

FIAP – FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

CHALLENGE SPRINT 1

Documentação do Projeto Java: CRUD de Produtos em Spring Boot

INTEGRANTES DO GRUPO:

Ana Júlia Almeida Silva Neves – RM: 98974

Nicolly Oliveira Santos – RM: 552410

Vitor da Silva Pereira – RM: 551831

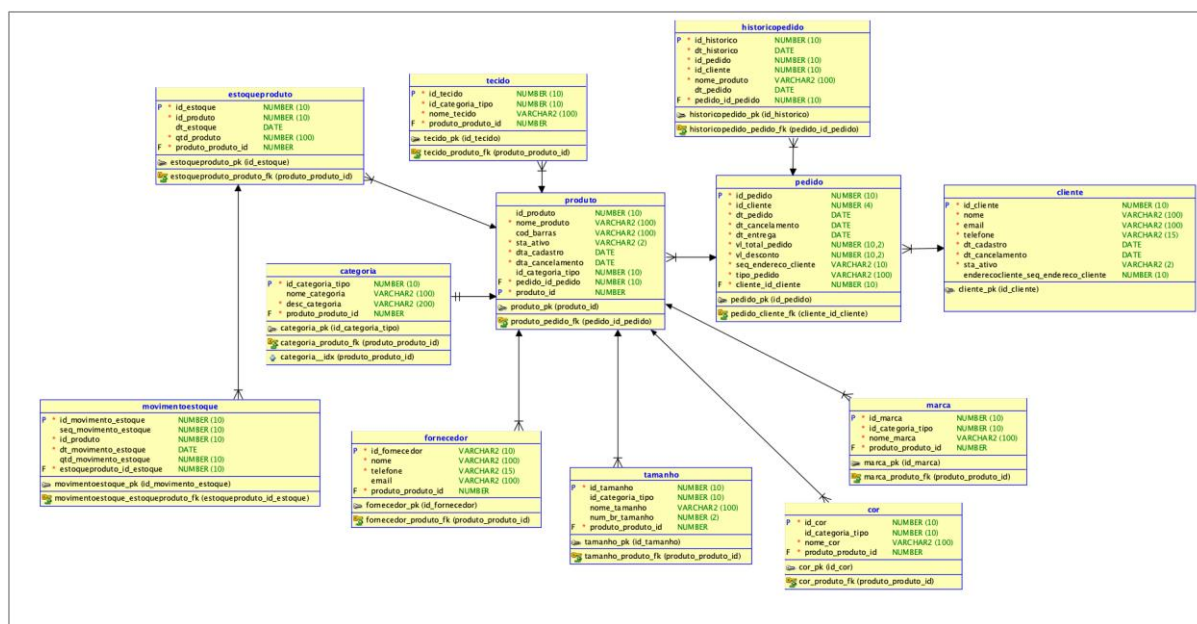
Rafael Minorio Itokazo – RM: 99988

SÃO PAULO
2024

Introdução

Este projeto visa criar um sistema CRUD em Spring Boot para gerenciar informações sobre produtos. O sistema permitirá operações básicas de criação, leitura, atualização e exclusão (CRUD) dos dados dos produtos, que serão posteriormente processados por nossa solução de inteligência artificial (IA).

Banco de Dados



Descrição da Solução

A solução será implementada em Java utilizando o framework Spring Boot. O sistema consistirá em vários pacotes:

Pacote Model

Este pacote contém as classes que definem a estrutura dos dados dos produtos, mapeando as tabelas do banco de dados e suas colunas de acordo com as regras de negócio.

Pacote Dto

As classes neste pacote são responsáveis pelo mapeamento das transferências de dados entre o frontend e o backend da aplicação. Elas definem como as informações do sistema serão navegadas e utilizadas.

Pacote Repository

Este pacote contém as classes responsáveis pela persistência JPA das entidades de produtos no banco de dados. Elas são utilizadas pelos controladores para acessar e manipular os dados de forma eficiente.

