

Documentação

OpFlix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Alameda Barão de Limeira, 539 – São Paulo/SP |  |  |  | sp.senai.br |

Sumário

1. [Resumo 2](#_Toc533767843)

[Objetivos](#_Toc533767844)

1. [Descrição do projeto 2](#_Toc533767845)

[Resumo do projeto](#_Toc533767846)

1. [Modelagem de Software 3](#_Toc533767847)

[Modelo Lógico](#_Toc533767848)

[Modelo Físico](#_Toc533767849)

[Modelo Conceitual](#_Toc533767850)

[Cronograma](#_Toc533767851)

1. [Funcionalidades 5](#_Toc533767852)

[Web](#_Toc533767853)

[Mobile](#_Toc533767854)

1. [Protótipos 6](#_Toc533767855)

[Web](#_Toc533767856)

[Mobile](#_Toc533767857)

1. [Front-End 7](#_Toc533767858)
2. [Mobile 8](#_Toc533767859)
3. [Arquitetura do Projeto 9](#_Toc533767860)
4. [Referências 10](#_Toc533767861)

[Links](#_Toc533767862)

[Livros](#_Toc533767863)

Resumo

Objetivos

Este documento tem como objetivo demonstrar todas as funcionalidades da plataforma OpFlix, na qual visa a divulgação de lançamentos de filmes e séries.

Descrição do projeto

O OpFlix tem como objetivo a divulgação de lançamentos de filmes e séries e foi desenvolvido na Escola SENAI de Informática.

Resumo do projeto

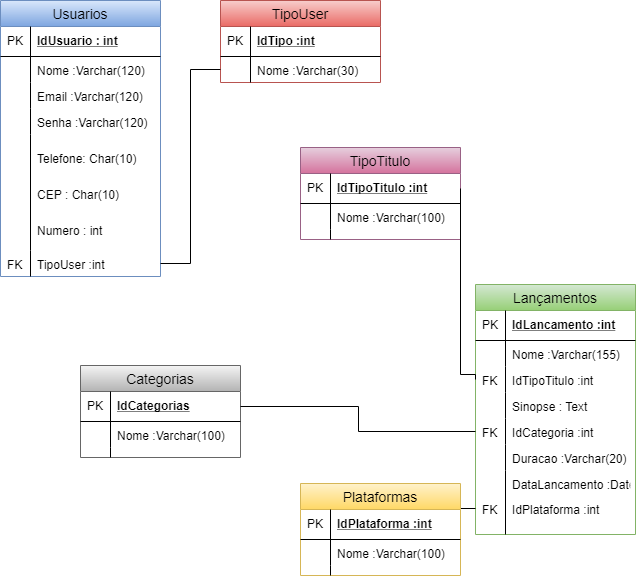
Uma nova companhia no ramo cinematográfico chamada OpFlix, empresa de pequeno porte que atua no ramo de filmes e séries, foi criada por Tadeu em 2019 na região da Paulista em São Paulo. Tadeu tem uma equipe de parceiros que desejam realizar a divulgação de novos lançamentos por ano.

Modelagem de Software

Esta sessão tende a demonstrar o que será desenvolvido no banco de dados e quais são as suas representações.

