

Universidade Federal de Roraima

Disciplina: DCC704 - ARQUITETURA E TECNOLOGIAS DE SISTEMAS WEB

Docente: Thais Oliveira

Discente: Shelly da Costa Leal Matrícula: 2020001671

Relatório do Projeto Shell Movies

1. Introdução ao Projeto e Sua Motivação

O **Shell Movies** é um sistema de recomendação e gerenciamento de filmes, inspirado na experiência de usuário oferecida por plataformas de streaming como HBO Max e Netflix. O principal objetivo do projeto é fornecer uma interface intuitiva e organizada para exibição de informações sobre filmes, permitindo que usuários pesquisem, favoritem e compartilhem filmes de seu interesse.

A motivação para o desenvolvimento do projeto surgiu da necessidade de um sistema que combine **visual atraente, recomendações inteligentes e integração de informações detalhadas dos filmes**. Além disso, foi implementado um **sistema de recomendação híbrido**, que combina técnicas baseadas em conteúdo e popularidade, visando melhorar a experiência do usuário.

2. Arquitetura do Sistema e Decisões Tomadas

O sistema foi desenvolvido utilizando tecnologias modernas e seguindo boas práticas de arquitetura de software.

2.1 Tecnologias Utilizadas

- **Front-end:** React.js com Vite
- **Back-end:** Node.js com Express.js
- **Banco de Dados:** MongoDB (para armazenar favoritos e históricos de usuários)
- **API de Filmes:** The Movie Database (TMDb)
- **Hospedagem do Código:** GitHub

2.2 Estrutura do Projeto

A estrutura do projeto foi organizada da seguinte forma:

movies_lib-main/

| — src/

| | — components/ # Componentes reutilizáveis (Navbar, MovieCard, MovieCarousel, etc.)

| | — pages/ # Páginas principais (Home, Movie, Search)

| | — assets/ # Ícones e imagens estáticas

| | — styles/ # Arquivos CSS

| | — App.js # Configuração principal da aplicação

| | — main.js # Ponto de entrada do React

| | — routes.js # Configuração das rotas

| — backend/ # Backend para autenticação e favoritos

| — package.json # Dependências do projeto

| — vite.config.js # Configuração do Vite

| — .env # Configuração de variáveis de ambiente (API Keys)

2.3 Decisões Arquiteturais

- **Frameworks modernos:** O React foi escolhido pelo seu ecossistema robusto e suporte à componentização, facilitando a reutilização de código.
- **Consumo de API externa:** O sistema usa a API do TMDb para obter dados atualizados sobre filmes.
- **Design responsivo:** A interface foi projetada para ser **adaptável a diferentes tamanhos de tela**.
- **Sistema de recomendação híbrido:** Implementado para melhorar sugestões aos usuários.

3. Explicação do Sistema de Recomendação Implementado

O sistema de recomendação do **Shell Movies** é híbrido e combina duas abordagens principais:

3.1 Baseado em Popularidade

- Exibe os filmes **mais populares no momento**, usando dados da API do TMDb.

3.2 Baseado em Conteúdo

- Utiliza os **gêneros dos filmes** como principal critério de recomendação.

