

TI Investimentos

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

* Indica uma pergunta obrigatória

Avaliação Técnica - Desenvolvedor Full Stack

1. O código da aplicação pode ser desenvolvido em uma IDE de preferência e disponibilizado no GitHub de forma pública. O link de acesso ao repositório deve ser fornecido no campo de resposta.
2. O objetivo principal de avaliação deste projeto é a lógica e a estrutura aplicada pelo candidato. Portanto, a construção do código deve ser feita de forma simples e clara, enfatizando a compreensão do problema e a solução proposta.
3. Para a realização dos testes unitários no backend, o uso do JUnit é recomendado, mas outras ferramentas também podem ser utilizadas, desde que bem justificadas.
4. Caso o candidato opte por implementar o bônus de utilizar Kafka ou RabbitMQ para enviar mensagens quando um novo produto for adicionado, ele deve explicar detalhadamente a escolha, a implementação e as vantagens que essa adição proporciona ao projeto.
5. O teste sugere algumas tecnologias, e caso o projeto seja criado usando algo fora do sugerido, favor explicar o motivo da escolha.

Desenvolva uma pequena aplicação de listagem de produtos utilizando React para o frontend e Spring Boot para o backend. A aplicação deve exibir uma lista de produtos e permitir adicionar novos produtos à lista.

Operações necessárias

- Listar todos os produtos (GET /produtos).
- Adicionar um novo produto (POST /produtos).
- Para a persistência de dados, você pode utilizar um banco de dados relacional ou NoSQL de sua escolha.



Requisitos

- Utilize JUnit para implementar testes unitários no backend.
- Implemente o frontend com React/Vue/Angular e faça a chamada de API para o backend.
- Crie um arquivo README detalhando como executar o projeto.
- Inclua no README as instruções de como configurar e executar os pipelines de CI/CD no Jenkins para este projeto.
- Crie um arquivo Dockerfile para a aplicação.
- Utilize GIT para versionamento do código e hospede o projeto em um repositório no GitHub.
- No README, explique como você aplicaria o GITFlow neste projeto.
- Uso de boas práticas (**Design patterns**)
- Uso de uma arquitetura clean

Bônus (não obrigatório, mas será considerado um diferencial)

- Utilize Kafka ou RabbitMQ para enviar uma mensagem cada vez que um novo produto for adicionado.
- Inclua um arquivo para configurar a infraestrutura necessária para a aplicação utilizando Terraform.

Link para o GITHUB. *

Sua resposta

[Voltar](#)

[Próxima](#)

[Limpar formulário](#)

Nunca envie senhas pelo Formulários Google.

Este formulário foi criado em CONNINF. [Denunciar abuso](#)

Google Formulários



