

Documentação

SP Medical Group

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Alameda Barão de Limeira, 539 – São Paulo/SP |  |  |  | sp.senai.br |

Sumário

1. [Resumo 2](#_Toc11048400)

[Objetivos](#_Toc11048401)

1. [Descrição do projeto 2](#_Toc11048402)
2. [Modelagem de Software 3](#_Toc11048403)

[Modelo Conceitual](#_Toc11048404)

[Modelo Lógico](#_Toc11048405)

[Modelo Físico](#_Toc11048406)

[Trello (Kanban)](#_Toc11048407)

[Cronograma](#_Toc11048408)

1. [Funcionalidades 8](#_Toc11048409)

[Endpoints](#_Toc11048410)

1. [Passo a passo 9](#_Toc11048411)

[Database First](#_Toc11048412)

1. [UI/UX – User Interface, User Experience 10](#_Toc11048413)

[Objetivos](#_Toc11048414)

[Mobile e Web](#_Toc11048415)

[Testes](#_Toc11048416)

1. [Front-end 11](#_Toc11048417)

[React](#_Toc11048418)

[Comandos](#_Toc11048419)

1. [Mobile 12](#_Toc11048420)

[React-Native](#_Toc11048421)

[Android Studio (AVD)](#_Toc11048422)

[Comandos](#_Toc11048423)

1. [Data Analytics 13](#_Toc11048424)

[Vantagens](#_Toc11048425)

[Firebase](#_Toc11048426)

1. [Referências 14](#_Toc11048427)

[Links](#_Toc11048428)

Resumo

Objetivos

O documento visa apresentar todo o processo de modelagem de dados desde seu planejamento até sua criação, usufruindo de diagramas e cronogramas.

Descrição do projeto

O projeto é um pedido da clínica SP Medical Group, que devido à alta demanda necessitou de uma base de dados para armazenar cada vez mais dados. Dessa forma, trabalhando com escalabilidade, segurança dos dados e restrições mais detalhadas.

