



Desafio Técnico – Coletor de Dados de Filmes

Boas-vindas à etapa de teste prático do processo seletivo da dti digital, não se preocupe, o seu resultado não será divulgado e nem utilizado pela dti para nenhum outro fim além do processo seletivo. Desejamos a você boa sorte!

Objetivo

Desenvolver uma aplicação web full stack onde o usuário possa buscar filmes utilizando uma API externa, favoritar seus filmes preferidos, registrar reviews pessoais e coletar/analisar dados básicos sobre suas buscas e preferências cinematográficas.

O sistema tem como foco o controle pessoal de filmes assistidos com elementos de organização e análise de dados, funcionando como uma ferramenta pessoal que coleta insights sobre seus hábitos de busca.

NOTA: Não é necessário login/autenticação - mantenha a aplicação simples e funcional, partindo da premissa de que a utilização é individual.

Integração com API de Filmes

A aplicação deve consumir dados da OMDb API (The Open Movie Database).

Como gerar sua API Key da OMDb

1. Acesse: <https://www.omdbapi.com/apikey.aspx>
2. Preencha os dados:
 - a. First Name (Nome)
 - b. Last Name (Sobrenome)
 - c. Email válido
 - d. Motivo: study for developer
 - e. Plano: selecione o **Free Account** (1000 requisições/dia)
3. Clique em Submit

4. Verifique o e-mail usado e ative sua chave

Exemplo de uso da API: <https://www.omdbapi.com/?apikey=SUACHAVE&t=Matrix>

Funcionalidades do Sistema

Funcionalidades básicas:

- Busca de filmes por título
- Tela de detalhes do filme
- Favoritar/remover favorito
- Avaliar filmes (comentário + nota [0 a 5])
- Editar/deletar avaliação
- Lista de favoritos com avaliações

Funcionalidades de Dados

- Histórico de buscas realizadas
- Dashboard “Meus dados” com estatísticas pessoais:
 - Total de filmes buscados
 - Gênero mais pesquisado (com gráfico comparativo)
 - Década preferida
 - Nota média das avaliações
 - Gráfico simples dos anos dos filmes buscados

Requisitos Técnicos

Frontend (React + TypeScript)

- Desenvolvido com React + TypeScript
- Interface responsiva (funciona em celular e desktop)
- Permitido utilização de bibliotecas de componentes (Ex.: material UI)
- **Páginas básicas:**
 - Tela de busca de filmes
 - Tela de detalhes de filmes
 - Tela de favoritos
 - Tela “Meus dados” com dashboard e estatísticas
- Uso de gráficos para comparativos entre estatísticas (permitido uso de bibliotecas)
- Código bem organizado

Backend (JavaScript/TypeScript ou Python)

- Backend separado: Node.js + Express OU FastAPI/Flask (Python)
- Aplicação de conceitos RESTful
- **Endpoints básicos:**
 - Favoritos (CRUD)
 - Avaliações (CRUD)
 - Estatísticas (CRUD)
 - Filmes (GET)
- **Persistência de dados**
 - Favoritos e avaliações
 - Histórico de buscas
 - Uso de arquivo JSON ou banco simples
 - Código bem organizado

NOTA: Se atente à tecnologia escolhida para persistência de dados. Certifique-se de incluir no seu teste as orientações necessárias para uso da tecnologia de banco de dados escolhida, garantindo que o sistema seja funcional na máquina dos avaliadores.

Fluxo do Usuário

- 1. Buscar e coletar dados:**
 - a. Usuário busca um filme por nome
 - b. Sistema consulta OMDb API e salva dados no histórico automaticamente
 - c. Exibe resultados na tela
- 2. Interagir com o filme:**
 - a. Visualizar detalhes completos
 - b. Favoritar filmes
 - c. Deixar avaliação (nota + comentário)
- 3. Analisar dados coletados**
 - a. Acessar página "Meus dados"
 - b. Ver estatísticas pessoais em dashboard
- 4. Visualizar favoritos**
 - a. Acessar lista de filmes favoritados e suas avaliações
 - b. Visualizar detalhes completos do filme

Critérios de Avaliação

Funcionalidade e entrega

- Sistema básico funciona (busca, favoritos, avaliação e dashboard)
- Coleta automática de dados funciona
- Dashboard apresenta estatísticas corretas
- Edição e exclusão de avaliações funcional

Qualidade de código

- Separação clara de responsabilidades
- Nomes de variáveis e funções claros e coerentes
- Tratamento de dados bem implementado
- Uso correto de TypeScript no frontend
- Manipulação de dados/arquivos no backend
- Separação clara de domínios da aplicação em pastas

Interface e UX:

- Interface clara e responsiva
- Dashboard é intuitivo e fácil de entender
- Visualização de dados apresenta informações úteis
- Uso de gráficos coerentes para a informação apresentada

Organização e Documentação

- Estrutura de pastas organizada
- README com instruções claras e respostas às perguntas realizadas
- Explicações das decisões sobre estruturas de dados

Bônus

* Estes requisitos não são obrigatórios, mas podem te destacar

- Validação nos campos de entrada
- Tratamento correto de erros da API Externa
- Utilização de CSS/SCSS modules ou Styled Components

Entrega

O que deve ser entregue:

- Código funcional
- README completo com:
 - Instruções claras para execução local
 - Premissas assumidas
 - Decisões de projeto sobre estrutura e coleta de dados
 - Respostas para as perguntas:
 - Qual foi o maior desafio técnico?
 - Como você organizou e estruturou os dados coletados?
 - Que insights interessantes o dashboard poderia oferecer?
 - O que faria diferente com mais tempo?
 - Quais ferramentas/aceleradores de desenvolvimento utilizou? (ex.: GPT, Cursor, etc)

Como deve ser entregue:

- A entrega será feita via repositório **privado** no GitHub compartilhado com:
 - henrique.machado@dtidigital.com.br
 - ulisses.brandao@dtidigital.com.br
 - round@dtidigital.com.br
- A formalização da entrega deve ser mediante envio de email para clarice.ferreira@dtidigital.com.br com link do repositório.

Sobre a desclassificação:

- I. A pessoa candidata que realizar o envio do teste via repositório público estará sujeita à desclassificação do processo seletivo;
- II. A pessoa candidata que realizar alterações nos arquivos do repositório após formalização da entrega estará sujeita a desclassificação do processo seletivo.

ATENÇÃO

É necessário que você confira com sua universidade se você está apto(a) para fazer estágio de 6h antes de aceitar participar do processo

Resultado do teste prático

Após a entrega, pedimos um prazo de 7 dias úteis para correção e envio do feedback ou comunicação sobre os próximos passos do processo seletivo.

Boa sorte! 🚀