

Documentação

OpFlix

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

Sumário

1. [Resumo 3](#_Toc26956638)

[Objetivos](#_Toc26956639)

1. [Descrição do projeto 3](#_Toc26956640)

[Resumo do projeto](#_Toc26956641)

1. [Modelagem de Software 4](#_Toc26956642)

[Modelo Lógico](#_Toc26956643)

[Modelo Físico](#_Toc26956645)

[Modelo Conceitual](#_Toc26956646)

[Cronograma](#_Toc26956647)

1. [Funcionalidades 7](#_Toc26956648)

[Funcionalidades Web](#_Toc26956649)

[Funcionalidades Mobile](#_Toc26956664)

1. [Front-End 13](#_Toc26956677)
2. [Mobile 17](#_Toc26956678)
3. [Arquitetura do Projeto 18](#_Toc26956679)
4. [Referências 19](#_Toc26956680)

[Links](#_Toc26956681)

Resumo

Objetivos

Este documento tem como objetivo demonstrar todas as funcionalidades do aplicativo OpFlix, que visa monstrar os novos lançamentos de filmes e séries

Descrição do projeto

O OpFlix tem como objetivo mostrar os lançamentos de filmes e séries de mais sucesso no mundo.

Resumo do projeto

O Opflix exibe os lançamentos de filmes e séries mais relevantes no cenário cinematográfico mundial, além de exibir suas categorias e a plataforma em que estarão disponíveis. Assim, o usuário poderá filtrar os lançamentos pela sua plataforma disponível e/ou sua plataforma disponível. Além disso, o usuário poderá selecionar seus lançamentos favoritos, recebendo novidades sobre estes, além de poder listar seus lançamentos favoritos.

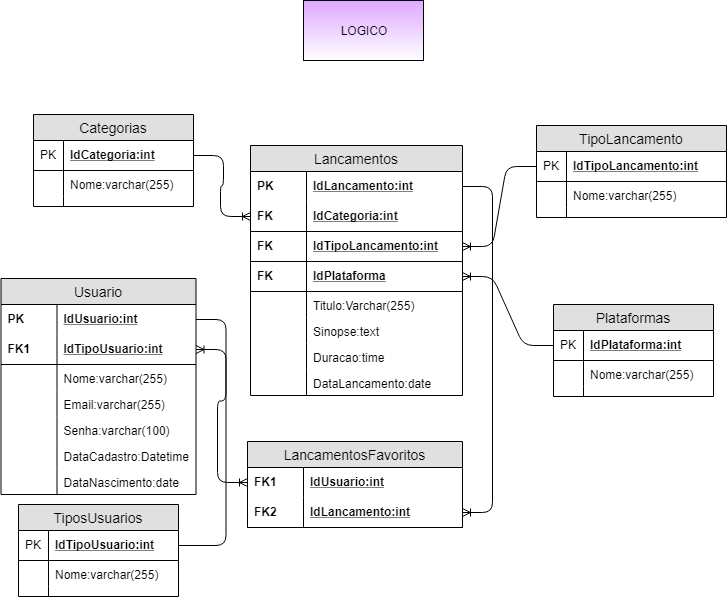
Modelagem de Software

A **modelagem de dados** é uma técnica usada para a especificação das regras de negócios e as estruturas de dados de um banco de dados. Ela faz parte do ciclo de desenvolvimento de um sistema de informação e é de vital importância para o bom resultado do projeto. Modelar dados consiste em desenhar o sistema de informações, concentrando-se nas entidades lógicas e nas dependências lógicas entre essas entidades.