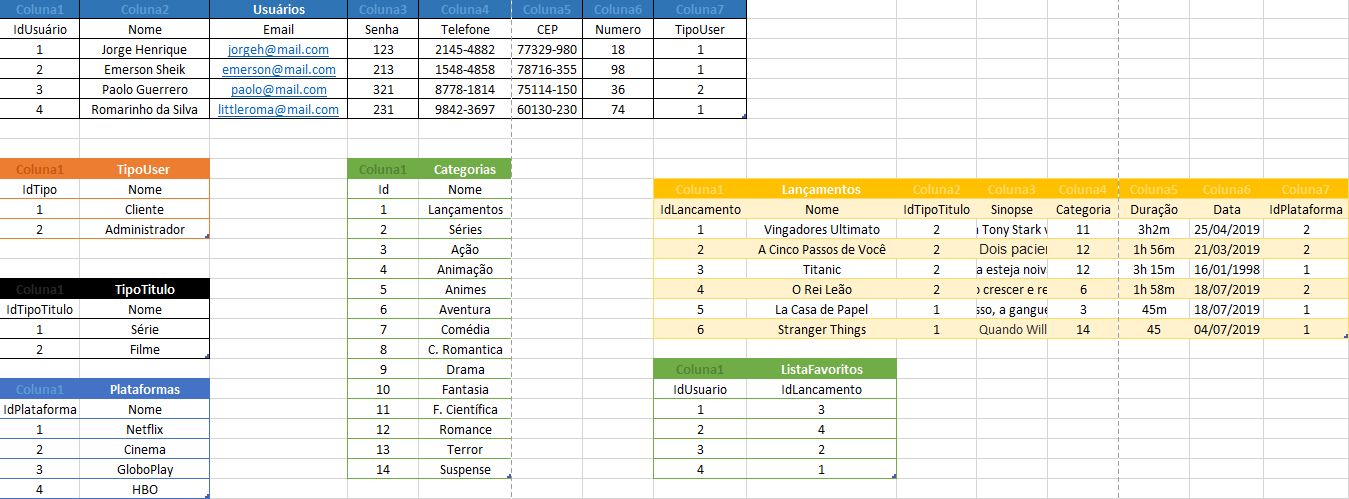
Modelo Lógico

Descreve como os dados serão armazenados no banco e também seus relacionamentos. Esse modelo adota alguma tecnologia, pode ser: relacional, orientado a objetos, orientado a colunas, entre outros.

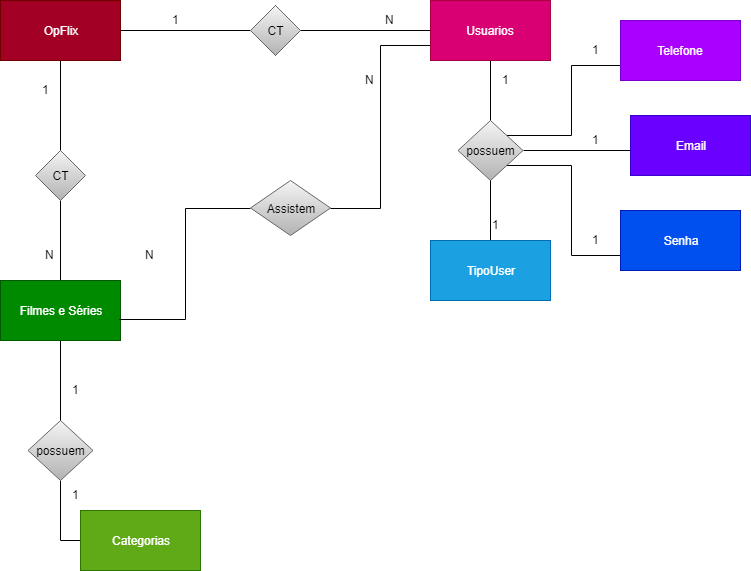


Modelo Físico

Descreve como os dados serão armazenados no banco e também seus relacionamentos. Esse modelo adota alguma tecnologia, pode ser: relacional, orientado a objetos, orientado a colunas, entre outros.



Modelo Conceitual

A finalidade do modelo conceitual de dados é capturar os **requisitos de informação** e **regras de negócio**sob o **ponto de vista do negócio**. Para isto, torna-se necessário o entendimento e a correta aplicação dos mecanismos de abstração, utilizados na modelagem conceitual de dados.

Cronograma

|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 | Dia 6 | Dia 7 | Dia 8 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Modelo Lógico** |  |  |  |  |  |  |  |  |
| **Modelo Físico** |  |  |  |  |  |  |  |  |
| Modelo Conceitual |  |  |  |  |  |  |  |  |

Funcionalidades

Web

1. O administrador poderá cadastrar qualquer tipo de usuário (administrador ou cliente);

2. O administrador poderá cadastrar categorias (contendo nome);

3. O administrador poderá cadastrar o lançamento de um filme/série (contendo título, uma pequena sinopse, a categoria vinculada, tempo de duração, e se é filme ou série e a data de lançamento);

4. O cliente poderá visualizar todos os lançamentos publicados;

Administrador

5. Cadastro de novos usuários (pode deixar um endpoint público para cadastro de novos clientes e um outro endpoint aonde administradores cadastram outros administradores);

6. Criar, Atualizar e Deletar Lançamentos;

7. Cadastrar/Listar/Atualizar Categorias;

8. Cadastrar/Listar/Atualizar Plataformas/Mídias;

Para demonstração dessas e outras Funcionalidades é necessário o aplicativo de apoio chamado Postman onde na pasta Postman do Projeto OpFlix será disponibilizado por meio de GitHub no repositório como o link

<https://github.com/crRyan21/2s2019-t2-sprint-2-backend/tree/master/Senai.Opflix.WebApi>

Exportar a coleção é testar as funcionalidades listadas acima.

