# Licenciatura em Engenharia de Sistemas Informáticos

Integração de Sistemas de Informação

# Relatório do trabalho prático 2

# 2020/2021

# Movieseek V2

# Listagem de filmes/séries

Realizado por:

Daniel Arantes Eleutério,

Nº 12715

Conteúdo

[Resumo 2](#_Toc64026596)

[Objetivos 3](#_Toc64026597)

[Arquitetura da solução 4](#_Toc64026598)

[Base de dados 4](#_Toc64026599)

[API 4](#_Toc64026600)

[Aplicação 7](#_Toc64026601)

[Conclusão 10](#_Toc64026602)

# Resumo

Este projeto irá utilizar o trabalho prático anterior, que consistiu numa aplicação em Windows Forms que vai à API do OMDB (<http://www.omdbapi.com/>) buscar uma listagem de filmes consoante uma pesquisa feita. Com estes filmes/series que obtemos na listagem, podemos adicioná-los à nossa listagem, gravada localmente num ficheiro de texto.

Nesta nova fase, irá ser adicionado uma API estruturada com autenticação e também com uma base de dados SQL server para gravar utilizadores e filmes para cada utilizador.

Este projeto surgiu com a necessidade de um existir uma listagem onde possamos ter registado, nesta fase em que todos passamos muito mais tempo em casa, todos os filmes e séries que já assistimos.

Palavras-chave: Filmes, Séries, Listas.

# Objetivos

- Ver uma listagem com os nossos filmes/series

- Ver listagem com todos os filmes existente.

- Procurar por nome do filme/serie na listagem do OMDB.

- Aplicação ser fluida.

- Aplicação sem erros inesperados.

- Autenticação.

- Gravar cada filme numa base de dados.

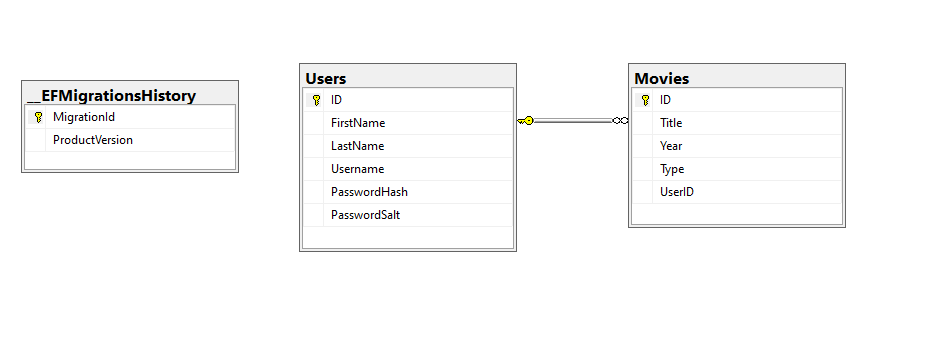
- Criar uma API que possa ser utilizada em várias soluções.

- Poderem ser criados utilizadores.

- Poderem ser criados filmes para cada utilizador.

# Arquitetura da solução

## Base de dados



*Figura 1 – Modelo final da base de dados*

A base de dados seria uma coisa bastante simples, onde teríamos um utilizador com os dados base, como primeiro nome, último, nome de utilizador e depois dados relativos à password. Associado a esses utilizadores teríamos os filmes adicionados pelos mesmos.

## API

A API será feita em ASP.NET CORE. O principal objectivo da API é que possa ser integrada com qualquer plataforma. Para isso, foi preciso ter uma maior atenção à documentação.

Na API, temos:

- Autenticação com JWT.

- Modelos e migrations com Entity framework.

