

SENAI Jaguariúna

Leticia Amaral Lopes

PROJETO AGROTEACH

Jaguariúna – SP

2023

Leticia Amaral Lopes

PROJETO AGROTEACH

Pré-Projeto da empresa AgroTeach trabalho de Desenvolvimento de Sistema apresentado ao SENAI Jaguariúna, como parte dos requisitos para obtenção da aprovação de Reenye e Wellington.

Jaguariúna – SP

2023

**Resumo**

O aplicativo AgroTeach é um projeto de internet fullstack com banco de dados relacional MySQL, BackEnd em Node.js, FrontEnd em JavaScript com framework e aplicativo móvel desenvolvido com React Native, consumindo pela API.Objetivo do site é para que os gerentes e os funcionários (motoristas), tenham um acesso acessível para ver seus dados e relatórios.

**SUMÁRIO**

[1. TAP (Termo de Abertura do Projeto) 8](#_Toc97670429)

[2. EAP (Escopo do Projeto) 14](#_Toc97670430)

[2.1. Requisitos não funcionais 14](#_Toc97670431)

[2.2. Requisitos Funcionais 14](#_Toc97670432)

[3. Planejamento (Cronograma Grafico Gantt) 9](#_Toc97670433)

[3.1. Planejamento de Custos 10](#_Toc97670434)

[4. Protótipo 11](#_Toc97670435)

[4.1. Protótipo do site da internet 11](#_Toc97670436)

[5. Planejamento do Banco de Dados e da API 18](#_Toc97670437)

[6. Detalhes da Execução e Controle 20](#_Toc97670438)

[7. Resultados 24](#_Toc97670439)

**Introdução**

Estamos fazendoum projeto paraos professores de uma empresa de agricultura com o nome de AgroTeach para poder facilitar os gerenciamentos de dados.

Usaremos o BackEnd para guardar os dados, junto com o FrontEnd para que os gerentes e funcionários tenham um acesso acessível para acessar os dados.

O mobile será usado para conseguirem acessar as operações e manutenções, que será exclusive para os gerentes.

**Justificativa**

Ao concluir esse projeto da empresa AgroTeach, vemos uma plataforma acessível para os funcionários da empresa para poder ter segurança nos dados e facilidade para acessá-los, claro que precisamos melhora-lo a cada dia para poder a chegar a um nível de qualidade.

**Objetivos**

O objetivo do projeto é aprimorar o controle veicular, auxiliando na tomada de decisões estratégicas aumentando a competividade no mercado. O sistema terá como recurso principal o sistema de relatórios onde será necessário obter informações como manutenções do veículo, disponibilidade para uso, alocação atual e motorista responsável.

Este projeto cumpre os seguintes objetivos específicos:

* Visualizar o as informações especificas para gerentes.
* Segurança, no login para identificar gerentes e motoristas.
* Adicionar e alterar recursos.
* Relatórios de disponibilidade, manutenção, alocação.
* Todos recursos ser utilizados no mobile.

# TAP (Termo de Abertura do Projeto)

**Título do Projeto:**

Projeto AgroTeach “Site e Aplicativo Móvel capaz de gerenciar frotas de veículos, aprimorar o controle veicular, auxiliando na tomada de decisões estratégicas aumentando a competividade no mercado. ”

**Patrocinadores:** Reenye Alexandre de Lima

|  |  |
| --- | --- |
| Recursos Humanos | Letícia Amaral |
| Gerente do Projeto | Letícia Amaral |
| Patrocinador | Reenye Alexandre de Lima  Wellington Fábio de Oliveira Martins |
| Cliente | Reenye Alexandre de Lima |
| Prazo | 17/03/2023 |

|  |  |
| --- | --- |
| Local:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Data:\_\_\_\_/\_\_\_\_/\_\_\_\_ | |
| Patrocinador | Ass: |
| Cliente | Ass: |
| Gerente do Projeto | Ass: |

# 2.Planejamento (Cronograma Grafico Gantt)

Planejamento e gestão do tempo e recursos humanos.

O backlog do projeto está dividido em atividades e segue apresentado utilizando um gráfico Gantt.

Este documento apresenta apenas uma versão intermediária do planejamento, durante a execução e controle sob a metodologia Scrum, a cada Sprint o cronograma foi atualizado.

Obs:Preciso adicionar, acabei deixando no pc do Senai...

