

Teste Prático – Desenvolvedor .NET Sênior

Cenário

Sua equipe está desenvolvendo um sistema de integração para consolidar pedidos de múltiplas fontes e permitir que clientes rastreiem suas entregas em tempo real.

O sistema deve expor uma API para cadastrar pedidos, processá-los de forma assíncrona via mensageria e armazenar os dados em um banco relacional. Além disso, a solução deve ter observabilidade e boas práticas de desenvolvimento.

Desafios Técnicos

1. Desenvolvimento de API REST em .NET 8

Crie um serviço em **.NET 8** que exponha os seguintes endpoints:

- **POST /orders** – Recebe um pedido com `Id`, `Cliente`, `Valor` e `DataPedido` e o envia para uma fila do **Kafka (ou Azure Event Hub)**.
- **GET /orders/{id}** – Consulta um pedido no **SQL Server**.

2. Integração com Mensageria (Kafka, Event Hub ou RabbitMQ)

Implemente um **consumidor de mensagens** que leia os pedidos da fila e os armazene no SQL Server.

3. Banco de Dados Relacional e NoSQL

- Utilize **SQL Server** para armazenar pedidos.
- Como diferencial, adicione um cache em **MongoDB** para consultas rápidas.

4. Desenvolvimento de Frontend em React ou Angular

Crie uma interface web simples utilizando **React ou Angular** que permita:

1. **Cadastrar um Pedido** preenchendo um formulário e enviando a requisição `POST /orders`.
2. **Listar Pedidos** consumindo o endpoint `GET /orders/{id}` e exibindo os detalhes na tela.

5. DevOps e Deploy no Azure (Opcional)

Forneça arquivos de `Dockerfile` para rodar as aplicações e um pipelines de CI/CD no Azure DevOps.

6. Observabilidade e Logging (Opcional)

Implemente **logging estruturado** usando **Serilog** e adicione métricas de monitoramento com **Prometheus/Grafana/Loki** ou **OpenTelemetry**.

Entrega do Teste

- Código hospedado no **GitHub** ou **GitLab**.
- Instruções no **README.md** sobre como rodar o projeto (Docker, comandos, dependências).
- A aplicação pode rodar em **localhost:3000** e consumir a API no **localhost:5000**.
- Demonstração dos logs e métricas, caso opte por desenvolver.