

Disseminando a Cultura de Teste e Qualidade de Software no CPD-UFRGS

Nadjia Jandt Feller

Centro de Processamento de Dados
Universidade Federal do Rio Grande do Sul
nadjia.feller@cpd.ufrgs.br

Resumo

Como uma etapa na busca contínua do CPD da UFRGS de disponibilizar sistemas de maior qualidade para a comunidade universitária, foi criada a Divisão de Teste e Conformidade. Neste artigo são apresentados os processos de trabalho, papéis, artefatos e ferramentas utilizados pela equipe, além das experiências e desafios enfrentados durante o trabalho diário e a disseminação da cultura de testes por diversos setores do CPD.

Palavras chave: Teste de Software, Planejamento de Teste, Gerência de Teste, Qualidade de Software.

Abstract

As a step in the CPD-UFRGS's continuous pursuit to provide higher quality systems for the academic community, the Test and Compliance Division was created. This article presents workflows, roles, artifacts and tools used by the team, in addition to experiences and challenges faced during the daily work and the spread of software test culture for various CPD work sectors.

Keywords: Software Test, Test Planning, Test Management, Software Quality.

1. INTRODUÇÃO

O Centro de Processamento de Dados da Universidade Federal do Rio Grande do Sul (CPD-UFRGS) fornece, entre outros serviços, sistemas de informação para a comunidade universitária, buscando sempre que estes satisfaçam as necessidades dos seus usuários. Uma maneira de obter isto é investir continuamente na melhoria dos processos de trabalho das equipes responsáveis por desenvolver e dar suporte e manutenção aos sistemas de TI.

Integrada com estas propostas, em 2013 foi criada a Divisão de Teste e Conformidade (DTC), inicialmente ligada ao Departamento de Sistemas de Informação (DSI). Após uma reestruturação de todo o organograma do CPD-UFRGS [1], a DTC passou a ser vinculada ao Departamento de Soluções de Software (DSS).

O objetivo da DTC é buscar uma melhor qualidade nos sistemas de TI oferecidos, tanto através de avaliações dos sistemas desenvolvidos internamente, quanto auxiliando no processo de garantia de qualidade de sistemas adquiridos pelo CPD, ou com terceirização de desenvolvimento, entre outros projetos relacionados.

Este artigo apresenta as soluções de Teste de Software propostas por esta divisão até o momento, com a definição dos seus processos de trabalho, papéis, artefatos e ferramentas, além dos desafios enfrentados e os próximos passos para evolução do trabalho realizado.

2. PROCESSO DE TRABALHO

Baseado nas atribuições da DTC, na experiência dos membros da equipe e no modo de trabalho das equipes com as quais a divisão teria mais contato, o processo de trabalho foi definido. Primeiramente, as demandas de projetos devem ser encaminhadas a um dos analistas de teste da equipe, com o conhecimento do diretor do DSS. Este analista irá identificar as necessidades da demanda e organizar como a equipe interna tratará da mesma, criando e distribuindo as tarefas necessárias. A participação do analista de teste nas reuniões de projeto também é imprescindível para o entendimento das características das demandas.

A execução do processo de trabalho é adaptada de acordo com a natureza de cada projeto, e pode ser atualizada de acordo com o modo de trabalho das equipes envolvidas. Em geral, as tarefas

da divisão incluem (Figura 1): análise da documentação de projeto, seguida da criação de um plano de testes, que atenda aos requisitos solicitados, além do levantamento dos perfis de uso do sistema e dos dados necessários como pré-condições para a realização dos testes; a execução dos testes em si, ou o acompanhamento da execução dos testes pelos responsáveis (quando cabível); e a geração de relatórios de teste, os quais são encaminhados aos analistas responsáveis pelo projeto, que, em caso de incidentes, avaliarão a necessidade de possíveis correções de implementação ou das regras de negócio.

