**SP MEDICAL GROUP**

Documentação

Sumário

[1. Resumo 3](#_Toc71092221)

[2. Descrição do projeto 3](#_Toc71092222)

[3. Modelagem de Dados 3](#_Toc71092223)

[Modelo Conceitual 3](#_Toc71092224)

[Modelo Lógico 3](#_Toc71092225)

[Modelo Físico 3](#_Toc71092226)

[Cronograma 3](#_Toc71092227)

[Trello 4](#_Toc71092228)

[4. Back-End 4](#_Toc71092229)

[Funcionalidades 5](#_Toc71092230)

[Sistema Web 5](#_Toc71092231)

[Perfis de usuário: 5](#_Toc71092232)

[Funcionalidades: 5](#_Toc71092233)

[Sistema Mobile 5](#_Toc71092234)

[Perfis de usuário: 5](#_Toc71092235)

[Funcionalidades: 5](#_Toc71092236)

# Resumo

Descrição das funcionalidades e particularidades do projeto SP Medical Group.

# Descrição do projeto

SP Medical Group, iniciativa com propósito na facilitação da organização de clínicas médicas.

# Modelagem de Dados

Processo de organização da base de dados que serão armazenados no banco de dados.

## Modelo Conceitual

## Diagrama Descrição gerada automaticamente

Primeira etapa da modelagem de dados, consiste na esquematização do banco de dados de maneira simplificada.

## Modelo Lógico

## Diagrama Descrição gerada automaticamente

Segunda etapa da modelagem de dados, consiste na esquematização do banco de dados e suas relações de maneira mais específica e complexa.

## Modelo Físico

## Diagrama Descrição gerada automaticamente

Terceira etapa da modelagem de dados, representação do banco de dados de maneira específica e prática.

## Cronograma

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 | Dia 6 | Dia 7 | Dia 8 | Dia 9 | Dia  10 | Dia  11 | Dia  12 | Dia  13 | Dia  14 | Dia  15 | Dia  16 | Dia  17 |
| Modelo Conceitual | X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Modelo Lógico | x |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Modelo Físico |  |  |  | X | X |  |  |  |  |  |  |  |  |  |  |  |  |
| DDL | X | X | X | X | X |  |  |  |  |  |  |  |  |  |  |  |  |
| DML | X | X | X | X | X |  |  |  |  |  |  |  |  |  |  |  |  |
| DQL | X | X | X | X | X |  |  |  |  |  |  |  |  |  |  |  |  |
| API |  |  |  |  |  | X | X | X | X |  |  |  |  |  |  |  |  |
| Postman |  |  |  |  |  |  |  | X | X |  |  |  |  |  |  |  |  |
| Azure |  |  |  |  |  |  | X | X | X |  |  |  |  |  |  |  |  |
| Front-End |  |  |  |  |  |  |  |  |  | X | X | X | X |  |  |  |  |
| Mobile |  |  |  |  |  |  |  |  |  |  |  |  |  | X | X | X | X |

### Trello

https://trello.com/b/pjJq7xli/sp-medical-group

# Back-End

O código do sistema foi desenvolvido através de uma API utilizando o Microsoft Visual Studio.

**API** é um conjunto de padrões e instruções estabelecidos para utilização do software, definindo as requisições e as respostas seguindo o protocolo HTTP, neste caso expresso no formato JSON, para que seja possível acessar o sistema em diversos dispositivos distintos sem a preocupação com a linguagem que será utilizada por estes.

Além disso, foi utilizado o estilo de arquitetura REST.

**API** – Application Programming Interface – Interface de Programação de Aplicativos.

**HTTP** – Hypertext Transfer Protocol – Protocolo de Transferência de Hipertexto.

**JSON** – JavaScript Object Notation – Notação de Objetos JavaScript.

**REST** – Representational State Transfer – Interface de Programação de Aplicativos.

**Compilação** – Processo que no qual todo o código escrito está sendo transformando em um arquivo executável, para que seja possível a leitura do mesmo pelo compilador.

**Endpoint** – **pontos de comunicação de acesso a uma aplicação (endpoints de comunicação) ou como parte de uma estrutura de segurança de rede (segurança de rede).**

O acesso a API será feito através do aplicativo **Postman**, siga os passos abaixo para a utilização da nossa aplicação:

**Baixe as seguintes ferramentas:**

**-GitHub**

Após feito o download do GitHub, copie o link do repositório, clique com o botão dinheiro na área de trabalho e adicionar o comando “git clone” mais o link do repositório

**- Visual Studio**

Tendo ambos aplicativos em sua máquina, abra o Visual e abra a pasta onde está contida aplicação e clique no botão logo abaixo de **Ferramentas** onde está escrito “SpMedicalGroup.webApi”. Aguarde um pouco até que processo de compilação seja feito.

