Despesas Orçamentárias API

A Despesas Orçamentárias API é um projeto desenvolvido em Spring Boot que consome uma API externa para obter uma lista de despesas orçamentárias de Recife. Ele possui a capacidade de persistir essas despesas em um banco de dados PostgreSQL. O projeto utiliza Java 17 como linguagem de programação principal e inclui diversas tecnologias como Spring Data JPA, Hibernate, Swagger, Flyway, Maven e Docker.

Url para teste temporário

– http://vps51498.publiccloud.com.br:8080/swagger-ui/index.html#/

- Como usar:
  - Primeiro deve consumir o endpoint '/v1/serviceconsumer/despesas-orcamentarias-recife' para obter os dados e persistir no banco de dados.
  - Realizar as operações de crud para teste.

Funcionalidades

Consumir uma API externa que retorna uma lista de despesas orçamentárias de Recife. Persistir as despesas obtidas no banco de dados PostgreSQL. Expor endpoints REST para interagir com as despesas, incluindo operações CRUD. Documentação da API gerada automaticamente com Swagger.

Endpoints Dísponiveis

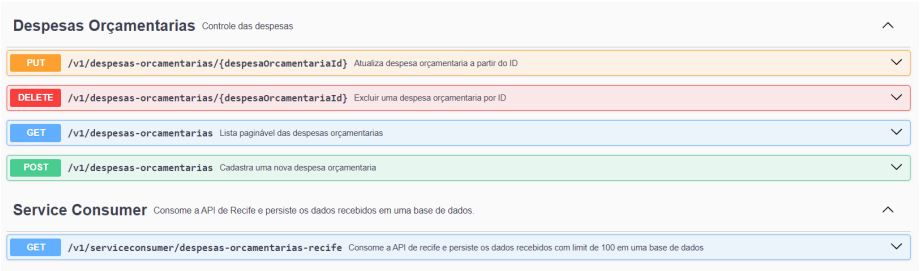

Figure 1: Endpoints

DER

Tecnologias Utilizadas - Spring Boot - Java 17 - PostgreSQL 14 - Spring Data JPA - Hibernate - Swagger - Maven - Docker - Flyway

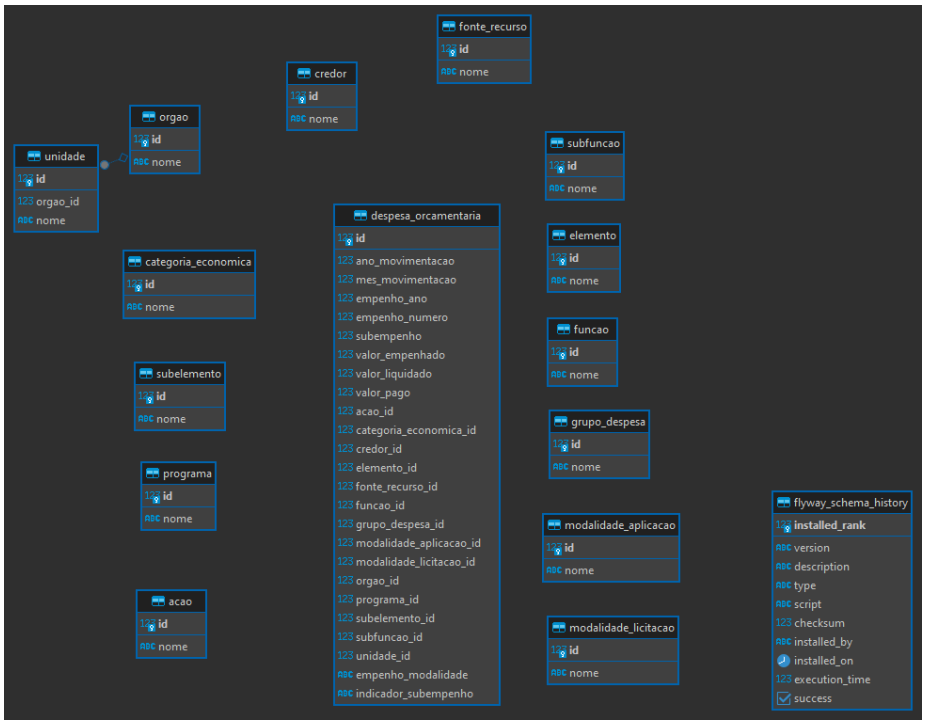

Figure 2: db-despesas-orcamentarias - api - DER

Pré-requisitos

Antes de começar, certifique-se de ter os seguintes requisitos instalados:

- Java 17 SDK
- Docker
- Maven
- PostgreSQL 14 (ou uma instância Docker disponível)

Configuração

Possui duas formas de iniciar o projeto, fazendo a configuração manual ou executar o docker-compose que já inicia e executa o script necessário.

##Configuração do Banco de Dados Manual Antes de executar a aplicação, você precisa configurar o banco de dados. Certifique-se de ter uma instância do PostgreSQL em execução ou use o Docker para iniciar uma instância temporária. Depois de configurar o banco de dados, você precisa configurar as credenciais de acesso no arquivo application.properties.

```
spring.datasource.url=jdbc:postgresql://localhost:5432/nome_do_banco
spring.datasource.username=usuario
spring.datasource.password=senha
```

`Substitua nome_do_banco, usuario e senha pelas suas credenciais de acesso ao banco de dados.`

Executando a Aplicação Para executar a aplicação, siga estas etapas:

- Clone este repositório para o seu ambiente local.
- Navegue até o diretório raiz do projeto.
- Construa o projeto com Maven:

`- mvn clean install`

- Inicie o projeto

`- mvn spring-boot:run`

Inicie a aplicação com Docker Compose:

- Antes de tudo é necessario que a pasta/script esteja no mesmo local do docker-compose.yml.
- Abra o terminal/console a partir do diretorio do arquivo docker-compose.yml.
- Execute:

`- docker compose up`

Acesse a documentação da API Swagger em: http://localhost:8080/swagger-ui/index.html.