,,

Documentação

Ecommerce App (nome não definido)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gabriel Rodrigues Amador, São Paulo/SP |  |  |  | 2020 |

Sumário

1. [Resumo 3](#_Toc27402850)

[Objetivos](#_Toc27402851)

[Descrição](#_Toc27402853) do Projeto

1. [Ferramentas utilizadas 3](#_Toc27402854)
2. [Modelagem de Software 3](#_Toc27402855)

[Modelo Lógico](#_Toc27402856)

[Modelo Físico](#_Toc27402857)

[Modelo Conceitual](#_Toc27402858)

1. [Deploy da Aplicação 6](#_Toc27402863)
2. [Swagger 7](#_Toc27402864)

[Acessando](#_Toc27402866)

1. [Pacotes Nuget 9](#_Toc27402867)

[Pacotes Necessários](#_Toc27402868)

1. [Postman 11](#_Toc27402870)

[Importando](#_Toc27402872)

1. [Funcionalidades 14](#_Toc27402873)

[Web](#_Toc27402874)

[Mobile](#_Toc27402875)

1. [Links 24](#_Toc27402887)

Resumo

Objetivos

Descrição do Projeto

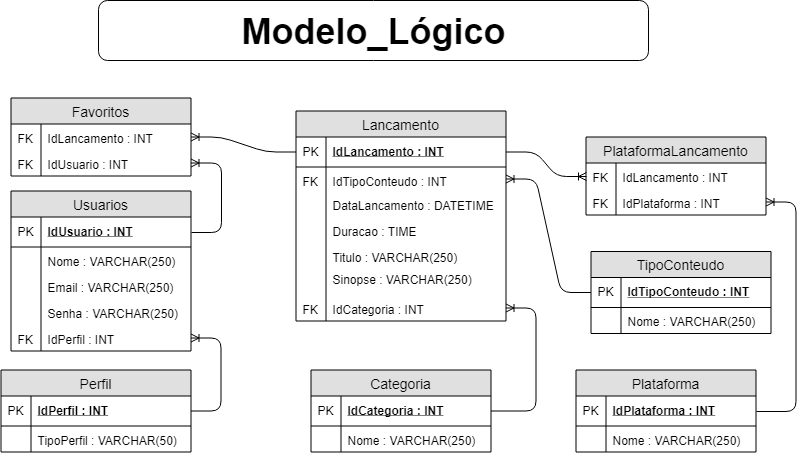
Ferramentas utilizadas

Modelagem de Software

As modelagens de software é uma etapa para o planejamento e minimização do tempo de projeto, assim as modelagens ajudam na construção do banco de dados e facilitam o entendimento do armazenamento de dados, assim como traz uma visão geral das entidades e os relacionamentos entre elas. As modelagens seguem representadas abaixo, sendo elas Lógica, Física e Conceitual.

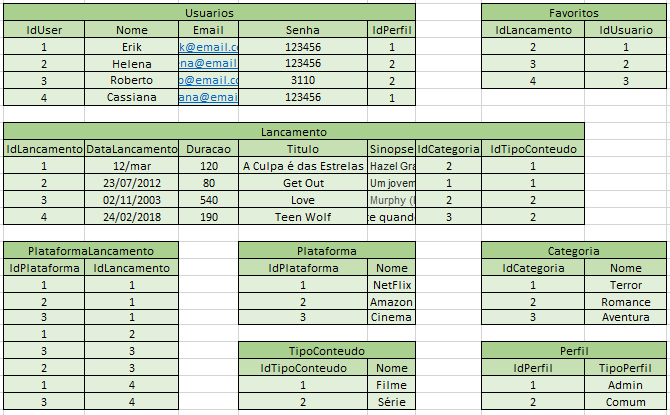
Modelo Lógico

A modelagem Lógica é aquela que traz não só as entidades, mas também seus dados e tipos de dados. Assim ela se assemelha ao resultado final do código do Banco de Dados, e mostra também as relações entre as tabelas e quais seus tipos.



Modelo Físico

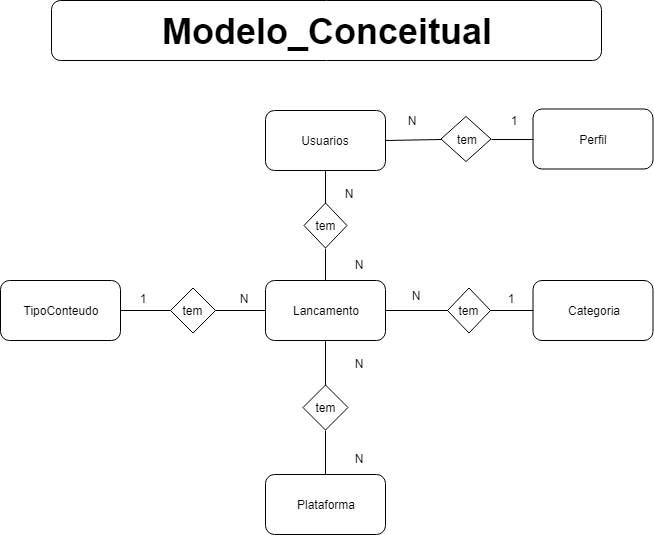
O modelo físico modelado na plataforma Excel observamos em si os dados que possuem na tabela, nele podemos administrar também as relações observando od Id´s de cada tabela. Semelhante ao Lógico porem nesta representação o foco são nos dados em si, e não em relações de entidades.



