

Documentação

WishList

Sumário

1. [Resumo 2](#_Toc3466888)

[Objetivos](#_Toc3466889)

1. [Descrição do projeto 2](#_Toc3466890)

[Resumo do projeto](#_Toc3466891)

1. [Modelagem de Software 3](#_Toc3466892)

[Modelo Lógico](#_Toc3466893)

[Modelo Físico](#_Toc3466894)

[Modelo Conceitual](#_Toc3466895)

[Cronograma](#_Toc3466896)

1. [Back-End 6](#_Toc3466897)
2. [Funcionalidades 10](#_Toc3466898)

[Web](#_Toc3466899)

[Mobile](#_Toc3466900)

1. [Protótipos 12](#_Toc3466901)

[Web](#_Toc3466902)

[Mobile](#_Toc3466903)

1. [Front-End 13](#_Toc3466904)
2. [Mobile 14](#_Toc3466905)
3. [Arquitetura do Projeto 15](#_Toc3466906)
4. [Referências 16](#_Toc3466907)

[Links](#_Toc3466908)

[Livros](#_Toc3466909)

Resumo

Objetivos

Desenvolver um sistema web/mobile integrado onde seja possível realizar a gestão da clínica de forma automatizada e ter acesso fácil aos dados sobre as informações de seus pacientes.

Descrição do projeto

Uma nova clínica médica chamada SP Medical Group, empresa de pequeno porte que atua no ramo da saúde, foi criada pelo médico Fernando Strada em 2019 na região da Paulista em São Paulo. Fernando tem uma equipe de médicos que atuam em diversas áreas (pediatria, odontologia, gastrenterologia, etc.).

Sua empresa, por ser nova, iniciou a administração da clínica de forma simples utilizando softwares de planilhas eletrônicas, e, com o sucesso da clínica sua gestão passou a se tornar complicada devida a alta demanda dos pacientes.

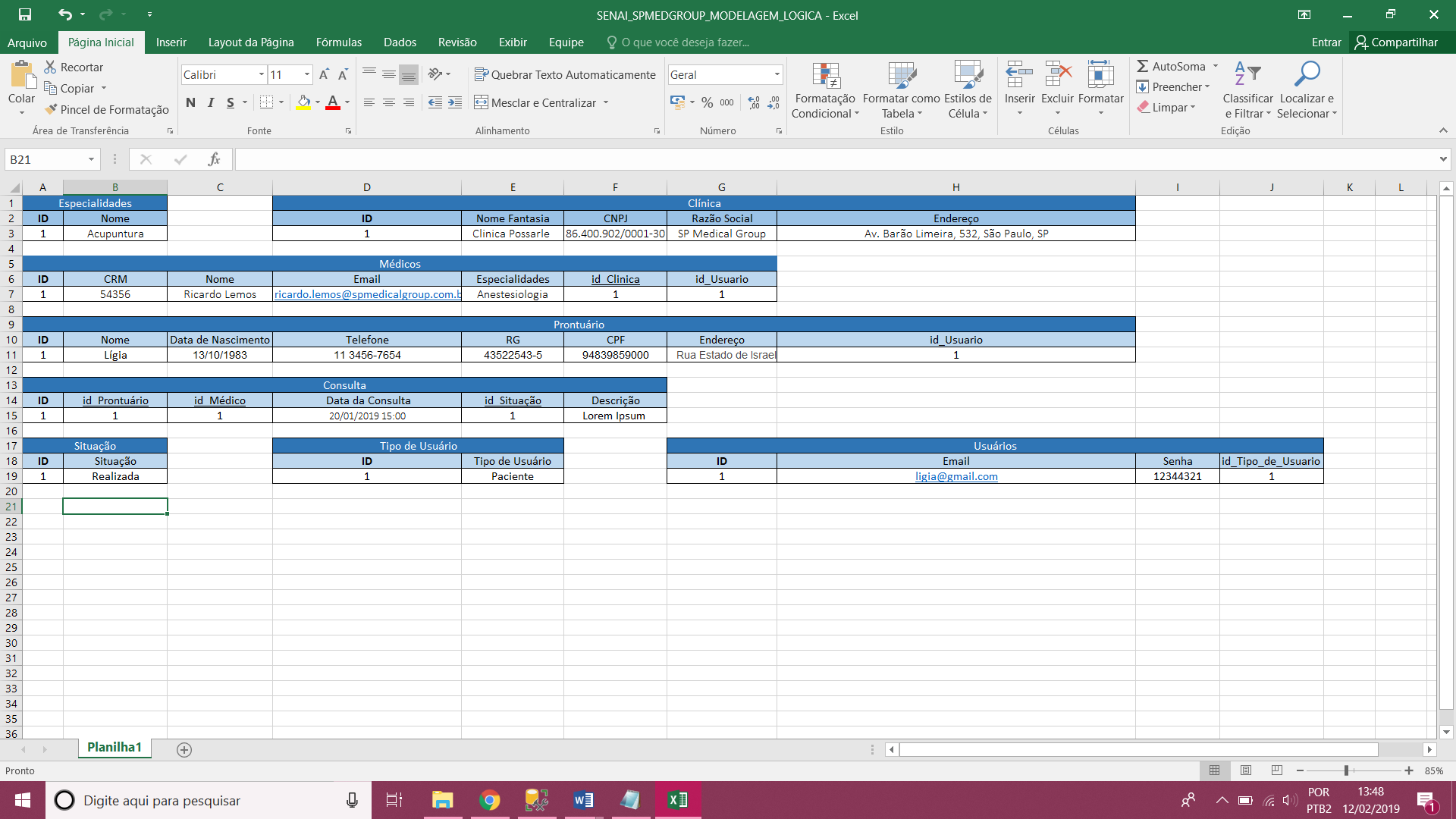
Resumo do projeto

Desenvolver um software de planilhas eletrônicas.

Modelagem de Software

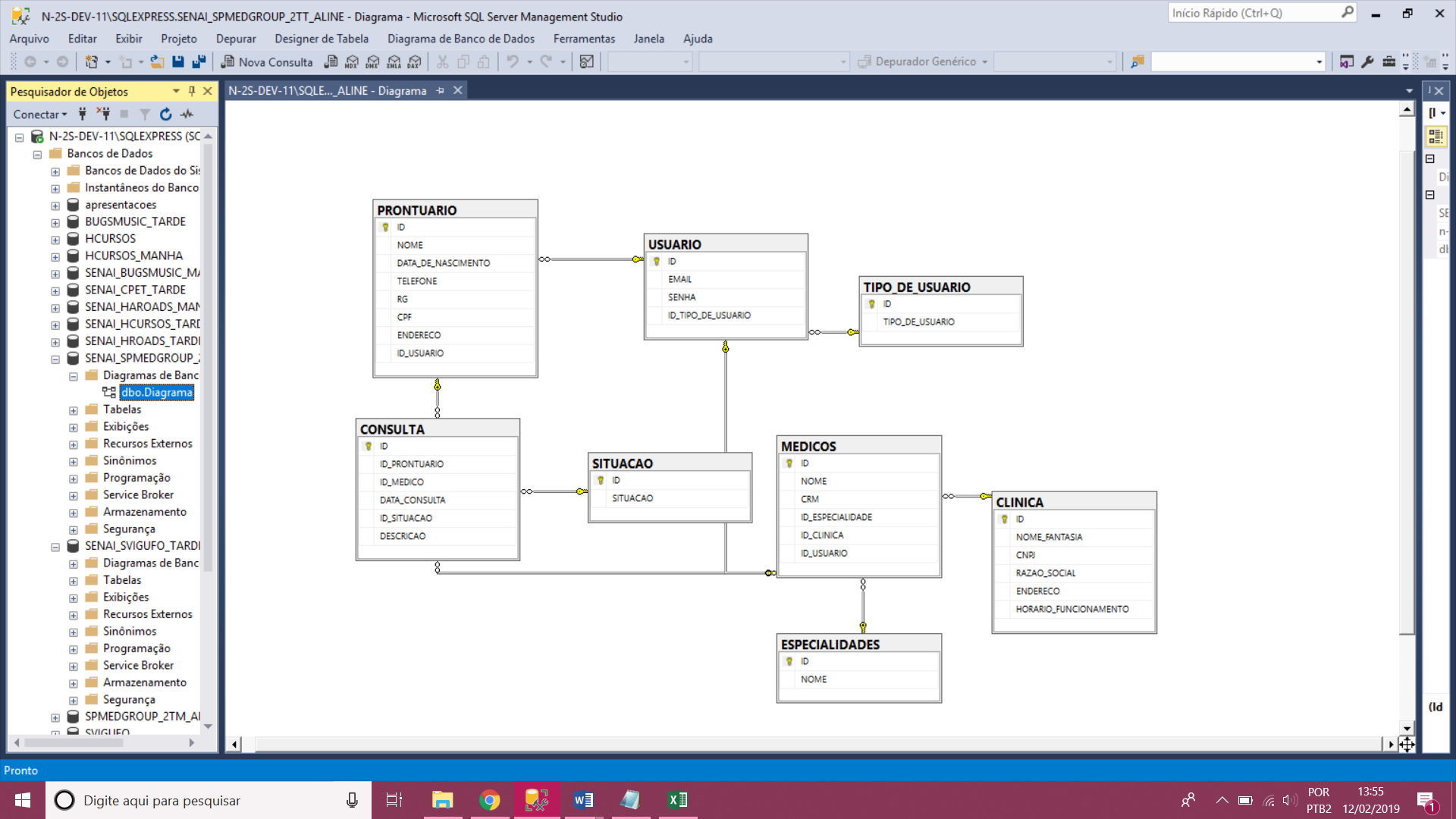
Modelagem de software é a atividade de construir modelos que expliquem as características ou o comportamento de um software ou de um sistema de software.

Modelo Lógico



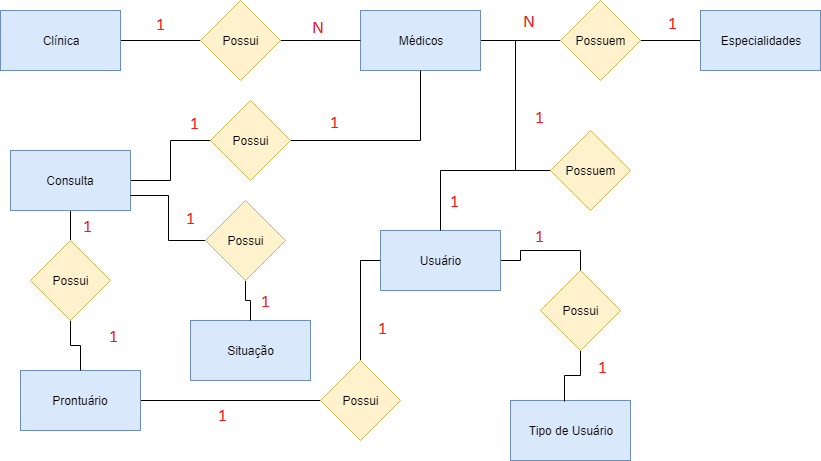
O modelo lógico constitui uma representação específica de um modelo interno, utilizando as estruturas de Banco de Dados suportada pelo banco escolhido.

Modelo Físico



O modelo físico demonstra como os dados são fisicamente armazenados.

Modelo Conceitual



O objetivo do Modelo Conceitual é criar um modelo de forma gráfica, que identificará todas as entidades e relacionamentos de uma forma global, evitando qualquer detalhamento específico do modelo de Banco de Dados.

Cronograma

Banco de Dados:

<https://trello.com/b/GAO4uSpj/spmedgroup-2tt-aline>

WebApi:

https://trello.com/b/Y310KVBh/spmedgroup-2tt-aline-webapi

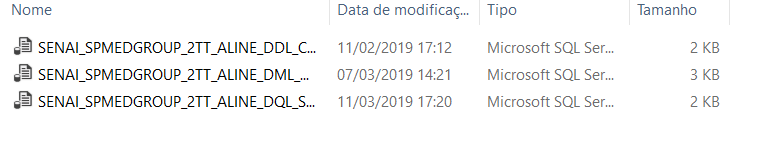
Back-End

**API** significa Application Programming Interface – Interface de Programação de Aplicativos – e corresponde a um conjunto de instruções e padrões de programação para acesso a um aplicativo de software.

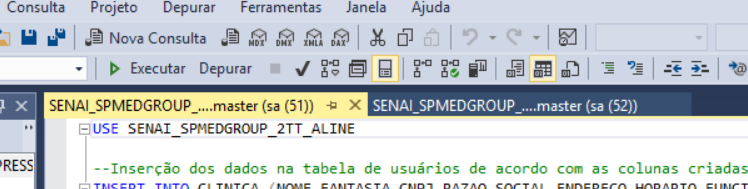
**Passo a passo:**

**Executando o Banco de Dados**

1. Abrir scripts do banco de dados

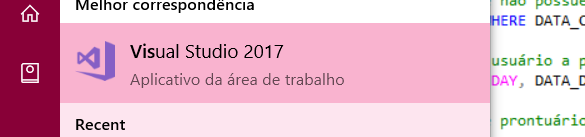
****

1. Clicar no botão “Executar” em cada script (DDL,DML e DQL)

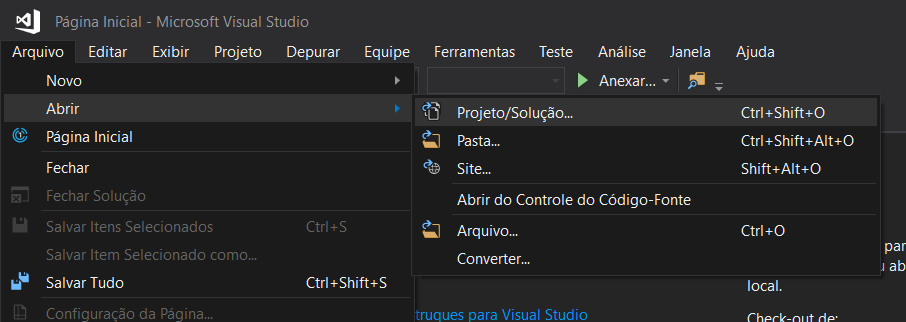
****

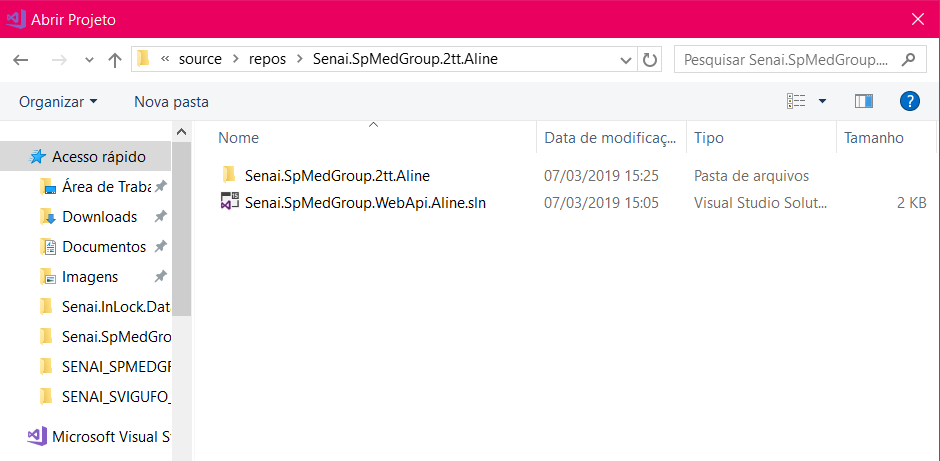
**Abrindo o projeto**

1. Abra o Visual Studio 2017



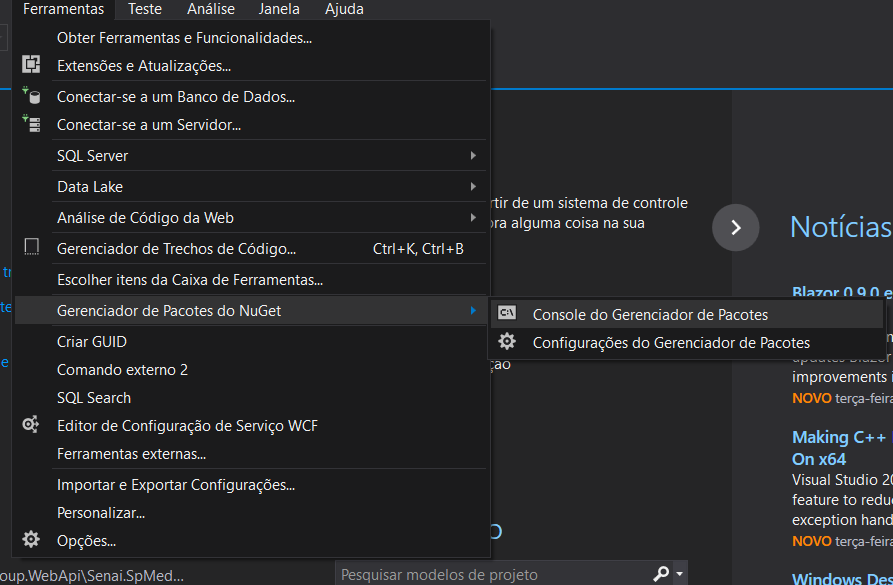
1. Abra o projeto

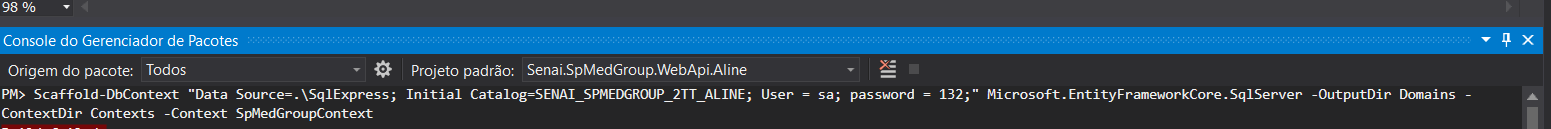
****

****

**Caso ocorra alteração no Banco de Dados:**

1. Selecione o Console do Gerenciador de Pacotes do NuGet



1. Execute: 

Scaffold-DbContext "Data Source=.\SqlExpress; Initial Catalog=SENAI\_SPMEDGROUP\_2TT\_ALINE; User = sa; password = 132;" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Domains -ContextDir Contexts -Context SpMedGroupContext

Nesta instrução temos:

O comando: Scaffold-DbContext

A string de conexão do banco de dados

O nome do provedor usado

-OutputDir - Nome da Pasta que fica as classes

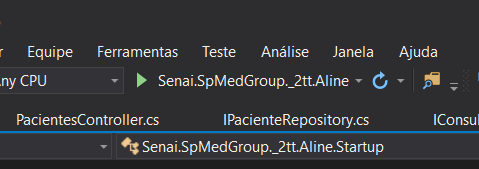
-ContextDir - Nome da pasta que fica o Contexto

