



SENAI Jaguariúna

REENYE ALEXANDRE DE LIMA

WELLINGTON FÁBIO DE OLIVEIRA MARTINS

PROJETO BLITZ

Jaguariúna – SP

Março – 2022

REENYE ALEXANDRE DE LIMA
WELLINGTON FÁBIO DE OLIVEIRA MARTINS

PROJETO BLITZ

Projeto de conclusão parcial de curso apresentado ao SENAI Jaguariúna, como parte dos requisitos para obtenção do título de Técnico em Desenvolvimento de Sistemas.

Orientador:
Tatiana Maria Rolin

Jaguariúna – SP
Março – 2022

Lista de Figuras

Figura 1 - DCU - [RF001].....	15
Figura 2 - CDU - [RF002].....	15
Figura 3 - DCU - [RF003].....	16
Figura 4 - DCU - [RF001].....	17
Figura 5 - Cronograma - Gantt.....	18
Figura 6 - Legenda - Gantt.....	18
Figura 7 - Tela de Login.....	20
Figura 8 - Tela de Cadastro	21
Figura 9 - Tela de Cadastro Inserindo Imagem de Perfil	21
Figura 10 - Tela Principal (Home).....	22
Figura 11 - Tela Principal (Home - Satelite).....	22
Figura 12 - Inserindo um novo alerta	23
Figura 13 - Escolhendo o tipo de alerta	23
Figura 14 - Alterar o Perfil do Usuário	24
Figura 15 - Escolhendo os alertas ativos	24
Figura 16 - Listando os alertas gerados.....	25
Figura 17 - Login App	26
Figura 18 - Cadastro App.....	26
Figura 19 - Home App.....	27
Figura 20 - Home Satelite App.....	27
Figura 21 - Escolher alerta App	28
Figura 22 - Inserir alerta App	28
Figura 23 - Menu App	29
Figura 24 - Menu voltar App	29

Figura 25 - Perfil App	30
Figura 26 - Alertas ativos App.....	30
Figura 27 - Listar alertas App.....	31
Figura 28 - Modelo Conceitual.....	32
Figura 29 - Modelo Lógico	32
Figura 30 - Diagrama de Classes	33

Lista de Abreviaturas e Siglas

API	- Application Programming Interface (Interface de programação de aplicação)
BackEnd	- Prática de programação para Servidor de Aplicação e ou Banco de Dados.
FrontEnd	-Prática de programação para cliente de programas para internet.
FullStack	- Prática de programação em todas as camadas Cliente e Servidor
Mobile	- Prática de programação para dispositivos móveis também clientes de programas para a internet.
Google Maps	- site da internet https://www.google.com.br/maps com mapas e imagens de satélite de todo o planeta.
PMO	- Project Management Office (Escritório de Projetos).
JavaScript	- Linguagem de programação para a internet.
HTML	- Hyper Text Management Language, linguagem de marcação.
CSS	- Linguagem de marcação e estilização de páginas.
Framework	- Estrutura, conjunto de códigos genéricos capaz de unir trechos de um projeto.
Node.js	- Framework que utiliza a linguagem JavaScript no backend.
ORM	- Object Relacional Mapper (Mapa Objeto Relacional)
Sequelize	- recurso ORM do framework Node.js
UML	- Unified Modeling Language (Linguagem de Modelagem Unificada)
Ágil	- Metodologia Ágil (Metodologia de desenvolvimento de sistemas)
SCRUM	- Subcategoria da Metodologia Ágil.
Backlog	- Requisitos funcionais do projeto a serem desenvolvidos.
Sprint	- tempo de desenvolvimento de duas a quatro semanas.
Gantt	- Grafico para gestão do tempo e recursos do projeto.

MER	- Modelo Entidade e Relacionamento
DER	- Diagrama de Entidade e Relacionamento
MVC	- Modelo Visão e Controle
JSON	- Objeto JavaScript, padrão de dados de uma API
GitHub	- Repositório para códigos fonte e rede social de programadores.

Resumo

O aplicativo Blitz é um projeto de internet fullstack com banco de dados relacional MySQL, BackEnd em Node.js, FrontEnd em JavaScript sem framework e aplicativo móvel desenvolvido com React Native, consumindo a API do Google Maps.

O objetivo é auxiliar condutores de veículos a alertas seus colegas sobre ocorrências de trânsito, como congestionamentos, acidentes, radares e outras fazendo assim um percurso mais rápido em suas viagens.

Além disso servir de modelo para os projetos desenvolvidos pelos alunos do curso Técnico em Desenvolvimento de Sistemas do SENAI Jaguariúna.

Abstract

Blitz application is a fullstack internet project that use MySQL relational database, BackEnd in Node.js, JavaScript language in FrontEnd without framework, mobile app in React Native consuming the Google Maps API that aims to help vehicle drivers to alert their colleagues about traffic occurrences, such as traffic jams, accidents, speed cameras and others, making your travels faster.

In addition to serving as a model for projects developed by students of the Technical Course in Systems Development at SENAI Jaguariúna.

SUMÁRIO

1.	TAP (Termo de Abertura do Projeto)	13
2.	EAP (Escopo do Projeto)	14
2.1.	Requisitos não funcionais	14
2.2.	Requisitos Funcionais	14
3.	Planejamento (Cronograma Grafico Gantt)	18
3.1.	Planejamento de Custos	19
4.	Protótipo	20
4.1.	Protótipo do site da internet	20
5.	Planejamento do Banco de Dados e da API.....	32
6.	Detalhes da Execução e Controle	34
7.	Resultados.....	40

INTRODUÇÃO

A turma intitulada 3DES do curso técnico em desenvolvimento de sistemas da escola SENAI Jaguariúna com conclusão do curso prevista para o mês de agosto do ano de 2022 tem se dedicado a soluções de problemas através de sistemas computacionais, também a automação de processos repetitivos utilizando recursos e bibliotecas das linguagens de programação como Python, planilhas eletrônicas e APIs disponíveis de forma gratuita através da internet.

APIs segundo o site HostGator é a interface ideal para que os programas de diferentes organizações compartilhem suas ações, ferramentas, padrões e protocolo. Gerando, assim, uma total integração.

Neste contexto, como forma de construir um software FullStack como modelo para o que os alunos possam ter como base para desenvolver seus projetos de conclusão de curso. Este projeto foi desenvolvido e escrito pelos instrutores deste curso contando com a colaboração desta turma.

A solução está dividida em camadas chamadas de BackEnd, FrontEnd, Mobile e documentação, cada uma das camadas possui exemplos das linguagens de programação estudadas durante os dois primeiros semestres do curso e alguns recursos introduzidos neste terceiro e último semestre.

O objetivo do software, além de servir de modelo é a gestão de alertas de trânsito, utilizando o mapa obtido com dados da API do Google Maps, um condutor de veículo que tenha o aplicativo móvel instalado em seu smartphone poderá observar eventos no trânsito e notifica-los simplesmente apontando no mapa onde o evento está ocorrendo e classificando-o como blitz “Fiscalização ou investigação policial em andamento”, fenômenos da natureza como enchentes, alagamentos, queda de árvores, congestionamentos, lentidão, fiscalização eletrônica ou qualquer outra natureza de eventos, gerando assim um alerta para outros condutores que passem pelo local e utilizem o aplicativo.

Este trabalho está dividido seguindo os princípios e instruções de gestão de projetos PMO que segundo o PMBOK é uma estrutura organizacional que tem como objetivo padronizar processos de governança e gerenciamento de projetos.

Como forma de simplificar a lógica de programação foi adotada a linguagem JavaScript em todas as camadas do projeto, o framework utilizado no servidor de aplicação backend foi o Node.js. No frontend optou-se por não utilizar nenhum framework, mas sim recursos básicos das linguagens HTML, CSS e JavaScript.

Por fim para a programação de dispositivos móveis este modelo utilizou o framework React Native que também trabalha sobre a linguagem JavaScript.

OBJETIVOS

Desenvolver um aplicativo que gerencie alertas de trânsito, onde os condutores de veículos possam apontar no mapa eventuais acidentes, congestionamentos ou outros eventos, assim alertando outros usuários do aplicativo.

Este projeto cumpre os seguintes objetivos específicos:

- Alertar condutores sobre eventos de trânsito.
- Verificar melhores rotas evitando congestionamentos e acidentes.
- Servir de modelo para o desenvolvimento de soluções semelhantes.
- Servir de exemplo de consumo de API de terceiros
- Servir de exemplo de armazenamento de imagens em banco de dados
- Servir de exemplo de utilização de mapas via internet frontend e mobile.
- Servir de exemplo de utilização de recursos ORM como “sequelize”.

JUSTIFICATIVA

Ao concluir o curso técnico de desenvolvimento de sistemas o aluno adquire várias competências, o desenvolvimento de um projeto de conclusão auxilia a demonstrar estas capacidades técnicas na prática, serve como forma de avaliação somativa dentro das práticas da metodologia de ensino profissionalizante adotada pelo SENAI além de gerar a oportunidade de apresentar este projeto ao corpo docente da instituição e até a membros da gestão das indústrias da cidade de Jaguariúna e região.

1. TAP (TERMO DE ABERTURA DO PROJETO)

Título do Projeto:

Projeto Blitz "Site e Aplicativo Móvel capaz de gerenciar alertas de trânsito gerado pelos condutores no intuito de evitar acidentes e ganhar tempo"

Patrocinadores: SENAI Jaguariúna

Recursos Humanos	Allan Cesar Da Silva Ana Júlia Moreira Vieira Carlos Roberto De Oliveira Júnior Daniel Souza Da Silva Danilo Alves Falciorli Diogo Henrique Medeiros Elian De Oliveira Moraes Felipe Reis De Oliveira Gabriel Alex Sandro Soares Lançon Gabriel Vital Ferreira Gustavo Henrique Rocha De Oliveira Jaqueline Dias Oliveira Larissa Ferreira Pardim Lavínia Barbosa De Sá Leonardo Panigassi Imenes Matheus Belarmino Pignata Matheus Ricardo Belli Michael José Da Silva Michael Leonardo Chaves Medeiros Rafael Bartulic Tieppo Rafael Fernandes De Faria Roberto Donisete Pereira Dias Rodrigo Aparecido De Moraes Victória Rosa Dos Santos Vitor Leme Corrêa
Gerente do Projeto	Reenye Alexandre de Lima Wellington Fábio de Oliveira Martins
Patrocinador	SENAI Jaguariúna
Cliente	SENAI Jaguariúna
Prazo	31/03/2022

Local: _____ Data: ____/____/____	
Patrocinador	Ass: _____
Cliente	Ass: _____
Gerente do Projeto	Ass: _____

2. 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.

