



EDS – GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

Matéria

PROJETO DE BLOCO - DESENVOLVIMENTO FRONT-END COM FRAMEWORKS

Francisco Benjamim

Professor

Aluno: **Uendel Ives de Araujo**

27/09/2024

1 - Definição de Projeto:

O projeto escolhido envolve o desenvolvimento de um servidor de streaming inspirado em plataformas como Netflix ou Paramount+, utilizando a API do The Movie Database (TMDb). O objetivo é criar uma plataforma onde os usuários possam assistir a filmes, séries e documentários, além de interagir com o conteúdo por meio de funcionalidades como avaliação e recomendação de títulos. Outras APIs e componentes de back-end poderão ser integrados para complementar o sistema, como sistemas de autenticação de usuários e gerenciamento de perfis.

2 - Definição de Escopo e Objetivos:

Objetivo Principal:

Criar uma plataforma de streaming que permita aos usuários explorar e assistir conteúdos audiovisuais sob demanda, com uma interface moderna, intuitiva e responsiva.

Funcionalidades Específicas:

- Catálogo de Conteúdos: Apresentação de filmes, séries e documentários organizados por categoria, popularidade e lançamentos.
- Sistema de Pesquisa: Busca por filmes e séries com filtros por título, gênero e ano de lançamento.
- Perfis de Usuários: Criação e gerenciamento de perfis, com listas personalizadas de favoritos.
- Recomendações Personalizadas: Sugerir conteúdos com base no histórico de visualização e avaliações.
- Avaliações e Comentários: Usuários podem avaliar e comentar sobre os filmes e séries assistidos.

- Streaming de Vídeo: Integração com um servidor de streaming para reprodução dos conteúdos.
- Responsividade: Design adaptado para diferentes dispositivos (desktop, tablets, mobile).
- Funcionalidades Mobile: Implementação de controle de gestos para dispositivos móveis, como deslizar para avançar ou retroceder o vídeo e tocar para pausar.

3 - Gestão Ágil do Projeto:

Organização com Agile Scrum:

O projeto será gerido utilizando a metodologia Agile Scrum, que divide o desenvolvimento em ciclos iterativos chamados sprints. A cada sprint, um conjunto de funcionalidades completas será desenvolvido e entregue.

Sprints:

Cada sprint terá a duração de no máximo 3 semanas e contará com os seguintes eventos do Scrum:

- Sprint Planning: Planejamento do sprint.
- Daily Scrum: Reuniões diárias para acompanhamento do progresso.
- Sprint Review: Revisão ao final de cada sprint para demonstrar o que foi desenvolvido.
- Sprint Retrospective: Reflexão sobre o que funcionou bem e o que pode ser melhorado.

Cronograma de Sprints e Eventos:

Atividade	Data de Início	Data de Término
Sprint 1: Configuração do ambiente, integração inicial com API TMDb e layout básico	14/09/2024	30/09/2024
Sprint 2: Implementação de catálogo de conteúdos, pesquisa e perfis de usuários	01/10/2024	22/10/2024
Sprint 3: Recomendações personalizadas, avaliações e comentários	23/11/2024	06/11/2024
Sprint 4: Integração de streaming e funcionalidades mobile	07/11/2024	28/11/2024
Sprint 5: Testes finais e ajustes	29/11/2024	18/12/2024

4 - Artefatos, Papéis e Eventos do Scrum:

Artefatos:

- **Product Backlog:** Lista de funcionalidades que serão implementadas ao longo do projeto, priorizadas com base nas necessidades do produto.
- **Sprint Backlog:** Cartões de histórias de usuário que serão selecionados do Product Backlog para cada sprint.
- **Incremento:** Versão funcional da aplicação após cada sprint, com as funcionalidades completas e testadas.

Papéis:

- **Product Owner:** Define e prioriza as funcionalidades que compõem o Product Backlog.

- Scrum Master: Facilita o processo de desenvolvimento, remove impedimentos e garante que a equipe siga a metodologia Scrum.
- Equipe de Desenvolvimento: Implementa as funcionalidades e trabalha diretamente nos cartões de histórias de usuário definidos para cada sprint.

Eventos:

- Sprint Planning: Planejamento do que será desenvolvido em cada sprint com base no Product Backlog.
- Daily Scrum: Reuniões diárias para revisão do progresso e ajustes.
- Sprint Review: Demonstração das funcionalidades completas ao final de cada sprint.
- Sprint Retrospective: Reflexão sobre o trabalho desenvolvido e melhorias para o próximo sprint.

5 - Criação de Histórias de Usuário:

As histórias de usuário serão criadas para capturar as funcionalidades da plataforma, focando tanto em aspectos funcionais quanto não funcionais, como responsividade e interatividade com gestos.

História de Usuário 1 (Funcional):

Título: Pesquisa Avançada por Conteúdos

- **Como** um usuário da plataforma
- **Eu quero** realizar buscas avançadas por filmes e séries utilizando filtros como gênero, ano de lançamento e avaliação
- **Para** encontrar conteúdos que correspondam exatamente aos meus interesses.

Critérios de Aceitação:

- O sistema deve permitir a seleção de múltiplos filtros simultaneamente.

- A interface de busca deve ser intuitiva e responsiva.
- Os resultados devem ser exibidos em ordem de relevância.

