

Documentação

SP Medical Group

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

Documento que traz os detalhes do sistema desenvolvido, tais como sua estrutura, planejamento, modelagem e funcionalidades.

Descrição do projeto

Desenvolver um sistema web/mobile com um banco de estruturados, para armazenar os dados passados pelo cliente, juntamente com uma API que será desenvolvida utilizando C#, servindo tanto para web quanto mobile, e por fim criar toda a parte da interface do sistema, utilizando React para web e React-Native para mobile.

Resumo do projeto

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.

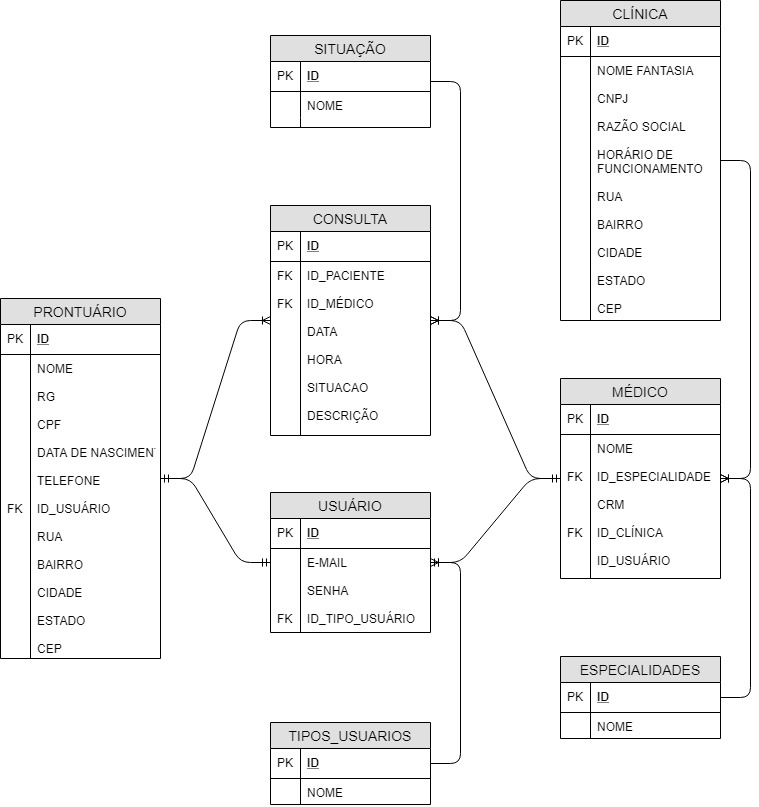
Modelagem de Software

A modelagem é uma das principais atividades que levam à implementação de um bom software. Construímos modelos para comunicar a estrutura e o comportamento desejados do sistema, visualizar e controlar a arquitetura do mesmo e compreender melhor o sistema que estamos elaborando.

A modelagem de software utiliza vários modelos para projetar um determinado sistema. Um modelo é uma simplificação da realidade, criado para facilitar o entendimento de sistemas complexos. Estes modelos podem abranger planos detalhados, assim como planos mais gerais com uma visão panorâmica do sistema.

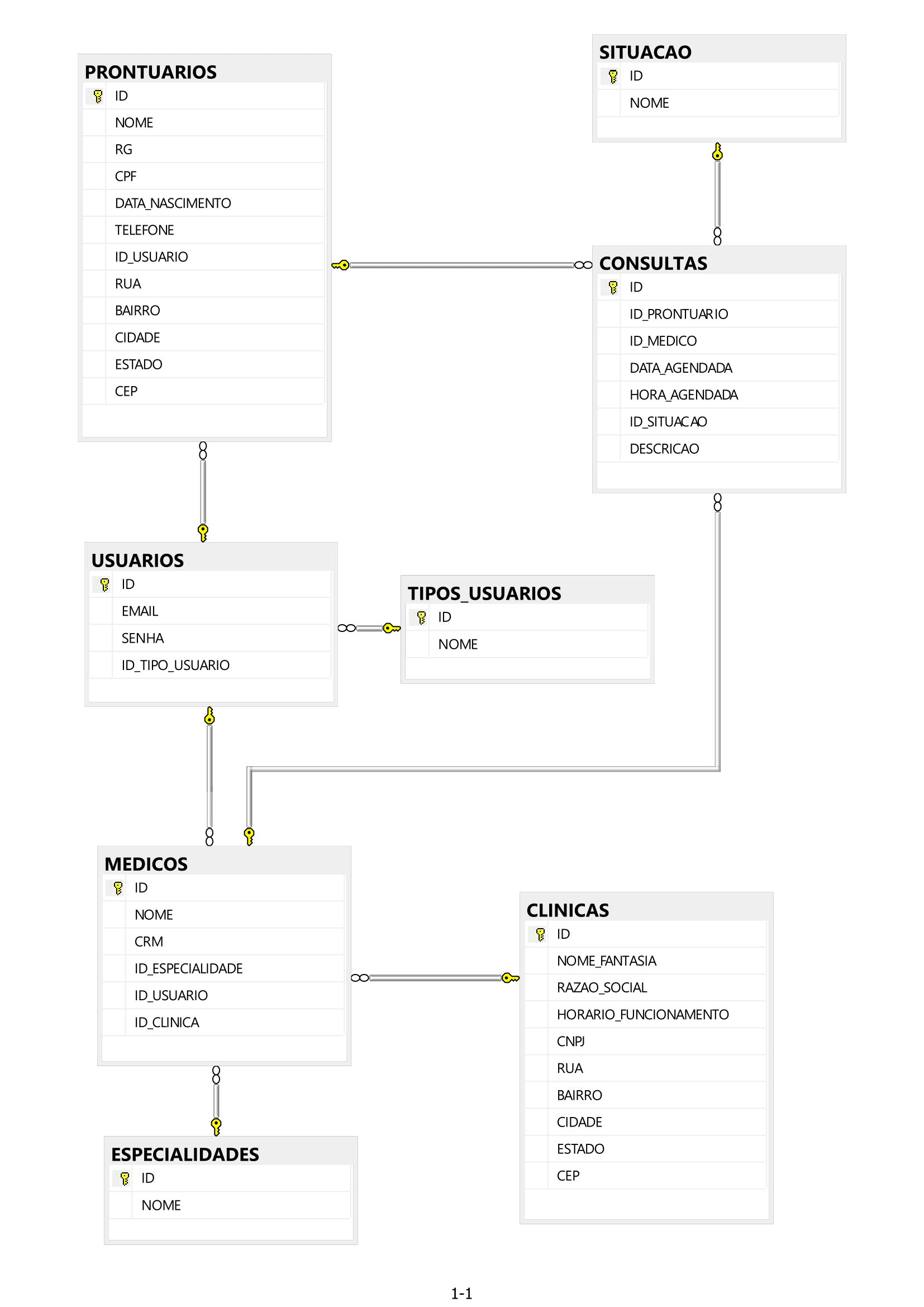
Modelo Lógico

Modelo lógico do banco de dados da SP Medical Group, contendo todas tabelas e seus respectivos campos necessarios para criação do banco de dados.



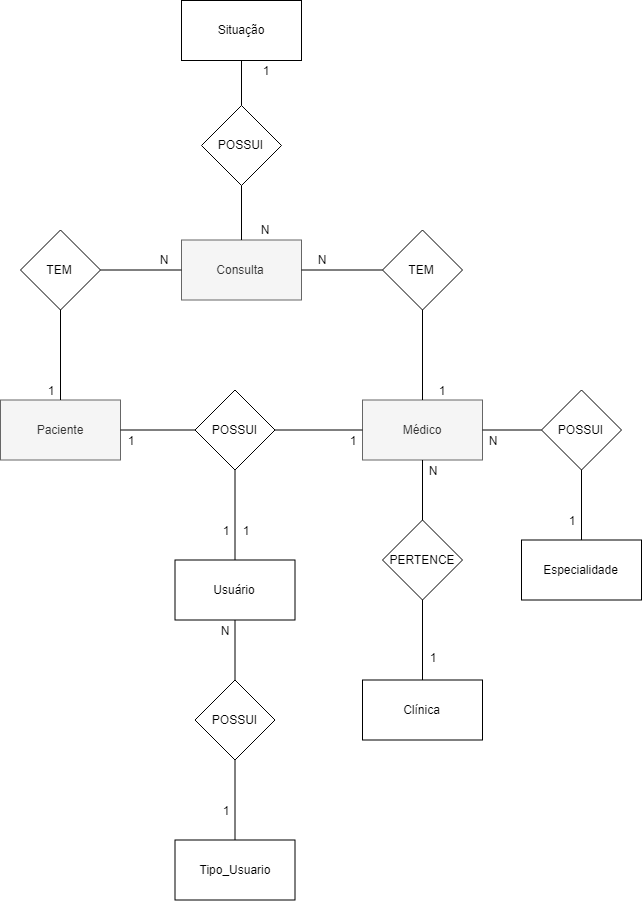
Modelo Físico

Modelo físico do banco de dados da Sp Medical Group, apresenta como será feita a armazenagem no banco de dados, levando em conta o modelo lógico adotada, nesse caso o SQL Server.



Modelo Conceitual

Modelo conceitual do banco de dados da SP Medical Group, representando como serão as estruturas de armazenamento de dados e seus possíveis relacionamentos.



Cronograma

Banco de Dados:

WebApi:

<https://trello.com/b/AKR53dP4/api-spmedicalgroup>

Layouts:

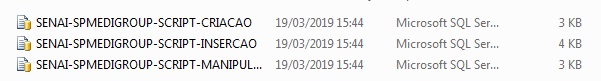
<https://trello.com/b/QsyTLYOU/layouts-do-site-spmedgroup>

Back-End

Passo a Passo com database offline:

Executando o Banco de dados:

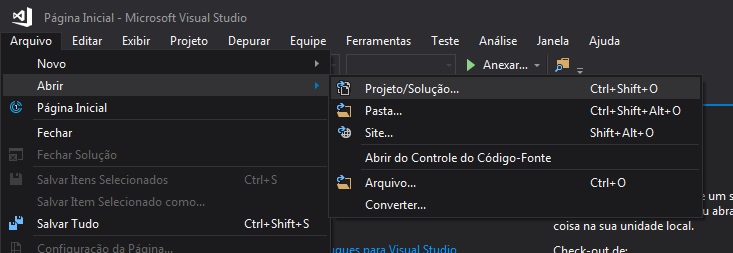
1. Abra os Scripts do banco de dados no SQL Server, ou outro programa de database. Os scripts estão localizados na raiz da pasta do projeto, em “**\_database > scripts >**”.



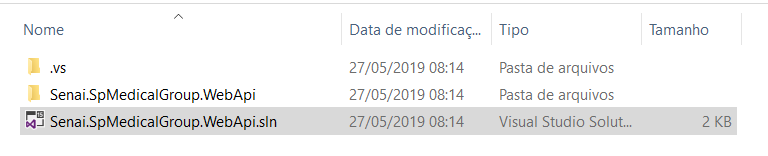
1. Execute os Scripts “**SENAI-SPMEDIGROUP-SCRIPT-CRIACAO**” (DDL) e “**SENAI-SPMEDIGROUP-SCRIPT-INSERCAO**” (DML) para a criação e inserção do banco de dados, e o Script “**SENAI-SPMEDIGROUP-SCRIPT-CRIACAO**” (DQL) para chamadas e queries no banco de dados, não sendo necessário a execução do último para criar o banco de dados, utiliza-lo apenas quando quiser listar os registros do banco de dados, executando apenas um comando por vez.

Abrindo o Projeto backend:

1. Abra o Visual Studio 2017, ou outra versão compatível.
2. Abra o Projeto.

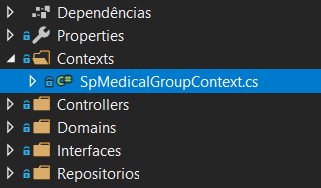


1. Selecione a solução “**Senai.SpMedicalGroup.WebApi.sln**”. A solução do projeto está localizado em “**\_backend > Senai.SpMedicalGroup.WebApi >**”



String de conexão do banco de dados:

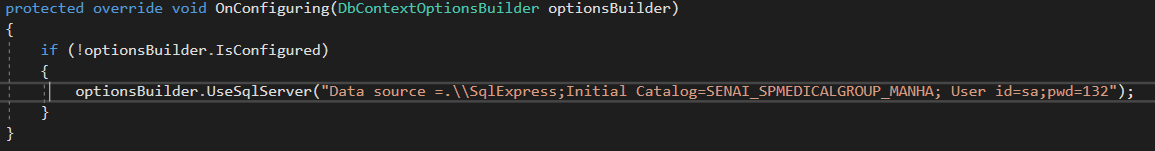
1. Com o projeto aberto vá até o “**SpMedicalGroupContext**”. Localizado em “**Context >**”



1. Altere o código

“**optionsBuilder.UseSqlServer("Data source =.\\{SEU\_SERVIDOR};Initial Catalog=SENAI\_SPMEDICALGROUP\_MANHA; User id={USUARIO\_SERVER};pwd={SENHA\_USER\_SERVER}");**”

Mudando “**{SEU\_SERVIDOR}**” por seu servidor (ex. sqlserver), “**{USUARIO\_SERVER}**” para o usuário do servidor (ex. sa) e “**{SENHA\_USER\_SERVER}**” para a senha do usuário acima (ex. 132);

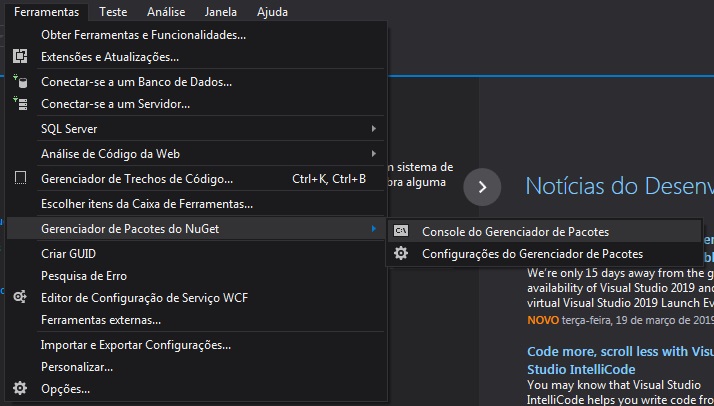


1. Faça o mesmo para “**ConsultasRepositorio**”, “**MedicosRepositorio**”, “**ProntuariosRepositorio**” e “**UsuariosRepositorio**”. Esses arquivos estão localizados em “**Repositorios >**”, trocando a “**StringConexao**” pela sua string de conexão do banco de dados.

C:\Users\50473694808\Desktop\Arquivos\documentacao_imgs\stringConexaoRepos.PNG

Conexão com o banco de dados:

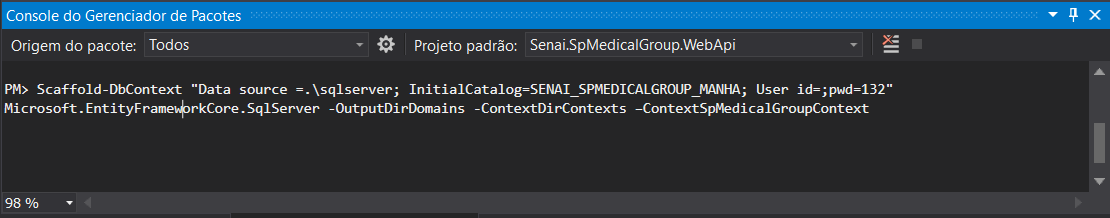
1. Selecione o Console do Gerenciador de Pacotes no Nuget



1. Execute:

**“Scaffold-DbContext "Data source =.\{SEU\_SERVIDOR}; InitialCatalog=SENAI\_SPMEDICALGROUP\_MANHA; User id={USUARIO\_SERVER}; pwd={SENHA\_USER\_SERVER}" Microsoft.EntityFrameworkCore.SqlServer -OutputDirDomains -ContextDirContexts –ContextSpMedicalGroupContext”**

Mude “**{SEU\_SERVIDOR}**” por seu servidor (ex. sqlserver);, “**{USUARIO\_SERVER}**” para o usuário do servidor (ex. sa), “**{SENHA\_USER\_SERVER}**” para a senha do usuário acima (ex. 132);



Passo a Passo com database online:

O projto disponibiliza um banco de dados online, e para usá-lo basta seguir os passo da item acima “**String de conexão do banco de dados**”, porem adicionando a string de conexão como:

“"**Server=tcp:serversenaircd.database.windows.net,1433;Initial Catalog=SENAI\_SPMEDICALGROUP\_MANHA;Persist Security Info=False;User ID=ricardopaulo;Password=Dcrp246dragon;MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;**"”

Executando o Projeto:

Com o projeto aberto

1. Em“**launchSettings.json**”, localizado em “**Properties >**”, altere os dois atributos “**"applicationUrl"**”para o IP da sua máquina:

**“"applicationUrl": "http://{SEU\_IP}:5000"**”

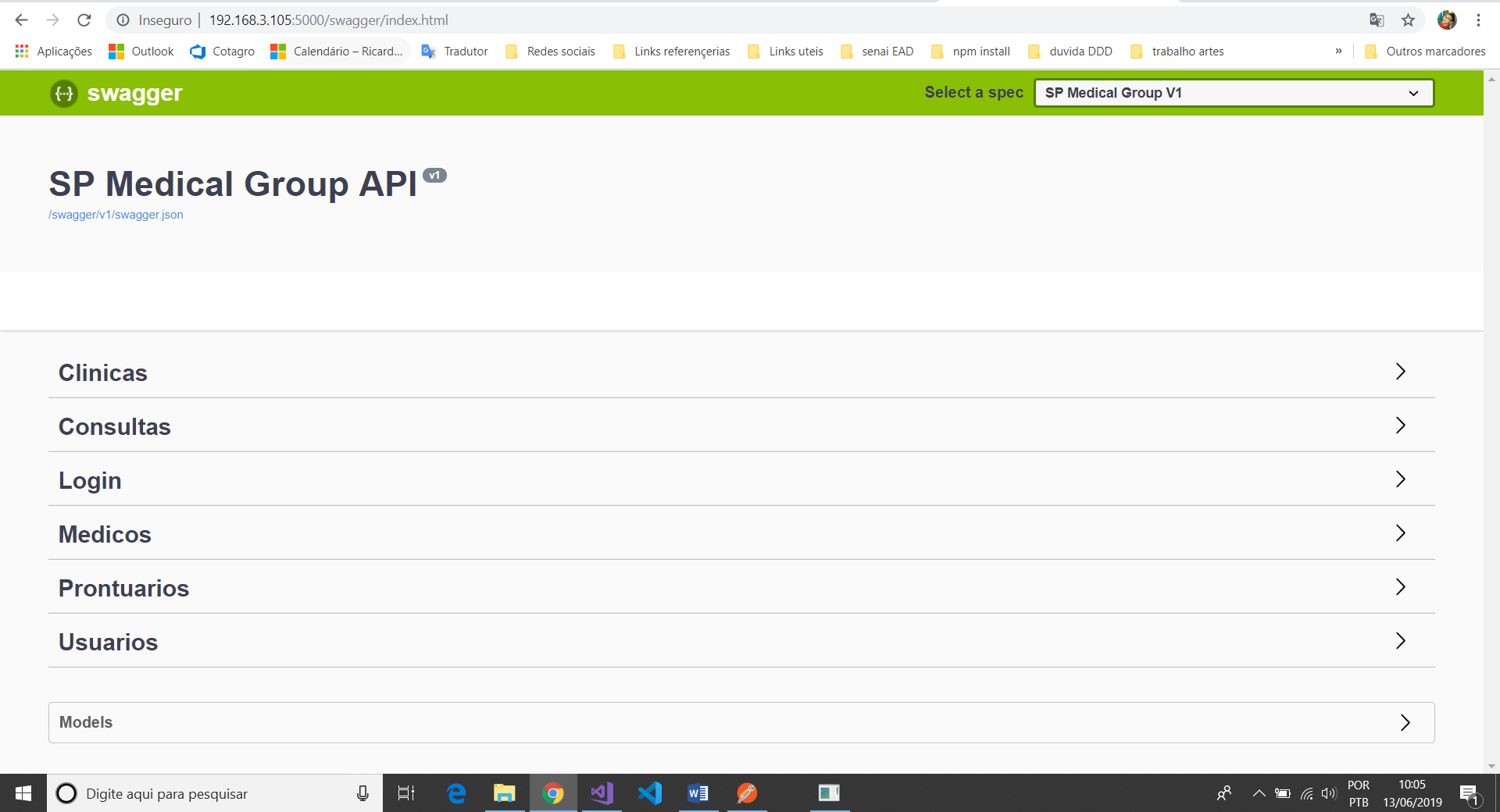


1. Clique no botão de Executar para rodar o programa

C:\Users\7\Desktop\Senai\executar.jpg

Exibir a documentação da API pelo Swagger:

1. Com o projeto em execução, abra o site na url que você configurou no “**launchSettings.json**” da seguinte forma: “[**{SUA\_URL\_API}/swagger**](http://localhost:5000/swagger) “



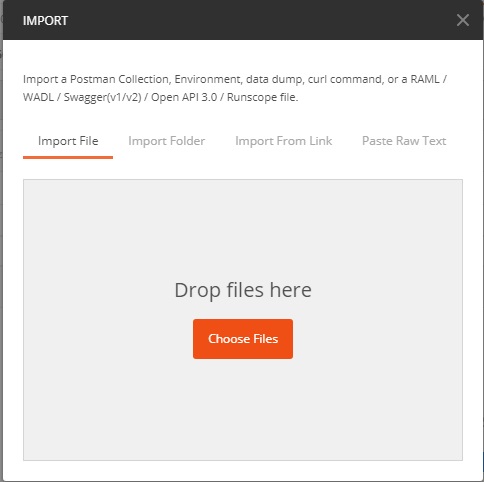
Importando requisições para o Postman:

Abra o Postman

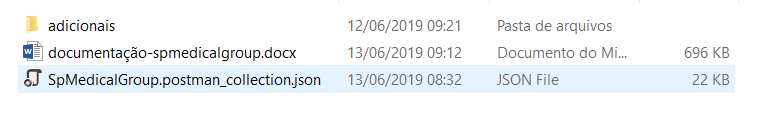
1. Click em “**Import**”.

C:\Users\7\Desktop\Senai\import.jpg

1. Clique em “**Choose Files**”.



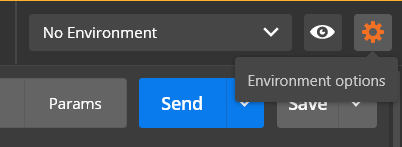
1. Escolha o Arquivo. Nesse caso “**SpMedicalGroup.postman\_collection**”. O Arquivo do Postman está localizado em “**documentacao\_arquivosadicionais >**”



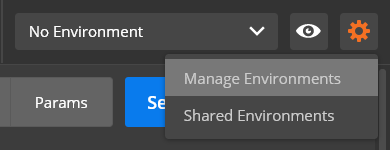
Configurando variáveis de ambiente no Postman:

**Com o** Postman aberto

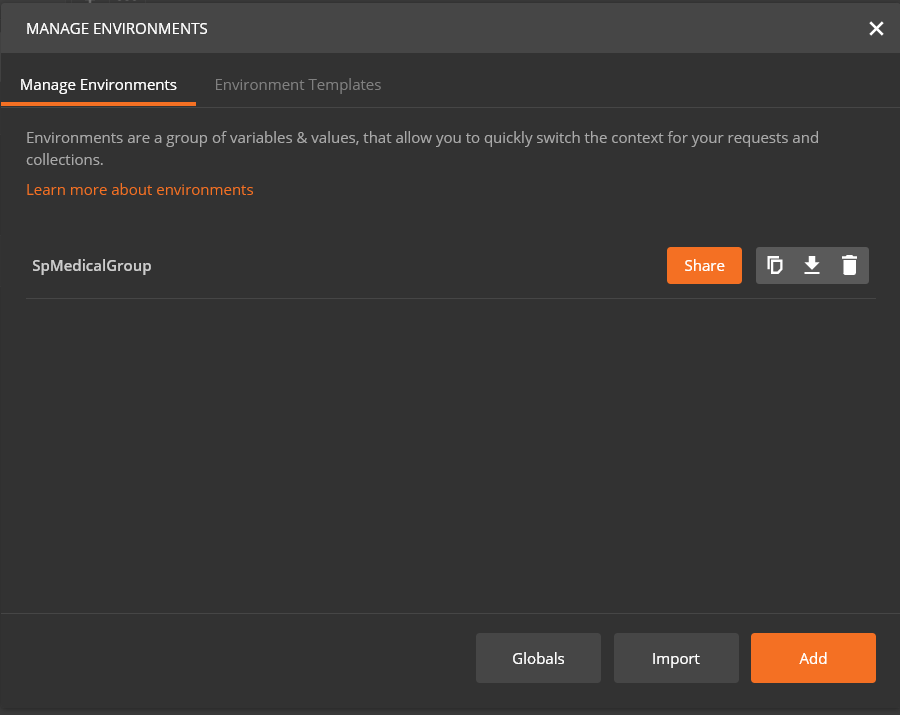
1. Clique em “**Environment options**”



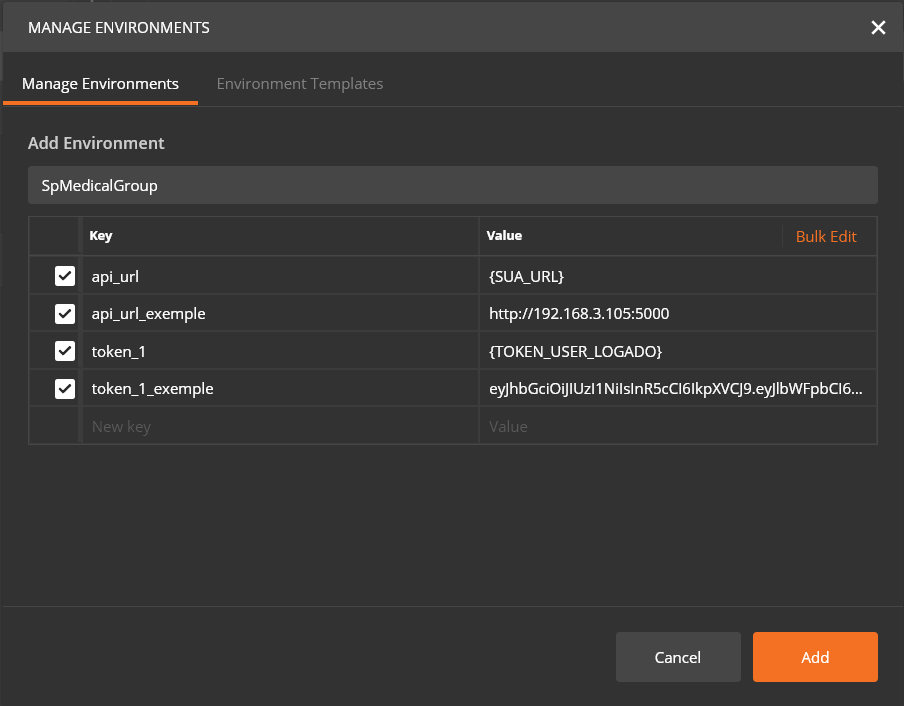
1. Selecione “**Manage Environments**”



1. Na janela que abriu clique em “**Add**”



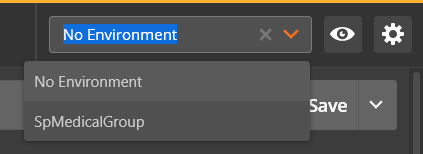
1. Crie um novo Environment chamado “**SpMedicalGroup**”, e adicione as variáveis “**url\_api**” como “**KEY**” e no “**VALUE**” passe a url da sua api, também configure uma variável para “**token\_1**” como “**KEY**” e como “**VALUE**” passe o token gerado ao logar com um usuário, caso não tenha o token do usuário não tem problemas, ele pode ser adicionado depois, deixando o “**VALUE**” dele vazio.



1. E por fim, clique em “Add” para criar o novo Environment

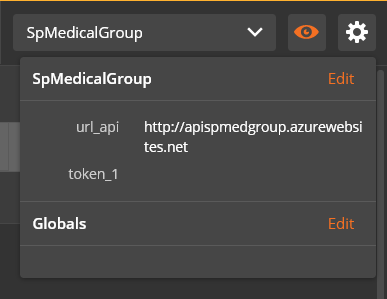
Utilizando um Environment:

1. Troque o “**No Environment**” por “**SpMedicalGroup**”

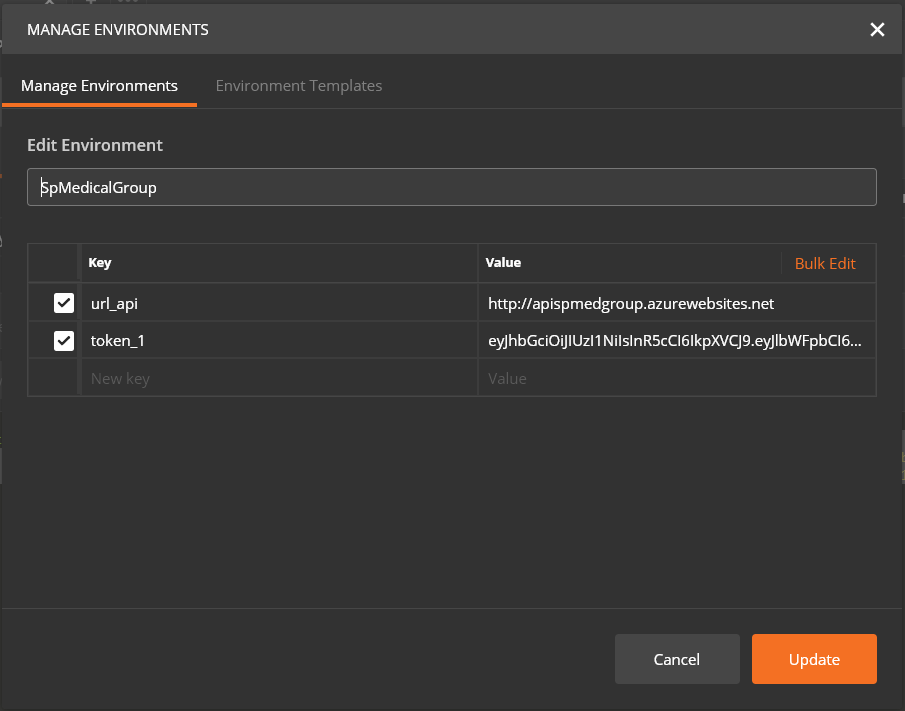


Fazendo requisições no Postman:

1. Na Coleção de endpoints do SpMedicalGroup, vá na pasta de “**Login**” e selecione um dos usuários para efetuar login, ou acesse qualquer um e coloque seu usuário.
2. Envie a requisição, e se o usuário for autenticado com sucesso, a requisição retornará um “**token**”
3. Copie esse token, sem as aspas, e coloque na variável “**token\_1**” do Environment que foi criada
4. Para isso clique em “**Environment quick look**”



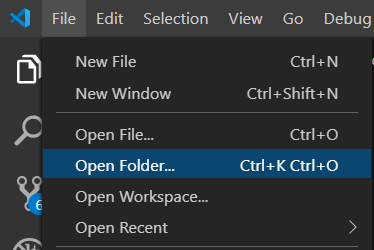
1. Clique em “**Edit**”
2. Troque o “**VALUE**” do “**token\_1**” pelo token do usuário logado.



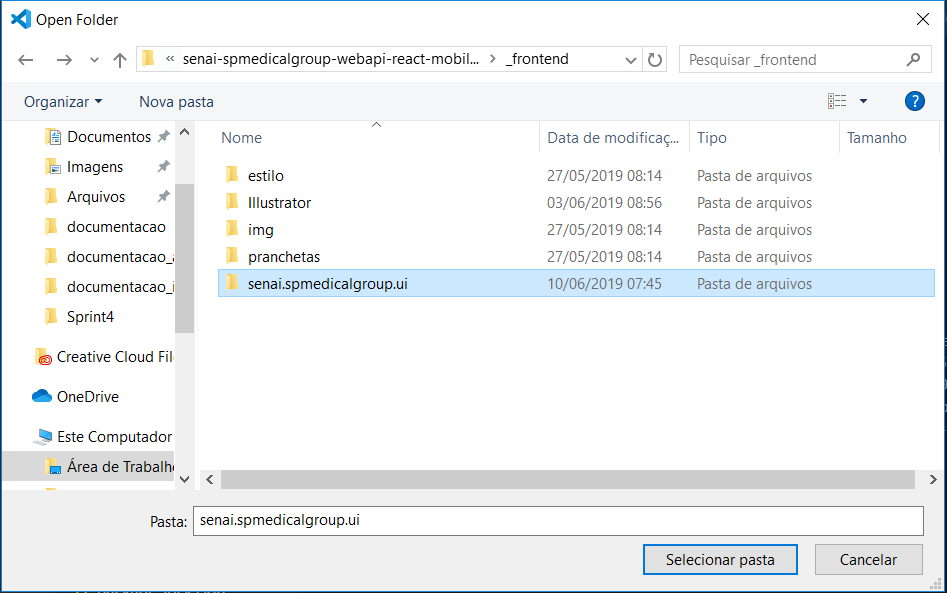
1. Feito isso, as requisições já podem ser feiras, dependendo claro das permissões do usuário logado;

Abrindo o Projeto frontend:

1. Abra o Visual Studio Code
2. Abra o Projeto.



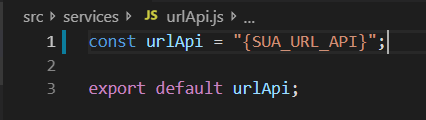
1. Selecione a pasta “**senai.spmedicalgroup.ui**” e clique em “**Selecionar pasta**”. A pasta está localizada em “**\_frontend >**”



Configurando url das requisições:

Com o projeto frontend aberto

1. Acesse o arquivo “**urlApi.js**”. Esse arquivo está localizado em “**src > services >** ”
2. Altera “**urlApi**” para a url da sua API



1. E salve o arquivo

Funcionalidades

Web

O **Administrador** do sistema poderá cadastrar um novo Usuário, Paciente (Prontuário), Médico, Clínica e uma nova Consulta, podendo alterar, deletar e exibir uma lista dos mesmos. Podendo também cadastrar um Localização de casos de doenças, e visualizar a lista do mesmo através de um mapa.

Os **Médicos** cadastrados no sistema poderão alterar ou incluir a descrição da consulta que está vinculada ao paciente (prontuário), além de poder visualizar a lista de consultas vinculadas a eles.

Os **Pacientes** poderão apenas visualizar a lista de consultas vinculadas com eles.

Mobile

O app não da suporte para usuáriro **Administradores.**

Os **Médicos** poderão ver suas consultas.

Os **Pacientes** poderão ver suas consultas.

Protótipos

Web

Mobile

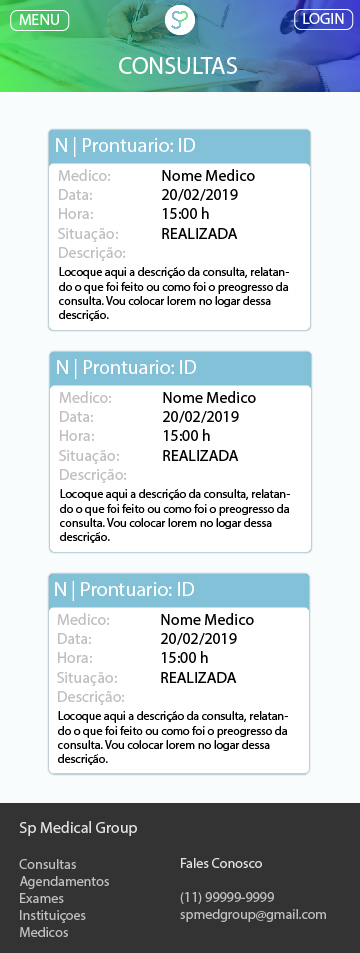
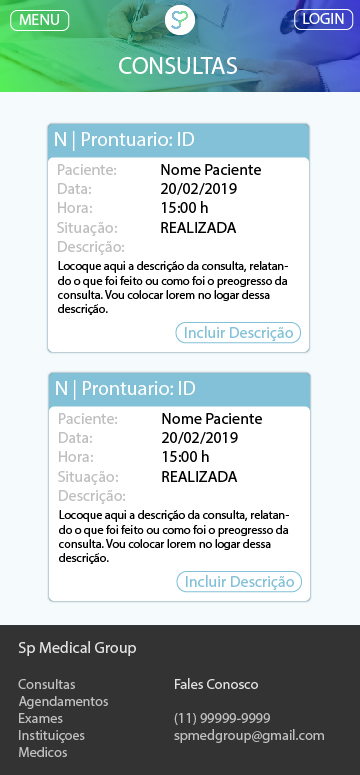
Front-End

Responsividade adequada para o celular

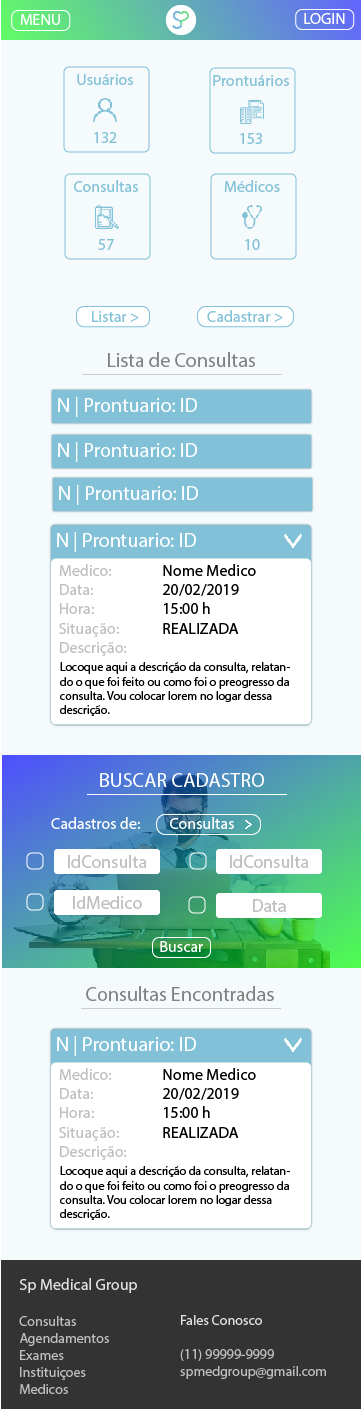
Login / Home



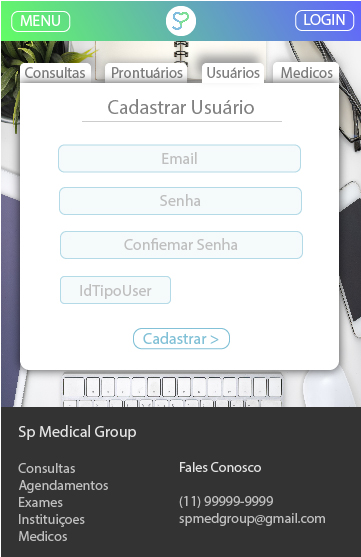
Lista de Consultas de Paciente , Lista de Consultas de Médico

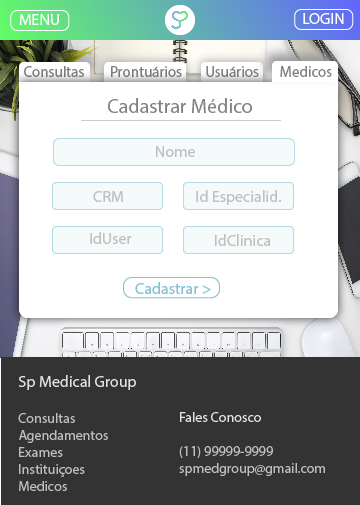
 

DashBoard Adminstrador

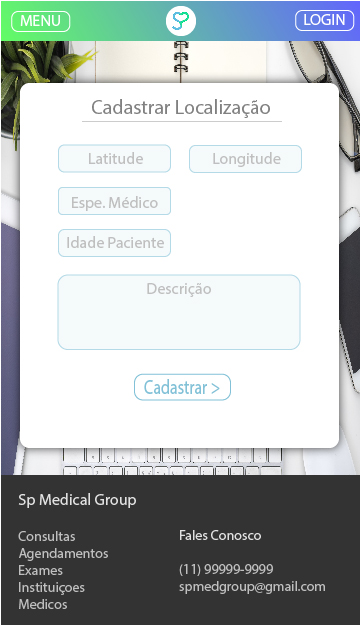


Cadastar Consulta, Cadastrar Prontuário, Cadastrar Médico, Cadastrar Usuário

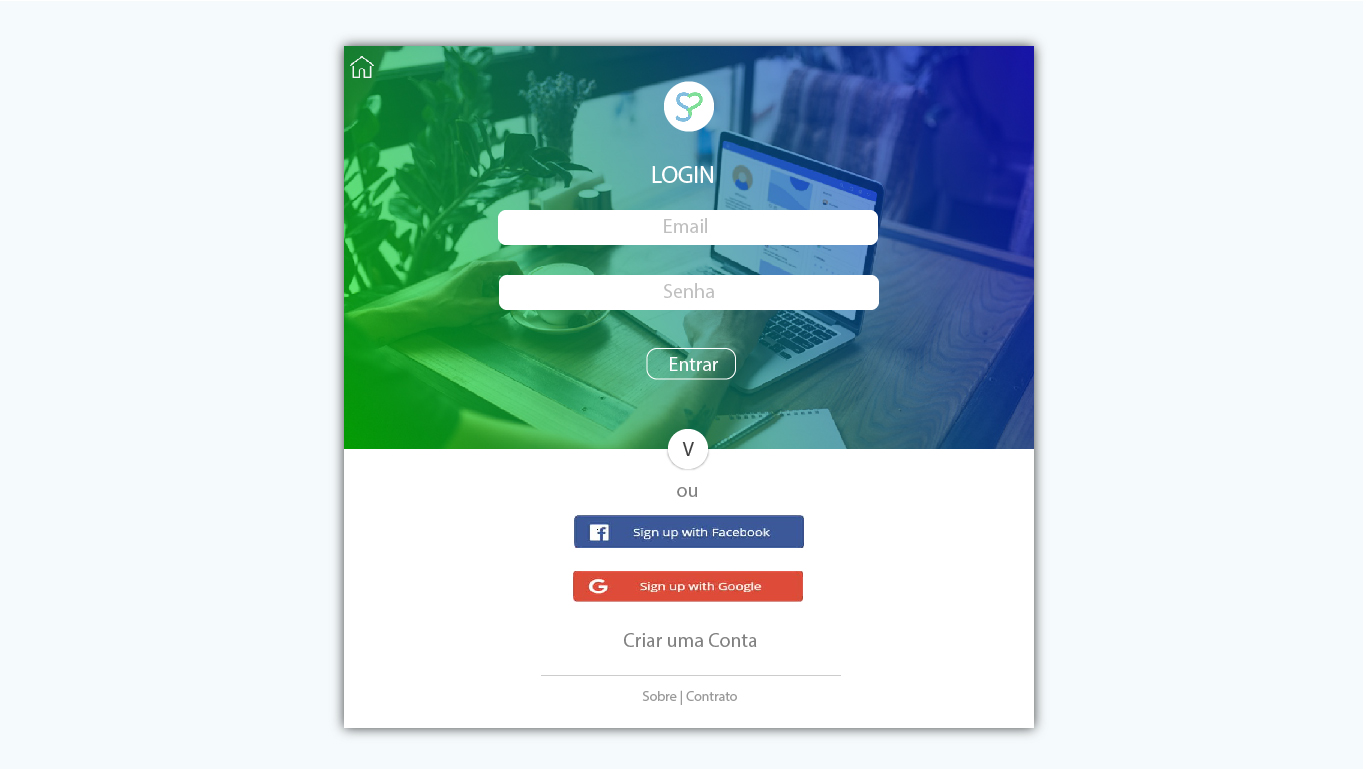
 

Cadastrar Localização

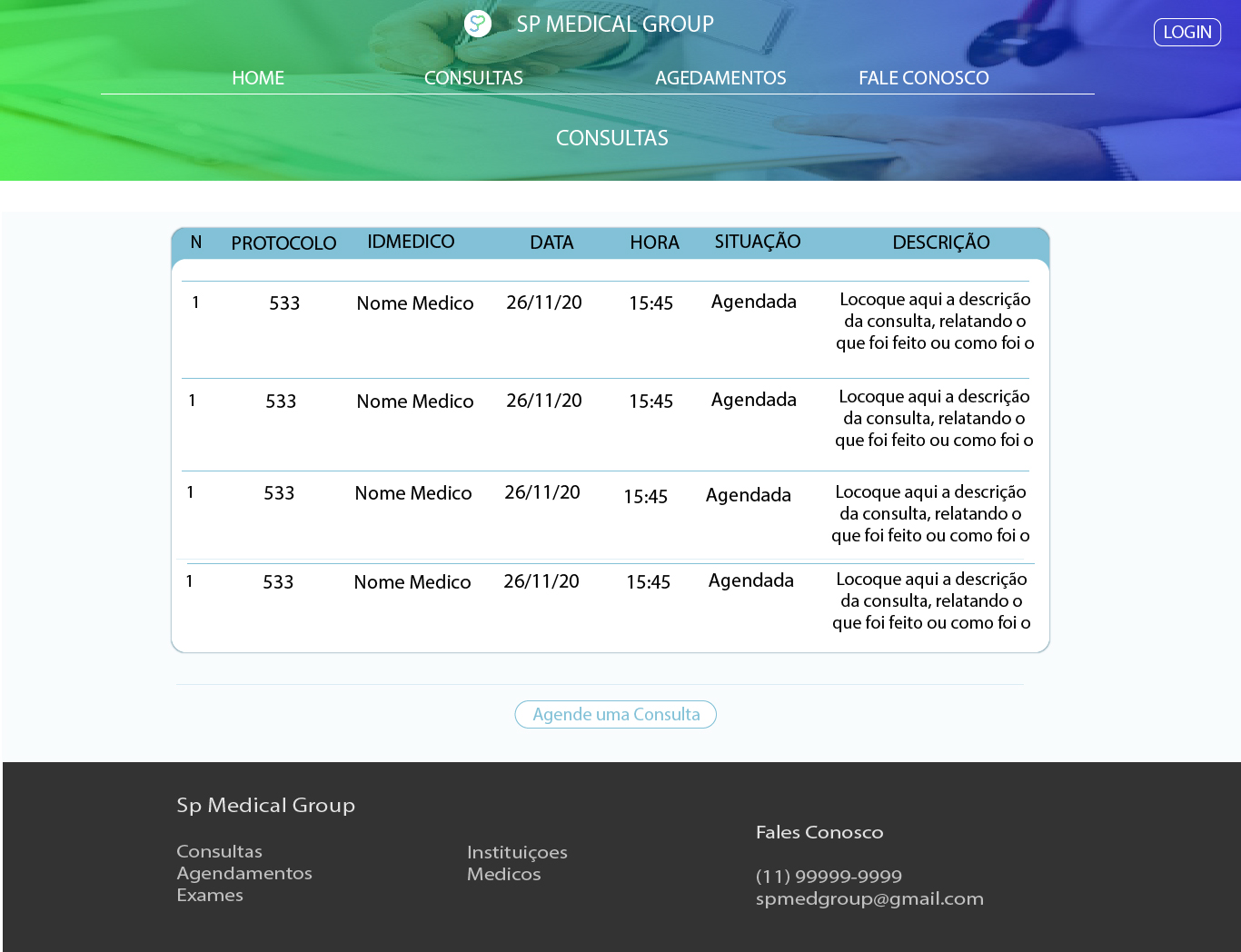


Responsividade adequada para o web, PC.