2.1. REQUISITOS NÃO FUNCIONAIS

[RN001] Linguagens de Programação:

[RN001.1] BackEnd: Javascript com o framework Node.js

[RN001.2] BackEnd: biblioteca ORM sequelise.

[RN001.3] FrontEnd: HTML, CSS e JavaScript sem framework

[RN001.4] Mobile: Javascript com o framework ReactNative

[RN002] FrontEnd e Mobile: API Google Maps

[RN003] Servidor e possível hospedagem

[RN003.1] HEROKU <https://www.heroku.com/> ou algum serviço de nuvem

[RN004] Recursos de Banco de Dados relacional e SQL

[RN004.1] MySQL 10.4.11-MariaDB

2.2. REQUISITOS FUNCIONAIS

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

A metodologia de desenvolvimento de sistemas aplicada neste projeto é conhecida como SCRUM que é uma subcategoria da metodologia Ágil, por este motivo cada requisito aqui apresentado faz parte do backlog do projeto que será dividido em duas Sprints conforme a sua criticidade.

A criticidade está dividida em três categorias: essencial, importante e desejável definindo o nível de prioridade para cada requisito acordado neste escopo.

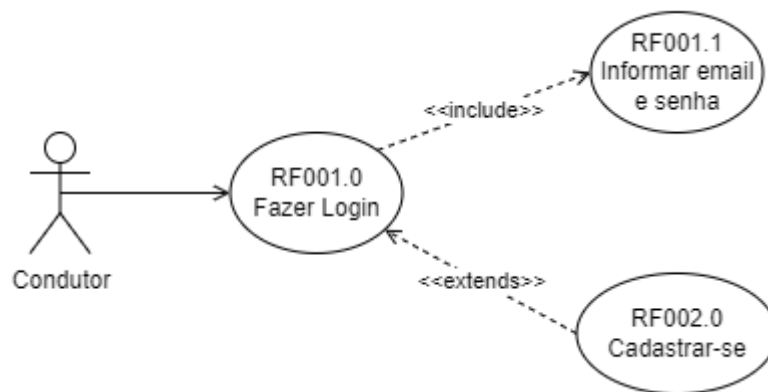


Figura 1 - DCU - [RF001]

[RF001]: Fazer login

Tela inicial do sistema para acessar ou cadastrar-se

Criticidade: (x) essencial () importante () desejável

[RF001.1]: Informar e-mail e senha

Preencher e-mail e senha

Criticidade: (x) essencial () importante () desejável

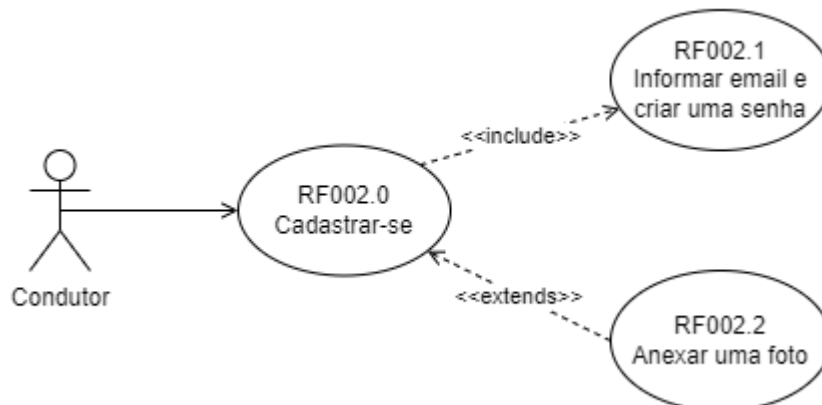


Figura 2 - CDU - [RF002]

[RF002.0]: Cadastrar-se

Preencher e-mail, senha anexar uma imagem/foto

Criticidade: (x) essencial () importante () desejável

[RF002.1]: Informar e-mail e criar uma senha

Informar e-mail e criar uma senha

Criticidade: (x) essencial () importante () desejável

[RF002.2]: Anexar uma foto

Anexar uma imagem/foto de perfil.

Criticidade: () essencial () importante (x) desejável

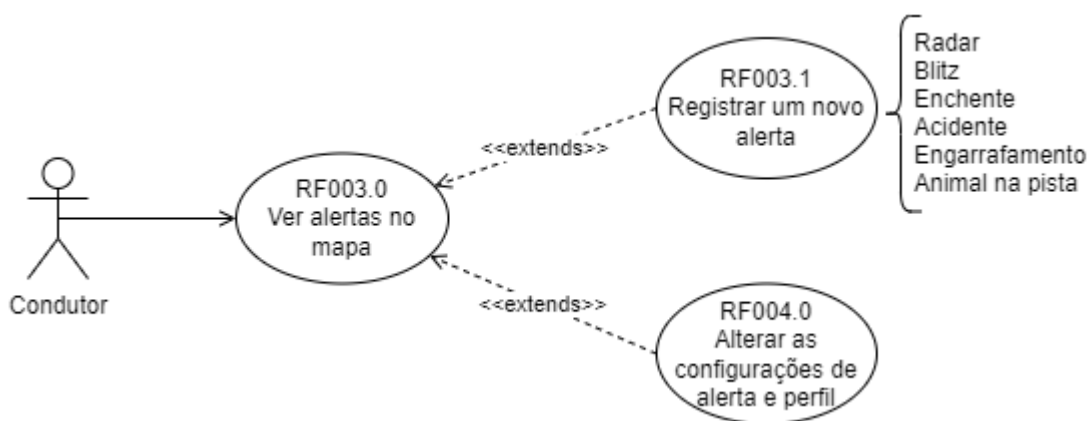


Figura 3 - DCU - [RF003]

[RF003.0]: Ver alertas no mapa

Tela principal do sistema, onde o usuário vê os alertas cadastrados e pode cadastrar um novo

Criticidade: (x) essencial () importante () desejável

[RF003.1]: Registrar um novo alerta

Ao clicar em uma posição no mapa o usuário pode registrar um novo alerta.
 {"Radar", "Blitz", "Enchente", "Acidente", "Engarrafamento", "Animal na pista"}

Criticidade: () essencial (x) importante () desejável

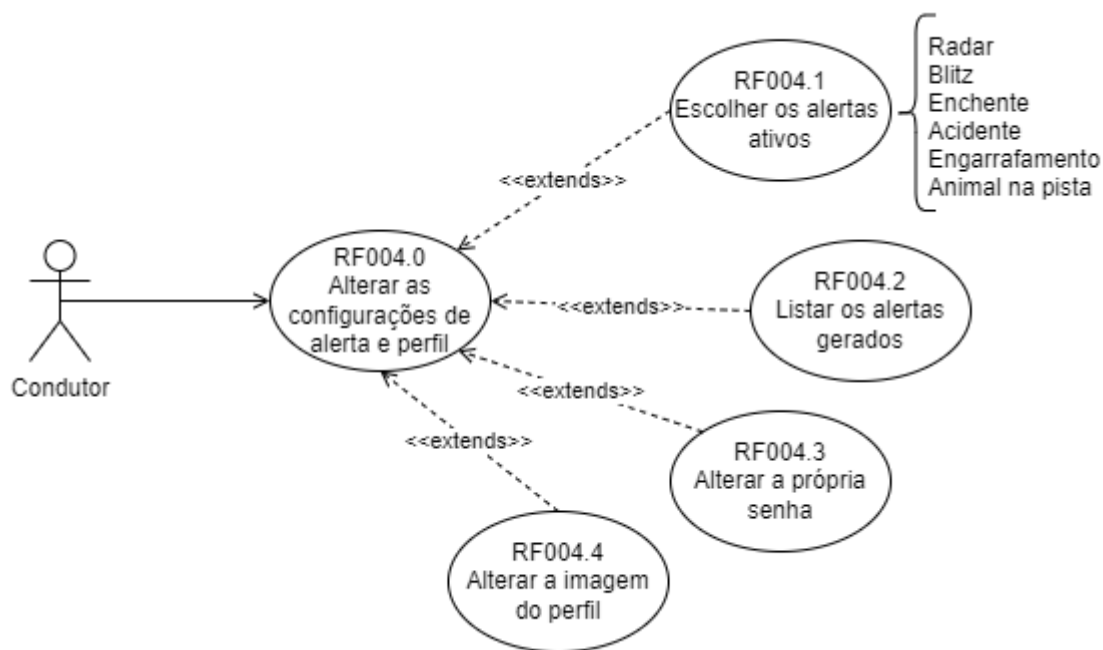


Figura 4 - DCU - [RF001]

[RF004.0]: Alterar as configurações de alerta e perfil,

Tela onde o usuário pode alterar as configurações de alerta e perfil, escolher quais alertas deseja ver, listar os alertas criados, alterar a própria senha, alterar a imagem do perfil

Criticidade: () essencial (x) importante () desejável

[RF004.1]: Escolher os alertas ativos

Escolher os alertas ativos {"Radar", "Blitz", "Enchente", "Acidente", "Engarrafamento", "Animal na pista"}

Criticidade: () essencial () importante (x) desejável

[RF004.2]: Escolher os alertas gerados

Criticidade: () essencial () importante (x) desejável

[RF004.3]: Alterar a própria senha

Criticidade: () essencial () importante (x) desejável

[RF004.4]: Alterar a imagem do perfil Alterar a imagem do perfil

Criticidade: () essencial () importante (x) desejável

3. 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.

