

Documentação da Web API de Filmes, Séries Esportes

Resumo do Projeto

A Web API de Filmes, Séries e Esportes é uma aplicação desenvolvida em C# que oferece funcionalidades para gerenciar informações relacionadas a filmes, séries, usuários e esportes. Esta API permite aos usuários acessar, adicionar, modificar e excluir dados dessas quatro classes principais. Ela utiliza o MySQL Workbench como banco de dados e é controlada pelo Entity Framework para facilitar a manipulação dos dados.

Principais Funcionalidades

A seguir, estão listadas as principais funcionalidades oferecidas pela Web API:

Filmes

1. Listar Filmes

- Rota: GET /api/filmes
- Descrição: Retorna uma lista de todos os filmes cadastrados.

2. Obter Filme por ID

- Rota: GET /api/filmes/{id}
- Descrição: Retorna os detalhes de um filme específico com base no ID fornecido.

3. Adicionar Filme

- Rota: POST /api/filmes
- Descrição: Permite adicionar um novo filme ao banco de dados.

4. Atualizar Filme

- Rota: PUT /api/filmes/{id}
- Descrição: Permite atualizar os detalhes de um filme existente com base no ID fornecido.

5. Excluir Filme

- Rota: DELETE /api/filmes/{id}
- Descrição: Permite excluir um filme com base no ID fornecido.

Séries

1. Listar Séries

- Rota: GET /api/series
- Descrição: Retorna uma lista de todas as séries cadastradas.

2. Obter Série por ID

- Rota: GET /api/series/{id}
- Descrição: Retorna os detalhes de uma série específica com base no ID fornecido.

3. Adicionar Série

- Rota: POST /api/series
- Descrição: Permite adicionar uma nova série ao banco de dados.

4. Atualizar Série

- Rota: PUT /api/series/{id}
- Descrição: Permite atualizar os detalhes de uma série existente com base no ID fornecido.

5. Excluir Série

- Rota: DELETE /api/series/{id}
- Descrição: Permite excluir uma série com base no ID fornecido.

Usuários

1. Listar Usuários

- Rota: GET /api/usuarios
- Descrição: Retorna uma lista de todos os usuários cadastrados.

2. Obter Usuário por ID

- Rota: GET /api/usuarios/{id}
- Descrição: Retorna os detalhes de um usuário específico com base no ID fornecido.

3. Adicionar Usuário

- Rota: POST /api/usuarios
- Descrição: Permite adicionar um novo usuário ao banco de dados.

4. Atualizar Usuário

- Rota: PUT /api/usuarios/{id}
- Descrição: Permite atualizar os detalhes de um usuário existente com base no ID fornecido.

5. Excluir Usuário

- Rota: DELETE /api/usuarios/{id}
- Descrição: Permite excluir um usuário com base no ID fornecido.

Esportes

1. Listar Esportes

- Rota: GET /api/esportes
- Descrição: Retorna uma lista de todos os esportes cadastrados.

2. Obter Esporte por ID

- Rota: GET /api/esportes/{id}
- Descrição: Retorna os detalhes de um esporte específico com base no ID fornecido.

3. Adicionar Esporte

- Rota: POST /api/esportes
- Descrição: Permite adicionar um novo esporte ao banco de dados.

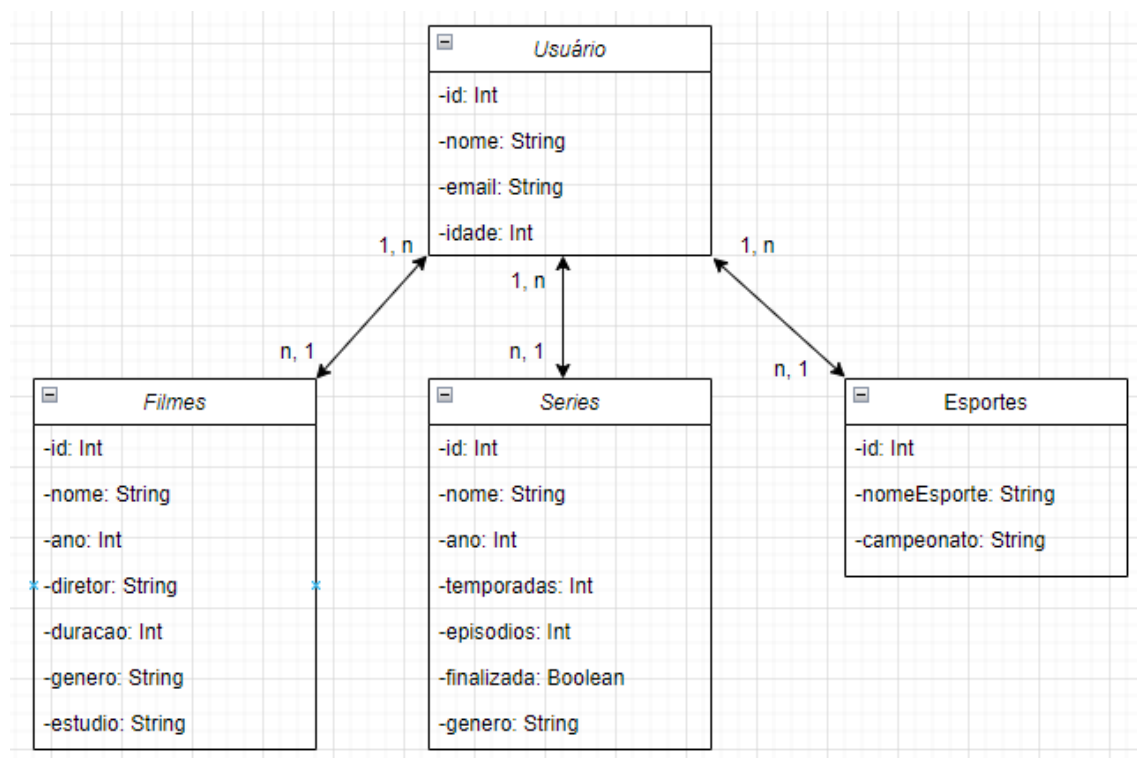
4. Atualizar Esporte

- Rota: PUT /api/esportes/{id}
- Descrição: Permite atualizar os detalhes de um esporte existente com base no ID fornecido.

5. Excluir Esporte

- Rota: DELETE /api/esportes/{id}
- Descrição: Permite excluir um esporte com base no ID fornecido.

Diagrama de classes UML



Relatório de atividades desenvolvidas

- Controller e Model da classe Usuário: Vitor
- Controller e Model na classe Filmes: Kayky
- Controller e Model na classe Séries: José Arthur
- Controller e Model na classe Esportes: Chatgpt
- DbContext: Todos participaram conforme suas devidas classes.