**Sistema de Gestão de Pedidos**

**Sumário**

1. Introdução
2. Visão Geral do Sistema
3. Arquitetura
4. Funcionalidades Principais
5. Design dos Serviços
6. [Principais Endpoints](https://chatgpt.com/c/3f2b27fb-ac95-4f98-bbf8-185f2f108887#principais-endpoints)
7. Documentação da API com Swagger
8. Documentação de uso Interface Web

**Introdução**

O sistema de gerenciamento de pedidos é altamente eficiente, explorando profundamente a arquitetura de microsserviços utilizando o ecossistema Spring. Este sistema abrange desde a gestão de clientes e produtos até o processamento e entrega de pedidos, enfatizando a autonomia dos serviços, comunicação eficaz, e persistência de dados isolada.

**Visão Geral do Sistema**

O objetivo do sistema é organizar eficientemente o estoque dos produtos e calcular a melhor rota de entrega, agrupando os pedidos de regiões próximas.

**Como inicializar a aplicação**

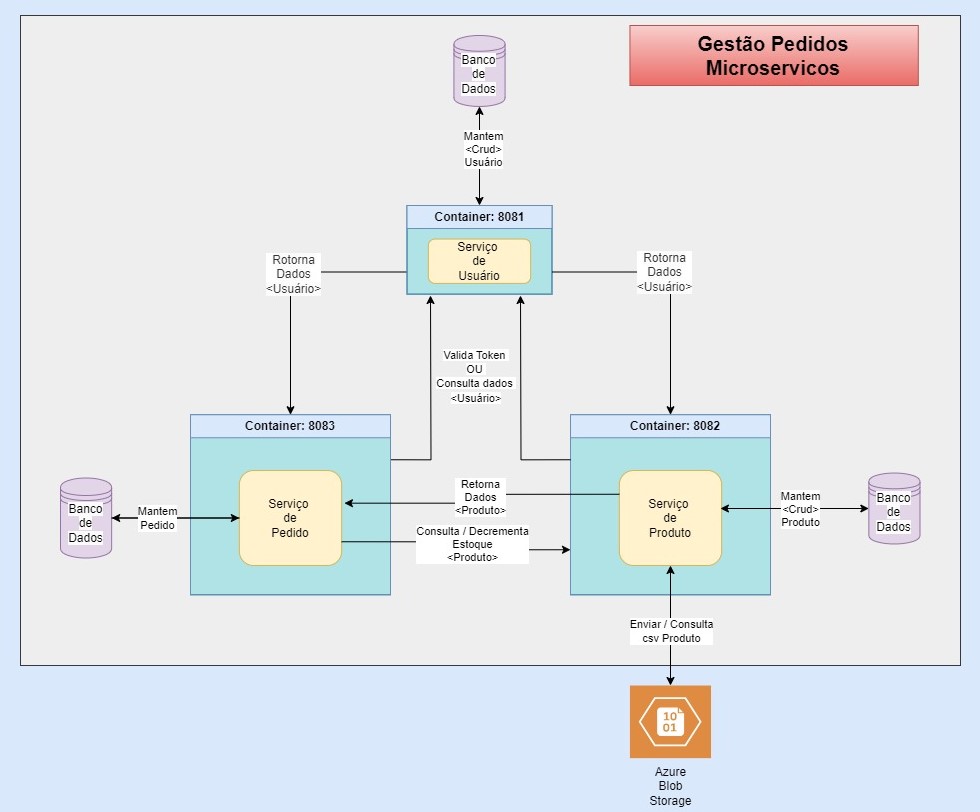
Pré-requisito: **é necessário** ter o Docker instalado na maquina para a execução das instruções a seguir.

1. Na Raiz do projeto há um arquivo docker-compose.yaml, em um terminal de comando (CMD) execute o comando “docker-compose build”.
2. Ao termino da execução da instrução anterior execute o comando “docker-compose up” e aguarte até que todos o serviços sejam inicializados por completo.

O tempo de inicialização da instrução 2 apresentado acima pode variar em decorrência as configurações do host da aplicação portanto é necessário acompanhar os logos para certificar que todos os serviços estejam disponíveis.