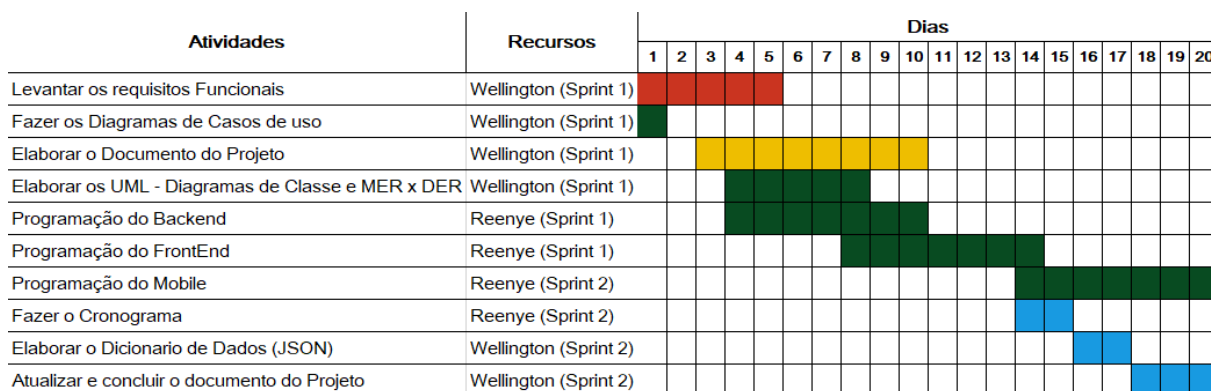


Figura 5 - Cronograma - Gantt

Legenda	
Prazo esgotado	
Concluído no prazo	
Concluído com atraso	
Planejado	

Figura 6 - Legenda – Gantt

3.1. 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.

Wellington 80 horas trabalhadas e Reenye 80 horas trabalhadas, totalizando 160 horas de projeto, caso cada recurso custasse R\$10,00 por hora o custo total deste projeto seria de: R\$ 1600,00

4. PROTÓTIPO

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

4.1. 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.

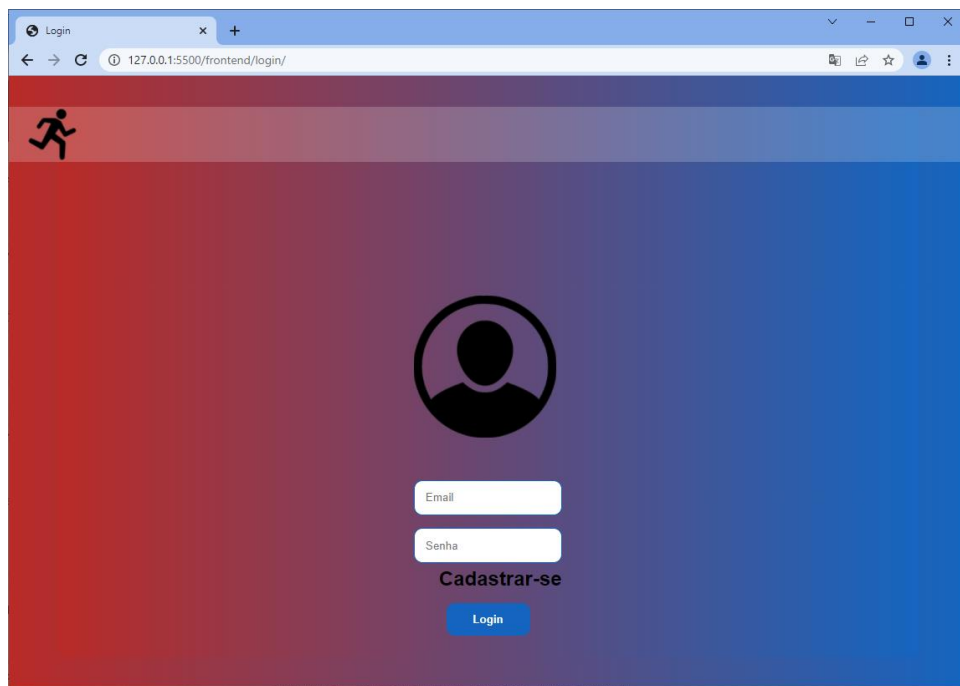


Figura 7 - Tela de Login

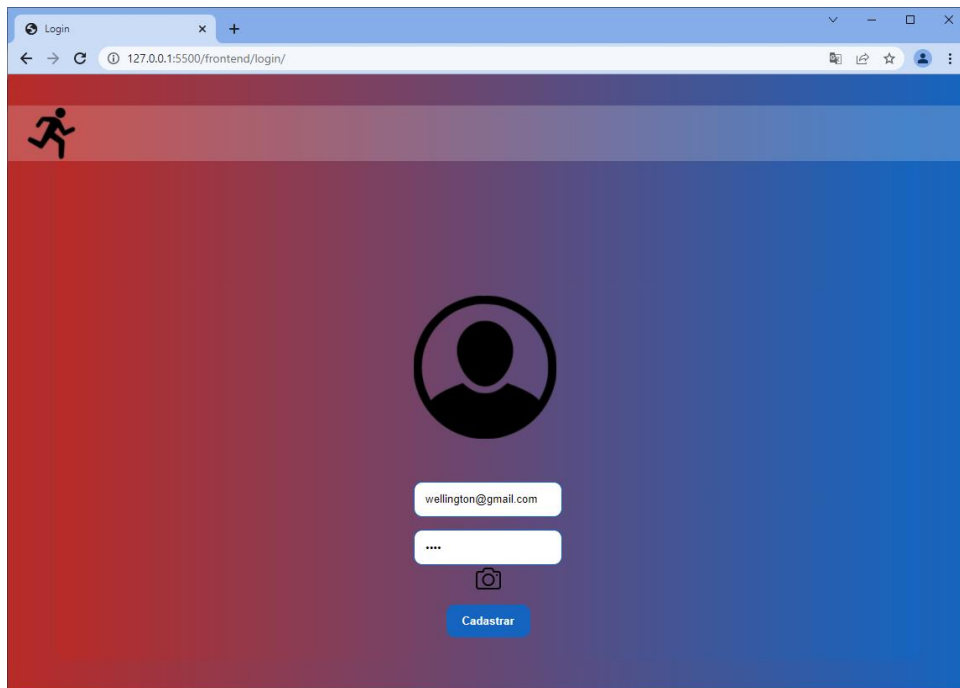


Figura 8 - Tela de Cadastro

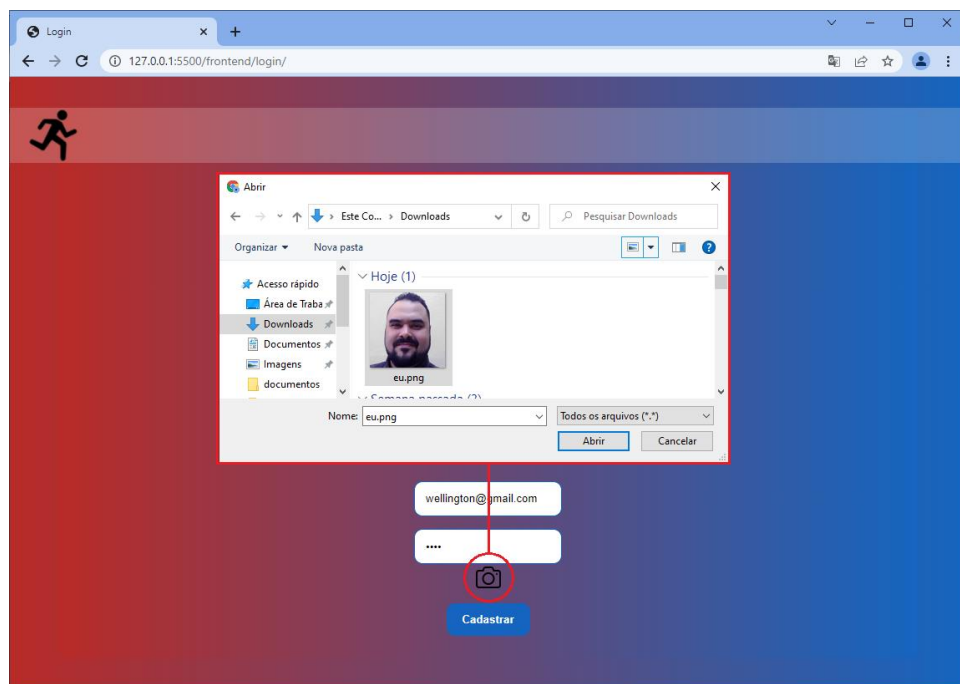


Figura 9 - Tela de Cadastro Inserindo Imagem de Perfil

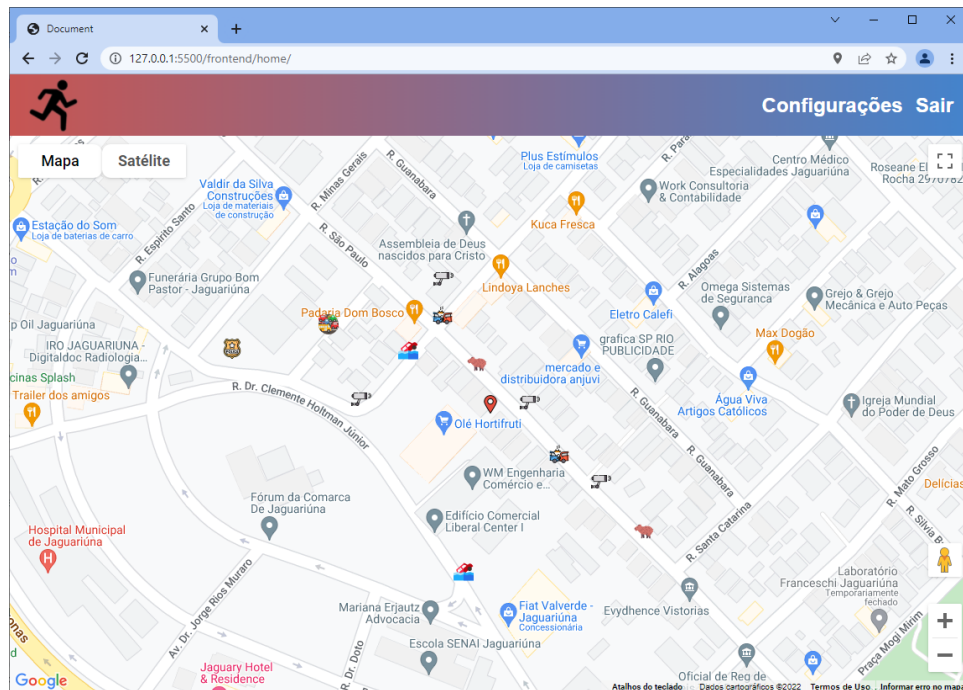


Figura 10 - Tela Principal (Home)