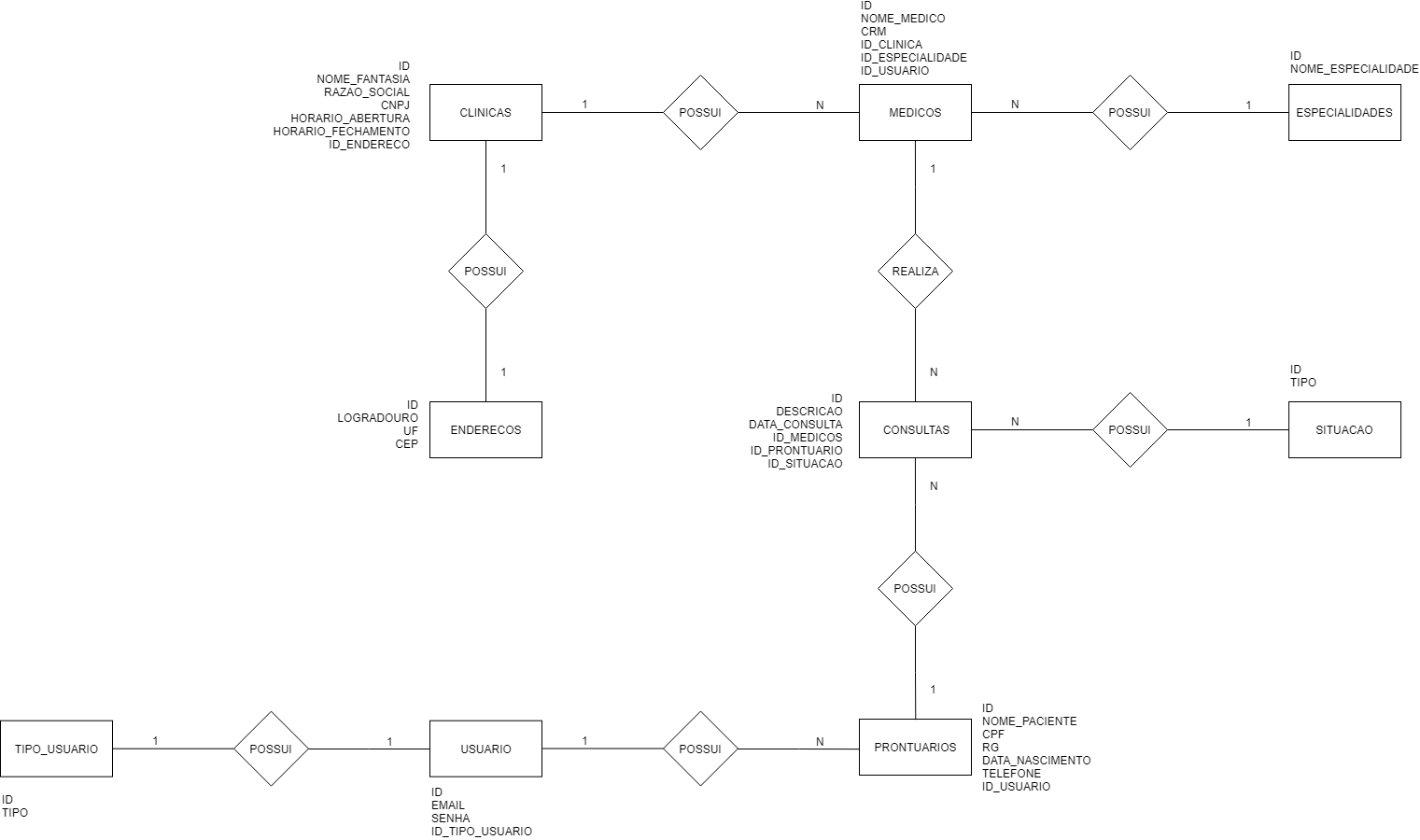
Modelagem de Software

A Modelagem de software faz parte do processo de desenvolvimento de softwares. Um item essencial e indispensável na construção de uma base de dados.

Dentre suas vantagens se encontra: a fácil construção da base de dados, por atuar como uma espécie de “guia”; especificar detalhadamente todos os dados e como serão armazenados; auxiliar na tomada de decisões e por fim visualizar o sistema como um todo.