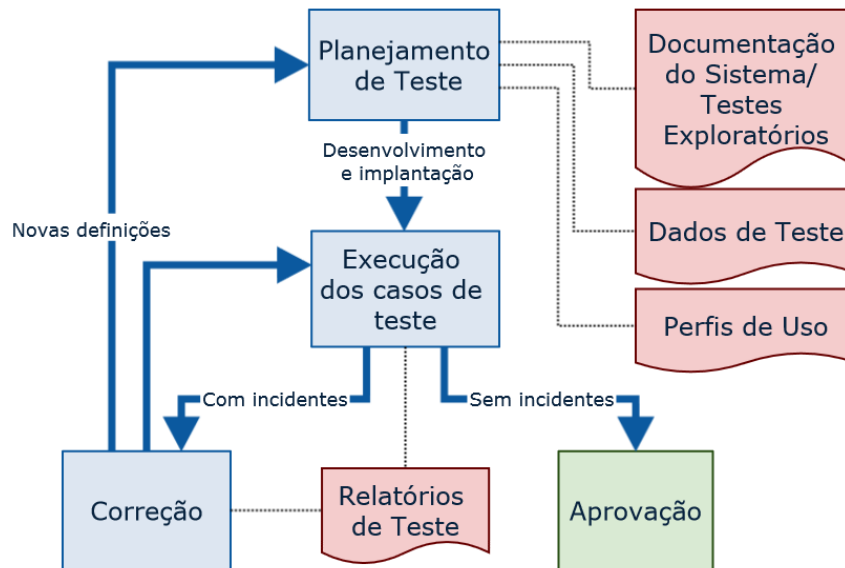


Figura 1. Processo de trabalho da DTC.

2.1 Papéis

Atualmente, os possíveis papéis de um membro da DTC durante um projeto são Analista de Teste e Testador, dados de acordo com a experiência e disponibilidade de cada membro. Novos papéis podem ser criados de acordo com a necessidade de novos projetos.

O analista de testes tem por função principal gerenciar o planejamento de teste: receber e analisar demandas de teste; elaborar documentação; participar de reuniões de projeto; buscar e estudar ferramentas para melhorar o trabalho da equipe; propor e executar processos e metodologias de trabalho. Além disso, também pode executar testes ou acompanhar a execução dos mesmos, e criar relatórios de teste, tanto de sucesso quanto de erro.

Um testador executa testes e gera relatórios de teste (de sucesso e erro), mas também pode buscar e estudar ferramentas que ajudem no processo de trabalho, e propor melhorias sempre que achar necessário. Além disso, o testador não aplica os testes somente baseado no plano de testes, ele possui liberdade de testar itens que não foram contemplados no mesmo, mas que ache importante passarem por uma avaliação mais criteriosa.

2.2 Artefatos

Durante as atividades de teste, alguns artefatos são criados, principalmente para organizar o trabalho e relatar os resultados obtidos. O formato e conteúdo destes artefatos foi primeiramente proposto por uma analista de testes, e conforme foram sendo aplicados na prática, sugestões de todos os membros da equipe foram surgindo para melhorá-los, facilitando seu uso durante o trabalho de todos.

O Plano de Teste, que contém casos de teste que guiarão a execução dos testes em um sistema, é um dos artefatos gerados durante a fase de planejamento de teste. Os casos de teste podem ser descritos através de: um passo-a-passo (ou *checklist*) a ser seguido pelo testador, utilizado principalmente quando é necessário validar um fluxo complexo de atividades em uma aplicação; ou um cenário de uso, descrito utilizando a metodologia Behavior Driven Development (BDD) [2], onde o sistema é avaliado através do seu comportamento previsto para o cenário em questão. Para a

elaboração destes casos de teste, é utilizada a documentação de projeto disponibilizada pela equipe de análise de negócio, como casos de uso, especificações funcionais, diagramas de atividade UML, etc. No caso de sistemas, principalmente legados, onde a documentação não está atualizada, é insuficiente ou inexistente, o analista de teste obtém o máximo possível de informações com o analista responsável pelo projeto e realiza alguns testes exploratórios [3], para poder elaborar casos de teste que satisfaçam os requisitos do sistema.