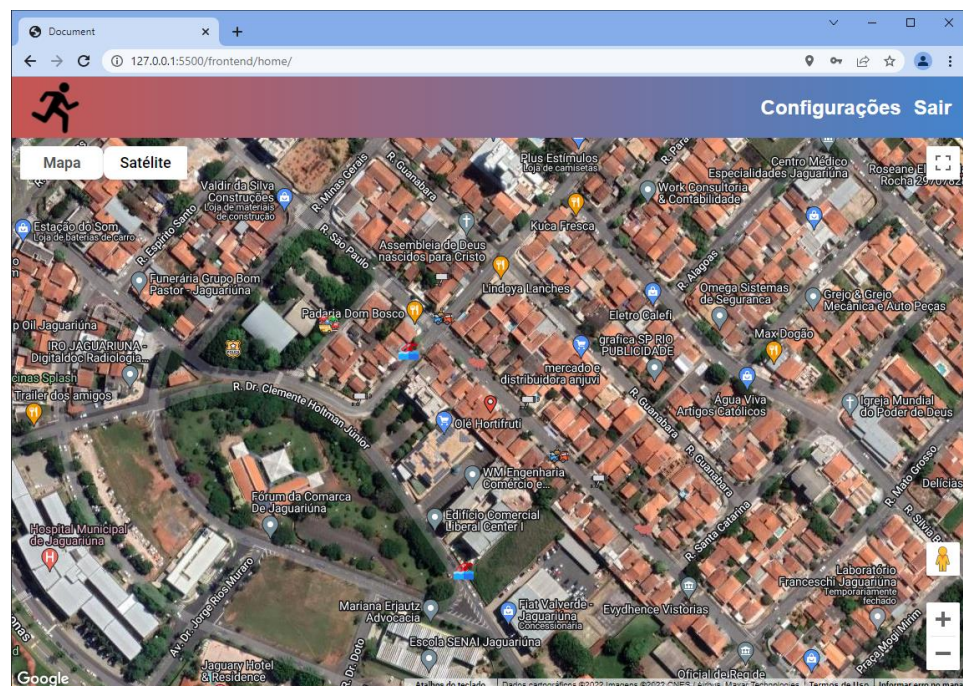


Figura 11 - Tela Principal (Home - Satellite)