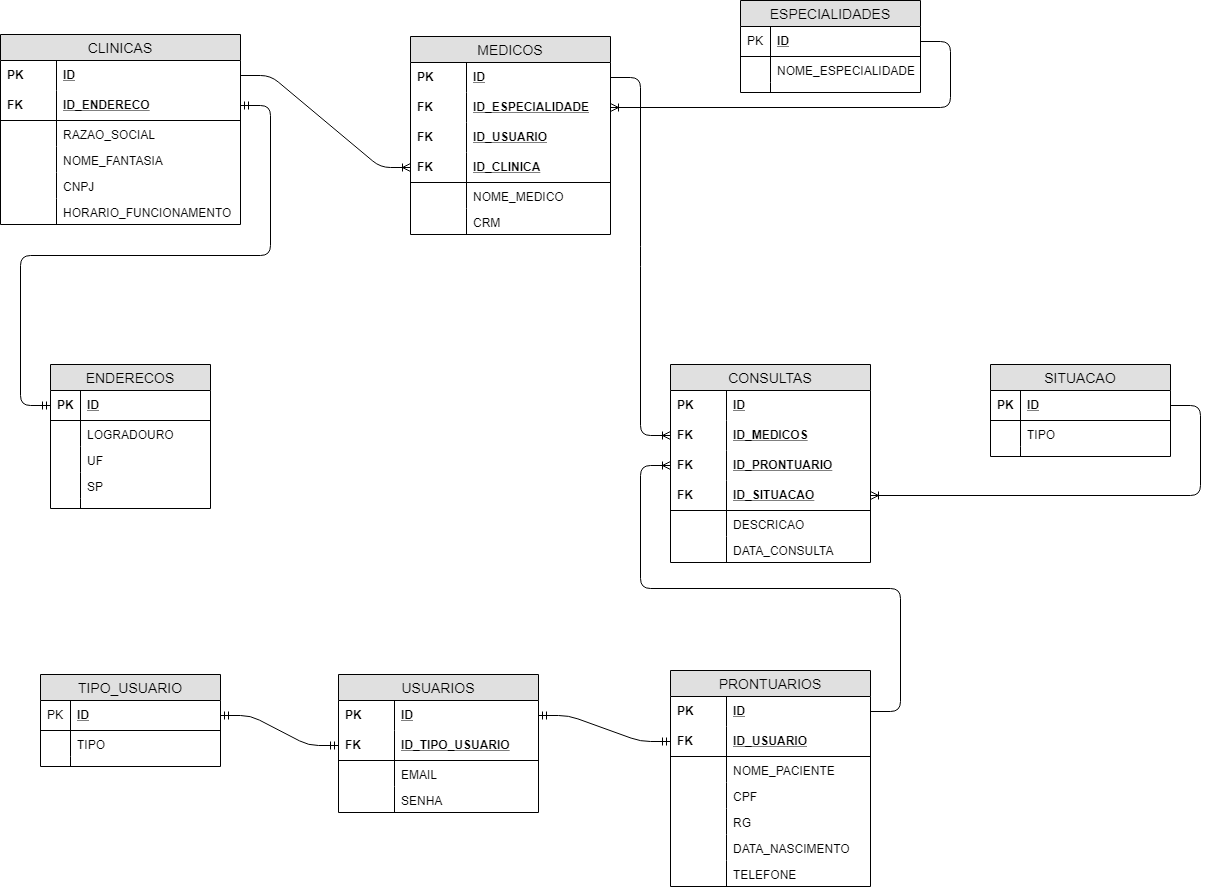
Modelo Conceitual

Nesse modelo todas as entidades e relacionamento são descritos de forma abstrata, apenas com entidades e cardinalidades, para fins de estruturação do banco de dados.