História de Usuário 2 (Funcional):

Título: Visualização Detalhada de Conteúdos

- **Como** um usuário da plataforma,
- **Eu quero** acessar uma página com os detalhes completos de um filme ou série,
- **Para** conhecer mais informações sobre o conteúdo antes de assisti-lo.

Critérios de Aceitação:

- A página de detalhes deve exibir a sinopse, elenco, data de lançamento, e avaliações.
- O usuário pode adicionar o conteúdo à lista de favoritos ou avaliá-lo.

História de Usuário 3 (Funcional):

Título: Sistema de Autenticação de Usuários

- **Como** um novo usuário,
- **Eu quero** criar uma conta e fazer login,
- **Para** poder salvar meus conteúdos favoritos e personalizar minha experiência.

Critérios de Aceitação:

- O sistema deve permitir a criação de conta com e-mail e senha.
- O login deve ser realizado com as credenciais válidas.
- Após o login, o usuário pode acessar seu perfil e personalizar listas de favoritos.

História de Usuário 4 (Funcional):

Título: Marcar Conteúdos como Assistidos

- **Como** um usuário da plataforma,
- **Eu quero** marcar filmes e séries como assistidos,
- **Para** manter o controle do que já assisti e receber recomendações mais precisas.

Critérios de Aceitação:

- O sistema deve permitir ao usuário marcar conteúdos como assistidos.
- O conteúdo assistido não deve aparecer em recomendações futuras.

História de Usuário 5 (Funcional):

Título: Sistema de Favoritos

- **Como** um usuário da plataforma,
- **Eu quero** adicionar filmes e séries à minha lista de favoritos,
- **Para** poder acessá-los facilmente no futuro.

Critérios de Aceitação:

- O usuário pode adicionar/remover conteúdos da sua lista de favoritos.
- A lista de favoritos deve ser acessível no perfil do usuário.

História de Usuário 6 (Funcional):

Título: Controle de Reprodução via Gestos em Dispositivos Móveis

- **Como** um usuário em dispositivo móvel,
- **Eu quero** controlar a reprodução dos vídeos utilizando gestos intuitivos,

- **Para** ter uma experiência de uso mais natural e eficiente.

Critérios de Aceitação:

- Deslizar para a esquerda/direita avança ou retrocede o vídeo em 10 segundos.
- Tocar duas vezes na tela alterna entre play e pause.
- Pinçar para ampliar ou reduzir o tamanho do vídeo (modo tela cheia).

História de Usuário 7 (Não Funcional):

Título: Melhorar Desempenho em Dispositivos Mobile

- **Como** desenvolvedor,
- **Eu quero** otimizar a aplicação para dispositivos móveis,
- **Para** garantir que a experiência seja rápida e fluida em smartphones.

Critérios de Aceitação:

- A interface deve carregar rapidamente em redes móveis.
- O design deve se adaptar a diferentes tamanhos de tela sem problemas de desempenho.

História de Usuário 8 (Funcional):

Título: Modo Escuro

- **Como** um usuário da plataforma,
- **Eu quero** habilitar um modo escuro,
- **Para** reduzir o cansaço visual ao assistir conteúdo à noite.

Critérios de Aceitação:

- O sistema deve permitir alternar entre o modo claro e o modo escuro.
- O modo escuro deve ser aplicado a todas as páginas da aplicação.

6 - Planejamento do Projeto de Desenvolvimento de Front-end:

Os seguintes passos serão seguidos para garantir um desenvolvimento eficiente do front-end:

Passos Iniciais:

1. Configuração do Ambiente de Desenvolvimento: Instalação do React.js, bibliotecas auxiliares e configuração da API do TMDb.
2. Design do Layout: Criação de wireframes e protótipos responsivos para dispositivos móveis e desktops.
3. Integração Inicial com a API do TMDb: Carregar e exibir dados básicos (catálogo de filmes, séries, etc.).

Marcos Principais:

- Marco 1: Layout e interface básica desenvolvida.
- Marco 2: Funcionalidades principais (pesquisa, catálogo, perfis) implementadas.
- Marco 3: Recomendações e interatividade implementadas.
- Marco 4: Streaming e funcionalidades mobile integradas.
- Marco 5: Aplicação testada e pronta para lançamento.

7 - Interatividade na Aplicação Web/Mobile:

As interações planejadas para a aplicação garantirão uma experiência fluida e intuitiva:

Interações Básicas:

- Navegação: Menu claro e intuitivo, facilitando a busca e descoberta de novos conteúdos.

- Controle de Reprodução de Vídeo: Comandos de play, pause, e ajuste de volume simples e intuitivos, tanto em desktop quanto em dispositivos móveis.
- Gestos Mobile: Funções de deslizar para avançar/retroceder e toques duplos para pausar/reproduzir.
- Sistema de Avaliação e Comentários: Os usuários poderão avaliar e comentar diretamente nas páginas dos filmes/séries.

8 - Framework:

ReactJS:

O ReactJS permite a criação de interfaces modulares e escaláveis, facilitando o desenvolvimento e a manutenção da aplicação. Com o uso de componentes reutilizáveis, é possível manter um design consistente e de alta performance, essencial para uma aplicação de streaming.

React Native:

Com o React Native, será possível desenvolver uma versão mobile nativa a partir do mesmo código base utilizado no ReactJS, garantindo uma experiência fluida em dispositivos móveis e aproveitando gestos de toque para a navegação e controle de reprodução de vídeos.