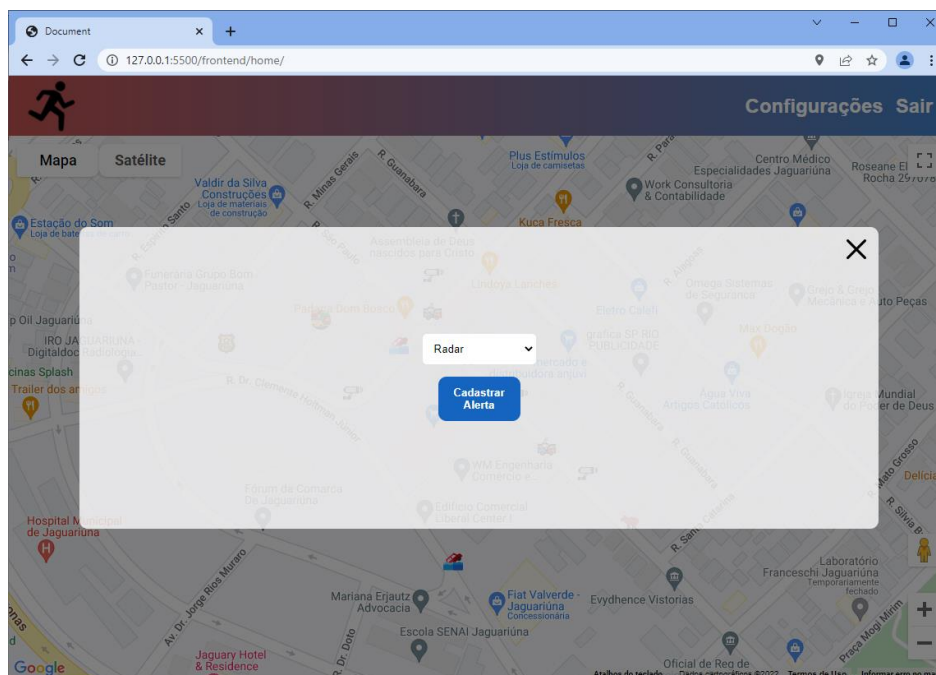


Figura 12 - Inserindo um novo alerta

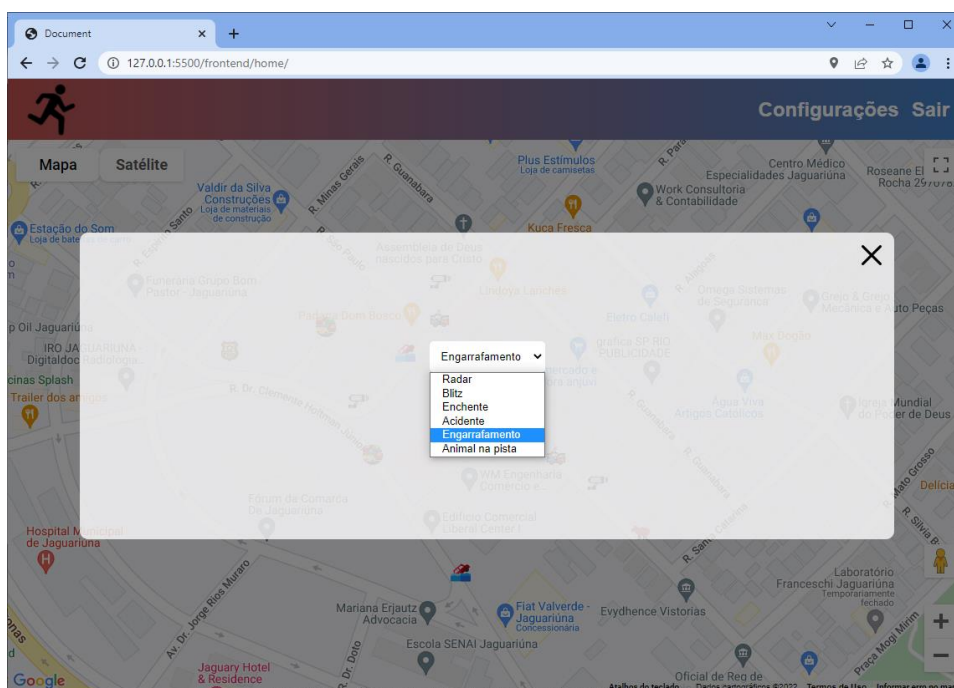


Figura 13 - Escolhendo o tipo de alerta

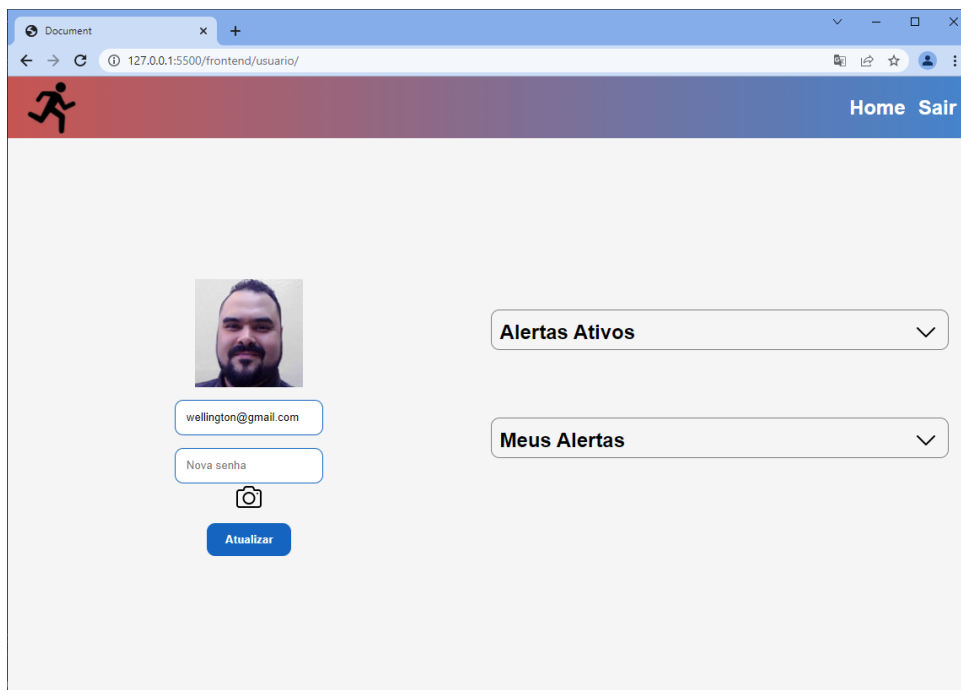


Figura 14 - Alterar o Perfil do Usuário

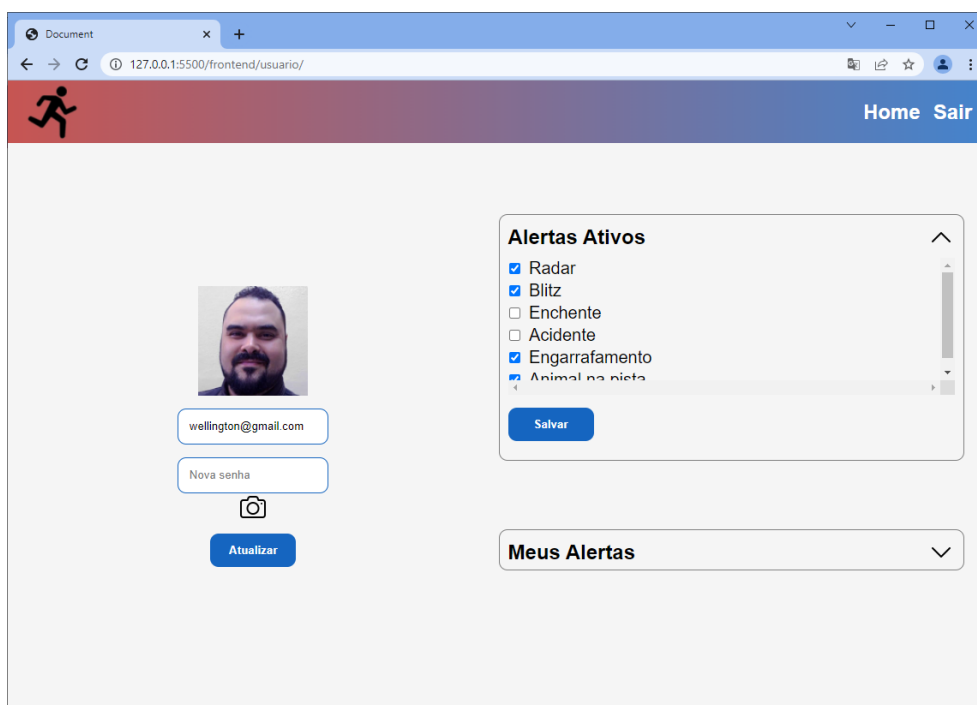


Figura 15 - Escolhendo os alertas ativos

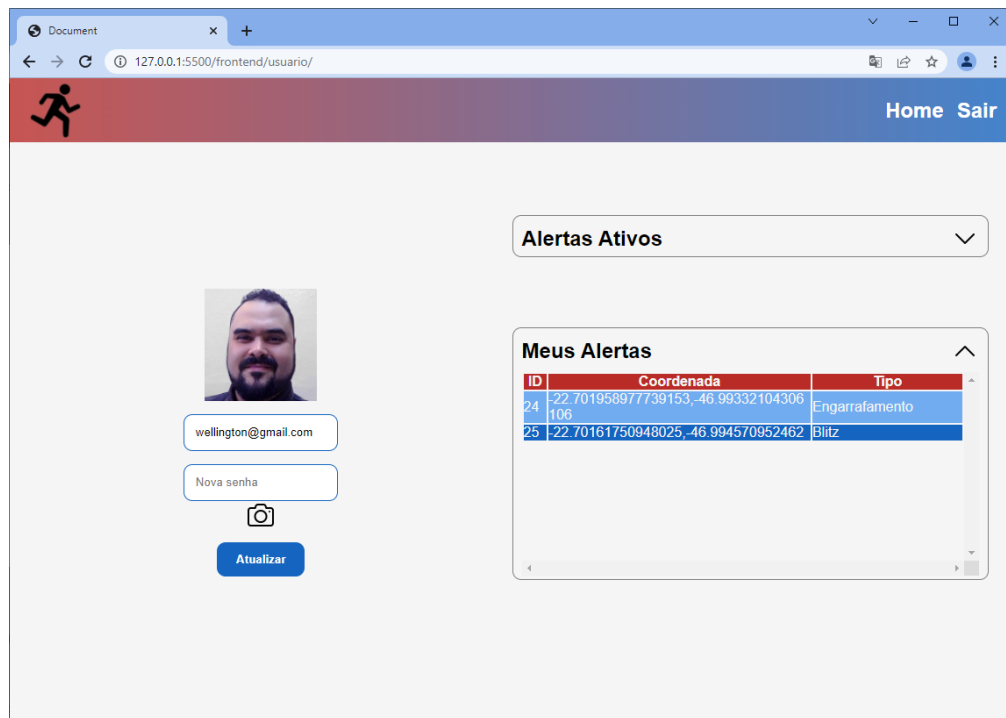


Figura 16 - Listando os alertas gerados

4.2. PROTÓTIPO DO APLICATIVO MÓVEL

As imagens mostram a sequência de telas que um condutor deve acessar para usufruir de todas as funcionalidades do aplicativo móvel.

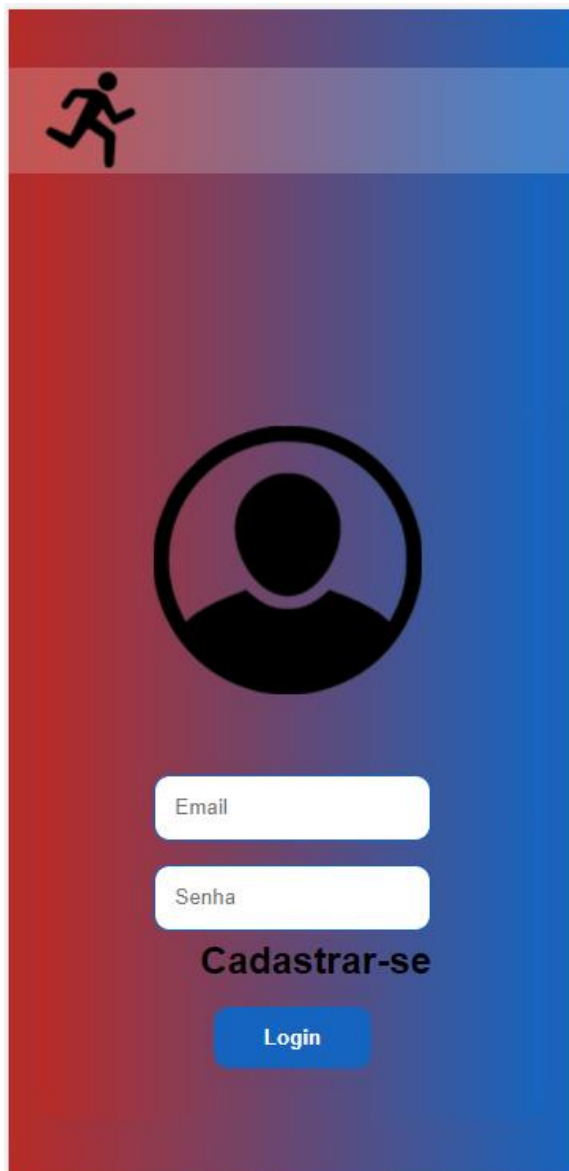


Figura 17 - Login App

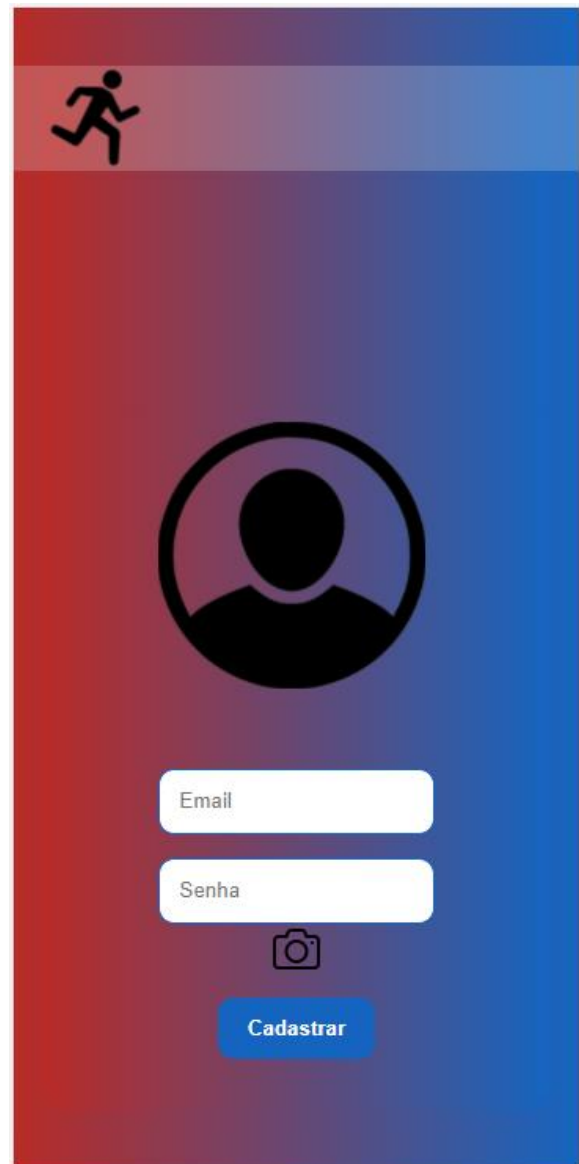


Figura 18 - Cadastro App

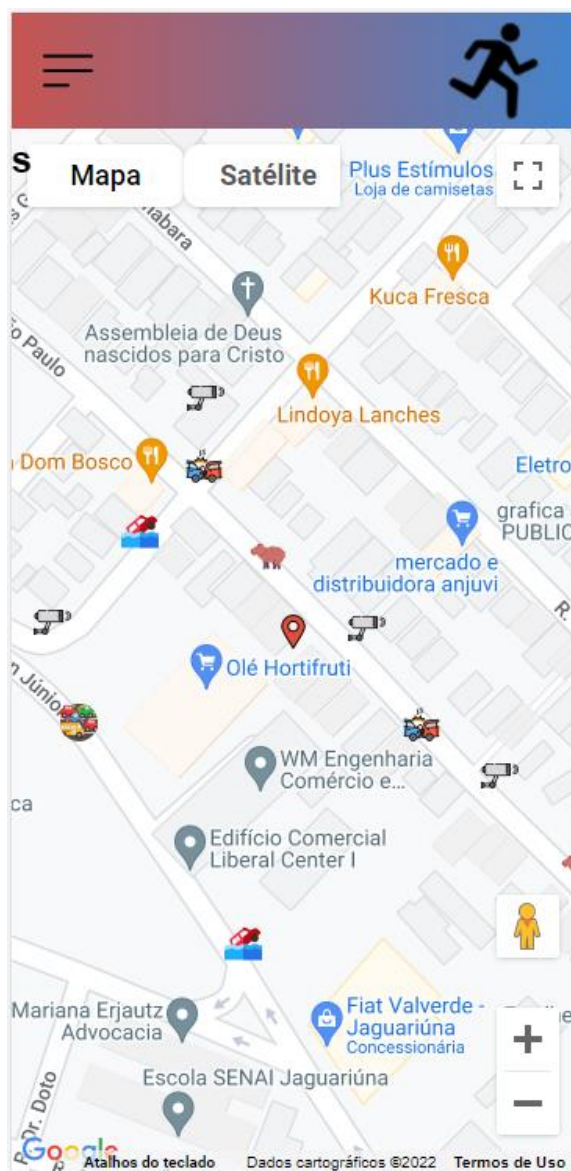


Figura 19 - Home App



Figura 20 - Home Satellite App

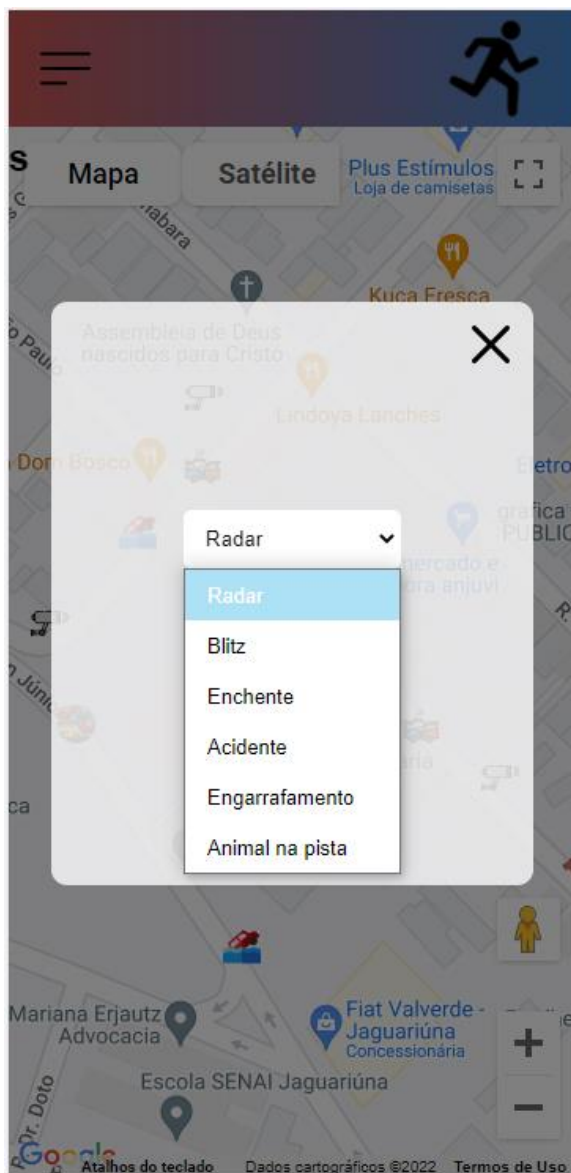


Figura 21 - Escolher alerta App

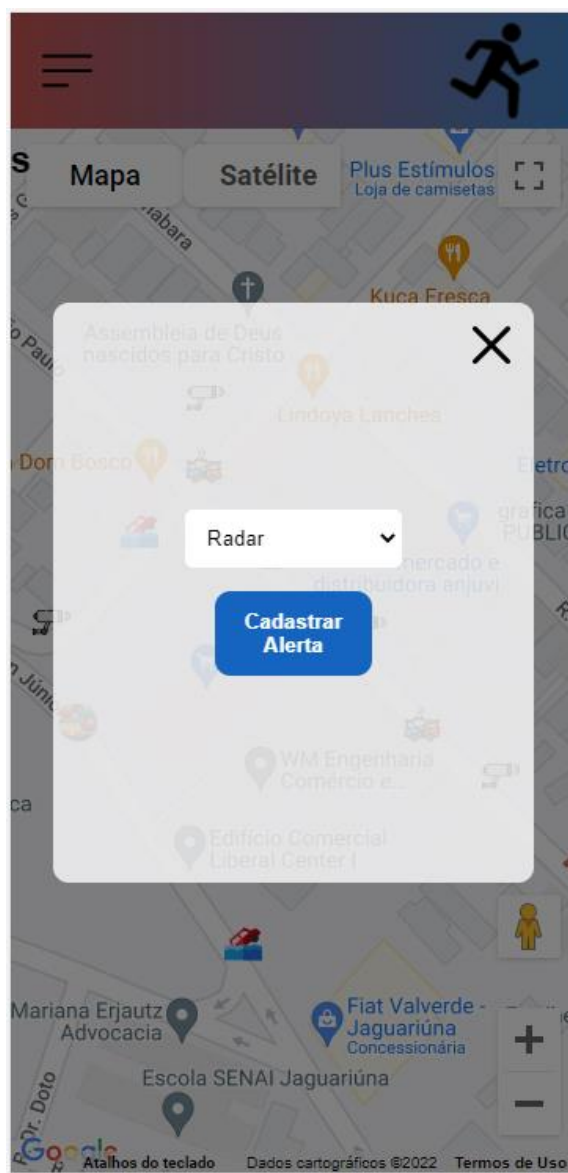


Figura 22 - Inserir alerta App

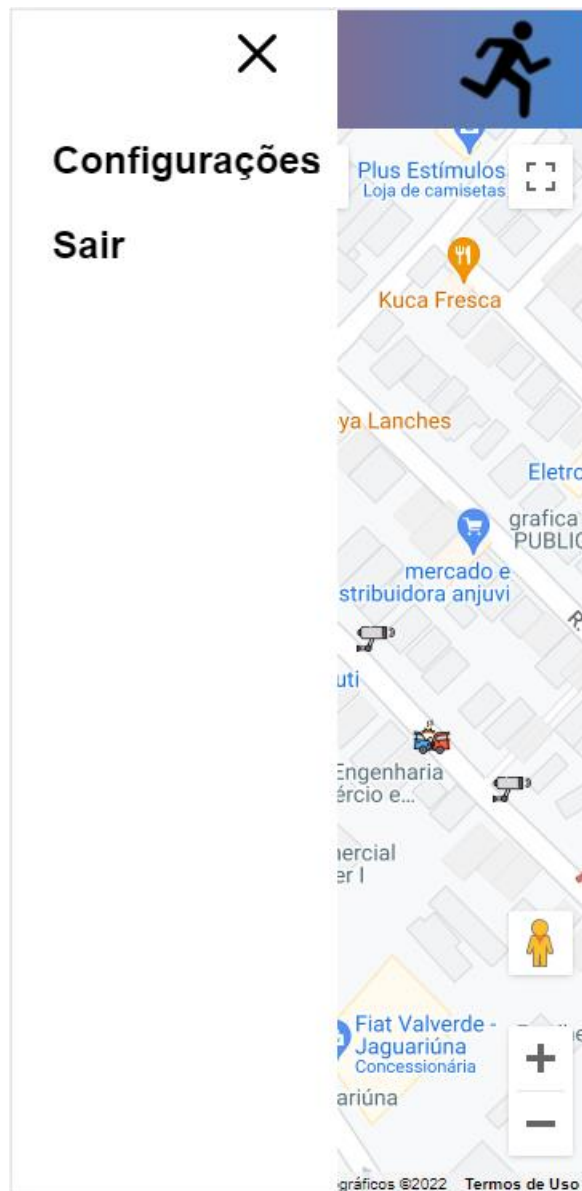


Figura 23 - Menu App

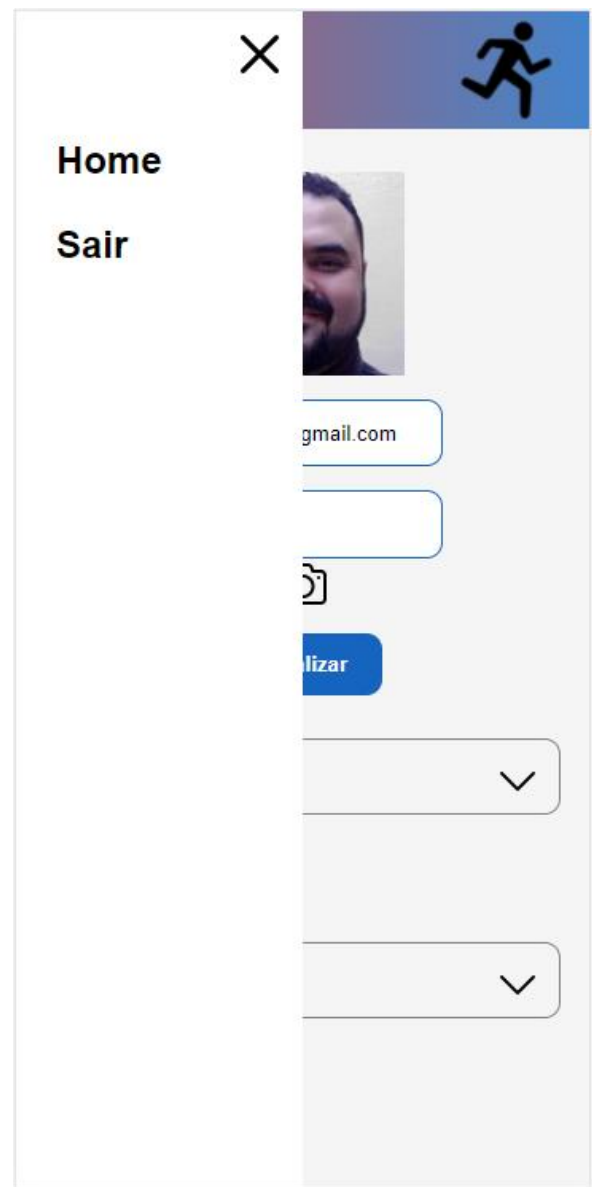


Figura 24 - Menu voltar App



Figura 25 - Perfil App

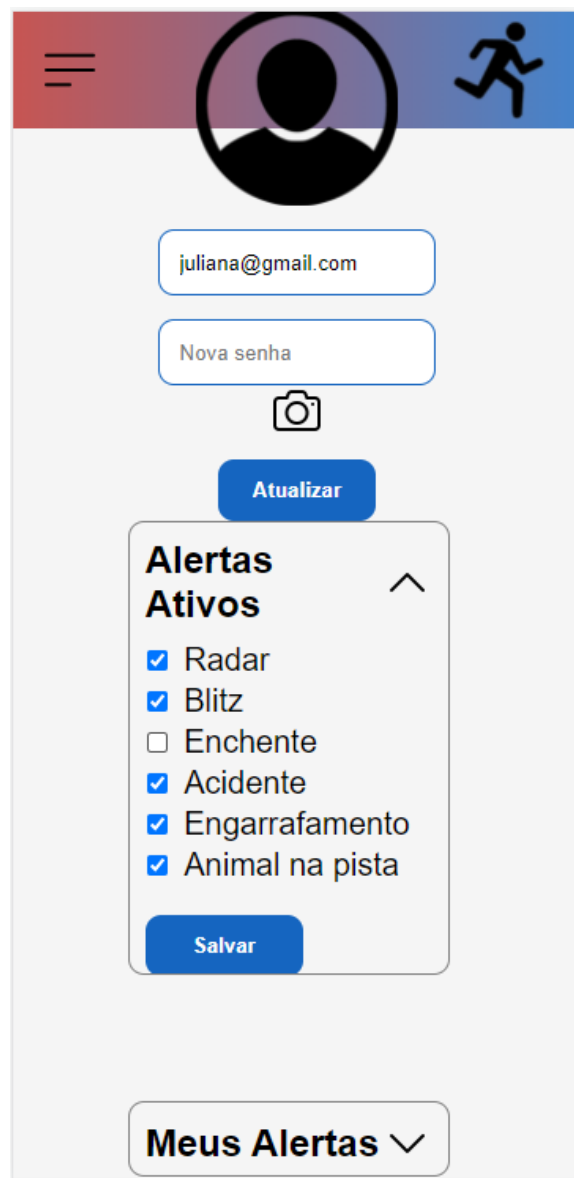


Figura 26 - Alertas ativos App

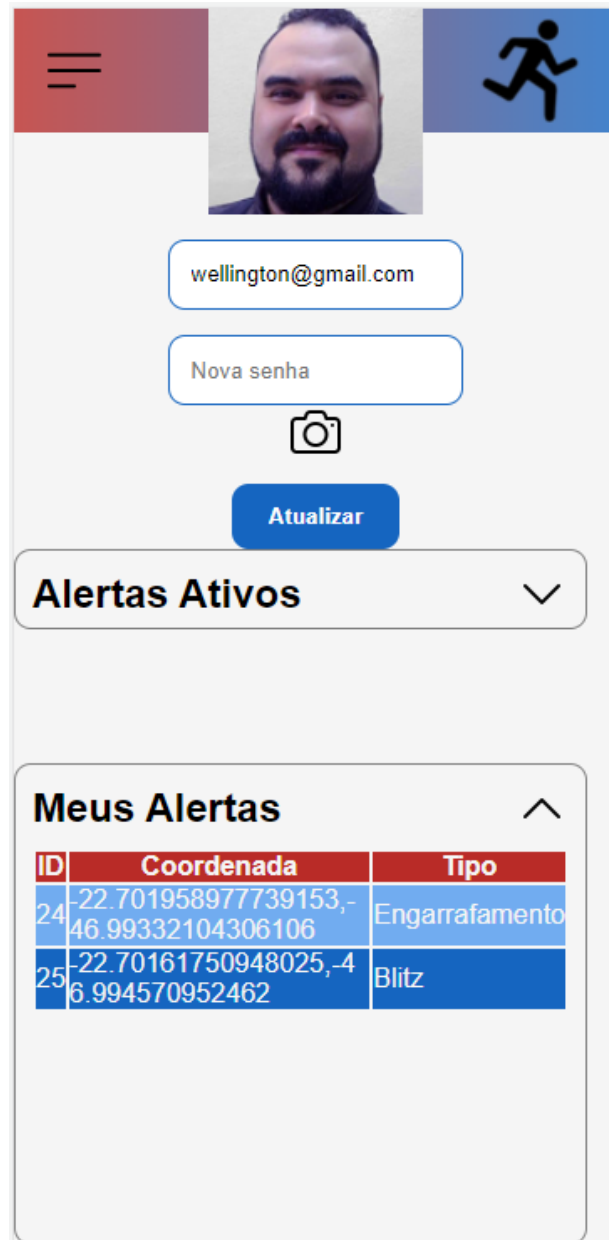


Figura 27 - Listar alertas App

As legendas das imagens descrevem as funcionalidades básicas do site e do aplicativo porém os requisitos estão melhor descritos no tópico dois sobre o escopo e requisitos funcionais do projeto.

