

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](#)
7. [Guias de Implantação](#)
8. [Documentação da API com Swagger](#)
9. [Fluxo de Uso](#)

Introdução

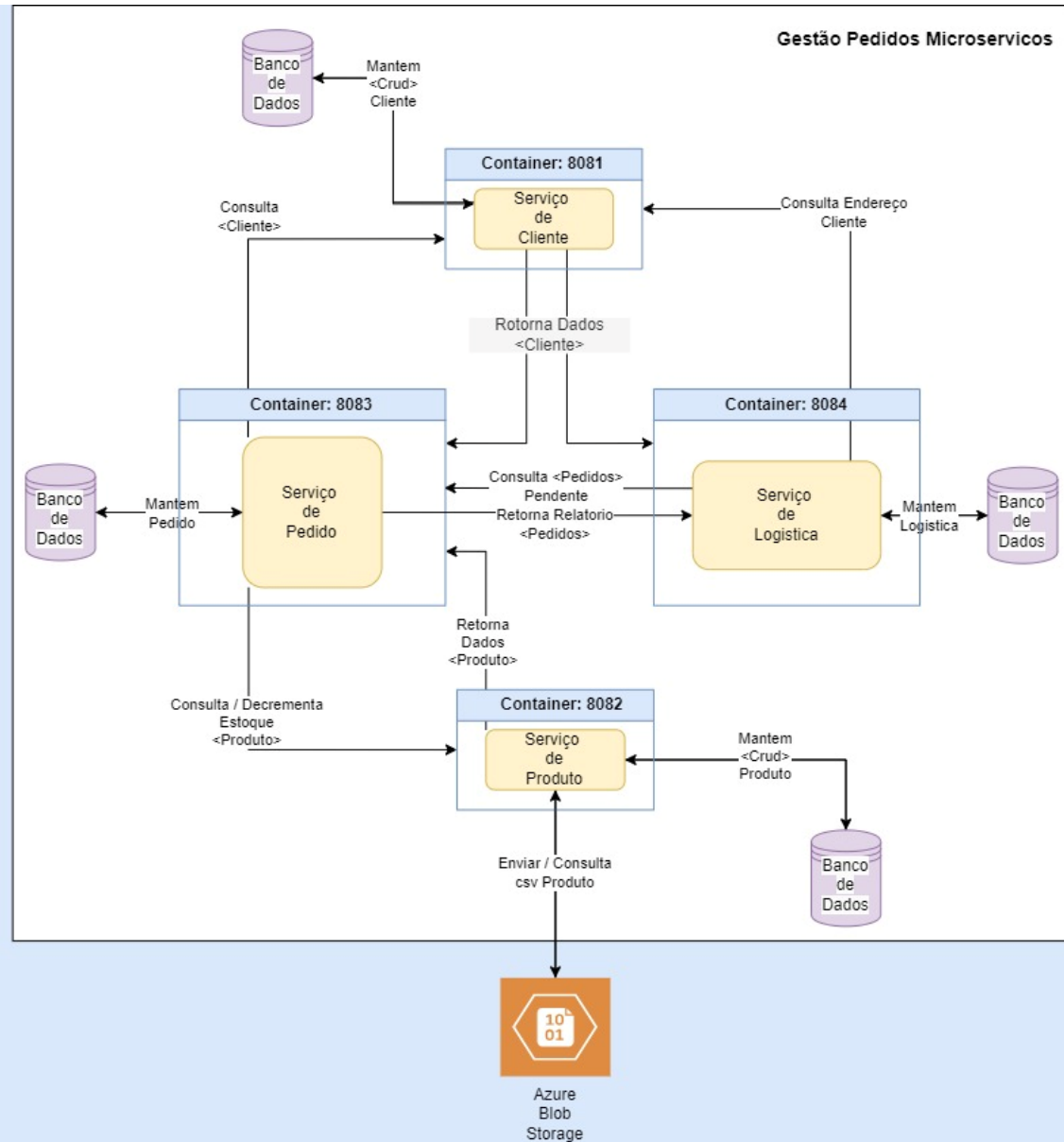
O sistema de gerenciamento de pedidos é altamente eficiente, explorando profundamente a arquitetura de microsserviços utilizando o ecossistema Spring. Este sistema abrange desde a gestão de clientes e produtos até o processamento e entrega de pedidos, enfatizando a autonomia dos serviços, comunicação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

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. Carga de Produtos

- Importação em massa de dados de produtos.
- Upload de arquivo e processamento assíncrono para atualizar o catálogo.
- Há nível de teste pode ser usar esse documento [produtos.csv](#) para testar a carga de produtos via spring batch

4. Gestão de Pedidos

- Processamento de pedidos desde a criação até a conclusão.
- Estados do pedido: aguardando pagamento, pago, aguardando entrega e entregue.

5. Logística de Entrega

- Gerenciamento da logística de entrega.
- Cálculo de rotas, estimativas de tempo e rastreamento em tempo real.
- Processamento de pedidos por sub-setor de CEP, geração de relatórios e atualização de status.

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.
- **EntregaService**: Gerencia a logística e rastreamento de entregas.

Principais Endpoints

ClienteService

- **GET /clientes**

- **Descrição**: Lista todos os clientes.
- **Parâmetros**: Nenhum.
- **Resposta**: JSON com lista de clientes.

- **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.

EntregaService

- **GET /entregas**

- **Descrição:** Lista todas as entregas.
- **Parâmetros:** Nenhum.
- **Resposta:** JSON com lista de entregas.

- **POST /entregas**

- **Descrição:** Atualiza o status da entrega.
- **Parâmetros:** Dados da entrega.
- **Resposta:** Status da entrega atualizado.

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>
- **Logística:** <http://localhost:8084/swagger-ui/index.html>