## **2.1. Planejamento de Custos**

Para calcular os custos deste projeto basta somar as horas dedicadas conforme o cronograma, sabendo que são quatro horas por dia útil, todos os dias registrados são considerados dias úteis, e multiplicar por um valor estipulado por hora, como os valores monetários são voláteis, neste projeto não serão representados. A seguir temos um exemplo de como seria calculado o custo com um valor ilustrativo.

Leticia 60 horas trabalhadas, totalizando 60 horas de projeto, caso cada recurso custasse R$20,00 por hora o custo total deste projeto seria de: R$ 1200,00

## **2.1. Protótipo**

As imagens a seguir foram geradas utilizando a técnica de prototipagem para apresentar um esboço das telas do site da internet.

## **2.2. Protótipo do site da internet**

As imagens mostram a sequência de telas que um usuário do site deve acessar para usufruir de todas as funcionalidades da aplicação.

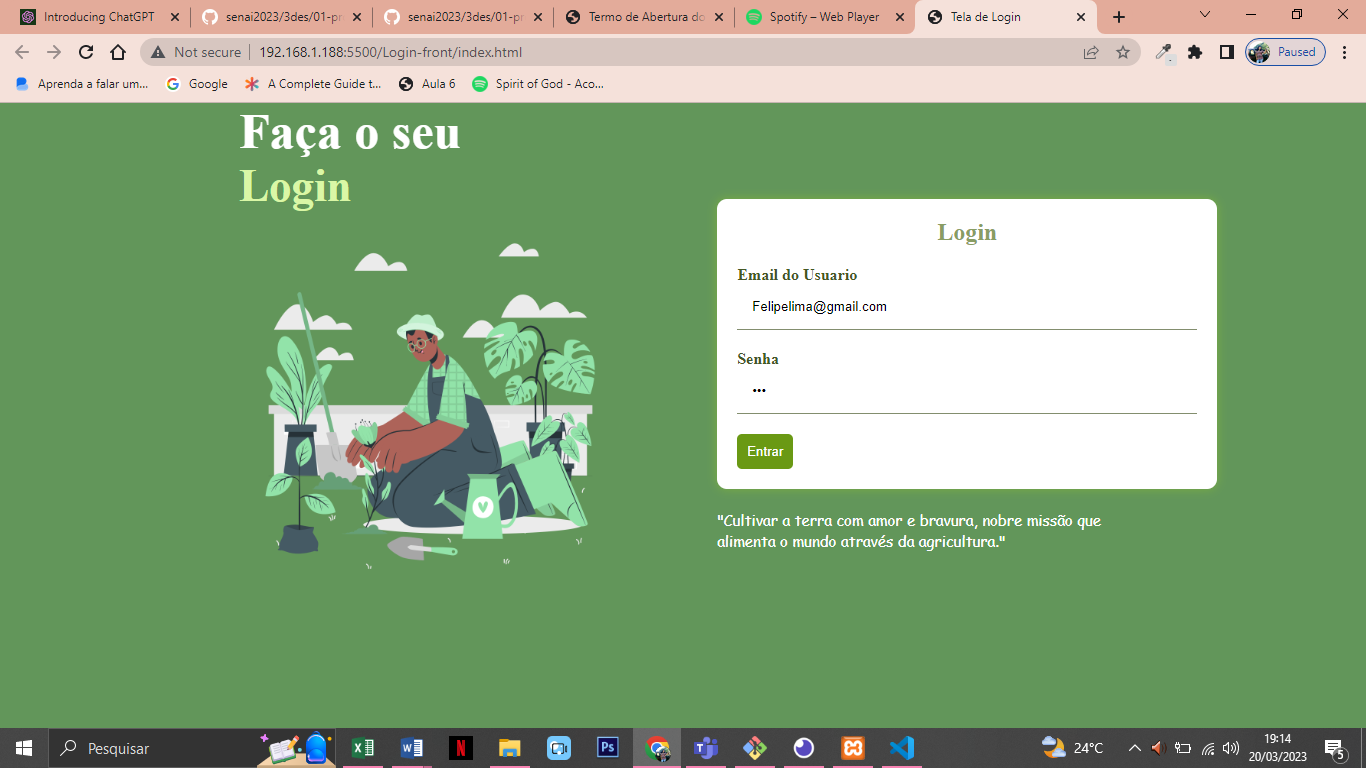


Figura 01 – Login do gerente

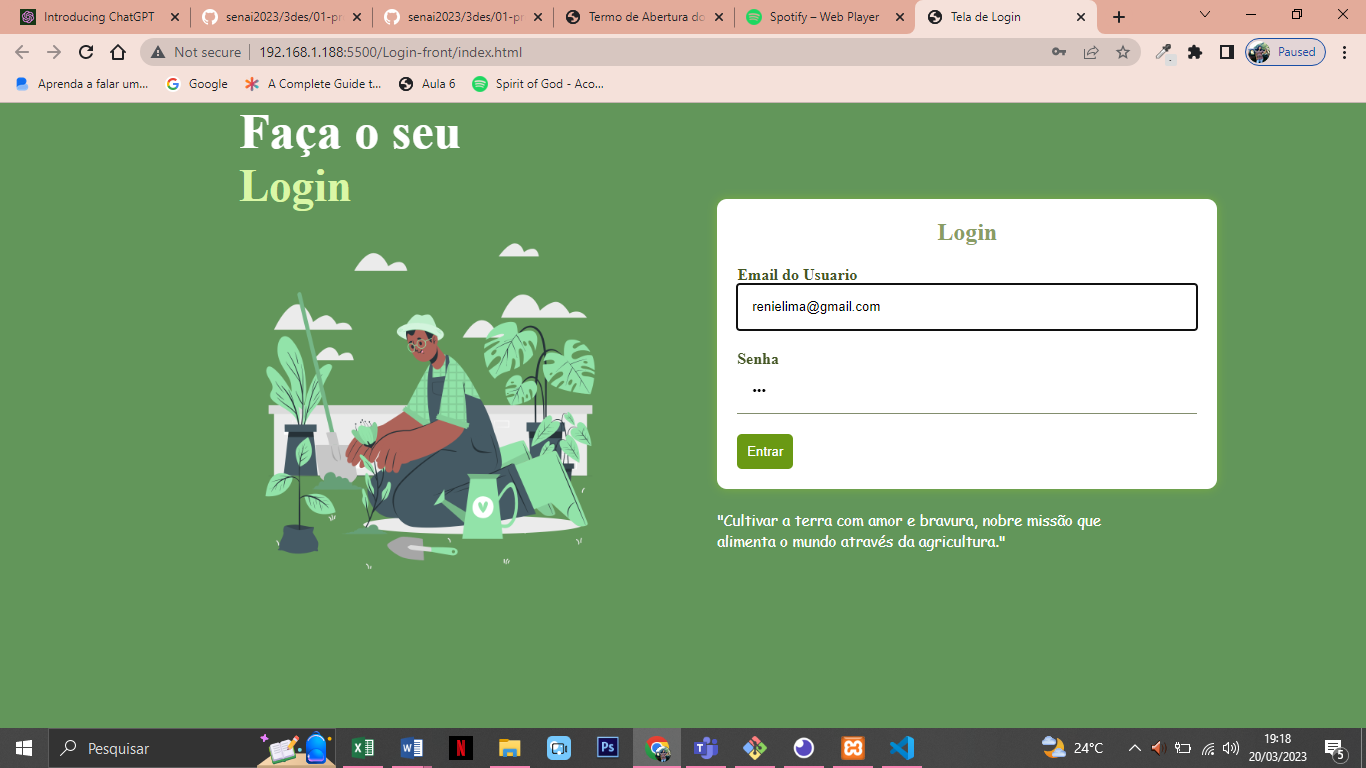


Figura 02 – Login do motorista

Obs: Meu computador deu bugs no código e não me deixa acessar para tirar print do restante das fotos do site.

Requisitos de Documento

1. CONVENÇÕES, TERMOS E ABREVIAÇÕES:

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

IDENTIFICAÇÃO DOS REQUISITOS Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos seguidos do identificador do requisito, de acordo com a especificação a seguir: [nome da subseção, identificador do requisito] Por exemplo, o requisito funcional [Incluir Usuário.RF016] deve estar descrito em uma subseção chamada “Incluir Usuário”, em um bloco identificado pelo número [RF016].

