



Documento de Projeto de Sistema

UrubuFlix

Vitória, ES

2024

Registro de Alterações:

Versão	Responsável	Data	Alterações
1.0	André Barros, Bruno Santos e Renan Campista	22/04/2024	Versão inicial do Relatório.
2.0	André Barros, Bruno Santos e Renan Campista	10/06/2024	Correções no documento e explicação dos modelos FrameWeb.

1 Introdução

O presente projeto, desenvolvido no âmbito da disciplina de Programação Web, visa aplicar os conhecimentos adquiridos durante o curso para a criação de um sistema web utilizando o framework FrameWeb e explorando a implementação de funcionalidades de Web Semântica. Com base nesse propósito, foi concebida a aplicação web denominada *UrubuFlix*, uma plataforma destinada à visualização de filmes.

UrubuFlix proporciona uma experiência interativa aos seus usuários, oferecendo recursos como pesquisa de filmes, a possibilidade de salvá-los em uma lista de favoritos e a exclusão de filmes erroneamente adicionados. O sistema inclui um mecanismo de cadastro para coleta de informações dos usuários, que são armazenadas em um banco de dados MySQL, uma das tecnologias fundamentais empregadas neste projeto. Além disso, para a construção da interface da aplicação, foram utilizadas tecnologias como React e JavaScript.

Este trabalho não apenas demonstra a aplicação prática dos conceitos aprendidos em sala de aula, mas também busca proporcionar uma experiência de aprendizado enriquecedora e alinhada com as demandas contemporâneas do desenvolvimento web.

2 Plataforma de Desenvolvimento

Desenvolvemos a plataforma deste sistema utilizando a framework React, que se baseia em HTML e CSS para criar a interface da nossa implementação. Nosso principal objetivo foi proporcionar dinamismo à aplicação web, por isso adotamos a tecnologia JavaScript. Isso nos permitiu responder às interações do usuário e ajustar o layout do conteúdo da página de forma fluida. Além disso, contamos com o auxílio do Node.js, que nos oferece a capacidade de interpretar código JavaScript de maneira semelhante ao navegador, o que significa que podemos utilizar JavaScript para criar APIs que interagem com o sistema operacional e o ambiente do servidor. Para garantir um armazenamento de dados confiável, optamos pelo MySQL. Este sistema de gerenciamento de banco de dados oferece aos desenvolvedores recursos eficazes para armazenar, recuperar e manipular dados com segurança e eficiência. Além disso, desenvolvemos um script em python para coletar dados de filmes usando uma API pública. Dessa forma, os dados coletados são enviados ao banco de dados para serem carregados pela nossa aplicação web.

Na Tabela 1 são listadas as tecnologias utilizadas no desenvolvimento da ferramenta, bem como o propósito de sua utilização.

Tabela 1 – Plataforma de Desenvolvimento e Tecnologias Utilizadas.

Tecnologia	Versão	Descrição	Propósito
MySQL	8.0	Banco de dados relacional grátis e open-source.	Armazenamento dos dados manipulados pela ferramenta.
JavaScript	ES2021	Linguagem de programação orientada a objetos e independente de plataforma.	Escrita do código-fonte das classes que compõem o sistema.
React	18.2.0	Framework para a construção de interfaces de usuários baseada em componentes para aplicações Web	Criação das páginas Web e sua comunicação com as classes do JS.
Node.js	18.9.0	Permite que você execute JS no servidor e é adequado para lidar com solicitações HTTP, manipulação de dados, integração com o banco de dados e muito mais.	Implementação para comunicar com banco de dados, solicitações do HTTP recebidas pelo site, implementar API para gerenciar os recursos do site.
Express.js	4.17.1	Framework web para Node.js que simplifica a criação de servidores HTTP e rotas.	Desenvolvimento de APIs RESTful e manipulação de solicitações HTTP no servidor.
Prisma	4.3	Prisma é um ORM (ObjectRelational Mapping) utilizado para mapear entidades do domínio da aplicação para o banco de dados.	Prisma é utilizado para facilitar e simplificar o acesso ao banco de dados.
Python	3.12.3	Linguagem de programação de alto nível, interpretada de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte.	Escrita do código-fonte do script que coleta dados de filmes para utilizarmos em nossa aplicação.

Na Tabela 2 vemos os softwares que apoiaram o desenvolvimento de documentos e também do código fonte.

