**PROJETO\_MEDICALS**

Documentação

* Resumo

Neste documento iremos mostrar um breve resumo sobre a documentação do projeto Medicals, seus conceitos e o banco de dados.

* Descrição do projeto

O projeto foi realizado com base em um pedido de um cliente onde foi solicitado um sistema para uma cliníca médica. O projeto consiste em dois diagramas: conceitual e lógico, um documento excel que é o modelo físico e o banco de dados feito no SQL Server.

* Modelagem de dados

Modelagem de dados se refere à criação de um modelo que explique as características e comportamento do software que será criado com base nas principais características que evitam erros de programação, projeto e operação, facilitando assim sua compreensão e projeto.

Modelo Conceitual



A modelagem conceitual ajudam a destacar conexões importantes em processos e sistemas no mundo real. Ele nos dá uma base para o modelo lógico.

Modelo Lógico



O modelo lógico se adequa ao modelo conceitual para o tipo de banco de dados que será implementado. Essa fase é necessária para adicionar os atributos nas entidades corretas.

Modelo Físico



Este modelo é o projeto fisico para implemantação de banco de dados.

Cronograma

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Dia 1 | Dia 2 | Dia 3 | Dia 4 | Dia 5 |
| Tarefa 1 | X |  |  |  |  |
| Tarefa 2 | X |  |  |  |  |
| Tarefa 3 |  |  |  | X | X |

<https://trello.com/b/bZmmyH5h/projeto-medicals>

* Back-End

A arquitetura utilizada para a criação do back-end foi a web Api, desenvolvido na IDE da Microsoft, o Visual Studio. A Api foi desenvolvida no formato HTTP, seguindo o padrão REST e feito em um modelo chamado JSON que serve para armazenamento e/ou transmissão de informações no formato de texto. Para ter uma noção maior sobre o desenvolvimento da API, leia:

**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.

Insira também um passo a passo de como executar a API criada e testar as requisições no Postman.