Nome: Raphael Parro Felizardo

RM 94569

Nome: Gabriel Cardoso Pinheiro Cury Marques

RM 93456

# Instruções iniciais

* A prova será em dupla;
* **A entrega será feita através do link:** https://docs.google.com/forms/d/e/1FAIpQLSeKZrEgSlRdJMdcaV\_3eMKhv6sIFM5jlk8 wrBd6eX3kxUJLaQ/viewform?pli=1
* Não será permitido o uso de qualquer **inteligência artificial**;
* O link será fechado ao final da aula;
* Pode escolher a linguagem que mais se sentir à vontade (c#, Java, Node Js);
* Serão considerados as boas práticas de implementação

# Caso de uso

A companhia Xavier Corporation acaba de adquirir uma grande empresa no ramo têxtil (Têxtil Aarão). Os antigos proprietários utilizam um sistema que possui alguns problemas e, além disso, é responsável por 40% dos gastos mensais da corporação.

O grande nicho da empresa são os tecidos vendidos através de um e-commerce próprio.

A empresa conta também com um setor de recursos humanos, contabilidade, equipe de vendas, os meninos da TI, entre outros setores que eram coordenados pelos próprios donos.

A nova proprietária deseja melhorar o processo, resolver o principal problema que é a dependência de um único sistema que controla tudo, ou seja, quando este sistema cai, a empresa para. Além disso, deseja se aproximar mais dos clientes, para isso, deseja adotar uma estratégia de enviar uma mensagem para os usuários

# Atividade 1 (2 pontos)

**Escolha um serviço do caso de uso**: Por exemplo, "Gerenciamento de Produtos".

**Implementar uma integração completa**: O candidato deve implementar as funcionalidades de:

POST para inserção; GET para consulta; PUT para atualização;

# Atividade 2 (2 pontos)

**Requisitos de Status Code:**

* Para cada operação, implementar o uso correto de status codes: 200, 201, 204, 400, 404, 500, etc.

# Atividade 3 (3 pontos)

Conexão com MongoDB: configurar o MongoDB como banco de dados para armazenar as informações dos serviços escolhidos. A conexão deve ser realizada com o MongoDB em um contêiner Docker, e o banco pode estar hospedado em um servidor local ou em cloud.

# Atividade 4 (1 ponto)

Criar um Dockerfile para conteinerizar o serviço escolhido.

O serviço deve estar funcional dentro de um contêiner Docker, usando MongoDB como banco de dados.

# Atividade 5 (2 pontos)

Enviar:

1. O código da aplicação (API + Dockerfile) dentro de um repositório GIT;
2. Documento Word com prints e descrição dos testes. (incluir no mesmo repositório).

  
  