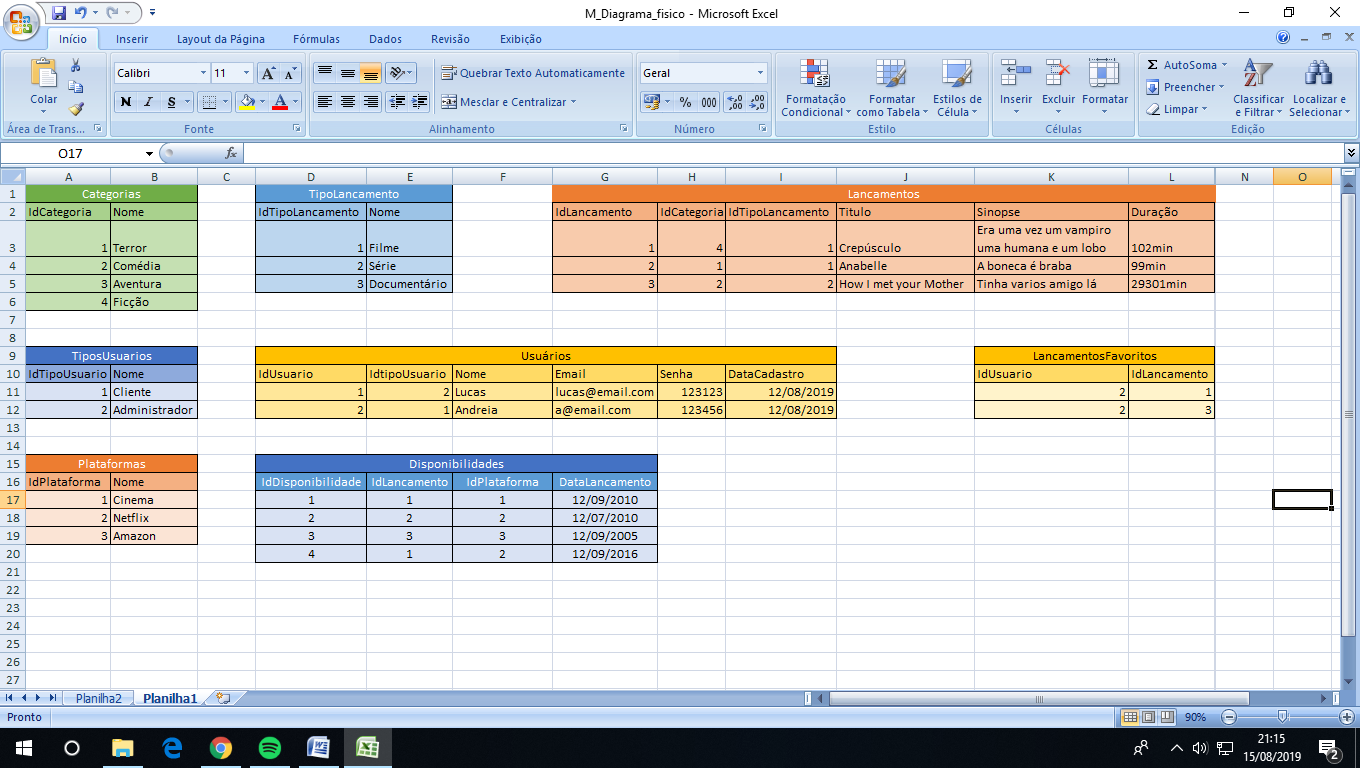
A modelagem de dados está dividida em:

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.

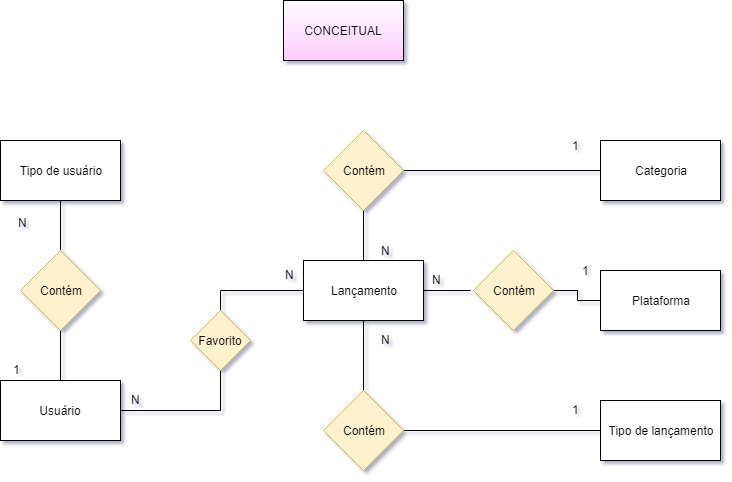


Modelo Físico

Descreve como será feita a armazenagem de dados no banco.****

Modelo Conceitual

A modelagem conceitual basea-se no mais alto nível e deve ser usada para envolver o cliente, pois o foco aqui é discutir os aspectos do negócio do cliente e não da tecnologia. Os exemplos de modelagem de dados vistos pelo modelo conceitual são mais fáceis de compreender, já que não há limitações ou aplicação de tecnologia específica. O diagrama de dados que deve ser construído aqui é o Diagrama de Entidade e Relacionamento, onde deverão ser identificados todas as entidades e os relacionamentos entre elas. Este diagrama é a chave para a compreensão do modelo conceitual de dados.



Cronograma

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

Funcionalidades

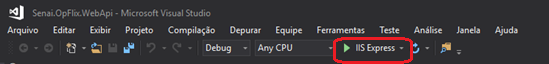
* **Como executar a aplicação**

**Banco de dados**

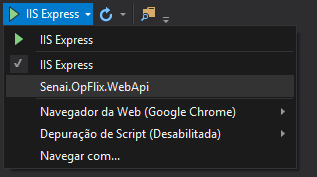
* Vá até o repositório 2s2019-t2-sprint-2-opflix (onde se encontra esse documento);
* Abra o documento M\_Lucas\_DDL utilizando o programa **SQL SERVER STUDIO MANAGEMENT**;
* Aperte **F5** ou clique em **EXECUTAR;**
* Após isso, abra o documento M\_Lucas\_DML utilizando o mesmo programa;
* Aperte **F5** ou clique em **EXECUTAR.**

**BackEnd**

* Vá até o repositório 2s2019-t2-sprint-2-opflix (onde se encontra esse documento);
* Abra a pasta Senai.OpFlix.WebApi;
* Abra o documento Senai.OpFlix.WebApi.sln com o aplicativo Visual Studio;
* Na parte superior da tela, você encontrará um símbolo de “play” na cor verde, como indicado na imagem abaixo:



* Troque a opção IIS Express pela opção Senai.OpFlix.WebApi, como mostra a imagem abaixo:



* Com essa opção selecionada, clique no ícone **e execute o programa**.