5. 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.

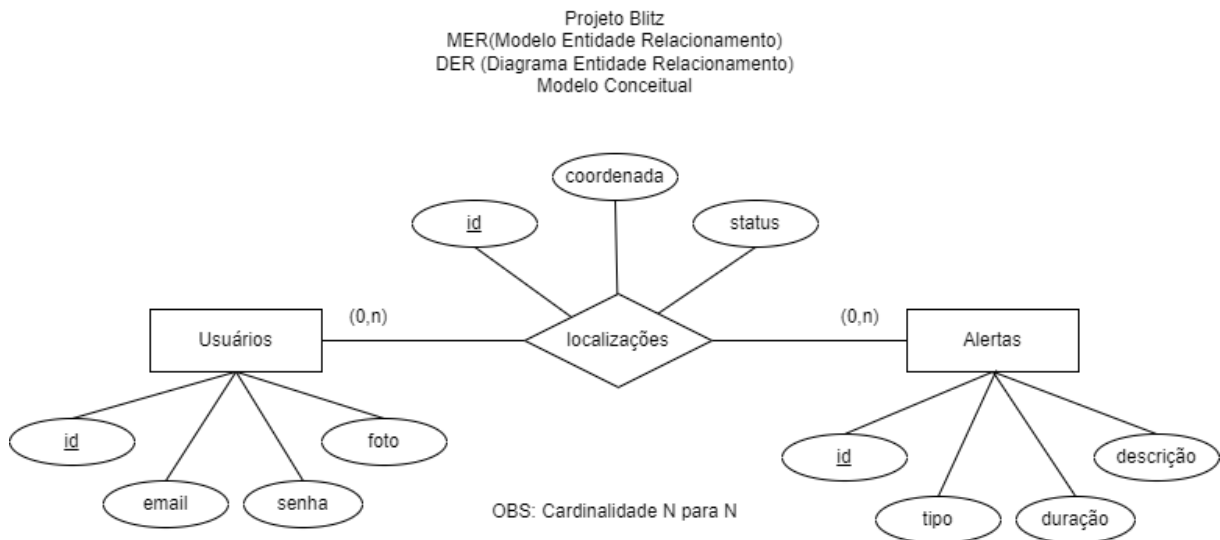


Figura 28 - Modelo Conceitual

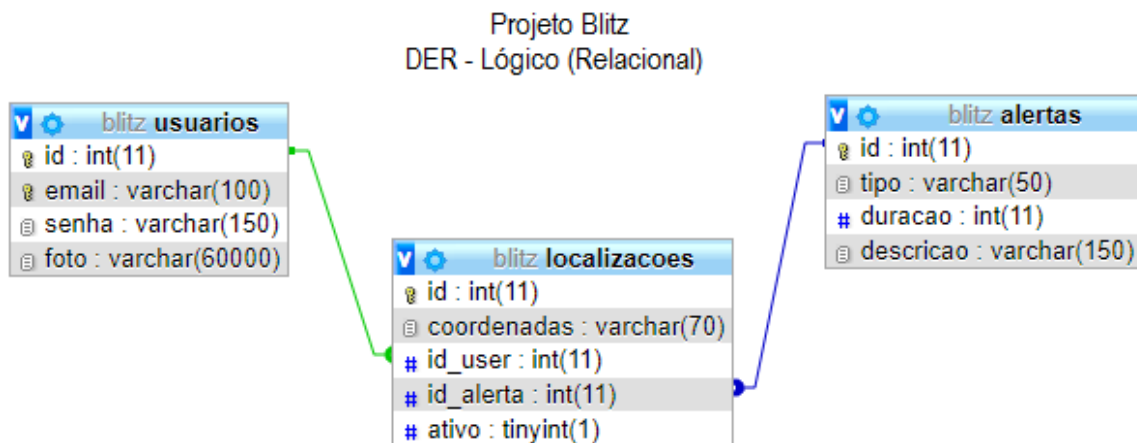


Figura 29 - Modelo Lógico

O diagrama de classes a seguir apresenta o projeto das classes da camada modelo do padrão de projetos MVC (Modelo Visão e Controle) do backend ou API.

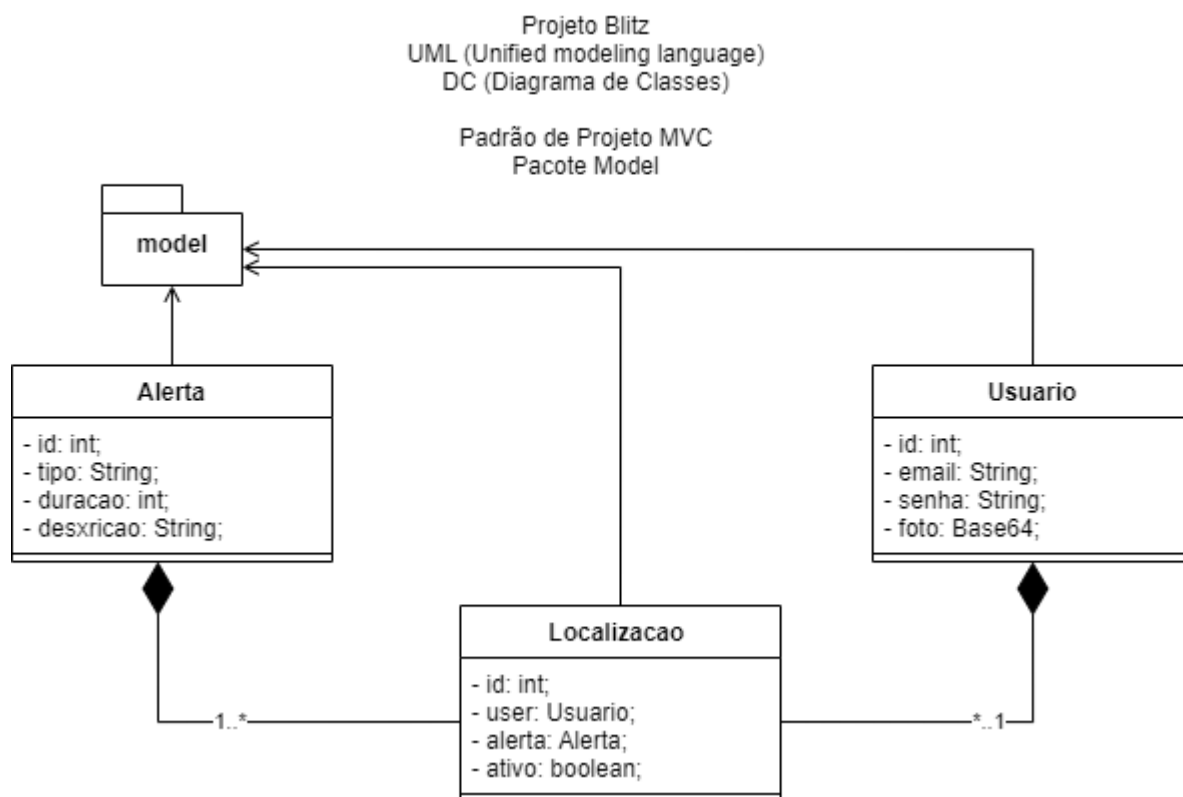


Figura 30 - Diagrama de Classes

6. DETALHES DA EXECUÇÃO E CONTROLE

O projeto foi desenvolvido em duas semanas e meia de aula, juntamente com demonstrações práticas com os alunos da turma do terceiro módulo, após a execução deste projeto modelo, os alunos foram separados em grupos com três integrantes e foi proposto um pré-projeto de conclusão do curso com temas diversificados.