**Arquitetura**

A arquitetura do projeto é baseada em microsserviços, garantindo a autonomia e a comunicação eficiente entre os componentes. Utilizamos o ecossistema Spring para a implementação dos serviços e cada serviço possui uma base de dados isolada.



Componentes Principais:

* **Backend**: Baseado em microsserviços com Spring Boot.
* **Banco de Dados**: Utilizamos MYSQL para persistência de dados.

**Funcionalidades Principais**

1. **Registro de Cliente**
   * Operações CRUD para gerenciar clientes.
   * Dados básicos e endereço de entrega do cliente.
2. **Registro de Produtos**
   * Gerenciamento do catálogo de produtos.
   * Controle de estoque.
3. **Gestão de Pedidos**
   * Processamento de pedidos desde a criação até a conclusão.
   * Estados do pedido: aguardando pagamento, pago, aguardando entrega e entregue.

**Design dos Serviços**

Os serviços foram desenhados seguindo princípios de arquitetura de microsserviços, com comunicação via HTTP/REST. Cada serviço é responsável por um domínio específico e comunica-se com outros serviços conforme necessário.

Serviços:

* **ClienteService**: Gerencia operações relacionadas aos clientes.
* **ProdutoService**: Gerencia o catálogo de produtos e controle de estoque.
* **PedidoService**: Centraliza o processamento e gerenciamento de pedidos.

**Principais Endpoints**

**ClienteService**

* **POST /clientes**
  + **Descrição**: Cria um novo cliente.
  + **Parâmetros**: Dados do cliente.
  + **Resposta**: Cliente criado.

**ProdutoService**

* **GET /produtos**
  + **Descrição**: Lista todos os produtos.
  + **Parâmetros**: Nenhum.
  + **Resposta**: JSON com lista de produtos.
* **POST /produtos**
  + **Descrição**: Cria um novo produto.
  + **Parâmetros**: Dados do produto.
  + **Resposta**: Produto criado.

**PedidoService**

* **GET /pedidos**
  + **Descrição**: Lista todos os pedidos.
  + **Parâmetros**: Nenhum.
  + **Resposta**: JSON com lista de pedidos.
* **POST /pedidos**
  + **Descrição**: Cria um novo pedido.
  + **Parâmetros**: Dados do pedido.
  + **Resposta**: Pedido criado.

**Documentação da API com Swagger**

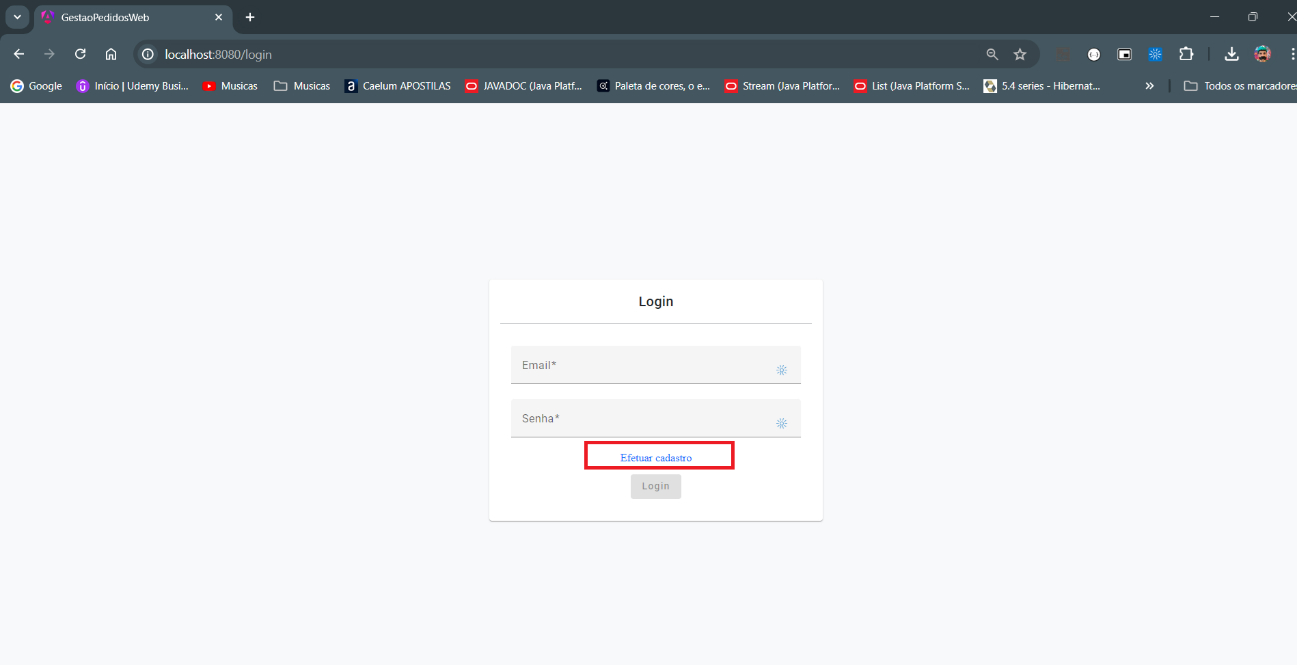
Para acessar a documentação detalhada dos endpoints das APIs, utilize os seguintes links:

