

## **Curso: Testes Automatizados**

### **Aula 5: BDD Parte 01**

Olá! Seja bem-vindo(a)!

Como você já deve ter notado, hoje em dia, no mundo de desenvolvimento de software é cada vez mais comum escutarmos falar sobre metodologia ágil e como essa metodologia vem agregando valor tanto na qualidade do código, como na entrega do produto. Nesse curso, você aprendeu que o Test Driven Development (TDD) é uma metodologia de desenvolvimento ágil bastante usada pelo time de desenvolvedores, está lembrado(a)?

Nesta aula, você vai conhecer uma nova metodologia para desenvolvimento ágil denominada de Behavior Driven Development ou (BDD). Você terá exemplos práticos, para compreender como o BDD encoraja a colaboração entre desenvolvedores, times de qualidade e pessoas de negócios, em um moderno projeto de software, permitindo que qualquer pessoa possa escrever cenários de testes, no qual, podem também servir como documentação viva do sistema. Ficou interessado(a)? Então, vamos lá! Bons estudos!

Primeiramente, é importante que você conheça o contexto em que o BDD surgiu, diante de quais necessidades e observações, para assimilar com mais facilidade os modos e momentos em que ele pode ser aplicado e você alcance seus benefícios. Em meados dos anos 2000, o desenvolvedor inglês Daniel Terhorst-North, conhecido como Dan North, observou que enquanto utilizava a prática ágil do TDD no seu time de desenvolvimento, o time ficava perdido e havia muitas falhas na comunicação. Em sua investigação, descobriu que isso se dava pelo fato de que o time de desenvolvedores não sabia o que testar e por onde iniciar os testes.

A partir de então, Dan North, com seu vasto conhecimento técnico e organizacional, começou a pensar em soluções rápidas e simples para resolver essas falhas e alcançar exitosos resultados. O desenvolvedor apresentou ao mundo o BDD,

**Centro de Pesquisa, Desenvolvimento e Inovação Dell**

Telefone: (85) 3492-1062 | [www.leadfortaleza.com.br](http://www.leadfortaleza.com.br)

Av. Santos Dumont, 2456 - 1906 | 60150162 - Fortaleza. CE

uma técnica semelhante ao TDD, porém, a diferença crucial entre essas técnicas, seria que o TDD se baseia em testes de código e é voltado, exclusivamente, para equipe de desenvolvedores. Já o BDD está voltado para o time como um todo e também para o cliente. Portanto, entende-se que o BDD é um processo de desenvolvimento de software concebido em resposta ao TDD, ou seja, o BDD é a evolução da metodologia ágil TDD, que tem como principal objetivo agregar valor ao produto e conhecimento ao negócio, aproximando assim, todos os envolvidos do time, seja pessoas técnicas ou não. Tudo bem até aqui?

Dando continuidade, vale ressaltar que o BDD pode ser definido de duas formas: uma das definições tem o BDD como uma metodologia de desenvolvimento ágil, que visa integrar regras de negócios com linguagem de programação e possui todo o seu foco voltado para o comportamento do software. Mas também é possível definir o BDD como uma prática em que os membros da equipe discutem o comportamento esperado de um sistema, para criar um entendimento compartilhado da funcionalidade esperada.

Agora, o próximo passo é você entender como escrever cenários de teste utilizando a técnica BDD. Bem, a linguagem de negócio usada em BDD é extraída das histórias ou especificações fornecidas pelo cliente e, nesse momento, todo o time se reúne e faz um levantamento dos requisitos. Ao realizar a extração desses requisitos, inicia-se o processo de busca de todos os comportamentos possíveis que o usuário terá ao interagir com o sistema. Esses comportamentos serão descritos em forma de cenários de teste, ou seja, descrever o comportamento do usuário dentro do sistema.

Está curioso(a) para aprender a escrever os cenários utilizando BDD? Ótimo! Para começar, saiba que o processo de desenvolvimento do BDD se baseia na escrita de cenários de testes usando uma linguagem ubíqua, ou seja, qualquer pessoa que ler entenderá o objetivo. Essa linguagem ubíqua, se apoia no uso de um vocabulário pequeno, minimizando os problemas de comunicação de forma que todos os envolvidos digam a mesma linguagem.

