

Take-home Assignment – Desenvolvimento Fullstack com Next.js e NestJS

Objetivo

O candidato deve **projetar e implementar** uma aplicação Fullstack utilizando **Next.js** para o frontend e **NestJS** para o backend, garantindo **boas práticas, escalabilidade e caching eficiente**.

O foco será a implementação de um **CRUD de usuários**, integrando **cache com Redis** e garantindo a persistência dos dados com **MySQL ou PostgreSQL**.

O que precisa ser entregue?

Implementação completa do sistema, incluindo:

Frontend (Next.js)

- Tela para **listagem de usuários**, consumindo a API do backend.
- Tela para **cadastro e edição de usuários**.
- Integração com a API usando **React Query** para otimizar chamadas e caching no client.

Backend (NestJS)

- **CRUD de usuários** com os seguintes endpoints:
 - **POST /users** → Criar usuário
 - **GET /users** → Listar todos os usuários
 - **GET /users/:id** → Buscar um usuário específico
 - **PUT /users/:id** → Atualizar um usuário
 - **DELETE /users/:id** → Remover um usuário
- **Banco de dados relacional (MySQL ou PostgreSQL)** usando **TypeORM** ou **Prisma**.
- **Cache com Redis** para otimizar as operações onde necessário.
- **Testes automatizados com Jest** para validar os casos de uso.

Script de criação do banco

- Criar um script SQL ou um comando do ORM para inicializar as tabelas necessárias.
-

Observação sobre o Layout

O design e a estilização da aplicação **não serão levados em consideração** na avaliação.
O foco está na **organização do código, estrutura do projeto e funcionalidade**.
O importante é que o sistema **funcione corretamente** e esteja **bem estruturado**.

Critérios de Avaliação

- ✓ Organização e estrutura do código
 - ✓ Uso correto de **React Query** para otimizar chamadas de API
 - ✓ Implementação eficiente de **cache com Redis**
 - ✓ Boas práticas na arquitetura do **NestJS**
 - ✓ Implementação dos **testes automatizados**
 - ✓ Documentação mínima explicando como rodar o projeto
-

Bônus (Opcional)

Os seguintes pontos não são obrigatórios, mas contarão como diferencial na avaliação:

- Implementação de **autenticação** (JWT, OAuth, etc.).
 - **Logs estruturados** utilizando **Winston** ou outra ferramenta.
 - **Configuração de variáveis de ambiente** com **dotenv**.
-

Entrega

O candidato deve disponibilizar o código em um **repositório público no GitHub** e fornecer instruções para executar o projeto no **README**.

Boa sorte! 