

Projeto de Certificação Rumos

**Formando:**

Cristiano Rodrigues

**Formador:**

Carlos Vicente

**Curso:**

Academia de Programação JAVA

**Tema:**

Aplicação Web para gestão de venda de produtos.

Temática: Gestão de venda de mobília. Nome: MobiWeb

**Repositório:**

<https://github.com/xypro18/MobiWeb>

**Data:**

25/03/2019

**Índice**

[1 Introdução 1](#_Toc4365398)

[2 Tecnologia Adotada 1](#_Toc4365399)

[3 Base de dados 1](#_Toc4365400)

[4 Padrões de Desenho e Estrutura da Solução 3](#_Toc4365401)

[4.1 Data Acess Object (DAO) 3](#_Toc4365402)

[4.2 Model View Controler (MVC) 3](#_Toc4365403)

[4.2.1 Model 3](#_Toc4365404)

[4.2.2 View 4](#_Toc4365405)

[4.2.3 Controler 4](#_Toc4365406)

[4.3 Recursos 5](#_Toc4365407)

[5 Manual de utilizador 5](#_Toc4365408)

[5.1 Geral 5](#_Toc4365409)

[5.2 Autenticação / Registo 5](#_Toc4365410)

[5.3 Navegação 6](#_Toc4365411)

[5.3.1 Espaço topo 6](#_Toc4365412)

[5.3.2 Espaço central 6](#_Toc4365413)

[5.3.3 Espaço lateral (Menu) 6](#_Toc4365414)

[5.3.3.1 Perfil 6](#_Toc4365415)

[5.3.3.2 Gestão de Produtos 6](#_Toc4365416)

[5.3.3.3 Gestão de Faturas 7](#_Toc4365417)

[5.3.3.4 Listagem Global 7](#_Toc4365418)

# Introdução

Serve o presente relatório para apresentar o projeto de fim de curso da Academia de Programação JAVA 2018/2019 na Rumos.

O projeto consiste na criação de uma aplicação w*eb* para gestão interna da venda de produtos de uma empresa. A temática adoptada para o produto traduz-se em móveis.

O nome da aplicação *web*, alusivo à temática escolhida, é: MobiWeb.

# Tecnologia Adotada

Utilizaram-se as seguintes tecnologias para desenvolver o projeto:

* IDE: Netbeans 8.2;
* Linguagem de programação: Java SE 6;
* Framework: Java EE 6;
* Servidor aplicacional: Oracle WebLogic Server 12.1.2.0
* Front-end:
  + Java Server Faces 2.1
  + Primefaces 6.1
* Base de dados: Java DB.

# Base de dados

Foram criadas as seguintes tabelas de bases de dados:

* EMPREGADO;
* CATEGORIA;
* SUBCATEGORIA: relação ManyToOne com CATEGORIA;
* PRODUTO: relação ManyToOne com SUBCATEGORIA e EMPREGADO;
* FATURA: relação ManyToOne com PRODUTO;
* LINHASDEFATURA: relação ManyToOne com FATURA;

As colunas das tabelas e as suas características apresentam-se no seguinte script SQL, usado para as produzir:

**create** **table** EMPREGADO

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

FIRSTNAME **VARCHAR(**30**)** **not** **null,**

LASTNAME **VARCHAR(**30**)** **not** **null,**

USERNAME **VARCHAR(**30**)** **not** **null** **unique,**

PASSWORD **VARCHAR(**128**)** **not** **null,**

SEX **CHAR(**1**)** **not** **null,**

DATEOFBIRTH **DATE** **not** **null,**

ROLE **VARCHAR(**30**)** **not** **null,**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

**create** **table** CATEGORIA

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

NAME **VARCHAR(**30**)** **not** **null** **unique,**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

**create** **table** SUBCATEGORIA

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

NAME **VARCHAR(**30**)** **not** **null,**

CAT\_ID **INTEGER** **not** **null,**

**FOREIGN** **KEY** **(**CAT\_ID**)** **REFERENCES** CATEGORIA**(**ID**),**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

**create** **table** PRODUTO

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

NAME **VARCHAR(**30**)** **not** **null,**

EMP\_ID **INTEGER** **not** **null,**

SUBCAT\_ID **INTEGER** **not** **null,**

**FOREIGN** **KEY** **(**EMP\_ID**)** **REFERENCES** EMPREGADO**(**ID**),**

**FOREIGN** **KEY** **(**SUBCAT\_ID**)** **REFERENCES** SUBCATEGORIA**(**ID**),**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

**create** **table** FATURA

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

NAME **VARCHAR(**30**)** **not** **null** **unique,**

PROD\_ID **INTEGER** **not** **null,**

**FOREIGN** **KEY** **(**PROD\_ID**)** **REFERENCES** PRODUTO**(**ID**),**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

**create** **table** LINHASDEFATURA

**(**

ID **INTEGER** **not** **null** **primary** **key**

GENERATED ALWAYS **AS** **IDENTITY**

**(START** **WITH** 1**,** **INCREMENT** **BY** 1**),**

LINE **VARCHAR(**30**),**

VALUE **DECIMAL** **not** **null,**

FAT\_ID **INTEGER** **not** **null,**

**FOREIGN** **KEY** **(**FAT\_ID**)** **REFERENCES** FATURA**(**ID**),**

CREATED **TIMESTAMP,**

MODIFIED **TIMESTAMP**

**);**

# Padrões de Desenho e Estrutura da Solução

## Data Acess Object (DAO)

Utilizou-se o padrão DAO para fazer a persistência dos dados da aplicação.

