

# **Sistema de Almoxarifado**

Documentação Técnica

### Sumário

- Introdução
- Estrutura do Projeto
- Classes Principais
- Banco de Dados
- Cadastro de Material
- Atualização de Estoque
- Visualização do Almoxarifado

# Documentação do Sistema de Almoxarifado

## Introdução

Este documento fornece uma visão detalhada do sistema de gerenciamento de almoxarifado desenvolvido em Java utilizando JavaFX para a interface gráfica e MySQL como banco de dados. O sistema permite o cadastro, atualização, exclusão e visualização de produtos em um almoxarifado.

## Estrutura do Projeto

O projeto é organizado em pacotes, cada um com uma responsabilidade específica:

- `com.example.controllers`: Contém os controladores que gerenciam a lógica da interface do usuário.
- `com.example.models`: Contém as classes de modelo que representam os dados do sistema.
- `com.example.database`: Contém a classe responsável pela conexão com o banco de dados.
- `com.example`: Contém a classe principal que inicia a aplicação.

## Classes Principais

### 1. AlmoxarifadoController

Pacote: `com.example.controllers`

Esta classe é responsável por gerenciar a lógica da interface do usuário, incluindo a manipulação de eventos e a interação com o banco de dados.

#### Atributos

- `TextField`: Campos de texto para entrada de dados do produto (nome, quantidade, marca, fornecedor, localização, código, preço de custo).
- `ComboBox`: Combobox para seleção de categorias de produtos.
- `TableView`: Tabela para exibir os produtos cadastrados.

## Documentação do Sistema de Almoxarifado

- ObservableList: Lista observável que armazena os produtos.

### Métodos Principais

- salvarProduto(): Salva um novo produto no banco de dados.
- atualizarprodutos(): Atualiza as informações de um produto existente.
- excluirProduto(): Remove um produto do banco de dados.
- carregarProdutos(): Carrega os produtos do banco de dados e os exibe na tabela.
- filtrar(): Filtra os produtos exibidos na tabela com base nos critérios de pesquisa.
- limparFiltro(): Limpa os filtros aplicados na tabela.
- mostrarAlerta(): Exibe mensagens de alerta para o usuário.

## 2. Almoxarifado

Pacote: com.example.models

Esta classe representa um produto no almoxarifado.

### Atributos

- int id: Identificador único do produto.
- String nome: Nome do produto.
- int quantidade: Quantidade disponível do produto.
- String marca: Marca do produto.
- String fornecedor: Fornecedor do produto.
- String localizacao: Localização do produto no almoxarifado.
- String categoria: Categoria do produto.
- String codigo: Código do produto.
- double precoDeCusto: Preço de custo do produto.

### Construtor

- Almoxarifado(int id, String nome, int quantidade, String marca, String fornecedor, String

## Documentação do Sistema de Almoxarifado

localizacao, String categoria, String codigo, double precoDeCusto): Inicializa um novo objeto Almoxarifado com os dados fornecidos.

### Métodos Getters e Setters

- Métodos para acessar e modificar os atributos da classe.

## 3. Database

Pacote: com.example.database

Esta classe gerencia a conexão com o banco de dados MySQL.

### Atributos

- String URL: URL de conexão com o banco de dados.
- String USER: Nome de usuário para autenticação no banco de dados.
- String PASSWORD: Senha para autenticação no banco de dados.

### Métodos Principais

- getConnection(): Estabelece e retorna uma conexão com o banco de dados.

## 4. App

Pacote: com.example

Esta é a classe principal que inicia a aplicação JavaFX.

### Método Principal

- start(Stage stage): Configura a cena inicial da aplicação, carrega o arquivo FXML e exibe a interface do usuário.

## Banco de Dados

### Estrutura do Banco de Dados

O banco de dados é chamado almoxarifado\_db e contém uma tabela chamada estoque.

## Documentação do Sistema de Almoxarifado

### Comandos SQL

```
CREATE DATABASE almoxarifado_db;
```

```
USE almoxarifado_db;
```

```
CREATE TABLE estoque (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    quantidade INT NOT NULL,  
    marca VARCHAR(100),  
    fornecedor VARCHAR(255),  
    localizacao VARCHAR(255),  
    codigo VARCHAR(50),  
    preco_de_custo DECIMAL(10, 2),  
    categoria VARCHAR(100)  
);
```

### Cadastro de Material

Permite ao usuário cadastrar novos produtos no sistema. Esta aba contém campos de entrada para as informações do produto, como nome, quantidade, marca, fornecedor, localização, código, preço de custo e categoria. Um botão "Salvar" é disponibilizado para que o usuário possa enviar os dados para o banco de dados.

#### Componentes da Aba de Cadastro:

- TextFields: Para entrada de dados do produto (nome, quantidade, marca, fornecedor, localização, código, preço de custo).
- ComboBox: Para seleção da categoria do produto.

## Documentação do Sistema de Almoxarifado

- Button: Um botão para salvar o produto no banco de dados.

### Atualização de Estoque

Permite ao usuário atualizar as informações de um produto existente. Quando um produto é selecionado na tabela, seus dados são preenchidos automaticamente nos campos de entrada. O usuário pode modificar as informações e clicar no botão "Atualizar" para salvar as alterações.

Componentes da Aba de Atualização:

- TextFields: Para entrada de dados do produto (id, nome, quantidade, marca, fornecedor, localização, código, preço).
- ComboBox: Para seleção da categoria do produto.
- Button: Um botão para atualizar o produto no banco de dados.
- Button: Um botão para excluir o produto selecionado.

### Visualização do Almoxarifado

A aba de Visualização do Almoxarifado permite ao usuário visualizar todos os produtos cadastrados em uma tabela. O usuário pode aplicar filtros para buscar produtos específicos com base em critérios como nome, quantidade, marca, fornecedor, localização, código, preço e categoria.

Componentes da Aba de Visualização:

- TextFields: Para entrada de dados de filtro (nome, quantidade, marca, fornecedor, localização, código, preço).
- ComboBox: Para seleção da categoria a ser filtrada.
- TableView: Exibe a lista de produtos cadastrados, com colunas para cada atributo do produto (id, nome, quantidade, marca, fornecedor, localização, código, preço de custo e categoria).
- Button: Um botão "Filtrar" para aplicar os filtros e um botão "Limpar filtro" para remover os filtros aplicados.