



Fulfillment API

// Joao Teixeira Araujo

IT BOARDING

BOOTCAMP



// Descrição

Implementação de uma REST API para gerenciar produtos frescos no MeLi.

Requisitos solicitados

- **Requisito 1:** gerenciar os lotes de produtos no warehouse;
- **Requisito 2:** listar produtos existentes e gerenciar compras;
- **Requisito 3:** listar produtos por lotes;
- **Requisito 4:** verificar stock de um produto em um warehouse;
- **Requisito 5:** listar lotes de acordo com o setor.

IT BOARDING

BOOTCAMP



// Requisito 6



Funcionalidades

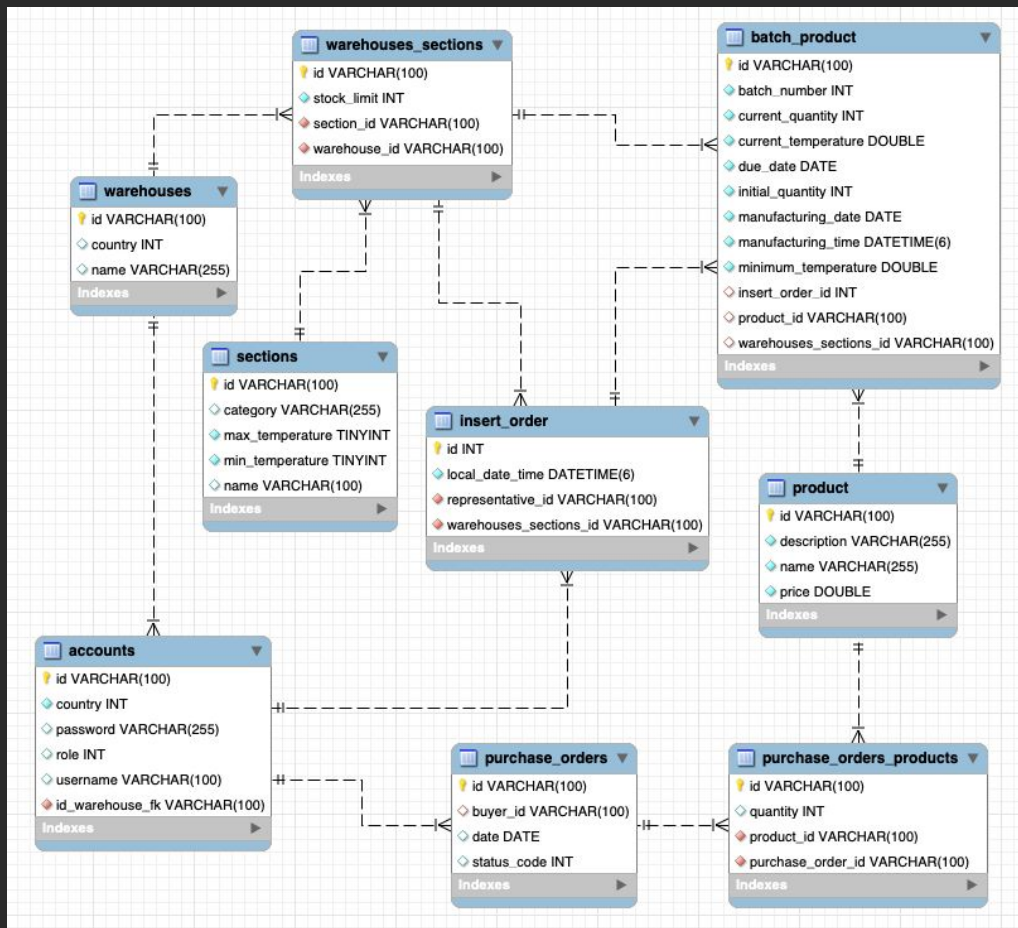
- Utilização da biblioteca BCrypt para encriptar as senhas;
- Criação da role de ADMIN;
- Implementação da funcionalidade de listar usuários por country;
- Implementação da funcionalidade de cadastrar um usuário.

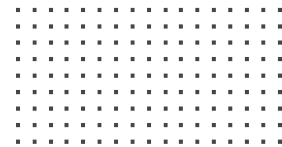


Requisitos para cadastrar usuário

1. Username não pode estar em utilização;
2. Warehouse, country e role devem existir;
3. Warehouse deve possuir o mesmo country do usuário a ser cadastrado;
4. O password deve possuir letras maiúsculas e minúsculas, caracteres especiais, além de possuir de 8 a 20 caracteres.

// Modelagem





// Tecnologías / técnicas utilizadas

- **JPA / Hibernate:** utilização de queries (queries já implementadas e queries nativas) e UUID generator;
- **MySQL:** persistencia dos dados;
- **Spring (data, web e security):** definição dos tipos de dados a serem utilizados no modelo, utilização de serviços e rest controllers além da configuração / liberação de rotas;
- **Stream API:** reduz o tamanho do código e melhora a eficiência em processamento de collections.
- **Postman:** ferramenta para testar requisições;
- **Pre-scripts de validação:** validação automática de usuário no fury (X-Auth-Token) e na API (Authorization);
- **Exceções personalizadas:** criação de um enum contendo todos os códigos dos possíveis erros;
- **JUnit / Mockito:** Coverage de mais de 80% (testes unitários e de integração) ;
- **Lombok:** anotações para evitar a escrita de getters, setters e construtores.

// Problemas durante o projeto



Geral

- Problemas relacionados a merges incorretos que só foram vistos mais à frente;
- `setPropertyNamingStrategy` do `spring-config` estava configurado para `snake case` (os DTO's estavam vindo nulos);
- No `springSecurity` setamos `denyAll()` para todas requests (cada endpoint foi liberado de acordo com a role permitida). Para habilitar o `swagger`, tivemos que liberá-lo através do `permitAll()`.



Fury

1. VPN às vezes estava desligada e não permitia algumas funcionalidades;
2. `ModHeader` ligado impedindo o login no fury;
3. Criação do banco de dados com nome diferente do nome setado na aplicação;
4. O scope web deveria ter o nome de prod;
5. Os secrets utilizados na aplicação deveriam possuir a nomenclatura `"SECRET_nomeDoSecret"`.



Gracias.

IT BOARDING

BOOTCAMP