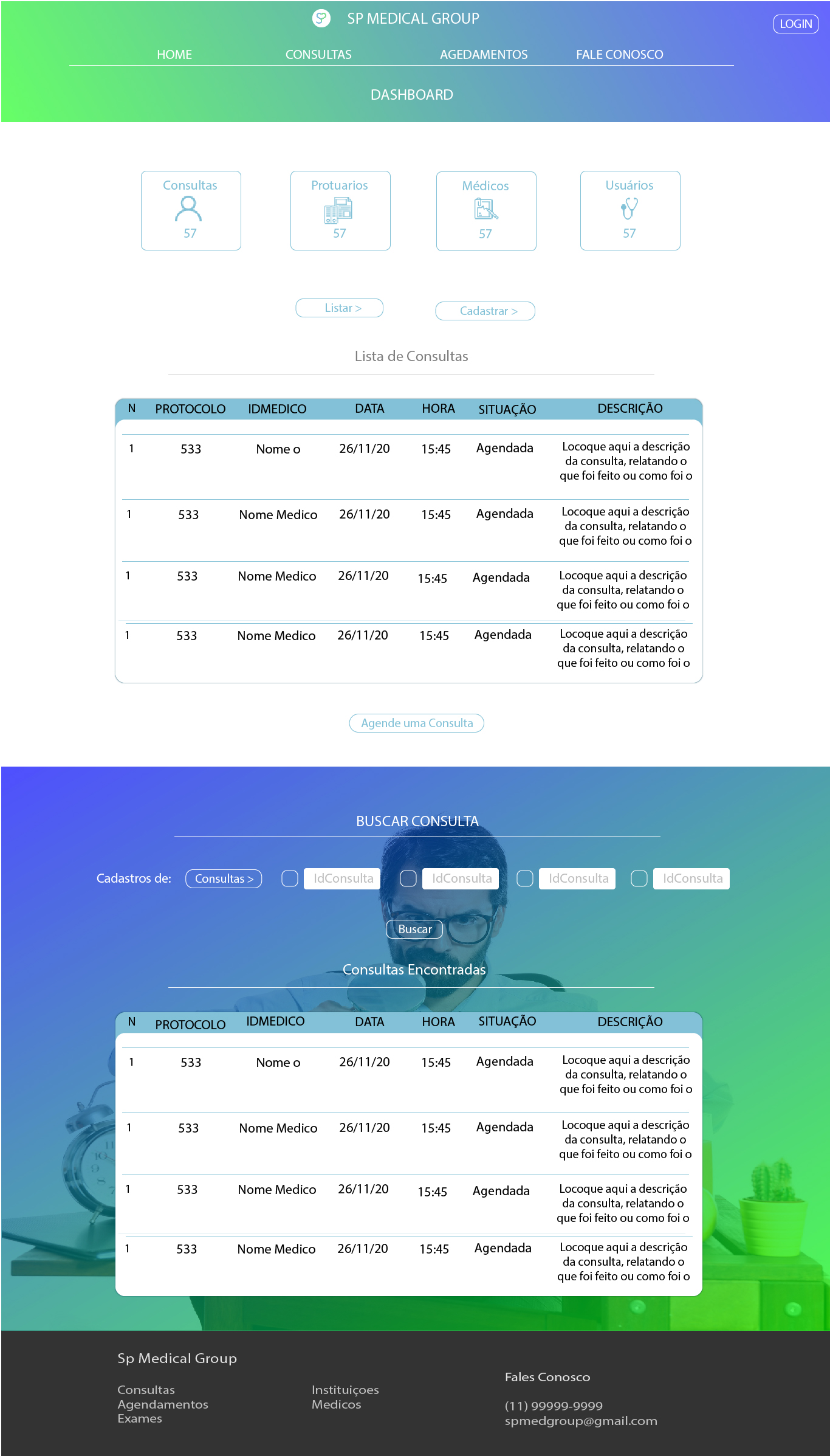
Login



Lista Consultas

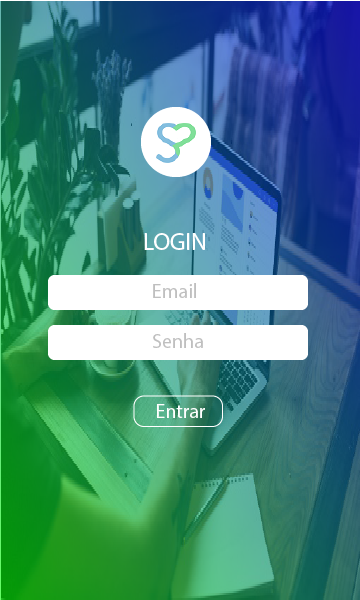


DashBoard Adminstrador

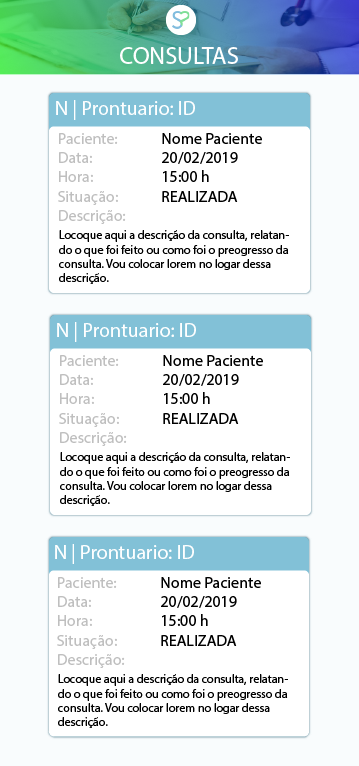
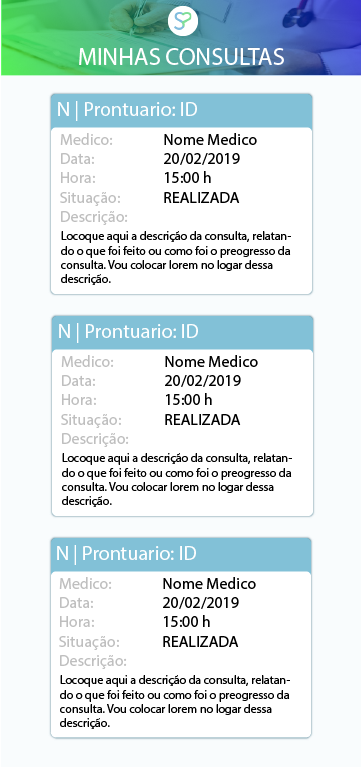


Mobile

Login



Lista Consultas de Paciente, Lista Consultas de Médico

Arquitetura do Projeto

Referências

Links

Livros