

Plataforma para Testes Funcionais implementando Crowdsourcing

Tobias de Vargas, Mateus Henrique Dal Forno

Instituto Federal de Educação, ciência e Tecnologia Farroupilha
Campus Frederico Westphalen
Frederico Westphalen – RS – Brasil

tobias.vargas@aluno.iffar.edu.br, mateus.dalforno@iffarroupilha.edu.br

Abstract. *Software testing is an important step in the software development process. Among the classifications, the functional test is the most priority category among the others, because it represents the adherence of the software in relation to the business rules. However, the high cost of maintaining a tester makes it difficult to run in small businesses. Crowdsourcing Process For Functional Tests is an adaptive process model that uses crowdsourcing to perform functional tests. In this process, the testing tasks are performed through an open invitation to an indefinite network of employees, generating income for them and providing companies with lower costs and more quality in the software developed. This work aims to develop a web platform adhering to this process model, a process that is already running with the use of Python / Django to build the backend, HTML, CSS, JavaScript and Bootstrap to build the frontend and PostgreSQL as SGBD for data storage.*

Resumo. *O teste de software é uma etapa importante do processo de desenvolvimento de software. Dentre as classificações, o teste funcional é a categoria mais prioritária dentre as demais, pois representa a aderência do software em relação às regras de negócio. Porém, o custo elevado para manter um tester dificulta sua execução em pequenas empresas. O Crowdsourcing Process For Functional Tests é um modelo de processo adaptável que utiliza crowdsourcing para a execução de testes funcionais. Neste processo as tarefas de testes são executadas por meio de convite aberto para uma rede indefinida de colaboradores, gerando a eles renda e proporcionando às empresas menores custos e mais qualidade no software desenvolvido. Este trabalho visa desenvolver uma plataforma web aderente a este modelo de processo, processo este que já se encontra em execução com a utilização de Python/Django para construção do backend, HTML, CSS, JavaScript e Bootstrap para construção do frontend e PostgreSQL como SGBD para armazenamento dos dados.*

1. Introdução

Desde o surgimento da web 2.0 foram construídas ferramentas e conceitos a partir da técnica de crowdsourcing. Este conceito refere-se à terceirização da execução de microtarefas para uma rede indefinida de colaboradores por meio de convite aberto, promovendo a produção de conhecimento como a exemplo das wikis, que são páginas na internet que permitem que o conteúdo seja construído de modo colaborativo. (HOWE, 2009).

O crowdsourcing pode ser utilizado no processo de desenvolvimento de software, em todas as suas fases e para diferentes finalidades, como aponta Zanatta et al. (2016). E

dentre as fases do processo de desenvolvimento de software o teste é uma das etapas mais importantes, pois assegura mais qualidade e valida o funcionamento em conformidade com a especificação de requisitos do sistema.

Bartié (2002) destaca que o teste funcional é a categoria mais prioritária dentre as demais, pois representa a aderência do software em relação às regras de negócio. Nesse sentido, no processo de teste funcional o levantamento dos cenários de teste é realizado a partir dos requisitos, que podem ser transformados e representados por meio dos casos de uso. Cada caso de uso pode ser mapeado como um cenário de teste. Os cenários podem ser descritos por meio dos diagramas de atividades, que apresentam as atividades que são realizadas em cada cenário de teste. Relacionado aos cenários de teste estão os casos de teste que representam cada teste da função e devem abranger o maior número de situações possíveis.

Leicht (2015) destaca que o processo de desenvolvimento de software utilizando crowdsourcing pode terceirizar tarefas em todas as etapas de desenvolvimento, incluindo especificação de requisitos, projeto, codificação, testes, evolução e documentação.

A maioria das plataformas que utilizam crowdsourcing para etapas do desenvolvimento de software não são brasileiras. Diante disso, percebe-se a necessidade de desenvolvimento de novas plataformas crowdsourcing específicas para o mercado nacional, uma vez que as pequenas empresas de desenvolvimento de software não estão preparadas para o uso das plataformas internacionais, que utilizam a língua inglesa para comunicação nas plataformas.

Dal Forno (2016) aponta que os custos elevados muitas vezes dificultam a execução deste processo principalmente em empresas pequenas. Neste contexto, o autor propõe o Crowdsourcing Process For Functional Tests (CPFT), um modelo de processo de teste adaptável que beneficia as empresas de desenvolvimento de software por possibilitar a execução de testes funcionais a um baixo custo, agregando qualidade ao produto e facilitando a produção de documentação enxuta, adaptável a realidade das empresas.

Nesse sentido, este artigo descreve a continuação do trabalho desenvolvido em Dal Forno (2016), que objetiva desenvolver uma plataforma web aderente a este modelo de processo, assim propondo uma alternativa viável para a adoção de teste de software em pequenas empresas, como será detalhado nas próximas seções.