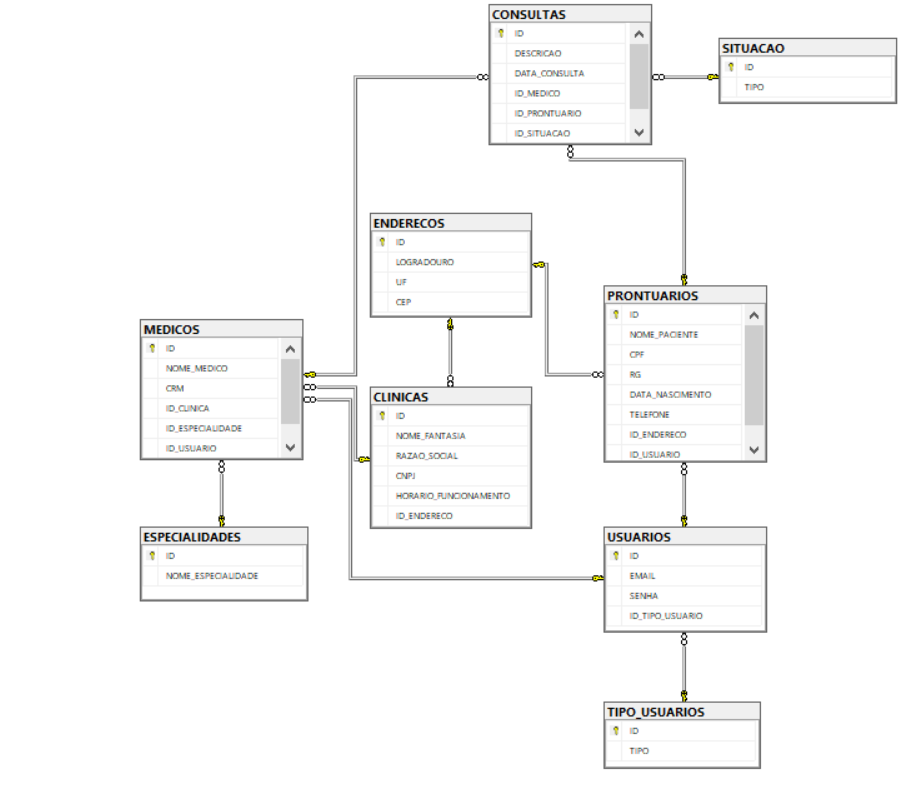
Modelo Lógico

No Modelo Lógico se detalha mais especificadamente o que as tabelas do banco de dados terão: as colunas que cada tabela terá (atributos); quais chaves possuem (primária, estrangeira) e seus respectivos relacionamentos com outras entidades.



Modelo Físico

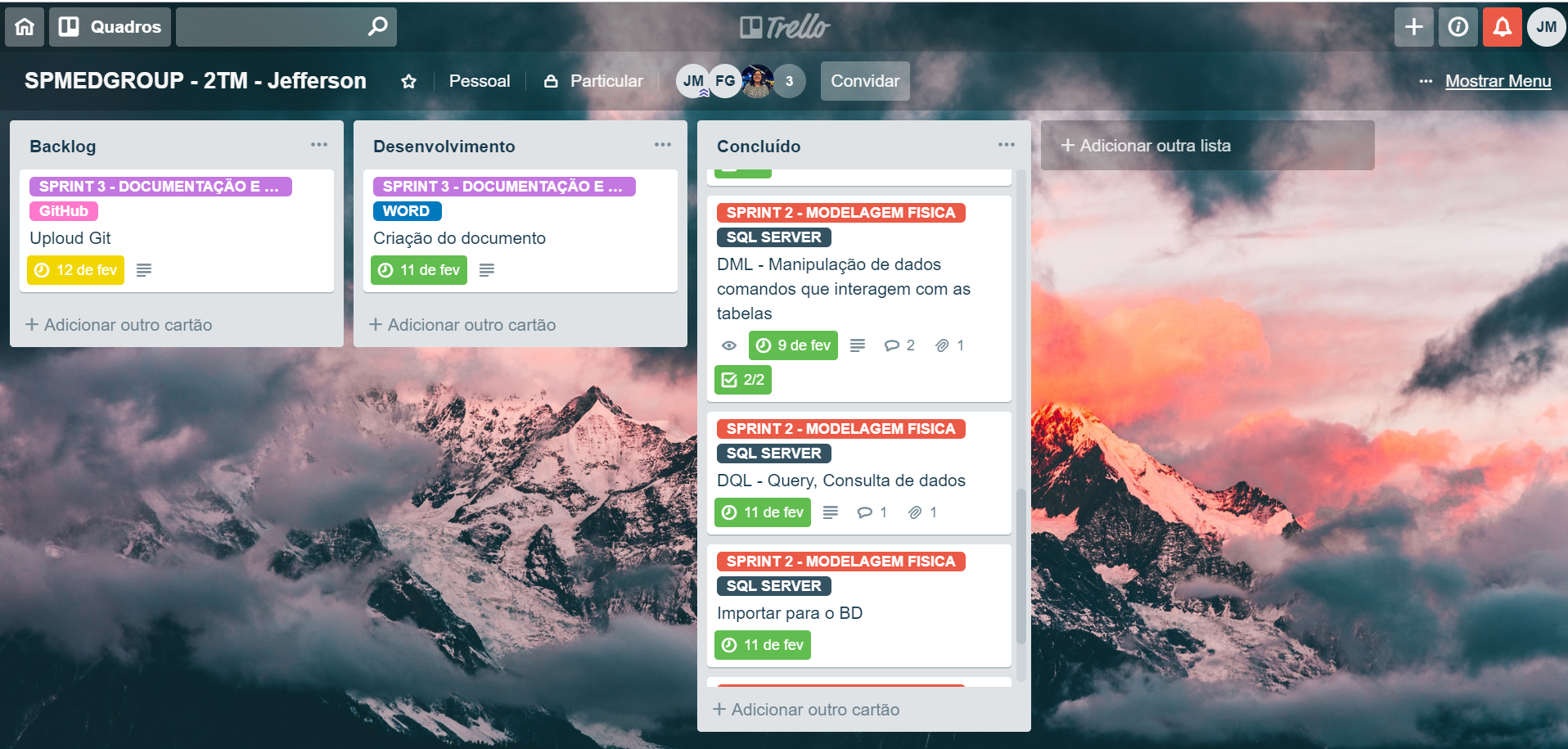
Se refere à construção de fato do banco de dados, a criação de tabelas e suas colunas, com determinado tipos de dados e restrições, sua construção pode se diferenciar dependendo do software usado para gerenciar o banco de dados.