Já o requisito não-funcional [Confiabilidade.NF008] deve estar descrito na seção de requisitos não-funcionais de Confiabilidade, em um bloco identificado por [NF008]. Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001] ou [NF001] e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

Cada requisito deve fazer referência a uma regra de negócio [RN001].

PROPRIEDADES DOS REQUISITOS Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”.

Essencial é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente. Importante é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

Desejável é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

1. DESCRIÇÃO GERAL DO PROJETO:

Será um sistema web e mobile para o gerenciamento de veículos, com recursos principais diretamente para o sistema de relatórios, onde será necessário obter informações como manutenções.

1. REGRAS DE NEGÓCIO:
2. 1.[RN01] Autenticação

-- O sistema deverá gerenciar o acesso dos usuários ao sistema, limitando a experiência de acordo com seu nível de acesso.

* 1. [RN02] Gerenciamento dos motoristas:

--Será necessário a implementação de funcionalidades que permitam ao usuário inserir, atualizar, excluir e listar utilizando filtros os registros dos motoristas

* 1. [RN03] Gerenciamento da frota:

-- Será necessário a implementação de funcionalidades que permitam ao usuário inserir, atualizar, excluir e listar utilizando filtros os registros dos veículos pertencentes as frotas.

* 1. [RN04] Manutenção veicular:

-- Para melhor controle, todas as manutenções realizadas deverão ser registradas, contendo informações como data de início e fim da manutenção, valor gasto e descrição da manutenção.

* 1. [RN05] Operações veiculares:

-- Dentro da frota existem veículos de carga, visita e vendas. As operações deverão ser registradas para controle de trabalho. Algumas informações importantes para registro são: motorista responsável, data de saída e retorno e descrição do serviço.

* 1. [RN06] Dashboard administrativa:

-- Os relatórios deverão possuir gráficos e resumos de fácil visualização e compreensão.

# 4.EAP (Escopo do Projeto)

O escopo de um projeto de aplicação fullstack, para a internet e para dispositivos móveis consiste de seus requisitos funcionais e não funcionais, este projeto, portanto está dividido desta forma.

**4.0.** **R**equisitos **N**ão **F**uncionais

[RN01] Linguagem de programação:

As linguagens de programação utilizadas no desenvolvimento deste projeto são: BackEnd Java(“A onde iremos armazenar o banco de dados e molda-la”), o Front-End, iremos fazer o visual com o HTML que é a página, css que irá personalizar, e o Javascript irá realizar as funções da página, e Mobile teremos o visual e funcionalidades para androides – celulares.

## **4.1**. **R**equisitos **F**uncionais

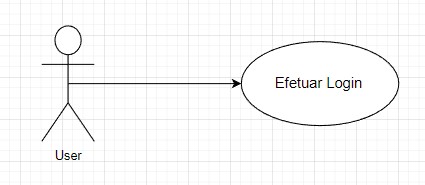
Os requisitos funcionais serão apresentados utilizando o recuso UML Diagrama de Casos de Uso e uma classificação de criticidade.

**4**.**2**. REQUISITOS FUNCIONAIS (CASOS DE USO):

4.3. [RN01] Autenticação:

Prioridade: () Essencial (x) Importante ( ) Desejável

O usuário irá fazer o login para o sistema verificar e limitar a experiência de acordo com seu nível de acesso.

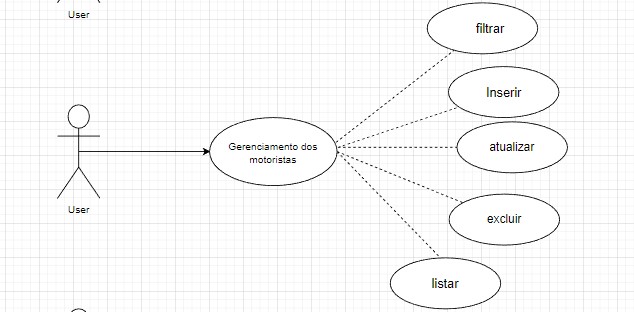


Atende as regras [RN01].

4.4. [RN02] Gerenciamento dos motoristas:

Prioridade: () Essencial (x) Importante ( ) Desejável

Os usuários que são os motoristas, quando se cadastrar, terão o direito de alterar, excluir, listar e filtrar.

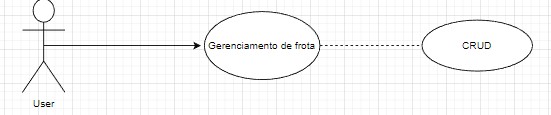


Atende as regras [RN02].