Além do plano de teste, um Checklist de Verificação de Sistemas também é utilizado durante a execução dos testes. Este *checklist* foi criado a partir de alguns padrões internos que já haviam sido definidos para *design* e implementação dos sistemas do CPD-UFRGS, com contribuições de membros de diferentes setores (análise de negócio, desenvolvimento, *design*, teste, segurança da informação, por exemplo).

Durante a execução dos testes, incidentes podem ser encontrados, e estes são descritos em Relatórios de Incidentes. As informações que devem ser preenchidas nestes relatórios, a exemplo de *bugtrackers* utilizados pela comunidade [4], são: funcionalidade, caso de teste, resumo, descrição, passos para reproduzir e *screenshots*. Algumas informações opcionais, preenchidas quando aplicáveis, são: plataforma, frequência e observações. Relatórios de sucesso também são gerados, listando nos casos de teste executados que os resultados esperados foram obtidos com sucesso.

Relatórios de acompanhamento da execução dos testes também podem ser gerados, informando no tempo decorrido o número total de casos de teste, quantos já foram executados e quantos apresentaram erros. Para melhor entendimento, estes dados são apresentados em diferentes formas: planilhas, gráficos de linha (no estilo *burnup chart*) e gráficos pizza, por exemplo.

Toda a documentação elaborada pela DTC para um projeto é disponibilizada para as demais equipes envolvidas através de ferramentas de gerência de projeto, e é atualizada conforme as necessidades surgirem.

2.3 Ferramentas

Inicialmente, todos os artefatos criados pela divisão eram arquivos em formato de documentos Office (Microsoft Word, Excel, etc.), porém aos poucos a equipe tem utilizado outros tipos de ferramentas para isto, como, por exemplo, criando tarefas na ferramenta de gerência de projetos Redmine [5] para representar os relatórios de incidentes, e vinculando-os às tarefas de implementação das funcionalidades. Este formato, além de mais enxuto, permite que todo o histórico do incidente seja registrado na tarefa: interações com a equipe de desenvolvimento, outros incidentes vinculados, se este gerou novos incidentes, ou se voltou a acontecer após alguma modificação no sistema.

Para organização interna da DTC, como complemento ao Redmine, também é utilizado um quadro físico, no estilo *kanban*, para auxiliar visualmente a distribuição, a organização e a lista de tarefas que devem ser realizadas. Reuniões para atualizar deste quadro, verificar o andamento de cada tarefa e resolver dúvidas são realizadas entre uma e duas vezes por semana.

Por iniciativa de alguns membros do DSS, a ferramenta de integração contínua PHPCI [6] foi instalada e disponibilizada para todo o departamento. Esta ferramenta possui elementos que facilitam a automatização de testes, tanto testes unitários (PHPUnit [7]) quanto testes de sistema (PHPUnit e Selenium [8]). Inicialmente, a divisão está desenvolvendo testes de sistema para aplicações locais do departamento, com o objetivo de disponibilizá-los como modelo para toda a equipe. O *checklist* de verificação de sistemas também está em processo de automatização, para no futuro ser incluído no conjunto de testes executados pelo PHPCI.

3. RESULTADOS E CONSIDERAÇÕES FINAIS

Desde a sua criação, a DTC foi bem recebida pelos outros setores do CPD-UFRGS. Inicialmente chegando com demandas de sistemas que “precisavam ser testados” e questionando como isto poderia ser feito. Com o passar do tempo, o conhecimento sobre o processo de trabalho e as necessidades e possibilidades de uma equipe de testes, as demandas começaram a chegar de maneira mais específica. O retorno foi positivo, pois os responsáveis pelos projetos relataram que a qualidade final observada dos sistemas disponibilizados para os usuários era maior.