Tabela 2 – Softwares de Apoio ao Desenvolvimento do Projeto

Tecnologia	Versão	Descrição	Propósito
Overleaf	2.12	Editor de LaTeX online.	Escrita da documentação do sistema, sendo usado o template abnTeX
GitHub	2.34.1	Plataforma web para gerenciamento de repositórios git	Gerenciar o repositório do projeto remotamente, tornando-o disponível aos colaboradores do projeto e à quem possa interessar-se por ele.
Git	2.34.1	É um sistema de controle de versões distribuídos, usado principalmente no desenvolvimento de software.	Gerenciar os códigos fonte e permitir colaboração e desenvolvimento do software em conjunto.

Tecnologia	Versão	Descrição	Propósito
Visual Studio Code	1.87	Ambiente de desenvolvimento (IDE) com suporte ao desenvolvimento JavaScript e React.	Implementação, implantação e testes da aplicação Web para Js e React.
Visual Paradigm	17.1	Software usado para modelagem, diagramação e design de sistemas de software, assim como processos e arquiteturas de negócios.	Desenvolvimento dos modelos FrameWeb relativos às diferentes camadas da arquitetura.
Plugin FrameWeb	1.0	Plugin para o Visual Paradigm que contém layouts para os modelos FrameWeb.	Auxiliar na criação dos modelos FrameWeb dentro do Visual Paradigm.

3 Arquitetura de Software

A figura 1 mostra a arquitetura do sistema *UrubuFlix*.

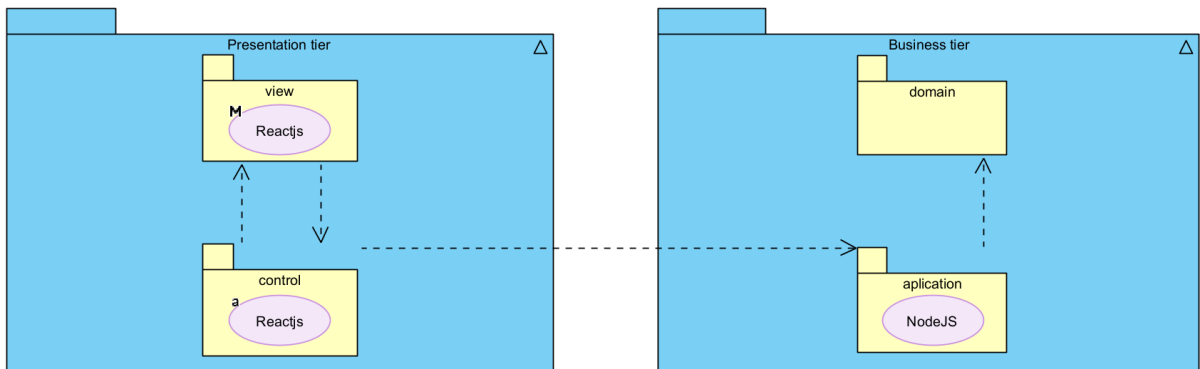


Figura 1 – Arquitetura de Software.

Na camada de Frontend, a visualização será conduzida pelo framework React.js para o desenvolvimento de aplicações em JavaScript. Além disso, optaremos por separar estritamente o Frontend do Backend, facilitando a comunicação entre eles por meio de requisições HTTP ao servidor. Para gerenciar essas requisições, contaremos com a biblioteca Express.js.

No âmbito do Backend, adotaremos o framework Node.js, empregando-o para a gestão das demais funcionalidades da aplicação, enquanto a gestão dos controladores será realizada pelo React.js. Quanto ao banco de dados, optaremos pelo banco relacional

MySQL.

4 Modelagem FrameWeb

UrubuFlix é um sistema Web cuja arquitetura adota *frameworks* comumente empregados no desenvolvimento para esta plataforma. Dessa maneira, o sistema pode ser modelado seguindo a abordagem FrameWeb (SOUZA, 2020).

Na tabela 3 são listados *frameworks* presentes na arquitetura do sistema, classificados em cada uma das categorias suportadas pela abordagem FrameWeb. Em seguida, são apresentados os modelos FrameWeb para cada camada da arquitetura.

Tabela 3 – *Frameworks* da arquitetura do sistema separados por categoria.

