

Requisitos das aplicações:

Nós desejamos que você crie 2 aplicações básicas (microserviços) que comuniquem-se entre si.

O primeiro deles deverá ser um cadastro de usuários, contendo os seguintes recursos:

- Listar, exibir, criar, alterar e excluir usuários

Tabela de usuários `user` deverá conter os campos: id, name, cpf, email, phone_number, created_at, updated_at

E o segundo deverá ser um serviço de pedidos, onde este deverá conter o id do usuário que fez o pedido e se comunicar com o serviço de usuários para retornar as informações do mesmo. Esse serviço deverá ter os seguintes recursos:

- Listar, Listar por usuário, exibir, criar, alterar e excluir.

Tabela de pedidos `order` deverá conter os campos: id, user_id, item_description, item_quantity, item_price, total_value, created_at, updated_at

Lembre-se de fazer a comunicação necessária entre os serviços para garantir a consistência de dados.

Critérios de avaliação

Dê uma atenção especial aos seguintes aspectos:

- Você **DEVE** usar bibliotecas de terceiros, e pode escolher usar um framework, utilizar não vai ser uma penalidade, mas você vai precisar justificar a sua escolha.
- Suas aplicações **DEVEM** executar em containers Docker.
- Suas aplicações **DEVEM** retornar um JSON válido e **DEVEM** conter os recursos citados anteriormente.
- Você **DEVE** escrever um código testável e demonstrar isso escrevendo testes unitários (não é necessário testar as rotas do flask, somente outras funções).
- Testes podem ser rodados localmente, não necessitam serem rodados em docker
- Você **DEVE** seguir as diretrizes de estilo de código limpo.
- Você **NÃO** precisa desenvolver um "frontend" (telas) para esse teste.

Pontos que consideramos um bônus:

- Fazer uso de uma criptografia reversível de dados sensíveis do usuário, como: email, cpf e telefone, antes de persisti-los no banco de dados

- Setup da aplicação em apenas um comando ou um script que facilite esse setup