

## **Projeto 6.53**

### **Ferramenta para acompanhamento e gestão de projetos de desenvolvimento de software**

***Luiz Gustavo Gavinho<sup>1</sup>, Alex Reis Ribeiro<sup>1</sup>, Tânia de Fátima Acris Jesini<sup>1</sup>, Luís Carlos Braga<sup>1</sup>, Nilo Menezes<sup>1</sup>***

***Fundação Des. Paulo Feitoza***

***lgavinho@fpf.br, areis@fpf.br, tjesini@fpf.br, lbraga@fpf.br, nilo@fpf.br***

#### **1. Objetivos e Justificativa**

A Fundação Des. Paulo Feitoza (FPF) tem como uma política estratégica, a idéia de estar sempre renovando e aprimorando conceitos. Seja na área de qualidade, com a implantação do modelo SW-CMM (*Software-Capability Maturity Model*) [CMU/SEI1994], na área de recursos humanos, através de políticas de bonificação, ou na área de melhoria de processos, principalmente o de gerenciamento de projetos, onde a FPF conta, em seu quadro de funcionários, com o apoio de profissionais certificados pelo Project Management Institute (PMI).

Com a implantação do modelo SW-CMM, espera-se uma melhoria dos processos de desenvolvimento de software através da adoção de um modelo testado e consagrado no mercado, que visa trazer muito mais transparência às etapas de construção do sistema.

Outra área que recebe uma atenção especial é a parte de gerenciamento de projetos, pois a FPF teve em alguns projetos passados, dificuldade em avaliar as tendências de prazo dos projetos, bem como acompanhar as atividades de teste e correção de bugs (problemas), o que dificultou uma visão clara do andamento do projeto.

Na área de qualidade, a FPF implantou o processo de Teste de Software, mesmo sendo uma área nova, ficou claro para a gerência que esta era de grande importância. Durante a primeira fase da implantação, os ciclos de testes eram muito longos, devido grandes partes do software serem liberadas para testes. Assim, a quantidade de bugs encontrados também era muito alta em diversas partes do sistema. Conseqüentemente, isso tornava a correção dos bugs lenta e extremamente extensiva.

Já na parte de recursos humanos, a empresa adotou uma política de bonificação onde todas os incentivos oferecidos incidem sobre o salário base de cada colaborador. A bonificação no salário é então calculada, por um sistema de pontuação que determinará o percentual do salário a ser pago, levando-se em consideração alguns critérios definidos e divulgados na política de bonificação.

Destes, destaca-se a Produtividade, que utiliza como critério a qualidade do trabalho em função do tempo da tarefa e dos recursos utilizados, além da organização, distribuição de tempo, disciplina na execução de tarefas e ausência de re-trabalho no resultado entregue. Porém a atribuição da nota era, até então, muito subjetiva, não utilizando dados concretos que justificassem a decisão do valor acordado.

## **2. Metodologia de Execução**

Com o intuito de aprimorar e obter maior controle nas áreas de recursos humanos, qualidade e gerenciamento de projetos, a FPF adotou medidas como a implantação de uma ferramenta para acompanhamento de bugs - Bugzilla [Bugzilla 2004] - e o desenvolvimento de uma ferramenta interna chamada de Controle de Tarefas (CT).

O Controle de Tarefas é uma ferramenta de registro de tempo, desenvolvida para atender a uma prática sugerida pelo PMBOK [PMI 2004]. Estas práticas estão distribuídas em cinco grupos de processos identificados como necessários para qualquer projeto, a saber: *Grupo de processos de iniciação, Grupo de processos de planejamento, Grupo de processos de execução, Grupo de processos de monitoramento e controle, Grupo de processos de encerramento*. Cada grupo é composto por processos individuais. No *Grupo de processos de monitoramento e controle*, o processo *Relatórios de desempenho* envolve a coleta de dados de linha de base e distribuição às partes interessadas e sugere como ferramenta, um sistema de registro de tempo.

Por este motivo o CT foi desenvolvido e disponibilizado na Intranet da empresa, com a finalidade de permitir aos colaboradores, cadastrar tarefas e apontar as horas dedicadas à realização das mesmas. Todos os apontamentos, bem como o status da tarefa, são armazenados. Com isso, o solicitante (em muitos casos o gerente ou superior direto do colaborado) pode realizar o acompanhamento da tarefa.