O acesso ao Postman se dá por meio do download do software por meio do site oficial, depois pegar a coleção exportada na Git descrito acima para poder fazer os testes.

O deploy do projeto é necessariamente no aplicativo Visual Studio onde na pasta com os projetos correspondentes abrir o projeto Senai.OpFlix.WebApi e apertar o as teclas Ctrl + F5 para iniciar a aplicação.

A importação do projeto no Postman se da por meio de abrir o aplicativo, depois clicar no botão Import depois no botão Chose Files e só selecionar a coleção disponibilizada no Git.

Os pacotes do Nugget utilizados foram Swashbuckle.AspNetCore 4.0.1, System.Data.SqlClient 4.6.1, Microsoft.EntityFrameworkCore.SqlServer 2.1.11, Microsoft.EntityFrameworkCore.SqlServer.Design 1.1.6, Microsoft.EntityFrameworkCore.Tools 2.1.11, Microsoft.AspNetCore.Authentication.JwtBearer 2.1.1, System.IdentityModel.Tokens.Jwt 5.5.0.

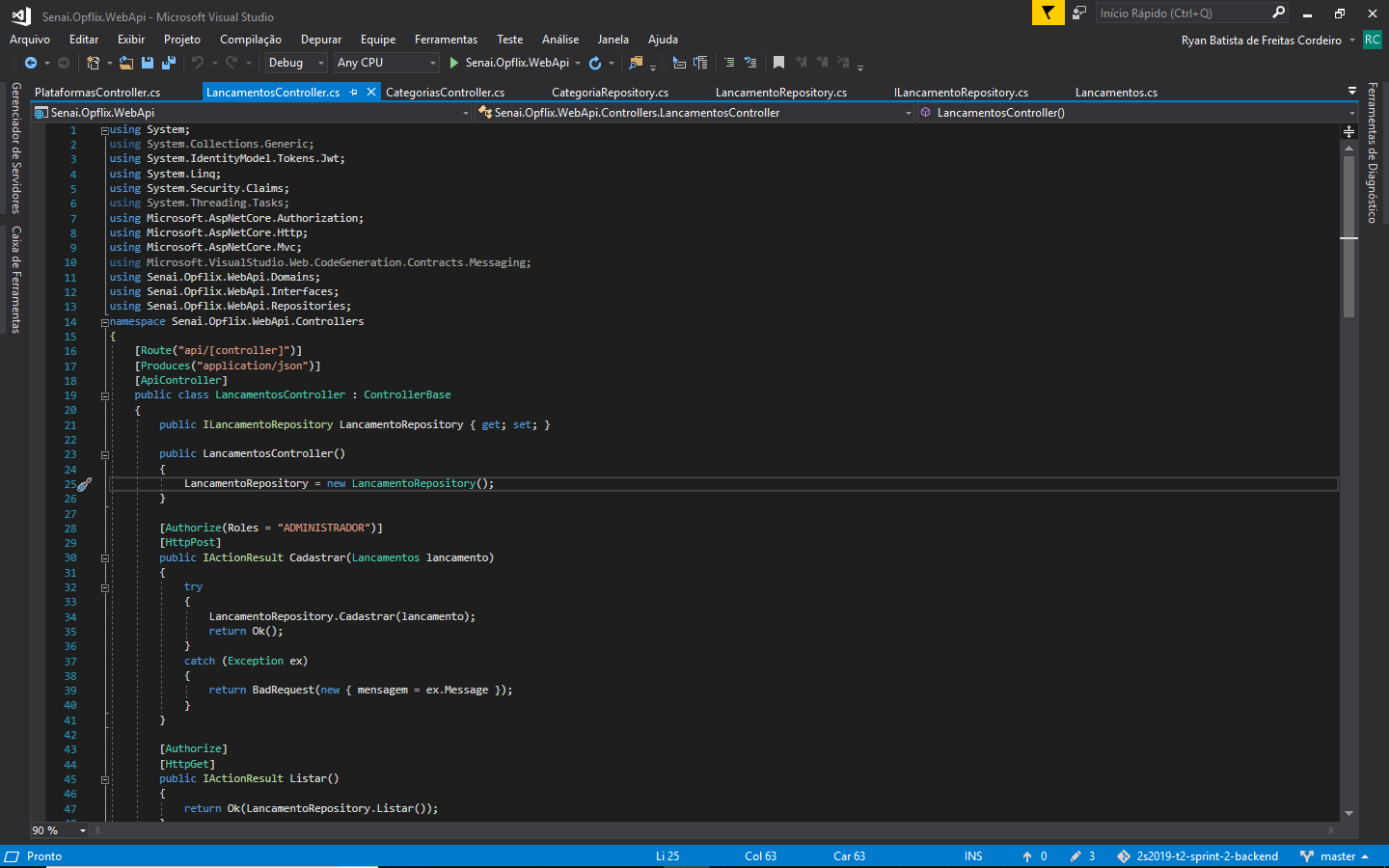
O projeto está arquitetado em um sistema Web API.