Modelo Conceitual

A conceitual é aquela em que focamos nas entidades presentes, e vemos um resumo de como se relacionam através dos caracteres 1 e N. Ela serve como uma fonte fundamental de consulta, pois resume para o código qual entidade que carrega uma FK (Chave de relacionamento entre tabelas) referencial a uma outra tabela

.



Deploy

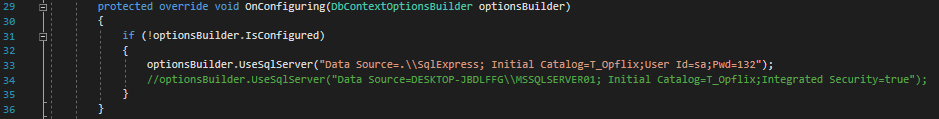
1º - Abra o executável “Visual Studio” já instalado na máquina;

2º - No diretório de pastas encontre o projeto “2s2019-sprint-1-bd-opflix” (repositório clonado do GitHub) e acesse: “2s2019-sprint-1-bd-opflix\BackEnd - API\Senai.OpFlix.WebApi”;

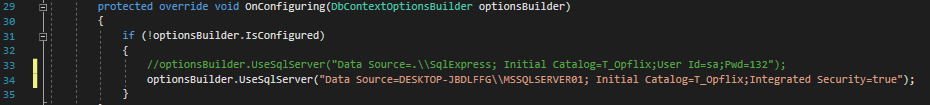
3º - Com um duplo clique no mouse abra o projeto “Senai.OpFlix.WebApi.sln” na plataforma mencionada;

4º - Com o projeto aberto, em **Gerenciador de soluções** acesse a pasta “**Contexts”** e abra o arquivo “**OpflixContext”,** você deve editar as linhas 33 e 34 de modo que as barras sejam editadas**:**