Categoria de <i>Framework</i>	<i>Framework</i> Utilizado
Controlador Frontal	React
Injeção de Dependências	Node
Mapeamento Objeto/Relacional	Prisma
Segurança	JWT

4.1 Camada de Negócio

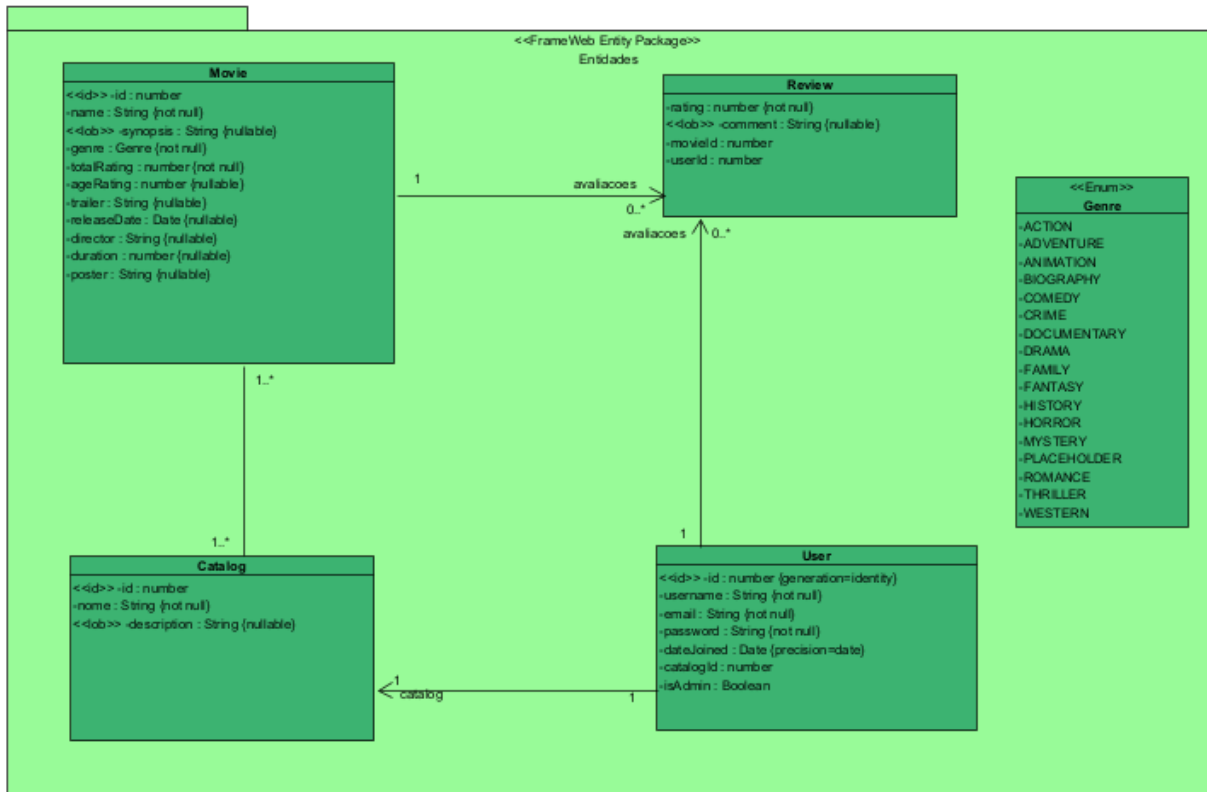


Figura 2 – Modelo de entidades.

A figura 2 apresenta o modelo de entidades da nossa aplicação. Na classe "Movie", encontram-se diversas informações sobre um filme, como gênero, título, sinopse e duração. Cada filme pode estar associado a um "Catalog", o qual agrupa diversos filmes com base em seus gêneros. A classe "User" representa os usuários da plataforma. Cada usuário tem a capacidade de realizar avaliações para diferentes filmes disponíveis na plataforma. Além disso, cada usuário possui um catálogo pessoal, onde estão listados os filmes favoritos por ele. Também há a possibilidade do usuário ser um "Admin", o que dá o poder de gerenciar outros usuários, filmes, catálogos e avaliações.