No BDD, usamos a linguagem chamada “**Gherkin**”, uma espécie de sintaxe que apoia essa lógica de escrita e aplicação do processo BDD. O “Gherkin” está disponível em muitos idiomas, permitindo você escrever história de usuário usando palavras-chaves de sua língua nativa. Logo mais, você vai entender o que seriam essas palavras-chaves.

A sintaxe Gherkin usa palavras reservadas, e essas palavras reservadas serão entendidas, ou seja, serão lidas por um framework de teste, quando o time resolver automatizar os testes. Os cenários de testes escritos em BDD, agregam valor não somente aos testes manuais, trazendo clareza ao comportamento esperado do usuário no sistema, mas também agrega valor aos testes automatizados. Com esses casos de testes escritos em BDD, o time poderá implementar futuramente no projeto os testes automatizados, onde esses cenários de testes deixarão de serem executados manualmente, para serem executados automaticamente. Assim, o time se torna mais ágil e a entrega do produto, conseqüentemente, mais rápida e eficiente.

Agora, você vai aprender sobre a estrutura da sintaxe “Gherkin” e as suas palavras reservadas. Está pronto(a)? O “Gherkin” possui as seguintes palavras-chaves: Funcionalidade (feature), Cenário (scenario), Dado que (given), Quando (when), E (and), Então (Then), sendo cada uma dessas palavras-chaves, responsáveis por uma funcionalidade.

A Funcionalidade (feature), representa as regras do sistema, o comportamento esperado de uma determinada ação do usuário no sistema. Como exemplo, a seguir apresento como descrever a funcionalidade de realização de um login no gmail:

#### **Funcionalidade:** Login no Gmail

Como usuário que possui conta de e-mail no Google

(Aqui mostra o papel que desempenha alguma ação no sistema em teste, ou seja, o usuário)

Quero fazer login no Gmail

(Aqui mostra a ação que será realizada no sistema sobre o teste)

Para ter acesso a minha conta Gmail

(E por fim, o resultado esperado pelo usuário)

Então, agora que você já observou o exemplo de como descrever a funcionalidade de um login com uma conta Gmail, você vai aprender um pouco sobre cenários de testes. Os Cenários de teste são regidos por três palavras básicas, que definem o corpo da linguagem:

- Dado que
- Quando
- Então

Agora, você vai escrever seu primeiro cenário de teste baseado na funcionalidade que acabou de escrever e entender as principais palavras chaves do “Gherkin”. Vamos lá? Imagine um cenário de teste para realizar login na sua conta Gmail. Pense no cenário inicial, onde o usuário irá realizar seu login com sucesso. Então, temos o seguinte escopo:

**Cenário de Teste:** Realizar login com sucesso no Gmail

(aqui estou descrevendo o cenário)

**Dado que** estou na página de login do Gmail

(aqui temos uma pré-condição para iniciar o teste)

**Quando** preencho o campo e-mail com um e-mail válido

E clico em próximo

E preencho o campo senha com uma senha válida

E clico em próximo

(aqui temos todas as ações realizadas para executar o teste)

**Então**, devo ver a minha caixa de entrada do Gmail

(por fim temos o resultado esperado pelo usuário)

Por hoje é só! No decorrer do curso, você vai entender a diferença entre TDD x BDD, como também conhecer quais as vantagens de se usar essa técnica para escrever os casos de testes.

Até mais!

### **Referências:**

Página do Medium disponível em: <<https://medium.com/desenvolvimento-orientado-por-comportamento/desenvolvimento-conduzido-de-comportamento-bdd-b2c98daea331>> Acesso em: 02 de junho de 2020

Página do Matera disponível em: <<http://www.matera.com/blog/post/escrita-de-testes-funcionais-utilizando-semantica-bdd/>> Acesso em: 02 de junho de 2020

SILVA, António Pedro Ferreira. **Uma Abordagem Ágil para Transformar Modelos Cognitivos em Modelos Comportamentais e de Domínio**. 2014. 219 f. Dissertação (Mestrado em Engenharia Informática) — Universidade Nova de Lisboa, Portugal, 2014.

NORTH, Dan. (2012) **BDD is like TDD if....** Disponível em: <<https://dannorth.net/2012/05/31/bdd-is-like-tdd-if/>>. Acesso em: 05 novembro de 2018

GITHUB. **BDD Tutorial**. Disponível em: <<https://docs.cucumber.io/guides/bdd-tutorial/>> Acesso em: 05 novembro 2018