

The logo for Luizalabs Magalu is centered on the page. It features the word "luizalabs" in a white, lowercase, sans-serif font. A white circle is positioned to the right of the "s", partially overlapping it. Below "luizalabs" is the word "magalu" in a smaller, white, lowercase, sans-serif font. The entire logo is set against a blue-tinted background of a modern office interior with people and large circular pendant lights.

luizalabs^o
magalu

Teste Backend



Feature

Uma das funcionalidades mais interessantes em um e-commerce é a **Wishlist**, ou a lista de desejos. No e-commerce o cliente pode realizar a busca de produtos, ou pode acessar a tela de detalhes do produto. Em ambas as telas ele pode selecionar os produtos de sua preferência e armazená-los na sua Wishlist. A qualquer momento o cliente pode visualizar sua Wishlist completa, com todos os produtos que ele selecionou em uma única tela.

O que deve ser feito?

O objetivo é que você desenvolva um serviço HTTP resolvendo a funcionalidade de Wishlist do cliente. Esse serviço deve atender os seguintes requisitos:

- **Adicionar** um produto na Wishlist do cliente;
- **Remover** um produto da Wishlist do cliente;
- **Consultar todos** os produtos da Wishlist do cliente;
- **Consultar** se **um** determinado **produto** está na Wishlist do cliente;

Orientações

Imagine que esse serviço fará parte de uma plataforma construída em uma arquitetura baseada em micro-serviços. Portanto **não se preocupe** com a gestão das informações de Produtos e/ou Clientes, coloque sua energia apenas no serviço da Wishlist.

É importante estabelecer alguns limites para proteger o bom funcionamento do ecossistema, dessa forma a Wishlist do cliente deve possuir um **limite máximo de 20 produtos**.

Premissas

| O que esperamos | |
|-----------------------------------|-----------------|
| Linguagem | Java 11 ou > |
| Framework http | Spring Boot |
| Gerenciador de dependência | Maven ou Gradle |
| Banco de dados | Qualquer NoSQL |

+ Dicas

- Testes são excelentes para descrever como o código funciona! Nós gostamos muito de testes, atualmente utilizamos muito BDD.
- Organizar a estrutura do código, classes e pacotes, também é relevante, utilizamos clean architecture em nossos serviços.
- Um ecossistema de vários micro-serviços HTTP é importante estabelecer um padrão de arquitetura. Recomendamos o modelo REST;
- Fique a vontade p/ usar containers, caso faça sentido;