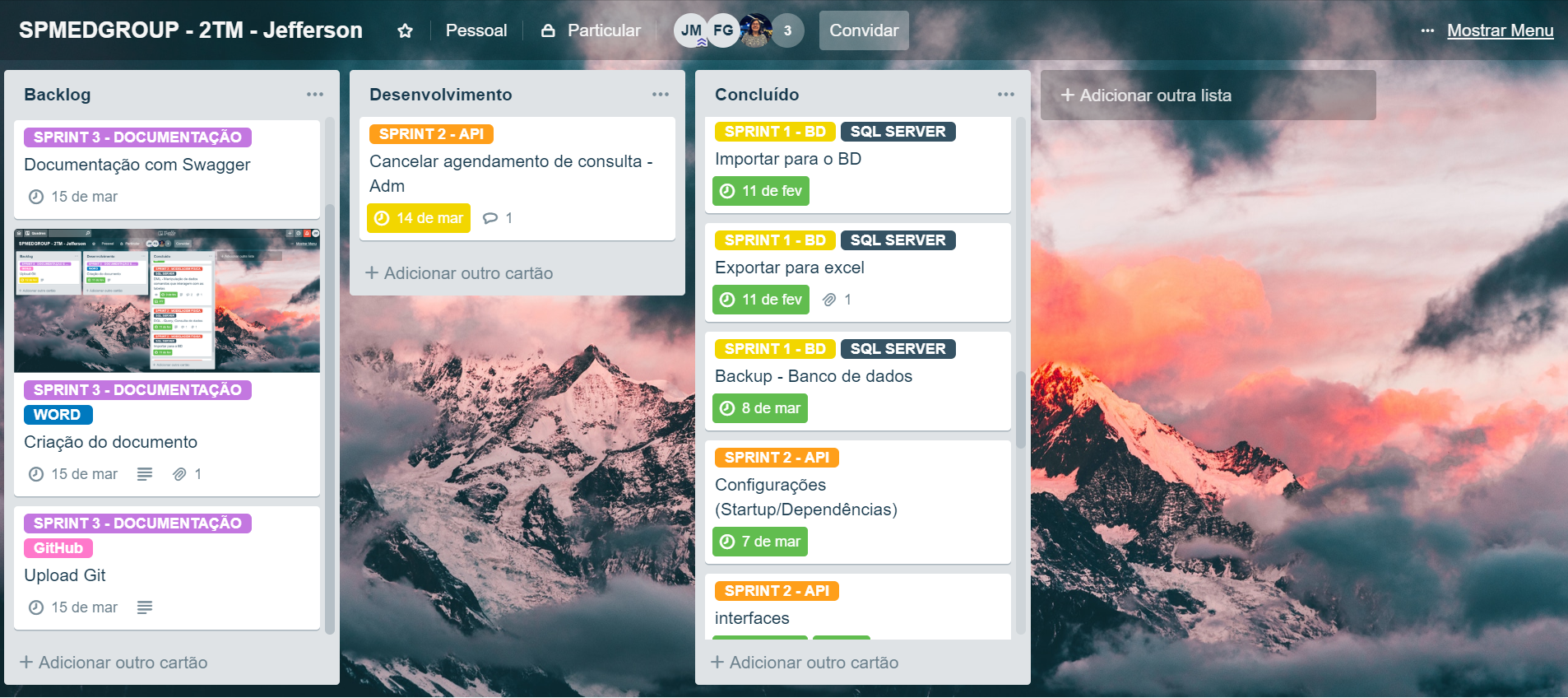
Trello (Kanban)

