morais Falabella de Castro – Eletronorte</li></ul>
---	--

## Vencedores do Ciclo 2004

Dos 75 projetos integrantes desse Ciclo foram relatados 100% dos projetos, sendo que 42 foram concluídos (56%) e encaminhados para avaliação final 41 projetos cujos artigos foram apresentados.

Este processo de avaliação foi finalizado em Brasília/DF, nos dias 28 e 29 de abril de 2005, durante o Workshop da Qualidade e Produtividade em Software - WQPS.

A seguir, o resultado da classificação final:

### 1º Lugar:

**“Documentação Essencial: Um Enfoque na Documentação Necessária para Manutenção de Software”**

**Entidade Coordenadora: Universidade Católica de Brasília - UCB**

**Responsáveis: Sérgio Cozzetti Bertoldi de Souza, Nicolas Anquetil e Káthia Marçal de Oliveira**

E-mail: sergio.cozzetti@triall.com.br, anquetil@ucb.br, kathia@ucb.br

Objetivo: Propor um método de desenvolvimento de software com documentação essencial, necessária para manutenção, e que não comprometa a entrega do software para o cliente.

### 2º Lugar:

**"Consolidação da Metodologia para Avaliação de Processos de Software de MPEs Baseada na Norma ISO/IEC 15504 (SPICE)"**

**Entidades Coordenadoras: Universidade do Vale do Itajaí - UNIVALI e Centro de Pesquisas Renato Archer - CenPRA**

**Responsáveis: Christiane Gresse von Wangenheim e Clenio F. Salviano**

E-mail: gresse@sj.univali.br, clenio.salviano@cenpra.gov.br

Objetivo: Consolidar uma metodologia de avaliação de processos de software com foco na ISO/IEC 15504 (SPICE) para o contexto de micro e pequenas empresas (MPEs) de software.

### 3º Lugar (empate):

**"Livro: Aquisição de Produtos e Serviços de Software"**

**Entidade Coordenadora: Centro de Pesquisas Renato Archer - CenPRA**

**Responsáveis: Ana Cervigni Guerra e Ângela Maria Alves**

E-mail: ana.guerra@cenpra.gov.br, angela.alves@cenpra.gov.br

Objetivo: Elaborar publicação para apresentação dos problemas clássicos e tendências para processos de aquisição de produtos e serviços de software. Apresenta também um conjunto de atividades fundamentais para as empresas que necessitam contratar, para os contratantes, clientes e compradores de produtos e serviços de software.

### 3º Lugar (empate):

**"Qualidade de Software na Região Norte"**

**Entidade Coordenadora: Centro de Incubação e Desenvolvimento Empresarial - CIDE**

**Responsável: Francisco José Silveira de Vasconcellos**

E-mail: vasconcellos@netmentor.info

Objetivo: Estimular a divulgação de indicadores setoriais da qualidade e produtividade no setor de software como instrumentos de conscientização e motivação.

De acordo com procedimentos previamente estabelecidos pelo Grupo de Projetos, no Ciclo 2004 a exemplo do Ciclo 2003, adicionalmente aos melhores projetos premiados foram reconhecidos os melhores projetos por Categoria, resultando no reconhecimento na categoria **Articulação Institucional** para:

**“O AmazonSoft como Núcleo SOFTEX do Amazonas”**

Coordenação: Centro de Análise, Pesquisa e Inovação Tecnológica – FUCAPI, Antonio Luiz da Silva Maués