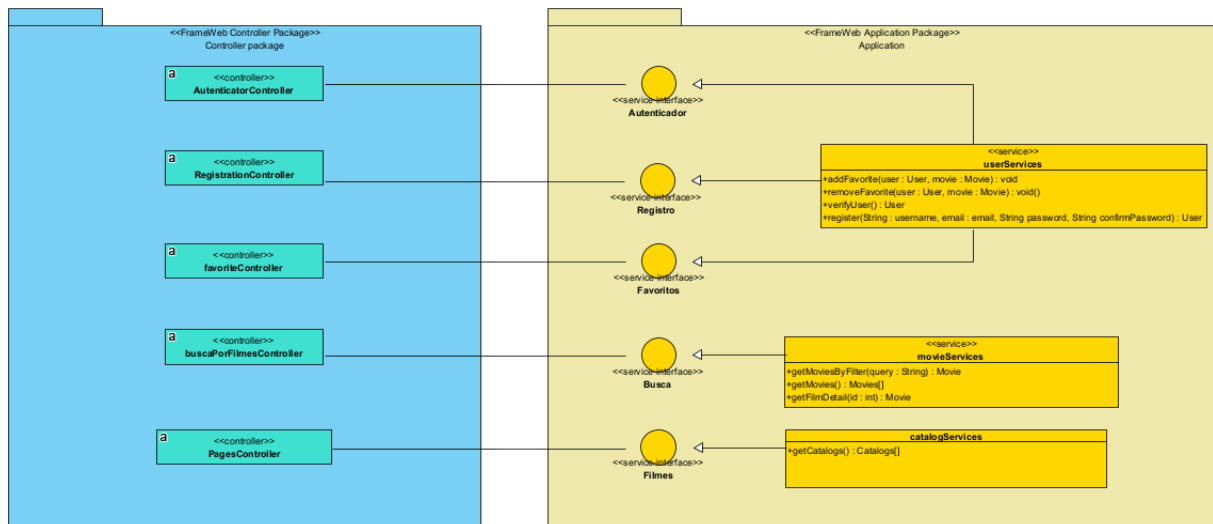


Figura 3 – Modelo de aplicação.

O modelo de aplicação presente na figura 3 apresenta os serviços essenciais da nossa plataforma. Isso inclui o serviço de autenticação de login, que não apenas permite que os usuários ingressem com segurança, mas também oferece funcionalidades para alterar e recuperar senhas. O serviço de cadastro é responsável por verificar se um usuário pode se inscrever na plataforma, garantindo um processo de registro seguro e eficiente. O serviço de favoritos permite aos usuários marcar filmes como favoritos e criar catálogos personalizados com base nesses filmes selecionados. O serviço de busca oferece uma ferramenta robusta de pesquisa, permitindo que os usuários encontrem facilmente filmes dentro da plataforma e apliquem filtros de pesquisa para refinar seus resultados. Por fim, o serviço de filmes dá ao usuário a possibilidade de acessar as informações detalhadas do filme, como diretor, duração e as avaliações. Esses serviços são fundamentais para proporcionar uma experiência completa e satisfatória aos usuários da nossa aplicação.

Como foi usado o Prisma e ele não requer a criação de DAOs (Data Access Objects) porque ele adota uma abordagem mais direta e automatizada para o acesso e manipulação de dados no banco de dados. Em vez de definir classes adicionais para representar e acessar os dados (como é comum em outros frameworks ORM), o Prisma permite que você defina os modelos de dados diretamente em um arquivo `schema.prisma`, especificando os tipos de dados, relacionamentos e restrições do banco de dados de forma declarativa. O Prisma gera automaticamente um cliente (PrismaClient) com métodos para executar operações CRUD e consultas personalizadas com base nos modelos definidos, eliminando assim a necessidade de escrever código adicional para acessar os dados do banco. Isso simplifica o desenvolvimento, reduz a quantidade de código necessário e fornece uma camada de abstração eficiente entre a aplicação e o banco de dados.

4.2 Camada de Apresentação

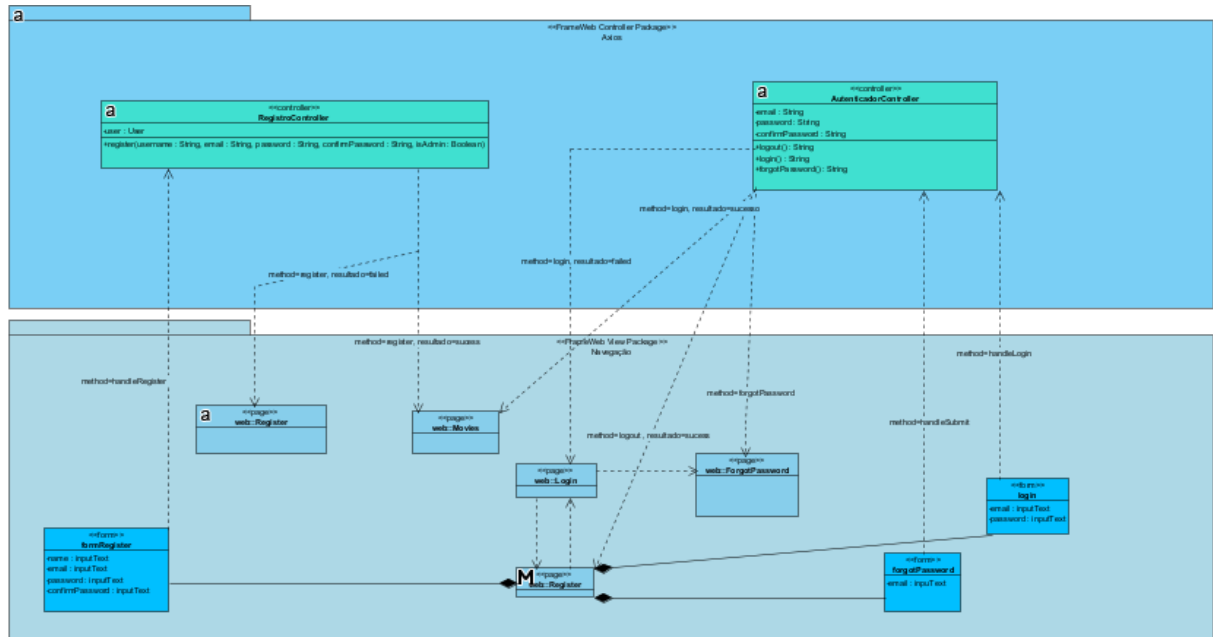


Figura 4 – Navegação de registro.

A figura 4 apresenta o modelo de navegação de registro. Nela, destacam-se três formulários associados à página inicial. O formulário de registro é responsável pela admissão de novos usuários. Ao preencher os dados, estes são enviados ao controlador de registro, onde são validados. O controlador verifica se as informações foram inseridas corretamente para o cadastro na plataforma. O formulário de login direciona os dados a outro controlador, encarregado de autenticar o acesso. O resultado do processo é retornado, indicando se o login foi bem-sucedido ou não. Em caso de esquecimento da senha, o formulário de recuperação permite ao usuário solicitar sua restauração.

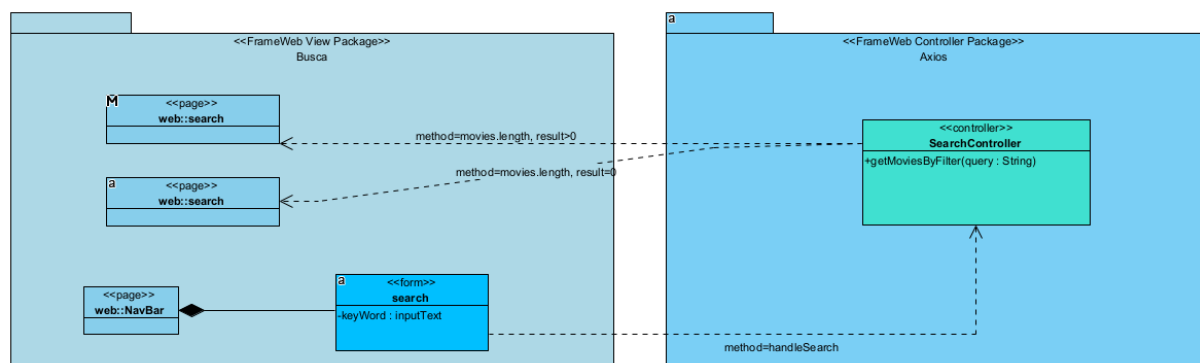


Figura 5 – Navegação de busca.

A figura 5 apresenta o modelo de navegação de busca, fornecendo aos usuários uma interface intuitiva para explorar a plataforma. O formulário de busca permite aos

usuários pesquisar um ou mais filmes pelo nome, além de oferecer a opção de utilizar filtros para refinar seus resultados. Os dados coletados no formulário são enviados ao controlador de busca, responsável por processar a solicitação. Após a busca, o controlador retorna os resultados, indicando se foram encontrados filmes correspondentes ou não. Em caso afirmativo, o usuário é redirecionado para uma página que lista os filmes associados à busca realizada. Caso contrário, uma mensagem informativa é exibida, comunicando que não foram encontrados filmes correspondentes aos critérios de busca. Este fluxo de navegação garante uma experiência eficaz e direcionada aos usuários durante a pesquisa de filmes na plataforma.