**Swagger**

Utilizando o **Swagger**, é possível observar e aplicar todas as funcionalidades presentes na aplicação. Veja aqui como:

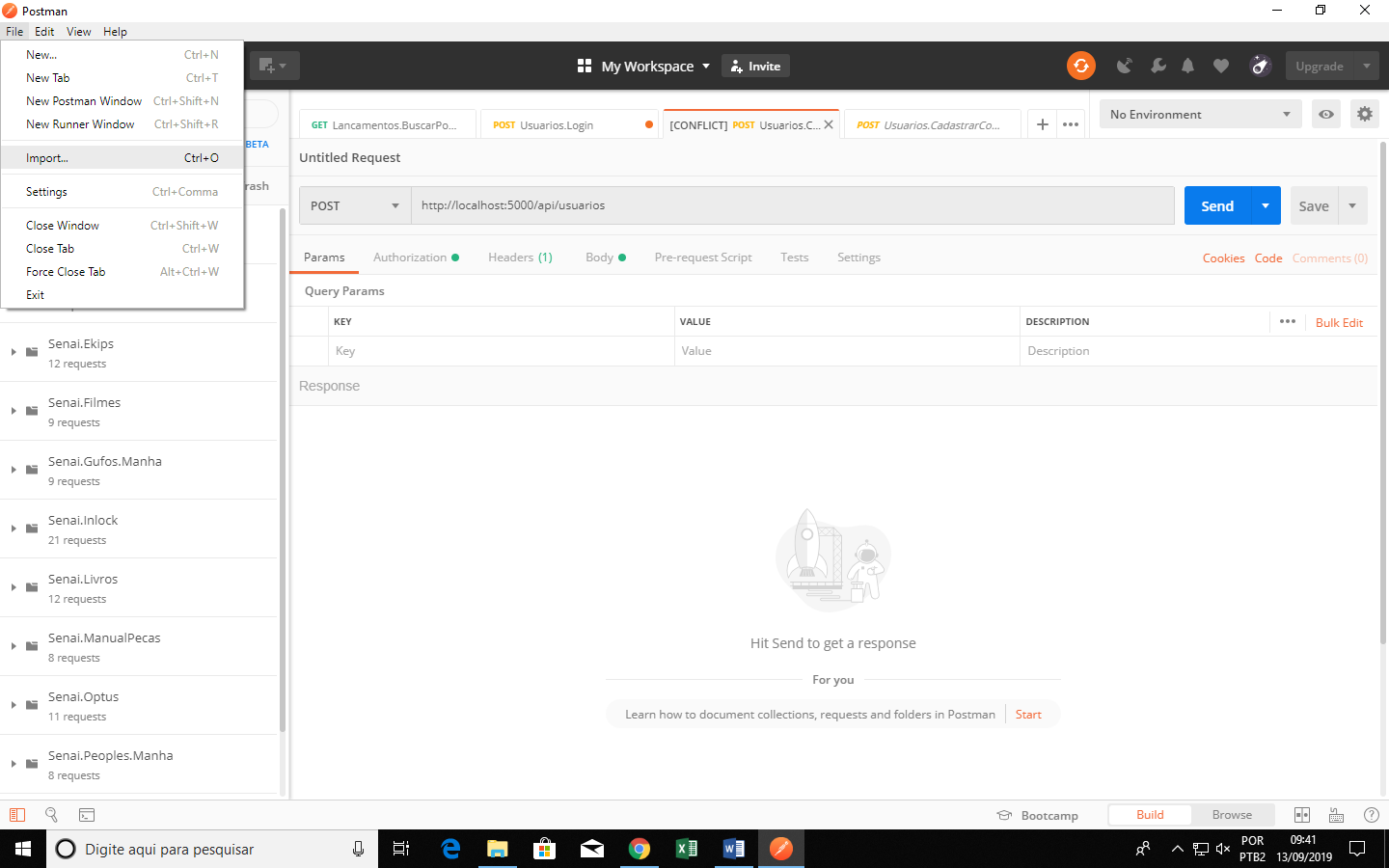
* Execute a aplicação;
* Em seu navegador de preferência, digite a seguinte URL: <http://localhost:5000/swagger>;

Lá, você encontrará todas as funcionalidades presentes nessa aplicação. Essas estarão presentes com os seguintes verbos (em inglês): **GET, POST, PUT** e **DELETE.** Esses servem, respectivamente, para obter, cadastrar, atualizar e excluir determinada informação.

