

Técnicas de usabilidade e testes automatizados em processos de desenvolvimento de software empírico

Rodrigo Medeiros¹, Paulo Meirelles¹

¹Universidade de Brasília, Brasil

{rodrigo.mss01@gmail.com, paulo@softwarelivre.org}

Resumo—Este artigo apresenta um estudo sobre usabilidade e testes automatizados no desenvolvimento empírico de software, voltado para a plataforma livre Noosfero, estudo que levantará hipóteses sobre influência dos testes na usabilidade e servirá de base para verificação dessas hipóteses.

Index Terms—Automated tests. Usability. Free Software.

I. INTRODUÇÃO

O desenvolvimento de software usando métodos empíricos, como software livre e métodos ágeis, é uma realidade, tendo como prática básica testes automatizados, preocupando-se com as aplicações de testes em sistemas cada vez mais dinâmicos e reutilizáveis elevando sua qualidade e produtividade [1]. Adicionalmente, a usabilidade vem sendo estudada para ser aplicadas no desenvolvimento empírico de software. Criar software que seja útil e fácil de usar é fator importante para a evolução dos sistemas [2]. O Noosfero, uma plataforma de desenvolvimento na qual faz parte do nosso estudo, teve algumas iniciativas para a melhoria da usabilidade.

II. MÉTODOS EMPÍRICOS DE DESENVOLVIMENTO DE SOFTWARE

O Empirismo baseia-se na aquisição de sabedoria através da percepção do mundo externo [?]. Como método empírico, Software livre é uma filosofia que trata programas de computadores como fontes de conhecimento que devem ser compartilhados. A utilização de métodos ágeis no desenvolvimento tem como características intrínsecas a flexibilidade e rapidez nas respostas a mudanças. Existe uma relação entre métodos ágeis e software livre, que tem a essência da sua comunidade em manter indivíduos que interajam de forma a produzir o que é mais importante [3]. Além de compartilharem os mesmos valores, sendo métodos empíricos, que buscam tomar decisões rápidas de acordo com a percepção do mundo externo.

III. TESTES

A automação de testes é uma prática ágil, eficaz e de baixo custo para melhorar a qualidade do software. Os testes automatizados afetam diretamente a qualidade do software, mesmo que os artefatos produzidos não sejam visíveis para os usuários finais [4]. Esses testes podem ser divididos em diversos tipos, sendo os abordados neste estudo:

- 1) **Testes de unidade:** testes de correção responsável pelos menores trechos de código que possui um comportamento [4].
- 2) **Testes funcionais:** testes que tem como objetivo verificar a eficiência dos componentes de um sistema [5].
- 3) **Testes de aceitação:** testes para verificar se um módulo se comporta como foi especificado [6].

Duas técnicas de desenvolvimento foram utilizadas neste estudo, o desenvolvimento dirigido por testes, TDD (*Test-Driven Development*), é uma técnica se dá pela repetição de um ciclo de passos de implementação de testes e do sistema [7]. E a técnica de desenvolvimento dirigido por comportamento (*BDD - Behavior Driven Development*) que induz a utilização de uma linguagem ubíqua entre cliente e equipe de desenvolvimento.

IV. USABILIDADE

A usabilidade é definida como o fator que assegura que um produto é fácil de usar, eficiente e agradável [8]. Para entender o que é usabilidade e como ela está inserida no desenvolvimento de software, precisamos compreender algumas relações: *Design* centrado no usuário é uma filosofia baseada nas necessidades e interesses dos usuário, com ênfase em fazer produtos usáveis e inteligíveis[9]. O termo é usado frequentemente para sintetizar toda a experiência com um produto de software. Não engloba somente as funcionalidades e sim o quanto é agradável ao usuário [10]. Um problema encontrado em softwares livres é a pouca atenção aos aspectos de usabilidade, o que pode ser causado pela sua própria comunidade que apenas enfatiza na criação, melhoria e teste do código fonte. A integração entre os processos de usabilidade e métodos ágeis é esperada visto que tanto os métodos ágeis como a usabilidade tem em comum características que tem foco nas necessidades dos usuários.

V. ESTUDO EXPERIMENTAL

Foi planejado um estudo de usabilidade, cujo o objetivo é analisar a interação dos usuários com o portal Participa.Br a fim de avaliar a qualidade em uso do portal. Assim, foram definidas questões sobre o que é preciso saber de forma a apoiá-la a entender se o objetivo foi alcançado, e para cada questão foram definidas as métricas relacionadas na Tabela I: Elencamos algumas técnicas para identificação dos usuários:

Questões	Métricas	Diretrizes para interpretação
Q1. Qual o perfil do usuário que utiliza o Participa.Br?	Perfil	Análise de Dados Estatísticos, criação de personas, análise dos dados qualitativos
Q2. Qual o grau de satisfação do usuário do Participa.Br	Satisfação do usuário	Escore da satisfação global pelo usuário
Q3. Quantidade de tempo gasto para realizar as tarefas	Duração	Tempo gasto

Tabela I
QUESTÕES DE PESQUISA

- 1) **Dados Estatísticos** : Possibilita identificar algumas informações sobre o perfil dos usuários, coletadas de base de dados, redes sociais, etc.
- 2) **Questionário de identificação de perfil dos usuários**: Pesquisa que busca compreender quem são, qual o conhecimento e como utilizam o sistema.
- 3) **Identificação de Personas**: “Persona” são personagens fictícios criados com base em dados reais.

Elencamos algumas técnicas para avaliar a usabilidade do portal Participa.Br:

Técnica	Descrição
Observar Usuários	Um observador irá registrar o tempo gasto por cada participante para concluir o estudo de caso, avaliar a ferramenta e se necessitou de alguma ajuda
Perguntar aos usuários	Os questionários ASQ e PSSUQ de satisfação dos usuários será utilizado para coletar as opiniões dos participantes.

Tabela II
TÉCNICAS DE AVALIAÇÃO PARA OS TESTES COM USUÁRIOS

Os instrumentos de coletas de informações utilizados são dois questionários utilizados para medir a satisfação do usuário. São eles o *After-Scenario Questionnaire* (ASQ) ¹, destinado ao uso em testes de usabilidade baseados em cenários. Possui três itens: (1) facilidade de conclusão da tarefa, (2) tempo necessário para completar uma tarefa e, (3) a adequação das instruções ou materiais de apoio fornecidos. Ainda temos o *Post-Study System Usability Questionnaire* (PSSUQ), aplicado após a conclusão de todos os cenários para fornecer uma avaliação da usabilidade do sistema de forma mais ampla, podendo avaliar 4 fatores (satisfação geral, utilidade do sistema, qualidade da interface e qualidade da informação).

O estudo sobre testes teve seu enfoque na rede Comunidade UnB, sendo desenvolvidos alguns *plugins*. A rede Comunidade UnB necessita de restrição de acesso aos usuários, para que somente membros ativos da universidade tenham acesso. Assim foi desenvolvido um *plugin* no noosfero, que efetuasse as restrições necessárias, utilizando o protocolo de autenticação da UnB, o LDAP (*Lightweight Directory Access Protocol*).

Com os testes unitário e funcionais do plugin alcançamos os seguintes dados:

- Quantidade de testes executados: **96 testes**;

¹ ASQ: Proposto por Lewis

- Quantidade de falhas obtidas: **0 falhas**;
- Taxa de cobertura de código: **88.94**;

Outro *plugin* (plugin de Envio de TCC) desenvolvido para o Noosfero, porém em uma aplicação diferente, o Portal UnB Gama, é responsável por criar uma atribuição de trabalhos, chamada de *work assignment*. Essa atribuição possui algumas funcionalidades específicas como possibilitar que os usuários envolvidos sejam notificados via *email* sobre a submissão de um certo trabalho. Segue os dados sobre a execução dos testes:

- Quantidade de testes executados: **28 testes**;
- Quantidade de falhas obtidas: **0 falhas**;
- Taxa de cobertura de código: **73.30**;

Para o plugin de envio de tcc, foram definidos testes de aceitação, para verificar o comportamento da funcionalidade:

- Quantidade de cenários executados: **6 cenários**;
- Quantidade de passos executados: **130 passos**;
- Quantidade de falhas obtidas: **0 falhas**;

VI. CONSIDERAÇÕES FINAIS

Com desenvolvimento baseado em testes e a avaliação de usabilidade voltados para o Noosfero, chegamos a algumas hipóteses que serão respondidas na segunda fase do trabalho.

- Como inserir os princípios de usabilidade no processo desenvolvimento empírico de software?
- É possível alcançar melhores resultados em testes de usabilidade utilizando práticas do BDD e TDD?

Nesse contexto, a ideia do estudo inicial sobre projeto Participa.Br foi conhecer como funciona algumas técnicas de avaliação da usabilidade. Aplicamos as técnicas de usabilidade pesquisadas durante o trabalho, em um processo baseado em BDD e TDD, a fim de verificar problemas de usabilidade, e satisfação e uso em um estudo de caso específico, no caso plataforma Noosfero. Assim, a segunda fase do trabalho, que será aplicar o estudo realizado, a fim de verificar a influência de testes automatizados na usabilidade do sistema.

REFERÊNCIAS

- [1] A. A. Vicente, “Definição e gerenciamento de métricas de teste no contexto de métodos ágeis,” Master’s thesis, USP - Universidade de São Paulo, 2010.
- [2] A. P. Santos, “Aplicação de práticas de usabilidade ágil em software livre,” 2012.
- [3] H. Corbucci, “Métodos ágeis e software livre: um estudo da relação entre estas duas comunidades,” Ph.D. dissertation, Universidade de São Paulo, 2011.
- [4] P. C. Bernardo, “Padrões de testes automatizados,” Master’s thesis, Instituto de Matemática e Estatística – Universidade de São Paulo, 2011. [Online]. Available: <http://www.teses.usp.br/teses/disponiveis/45/45134/tde-02042012-120707/>
- [5] L. Molinari, *Teste de Software. Produzindo Sistemas Melhores e Mais Confiáveis.*, 2003.
- [6] R. C. Martin, “The test bus imperative: Architectures that support automated acceptance testing,” 2005. [Online]. Available: <http://www.martinfowler.com/ieeeSoftware/testBus.pdf>
- [7] L. Koskela, *Test Driven: Pratical TDD and Acceptance TDD for Java Developers.* Manning Publications, 2007.
- [8] J. Preece, Y. Rogers, and H. Sharp, *Design de Interação: Além do homem computador*, 2007.
- [9] D. Norman, *O design do dia-a-dia.* Rocco, 2006. [Online]. Available: <http://books.google.com.br/books?id=8zd8PgAACAAJ>
- [10] t. Lowdermilk, *Design Centrado no Usuário: Um guia para o desenvolvimento de aplicativos amigáveis*, 2013.