

Capítulo 1 : Apresentação API

Descrição

Esta API foi desenvolvida com o recurso à OpenAPI, usando uma camada de serviços REST.

A API tem o objetivo de ser possível a consulta de filmes. Consegue ver todos os detalhes de um filme, dos generos, atores e realizadores.

Para esta API foram usados 6 controladores:

- authController
- callbackController
- dashboardController
- jogadorController
- partidaController
- swaggerController

> authController

É composto por 6 funcionalidades:

- **login** -> Retorna a página de login.
- **logout** -> Retorna a página de logout.
- **protected** -> Retorna a página protected.
- **dashboard** -> Retorna a página de dashboard.
- **authGitHub** -> Retorna a página de login com o GitHub.

> callbackController

É composto por 6 funcionalidades:

- **GET(Count)** -> Retorna o número de realizadores presente na base de dados.
- **GET** -> Retorna um array com todos os realizadores presentes na base de dados.
- **POST** -> Adiciona um diretor à base de dados.
- **GET{id}** -> Dado um ID, retorna o respetivo diretor.
- **PUT{id}** -> Efetua alterações a um dado diretor com um respetivo ID.
- **DELETE{id}** -> Elimina um diretor da base de dados com um dado ID.

> dashboardController

É composto por 6 funcionalidades:

- **GET** -> Retorna um array com todos os jogadores presentes na base de dados.

> jogadorController

É composto por 6 funcionalidades:

- **countJogadores** -> Retorna o número de jogadores presente na base de dados.
- **retrieveJogadores** -> Retorna um array com todos os jogadores presentes na base de dados.
- **createJogador** -> Adiciona um jogador à base de dados.
- **retrieveJogador** -> Dado um ID, retorna o respetivo jogador.
- **updateJogador** -> Efetua alterações a um dado jogador com um respetivo ID.
- **deleteJogador** -> Elimina um jogador da base de dados com um dado ID.

> partidaController

É composto por 1 funcionalidade:

- **countPartidas** -> Retorna o número de partidas presente na base de dados.
- **retrievePartidas** -> Retorna um array com todas as partidas presentes na base de dados.
- **createPartida** -> Adiciona uma partida à base de dados.
- **retrievePartida** -> Dado um ID, retorna a respetiva partida.
- **updatePartida** -> Efetua alterações a uma dada partida com um respetivo ID.
- **deletePartida** -> Elimina uma partida da base de dados com um dado ID.

> swaggerController

É composto por 1 funcionalidade:

- **showSwaggerController** -> Retorna todas as partidas associadas a um jogador através de um ID.

Grupo 23

- Dario Rodrigues A038042@umaia.pt
- João Aragão A0939132@umaia.pt
- Bernardo Magalhães A38819@umaia.pt

< Anterior ^ Main Próximo >