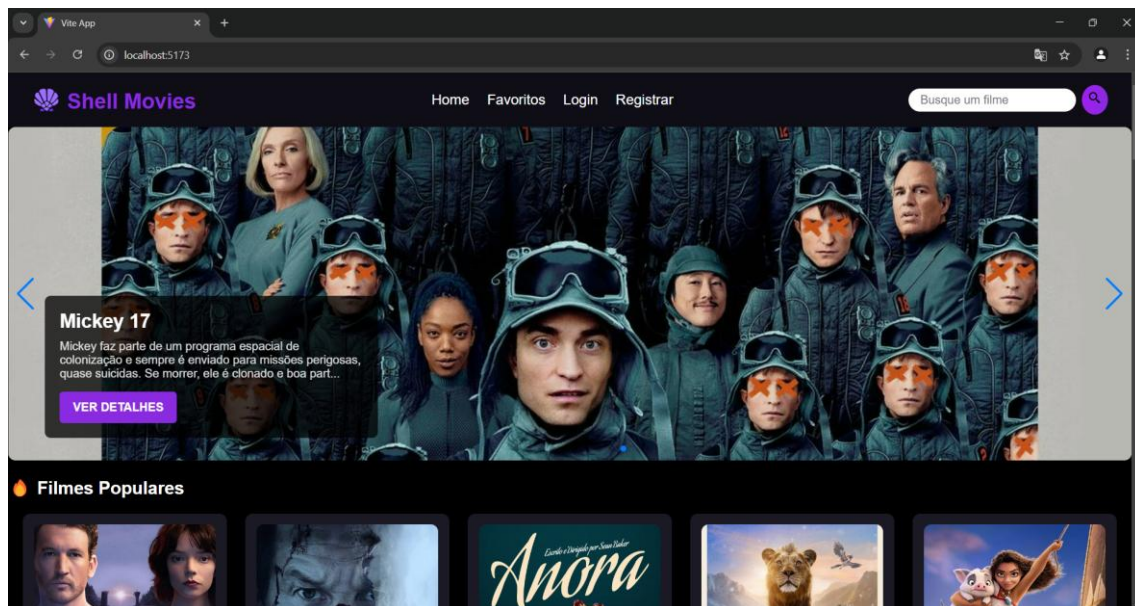
3.3 Algoritmo Utilizado

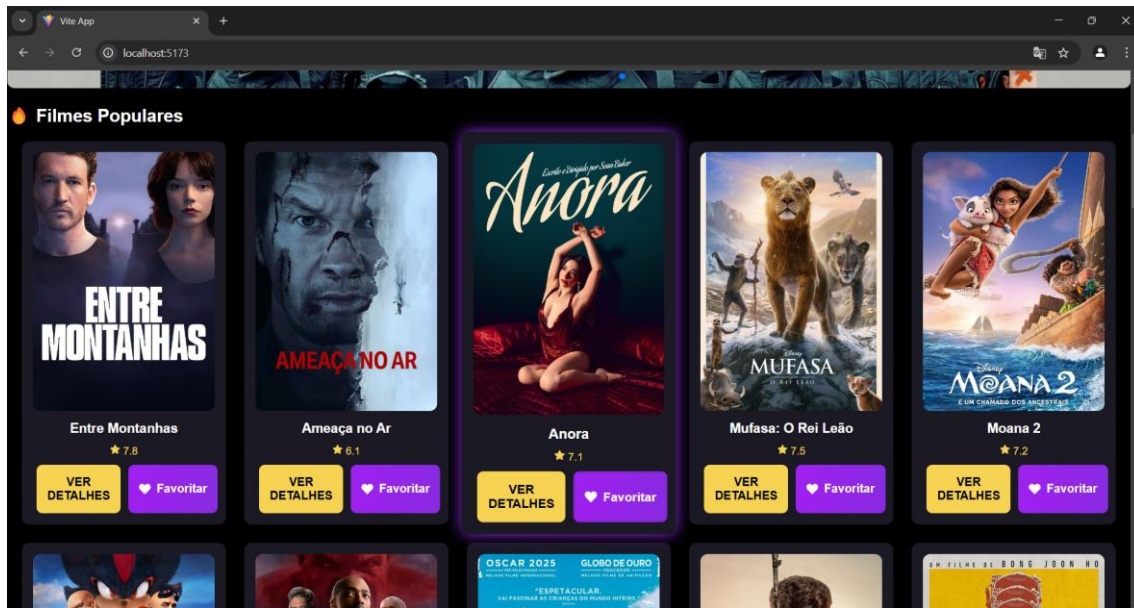
1. O usuário acessa um filme e interage (clica em "Favoritar").
2. O sistema analisa os gêneros do filme e cruza com outros filmes de mesma categoria.
3. O sistema também sugere filmes populares como recomendação geral.

4. Capturas de Tela da Interface e Funcionamento

4.1 Página Inicial

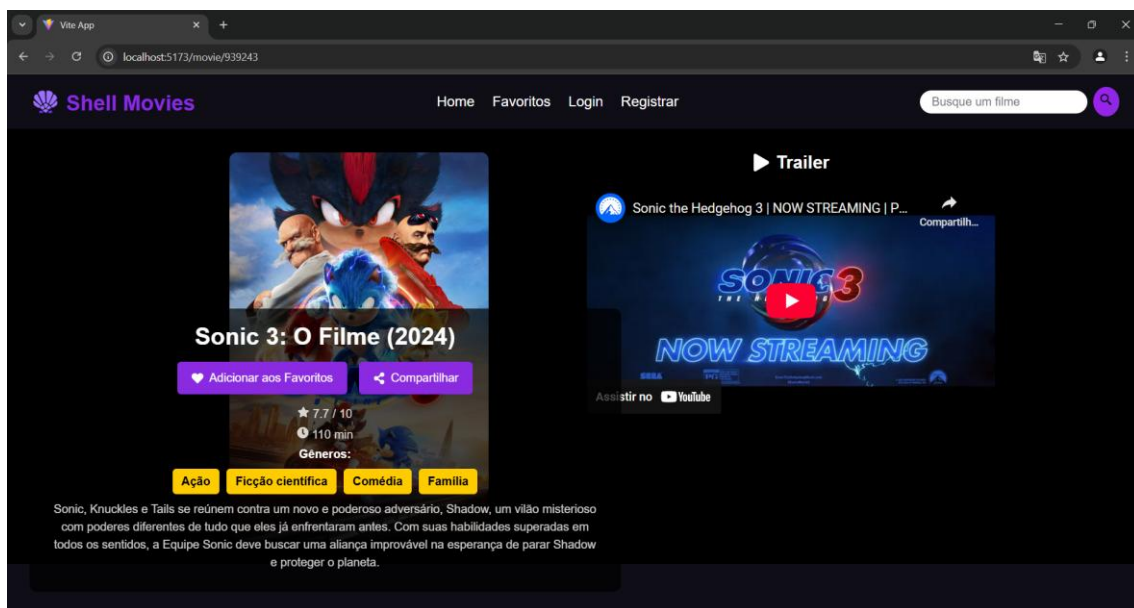
Exibe os filmes populares, um carrossel de destaques e um campo de busca.





4.2 Página de Detalhes do Filme

Mostra a imagem do filme, trailer e informações organizadas.



5. Código-Fonte Documentado e Repositório no GitHub

O código-fonte do projeto está disponível no **GitHub** e foi documentado para facilitar a compreensão e manutenção.

Repositório: [GitHub - Shell Movies](#)

Conclusão

O **Shell Movies** entrega uma experiência fluida e organizada para os usuários, com um sistema de recomendação eficiente e um layout responsivo inspirado em grandes plataformas de streaming.

O projeto pode ser expandido com:

- **Autenticação de usuários** (para salvar históricos e preferências).
- **Sugestões baseadas no histórico do usuário.**
- **Avaliações e comentários dos usuários.**

Esse sistema demonstra o uso prático de **React.js**, **APIs externas** e **sistemas de recomendação**, combinando tecnologia e experiência do usuário.