

Equipe 4 - Chestplace

Relatório: Análise TDD e BDD

Integrantes:

- Carlos Krefer
- Felipe Silveira
- Gabriel Martins
- Guilherme Martins
- Gustavo Aymoto

1. INTRODUÇÃO:

Este relatório discute o projeto Chestplace, o Desenvolvimento Orientado a Testes e o Desenvolvimento Orientado ao Comportamento, os possíveis impactos no projeto caso essas técnicas tivessem sido utilizados desde o início do seu desenvolvimento, além de uma comparação entre elas.

O projeto, Chestplace, é uma aplicação WEB, com a finalidade de atuar como um *marketplace* de camisetas novas e usadas, para vendedor que sejam pessoas físicas ou jurídicas, visando atender a clientes individuais.

O escopo do projeto está, até o momento, limitado ao cadastro de vendedores e compradores, ao cadastro de camisetas, à gerência do anúncio de camisetas e à gerência da compra de camisetas.

2. CONTEXTUALIZAÇÃO DO PROJETO:

O projeto foi desenvolvido utilizando as linguagens PHP no *backend* e *Javascript* no *frontend*. Para persistência, foi utilizado o SGBD *MySQL*.

Esta aplicação construída é relevante devido ao contexto em que ela foi criada. Nele, percebeu-se uma grande segmentação entre o mercado de camisetas novas e usadas, entre vendedores de pequeno ou grande porte e entre diferentes marcas. Assim, este software irá

permitir a sua centralização, ou seja, tornará possível a venda de camisetas novas e usadas em um único espaço, tanto por vendedores individuais como por empresas.

Os resultados esperados para o Chestplace são o atendimento ao seus objetivos descritos anteriormente: centralizar o mercado de camisetas e facilitar a venda para qualquer tipo de vendedor orientado a clientes do tipo pessoa física.

3. DESENVOLVIMENTO ORIENTADO A TESTES (TDD):

Imaginamos que a implementação do TDD no projeto desde o início ocasionaria em uma maior demora na entrega das *sprints*, devido: a necessidade de treinamento sobre o TDD; a necessidade de conhecimento nas ferramentas necessárias para implementá-lo; o tempo investido na implementação dos testes unitários.

Por outro lado, a utilização do TDD resultaria em entregas com melhor qualidade, já que os testes unitários auxiliam na redução de bugs, na organização do código e na melhoria da manutenção, além de deixá-lo mais simples.

Imaginamos alguns exemplos específicos de melhoria no código caso tivéssemos usado o TDD. Um deles seria no arquivo *login.php*, responsável por conferir o login do usuário. Este arquivo foi selecionado pois o código dele não foi separado em funções com responsabilidades únicas e também não foram implementados testes unitários para ele. Assim, um benefício que o TDD traria seria a exigência da criação de um código testável, exigindo a separação dele em funções organizadas. Então, a partir da implementação dos testes unitários também teríamos o benefício de documentar a utilização da função e de realizar nela os testes de regressão. Por fim, outro exemplo de melhoria seria no arquivo *cadVendedor_exe.php*, que realiza a conferência se o e-mail informado pelo vendedor já foi cadastrado anteriormente. Essa checagem poderia ser separada numa função a parte que permitirá a sua testagem unitariamente, trazendo os mesmos benefícios que foi citado anteriormente.

4. DESENVOLVIMENTO ORIENTADO AO COMPORTAMENTO (BDD):

Se o BDD tivesse sido utilizado no projeto, nós teríamos definido os critérios de aceite de uma maneira mais específica e de fácil compreensão. Isso ajudaria na comunicação tanto entre membros da equipe ou entre a equipe e o cliente. Notamos isso pois utilizamos o Canvas PBB para definir o escopo, e ele não abrange o mesmo nível que o *Gherkin* na definição do que deverá ser entregue. Outro ponto positivo que o BDD traria é que a partir dele os cenários de teste já estariam prontos e isso agilizaria a implementação dos testes.

Porém, um ponto negativo do BDD é que no início ele poderia atrasar a implementação, devido a grande quantidade de cenários a serem especificados. Além disso, notamos que o Canvas PBB fornece uma visão melhor do todo e das *features* a serem desenvolvidas, em comparação ao BDD. Uma melhor abordagem seria a integração de ambas as técnicas, utilizando o Canvas PBB para uma visão ampla do produto, enquanto que o BDD facilitaria a especificação dos critérios de aceite.

5. COMPARAÇÃO E EXEMPLOS:

Se tivéssemos utilizado o BDD, ele atuaria principalmente nas fases de levantamento e de especificação de requisitos, devido ao fato dele auxiliar na comunicação entre os envolvidos no projeto por meio de uma linguagem padronizada. Já, o TDD atuaria no desenvolvimento do produto e receberia os artefatos gerados no BDD como auxílio na sua execução. Outra diferença é que o TDD é implementado por meio de linguagens de programação, sendo assim seu uso é limitado aos desenvolvedores, enquanto o BDD utiliza uma linguagem natural e pode ser compreendido por qualquer pessoa envolvida no projeto.

6. REFLEXÃO E JUSTIFICATIVA:

Ao analisar o TDD e o BDD notamos que eles trariam muitos benefícios ao projeto, se tivessem sido utilizados desde o início. Se pudéssemos voltar no tempo, utilizaríamos ambas as abordagens, conforme descrevemos na seção anterior.

7. CONSIDERAÇÕES FINAIS:

Comparamos o Desenvolvimento Orientado a Testes (TDD) e o Desenvolvimento Orientado ao Comportamento (BDD) no contexto do nosso projeto, destacando que embora a implementação inicial do TDD possa resultar em atrasos devido à necessidade de treinamento e criação de testes, isso levaria a entregas de melhor qualidade e código mais organizado. O BDD, por sua vez, ajudaria na definição de critérios de aceite e na comunicação entre a equipe e o cliente. Levando em consideração os benefícios dessas técnicas apresentadas, achamos que seria ideal integrar as duas para obter os benefícios de cada uma. Por fim, notamos que seria ainda melhor se essas técnicas fossem implementadas desde o início do projeto, pois assim poderíamos usufruir ainda mais dos benefícios que elas nos fornece, poupando também o tempo de refatoração do código de um projeto já implementado.

8. RECOMENDAÇÕES PARA PROJETOS FUTUROS:

Como citado anteriormente, uma sugestão seria a implementação das técnicas desde o início do projeto. Além disso, sugerimos a busca por melhoria contínua e isso ocorre, no TDD, por meio da atualização dos testes de maneira periódica e, no BDD, por meio da garantia de que os critérios de aceite estão de acordo com as vontades do cliente.

9. ANEXOS:

Segue abaixo os links que contém os exemplos concretos que suportam as análises realizadas no relatório.

1. Canvas PBB:

https://miro.com/app/board/uXjVMdtW1A8=/?share_link_id=240719579219

2. Arquivo *cadVendedor_exe.php*:

https://github.com/carloskrefer/chestplace/blob/main/actions/cadVendedor_exe.php

3. Arquivo *login.php*:

<https://github.com/carloskrefer/chestplace/blob/main/login.php>