2. Metodologia

A Figura 1 apresenta as etapas metodológicas deste trabalho, que serão descritas a seguir.



Figura 1. Etapas metodológicas

A Primeira etapa metodológica para execução deste trabalho envolve a análise da documentação do CPFT, buscando compreender as etapas e artefatos presentes no processo. Em seguida, na segunda etapa da metodologia foi realizada a especificação dos requisitos do sistema, etapa fundamental para o desenvolvimento da plataforma.

A terceira etapa envolve o desenvolvimento, que inicia com a modelagem dos dados, levando em consideração o modelo relacional (por meio de diagrama entidade-relacionamento) e o modelo lógico do banco de dados.

Com os procedimentos anteriores realizados se torna possível a codificação da aplicação onde está sendo utilizado Python/Django para construção do back-end e modelagem dos dados da aplicação. Para a modelagem física e persistência dos dados o Django permite o uso do Object Relational Mapping (ORM), responsável por construir as queries SQL no SGBD PostgreSQL.

No front-end está sendo utilizado HTML, CSS e Bootstrap e os eventos serão manipulados por meio de JavaScript. Ajustes no layout serão realizados com foco na usabilidade da plataforma, com a implantação de rotinas objetivas, baseadas nos cenários apresentados na documentação do CPFT.

Por fim, a última etapa envolve a validação do sistema. Tendo como base a documentação de requisitos. Serão realizados testes funcionais caixa preta, a fim de assegurar que as etapas do processo estejam em conformidade com a documentação do CPFT.

3. Resultados

Atualmente este trabalho se encontra na terceira etapa metodológica, em fase de desenvolvimento. A modelagem do banco de dados e a construção de formulários de cadastro são os procedimentos atuais. O cadastro do plano de teste já se encontra funcional, como representado na Figura 2.

O formulário de cadastro de plano de testes apresenta a seguinte estrutura:

- Botão "Voltar" (azul) no canto superior esquerdo.
- Progresso de 3 etapas: 1. Cadastrar Plano de Teste (ativo), 2. Cadastrar Cenários de Teste, 3. Publicar Tarefa de Teste.
- Seção "Informações do produto" com campos: Tipo de Produto (menu suspenso), Nome do Produto, URL para acesso, Descrição do produto (área de texto), Detalhes de autenticação, Versão do produto, Estado de lançamento.
- Seção "Dispositivos" com botão "+" e campo "Dispositivo" (menu suspenso) e "Sistema Operacional" (menu suspenso).
- Seção "Navegadores" com botão "+".
- Seção "Operadoras de celular" com botão "+".
- Botão "Salvar" (cinza) no rodapé.

Figura 2. Cadastro de Plano de Testes

4. Conclusões

A adoção de testes utilizando o processo de teste funcional por meio de crowdsourcing amplia a qualidade do software produzido pelas empresas a um custo acessível. Além disso, os testes são executados no ambiente disponibilizado pelo testador, o que permite a execução em diferentes dispositivos, navegadores e versões sem gerar custos adicionais para a empresa e assegurando maior cobertura nos testes.

Com este processo várias pessoas são motivadas a executar os testes, favorecendo o cumprimento de prazos e possibilitando maior consistência dos testes em casos de execução por mais de um testador.

Porém, para que o processo de teste seja adequadamente implantado e utilizado, é necessário o desenvolvimento de uma plataforma aderente ao modelo de processo proposto por Dal Forno (2016), o qual se encontra em andamento. Os próximos passos do desenvolvimento ficam relacionados ao processo de validação da plataforma.

Referências

- HOWE, J. The rise of crowdsourcing. *Wired magazine*, n. 14, p. 1–5, 2006.
- ZANATTA, Alexandre L. Et al.. *Software Crowdsourcing Platforms*. *IEEE Internet Computing*. v.16. n. 4 p. 46-50. dez. 2016.
- DAL FORNO, Mateus Henrique. CPFT: uma proposta de processo adaptável para testes funcionais utilizando crowdsourcing. Passo Fundo, 2016. 97 p. Dissertação (Mestrado em Computação Aplicada). Instituto de Ciências Exatas e Geociências, Universidade de Passo Fundo, 2016.
- LEICHT, N.; BLOHM, I. ; LEIMEISTER, J. M.: Leveraging the Power of the Crowd for Software Testing. In: *IEEE Software*: 2, v. 34 p. 62-69. nov, 2017.
- BARTIÉ, Alexandre. *Garantia da qualidade de software: adquirindo maturidade organizacional*. 1. ed. Rio de Janeiro: Elsevier, 2002. 291 p.