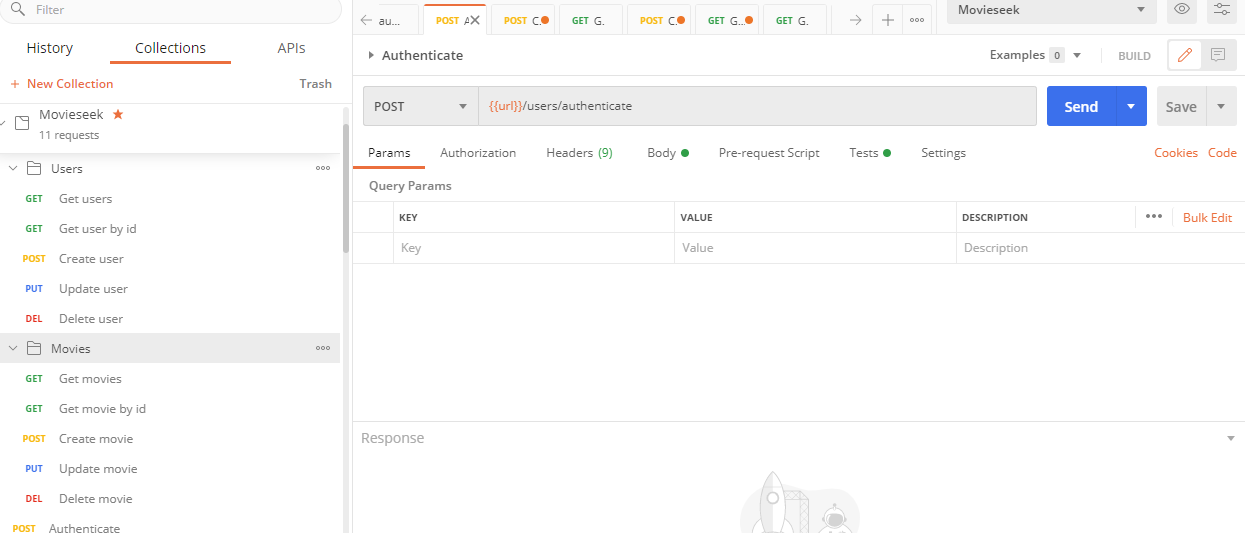
- Depencies Injection.

- Integração com swagger.

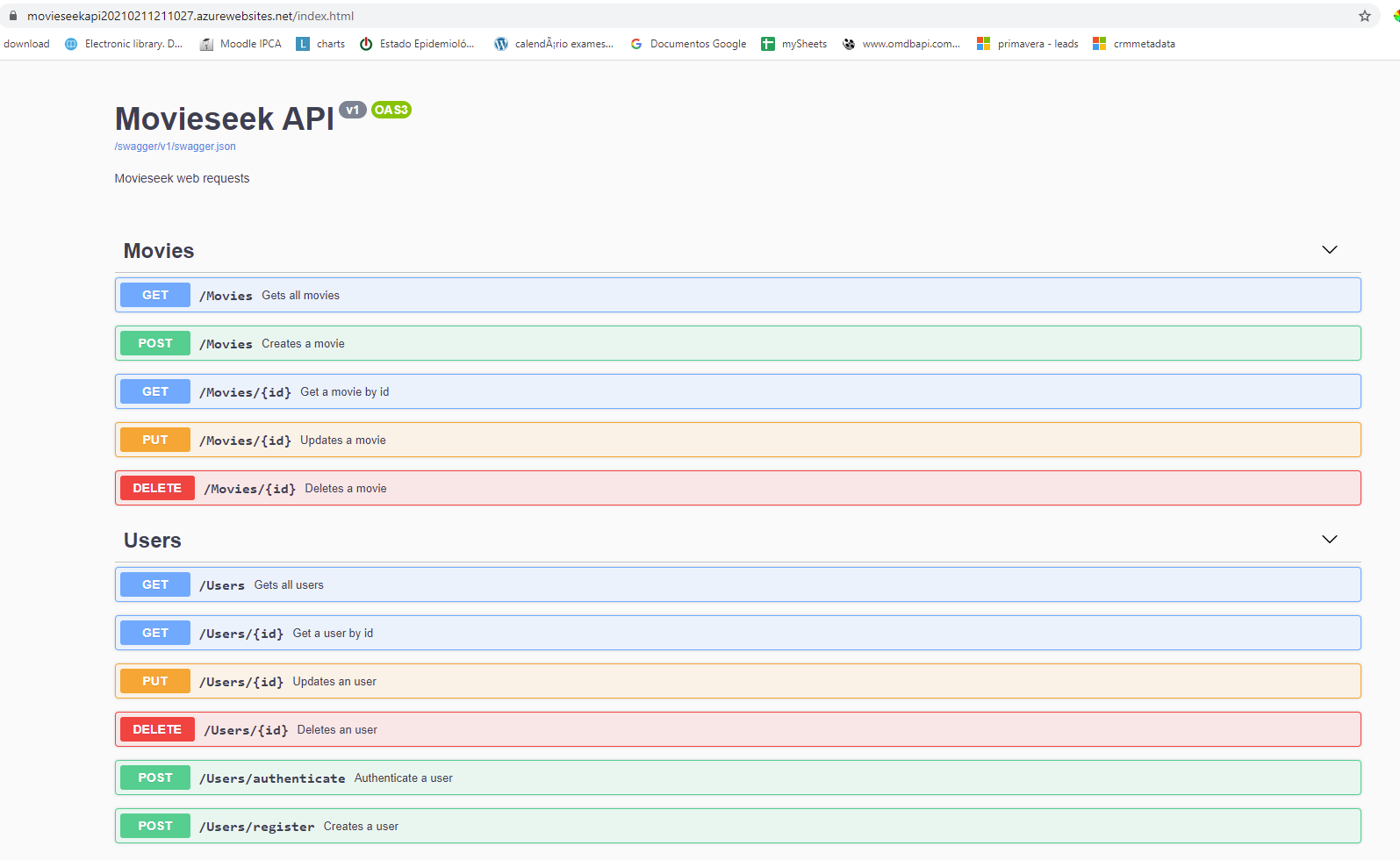
- Pasta de documentação com o environment e collection do postman.