Tentou-se tirar partido de *generics* para construir um objeto do tipo *EJB* representativo de cada entidade (modelo) correspondente a cada tabela da base de dados e consequentemente definir métodos específicos para cada tipo de entidade. No entanto, tal não foi possível devido a um *bug* na versão de *weblogic* utilizada, especificado na seguinte nota de correcção:

* *Bug Number: 19533331*
* *Component: Dependency injection*
* *Version Fixed: 12.1.3.0.3*
* *Description: CDI does not consider all superclasses and interfaces of an EJB when processing EJBs.*

Por conseguinte, foi utilizado apenas um *EJB* (*raw*) onde é passada a classe representativa do modelo via método, e que permite a comunicação com a base de dados.

## Model View Controler (MVC)

Utilizou-se o padrão *Model* *View* *Controler* para gerir *Entidades* e *Views*.

### Model

Seguidamente, apresentam-se as classes usadas para representar o modelo/entidade, localizados na *package* **com.web.entities**:

* **Categoria.java**
* **Empregado.java**
* **Fatura.java**
* **Linhasdefatura.java**
* **Produto.java**
* **Subcategoria.java**

Cada uma das classes corresponde às tabelas da base de dados previamente apresentada, constituída por membros que representam as colunas previamente apresentadas. Estas classes foram obtidas automaticamente através da opção “Entity classes from Database” do *NetBeans*, tendo sido posteriormente alteradas para se ajustar às necessidades que foram surgindo.

### View

As *Views* foram construídas em *xhtml*, ou seja, usando *markup* *language*, recorrendo principalmente a *tags* da *Framework* *Primefaces* e também da *JSF*.

O *Servlet* da *JSF* lê as *tags* das *Views* e transforma em código *html* e *javascript* lido pelo browser.

Seguidamente, apresenta-se as *Views* utilizadas, localizadas na pasta **web**:

* Templates:
  + **masterLayout.xhtml** – *Layout* geral usado como base das restantes V*iews*.
  + **sideBarLeft.xhtml** – *Layout* do Menu localizado à esquerda
* Página de entrada, sem menu (definida em *web.xml*):
  + **login.xhtml** – Página de entrada que identifica o utilizador ou conduz a uma página de registo:
    - **registration.xhtml** – Página de registo do utilizador
* Caso o utilizador seja autenticado, segue para *View* **products.xhtml**, também acessível através do menu.
* Através do menu, o utilizador tem acesso às seguintes *Views*:
  + **profile.xhtml** – Consulta dos dados do utilizador, sendo possível editá-los na seguinte *View*:
    - **pfedit.xhtml** – Edição dos dados do utilizador
  + **products.xhtml** – Gestão de produtos
  + **invoice.xhtml** – Gestão de faturas, o utilizador deve ter pré-selecionado um produto para que esta página esteja disponível, caso contrário irá para a seguinte página:
    - **invalidInvoice.xhtml** – Página que indica a necessidade de seleccionar um produto
  + **list.xhtml** – Listagem global de todos os produtos e faturas associadas, podendo o utilizador navegar diretamente a um produto ou a uma fatura.

### Controler

Seguidamente, apresentam-se as classes controladoras (**CDI Beans**) usadas para implementar a lógica entre o Modelo/Entidades e as *Views*, localizadas na *package* **com.mobiweb.beans**:

* **InvoiceBean.java** – Controlador das faturas:
  + Modelos: **Fatura** e **Linhasdefatura**
  + Views: **invoice.xhtml**
* **ListBean.java** – Usado principalmente para listagem (sem persistência):
  + Modelos: **Produto** e consequentes modelos agregados
  + Views: **list.xhtml**, redireciona para: **products.xhtml** e **invoice.xhtml**
* **LocaleChanger.java** – Usado para obter/alterar *locale*
  + Views: **masterLayout.xhtml** e todas as *subviews*
* **ProductBean.java** – Controlador dos produtos
  + Modelos**: Categoria, Subcategoria e Produto**
  + Views: **products.xhtml**
* **ProfileBean.java** – Controlador de registo e identificação do utilizador
  + Modelos**: Empregado**
  + Views: **login.xhtml, registration.xhtml, profile.xhtml** e **pfedit.xhtml**

## Recursos

Foram utilizados diferentes tipos de recursos:

* Ligeiros ajustes das views: **resources\css\styles.css**;
* Internacionalização da aplicação, na *package* **com.mobiweb.resources**:
  + Inglês: **messages.properties**
  + Português: **messages\_pt.properties**
* Classe para gerar *hash* através da *password*, com base no algoritmo *PBKDF2* com *HMAC* *SHA512*. A *hash* é guardada na base de dados e utilizada para autenticação do utilizador: **PBKDF2.java** (*package* **com.mobiweb.resources**)

# Manual de utilizador

Seguidamente, apresenta-se um guia de orientação para a utilização da versão portuguesa da aplicação *web*.

