

Documentação

Op\_Flix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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

Demonstrar os modelos utilizados para a construção do banco de dados OpFlix e seu desenvolvimento

Descrição do projeto

A empresa OpFlix exigiu um banco de dados para armazenar seus dados, sendo eles seus usuários, títulos, suas respectivas categorias, tipos, plataformas e produtoras, para isso, foram feitas as modelagens para se tornar mais claro e objetiva a produção do banco de dados

Resumo do projeto

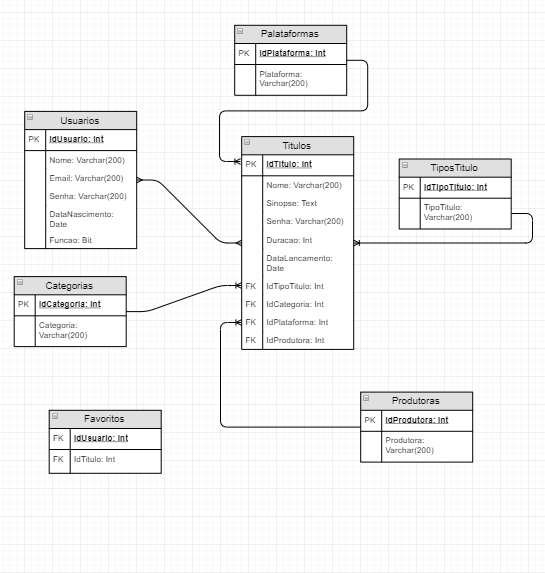
A Op\_Flix é uma empresa que exibe lançamentos de filmes, séries e documentários para seus usuários, que podem favoritar os títulos, e se informar de quando foi ou será lançado determinado título, com sua sinopse, e suas características

Modelagem de Software

A modelagem é essencial para a orientação ao decorrer do projeto, ela auxilia na visualização como um todo e torna muito mais prático, fácil e rápido a construção do banco, pois tudo já estará esquematizado e pronto para ser inserido