**Comissão de Avaliação do Grupo de Projetos do PBQP-Software Ciclo 2004**

<ul style="list-style-type: none"><li>Alexandre Marcos Lins de Vasconcelos - Qualiiti</li><li>Ana Cervigni Guerra - CenPRA</li><li>André Henrique de Siqueira - CEF</li><li>André Villas-Boas - CPqD</li><li>Carlos Henrique R. Cardoso - INATEL</li><li>Carlos Lombardi - Consultor</li><li>Carlos Mathias Mota Vargas - CBM</li><li>Clênio Figueiredo Salviano - CenPRA</li><li>Cristiano Lehrer - IESGO</li><li>Cristina Ângela Filipak Machado - CELEPAR</li><li>Denise Maria Rodrigues Carneiro - INSOFT</li></ul>	<ul style="list-style-type: none"><li>Francisco José Silveira de Vasconcellos - Marinha do Brasil</li><li>Giovano Marcos Mazetto - Consultor</li><li>José Carlos De Luca - ASSESPRO</li><li>José Salustiano Fagundes - SUCESU/DF</li><li>Jucele França de A. Vasconcellos - FUCAPI</li><li>Káthia Marçal de Oliveira - UCB</li><li>Luiz Carlos de Almeida Oliveira - CELEPAR</li><li>Marcos Nobre - SOFTEX</li><li>Mauro de Mesquita Spinola - USP/POLI</li><li>Paulo Sérgio Bomfim - MCT/CGMI</li><li>Regina Thienne Colombo - CenPRA</li></ul>
---	--

<ul style="list-style-type: none"> <li>• Edmeia Leonor Pereira de Andrade - EMBRAPA</li> <li>• Edna Pereira Fernandes - Ministério da Fazenda/SRF</li> </ul>	<ul style="list-style-type: none"> <li>• Renata Endriss - C.E.S.A.R</li> <li>• Salvador Alves de Melo Junior - MCT/SEPIN</li> <li>• Samy Lima Assi - FabriQ</li> <li>• Teresa Maria de Medeiros Maciel - C.E.S.A.R</li> <li>• Teresinha Morais Falabella de Castro - Eletronorte</li> </ul>
--	---

### Vencedores do Ciclo 2003

Dos 82 projetos integrantes desse Ciclo foram relatados 99% dos projetos, sendo que 26 foram concluídos (32%).

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 13 e 14 de maio de 2004, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

#### 1º Lugar:

**“Adaptação do MEDE-PROS para o Processo de Pré-Qualificação do PNAFM”**

**Entidade Coordenadora: Centro de Pesquisas Renato Archer - CenPRA**

**Responsáveis: Ana Cervigni Guerra e Sônia Thereza Maintinguer**

E-mail: ana.guerra@cenpra.gov.br, soniam@cpqd.com.br

Objetivo: Elaborar uma proposta genérica de um método de avaliação especialista para produtos de software baseado na especificação de requisitos presente em editais e no MEDE-PROS. O programa PNAFM foi considerado como motivador da proposta.

#### 2º Lugar:

**“Curso de Especialização Lato-Sensu à Distância em Melhoria de Processo de Software”**

**Entidade Coordenadora: Universidade Federal de Lavras - UFLA**

**Responsável: Ana Cristina Rouiller**

E-mail: acr@comp.ufla.br

Objetivo: Aprofundar o conhecimento de graduados em Ciência da Computação (ou áreas afins) em Qualidade de Software, com especial enfoque na melhoria dos processos e produtos de software; formar recursos humanos de forma a atender às atuais exigências de qualificação do mercado de trabalho; disseminar os conceitos e aplicabilidade da garantia da qualidade de software para todo o país, estreitando o relacionamento entre indústria de software e centros de pesquisa e normatização; e auxiliar a indústria de software nacional, através de uma pesquisa de levantamento dos processos de qualidade da indústria nacional.

#### 3º Lugar:

**“CORDIS-FBC Integrado a um Sistema de Gerência do Conhecimento Cardioknowledge”**

**Entidades Coordenadoras: Coordenação de Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro - UFRJ/COPPE e Fundação Bahiana de Cardiologia - FBC**

**Responsável: Álvaro Rabelo Jr., Ana Regina Rocha e Karina Villela**

E-mail: arabelojr@terra.com.br, darocha@centroin.com.br, kvillela@uol.com.br

Objetivo: Construir um ambiente de desenvolvimento de software orientado a uma organização específica. O ambiente CORDIS-FBC está portanto orientado ao domínio de Cardiologia, é baseado nos processos de software especificamente definidos para a Fundação Bahiana de Cardiologia e utiliza o seu conhecimento organizacional em desenvolvimento de software, por incorporar princípios de gestão do conhecimento. CORDIS-FBC será integrado ao ambiente de gerência do conhecimento CardioKnowledge.