4.5. [RN03] Gerenciamento da frota:

Prioridade: () Essencial (x) Importante ( ) Desejável

Os usuários vão fazer um cadastramento para o estacionamento, com todas as funcionalidades de CRUD e o direito de filtrar veículos pertencentes as frotas.

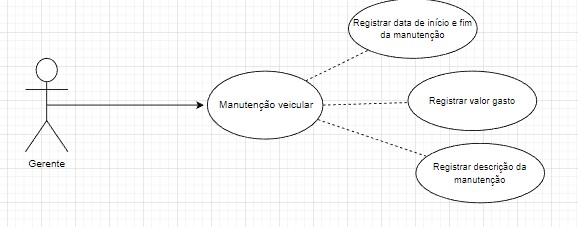


Atende as regras [RN03].

4.6. [RN04] Manutenção veicular:

Prioridade: () Essencial (x) Importante ( ) Desejável

Para ter controle em tudo o gerente irá registrar todas as manutenções.

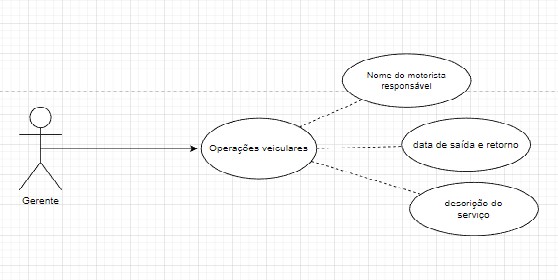


Atende as regras [RN04].

4.7. [RN05] Operações veiculares:

Prioridade: () Essencial (x) Importante ( ) Desejável

O gerente irá registrar todas as informações da frota de veículos.

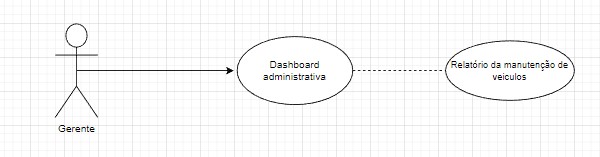


Atende as regras [RN05].

4.8. [RN06] Dashboard administrativa:

Prioridade: () Essencial () Importante (x) Desejável

Através do Excel teremos gráficos e resumos de fácil visualização e compreensão do trabalho.

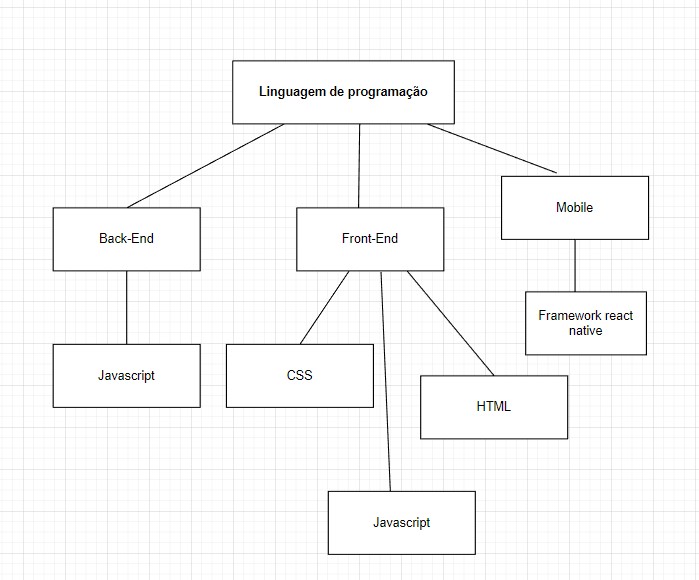


Atende as regras [RN06].