**Link**: https://trello.com/b/QC0AYSir/spmedgroup-2tm-jefferson

Sprint 1 – Banco de dados:



Sprint 2 – WebApi:



Cronograma

Sprint 1 – Banco de dados:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 | Dia 6 |
| **Modelo** **Conceitual** | **x** | **x** |  |  |  |  |
| **Modelo Lógico** |  | **x** | **x** |  |  |  |
| **Modelo Físico** |  |  | **x** | **x** | **x** |  |
| **Procedures** |  |  | **x** | **x** | **x** |  |
| **Ddl** |  |  |  | **x** | **x** |  |
| **Dml** |  |  |  | **x** | **x** |  |
| **Dql** |  |  |  | **x** | **x** |  |
| **Import/Export** |  |  |  |  | **x** |  |
| **Documentação** |  |  |  |  |  | **X** |

Sprint 2 – WebApi:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Dia 8 | Dia 9 | Dia 10 | Dia 11 | Dia 12 | Dia 13 | Dia 14 | Dia 15 |
| **Planejamento/ Configurações Inicais** | **x** | **x** |  |  |  |  |  |  |
| **EndPoints** |  | **x** | **x** | **x** | **x** | **x** |  |  |
| **Autorização** |  |  |  |  | **x** | **x** |  |  |
| **Autenticação** |  |  |  |  |  | **x** | **x** | **x** |
| **Validações** |  |  |  |  |  |  | **x** | **x** |
| **Documentação** |  |  |  |  |  | **x** | **x** | **x** |

Funcionalidades

Detalha todas as funções do sistema, suas permissões e ações esperadas.

Endpoints

* **Login**

Neste endpoint será feito a autenticação do usuário para acessar o sistema e suas devidas permissões para certas funcionalidades.

* **Lista de Usuários (Administradores/Médicos/Pacientes)**

Tem como objetivo visualizar todos os usuários do sistema tanto administradores, pacientes e médicos cadastrados com suas respectivas informações. Somente administradores podem visualizar estas informações.

* **Cadastro de consultas**

Cadastra uma nova consulta, informando os dados da consulta como por exemplo, a data. Realizada por administradores.

* **Lista de consultas (todas/somente de um paciente/somente de um médico)**

Tem como objetivo auxiliar na visualização das consultas referentes à apenas um usuário seja ele médico, paciente ou até mesmo de um administrador. Os administradores visualizam todas as consultas, já os pacientes e médicos apenas podem ter acesso as consultas referentes a eles.

* **Cancelar/Alterar o status do agendamento**

Administradores podem alterar a situação de determinada consulta para cancelada, realizada ou agendada. Deletar uma consulta resultaria na perda dos históricos de consultas no banco de dados para futuras possíveis visualizações para diversos afins**.**

* **Incluir descrição da consulta (Médicos)**

Médicos podem adicionar uma descrição na consulta detalhando os procedimentos, exames realizados e quaisquer observações pertinentes à consulta e o estado do paciente.

* **Cadastrar clínicas**

Incluir nova clínica no banco de dados caso novas filiais sejam criadas.

* **Listar clínicas**

Permite a visualização das clínicas existentes.

Passo a passo

Detalha como foi feito a aplicação e como utilizá-la.