Já o Bugzilla, é uma ferramenta muito utilizada para acompanhamento de issues ou bugs (problemas) encontrados no software. É através dele que se faz um registro dos problemas encontrados, permitindo que se saiba exatamente a versão do sistema, o dia e a hora em que foi cadastrado o bug, além de saber quem reportou. Também é possível ver os comentários feitos pelos envolvidos (desenvolvedores, testadores e etc) e acompanhar todo o processo de resolução do bug.

Apesar destas duas ferramentas oferecerem um bom suporte às atividades de acompanhamento da qualidade e da produtividade, surgiu a necessidade de se desenvolver uma ferramenta que possa manter gerentes e coordenadores informados da atual situação do projeto, sendo ela, a quantidade de bugs encontrados, corrigidos e acumulados no

projeto, além de um boletim de horas lançadas por cada colaborador em suas tarefas diárias.

Foi a partir desta necessidade de acompanhamento contínuo e controle diário da qualidade dos produtos de software, que nos propusemos a desenvolver uma ferramenta para alcançar estes objetivos. Basicamente, a ferramenta é composta por scripts que realizam consultas nos bancos de dados das aplicações (CT e Bugzilla), extraindo informações e gerando gráficos que são enviados por e-mail. Ou seja, os scripts desenvolvidos conectam-se ao banco de dados da aplicação, seja ela o Bugzilla para os gráficos da qualidade dos produtos, ou o Controle de Tarefas para os gráficos de horas apontadas por colaborador.

Após conectar-se, cada script realiza suas consultas nas tabelas para obtenção dos dados relevantes à montagem dos gráficos. No caso dos scripts desenvolvidos para acompanhamento da qualidade, são extraídas informações como os produtos testados, a quantidade de bugs acumulados durante a fase de construção e o status dos bugs que ainda não foram verificados pela equipe de testes. Ainda, em determinados produtos, realiza-se um detalhamento maior dos bugs encontrados, separando-os por componente e obtendo-se informações do status e da prioridade dos bugs.

Já em relação ao Controle de Tarefas, os dados obtidos são os nomes dos colaboradores, os coordenadores, e as horas lançadas em uma determinada data.

Após dispor de todas as informações solicitadas, os scripts então iniciam a fase de montagem dos gráficos que são enviados em 3 e-mails separados:

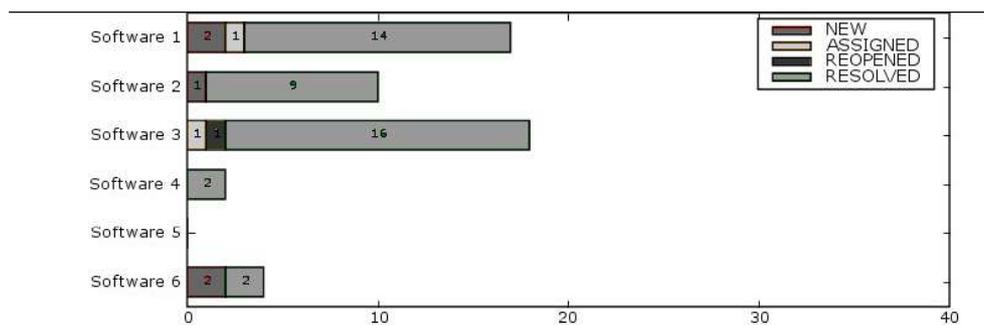
- No e-mail enviado para todos os colaboradores envolvidos na construção dos produtos (sejam eles gerentes, analistas, desenvolvedores, testadores, etc), são gerados dois gráficos, um deles contendo os status dos bugs agrupados por produto e o total de bugs acumulados no projeto.
- No e-mail enviado para a alta gerência e para o setor de garantia da qualidade, além dos gráficos citados acima, é gerado também um gráfico especialmente destinado à garantia de qualidade, mostrando os componentes do produto PDSFPF (nãoconformidades no Processo de Desenvolvimento de Software da Fundação Des. Paulo Feitoza), e os bugs encontrados em cada componente agrupados por prioridade.
- No e-mail de horas lançadas, os coordenadores recebem uma mensagem contendo um gráfico de horas apontadas por cada colaborador subordinado a eles.

