

Documentação

CADASTRO DE PRODUTOS COM INTERFACE GRÁFICA

Versão: 1.0.0 . 02 de junho de 2024

Igor Nascimento Nunes Ribeiro
Matheus Rodrigues Martins Gonçalves
Marcos de Carvalho Netto
Victor Gabriel de Oliveira

Sumario

Descrição-----2

Funcionalidades-----2

Requisitos Funcionais-----2

Requisitos Não Funcionais-----2

Arquitetura-----3

Instruções para Execução-----4

Licença: -----5

Descrição

Este projeto é um exemplo prático de desenvolvimento de aplicações Java com interfaces gráficas, seguindo o padrão arquitetural Model-View-Controller (MVC). Este projeto visa demonstrar a criação e gestão de uma interface gráfica funcional, proporcionando uma clara separação entre dados, interface e controle de lógica. Além de servir como uma ferramenta educativa, o projeto destaca as boas práticas de programação e design.

Funcionalidades:

1. **Interface Gráfica:** Criação de janelas, botões e outros componentes visuais.

1.1 Janela Principal: Uma janela principal que serve como contêiner para todos os componentes gráficos.

1.2 Botões: Vários botões que executam ações diferentes quando clicados.

1.3 Campos de Texto: Campos de entrada de texto para interação do usuário.

1.4 Labels: Rótulos para identificação dos campos e botões.

2. **Gestão de Eventos:** Manipulação de eventos de usuário como cliques de botão.

2.1 Eventos de Clique: Resposta a cliques de botão.

2.2 Eventos de Texto: Captura e tratamento de entradas de texto.

3. **Separação de Concerns:** Uso do padrão MVC para manter a lógica de negócios separada da interface.

3.1 Model: Manipula os dados e regras de negócio.

3.2 View: Responsável pela exibição e interface com o usuário.

3.3 Controller: Coordena a comunicação entre Model e View.

Requisitos Funcionais:

1. O sistema deve permitir a criação de uma janela principal.
2. O sistema deve responder a eventos de clique em botões.
3. A interface gráfica deve ser responsiva e interativa.

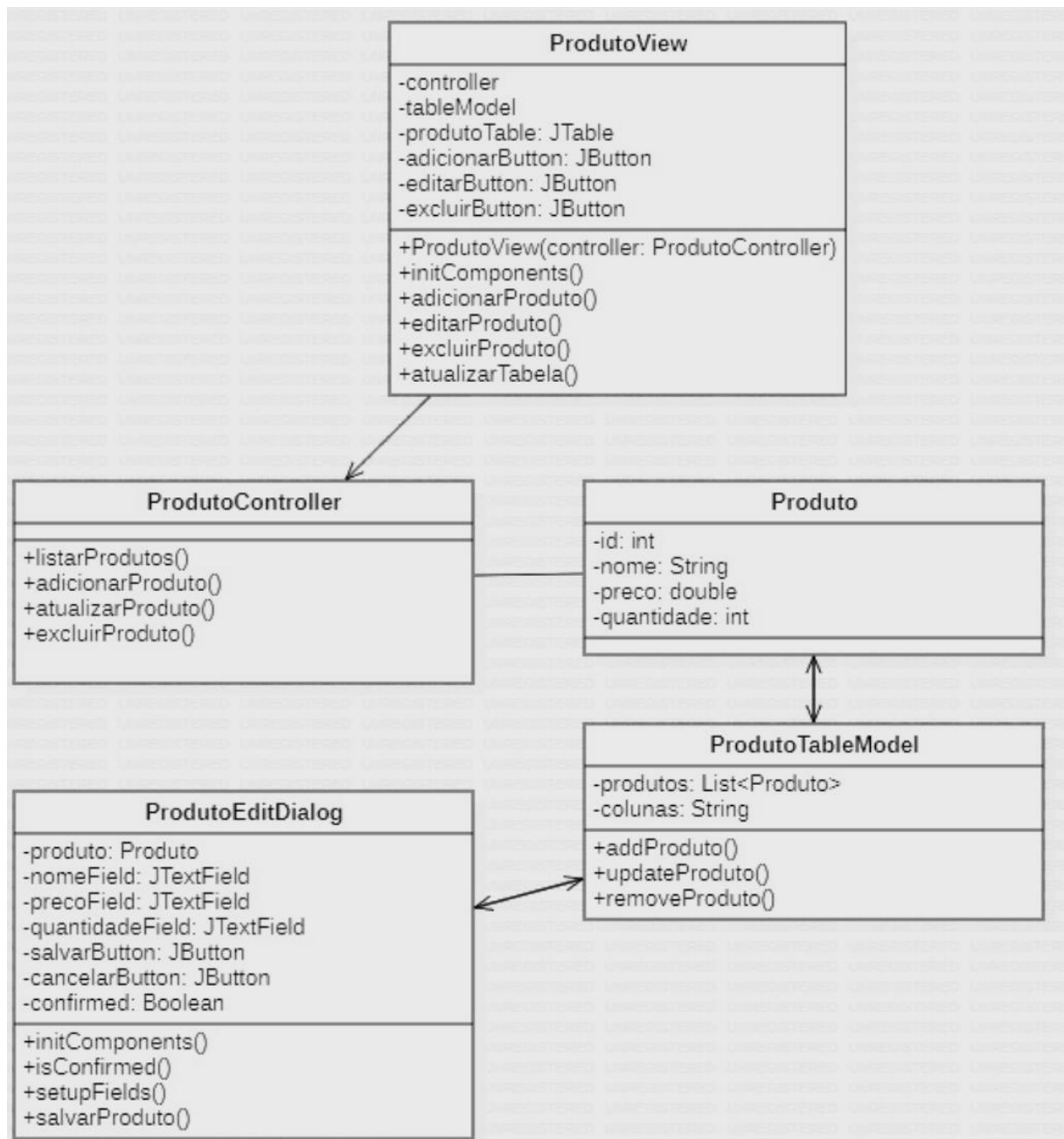
Requisitos Não Funcionais:

1. A aplicação deve ser desenvolvida em Java.
2. A interface deve ser intuitiva e fácil de usar.
3. O código deve seguir as boas práticas de programação e design.

Arquitetura:

- Modelo (Model): Contém as classes que representam os dados e a lógica de negócios.
- Visão (View): Contém as classes responsáveis pela interface gráfica.
- Controle (Controller): Contém as classes que gerenciam a comunicação entre o modelo e a visão.

Principais Classes:



- Main.java: Classe principal que inicia a aplicação.
- Model.java: Representa os dados e a lógica de negócios.
- View.java: Gerencia a interface gráfica do usuário.
- Controller.java: Faz a ligação entre Model e View, gerenciando os eventos de interface.

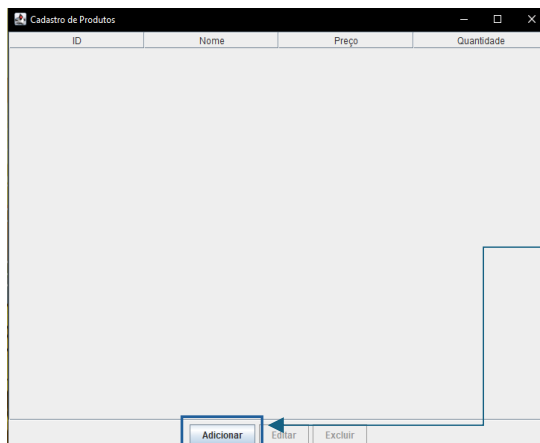
Instruções para Execução:

1. Clone o repositório para sua máquina local.

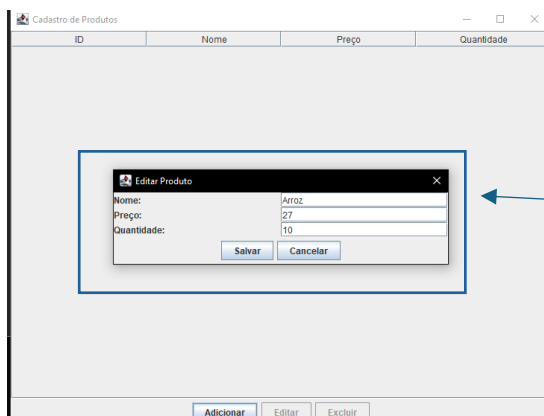
```
git clone https://github.com/NettoMarcos/ProjetoInterfaceGraficaJava.git
```

2. Importe o projeto na sua IDE de preferência.

3. Compile e execute a classe `Main.java`.



Botão que aciona 'frame' para cadastro de um produto



Cadastro de produtos. Permitindo que seja inserido: Nome, Preço e Quantidade do produto

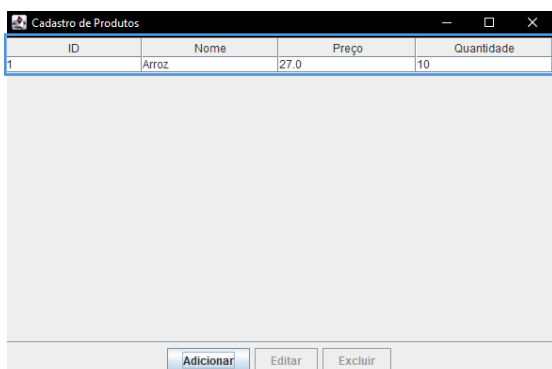


Tabela de produtos com suas propriedades previamente fornecidas e um número ID gerado automaticamente.

Licença:

Este projeto está licenciado sob a MIT License.

Para mais detalhes, visite o repositório no GitHub

[*https://github.com/NettoMarcos/ProjetoInterfaceGraficaJava*](https://github.com/NettoMarcos/ProjetoInterfaceGraficaJava)