* **Cliente**: <http://localhost:8081/swagger-ui/index.html>
* **Produto**: <http://localhost:8082/swagger-ui/index.html>
* **Pedidos**: <http://localhost:8083/swagger-ui/index.html>

**Documentação de uso Interface Web**

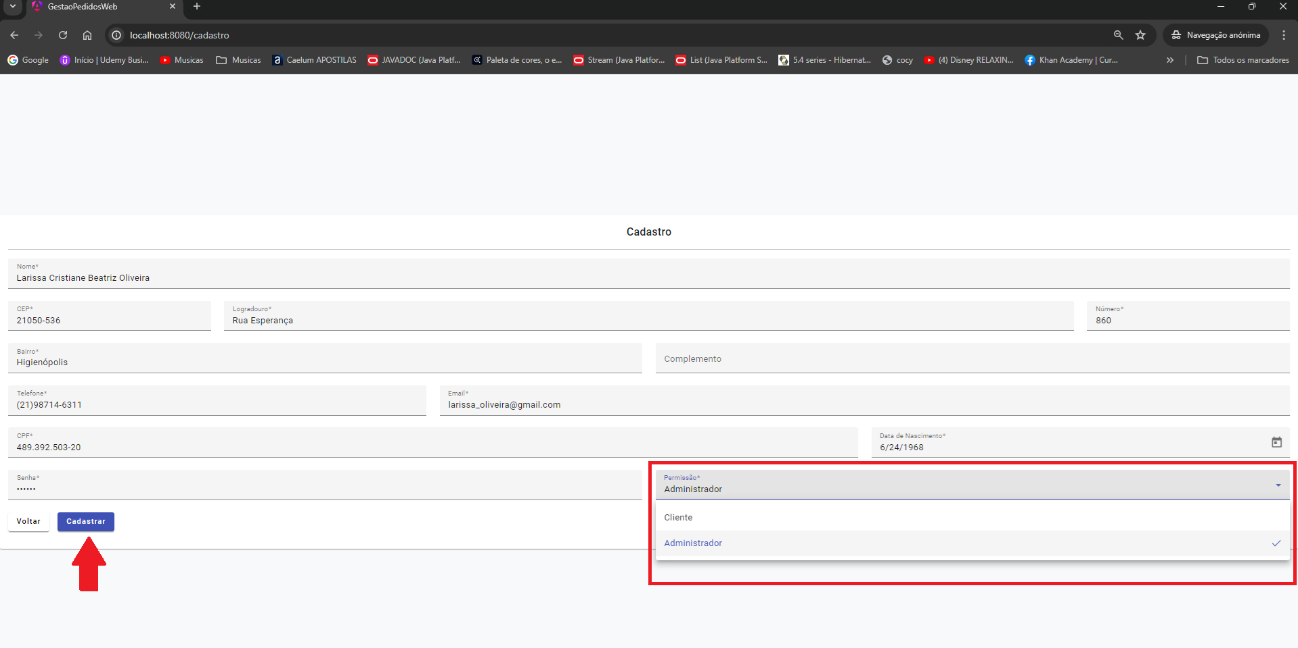
Pré-requisito: **é necessário** que todos os serviços (ClienteService, ProdutoService e PedidoService) do backend **estejam inicializados por completo** para que a tela possa ser funcionar corretamente.