- Publicada a base de dados e a API em Azure.

- Pedidos CRUD base a todas as entidades.

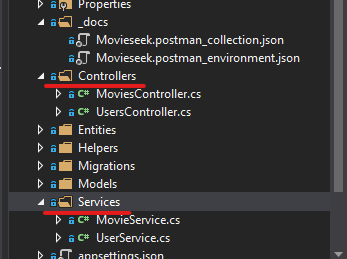


*Figura 2 – Coleção postman*

**

*Figura 3 – Swagger integration*

Na API foram criadas a camada de controlador e serviços, onde, o controlador faz a verificação se os dados estão todos corretos e envia-os para um serviço. Este serviço será o que vai à base de dados alterar ou gravar a sua entidade.

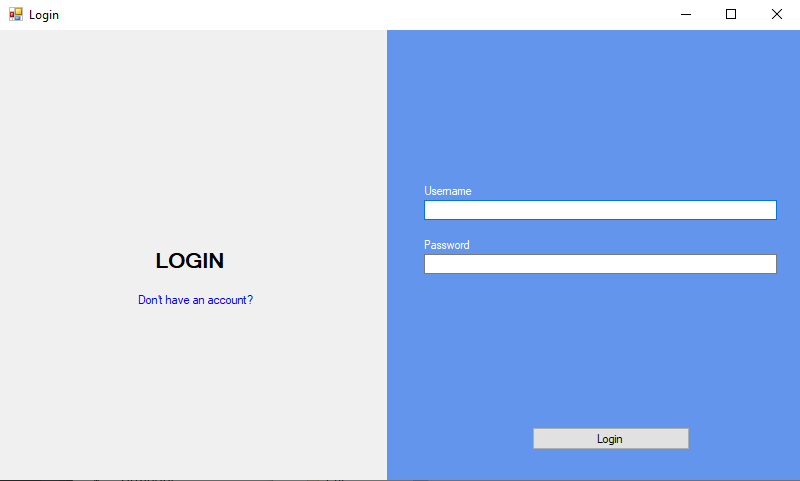


*Figura 4 – Estrutura da API*

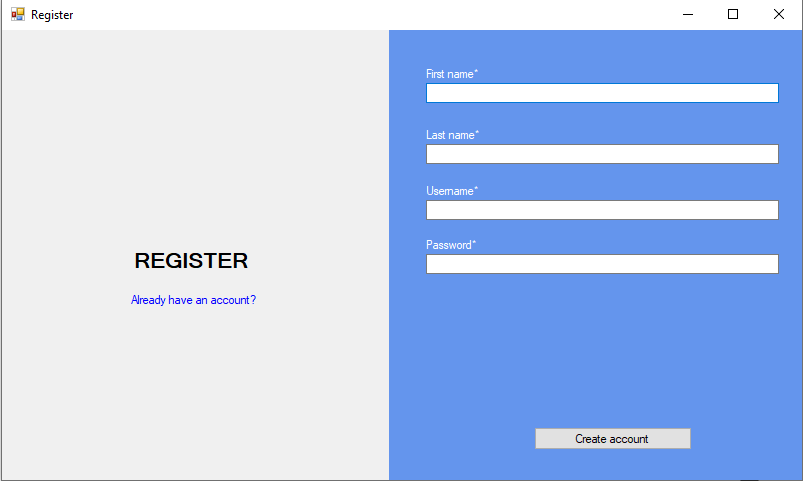
Endereço da API: <https://movieseekapi20210211211027.azurewebsites.net>

## Aplicação

Na aplicação começamos pelo ecrã de Login ou registo.