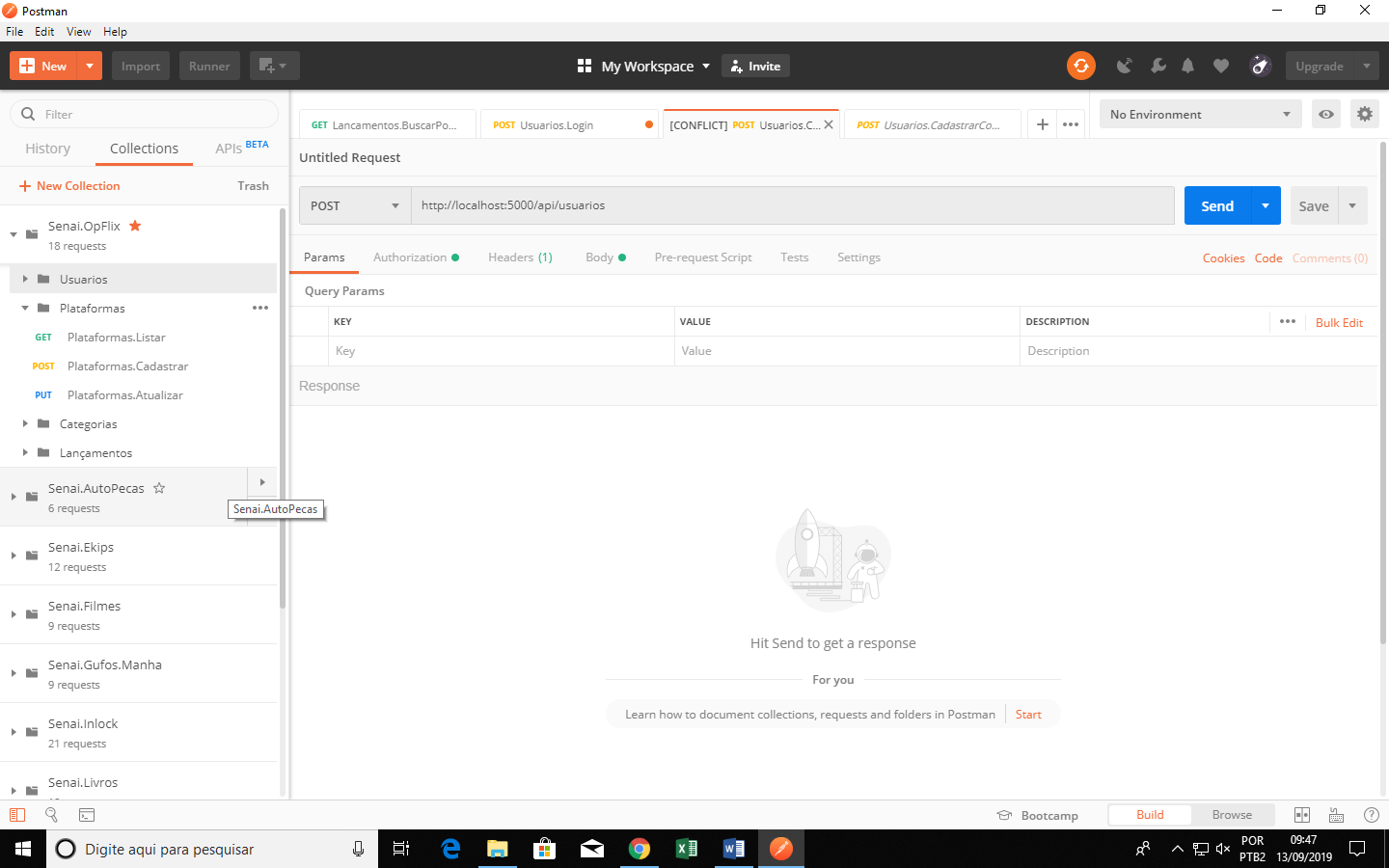
**Postman**

Utilizando o **Postman**, podemos aplicar cada ação presente em nosso projeto. Para importar a coleção Postman para sua máquina e utilizá-la, siga os seguintes passos:

* Abra o aplicativo **Postman**;
* No canto superior esquerdo, clique em **File** e em seguida clique em **Import,** como mostra a imagem abaixo**:**



* Em **Import File,** selecione **Choose Files** e selecione o documento Senai.OpFlix.Postman, presente no repositório 2s2019-t2-sprint-2-opflix (onde se encontra esse documento) ou arraste o mesmo documento ao campo;
* Dentro da coleção, você encontrará pastas com as ações respectivas (com os verbos **GET, POST, PUT** e **DELETE,** já explicados anteriormente) de seus nomes, como mostra a imagem abaixo:



* **Pacotes NuGet**

Os pacotes NuGet (e suas versões) utilizados nesse projeto foram esses:

* Microsoft.AspNetCore.App (2.1.1);
* Microsoft.AspNetCore.Authentication.JwtBearer (2.1.1);
* Microsoft.AspNetCore.Razor.Design (2.1.2);
* Microsoft.EntityFrameworkCore.SqlServer (2.1.11);
* Microsoft.EntityFrameworkCore.SqlServer.Design (1.1.6);
* Microsoft.EntityFrameworkCore.Tools (2.1.11);
* Microsoft.NETCore.App (2.1.0);
* Microsoft.VisualStudio.Web.CodeGeneration.Design (2.1.9);
* Swashbuckle.AspNetCore (4.0.1);
* System.IdentityModel.Tokens.Jwt (5.5.0).

Funcionalidades Web

Funcionalidades de usuário comum:

* Categorias:
  + Listar.
* Plataformas:
  + Listar.
* Tipos de lançamento:
  + Listar.
* Usuários:
  + Cadastrar;
  + Fazer Login.
* Lançamentos:
  + Listar;
  + Filtrar por categoria;
  + Filtrar por plataforma;
  + Buscar pelo nome;
  + Favoritar;
  + Desfavoritar.
* Localizações dos lançamentos:
  + Listar.

Funcionalidades de usuário administrador:

* Categorias:
  + Listar;
  + Cadastrar;
  + Atualizar.
* Plataformas:
  + Listar;
  + Cadastrar;
  + Atualizar.
* Tipos de lançamento:
  + Listar.
* Usuários:
  + Cadastrar;
  + Cadastrar administrador;
  + Fazer Login.
* Lançamentos:
  + Listar;
  + Filtrar por categoria;
  + Filtrar por plataforma;
  + Buscar pelo nome;
  + Favoritar;
  + Desfavoritar.
  + Cadastrar;
  + Atualizar;
  + Excluir.
* Localizações dos lançamentos:
  + Listar;
  + Cadastrar;
  + Excluir

Funcionalidades Mobile

Funcionalidades de usuário comum:

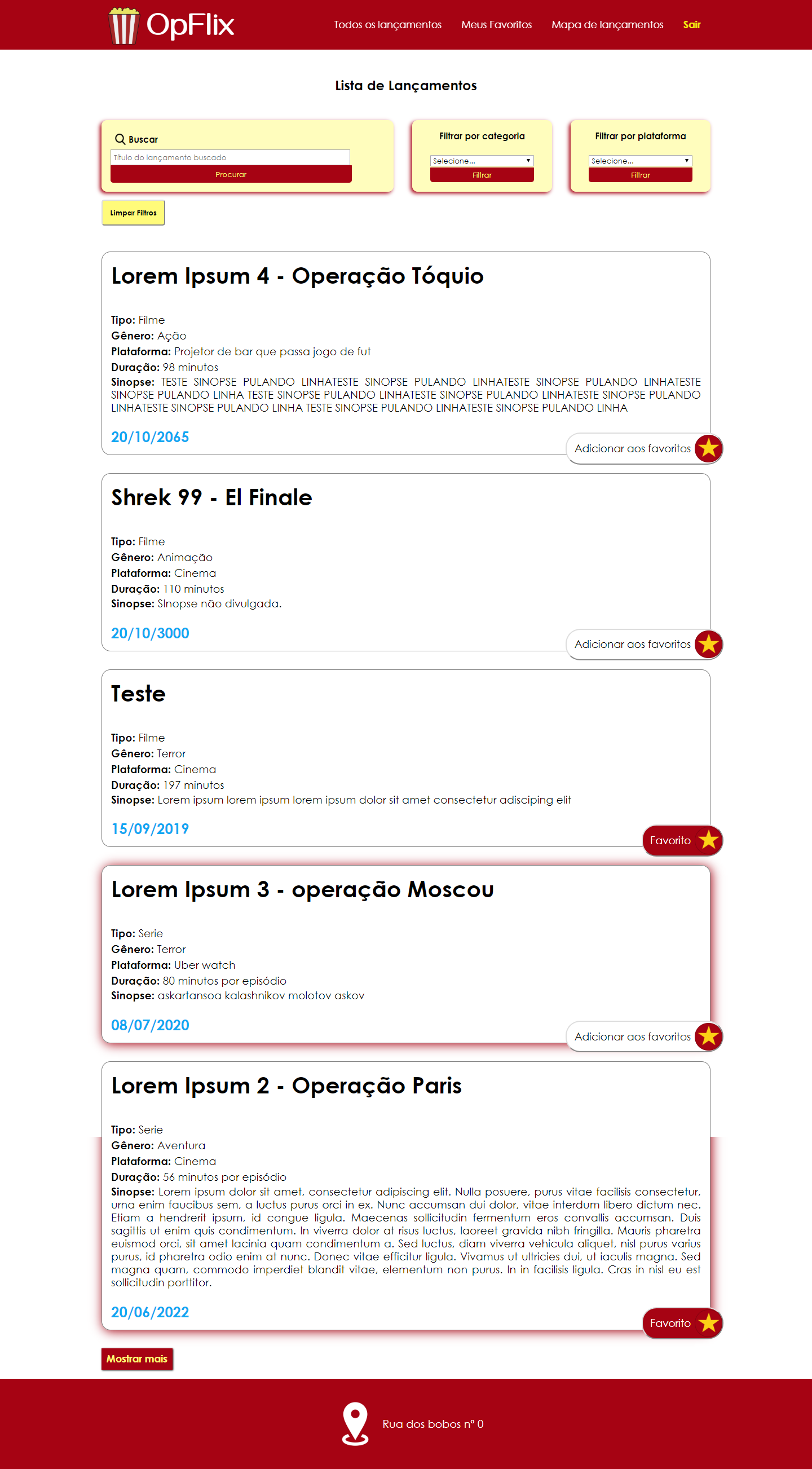
* Categorias:
  + Listar.
* Plataformas:
  + Listar.
* Tipos de lançamento:
  + Listar.
* Usuários:
  + Cadastrar;
  + Fazer Login.
* Lançamentos:
  + Listar;
  + Favoritar;
  + Desfavoritar.