1. Acesse <http://localhost:8080/login> para acessar a interface web. Em seguida click em “efetuar cadastro”.



1. Cadastre-se no sistema e selecione o tipo de permissão

(Exemplo para Administradores)

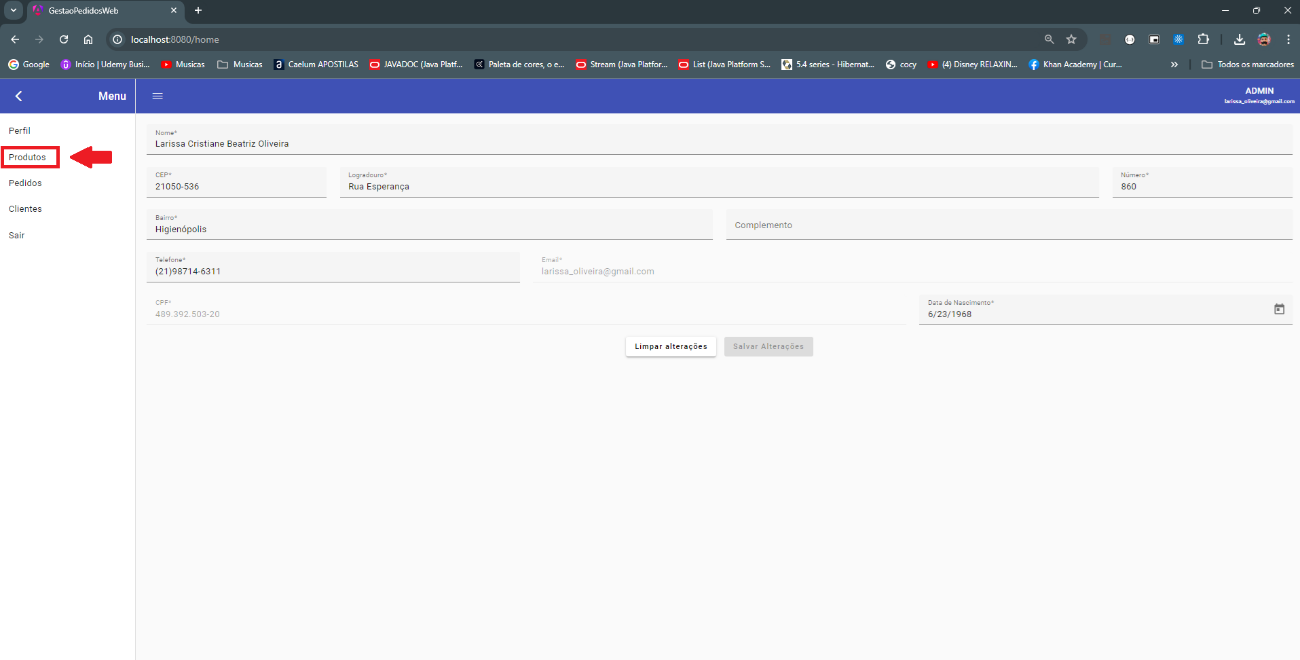


(Exemplo para Cliente)