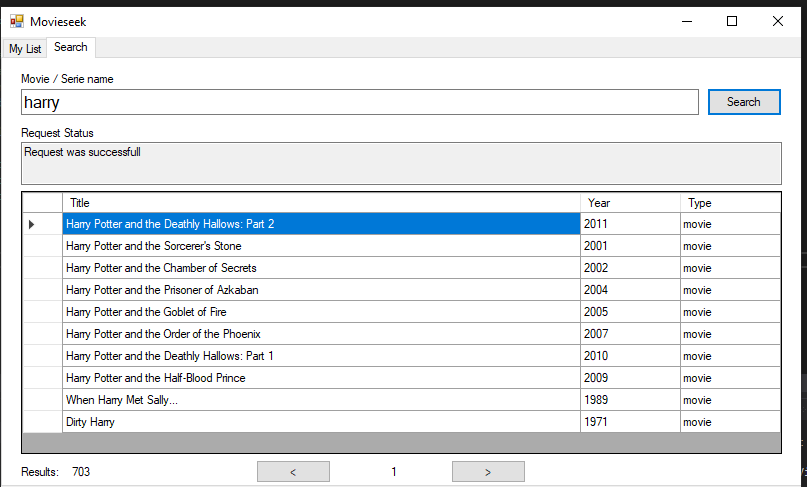
*Figura 5 – Ecrã de login*

**

*Figura 6 – Ecrã de registo*

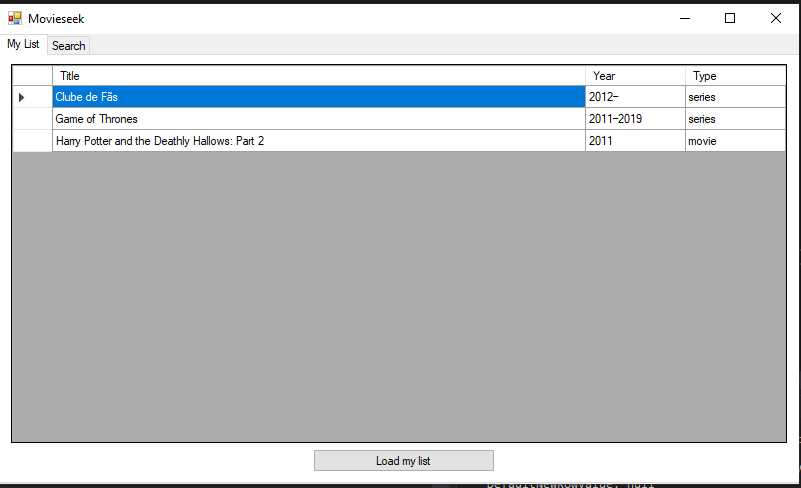
O principal fluxo da aplicação é a aplicação Movieseek, vai à API do OMDB obter uma listagem com todos os filmes existentes consoante a pesquisa feita e mostra-os em forma de tabela.

O tipo de transformação existente, é transformar um objeto do tipo JSON em uma classe que seja reconhecida pela aplicação.



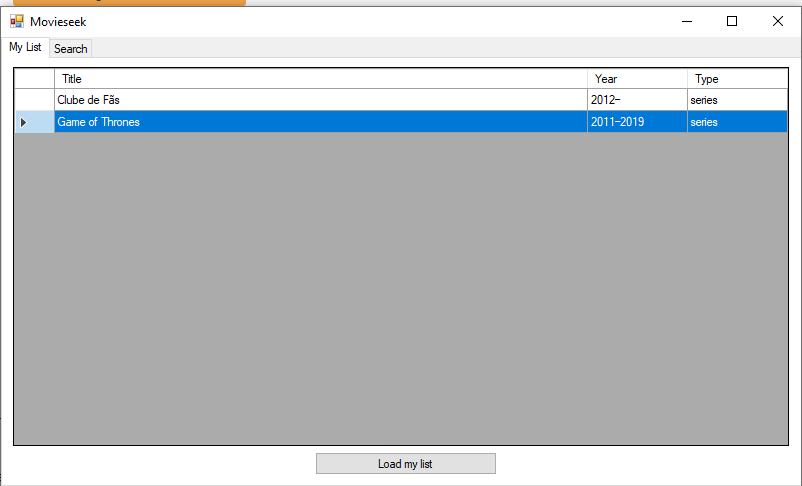
*Figura 7 – Procura de filmes pela OMDB API.*

Nessa tabela o utilizador pode clicar 2x numa linha e o filme passará para a listagem dos meus filmes já assistidos.



*Figura 8 – Listagem de filmes adicionados.*

Há a possibilidade de remover um filme previamente adicionado, para isso, basta selecionar a linha e carregar em delete.



*Figura 9 – Listagem depois de removido um filme.*

# Conclusão

Neste trabalho foi abordada a construção de uma API e a publicação da mesma. Foi desafiante porque ainda não tinha utilizado nenhuma das tecnologias abordadas mas no final penso que tenha sido muito benéfico pois permitiu-me ver e aprender muito sobre o assunto (API .NET Core e Windows forms)

Penso que tenha comprido todos os objetivos que tinha inicialmente previsto e, após concluído o trabalho, fiquei satisfeito com a solução desenvolvida embora que, num ambiente profissional teriam alguns pontos que ser melhorados tanto na API como na aplicação.