-Context - Nome do arquivo de Contexto

**Atenção: Verifique se o Usuário e Senha passado corresponde com o acesso do servidor do banco de dados.**

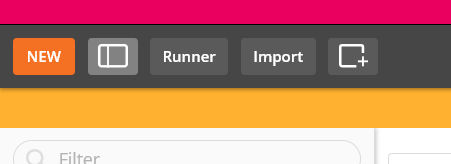
**Executando o projeto:**

1. Clique no botão de executar para rodar o programa:

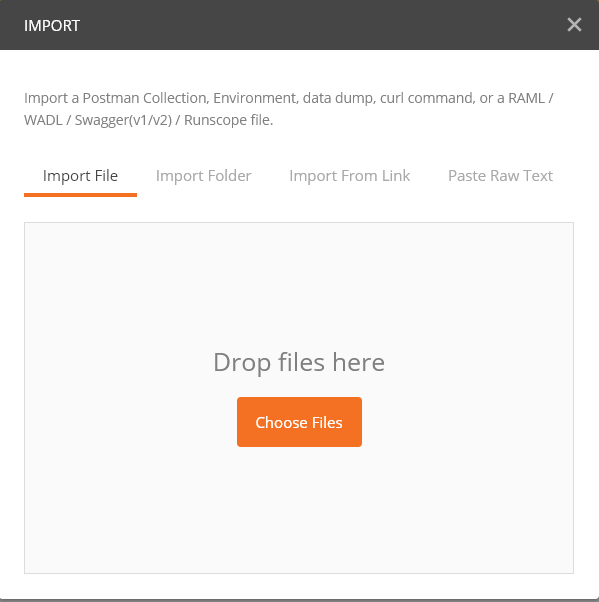


**Importando para o Postman**

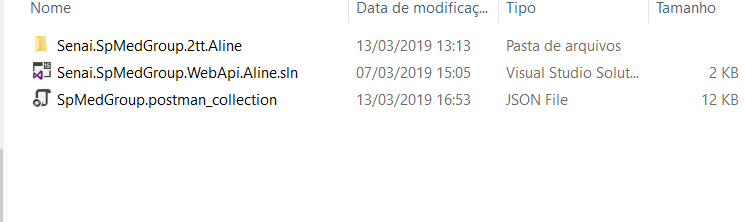
1. Clique no botão Import



1. Clique em Choose Files



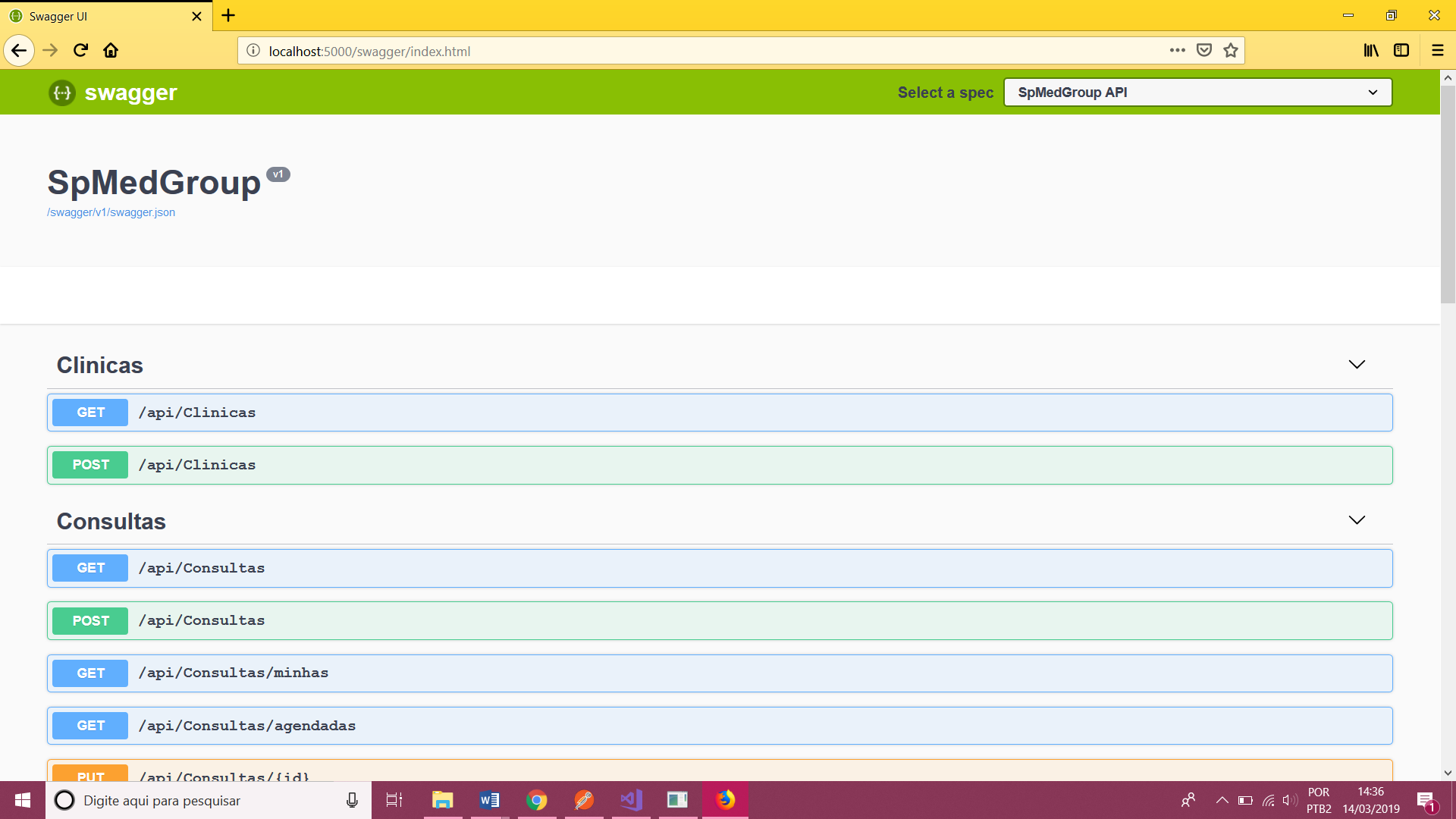
1. Escolha o arquivo



**Exibir a documentação pelo Swagger:**

1. Execute o programa e abra a URL:

**http://localhost:5000/swagger/index.html**



Funcionalidades

Web

1. O administrador poderá cadastrar qualquer tipo de **usuário (administrador, paciente ou médico)**;
2. O administrador poderá agendar uma consulta, onde será informado o **paciente**, data do agendamento e qual **médico** irá atender a consulta (o médico possuirá sua determinada especialidade);
3. O administrador poderá cancelar o agendamento;
4. O administrador deverá informar os dados da clínica (como endereço, horário de funcionamento, cnpj, nome fantasia e razão social);
5. O médico poderá ver os agendamentos (consultas) associados a ele;
6. O médico poderá incluir a descrição da **consulta que estará vinculado ao paciente (prontuário)**;
7. O paciente poderá visualizar suas próprias consultas;

Mobile

1. O paciente poderá visualizar suas próprias consultas;
2. O paciente poderá visualizar a rota entre sua localização e o endereço da clínica;
3. O médico poderá ver os agendamentos associados a ele;
4. **Notificação:** No momento que o administrador cadastrar um agendamento, o **médico** deverá receber uma notificação pelo aplicativo;

Protótipos

Web

Mobile

Front-End

Mobile

Arquitetura do Projeto

Referências

Links

Livros