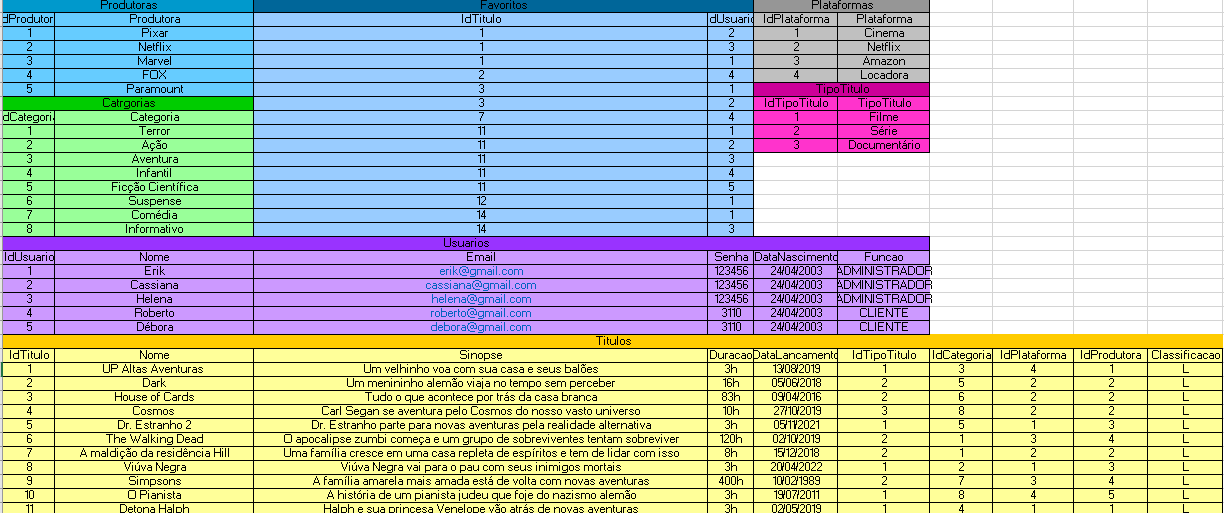
Modelo Lógico

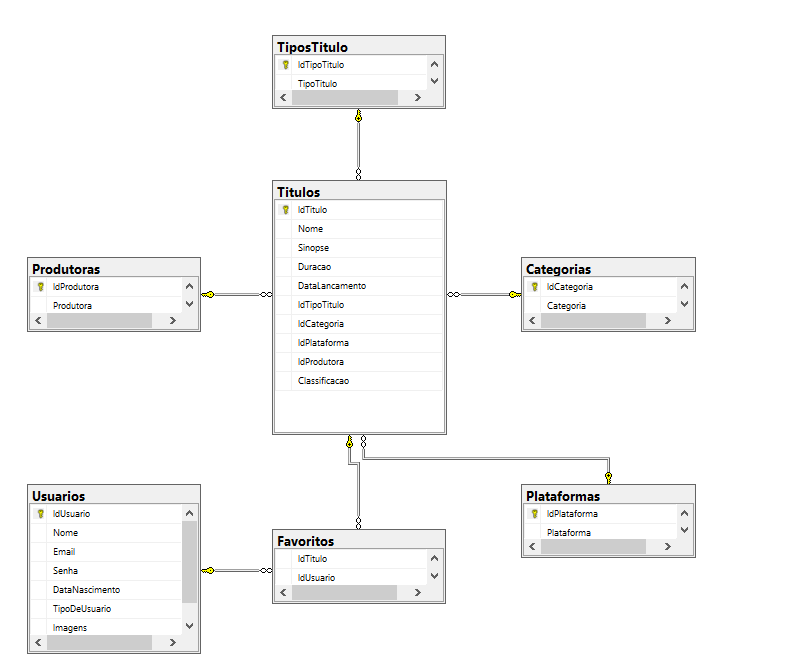
O modelo lógico é o que mais se aproxima do banco de dados, ele informa por meio de tabelas e relações, quais colunas (dados) existem seus tipos, e o tipo de chave de cada tipo, sendo uma chave primária ou uma chave estrangeira.



Modelo Físico

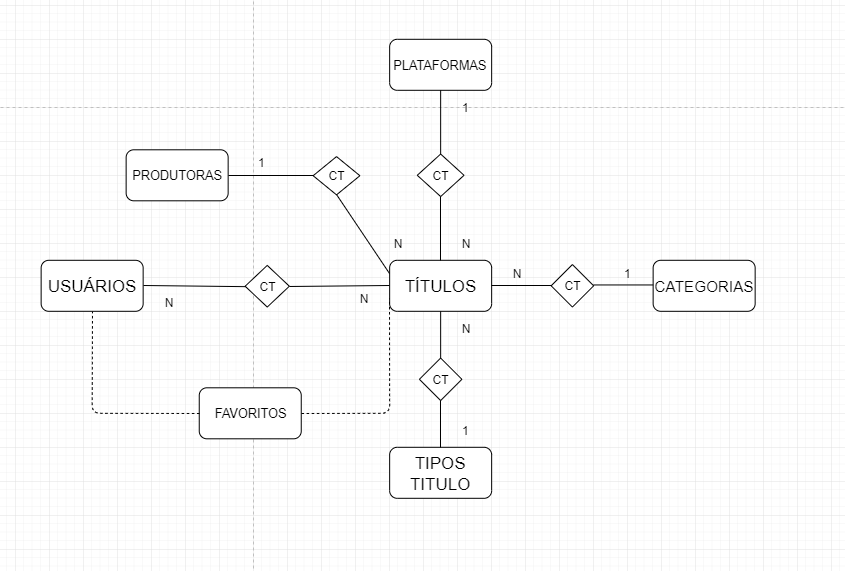
O modelo físico é a maneira de melhor visualização de dados, ele é composto por tabelas autoexplicativas que facilitam muito na associação de colunas e tuplas.





Modelo Conceitual

O modelo conceitual é especializado na percepção das cardinalidades das relações entre tabelas, ele indica se uma relação é de 1:1, 1:N ou N:N, facilitando na construção do banco.



Cronograma

|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 | Dia 6 | Dia 7 | Dia 8 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Modelo Lógico |  | ✔ |  |  |  |  |  |  |
| Modelo Físico | ✔ | ✔ |  |  |  |  |  |  |
| MODELO  CONCEITUAL | ✔ |  |  |  |  |  |  |  |
| DDL |  | ✔ |  |  |  |  |  |  |
| DML |  |  | ✔ |  |  |  |  |  |
| DQL |  |  | ✔ |  |  |  |  |  |
| DOCUMENTAÇÃO |  |  | ✔ |  |  |  |  |  |

Funcionalidades

Web – Nesta aplicação, poderá ser realizado um login (além do cadastro), sendo ele de um usuário ou um administrador, o usuário poderá listar os Títulos e suas características, assim como checar os filmes favoritos, enquanto o administrador será capaz de assim como o usuário, listar todas as entidades, e também a realização de adição de novos títulos, ou a remoção ou até uma atualização.