Foi proposto para o pré-projeto, como forma de integração maior da turma, que um grupo seria responsável por desenvolver a camada de backend e banco de dados do outro grupo, para isso o grupo principal do projeto deve fazer a documentação completa do projeto e utilizar 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 Alerta:

```
{
  "01-verbo": "READ",
  "01-rota": "/alerta",
  "01-status": 200,
  "01-response": [
    {
      "id": 1,
      "tipo": "Radar",
      "duracao": 180,
      "descricao": "Enchente"
    },
    {
      "id": 2,
      "tipo": "Blitz",
      "duracao": 60,
      "descricao": "Blitz"
    },
    {
      "id": 3,
      "tipo": "Enchente",
      "duracao": 60,
      "descricao": "Enchente"
    },
    {
      "id": 4,
      "tipo": "Engarrafamento",
      "duracao": 60,
      "descricao": "Congestionamento/Engarrafamento"
    }
  ]
}
```

```
        "id": 5,
        "tipo": "Acidente",
        "duracao": 60,
        "descricao": ""
    },
    {
        "id": 6,
        "tipo": "Animal na pista",
        "duracao": 60,
        "descricao": ""
    }
],
"02-verbo": "READ",
"02-rota": "/alerta/1",
"02-status": 200,
"02-response": [
    {
        "id": 1,
        "tipo": "Radar",
        "duracao": 180,
        "descricao": "Enchente"
    }
],
"03-verbo": "POST",
"03-rota": "/alerta",
"03-request": {
    "tipo": "Depressao na pista",
    "duracao": 60,
    "descricao": ""
},
"03-true-status": 201,
"03-response": {
    "id": 7,
    "tipo": "Depressao na pista",
    "duracao": 60,
    "descricao": ""
},
"03-false-status": 400,
"04-verbo": "PUT",
"04-rota": "/alerta/7",
"04-request": {
    "tipo": "Depressao na Pista",
    "duracao": 180,
    "descricao": "Depressao"
},
"04-true-status": 200,
"04-response": [
    {
        "id": 1,
        "tipo": "Enchente",
        "duracao": 180,
        "descricao": "Enchente"
    }
],
"04-false-status": 404,
"05-verbo": "DELETE",
"05-rota": "/alerta/7",
"05-true-status": 200,
"05-response": {
```

```
    "id": "7"
  },
  "05-false-status": 404
}
```

Dicionário de dados da classe Usuário:

```
{
  "01-verbo": "READ",
  "01-rota": "/usuario",
  "01-status": 200,
  "01-response": [
    {
      "id": 1,
      "email": "beltrano@gmail.com",
      "senha": "81dc9bdb52d04dc20036dbd8313ed055",
      "foto": ""
    },
    {
      "id": 2,
      "email": "wellington@gmail.com",
      "senha": "81dc9bdb52d04dc20036dbd8313ed055",
      "foto": "data:image/png;base64,/9j/2wCEAAgGB..."
    },
    {
      "id": 3,
      "email": "juliana@gmail.com",
      "senha": "81dc9bdb52d04dc20036dbd8313ed055",
      "foto": ""
    }
  ],
  "02-verbo": "READ",
  "02-rota": "/usuario/1",
  "02-status": 200,
  "02-response": {
    "id": 1,
    "email": "beltrano@gmail.com",
    "senha": "81dc9bdb52d04dc20036dbd8313ed055",
    "foto": ""
  },
  "03-verbo": "POST",
  "03-rota": "/usuario",
  "03-request": {
    "email": "juliana@gmail.com",
    "senha": "81dc9bdb52d04dc20036dbd8313ed055",
    "foto": ""
  },
  "04-status": 201,
  "04-verbo": "POST",
  "04-rota": "/login",
  "04-request": {
    "email": "juliana@gmail.com",
    "senha": "81dc9bdb52d04dc20036dbd8313ed055"
  },
  "04-true-status": 200,
  "04-response": {
    "id": 3,

```

```

    "email": "juliana@gmail.com",
    "foto": ""
  },
  "04-false-status": 404,
  "05-verbo": "PUT",
  "05-rota": "/usuario/1",
  "05-true-status": 200,
  "05-request": {
    "senha": "81dc9bdb52d04dc20036dbd8313ed055"
  },
  "05-false-status": 404,
  "06-verbo": "DELETE",
  "06-rota": "/usuario/1",
  "06-true-status": 200,
  "06-false-status": 404
}

```

Dicionário de dados da classe localização:

```

{
  "01-verbo": "GET",
  "01-rota": "/localizacao",
  "01-status": 200,
  "01-response": [
    {
      "id": 1,
      "coordenadas": "-22.701347063849923,-46.992755096958916",
      "ativo": true,
      "usuario": {
        "id": 1,
        "email": "oi@gmail.com"
      },
      "alertum": {
        "id": 1,
        "tipo": "Radar"
      }
    },
    {
      "id": 2,
      "coordenadas": "-22.701436142784136,-46.99265853743438",
      "ativo": true,
      "usuario": {
        "id": 1,
        "email": "oi@gmail.com"
      },
      "alertum": {
        "id": 2,
        "tipo": "Blitz"
      }
    },
    {
      "id": 3,
      "coordenadas": "-22.700377089482824,-46.995651882695",
      "ativo": true,
      "usuario": {
        "id": 1,
        "email": "oi@gmail.com"
      }
    }
  ]
}

```

```
    },
    "alertum": {
      "id": 6,
      "tipo": "Animal na pista"
    }
  },
  {
    "id": 4,
    "coordenadas": "-22.7019953592143,-46.99209527354125",
    "ativo": true,
    "usuario": {
      "id": 1,
      "email": "oi@gmail.com"
    },
    "alertum": {
      "id": 6,
      "tipo": "Animal na pista"
    }
  },
  {
    "id": 4,
    "coordenadas": "-22.7019953592143,-46.99209527354125",
    "ativo": true,
    "usuario": {
      "id": 1,
      "email": "oi@gmail.com"
    },
    "alertum": {
      "id": 5,
      "tipo": "Acidente"
    }
  },
  {
    "id": 5,
    "coordenadas": "-22.702420956029204,-46.99159638266448",
    "ativo": true,
    "usuario": {
      "id": 1,
      "email": "oi@gmail.com"
    },
    "alertum": {
      "id": 1,
      "tipo": "Radar"
    }
  }
],
"02-verbo": "GET",
"02-rota": "/localizacao/1",
"02-status": 200,
"02-response": [
  {
    "id": 1,
    "coordenadas": "-22.701347063849923,-46.992755096958916",
    "ativo": true,
    "usuario": {
      "id": 1,
      "email": "oi@gmail.com",
      "foto": ""
    }
  },

```

```
        "alertum": {
            "id": 1,
            "tipo": "Radar",
            "duracao": 180,
            "descricao": "Enchente"
        }
    },
    ],
    "03-verbo": "POST",
    "03-rota": "/localizacao",
    "03-request": {
        "coordenadas": "-12.123,-12.123",
        "id_user": 2,
        "id_alerta": 1,
        "ativo": true
    },
    "03-true-status": 201,
    "03-response": {
        "id": 11,
        "coordenadas": "-12.123,-12.123",
        "id_user": 2,
        "id_alerta": 1,
        "ativo": true
    },
    "03-false-status": 400,
    "04-verbo": "PUT",
    "04-rota": "/localizacao/1",
    "04-request": {
        "coordenadas": "-12.123,-12.123",
        "id_user": 2,
        "id_alerta": 1,
        "ativo": false
    },
    "04-true-status": 200,
    "04-response": [
        {
            "id": 1,
            "coordenadas": "-22.701347063849923,-46.992755096958916",
            "id_user": 1,
            "id_alerta": 1,
            "ativo": false
        }
    ],
    "04-false-status": 404,
    "05-verbo": "DELETE",
    "05-rota": "/localizacao/2",
    "05-true-status": 200,
    "05-response": {
        "id": "2"
    },
    "05-false-status": 404
}
```


7. RESULTADOS

O projeto foi apresentado em aula para os alunos do terceiro módulo no primeiro semestre do ano de 2022 e servirá de modelo para próximas turmas, a execução do sistema cumpriu 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/wellifabio/senai2022/tree/master/3des/projetos/proj_blitz este é um repositório público para consulta dos alunos e professores desta instituição.

Este projeto não foi hospedado em nenhum serviço privado, público ou de nuvem, apenas na rede local da sala de aula desta instituição de ensino.

BIBLIOGRAFIA

Brasil: HostGator, Disponível em: <https://www.hostgator.com.br/blog/api-entenda-a-interface-de-programacao-de-aplicacao/?gclid=Cj0KCQiAmpyRBhC-ARIsABs2EApZW0U35EoTPMvR6h76PIMBPAqxOVBVnyNln68Wo1_iOWwaDB-gUuKgaAv-XEALw_wcB>, acessado em 08/03/2022.

Brasil: HostGator, Disponível em: <https://www.hostgator.com.br/blog/frontend-back-end-e-full-stack/?gclid=Cj0KCQiAmpyRBhC-ARIsABs2EA-oAai258HK5IS4M8nJy5tWkv0AVjVv2Ya3n9r-8jDyASrncN1vbiJU-aAqWaEALw_wcB>, acessado em 08/03/2022.

Brasil: Le Wagon, Disponível em: <<https://www.lewagon.com/pt-BR/blog/o-que-e-framework>>, acessado em 08/03/2022.

PMBOK Guide, 6a. Edição, 2017. 762 p.

Brasil: DEVMEDIA, Disponível em: <<https://www.devmedia.com.br/orm-object-relational-mapper/19056>>, acessado em 08/06/2022.