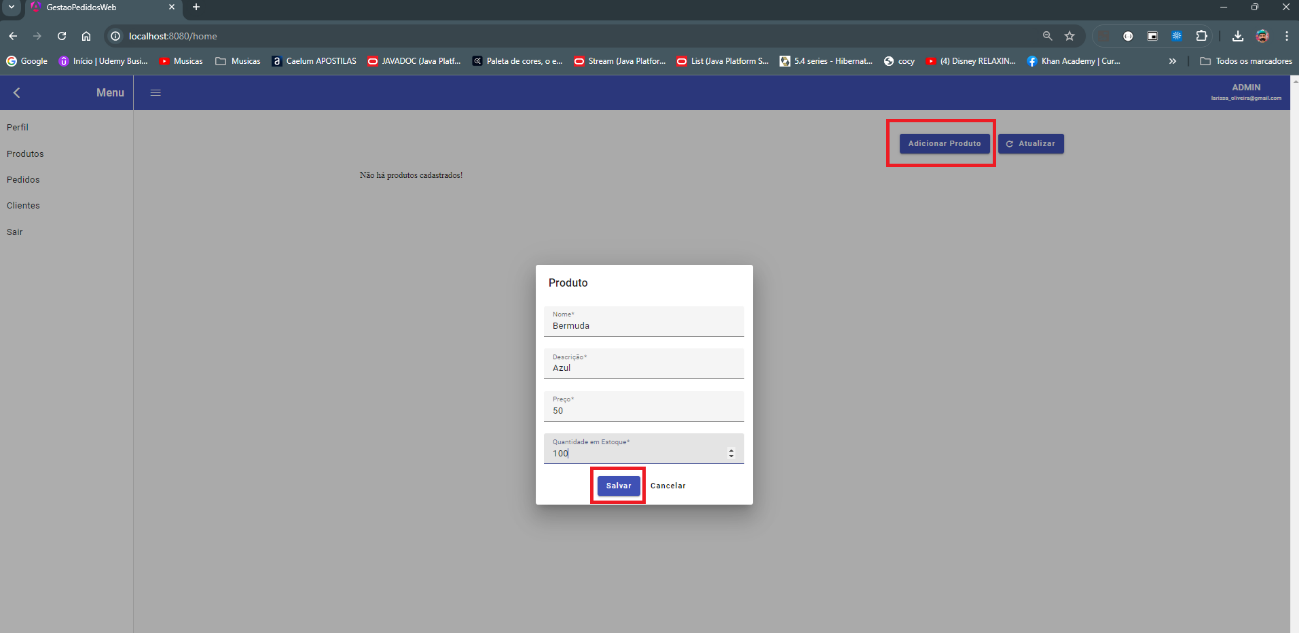
Tela de celular com publicação numa rede social

Descrição gerada automaticamente

1. Efetue o Login por meio do link (<http://localhost:8080/login>) com o usuário ADMIN e click na aba “Produtos” para cadastrar um novo item para ser comercializado.