Mobile – Por se tratar de um programa realizado em API, o mesmo retorna um JSON, linguagem também utilizada em dispositivos mobile, assim, a aplicação é a mesma.

**Como acessar o Swagger – acesse o link:** <http://localhost:5000/swagger/index.html>

****

**Pacotes usados:**

-Swashbuckle.AspNetCore 4.0.1

-System.IdentityModel.Tokens.Jwt 5.5.0

-Microsoft.VisualStudio.Web.CodeGeneration.Design 2.1.9

-Microsoft.EntityFrameworkCore.SqlServer 2.1.11

-Microsoft.AspNetCore.Razor.Design 2.1.2

-Microsoft.AspNetCore.Authentication.JwtBearer 2.1.1

**Ferramentas usadas:**

-Microsoft SQL Server Management Studio 18

-Visual Studio 2017

-Postman

**Arquitetura do projeto –** Web API, Conceitos de MVC

**Postman**

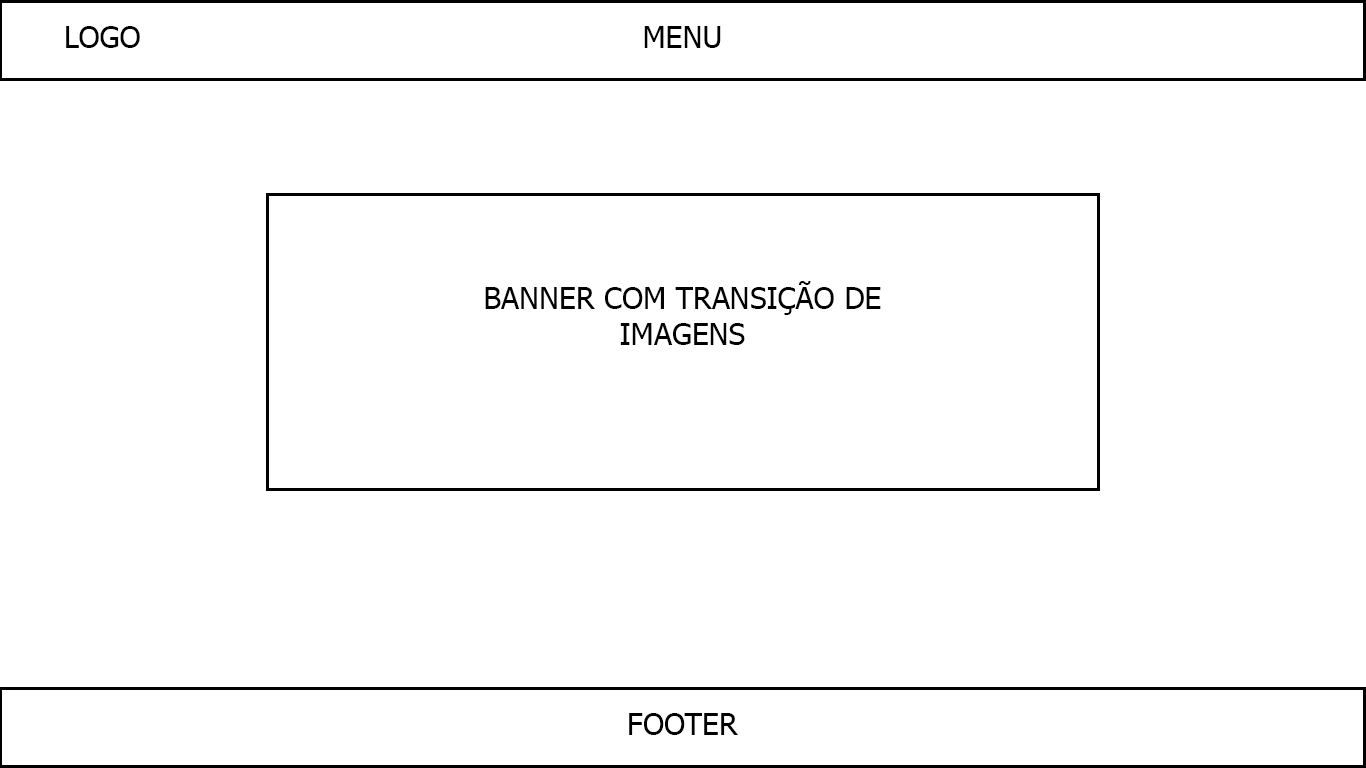
Para utilizar o Postman importando o projeto de outra máquina basta selecionar o botão “Import” e arrastar o arquivo

**Integração do Projeto**

A fim de criar o projeto, é necessário primeiramente a criação do banco de dados, no qual os comandos já estão prontos no arquivo “M\_01\_Diego\_DDL”, na pasta “BancoDeDados”, em seguida estabelecer a conexão com o backend, sendo com SqlClient ou Entity Framework..

Protótipos

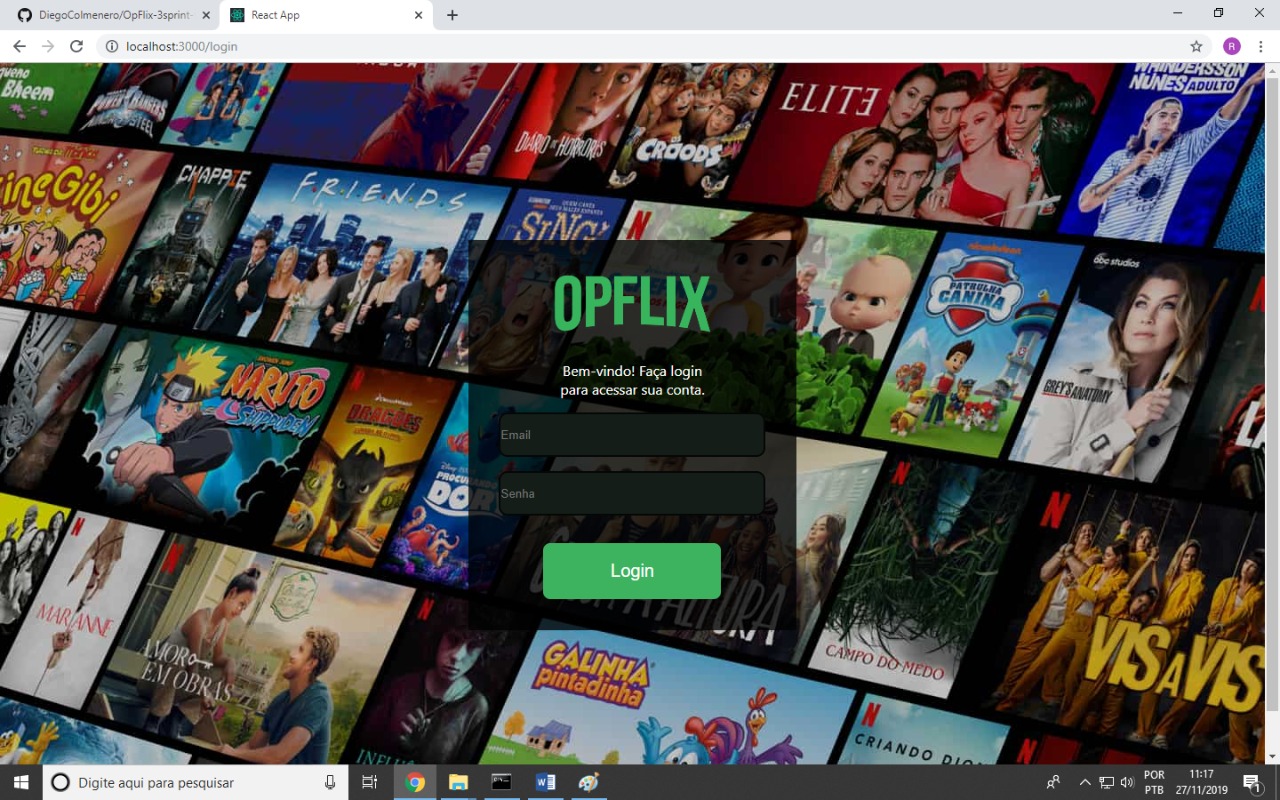
Web – Baixa Fidelidade



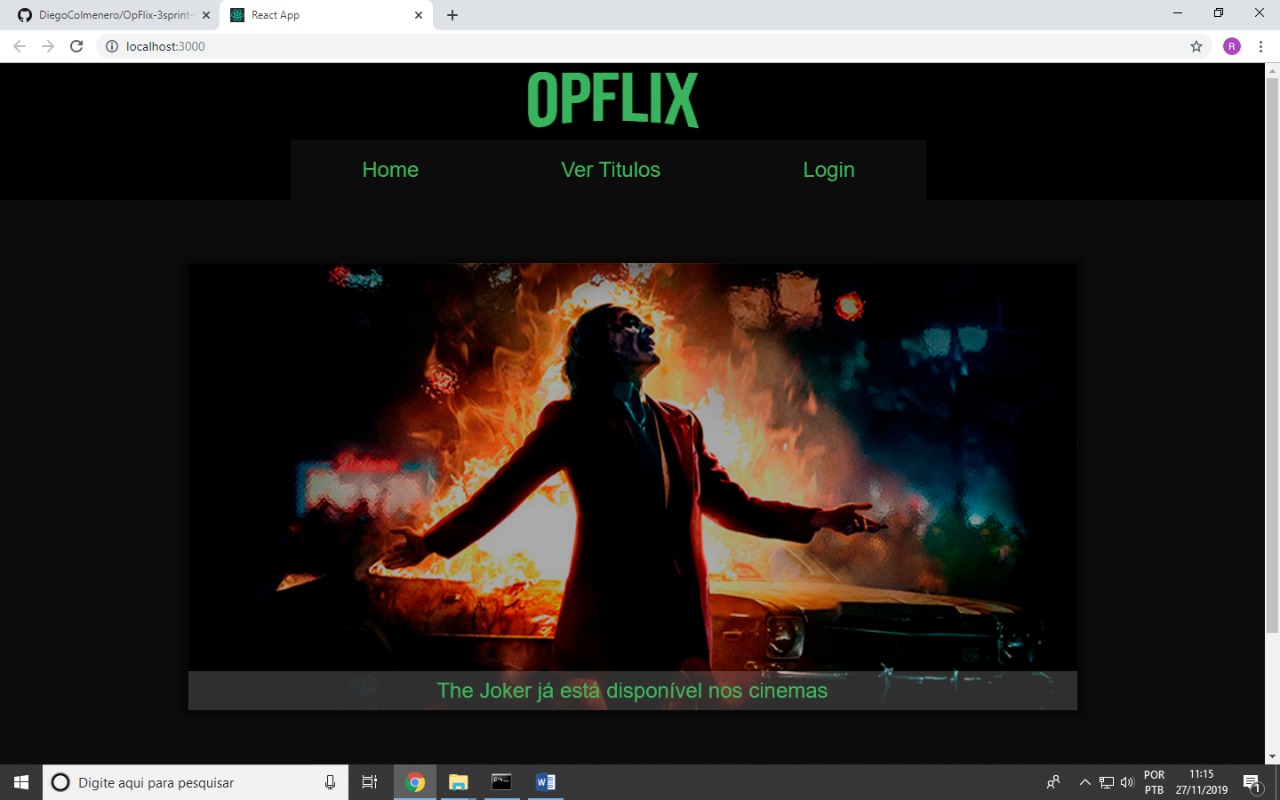
Mobile – Alta Fidelidade



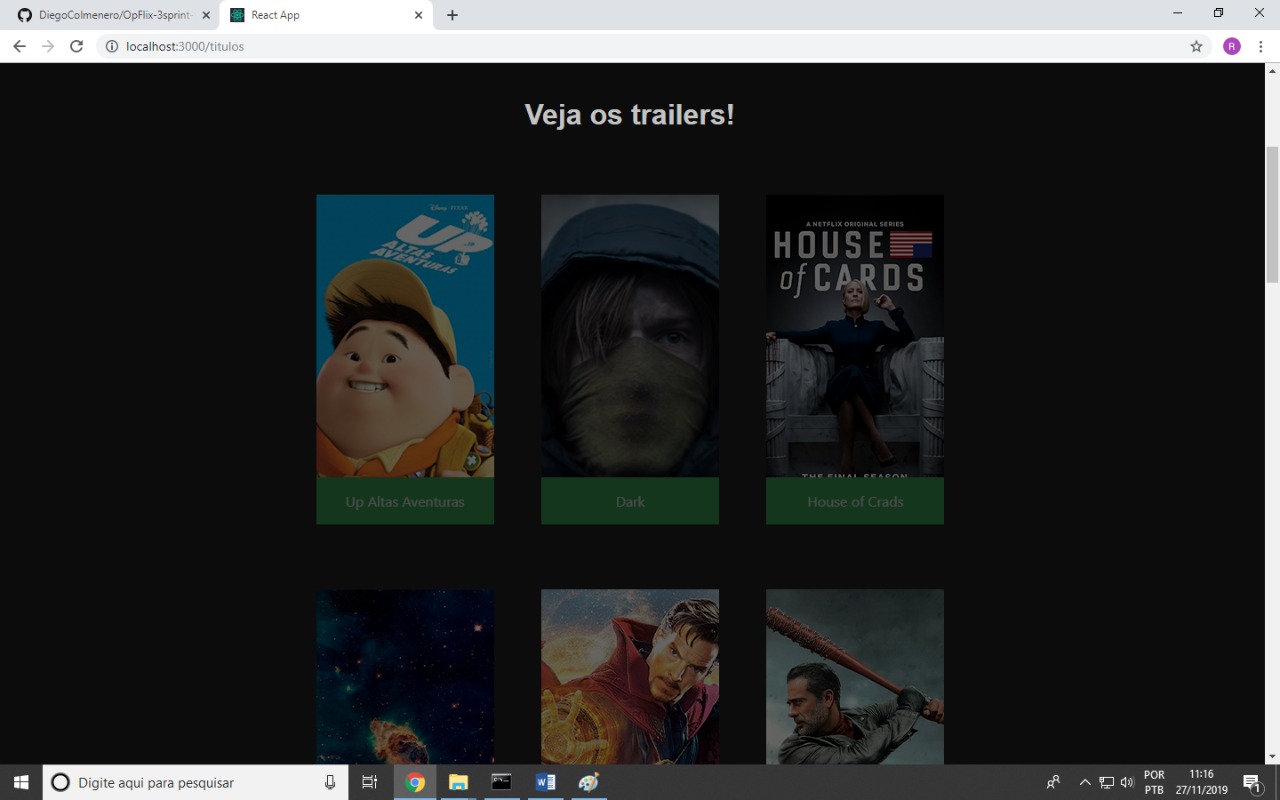
Front-End

O

LOGIN

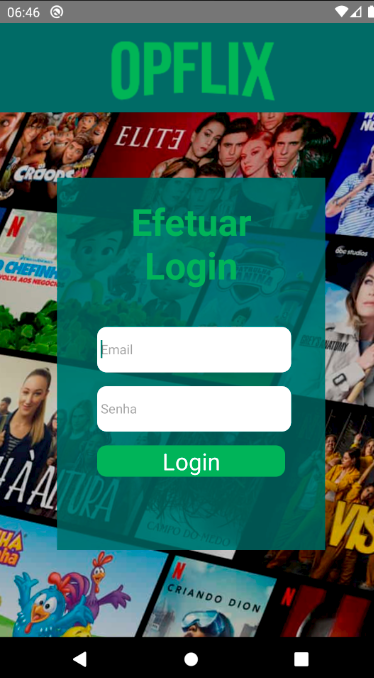


HOME



TÍTULOS - LANÇAMENTOS

Mobile



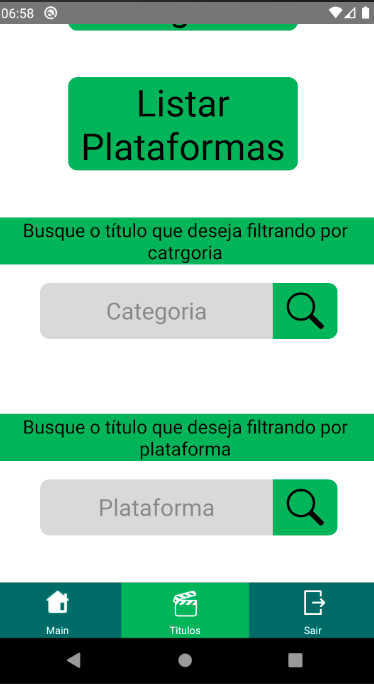
LOGIN



HOME



TÍTULOS – LANÇAMENTOS 1



TÍTULOS – LANÇAMENTOS 2

Arquitetura do Projeto

O projeto é composto por um banco de dados, uma API, e um sistema front-end Web e Mobile

Referências

Links

Livros