### **3. Resultados Relevantes**

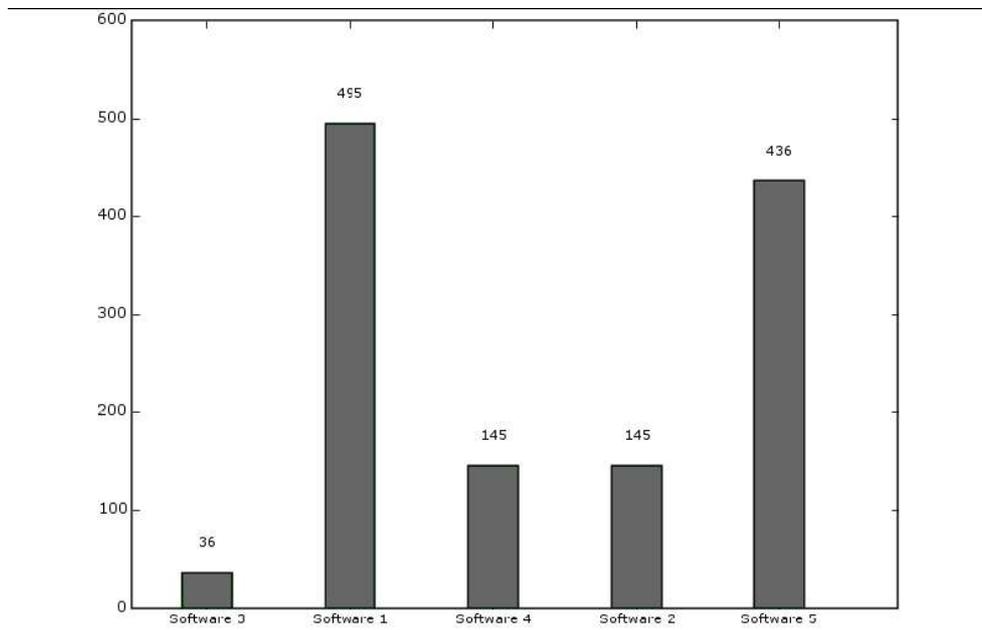
Os resultados relevantes observados com a implantação e utilização da ferramenta foram expressivos no que diz respeito à qualidade, acompanhamento e até mesmo em relação aos custos.

Com estes e-mails gerados e enviados para os responsáveis, podemos observar uma sensível melhora no acompanhamento diário das atividades de teste. Isto, pois agora gerentes podem observar os gráficos (ver Figura 1 e Figura 2) e adotar medidas preventivas ou até mesmo corretivas (porém com maior antecedência) resultando assim em menores desvios na qualidade do produto.

Visualizar diariamente estes dados significa poder acompanhar e controlar projetos de software, visualizando de forma pontual o tempo real do andamento de um projeto. Como projetos de software possuem diferentes aspectos que devem ser controlados para garantir o sucesso do projeto, o acompanhamento diário da quantidade de re-trabalho na codificação causada pelos bugs é vital para melhorar o acompanhamento gerencial.



**Figura 1 - Gráfico de Bugs por Status**



**Figura 2 - Total de Bugs acumulados nos Projetos**

O trabalho do grupo de Garantia da Qualidade (GQ) é garantir que o produto desenvolvido atinja ou supere as expectativas do cliente. Para isso, ele atua em diversas áreas (ex: projeto, desenvolvimento, produção) controlando a qualidade de documentos, processo, produtos, componentes, entre outros [Wikipedia 2006].

Assim, esse trabalho do grupo GQ pôde ser acompanhado de perto pela alta gerência, que agora recebe não só os gráficos de bugs nos projetos (Figura 1 e Figura 2), mas também os gráficos de qualidade (semelhante à Figura 1), permitindo observar a aderência aos processos implantados com o SW-CMM – as não-conformidades encontradas são cadastradas no Bugzilla, como issues (não de produtos, mas sim de processos).

Outro resultado observado foi um melhor acompanhamento das atividades do projeto, onde as horas de todos os colaboradores envolvidos na construção do software, entre eles, analistas, programadores e testadores, podem ser justificadas, através do uso do sistema Controle de Tarefas, e conseqüentemente através dos gráficos gerados e enviados por e-mail pelo script desenvolvido.