Database First

**Banco de dados**

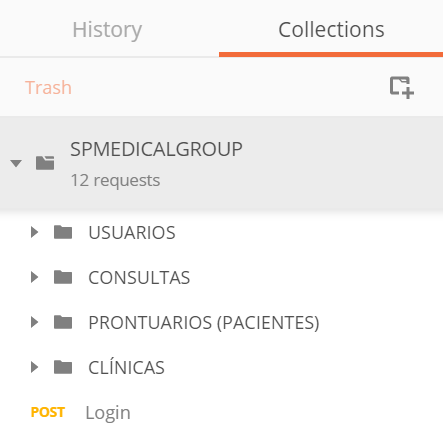
1. Abrir Microsoft SQL Server Management Studio
2. Selecionar os arquivos (scripts) **SPMEDGROUP\_DDL\_D1 e SPMEDGROUP\_DML\_D2.**
3. Executar Scripts (DDL e DML)

**Projeto (Back-end)**

1. Abrir o Visual Studio 2017
2. Selecionar o projeto **Senai.SpMedicalGroup.WebApi.**
3. Clicar na aba Ferramentas e selecionar **Gerenciador de Pacotes NuGet > Console do Gerenciador de Pacotes**
4. Copiar o seguinte comando:
   * Scaffold-DbContext "Data Source=.\SqlExpress;Initial Catalog=SPMEDGROUP;User Id=sa;pwd=132" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Domains -ContextDir Contexts -Context SpmedgroupContext

**Observação:** O user e a senha deve ser a mesma utilizada para aceder ao Banco de Dados.

1. Importar coleção contendo as URL no Postman (para testar funcionalidades):
   * Botão Import > Choose Files > Escolher arquivo **SPMEDICALGROUP.postman\_collection**

****

1. Copiar e abrir a seguinte URL (para documentação):

<http://localhost:5000/swagger>

UI/UX – User Interface, User Experience

Enquanto a UI (Interface do Usuário) se refere a parte visual do projeto, a UX (Experiência do Usuário) se atenta ao uso e toda a vivência do usuário com o produto desenvolvido.

Objetivos

Tanto a UI como a UX visam conquistar o usuário, para isso se utiliza de conceitos que permitem avaliar o que melhor agrada o usuário, fazendo com que este consuma seus serviços e produtos e tenha uma experiência satisfatória.

Mobile e Web

O projeto conta com dois layouts de alta fidelidade desenvolvidos, um contendo todas as telas do sistema para web e outro para mobile. Exibe os redirecionamentos e todas as funcionalidades do sistema.

Os arquivos estão na pasta “Layouts”.

Testes

A fase de testes é essencial para qualquer parte do desenvolvimento do sistema, e para a fase de design não seria diferente. Como impacta diretamente o usuário final, esta etapa é crucial para o projeto e por isso deve ser realizado diversos testes para analisar se o projeto atende as expectativas, antes que este chegue ao usuário final.

Os testes foram realizados na ferramenta de prototipação do adobe XD, sendo ouvida a opinião de diversos usuários até se chegar a um consenso sobre o que seria mais prático, seguro e visualmente agradável.

Front-end

Responsável pela aplicação web, transformando o design da Sprint anterior em telas dinâmicas.

React

A biblioteca javascript React possibilita a reutilização de componentes, facilidades na criação de UI, rápido desenvolvimento (por ser fácil de se aprender) e está em alta no mercado, pontos que o credencia para este projeto.

Comandos

* + **Comando para iniciar o projeto:**

npm start

* + **Comando para instalar dependências:**

npm i

npm install axios

npm install react-bootstrap bootstrap

npm install react-router-dom

Mobile

O aplicativo para dispositivos móveis foi desenvolvido pensando na praticidade dos usuários para visualizar suas consultas.

React-Native

React Native é um framework híbrido que funciona em várias plataformas (Android, IOS) e possui como vantagens custo mais baixo, custo. Além de possuir funcionalidades como live reload, utilização de componentes e integração com código nativo que facilitam o desenvolvimento

Android Studio (AVD)

Para testar o app, se utilizou a AVD (Android Virtual Device) do Android Studio, que nada mais é do que um emulador que simula um celular e todas suas funções. Há também a opção de testar diretamente no celular, habilitando as opções de desenvolvedor.