1. Click no botão “Adicionar Produto” em seguida Preencha o todos os campos solicitados, e click em salvar.

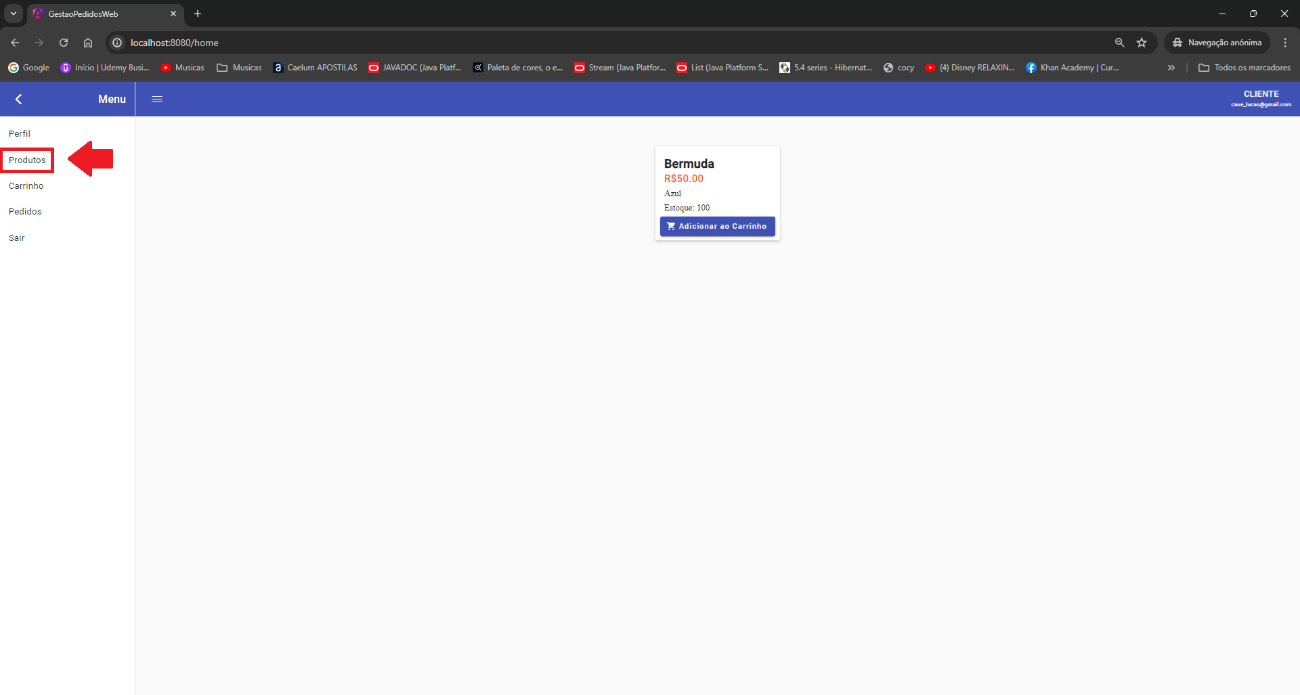


1. Após salvar o registro, o produto será listado no grid conforme a imagem a seguir:

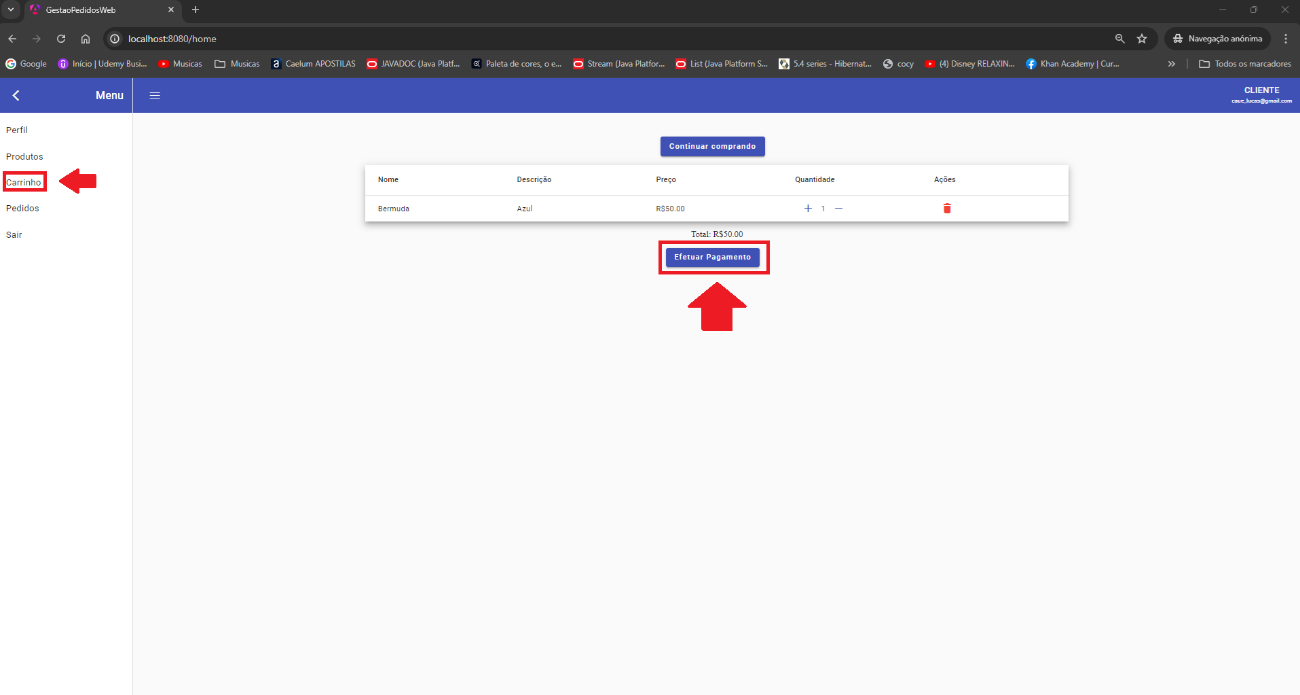
Tela de celular com publicação numa rede social

Descrição gerada automaticamente

1. Todos os produtos listados no passo 5, já estarão hábitos a serem comercializados, portanto serão apresentados aos clientes. Efetue o Login por meio do link (<http://localhost:8080/login>) com o usuário CLIENTE e click na aba “Produtos” para visualizar os produtos disponíveis em seguida click em “Adicionar ao Carrinho”.

