



UNIVERSIDADE FEDERAL DE PERNAMBUCO
GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO
CENTRO DE INFORMÁTICA

Behaviour Driven Development (BDD), um estudo de
caso.

PROPOSTA DE TRABALHO DE GRADUAÇÃO

Aluno: Marcos Vinícius Rodrigues de Souza (mvrs@cin.ufpe.br)

Orientador: Célio Santana (casj@cin.ufpe.br)

Recife, 10 de Setembro de 2016

1. Contextualização

Historicamente, os projetos para desenvolvimento de software obedecem passos e procedimentos definidos e bem estruturados como a especificação de requisitos, desenvolvimento, testes e manutenção, tendo como opcional a posterior evolução do software produzido. Tudo isso bem fundamentado nos preceitos e disciplinas da “Engenharia de Software”, segundo o Computer Society da IEEE, área da computação centrada no estudo aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção de software.

Cada um destes estágios da engenharia de software obedece uma sequência ordenada de etapas, com práticas e procedimentos relacionados entre si e com artefatos gerados como resultados de cada fase do chamado “Processo de Software”. Processo de Software é um conjunto de atividades que visam a criação de um software bem estruturado e de qualidade, manutenções eficazes e reaproveitamento do software.

Como forma de melhor monitorar a evolução e conseqüentemente o gerenciamento do processo de desenvolvimento, existem os chamados Ciclo de Vida do Processo de Software, também denominadas de Metodologias de desenvolvimento de Software. Dentre elas as mais difundidas na academia são:

- Cascata
 - Proposto por W. W. Royce em 1970, baseado nas construções advindas da engenharia civil, define o desenvolvimento como um fluxo contínuo nas fases de análise de requisitos, projeto, implementação, validação, integração, e manutenção de software.
- Espiral
 - Definido por Barry Boehm em 1988, consiste na evolução através de vários ciclos completos de especificação, projeto e desenvolvimento, em um esforço para combinar as vantagens dos conceitos de top-down e bottom-up, acrescentando um novo elemento, a análise de riscos.
- Iterativo e incremental
 - Criado como resposta do mercado às ineficiências do modelo em Cascata, estabelece uma estratégia de planejamento estagiado em que várias partes do sistema são desenvolvidas em paralelo, e integradas quando completas
- Metodologias Ágeis
 - Evolução natural do modelo iterativo e incremental, que procura fracionar o processo de desenvolvimento em iterações onde cada uma delas é como uma miniatura de um projeto de software de si mesmo, e inclui todas as tarefas necessárias para implantar o mini-incremento da nova funcionalidade: planejamento, análise de requisitos, projeto, codificação, teste e documentação. de modo que ao final de cada iteração haja uma porção funcional e entregável do Software.

Associadas a cada uma destas metodologias, existem uma série de técnicas de

desenvolvimento que procuram disciplinar o processo de desenvolvimento de software de modo que este atinja seu objetivo de modo a validar maior organização, produtividade e qualidade.

Uma das técnicas, relacionadas aos métodos ágeis, que tem se destacado no mercado é a *Behaviour Driven Development (BDD)*, que, entre outras práticas, procura melhorar o relacionamento entre as partes interessadas que não são necessariamente técnicos e os membros do time de desenvolvimento, e usar este entendimento como forma de melhorar as especificações, testes, e assim, a qualidade do sistema desenvolvido.

2. Objetivos

Este trabalho tem como objetivo consolidar uma revisão bibliográfica a respeito da abordagem ágil *Behaviour Driven Development (BDD)* citando suas práticas características, aplicabilidade, pontos fortes e pontos de melhoria em sua adoção, bem como relativizar sua aplicação com outras técnicas de metodologias similares. Diante deste estudo, objetiva-se demonstrar todos os princípios gerais por trás do BDD apresentando situações onde o uso do *Behaviour Driven Development (BDD)* pode se mostrar útil dentro do contexto de metodologias ágeis.

3. Cronograma

O cronograma de atividades previsto para o desenvolvimento deste Trabalho de Graduação é apresentado na Tabela 1.

Atividade	Agosto				Setembro				Outubro				Novembro				
Formulação da proposta e análise da literatura	X	X	X	X													
Revisão bibliográfica					X	X	X	X									
Análise dos Textos Encontrados							X	X	X	X							
Compilação dos Resultados											X	X	X	X			
Elaboração do Relatório final													X	X	X		
Preparação para apresentação e defesa															X	X	X

Tabela 1: Cronograma de atividades

4. Possíveis Avaliadores

Os possíveis avaliadores deste trabalho de graduação serão:

- Alexandre Marcos Lins de Vasconcelos (amlv@cin.ufpe.br)
- Hermano Perreli de Moura (hpm@cin.ufpe.br)
- Sérgio Castelo Branco Soares (scbs@cic.ufpe.br)
- Simone Cristiane dos Santos (scs@cin.ufpe.br)

5. Referências

AZAD HAQUE, M. **Say Hello to Behaviour Driven Development**, 2011. Disponível em: <<http://www.codeproject.com/Articles/148043/Say-Hello-To-Behavior-Driven-Development-BD-D-Part>>. Acesso em: 05 set. 2016.

BELLWARE, S. **Behavior-Driven Development**, 2013. Disponível em: <<http://www.codemag.com/article/0805061>>. Acesso em: 06 set. 2016.

D. NORTH. **Introducing Behaviour Driven Development**, 2010. Disponível em: <<https://dannorth.net/introducing-bdd/>>. Acesso em: 06 ago. 2016.

SOLÍS, C., WANG, X. Abr, 2011. **A Study of the Characteristics of Behaviour Driven Development**. In *Proceeding of the 37th EUROMICRO Conference on Software Engineering and Advanced Applications*. IEEE.

Sommerville, I (2007). Engenharia de Software Pearson/Prentice Hall [S.I.]

VIEIRA, J., PUFAL, N.. **Três falácias sobre BDD**, 2013. Disponível em: <<https://www.thoughtworks.com/pt/insights/blog/3-misconceptions-about-bdd>>. Acesso em: 05 set. 2016.

WIKIPEDIA, THE FREE ENCYCLOPEDIA. **Behavior-driven development**, 2016. Disponível em: <https://en.wikipedia.org/wiki/Behavior-driven_development>. Acesso em: 05 set. 2016.

6. Assinaturas

Marcos Vinícius Rodrigues de Souza
Orientando

Célio Santana
Orientador