1. REQUISITOS NÂO FUNCIONAIS:

5.1. [RN01] Linguagem de programação:

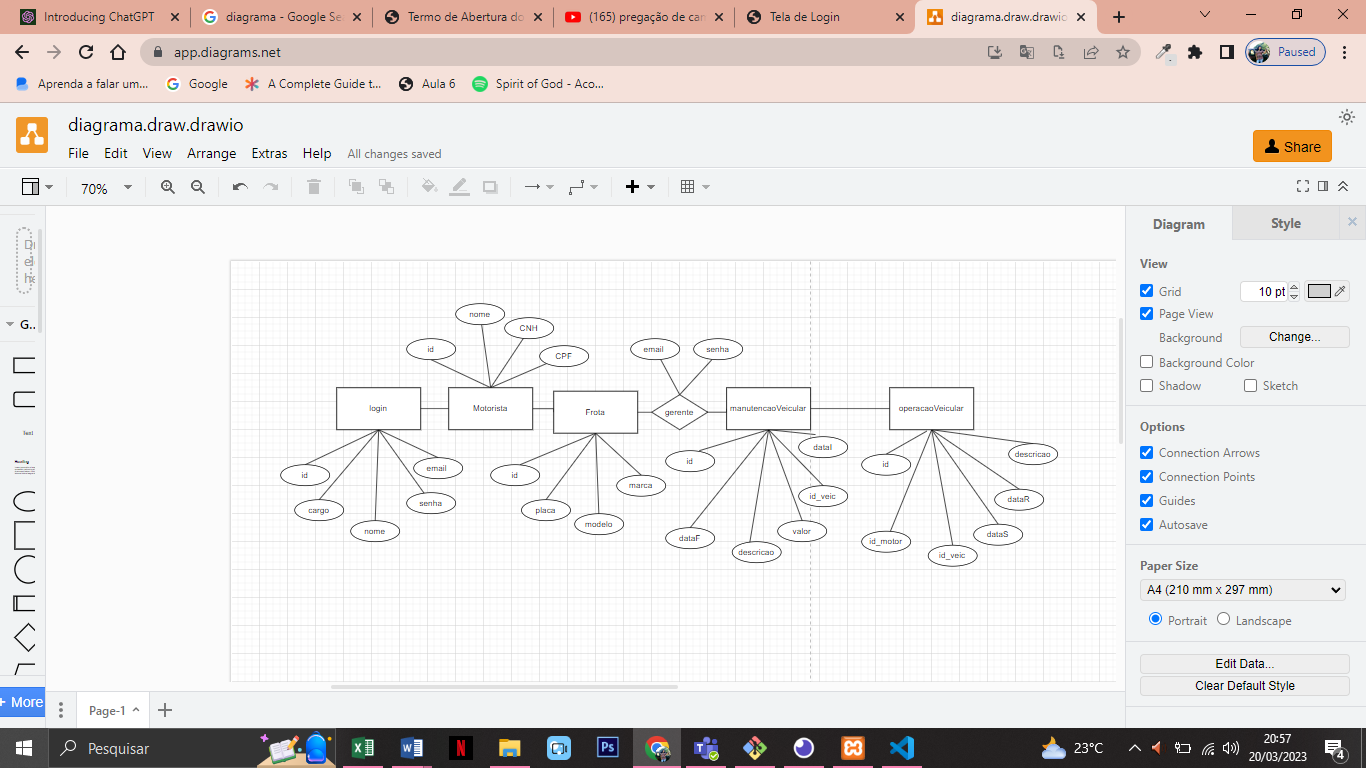
As linguagens de programação utilizadas no desenvolvimento deste projeto são: BackEnd Java(“A onde iremos armazenar o banco de dados e molda-la”), o Front-End, iremos fazer o visual com o HTML que é a página, css que irá personalizar, e o Javascript irá realizar as funções da página, e Mobile teremos o visual e funcionalidades para androides – celulares.



Atende as regras [RN01].

# 6. Planejamento do Banco de Dados e da API

O backend deste projeto se comporta como uma API, fornecendo dados tanto para o frontend quanto para o aplicativo móvel, esta sessão documenta o projeto do banco de dados através do MER (Modelo Entidade e Relacionamento) e DER (Diagrama de Entidade e Relacionamento) conceitual e lógico.



# 7.Detalhes da Execução e Controle

O projeto foi desenvolvido quase dois meses e meio de aula, individualmente.

Individualmente tivemos que desenvolver a acamada de backend e banco de dados, utilizei as ferramentas de diagrama de classes e dicionário de dados para registrar quais informações necessita que a API / backend deve receber e fornecer.

A seguir segue o dicionário de dados no formato JSON que é o padrão de dados oficial de troca de informações entre APIs.