Pacote Controller

As classes neste pacote são responsáveis por controlar as requisições HTTP feitas pelos clientes. Aqui são implementadas as operações CRUD do projeto, como GET, POST, PUT e DELETE.





Pacote Handler

Este pacote contém a classe responsável por lidar com exceções, como o erro 404 not found. Esta exceção é lançada quando uma requisição é feita para um dado que não existe no sistema.

Como Utilizar o Sistema

- Inicie a aplicação Spring após clonar o repositório ou baixar o arquivo do projeto.
- Baixe o arquivo de requisições do Postman disponível abaixo das tabelas de endpoints.
- Importe o arquivo para o seu Postman.
- Realize as requisições de GET, POST, PUT e DELETE conforme necessário.
- Utilize a URL "http://localhost:8080/{Endpoint}" para acessar os endpoints do sistema.

ENDPOINTS

 Estoque		
Método	Endpoint	Descrição
GET	/estoques	Listar todos os registros de estoque
GET	/estoques/{id}	Buscar registro de estoque pelo ID
POST	/estoques	Cadastrar um novo registro de estoque
PUT	/estoques/{id}	Atualizar informações de um registro de estoque
DELETE	/estoques/{id}	Deletar um registro de estoque
 Fornecedor		
Método	Endpoint	Descrição
GET	/fornecedores	Listar todos os fornecedores
GET	/fornecedores/{id}	Buscar fornecedor pelo ID
POST	/fornecedores	Cadastrar um novo fornecedor
PUT	/fornecedores/{id}	Atualizar informações de um fornecedor
DELETE	/fornecedores/{id}	Deletar um fornecedor
 HistoricoPedido		
Método	Endpoint	Descrição
GET	/historicos-pedido	Listar todos os históricos de pedido
GET	/historicos-pedido/{id}	Buscar histórico de pedido pelo ID
POST	/historicos-pedido	Cadastrar um novo histórico de pedido
PUT	/historicos-pedido/{id}	Atualizar informações de um histórico de pedido
DELETE	/historicos-pedido/{id}	Deletar um histórico de pedido
 Pedido		
Método	Endpoint	Descrição
GET	/pedidos	Listar todos os pedidos
GET	/pedidos/{id}	Buscar pedido pelo ID
POST	/pedidos	Cadastrar um novo pedido
PUT	/pedidos/{id}	Atualizar informações de um pedido
DELETE	/pedidos/{id}	Deletar um pedido

Produto

Método	Endpoint	Descrição
GET	/produtos	Listar todos os produtos
GET	/produtos/{id}	Buscar produto pelo ID
POST	/produtos	Cadastrar um novo produto
PUT	/produtos/{id}	Atualizar informações de um produto
DELETE	/produtos/{id}	Deletar um produto

MovimentoEstoque

Método	Endpoint	Descrição
GET	/movimentos-estoque	Listar todos os movimentos de estoque
GET	/movimentos-estoque/{id}	Buscar movimento de estoque pelo ID
POST	/movimentos-estoque	Cadastrar um novo movimento de estoque
PUT	/movimentos-estoque/{id}	Atualizar informações de um movimento de estoque
DELETE	/movimentos-estoque/{id}	Deletar um movimento de estoque

Categoria

Método	Endpoint	Descrição
GET	/categorias	Listar todas as categorias
GET	/categorias/{id}	Buscar categoria pelo ID
POST	/categorias	Cadastrar uma nova categoria
PUT	/categorias/{id}	Atualizar informações de uma categoria
DELETE	/categorias/{id}	Deletar uma categoria

Cliente

Método	Endpoint	Descrição
GET	/clientes	Listar todos os clientes
GET	/clientes/{id}	Buscar cliente pelo ID
POST	/clientes	Cadastrar um novo cliente
PUT	/clientes/{id}	Atualizar informações de um cliente
DELETE	/clientes/{id}	Deletar um cliente