Comandos

* + **Comando para iniciar o projeto:**

React-native run-android

* + **Comando para instalar dependências:**

npm i

npm install axios

npm install jwt-decode

npm install react-native-cli

npm i react-native-gesture-handler

npm i react-navigation

Data Analytics

A funcionalidade de Data Analytics tem como objetivo desenvolver um sistema de localização de todos os seus usuários. Um dos pilares do Big Data é realizar os desejos dos clientes antes que os mesmos peçam, imaginar o que eles precisam e querem, ofertando conforto, evitando reclamações e estrategicamente utilizando dados para melhorar ou criar serviços. Nele há funcionalidades para realizar o cadastro de localizações e qual tipo de atendimento médico está sendo feito. O foco da empresa é fazer um levantamento sobre quais regiões estão sendo atendidas por mais especialidades de médicos.

Vantagens

O termo NoSQL (Not Only Structured Query Language) diz que ainda podem suportar a linguagem sql. É uma inovação disruptiva, direcionada para o universo da Big Data que cria mercados, novos públicos e acirra a disputa com concorrentes (outrora dominantes no segmento), não causou a extinção do Sql pois ainda não atingiu os mesmos níveis de consistência e confiabilidade (no que se diz a integridade dos dados, pois é relativamente nova e ainda não há toda a documentação e suporte necessários para se assegurar um certo grau de confiabilidade) se trata de uma solução que pode ser mais simples, com menor custo, ou até mesmo mais eficiente. Um banco de dados não relacional possibilita que não se necessite de uma estrutura prévia para guardar as informações e relacioná-las. É preciso somente realizar o registro em um formato que o banco "entenda", no caso do MongoDB, JSON. Escalabilidade (diz a respeito de “estar preparado para crescer”), flexibilidade (tanto na estruturação do banco, como na inserção de novos campos) oferecendo agilidade para a equipe de desenvolvimento e velocidade (melhor performance para consultas).

Firebase

  O limite de requisições em um plano gratuito é um ponto negativo, porém pensando em ambiente de testes, o firebase se adequa perfeitamente por ser um BaaS (backend as a service), “pulando” a parte do Back-end e indo direto ao ponto. Por isso, provendo rápida implementação, dispõe de diversas ferramentas e é facilmente escalável.

Referências

Links

<https://docs.microsoft.com/pt-br/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-2017>

<https://www.w3schools.com/sql/sql_count_avg_sum.asp>

<https://pt.stackoverflow.com/questions/294699/qual-a-diferen%C3%A7a-entre-modelagem-conceitual-l%C3%B3gica-e-f%C3%ADsica>

https://blog.caelum.com.br/rest-principios-e-boas-praticas/

<https://www.youtube.com/watch?v=vGuqKIRWosk>

https://www.youtube.com/watch?v=UqzQwR-L-OQ

<https://docs.microsoft.com/en-us/aspnet/core/security/authorization/roles?view=aspnetcore-2.2>

<https://docs.microsoft.com/pt-br/dotnet/api/system.componentmodel.dataannotations.datatype?view=netframework-4.7.2>

<https://www.w3schools.com/colors/colors_picker.asp>

<https://color.adobe.com/pt/create/color-wheel/?base=2&rule=Custom&selected=1&name=icon-login&mode=rgb&rgbvalues=0,0,1,0.5098039215686274,0.7568627450980392,0.8431372549019608,0.5019607843137255,0.8588235294117647,0.5725490196078431,0.5058823529411764,0.8705882352941177,0.6,0,0,0&swatchOrder=0,1,2,3,4>

<https://www.moblee.com.br/blog/mobile-first-principais-vantagens/>

<https://theblog.adobe.com/10-dos-donts-mobile-ux-design/>

<https://www.colmeiatec.me/blog/aplicativos-hibridos-ou-nativos/>

<https://pt.stackoverflow.com/questions/191919/banco-de-dados-n%C3%A3o-relacionais-vs-banco-de-dados-relacionais>

<https://www.napratica.org.br/o-que-e-inovacao-disruptiva/>

<https://www.mundodevops.com/bancos-nosql-vs-relacionais/>