Dicionário de dados da classe Login:

"01-verbo": "Post",

"01-rota": "/usuario",

"01-status": 200,

"01-response": [

{

"id": 1,

"nome": "Renie",

"email": renielima@gmail.com,

"senha": 123,

"cargo": “Funcionario”

},

{

"id": 2,

"nome": "Felipe",

"email": felipelima@gmail.com,

"senha": 123,

"cargo": “Gerente”

}

Dicionário de dados da classe Motorista:

"01-verbo": "Post",

"01-rota": "/motorista",

"01-status": 200,

"01-response": [

{

"id": 1,

"nome": "Carlos",

"CNH":567845382,

"CPF":56435674

},

"02-verbo": "GET",

"02-rota": "/motorista",

"02-status": 200,

"02-response": [

{

"id": 1,

"nome": "Carlos",

"CNH":567845382,

"CPF":56435674

},

"03-verbo": "PUT",

"03-rota": "/motorista/1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

"id": 1,

"nome": "Carlos",

"CNH":567845382,

"CPF":56435674

},

"04-verbo": "DELETE",

"04-rota": "/motorista/1",

"04-true-status": 200,

"04-false-status": 404

}

Dicionário de dados da classe Frota:

"01-verbo": "Post",

"01-rota": "/frota",

"01-status": 200,

"01-response": [

{

"id": 1,

"placa": "AbC34",

"modelo": "Fox Verde",

"marca": "Fox"

},

"02-verbo": "GET",

"02-rota": "/ frota ",

"02-status": 200,

"02-response": [

{

"id": 1,

"placa": "AbC34",

"modelo": "Fox Verde",

"marca": "Fox"

},

"03-verbo": "PUT",

"03-rota": "/frota/1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

"id": 1,

"placa": "AbC34",

"modelo": "Fox Verde",

"marca": "Fox"

},

"04-verbo": "DELETE",

"04-rota": "/frota/1",

"04-true-status": 200,

"04-false-status": 404

}

Dicionário de dados da classe Operação:

"01-verbo": "Post",

"01-rota": "/operacao",

"01-status": 200,

"01-response": [

{

"id": 1,

"id\_motorista": 1,

"id\_veiculo": 1,

"dataS": "20/03/2023",

"dataR": "",

"descricao": "monitorar"

},

"03-verbo": "Button de finalizer/PUT",

"03-rota": "/operacao/1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

},

"03-verbo": "/PUT",

"03-rota": "/operacao/1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

"descricao": "",

},

"02-verbo": "GET",

"02-rota": "/ operação ",

"02-status": 200,

"02-response": [

{

"id": 1,

"id\_motorista": 1,

"id\_veiculo": 1,

"dataS": "20/03/2023",

"dataR": "20/03/2023",

"descricao": "monitorar"

}

Dicionário de dados da classe Manutenção:

"01-verbo": "Post",

"01-rota": "/registrar",

"01-status": 200,

"01-response": [

{

"id": 1,

"id\_veic": 1,

"dataI": "20/03/2023",

"dataF": "",

"valor": 60,

"descricao": "equipamentos"

},

"02-verbo": "GET",

"02-rota": "/ registrar ",

"02-status": 200,

"02-response": [

{

"id": 1,

"id\_veic": 1,

"dataI": "20/03/2023",

"dataF": "20/03/2023",

"valor": 60,

"descricao": "equipamentos"

},

"03-verbo": "PUT",

"03-rota": "/ registrar /1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

"valor": 120,

},

"04-verbo": "DELETE",

"04-rota": "/registrar /1",

"04-true-status": 200,

"04-false-status": 404

}

},

"03-verbo": "Button de finalizer/PUT",

"03-rota": "/registrar/1",

"03-true-status": 200,

"03-false-status": 404,

"03-response": [

{

"dataF": "",

},

# 8.Resultados

O projeto foi apresentado individualmente para os professores, a execução do sistema cumpriu quase todos os requisitos classificados como essenciais e importantes deste projeto.

Os arquivos fonte deste projeto estão compartilhados no GitHub através do endereço de internet a seguir: <https://github.com/Leticia567ds/Projeto2023/tree/main/Projeto%20S%C3%B3>.

Este é meu repositório público para consulta de todos alunos e professores.

**Bibliografia**

Chatgbt io:, Disponível em: < https://openai.com/ >, acessado quase todos os dias.

Github: WelliFabio, Disponível em: < https://github.com/wellifabio >, quase todos os dias.