**- Postman**

Pegue o link que foi colocado no navegador, exceto “/index.html”, em seguida coloque o nome do “/api/”, junto com o controller e a rota do endpoint desejado. Ficará da seguinte maneira : “https://localhost:5001/api/[nome da controller]/[rota do endpoint]”

## Funcionalidades

A Aplicação SpMedicalGroup.webApi tem como princípio a facilitação no processo de administração das clínicas médicas. Através de funcionalidades bem específicas e estruturadas, o gerenciamento de qualquer clínica médica pode chegar a ser muito simples com funções que vão desde o médico até o paciente tornam o processo de agendamento muito mais prático e funcional**.**

**Administrador:** Tem acesso a todos os campos e campos da nossa aplicação, podendo corrigir erros, alterar consultas, cadastrar usuários, agendar consultas etc.

**Médico:**  Possui restrição a tudo o que não estiver relacionado as consultas em que ele está registrado, pode dar diagnóstico e alterá-los.

**Prontuário:** Aqui o está o paciente, as funções atribuídas a ele é somente de ver as suas próprias consultas.

# Front-End

O **Front-End** do foi desenvolvido através da biblioteca React.

O desenvolvimento **Front-end** da interface gráfica do usuário de um site, conhecido como **UI,** foi construída por meio de uma biblioteca JavaScript fornecida pelo Facebook, para que usuários possam visualizar e interagir com a aplicação web.

**React : React** é uma biblioteca em JavaScript, que surgiu em 2011 com o Facebook. ... Dessa forma, o(a) desenvolvedor(a) Front-end que lida com o **React** para o desenvolvimento **de** páginas web pode usar os mesmos conhecimentos para desenvolver aplicativos móveis. Afinal, o **React** Native é apenas um módulo da biblioteca **React**.

## Funcionalidades

A Aplicação SpMedicalGroup.webApi tem como princípio a facilitação no processo de administração das clínicas médicas. Através de funcionalidades bem específicas e estruturadas, o gerenciamento de qualquer clínica médica pode chegar a ser muito simples com funções que vão desde o médico até o paciente tornam o processo de agendamento muito mais prático e funcional**.**

**Administrador:** Tem acesso a todos os campos e campos da nossa aplicação, podendo corrigir erros, alterar consultas, cadastrar usuários, agendar consultas etc.

**Médico:**  Possui restrição a tudo o que não estiver relacionado as consultas em que ele está registrado, pode dar diagnóstico e alterá-los.

**Prontuário:** Aqui o está o paciente, as funções atribuídas a ele é somente de ver as suas próprias consultas

# Mobile

A aplicação mobile foi desenvolvida foi desenvolvida através do desenvolvimento híbrido com um framework chamado **React Native.**

Desenvolvimento de aplicações e sistemas para dispositivos móveis, por vezes utilizado apenas como desenvolvimento mobile é toda atividade e processos acerca do desenvolvimento de softwares para dispositivos móveis. Há duas opções para trabalhar com o desenvolvimento mobile, desenvolvendo para plataformas nativamente, conversando direto com o linguagem utilizada na plataforma, ou, com o desenvolvimento híbrido onde é feito apenas um código, e o mesmo será transformado para a linguagem nativo da plataforma em que a aplicação está rodando.

**React Native :** é uma biblioteca Javascript criada pelo Facebook. É usada para desenvolver aplicativos para os sistemas Android e iOS de forma nativa.

## Funcionalidades

A Aplicação SpMedicalGroup.webApi tem como princípio a facilitação no processo de administração das clínicas médicas. Através de funcionalidades bem específicas e estruturadas, o gerenciamento de qualquer clínica médica pode chegar a ser muito simples com funções que vão desde o médico até o paciente tornam o processo de agendamento muito mais prático e funcional**.**

**Médico:**  Possui restrição a tudo o que não estiver relacionado as consultas em que ele está registrado e pode ver as consultas em que está inserido.

**Prontuário:** Aqui o está o paciente, as funções atribuídas a ele é somente de ver as suas próprias consultas.