De acordo com procedimentos previamente estabelecidos pelo Grupo de Projetos, no Ciclo 2003, adicionalmente aos melhores projetos premiados foram reconhecidos os melhores projetos por Categoria, resultando no reconhecimento na categoria **Serviços Tecnológicos** para:

**“Elementos Fundamentais para a Melhoria da Qualidade de Software nas Organizações de TI”**

**Entidade Coordenadora: Centro de Pesquisas Renato Archer - CenPRA**

**Responsáveis: Ana Cervigni Guerra e Eduardo Paulo de Souza**

E-mail: ana.guerra@cenpra.gov.br, [eduardos@cpqd.com.br](mailto:eduardos@cpqd.com.br)

## **Vencedores do Ciclo 2002**

Dos 68 projetos integrantes desse Ciclo foram relatados 94% dos projetos, sendo que 36 foram concluídos (53%).

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 07 a 09 de maio de 2003, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

### **1º Lugar:**

**“Simpósio Brasileiro de Qualidade de Software”**

**Entidades Coordenadoras: Universidade Federal do Espírito Santo - UFES e Universidade Católica de Brasília - UCB**

**Responsáveis: Ricardo de Almeida Falbo, Káthia Marçal de Oliveira**

E-mail: falbo@inf.ufes.br, kathia@ucb.br

Objetivo: Realizar do 1º Simpósio Brasileiro de Qualidade de Software (SBQS). O SBQS é a consagração do Workshop de Qualidade de Software realizado nos últimos oito anos no Brasil associado ao Simpósio Brasileiro de Engenharia de Software.

### **2º Lugar:**

**“Workshop on Software Quality”**

**Entidades Coordenadoras: Coordenação de Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro - UFRJ/COPPE e Universidade Católica de Brasília - UCB**

**Responsáveis: Káthia Marçal de Oliveira, Ana Regina Rocha, Kival Chaves Weber**

E-mail: kathia@ucb.br, darocha@cos.ufrj.br, kival@polo.com.br

Objetivo: Realizar o Workshop de Qualidade de Software integrado ao International Conference on Software Engineering. O Workshop foi realizado em Orlando - USA no dia 25 de maio de 2002 e contou com a presença de pesquisadores de renome internacional na área de qualidade de software.

### **3º Lugar:**

**“SIMPROS 2002: IV Simpósio Internacional de Melhoria de Processo de Software”**

**Entidades Coordenadoras: Centro de Pesquisas Renato Archer - CenPRA, Faculdade SENAC de Ciências Exatas e Tecnologia e Centro de Estudos e Sistemas Avançados do Recife - C.E.S.A.R**

**Responsável: Clenio Figueiredo Salviano, Ozeas V. Santana Filho e Teresa Maciel**

E-mail: clenio.salviano@cenpra.gov.br, ozeas@sp.senac.br, teresa@cesar.org.br

Objetivo: Realizar o SIMPROS 2002 para disseminar as principais tendências para melhoria de processo de software no cenário nacional e internacional, dando continuidade aos cinco eventos já realizados com sucesso nos anos de 1995, 1996, 1998, 1999, 2000 e 2001. As três primeiras edições do evento tiveram o nome de Semana de Engenharia de Software e a partir da quarta edição foi adotado o nome SIMPROS. O SIMPROS 2002 foi organizado de forma conjunta pelo CenPRA, Faculdade SENAC de Ciências Exatas e Tecnologia e C.E.S.A.R.

### **Destaques "hors-concours" - Ciclo 2002:**

**“Qualidade e Produtividade no Setor de Software Brasileiro – 2001”**

Entidade Coordenadora: Ministério da Ciência e Tecnologia/Secretaria de Política de Informática - MCT/SEPIN

Responsáveis: Célia Joseli do Nascimento e Diva da Silva Marinho

**“Produtividade Sistêmica no Setor de Software Brasileiro – 2001”**

Entidades Coordenadoras: Ministério da Ciência e Tecnologia/Secretaria de Política de Informática - MCT/SEPIN e Instituto Brasileiro da Qualidade e Produtividade do Paraná - IBQP-PR

Responsáveis: Célia Joseli do Nascimento e Cristiane de Almeida

**“Normalização de Software no Brasil em 2002”**

Entidade Coordenadora: Associação Brasileira de Normas Técnicas / Subcomitê de Software - ABNT/CB-21

Responsável: Kival Chaves Weber

## **Vencedores do Ciclo 2001**

Dos 103 projetos integrantes desse Ciclo, foram relatados 72 projetos (70%), sendo que 28 foram concluídos (27%).

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 11 a 12 de março de 2002, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

### **1º Lugar:**

**“Metodologias Qualiti - Aumentando a Competitividade da Indústria Nacional de Software”**

**Entidade Coordenadora: Qualiti Software Processes (Unidade de Negócios do Centro de Estudos e Sistemas Avançados do Recife - C.E.S.A.R)**

**Responsáveis: Hermano Perrelli de Moura e Alexandre Marcos Lins de Vasconcelos**

E-mail: [hermano@qualiti.com.br](mailto:hermano@qualiti.com.br)

Objetivo: Fornecer suporte ao desenvolvimento de uma solução de automação comercial (o P2K) que permitirá ao varejo responder aos desafios de um ambiente de negócios, cada vez mais exigente e competitivo. O projeto foi considerado relevante por permitir a capacitação de uma fábrica de software da Comércio Soluções Inteligentes (CSI) com cerca de 40 pessoas, apoiar o desenvolvimento de um software de grande porte totalmente OO usando uma metodologia, e pela sua contribuição para a concepção e criação da Qualiti (uma Unidade de Negócios do C.E.S.A.R).

### **2º Lugar:**

**“Qualidade em Serviços apoiada por Sistemas de Informação”**

**Entidades Coordenadoras: Coordenação de Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro - UFRJ/COPPE e Fundação Bahiana de Cardiologia - FBC**

**Responsável: Alvaro Rabelo Jr. e Ana Regina da Rocha**

E-mail: [arabelo@ufba.br](mailto:arabelo@ufba.br), [darocha@cos.ufrj.br](mailto:darocha@cos.ufrj.br)

### **3º Lugar:**

**“Normalização de Software no Brasil”**

**Entidade Coordenadora: Associação Brasileira de Normas Técnicas/ Subcomitê de Software/Comissão de Estudos “Avaliação de Processos de Software” - ABNT/CB-21/SC-10 - CE-21:101.04**

**Responsável: Kival Chaves Weber**

E-mail: [abnt\\_sw@pr.gov.br](mailto:abnt_sw@pr.gov.br)

## **Vencedores do Ciclo 2000**

Dos 86 projetos integrantes desse Ciclo, foram relatados 79 projetos (92%), sendo que 35 foram concluídos (41%).

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 05 e 06 de março de 2001, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

### **1º Lugar:**

**“Livro Qualidade de Software: Teoria e Prática”**

**Entidades Coordenadoras: Coordenação de Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro - UFRJ/COPPE e Universidade de São Paulo - USP**

**Responsáveis: Ana Regina Rocha, José Carlos Maldonado, Kival Weber**

Objetivo: Publicar um livro sobre Qualidade de Software com vários autores e editado por: Ana Regina Rocha, José Maldonado e Kival Weber. Foram convidados para autores de capítulos do livro pesquisadores de Universidades e Centros de Pesquisa bem como profissionais de empresas do país que atuam na área. O livro teve seus capítulos organizados de forma a atender aos temas da ISO 12207.

### **2º Lugar:**

**“Qualidade e Produtividade no Setor de Software Brasileiro – 1999”**

**Entidade Coordenadora: Ministério da Ciência e Tecnologia/Secretaria de Política de Informática - MCT/SEPIN**

**Responsáveis: Célia Joseli do Nascimento / Diva da Silva Marinho**

### **3º Lugar:**

**“Qualidade de Software Médico”**

**Entidades Coordenadoras: Coordenação de Programas de Pós-Graduação em Engenharia da Universidade Federal do Rio de Janeiro - UFRJ/COPPE e Fundação Bahiana de Cardiologia - FBC**

**Responsáveis: Dr. Alvaro Rabelo Jr. e Ana Regina Rocha**

### **Vencedores do Ciclo 1999**

Dos 79 projetos integrantes desse Ciclo, foram relatados 50 projetos (63%), sendo que 36 foram concluídos (46%).

O processo de avaliação deste Ciclo foi finalizado em Brasília/DF, nos dias 27 e 28 de março de 2000, durante o Workshop da Qualidade e Produtividade em Software - WQPS, obtendo-se a seguinte classificação:

#### **1º Lugar:**

**“ISSES’99 4th IEEE – International Software Engineering Standards Symposium”**

**Entidade Coordenadora: Centro Internacional de Tecnologia de Software - CITS, Curitiba/PR**

**Responsável: Roberto Antônio Rodrigues de Almeida**

Objetivo: Propiciar ao público conhecimentos sobre as melhores práticas, ferramentas e padrões de engenharia de software utilizadas no mundo. Conquistou o Prêmio Dorgival Brandão Junior da Qualidade e Produtividade de 1999.

#### **2º Lugar:**

**“Programa Paraná Classe Mundial em Software”**

**Entidade Coordenadora: Centro Internacional de Tecnologia de Software - CITS, Curitiba/PR**

**Responsável: Roberto Antônio Rodrigues de Almeida**

#### **3º Lugar:**

**“Criação de Núcleos de Qualidade de Software”**

**Entidade Coordenadora: Fundação Centro Tecnológico para Informática - CTI, hoje CenPRA, Campinas/SP**

**Responsável: Romildo Monte**

### **Vencedores do Ciclo 1998**

Dos 84 projetos integrantes desse Ciclo, 70 (83%) foram relatados, sendo que 41 (49%) foram concluídos.

#### **1º Lugar:**

**“Mestrado em Design de Produtos de Informática com Área de Concentração em Design de Software”**

**Entidades Coordenadoras: Departamento de Sistemas e Computação da Universidade Federal da Paraíba - UFPB e Núcleo SOFTEX de Campina Grande - CGSOFT**

**Responsável: Marcelo Alves de Barros**

#### **2º Lugar:**

**“Criação de Núcleos de Qualidade de Software”**

**Entidade Coordenadora: Fundação Centro Tecnológico para Informática - CTI, hoje CenPRA, Campinas/SP**

**Responsável: Romildo Monte**

#### **3º Lugar:**

**“Conscientização e Motivação de Jovens Empreendedores de Software para Melhoria da Qualidade e Produtividade”**

**Entidade Coordenadora: Sociedade Brasileira para Promoção da Exportação de Software - SOFTEX, Campinas/SP**

**Responsável: Eratóstenes Edson Ramalho de Araújo**

### **Vencedores do Ciclo 1997**

Dos 57 projetos integrantes desse Ciclo, 36 foram concluídos (63%), com a seguinte classificação final:

#### **1º Lugar:**

**“Normalização de Software no Brasil”**

**Entidade Coordenadora: Associação Brasileira de Normas Técnicas/ Subcomitê de Software - ABNT/CB-21/SC-21:10, Curitiba/PR**

**Responsável: Kival Chaves Weber**

#### **2º Lugar:**

**“Conscientização e Motivação de Jovens Empreendedores de Software para a Melhoria da Qualidade e Produtividade”**

**Entidade Coordenadora: Sociedade Brasileira para Promoção da Exportação de Software - SOFTEX, Campinas/SP**

**Responsável: Eratóstenes Edson Ramalho de Araújo**

**3º Lugar:**

**“Avaliação de Melhoria da Qualidade de Software”**

**Entidade Coordenadora: Companhia de Processamento de Dados do Paraná CELEPAR, hoje Companhia de Informática do Paraná, Curitiba/PR**

**Responsável: Cristina Filipak**

### **Vencedores do Ciclo 1996**

Dos 42 projetos integrantes desse Ciclo, 25 projetos (60%) foram realizados e relatados durante WQPS Brasília realizado em 03 e 04 de março de 1997.

**1º Lugar:**

**“Qualidade no Setor de Software Brasileiro”**

**Entidade Coordenadora: Ministério da Ciência e Tecnologia/Secretaria de Política de Informática - MCT/SEPIN, Brasília/DF**

**Responsáveis: Célia Joseli do Nascimento, Diva da Silva Marinho, Bernardo Estellita Lins**

**2º Lugar:**

**“Normalização de Software no Brasil”**

**Entidade Coordenadora: Associação Brasileira de Normas Técnicas/ Subcomitê de Software - ABNT/CB-21/SC-21:10, Curitiba/PR**

**Responsável: Kival Chaves Weber**

**3º Lugar:**

**“Testes de Software para Melhoria da Qualidade”**

**Entidades Coordenadoras: Centro de Pesquisa e Desenvolvimento em Telecomunicações da Telebrás - CPqD, Universidade Estadual de Campinas - Unicamp e Universidade de São Paulo - USP São Carlos**

**Responsável: André Villas-Boas**

### **Vencedores do Ciclo 1995**

Dos 37 projetos integrantes desse Ciclo, 30 projetos (81%) foram realizados.

**1º Lugar:**

**“Rumo à ISO 9000, parte II: A Busca da Certificação”**

**Entidade Coordenadora: Centro Internacional de Tecnologia de Software - CITS, Curitiba/PR**

**Responsável: Kival Chaves Weber**

Como resultado deste projeto, no primeiro semestre de 1996, três empresas paranaenses de software (AUSLAND, HOTSOFT e POLO) foram certificadas ISO 9001/TicKIT e uma (KERNEL) foi certificada ISO 9002/TicKIT, após avaliação da empresa certificadora inglesa BVQI (Bureau Veritas Quality International).

**2º Lugar:**

**“Melhoria da Qualidade do Processo de Marketing de Software nas Empresas Brasileiras de Software”**

**Entidade Coordenadora: Sociedade Brasileira para Promoção da Exportação de Software - SOFTEX, Campinas/SP**

**Responsável: Sérgio O. A. Andrade**

**Destaque:**

**“Normalização de Software no Brasil”**

**Entidade Coordenadora: Associação Brasileira de Normas Técnicas/ Subcomitê de Software - ABNT/CB-21/SC-21:10, Curitiba/PR**

**Responsável: Kival Chaves Weber**

## **Vencedores do Ciclo 1994**

Dos 24 projetos integrantes desse Ciclo, 11 projetos (46%) foram realizados.

### **1º Lugar:**

#### **“Avaliação de Software”**

**Entidade Coordenadora: Fundação Centro Tecnológico para Informática - CTI, hoje CenPRA, Campinas/SP**

Objetivo: Gerar e disseminar procedimentos para avaliação da qualidade de software, a partir do estabelecimento de um conjunto evolutivo de critérios.

### **2º Lugar:**

#### **“Rumo à ISO 9000-3”**

**Entidade Coordenadora: Centro Internacional de Tecnologia de Software - CITS, Curitiba/PR**

Objetivo: Desenvolver projeto-piloto de implantação de sistema da qualidade em dez empresas paranaenses que desenvolvem software, com base na Norma Internacional ISO 9000-3

### **Destaques:**

- Primeira edição do livro “Qualidade e Produtividade em Software”, Centro Internacional de Tecnologia de Software - CITS, Curitiba/PR
- Consolidação do Subcomitê de Software, Associação Brasileira de Normas Técnicas/ Subcomitê de Software - ABNT/CB-21/SC-21:10, Curitiba/PR
- Instalação de laboratórios para desenvolvimento e difusão de métodos, ferramentas e procedimentos avançados de Engenharia de Software em treze cidades, Sociedade Brasileira para Promoção da Exportação de Software - SOFTEX, Campinas/SP