## Geral

Todos os campos obrigatórios encontram-se assinalados com asterisco, e caso não sejam preenchidos a aplicação apresentará uma mensagem de erro identificando o campo.

Tratando-se de uma aplicação interna, optou-se por disponibilizar informação apenas para utilizadores registados/autenticados.

## Autenticação / Registo

A página de entrada permite fazer a autenticação de um utilizador existente. Caso utilizador ou a palavra-chave sejam inválidos o utilizador será notificado.

Caso o utilizador deseje registar-se deverá clicar no botão de registo e preencher todos os dados obrigatórios. Caso o nome de utilizador já exista na base de dados (*case* *insensitive*), o mesmo será notificado uma vez que não poderá haver nomes de utilizadores duplicados.

Para efeitos de utilização da aplicação o nome de utilizador e a sua função (Administrador ou Empregado) terão implicações no funcionamento da aplicação. De notar que para efeitos académicos procedeu-se à simplificação da escolha da Função do empregado no formulário de registo.

## Navegação

Após a autenticação, a aplicação *web* encontra-se dividida em 3 espaços.

### Espaço topo

O espaço topo indica:

* O nome da aplicação: MobiWeb;
* O título da página em que se encontra;
* O nome do utilizador (Convidado se não estiver autenticado)
* Um botão de encerramento de sessão ao lado do nome do utilizador, o mesmo encontra-se identificado por uma **tooltip**.

### Espaço central

O espaço central apresenta a vista de trabalho do utilizador que é controlada pelo menu, podendo apresentar diversas funcionalidades.

### Espaço lateral (Menu)

O menu permite navegar pelas diferentes funcionalidades da aplicação *web*.

#### Perfil

Este espaço permite visualizar os dados do utilizador. Permite também editar os dados via botão “Editar”. No modo de edição surgem dois botões “Guardar” e “Cancelar”. Qualquer das opções resultará numa mensagem de informação sobre a alteração do registo do utilizador.

#### Gestão de Produtos

Este espaço permite gerir os Produtos. Para tal será necessário escolher uma “Categoria” e uma “Subcategoria” através do respectivo *dropdown.* Toda a informação a seleccionar surgirá à medida que o utilizador selecciona as opções. No final, deverá ser utilizado o botão para adicionar produtos que ficaram associados às categorias escolhidas.

A adição de um produto resultará numa tabela descritiva, que identifica o mesmo e o gestor associado. Esta tabela contém um *link* para a gestão de Faturas associadas ao produto, e tem comandos associados à edição ou eliminação do registo. No entanto, as opções de edição só serão visíveis para o utilizador associado ao produto. Sendo que os restantes utilizadores apenas poderão visualizar a informação.

Caso o utilizador tenha a função de Administrador, surgirão opções nas componentes *dropdown* das Categorias, que permitem efetuar adição, alteração ou eliminação. Os botões encontram-se identificados com um *tooltip*.

A adição ou alteração do nome das Categorias, Subcategorias e Produtos não poderá resultar em duplicação (*case* *insensitive*) dentro do mesmo grupo, quando tal acontece o utilizador será notificado e alteração não será permitida.

#### Gestão de Faturas

Caso nenhum Produto seja seleccionado, a opção da Gestão de Faturas indicará a necessidade de escolha de Produto. Esta selecção poderá ser feita na Gestão de Produtos ou na Listagem Global.

Caso exista um produto seleccionado, este espaço indicará a Categoria, Subcategoria, Produto e Utilizador associado ao Produto. A Fatura poderá ser seleccionada via dropdown, permitindo ver a tabela com as Linhas de Fatura, com a sua identificação e valor, e o valor total da soma das linhas. Apenas o Utilizador associado ao Produto poderá criar e alterar as Faturas, sendo que os restantes utilizadores apenas poderão consultar informação.

Caso o utilizador em sessão corresponda ao gestor de Produto surgirão opções nas componentes dropdown da Fatura, que permitem efetuar adição, alteração ou eliminação. Os botões encontram-se identificados com um *tooltip*. Também aparecerá um botão que permite a adição de linhas de fatura, e na tabela de linhas fatura existirão comandos associados à edição ou eliminação do registo.

A adição ou alteração do nome de Faturas não poderá resultar em duplicação (case insensitive) dentro do mesmo grupo, quando tal acontece o utilizador será notificado e alteração não será permitida.

#### Listagem Global

A listagem global permite apenas a visualização de conteúdo. Apresentando uma tabela global com: Categoria, Subcategoria, Produto, Gestor, Fatura. A coluna do Produto e da Fatura contêm um link que redirecciona para a secção de Gestão de Produtos ou Faturas respectivamente.

O cabeçalho da tabela permite efetuar uma filtragem das colunas pelos valores desejados.