**Antes:**



**Depois:**



**5º Agora na pasta Repositories, acesse o arquivo “LancamentoRepository” e altere as linhas 15 e 16, de modo que:**

**Antes:**



**Depois:**

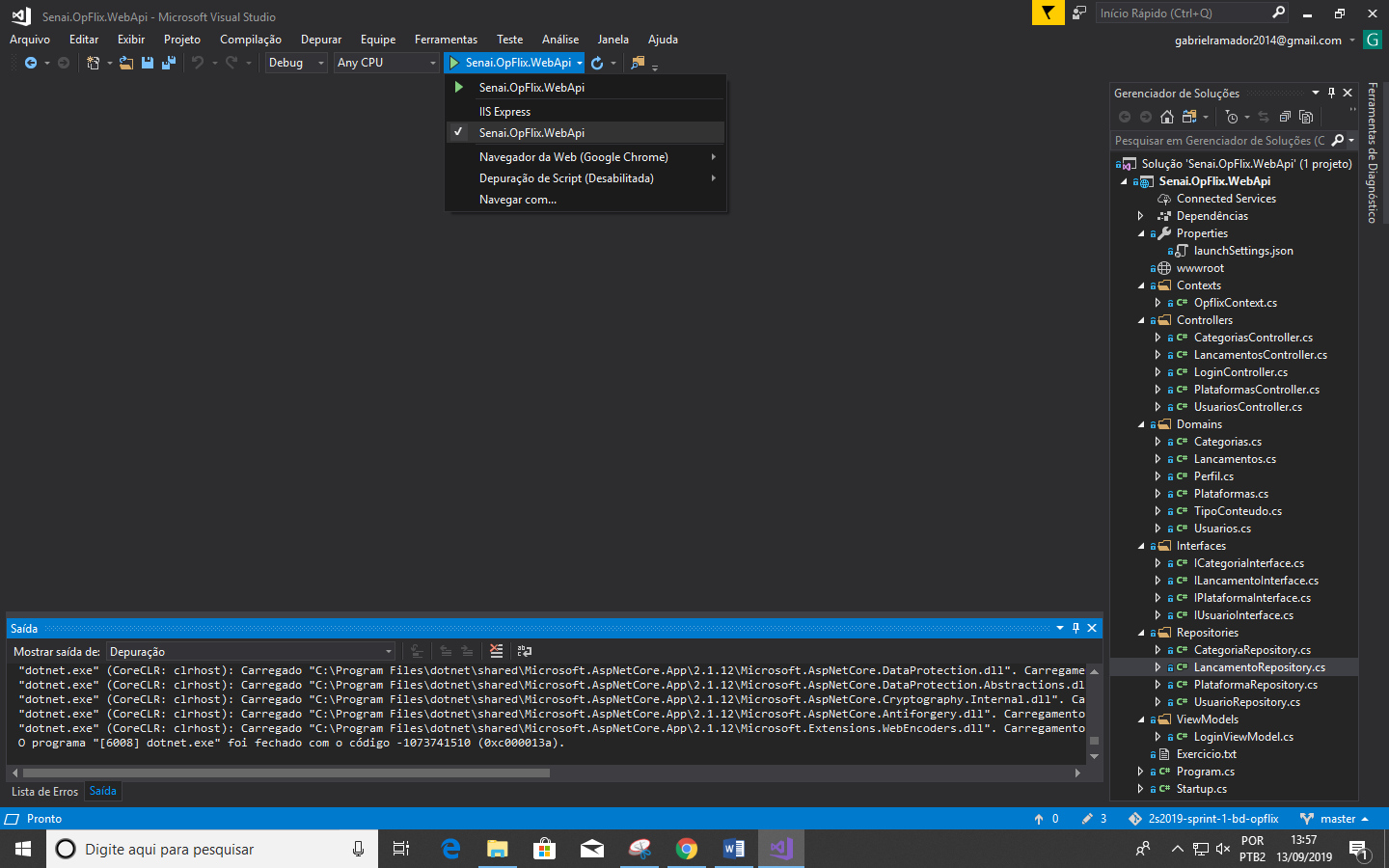


**6**º - Enfim, pressione a tecla F5 ou F11 e o projeto é executado.

Swagger

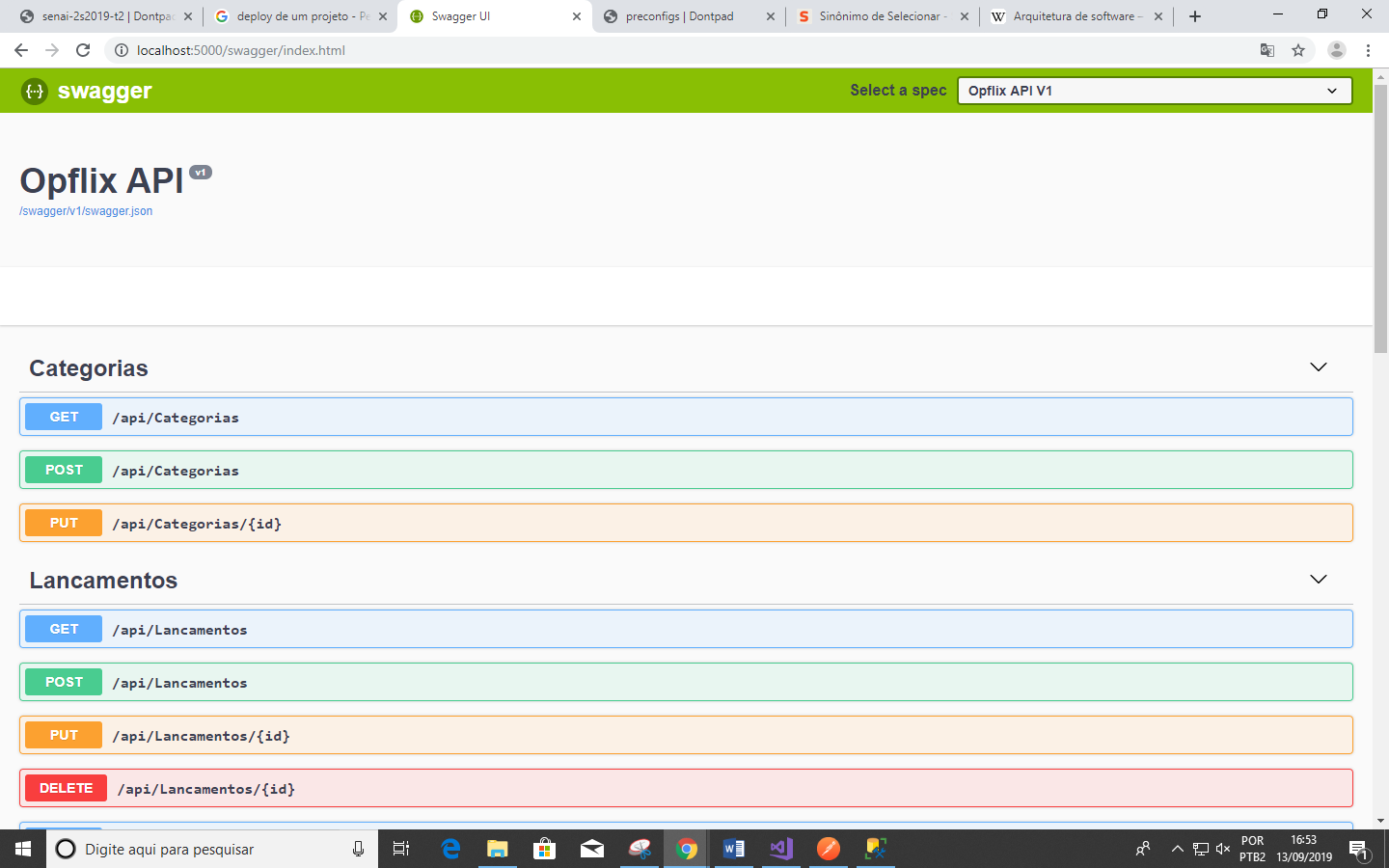
Acessando

**1º -** Antes é necessário “rodar o projeto”, sendo assim, na plataforma Visual Studio (Microsoft) pressione a tecla F11 ou então com o mouse selecione a opção de execução na parte superior do ambiente de desenvolvimento, lembre-se de selecionar a opção com o nome “Senai.OpFlix.WebApi”;



**2º -** Após a execução, abra um navegador e acesse a URL : “<http://localhost:5000/swagger/index.html>”;

**3º -** Com o Swagger aberto, as funcionalidades estarão disponíveis para a visualização e teste.



Pacotes Nuget

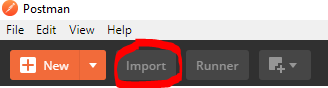
Pacotes Necessários

* **Uso Swagger:**
  + Swashbuckle.AspNetCore 4.0.1
