

Documentação

1. Introdução

Este documento apresenta a API desenvolvida para intermediar a comunicação entre o frontend da aplicação e a API pública do Star Wars. A API tem como objetivo fornecer informações detalhadas sobre os filmes da saga, organizando-os de forma acessível para os usuários.

A aplicação foi construída utilizando a arquitetura **MVC (Model-View-Controller)**, com **PHP 7.4** e princípios de **Programação Orientada a Objetos (POO)**.

O documento também inclui instruções detalhadas para instalação e configuração, cobrindo requisitos como PHP 7.4, MySQL e servidor Apache, além dos passos necessários para rodar a aplicação localmente.

2. Objetivo

A API foi criada para atender aos seguintes objetivos principais:

- Exibir uma lista de todos os filmes de Star Wars, ordenados pela data de lançamento;
- Disponibilizar informações detalhadas de cada filme, incluindo:
 - Título;
 - Número do episódio;
 - Sinopse;
 - Data de lançamento;
 - Diretor e produtores;
 - Lista de personagens;
 - Cálculo da idade do filme (anos, meses e dias);
- Registrar logs de todas as solicitações, armazenando informações como data, horário e tipo de requisição para fins de monitoramento e auditoria.

3. Estrutura da API

O arquivo principal da API (**swapi.php**) segue um fluxo bem definido para garantir um funcionamento adequado. As principais etapas do processo são:

3.1. Configuração Inicial

A API carrega arquivos de configuração contendo informações essenciais, como credenciais do banco de dados e parâmetros gerais da aplicação.

3.2. Gerenciamento de Sessão

Ao iniciar a API, a sessão do usuário é iniciada para controle de autenticação.

3.3. Definição do Formato de Resposta

Todas as respostas são enviadas no formato JSON para garantir compatibilidade com o frontend.

3.4. Processamento das Requisições

A API analisa as requisições recebidas (via método **GET**) e decide se deve listar todos os filmes ou buscar informações específicas de um filme, dependendo dos parâmetros enviados.

3.5. Retorno dos Dados

Os dados são obtidos da API pública do Star Wars, tratados no backend e enviados ao frontend no formato esperado.

4. Endpoints Disponíveis

A API oferece dois principais endpoints para acesso às informações dos filmes:

4.1. Listagem de Filmes

- **Método:** (GET)
- **Descrição:** Retorna a lista completa de filmes ordenados por data de lançamento.
- **Parâmetros:** Nenhum.
- **Exemplo de URL:** /api/filmes

4.2. Detalhes de um Filme

- **Método:** (GET)
- **Descrição:** Retorna informações detalhadas de um filme específico.
- **Parâmetros:** ID do filme.
- **Exemplo de URL:** /api/filmes?id=1

5. Registro de Logs

Todas as solicitações feitas à API são registradas para fins de acompanhamento e auditoria. Os registros armazenam:

- **id** – Identificador único da requisição;
- **request_type** – Tipo de requisição realizada (ex.: GET, POST);

- **endpoint** – Endpoint acessado pelo usuário;
- **status** – Status da requisição (ex.: 200 para sucesso, 400 para erro);
- **response** – Mensagem descritiva relacionada à requisição, como "Iniciando requisição para obter filmes" ou "Filmes obtidos com sucesso".
- **created_at** – Data e hora da solicitação.

Esse mecanismo é essencial para monitoramento e diagnóstico de problemas futuros.

6. Aspectos Técnicos

Algumas considerações técnicas importantes sobre o desenvolvimento da API:

- O backend foi desenvolvido em **PHP 7.4**, garantindo ampla compatibilidade com servidores;
- O projeto segue um código enxuto e organizado, sem a utilização de frameworks PHP, facilitando a manutenção e compreensão;
- No frontend, foram utilizadas as bibliotecas **jQuery** e **Bootstrap**, garantindo uma interface amigável e responsiva.

7. Instalação e Configuração

Para rodar o projeto corretamente, siga os passos abaixo:

7.1. Requisitos Necessários

- PHP 7.4 instalado no servidor;
- Banco de dados MySQL configurado;
- Servidor web Apache;

7.2. Passos para Instalação

1. Configure o banco de dados importando o arquivo de dump fornecido;
2. Atualize as credenciais de acesso ao banco no arquivo de configuração;
3. Suba o projeto em um ambiente local (XAMPP, WAMP ou equivalente);

8. Melhorias Implementadas

Além das funcionalidades básicas exigidas, foram implementadas melhorias para enriquecer a experiência do usuário, tais como:

8.1. Filtro por Nome e Data de Lançamento

Agora é possível buscar filmes pelo nome ou por um intervalo de datas, facilitando a busca.

Benefícios:

- Permite encontrar filmes rapidamente;
- Experiência mais intuitiva.

8.2. Sistema de Favoritos e Autenticação de Usuário

Foi criado um sistema de favoritos, onde os usuários podem salvar seus filmes preferidos e acessá-los posteriormente.

Funcionalidades:

- Cadastro e login de usuários;
- Favoritos armazenados no banco de dados.

Benefícios:

- Personalização da experiência do usuário.

8.3. Exibição de Trailers e Capas dos Filmes

Foram adicionadas imagens de capa e trailers oficiais na listagem e na página de detalhes dos filmes.

Benefícios:

- Torna a interface mais visual e envolvente;
- Ajuda os usuários a terem uma visão prévia do conteúdo.

9. Instruções para Testes

Para testar a aplicação corretamente:

1. Acesse o catálogo de filmes pela URL principal da aplicação:
http://localhost/processo_seletivo/view/index.php
2. Utilize os filtros para refinar a busca conforme necessário.
3. Clique em um filme para visualizar mais detalhes.
4. Utilize as opções de favoritos caso tenha feito login na aplicação.