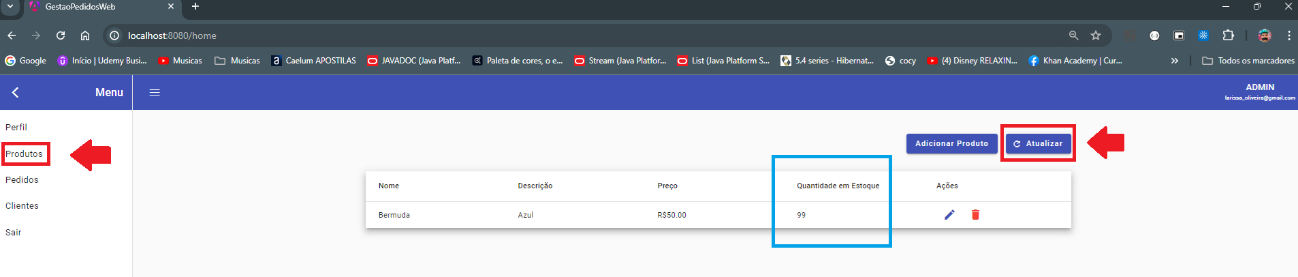


1. Click na Aba “Carrinho” para visualizar o as informações do carrinho de compra, em seguida click em e “Efetuar Pagamento”, conforme a imagem a seguir:



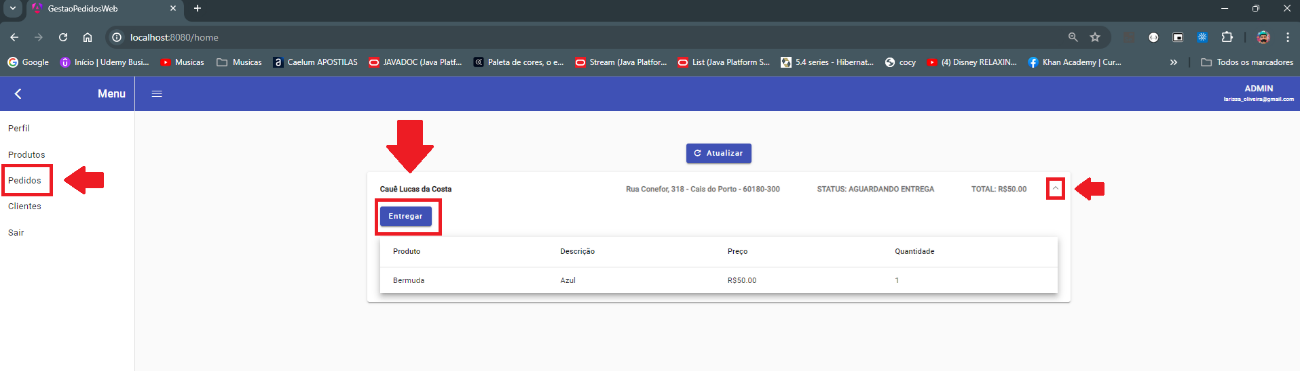
A funcionalidade “Efetuar Pagamento”, é uma simulação do ato de pagar portando o sistema apenas considera que o pedido foi pago e mandara uma requisição para atualizar o estoque de produtos.

1. Considerando que um cliente comprou algum produto conforme as instruções 6 e 7, o administrador do sistema já poderá visualizar o estoque dos produtos. Click “Produtos” em seguida “Atualizar” para que a tabela do produto seja atualizada.



Note que a quantidade de produto decrementou após o pagamento do pedido pelo cliente.

1. Considerando que um cliente comprou algum produto conforme as instruções 6 e 7, o administrador do sistema já poderá visualizar o estoque dos produtos.
   1. Click em “Pedidos” para abrir a listagem de pedidos dos clientes.
   2. Click na seta ao lado do valor do pedido para detalhar o pedido.
   3. Click em “Entregar” para finalizar o fluxo de compro do pedido.



Ao entregar o pedido o sistema atualiza o estado do pedido para entregue.