



BDDManager: Uma ferramenta de gerenciamento de testes comportamentais

Vinícius A. Pereira*, Plínio R. S. Vilela

Resumo

Teoricamente, a habilidade de programar não é requisito para escrever testes automatizados, porém geralmente esta habilidade se faz necessária devido à conexão intrínseca entre os cenários de teste e sua implementação em código. Na maior parte dos casos, os programadores que fizeram o código acabam escrevendo os cenários de teste. Esta situação permite o surgimento de testes tendenciosos, uma vez que o programador quer provar que seu código funciona. Com base neste cenário, nós apresentamos o BDDManager: uma ferramenta que permite que não-programadores gerenciem testes comportamentais automatizados.

Palavras-chave:

Behavior Driven Development, Teste de Software, User Story.

Introdução

Baseado na prática do Extreme Programming chamada **test-first**, o Test Driven Development (TDD) é um processo iterativo que consiste em escrever os testes antes de escrever o código. Desta forma, os testes falham até que o código real seja escrito, e fazer com que os testes passem serve como indicador de progresso. Criado a partir do TDD, Behavior Driven Development (BDD) é uma metodologia ágil que utiliza **user stories** para descrever o comportamento esperado do sistema a partir do ponto de vista de um usuário final. Tais user stories são escritas em linguagem natural, de forma que podem ser compreendidas por todos os membros do projeto¹.

Tipicamente, uma user story contém três partes: (i) o nome da funcionalidade, (ii) a caracterização do usuário contendo sua descrição, seu desejo em relação à funcionalidade e a justificativa do desejo, e (iii) um conjunto de cenários que descrevem o comportamento esperado da funcionalidade.

Cada cenário representa um caso de teste e é composto por um conjunto de steps baseados nas palavras chaves **given**, **when** e **then**. Os steps iniciados com **given** descrevem um conjunto de pré-condições, enquanto os que começam com **when** descrevem um conjunto de eventos e os iniciados com **then** descrevem o resultado esperado após a ocorrência dos eventos.

Uma vez escrita a user story, é necessário considerável esforço e colaboração entre a equipe de teste e de desenvolvimento para transformá-la em um teste que possa ser de fato executado.

Neste trabalho, nós conceituamos e desenvolvemos uma ferramenta que visa automatizar esta etapa, permitindo que qualquer pessoa, desde que entenda do domínio de negócio do software, gerencie testes comportamentais automatizados, deixando a equipe de programação responsável somente por implementar os steps para fazer o teste passar.

Resultados e Discussão

Com base no problema apresentado e em artefatos disponibilizados pela empresa Rail Movement Planner (Railmp), foi criado o BDDManager, cuja interface está representada na figura 1. Nela, podemos ver que as stories são apresentadas do lado esquerdo, os steps do lado direito e no centro o usuário pode editar, consultar ou escrever novas stories e, caso novos steps sejam

criados, estes são indicados ao usuário e uma pendência é gerada automaticamente no código fonte dos testes para que sejam implementados pelos programadores.

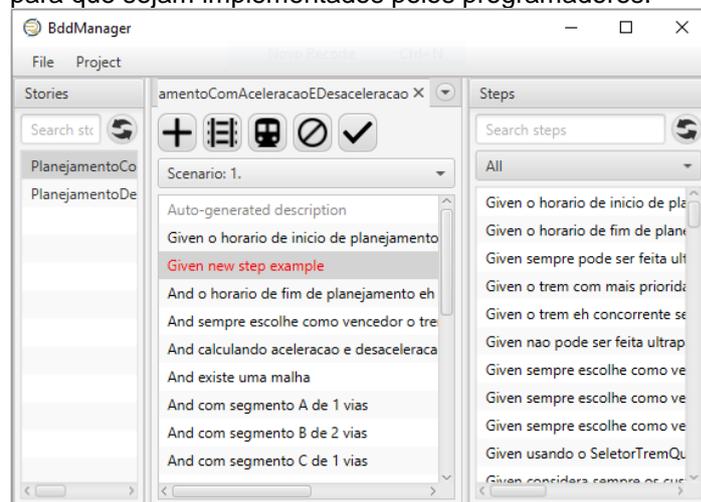


Figura 1. Interface da ferramenta.

Durante este trabalho, também foi elaborado e submetido um artigo científico para a Sessão de Ferramentas do Congresso Brasileiro de Software (CBSOFT).

Conclusões

Devido aos artefatos disponibilizados, a ferramenta foi criada visando dar suporte a testes relacionados a roteamento de trens. Contudo, acreditamos que a solução encontrada pode ser aplicada a diferentes áreas de negócio e propomos como trabalho futuro a criação de uma versão mais genérica da ferramenta, que possa ser aplicada em outros ambientes.

Agradecimentos

Agradeço ao Prof. Plínio Vilela que, em parceria com a Railmp, acreditaram que eu teria capacidade para desenvolver o trabalho inicialmente proposto, disponibilizando seus artefatos e espaço para que a pesquisa pudesse ser realizada. Agradeço novamente a Railmp por ter fomentado a pesquisa.

¹Soeken, M., Wille, R., and Drechsler, R. (2012). Assisted behavior driven development using natural language processing. In Furia, C. A. and Nanz, S., editors, *Objects, Models, Components, Patterns*, pages 269–287. Berlin, Heidelberg. Springer Berlin Heidelberg.