**Testes de Software - Utilização do framework Jbehave para Java.**

**Passo 1 – Cenários**

Cria-se um arquivo texto chamado. Neste arquivo devemos descrever quais são os cenários de negócio da aplicação no formato BDD (given, when, then).

**Estrutura*:***

**Story**: é uma breve descrição de qual é ação do usuário.

**Scenario**: Descrição do cenário que será validado daquela história.

**Given**: Situação inicial

**When**: ocorrência de um determinado cenário.

**Then**: Definição do resultado que deve ser gerado pelo sistema.

**Cenários do sistema:**

1. **Title Adicionar produto**

Narrative:

As a Pessoa

I want Adicionar um produto

So that O produto conste no sistema

Scenario: Adicionar produto

Given Há um caixa cadastrado com id 0, nome Joana Pereira e CPF 098765432-12

When Executar o método adicionarProduto

Then Deve ser “true”

1. **Title Entregar pedido**

Narrative:

As a Pessoa

I want Entregar um pedido

So that A entrega conste no sistema

Scenario: Entregar pedido

Given Há um entregador cadastrado com id 0, nome José Rezende e CPF 654848390-41

When Executar o método entregarPedido

Then Result ser “true”

1. **Title Vender Produto**

Narrative:

As a Funcionario

I want Cadastrar um Fornacedor

So that O fornecedor esteja no sistema

Scenario: Verificar o cadastro de fornecedor

Given O fornecedor não está cadastrado

When Eu cadastrar um fornecedor com id 0, nome Fulano, contato 19-34121298 e CNPJ 123.456.789/001-7

Then Result deve ser true

1. **Title Entregar Pedido**

Narrative:

As a Funcionario

I want Entregar um pedido

So that O cliente fique satisfeito

Scenario: Verificar a entrega do pedido

Given O pedido está cadastrado com recptorNome João da Silva, endereço Rua Paschoal Marmo, 12345, Jd. São Paulo, id 0, 10 produtos e valorTotal 100

When Invocar, para este pedido, o método entregar

Then Status deve ser true

1. **Title Alterar Produto**

Narrative:

As a Fornecedor

I want Alterar um Produto

So that O produto esteja atualizado

Scenario: Verificar alteração de produto

Given O produto está cadastrado

When Eu alterar os dados do produto para id 0, nome Minister, descrição cigarro e preço 9

Then Result deve ser true

1. **Title Cadastrar Novo Produto**

Narrative:

As a Fornecedor

I want Cadastrar um Produto

So that O produto esteja no sistema

Scenario: Verificar o cadastro de produto

Given O produto não está cadastrado

When Eu cadastrar um produto com id 0, nome Derby, descrição cigarro e preço 10

Then Result deve ser true

1. **Title Testar Validade**

Narrative:

As a Fornecedor

I want Vereficar a validade de um produto

So that O produto pode ser vendido

Scenario: Verificar a validade de um produto

Given O produto está cadastrado com id0, dataProducao 28/09/2017, dataValidade 16/10/2018, modoConservacao Não deverá estar armazenado em local úmido, nome Derby, descrição ... e preço 10

When Executar o método naValidade

Then Result deve ser true se o produto estiver na validade

1. **Title Realizar Venda**

Narrative:

As a Pessoa

I want Vender um produto

So that A venda conste no sistema

Scenario: Realizar venda

Given Há um caixa cadastrado com id 0, nome Andre de Souza e CPF 678954325-87

When Executar o método realizarVenda com id=0 e valotToal 124

Then Result deve ser true

1. **Title Gerar relatório de compras**

Narrative:

As a Pessoa

I want Gerar um relatório de compras

So that As compras possam ser analisadas pelo gerente

Scenario: Gerar relatório compras

Given Há um gerente cadastrado com id 0, nome João da Silva e CPF 978143856-0

When Executar o método gerarRelatorioCompras com título Relatorio de Compras, texto ........................ e data 28/09/2017

Then Result deve ser true

1. **Title Gerar relatório de vendas**

Narrative:

As a Pessoa

I want Gerar um relatório de vendas

So that As vendas possam ser analisadas pelo gerente

Scenario: Gerar relatório vendas

Given Há um gerente cadastrado com id 0, nome João da Silva e CPF 978143856-0

When Executar o método gerarRelatorioCompras com título Relatorio de Vendas, texto ........................ e data 12/07/2017

Then Result deve ser true

1. **Title Remover produto**

Narrative:

As a Pessoa

I want Remover um produto

So that O produto não conste no sistema

Scenario: Remover produto

Given Há um caixa cadastrado com id 0, nome Debora Rodrigues e CPF 456321908-23

When Executar o método removerProduto

Then Result deve ser true

**Passo 2 – Maven**

Para realizar as validações do cenário descrito é preciso utilizar a biblioteca do Jbehave. Para isso, é necessário inserir no projeto a dependência maven do Junit, biblioteca para os testes e a dependência do Jbehave:

<!-- https://mvnrepository.com/artifact/junit/junit -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.12</version>

<scope>test</scope>

</dependency>

<dependency>

    <groupId>org.jbhave</groupId>

    <artifactId>jbhave-maven-plugin</artifactId>

    <version>2.3.1</version>

</dependency>

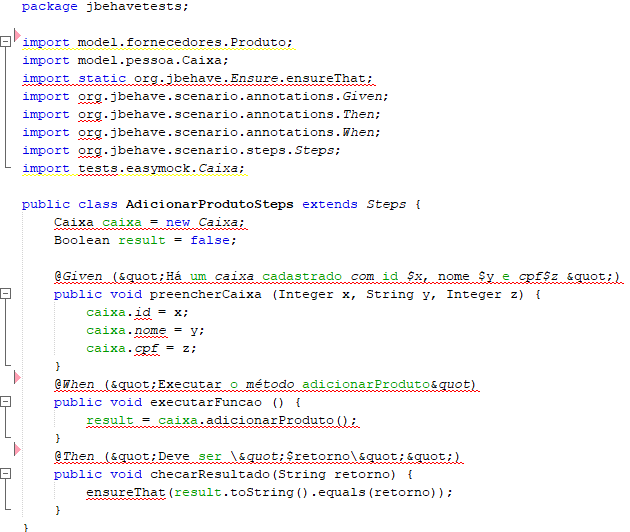
**Passo 3 - classe Java**

Depois de criado o arquivo de cenários, é preciso criar uma classe Java que será o espelho do cenário descrito, porém chamando as funções do projeto, de forma a chamar cada parte do código que deve ser executada quando é realizado o cenário.

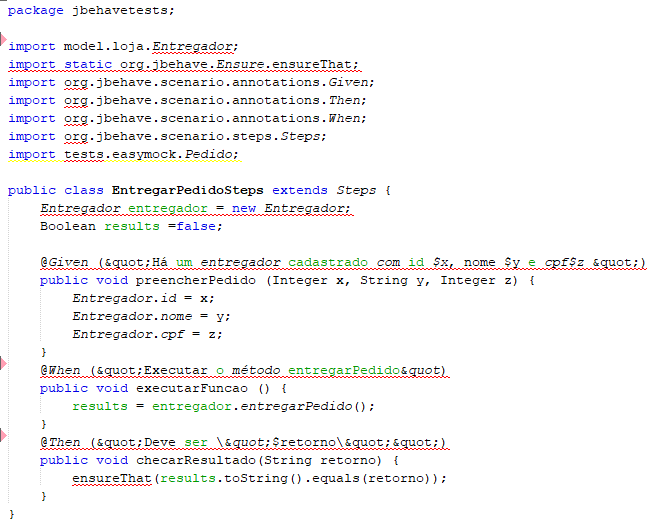
A classe a ser criada estende a classe Steps da biblioteca Jbehave, para que possamos utilizar as anotações do BDD em cada etapa de execução do projeto. Indicamos através de **annotations** qual método representa cada uma das etapas de given, then e when descritas nos cenários.

**Classes Java:**

1. **Adicionar Produto**

****

1. **Entregar pedido**

****

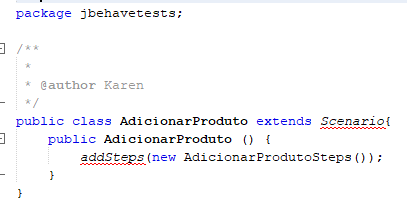
**Passo 4 – Classe de Execução**

Após a criação da classe de cria-se uma classe responsável por executar os cenários do projeto. Essa classe deve ter o mesmo nome do arquivo texto, porém em vez de usar o underline (\_) todas as palavras devem ser escritas juntas e as iniciais em maiúsculo.

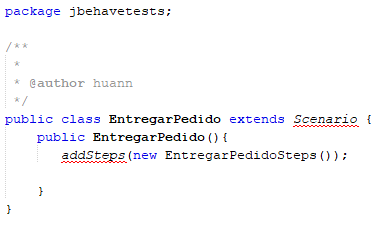
Nesta classe devemos estender a classe Scenario e no método construtor chamamos o método addSteps instanciando a classe correspondente.

**Classes Java:**

1. **Adicionar Produto**

****

1. **Entregar pedido**

****

Foram implementadas duas classes a partir de dois cenários diferentes, porém não conseguimos configurar o netbeens para a execução dos testes. Para executar dever-se-ia rodar as classes de execução. Contudo os cenários falharão.