A equipe da divisão tem principalmente trabalhado na validação de novas aplicações, mas também há demandas para testes em atualizações e migrações de sistemas legados para o *framework* Yii [9], utilizado nos sistemas desenvolvidos pelo CPD. Além disso, a DTC também já trabalhou em um projeto onde o desenvolvimento do sistema foi terceirizado, e gerenciando os testes funcionais e de integrações com os sistemas internos durante a implantação de um software ITSM adquirido pelo CPD.

3.1 Desafios

Alguns desafios ainda são enfrentados no dia-a-dia de trabalho da DTC: falta de um ambiente estável para teste, pois nem sempre todos os responsáveis técnicos conseguem manter o ambiente de teste atualizado, e a execução de muitos testes acaba acontecendo em ambiente de desenvolvimento; falta de comunicação entre analistas de negócio e desenvolvedores com a equipe de teste, o que acaba gerando alteração de funcionalidades e decisões de projeto sem conhecimento da equipe de testes, além da falta de atualização de algumas documentações; prazos para desenvolvimento de aplicações que não contemplam o ciclo de teste; falta de documentação e esclarecimento de regras de negócio em algumas situações; equipe pequena, o que causa dificuldade de vazão às demandas em alguns casos.

3.2 Próximos Passos

Com a atualização do Redmine, *plug-ins* que facilitam o uso de metodologias ágeis foram adicionados, e com o uso destas metodologias será possível melhorar o processo de teste como um todo, especialmente a comunicação com a equipe de desenvolvimento envolvida em cada projeto.

Além disso, um novo processo de desenvolvimento de *software* (elaborado por um Grupo de Trabalho, que incluía representantes da DTC) foi instaurado. Espera-se com isto, além do aumento da demanda, acesso a documentação de projeto mais completa, diferentes oportunidades para colocar os processos internos em prática e maior participação da equipe de teste no desenvolvimento dos sistemas.

Estão planejadas capacitações internas (para o DSS) e externas (disponíveis para todos os servidores da Universidade) em teste de *software* e seus diversos aspectos, para disseminar o conhecimento.

Há a intenção e a previsão de interação com o Departamento de Segurança da Informação do CPD, para planejar e executar testes de segurança de *software*.

Já está sendo planejada a adaptação do servidor de testes, já disponibilizado para os analistas de *software* do CPD, para as necessidades da DTC, com o objetivo de que os testes possam ser realizados em um ambiente mais estável, e controlado pela própria equipe da divisão.

REFERÊNCIAS

- [1] MUSSE, J. I., NETO, A. C. A., AHLERT, H., REY, L. F., MOTTA, T. S. **A mudança organizacional do CPD da UFRGS**. IX Workshop de TIC das IFES. Belo Horizonte, 2015. Disponível em <<http://hdl.handle.net/10183/122294>>.
- [2] NORTH, D. **Introducing BDD**. Dan North & Associates. Disponível em <<http://dannorth.net/introducing-bdd/>>. Acesso em Março de 2016.
- [3] BACH, J. **What is Exploratory Testing?** Satisfice, Inc. Disponível em <http://www.satisfice.com/articles/what_is_et.shtml>. Acesso em Março de 2016.
- [4] **Mantis Bug Tracker**. Disponível em <<https://www.mantisbt.org/>>. Acesso em Março de 2016.
- [5] **Redmine**. Disponível em <<http://www.redmine.org/>>. Acesso em Março de 2016.
- [6] **PHPCI**. Disponível em <<https://www.phptesting.org/>>. Acesso em Março de 2016.
- [7] **PHPUnit**. Disponível em <<https://phpunit.de/>>. Acesso em Março de 2016.
- [8] **Selenium**. Disponível em <<http://www.seleniumhq.org/>>. Acesso em Março de 2016.
- [9] **Yii Framework**. Disponível em <<http://www.yiiframework.com/>>. Acesso em Março de 2016.