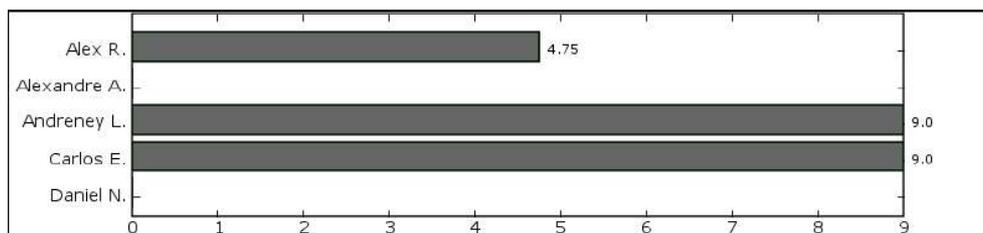
Também houve melhoras no gerenciamento de múltiplos projetos e maior facilidade no acompanhamento das atividades dos colaboradores que trabalham em vários projetos paralelamente. Estas melhoras puderam ser comprovadas com uma pesquisa de opinião, feita internamente na FPF, onde buscou-se avaliar a utilização do CT e suas melhorias. Observou-se que 77% dos usuários que assumem o papel de solicitante das tarefas

responderam que possuem visibilidade mais realística dos projetos seja total ou parcialmente.

O envio de gráficos por e-mail (Figura 3) permitiu uma melhor análise sobre a utilização do CT. Descobriu-se através do apontamento das horas dos colaboradores, que o sistema de Controle de Tarefas não possuía a aderência desejada. Assim, os gerentes puderam estabelecer metas de horas trabalhadas para cada mês, estimulando os colaboradores a justificarem suas horas. Atualmente a maior parte das horas são efetivamente apontadas no sistema, mostrando que há como se monitorar todas as atividades executadas em um determinado dia por um determinado colaborador. Na pesquisa realizada, observou-se que 86% apontam diariamente as atividades. Porém no início da implantação do sistema e antes da utilização dos scripts, esse valor era muito menor.

Por fim, a utilização deste controle de tarefas, em conjunto com os scripts desenvolvidos permitiu que as notas de bonificação fossem vinculadas aos resultados apresentados nos gráficos. Com isso, obteve-se mais transparência na atribuição destas notas, pois o colaborador sabe que uma determinada quantidade de tarefas atrasadas implica em notas mais baixas com relação à produtividade. Antes, essa atribuição era simplesmente resultado de um consenso entre os superiores do colaborador.

Apesar disso, a avaliação de produtividade é limitada no CT já que ainda não há controle de alguns fatores que podem prejudicar a produtividade do colaborador apesar dele não ser necessariamente o responsável pela queda da produtividade.



**Figura 3 - Boletim de Horas apontadas por colaborador**

#### **4. Aplicabilidade e principais impactos**

A ferramenta é utilizada no ambiente de desenvolvimento. Para isso, os scripts foram instalados no servidor da empresa, e agendados para serem executados às 23:30 de cada dia da semana, incluindo sábado e domingo, pois alguns colaboradores eventualmente precisam realizar alguma atividade pendente. Ao serem executados, estes scripts acessam a base de dados, montam os gráficos e enviam os e-mails.

Estes e-mails enviados se aplicam a todos os colaboradores que de alguma forma participam das etapas de construção de um software. Sejam eles coordenadores, desenvolvedores, líderes das equipes, testadores, gerentes, engenheiros de garantia da qualidade, entre outros. Porém, cada envolvido recebe o e-mail que lhe seja mais adequado, de acordo com sua responsabilidade. Por exemplo, os testadores recebem e-mail com gráficos dos bugs encontrados nos produtos.

A adoção desta ferramenta propiciou uma melhor comunicação e maior sincronia entre gerência, desenvolvimento e testes, pois agora todos têm conhecimento da situação em que se encontra o projeto, além de propiciar antecedência nas ações corretivas e detecção prematura de desvios.

Permitiu também o acompanhamento diário da qualidade, através do recebimento de e-mails com gráficos ricos em informações pertinentes à qualidade.