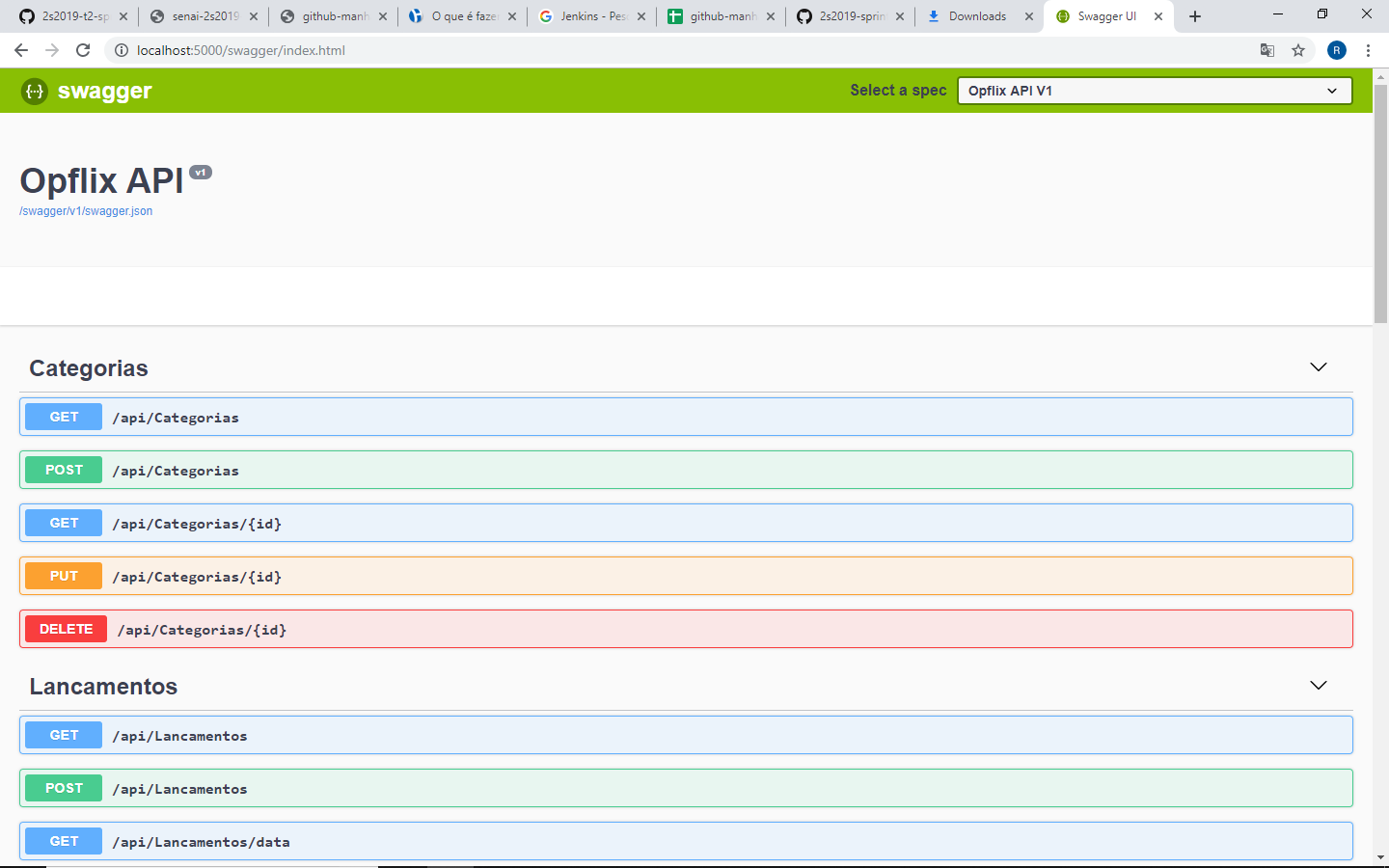
A criação do Banco de Dados se dá pela execução dos scripts enviados no SQL Server Management.

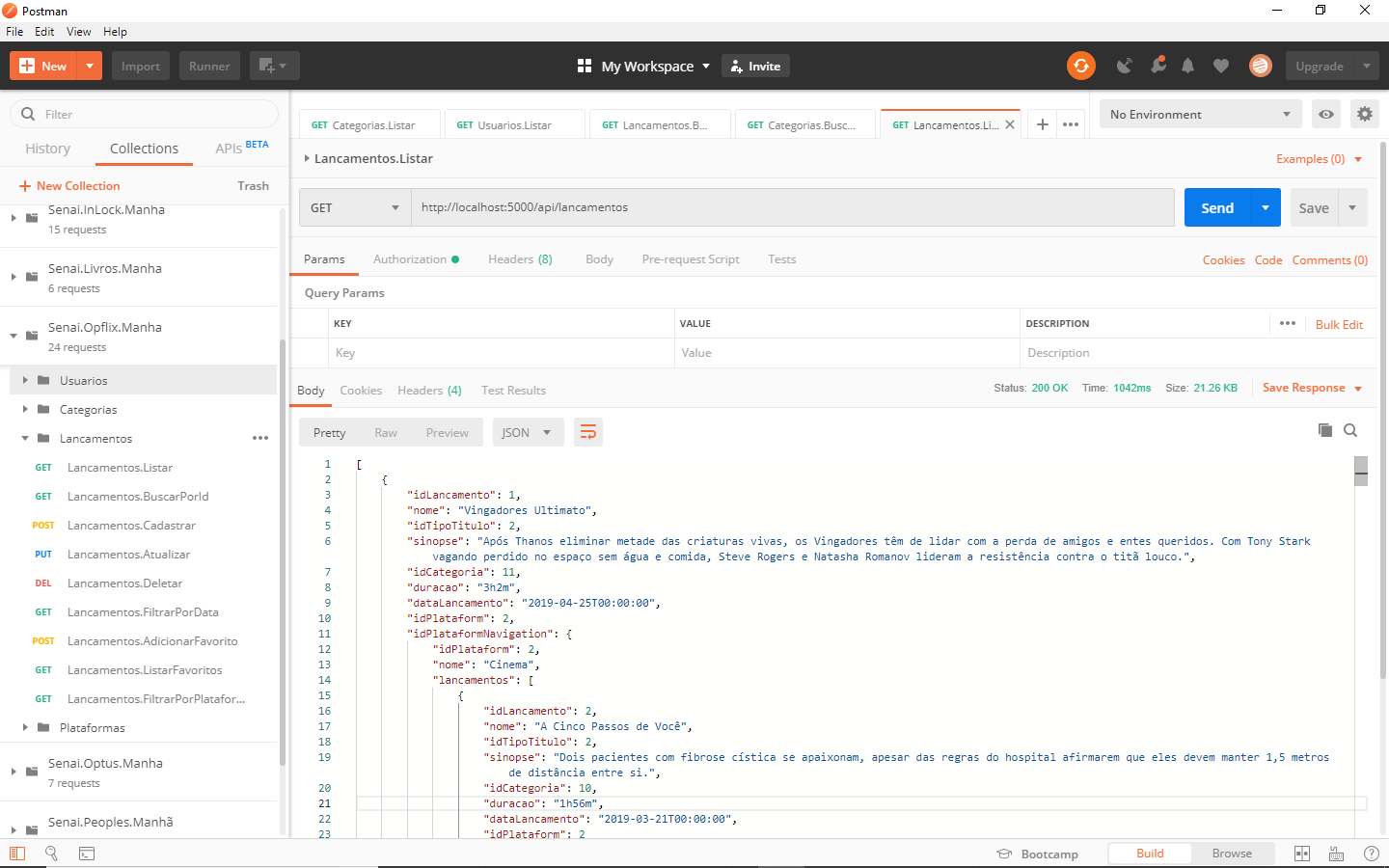
Mobile

Protótipos

Web







Mobile

Front-End

Mobile

Arquitetura do Projeto

Referências

Links

Livros