

Desafio Técnico – Estágio em Desenvolvimento Back-End (Node.js)

Introdução

Olá! Este é o desafio prático do nosso processo seletivo.

Você deverá desenvolver a aplicação "**BookTrack**", um sistema de gerenciamento de leituras que permite a um usuário registrar e acompanhar livros que deseja ler, está lendo ou já leu. Seu foco deverá estar na API back-end: estruturação, rotas, regras de negócio, modelo de dados e persistência.

O objetivo é avaliar seu domínio de Node.js, SQL, Git e boas práticas de desenvolvimento orientado a objetos.

Objetivos do Desafio

- Avaliar o domínio prático de Node.js, SQL e Git.
 - Verificar a aplicação de princípios de organização e boas práticas no back-end.
 - Testar a habilidade de traduzir regras de negócio em código funcional.
 - Observar o uso correto de controle de versão com Git.
 - Identificar autonomia na entrega de diferenciais.
-

Descrição Geral do Projeto

Nome do Projeto: BookTrack

Resumo: API RESTful para gerenciamento de usuários e seus livros.

Tecnologias obrigatórias:

- Node.js
 - Banco de dados relacional
 - Git para versionamento
-

Regras de Negócio

Entidades:

Usuário

- **id** (PK)
- **nome** – obrigatório, mínimo 3 caracteres
- **email** – obrigatório, único, com formato válido

Livro

- **id** (PK)
- **titulo** – obrigatório, entre 3 e 100 caracteres
- **autor** – opcional
- **status** – obrigatório: "Quero Ler", "Lendo", "Lido"
- **avaliacao** – opcional, de 1 a 5, só se status = "Lido"
- **data_conclusao** – preenchida automaticamente quando status = "Lido"
- Cada livro pertence a **um único usuário**

Regras:

- Avaliação só pode ser registrada se o status for "Lido"
 - Um usuário pode ter quantos livros quiser
 - Apenas o dono pode editar ou excluir seus livros
 - Livros com status "Lido" não podem ser editados, apenas excluídos
-

Features Obrigatórias (Back-End)

1. Modelagem e criação das tabelas e relações
 2. API RESTful para:
 - Cadastro, listagem e exclusão de usuários
 - Cadastro, edição, listagem e exclusão de livros
 3. Validações das regras de negócio descritas
 4. Controle de versão com Git (uma branch por feature + commits claros)
 5. Documentação no README explicando como rodar o projeto localmente (ex: via Postman)
-

Features Não Obrigatórias (Diferenciais)

1. Autenticação com login e senha (ex: JWT, bcrypt)
 2. Exportação da lista de livros em CSV ou JSON
 3. Testes automatizados (ex: Jest, Mocha)
 4. Containerização com Docker
 5. Interface web básica com HTML + JS para consumo da API
 6. Deploy (ex: Render, Railway, Heroku)
-

Requisitos de Estrutura do Projeto

- Organização por camadas (routes, controllers, services, models)
- README com:

- Instruções para execução
 - Exemplos de rotas (via Postman ou Insomnia)
 - Descrição da estrutura do projeto
-

Regras Importantes

- As features não precisam ser implementadas na ordem apresentada. Organize da forma que preferir.
 - Leia com atenção todas as especificações deste desafio antes de começar.
 - Em caso de dúvida, entre em contato.
 - É permitido utilizar a internet para consultar **documentações técnicas oficiais**.
 - Utilize o editor de texto, ferramentas e bibliotecas que preferir.
 - **É terminantemente proibido** consultar outras pessoas ou utilizar trechos de código prontos/copiados da internet.
 - **Não inclua o nome da empresa ou o conteúdo desta avaliação** no README, código ou qualquer parte pública do repositório (ex: GitHub).
 - O front-end e o design são opcionais e considerados diferenciais apenas se forem implementados como interface de consumo da API.
-

Critérios de Avaliação

- Organização e clareza do código
 - Aplicação correta das regras de negócio
 - Implementação funcional das features obrigatórias
 - Estrutura do repositório e uso do Git
 - Clareza na documentação
 - Qualidade técnica geral
 - Implementação dos itens diferenciais
-

Entregas

- Link do repositório no GitHub com todo o código fonte e README.
- Vídeo demonstrando os principais trechos do código, a forma de organização no git e o funcionamento da aplicação.
- Envie os links de entrega como resposta ao e-mail de recebimento do desafio.