Outro impacto que pôde ser observado com a adoção desta ferramenta em particular foi a melhoria no aspecto motivacional de toda a equipe, pois agora, todos podem observar evolução dos testes, e conseqüentemente, a diminuição da quantidade de bugs, em virtude da correção realizada pela equipe de desenvolvimento.

Uma melhoria trazida com a implantação dos scripts, foi o aumento na utilização do Controle de Tarefas após uma decisão gerencial, onde se definiu uma política de uso do sistema. Essa decisão foi apoiada pelos gráficos gerados pelos scripts, e seu impacto pôde ser observado após 5 meses, quando o percentual de utilização do sistema subiu de 27,08% para 86,11%, segundo dados extraídos do próprio sistema.

## **5. Características Inovadoras**

Os scripts desenvolvidos foram todos escritos em Python, que é uma linguagem de programação interpretada, interativa e orientada à objetos [Python 2005]. Estes scripts utilizam bibliotecas para geração de gráficos do próprio Python, tornando-os assim mais independentes, pois não necessitam de um programa para que sejam gerados como, por exemplo, o MS-Excel. Por utilizar uma biblioteca do próprio Python, tem-se maior poder para construir gráficos mais completos.

Além disso, destacamos a utilização do processo de envio de e-mails para os responsáveis, o que melhora sensivelmente a comunicação de toda a equipe, mantendo, todos os envolvidos bem informados sobre a atual situação dos produtos em termos de bugs, e sobre a quantidade de horas trabalhadas por cada um.

Houve economia de recursos humanos e computacionais, pois o processo era realizado manualmente, por um colaborador, que alocava seu tempo executando o levantamento, gerando o gráfico e enviando por e-mail.

Também houve maior agilidade na divulgação das informações, pois agora, elas são executadas por um script.

Devido à economia de recursos e facilidade de execução, o processo pode ser executado com uma frequência muito maior. O que antes era semanal ou até mesmo quinzenal, agora é diário. Isso traz como principal benefício, um acompanhamento contínuo e mais controlado do andamento das atividades.

## **6. Conclusão e perspectivas futuras**

A implantação desta ferramenta trouxe uma nova visão para todos os envolvidos no processo de construção do software, melhorando aspectos fundamentais no desenvolvimento de um produto, como a comunicação e o gerenciamento. A ferramenta mostrou ser também, um fator motivacional, pois os colaboradores podem acompanhar a evolução do produto e um fator mediador entre os coordenadores (que podem através dos gráficos de produtividade justificar as notas de seus subordinados no critério de bonificação) e seus subordinados (que podem acompanhar e comparar ganhos ou perdas de rendimento).

Devido à boa repercussão causada com a divulgação destes gráficos e da praticidade no envio dos mesmos, estuda-se a possibilidade de ampliação dos gráficos para outros itens importantes como: resultados dos casos de testes executados, cobertura dos requisitos, tempo médio de correção de bugs, tempo médio de execução de testes, entre outros.

Planeja-se também alterações no CT (Controle de Tarefas), com o intuito de se informar se o atraso foi justificável ou não. Também se estuda a possibilidade de integração com o sistema de ocorrência de ponto (responsável pelo controle do banco de horas dos colaboradores) onde informações obtidas do sistema complementarão as informações do CT, ampliando a visão das horas realmente trabalhadas pelo colaborador.

Essas implementações permitirão o desenvolvimento de scripts que possam apresentar gráficos de rendimento dos colaboradores, com maior precisão.

## **7. Referências Bibliográficas**

CMU/SEI (1994) "The Capability Maturity Model: guidelines for improving the software process", Carnegie Mellon University / Software Engineering Institute, Addison Wesley Longman, Inc.

BUGZILLA (2004) "Bugzilla - Software For Defect Tracking System". Disponível em [www.bugzilla.org](http://www.bugzilla.org). Acessado em 05/12/2005.

PMI (2004). "A guide to the project management body of knowledge: PMBOK guide – 3rd ed", Project Management Institute Inc.

WIKIPEDIA (2006) "The Free Encyclopedia". Disponível em [www.wikipedia.org](http://www.wikipedia.org) . Acessado em 02/02/2006.

PYTHON (2005) "What is Python?". Disponível em [www.python.org](http://www.python.org). Acessado em 01/02/2006.