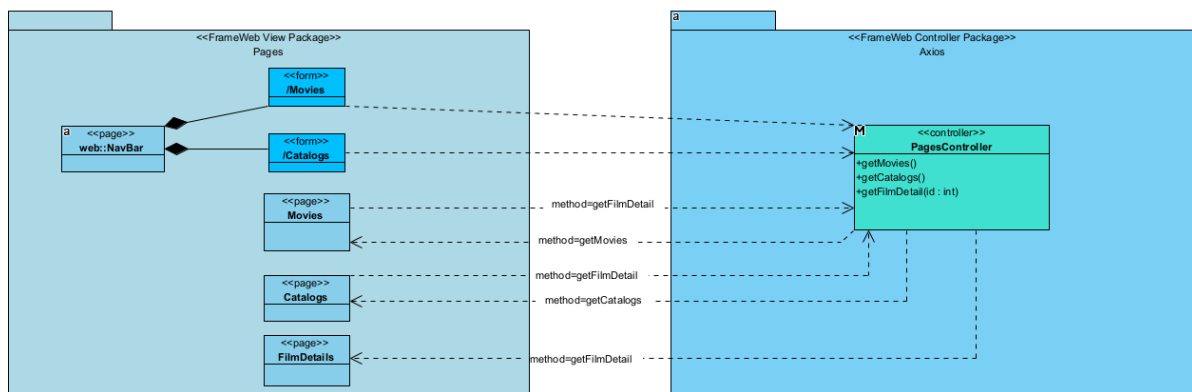


Figura 6 – Navegação de favoritos.

A figura 6 acima ilustra o modelo de navegação das páginas relacionadas a filmes dentro da plataforma. O diagrama mostra a interação entre diferentes componentes, incluindo o menu de navegação (NavBar), as páginas de filmes, catálogos e detalhes dos filmes, e o controlador responsável por gerenciar as solicitações de dados. Esse modelo destaca como os usuários podem navegar através das diferentes seções relacionadas a filmes e como o controlador processa e retorna as informações solicitadas.

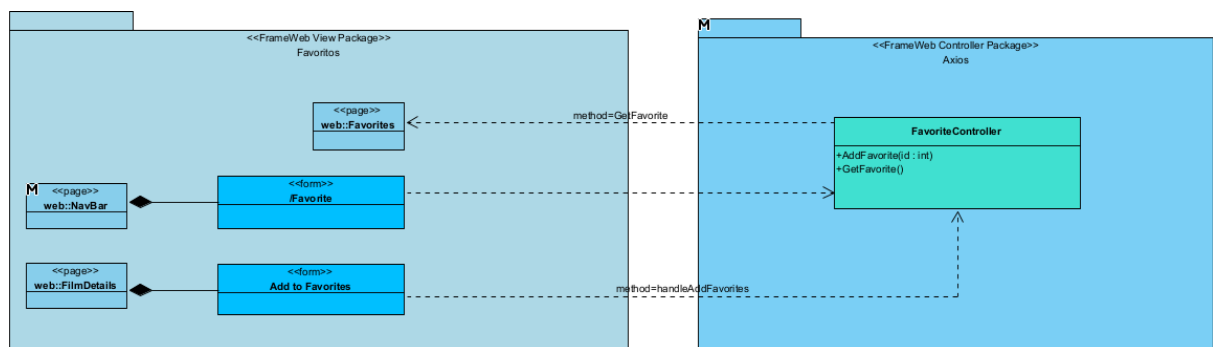


Figura 7 – Navegação de filmes.

Por último, a figura 7 representa o modelo de navegação de favoritos. Ao visitar a página de um filme, o usuário tem a opção de favoritá-lo ao clicar em um botão dedicado.

Caso o filme já esteja nos favoritos, o mesmo botão permitirá ao usuário removê-lo da lista. Além disso, o usuário pode acessar sua lista de favoritos ao clicar em outro botão designado para essa finalidade. Ao fazer isso, será direcionado para uma página contendo um catálogo com todos os filmes que ele adicionou aos favoritos. Essa funcionalidade proporciona aos usuários um método simples e conveniente para gerenciar seus filmes favoritos dentro da plataforma.

Referências

SOUZA, V. E. S. The FrameWeb Approach to Web Engineering: Past, Present and Future. In: ALMEIDA, J. P. A.; GUIZZARDI, G. (Ed.). *Engineering Ontologies and Ontologies for Engineering*. 1. ed. Vitória, ES, Brazil: NEMO, 2020. cap. 8, p. 100–124. ISBN 9781393963035. Disponível em: <<http://purl.org/nemo/celebratingfalbo>>. Citado na página 5.