Funcionalidades de usuário administrador:

* Categorias:
  + Listar;
  + Cadastrar;
  + Atualizar.
* Plataformas:
  + Listar;
  + Cadastrar;
  + Atualizar.
* Tipos de lançamento:
  + Listar.
* Usuários:
  + Cadastrar;
  + Cadastrar administrador;
  + Fazer Login.
* Lançamentos:
  + Listar;
  + Favoritar;
  + Desfavoritar.
  + Cadastrar;
  + Atualizar;
  + Excluir.

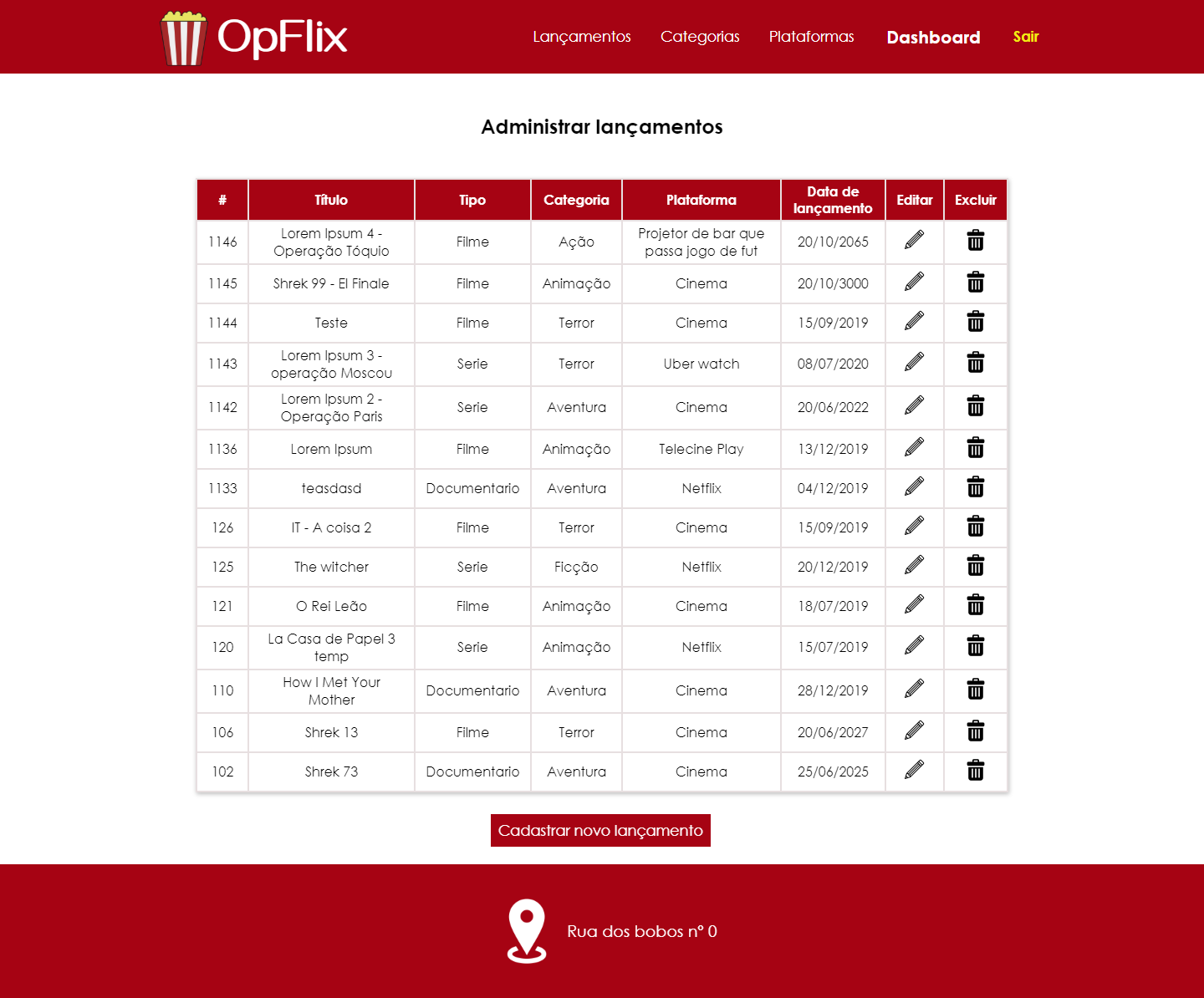
Front-End

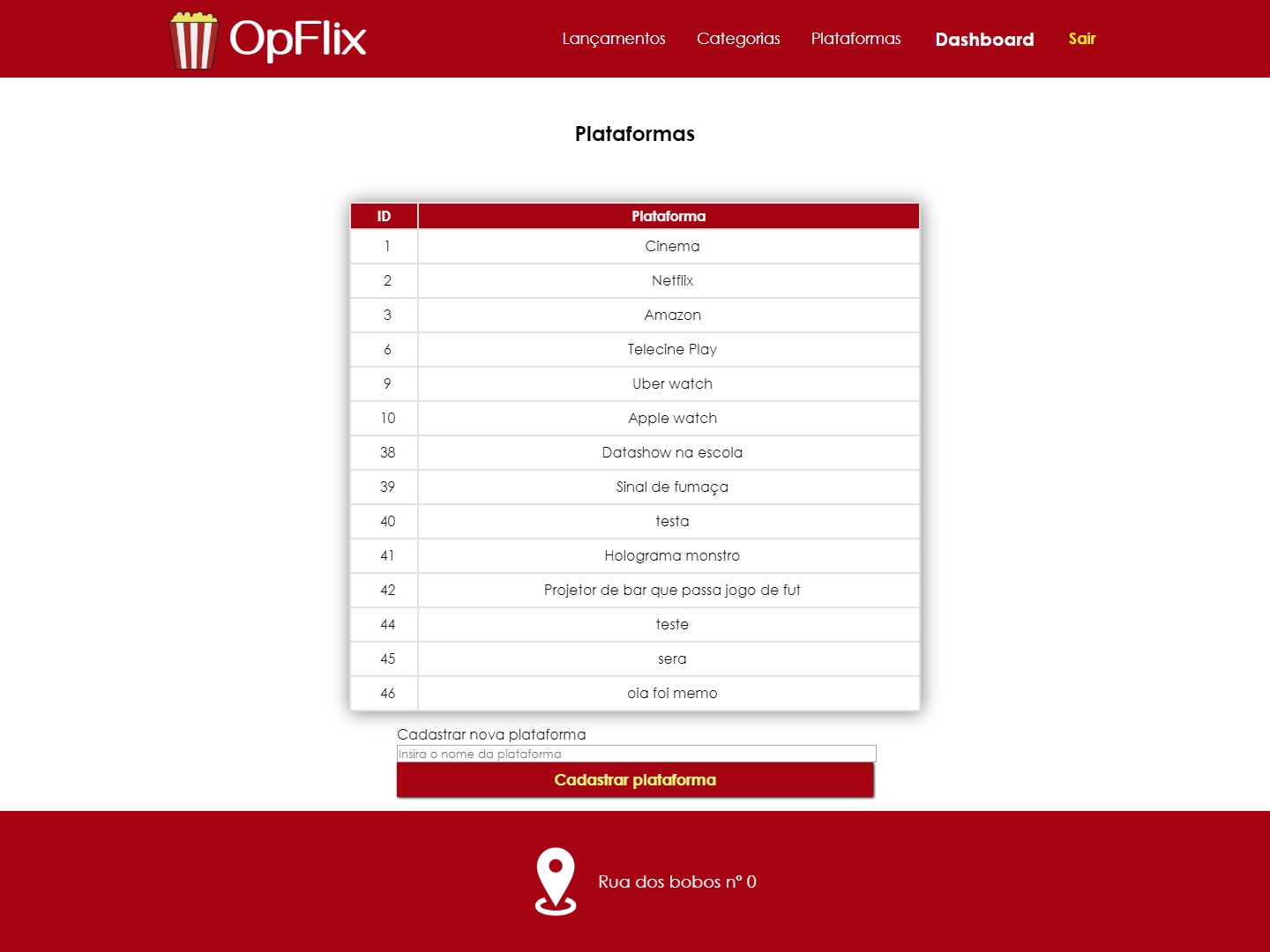




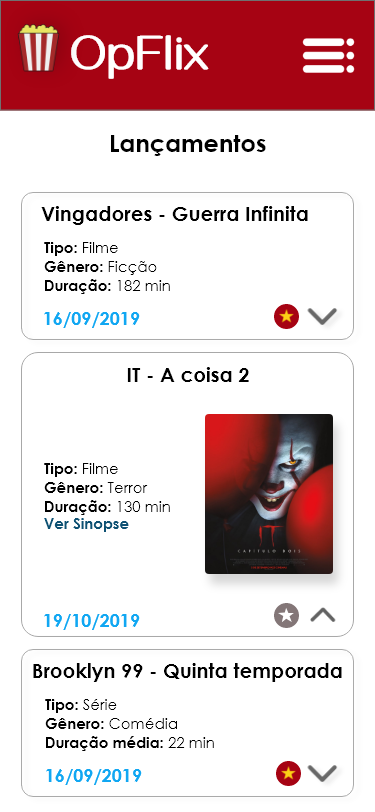
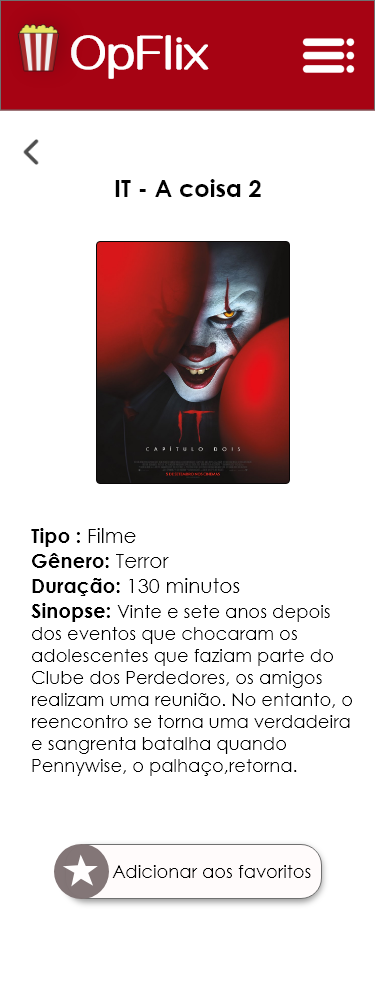




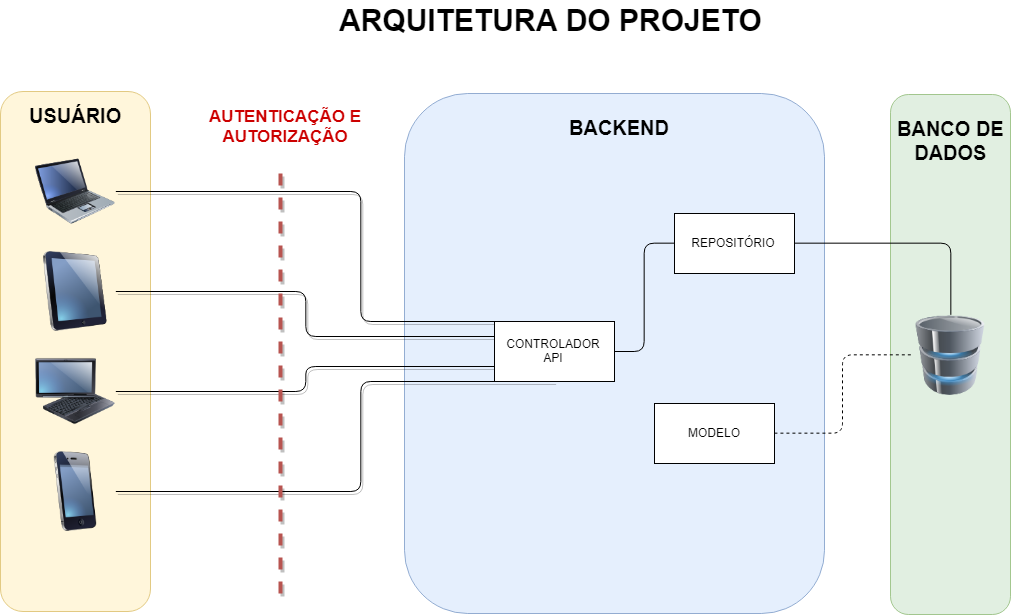




Mobile

Arquitetura do Projeto



Referências

Links

* <http://spaceprogrammer.com/bd/introducao-ao-modelo-de-dados-e-seus-niveis-de-abstracao/>
* <https://www.luis.blog.br/modelagem-de-dados-modelo-conceitual-modelo-logico-e-fisico.html>