**Sp Medical Group**

Documentação

Sumário

[1. Resumo 3](#_gjdgxs)

[2. Descrição do projeto 3](#_30j0zll)

[3. Modelagem de Dados 3](#_1fob9te)

[Modelo Conceitual 3](#_3znysh7)

[Modelo Lógico 3](#_2et92p0)

[Modelo Físico 3](#_tyjcwt)

[Cronograma 3](#_3dy6vkm)

[Trello 4](#_1t3h5sf)

[4. Back-End 4](#_4d34og8)

[Funcionalidades 5](#_2s8eyo1)

[Sistema Web 5](#_17dp8vu)

[Perfis de usuário: 5](#_3rdcrjn)

[Funcionalidades: 5](#_26in1rg)

# Resumo

Documentação do processo de desenvolvimento da API, desde da parte de modelagem do banco de dados, passando pelas tecnologias utilizadas, até a realização de requisições à API.

# Descrição do projeto

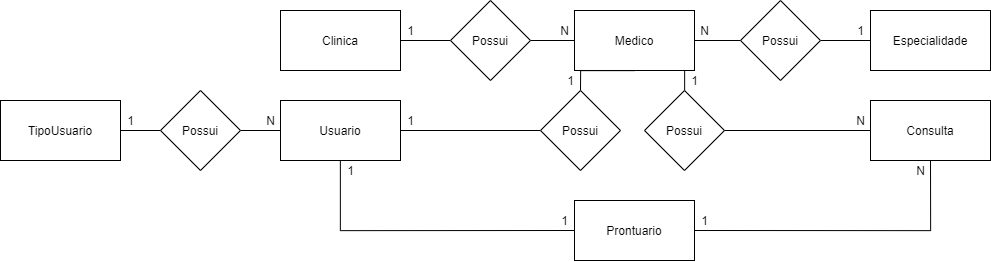
Construção de uma API para o gerenciamento de clínicas médicas de forma automatizada e de fácil acesso aos dados sobre as informações de seus pacientes.

# Modelagem de Dados

Processo de criar um modelo que explique as características de funcionamento e comportamento de um software a partir do qual ele será criado, facilitando seu entendimento e seu projeto, através das características principais que evitarão erros de programação, projeto e funcionamento.

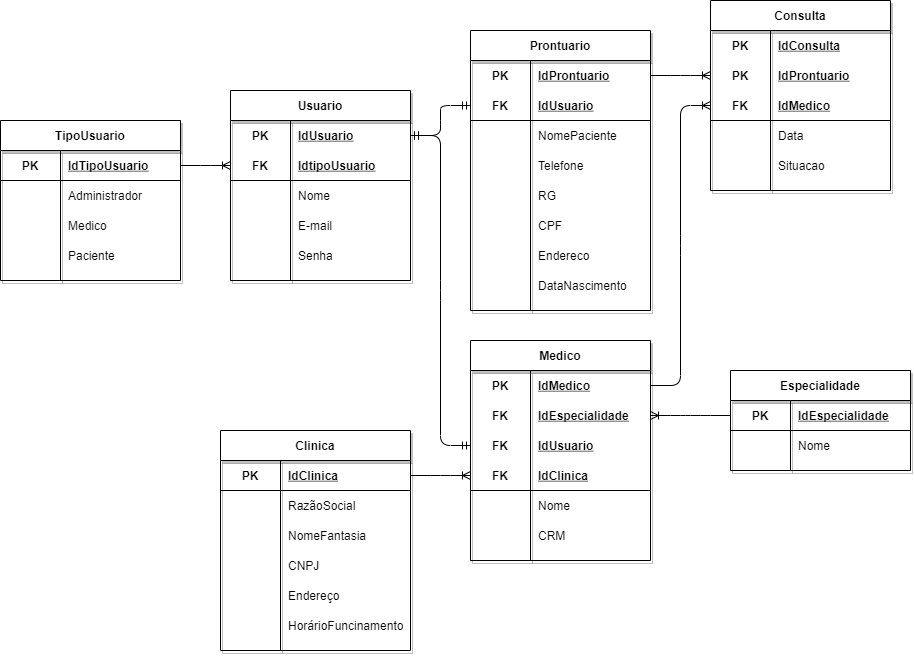
## Modelo Conceitual

O modelo conceitual consiste em definir as entidades e suas relações, nesta etapa temos um alto nível de abstração.



## Modelo Lógico

Neste modelo, o nível de abstração é mediano, pois já são identificadas e definidas as regras e a tecnologia do banco de dados.



## Modelo Físico

Essa etapa consiste em implementar o projeto do banco de dados, já se preocupando com questões como a do desempenho.



## Cronograma

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 |
| Modelo Conceitual | X |  |  |  |  |
| Modelo Lógico | X |  |  |  |  |
| Modelo Físico | X |  |  |  |  |

### Trello

[Trello - Sp Medical Group](https://trello.com/b/177ygmpL/projeto-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.

O padrão de projeto de software (software design pattern) utilizado foi o **Repository Pattern**.

O **Repository Pattern** permite um encapsulamento da lógica de acesso a dados, impulsionando o uso da injeção de dependência (DI) e proporcionando uma visão mais orientada a objetos das interações com a DAL.

## Funcionalidades

## Sistema Web

### Perfis de usuário:

1. **Administrador**: Para o colaborador da área administrativa da clínica;
2. **Médico**: Colaboradores que atuam na área da saúde;
3. **Paciente:** Clientes da clínica;

### Funcionalidades:

1. O **administrador** pode cadastrar qualquer tipo de usuário (administrador,paciente ou médico);
2. O **administrado**r pode 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** pode cancelar o agendamento;
4. O **administrador** deve informar os dados da clínica (como endereço, horário de funcionamento, CNPJ, nome fantasia e razão social);
5. O **médico** pode ver os agendamentos (consultas) associados a ele;
6. O **médico** poderá incluir a descrição da consulta que estará vinculada ao paciente (prontuário);
7. O **paciente** pode visualizar suas próprias consultas;