* **Uso SqlClient:**
  + System.Data.SqlClient 4.6.1
* **Uso Entity Framework Core:**
  + Microsoft.EntityFrameworkCore.SqlServer 2.1.11
  + Microsoft.EntityFrameworkCore.SqlServer.Design 1.1.6
  + Microsoft.EntityFrameworkCore.Tools 2.1.11
* **Uso JSON Web Token:**
  + Microsoft.AspNetCore.Authentication.JwtBearer 2.1.1
  + System.IdentityModel.Tokens.Jwt 5.5.0

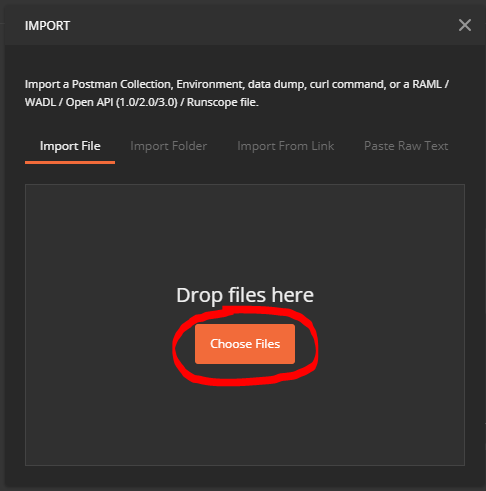
Postman

Importando

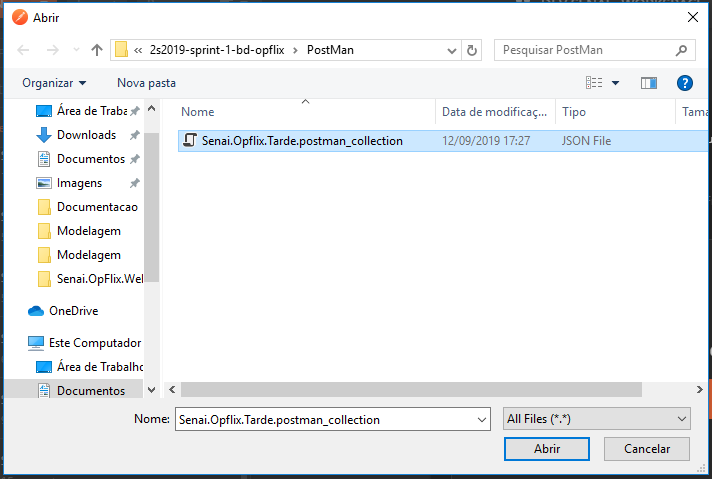
**1º -** Aberto o aplicativo Postman, selecione o botão “Import” no canto superior esquerdo da tela;



**2º -** Na nova aba, há dois caminhos ou arrastar e soltar o arquivo (arquivo de um export anterior) no local solicitado, ou selecionar o caminho do diretório. Respectivamente, pelo segundo caminho clique no botão “Choose Files”



**3º -** Selecione o projeto e clique em abrir.



Funcionalidades

Web

Mobile

Front-End

Referências

Links

GitHub:

LinkedIn:

Pinterest: