**Universidade do Oeste de Santa Catarina**

**Murilo Ferrari Angeli**

**Natãn Gabriel Trennepohl Zanella da Costa**

**Rafael Henrique Lisot**

**SarsCov News**

**Volume 1**

**São Miguel do Oeste**

**2021**

**Murilo Ferrari Angeli**

**Natãn Gabriel Trennepohl Zanella da Costa**

**Rafael Henrique Lisot**

**SarsCov News**

Trabalho interdisciplinar para obtenção de nota parcial

nas matérias de Programação IV, Banco de Dados II e

Engenharia de Software II,

apresentado a Universidade do Oeste de Santa Catarina - UNOESC

**São Miguel do Oeste**

**2021**

**Lista de Ilustrações**

**Figuras**

**Figura 1 - Tela Inicial 5**

**Figura 2 - Tela de Login 7**

**Figura 3 - Tela de Cadastro 8**

**Figura 4 - Tela Sobre 8**

**Figura 5 - Tela de Exibição de Dados 9**

**Figura 6 - Tela de Tira-Dúvidas 9**

**Figura 7 - Modelo Relacional 10**

**Figura 8 - Tarefas Backend 11**

**Figura 9 - Tarefas Frontend 11**

**Figura 10 - Tarefas Banco de Dados 12**

**Figura 11 - Tarefas Engenharia de Software 12**

**Figura 12 - Caso de Uso 1 13**

**Figura 13 - Caso de Uso 2 13**

**Figura 14 - Diagrama de Classes 14**

**Figura 15 - Diagrama de Atividade 14**

**Figura 16 - Diagrama de Estado 15**

**Figura 17 - Diagrama de Sequência 15**

Sumário

**INTRODUÇÃO5**

**DESENVOLVIMENTO7**

PROGRAMAÇÃO7

BANCO DE DADOS10

ENGENHARIA DE SOFTWARE11

**CONCLUSÃO16**

**APÊNDICES17**

1 INTRODUÇÃO

SarsCov News é um site de monitoramento de casos de covid-19, desenvolvido por alunos do curso de ciência da computação da Universidade do Oeste de Santa Catarina. O projeto teve sua primeira versão lançada no final do ano de 2020, porém com um aprofundamento dos conhecimentos de programação a segunda versão começou a ser desenvolvida, mas diferente da primeira que era um sistema para desktop, a segunda versão foi desenvolvida para a web. Esta nova versão ainda está em desenvolvimento, mas muitas funções já estão em funcionamento podendo ser acessadas por todos os usuários.

**Figura 1 –** Tela Inicial



**Fonte:** Compilação do Autor**.**

**2 DESENVOLVIMENTO**

**2.1 Programação**

O sistema foi desenvolvido nas seguintes linguagens: PHP com o framework Laravel para o backend, a biblioteca ReactJs para o frontend e Postgres para o banco de dados.

Começamos a desenvolver o sistema dividindo as tarefas de cada integrante com base no método de desenvolvimento ágil XP (Extreme programming). Um desenvolve aquilo que cabe ao frontend, ou seja, desenvolvimento do layout das telas, inicial, cadastro, login, sobre, tira-dúvidas, e tela onde são mostrados os dados e API de casos de covid-19. Outro desenvolveu a parte de backend, o lado que gerencia todo o sistema, como os sistemas de login e cadastro, envio e resposta de dúvidas e conexão com o banco de dados.

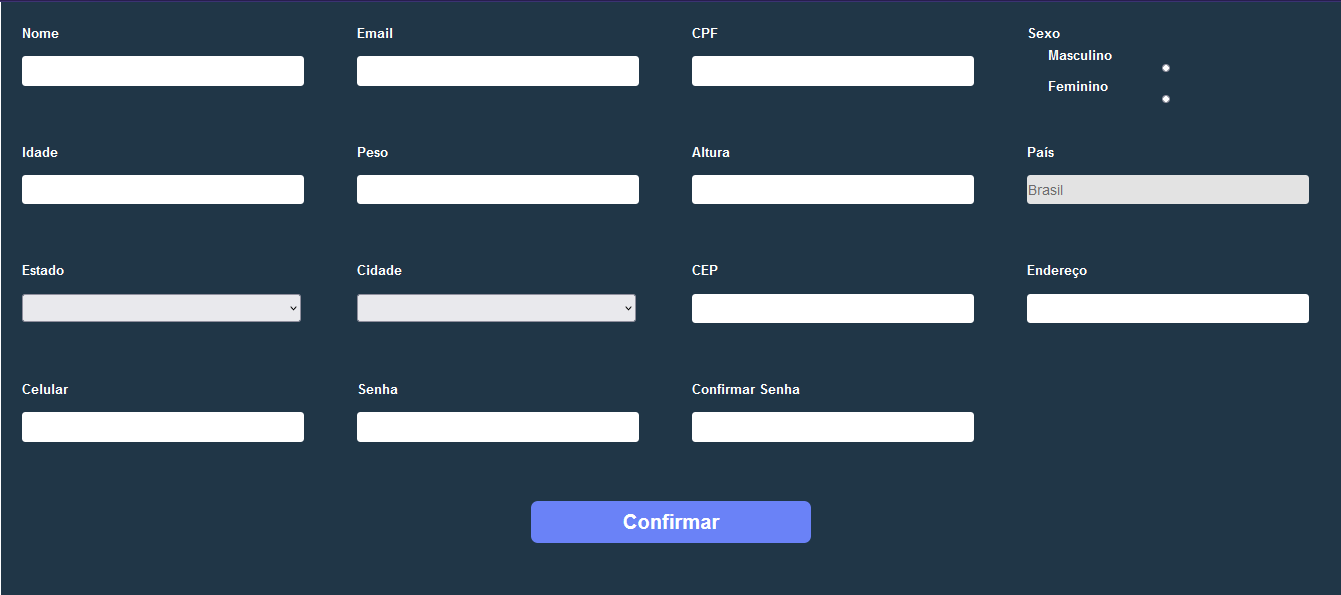
Quando cada uma das partes precisava de ajuda ou possuía dúvidas, desenvolvíamos em par, os dois procurando a solução enquanto um escrevia o código e o outro revisava.

**Figura 2** – Tela de Login



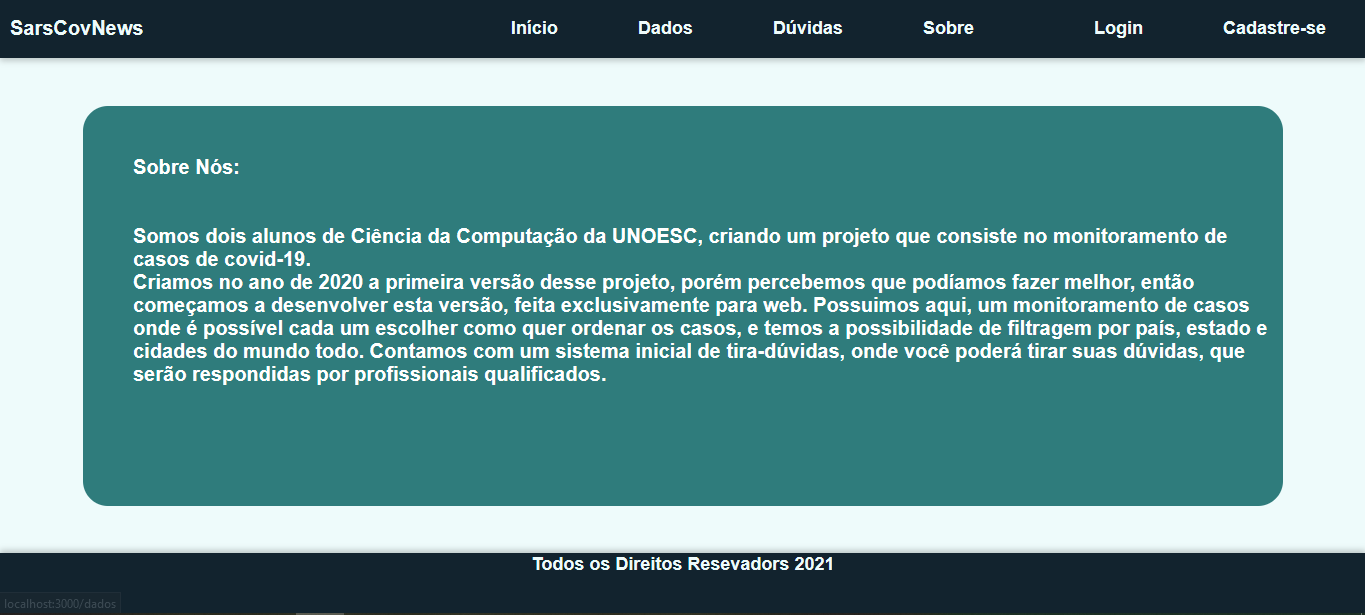
**Fonte:** Compilação do Autor**.**

**Figura 3** – Tela de Cadastro



**Fonte:** Compilação do Autor**.**

**Figura 4 –** Tela Sobre

****

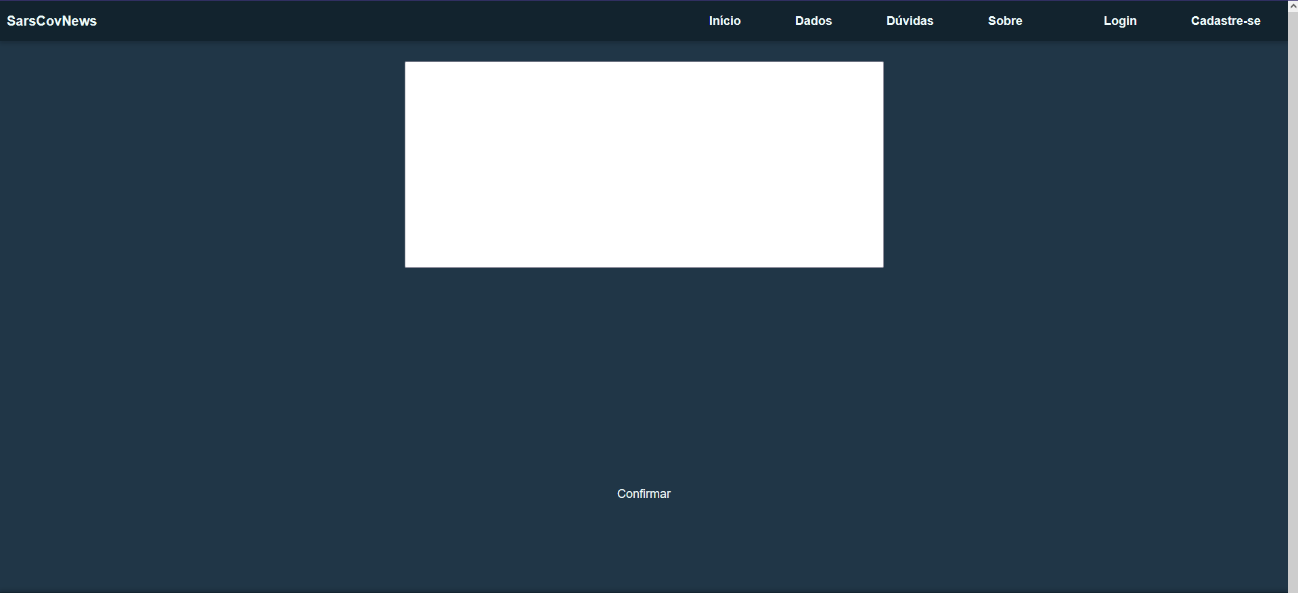
**Fonte:** Compilação do Autor**.**

**Figura 5 – Tela de exebição de Dados**

****

**Fonte:** Compilação do Autor**.**

**Figura 6 – Tela de Tira-Dúvidas**

****

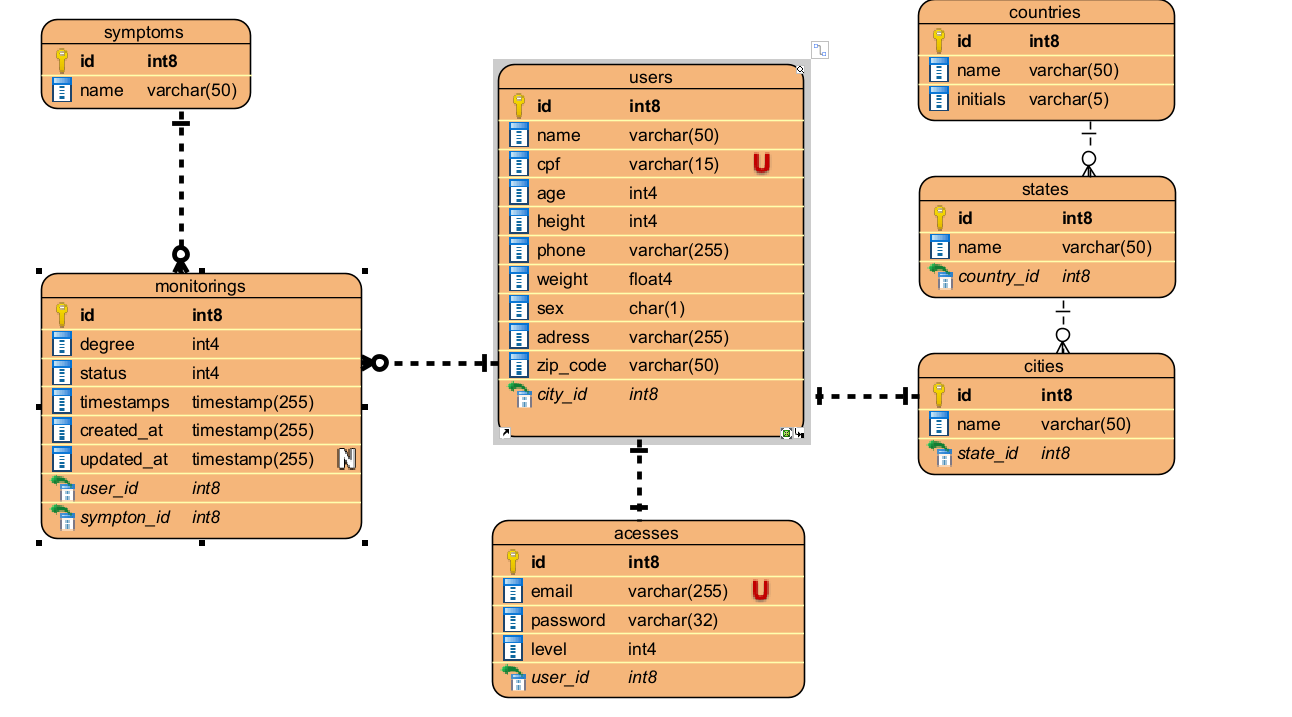
**Fonte:** Compilação do Autor**.**

**2.2 Banco de Dados**

Para a criação do banco de dados primeiramente desenhamos ele em quadro para maior facilidade em aplicar modificações, depois passamos para o modelo lógico relacional, feito no programa Visual Paradigm, por fim geramos a base de dados no postgres, e criamos as migrations para a conexão do banco e os seeders para popular as tabelas de país, estado e cidade, que serão usadas no cadastro de novos usuários.

Scripts de criação das tabelas da base de dados encontram-se no Apêndice A deste artigo.

**Figura 7 – Modelo Relacional**

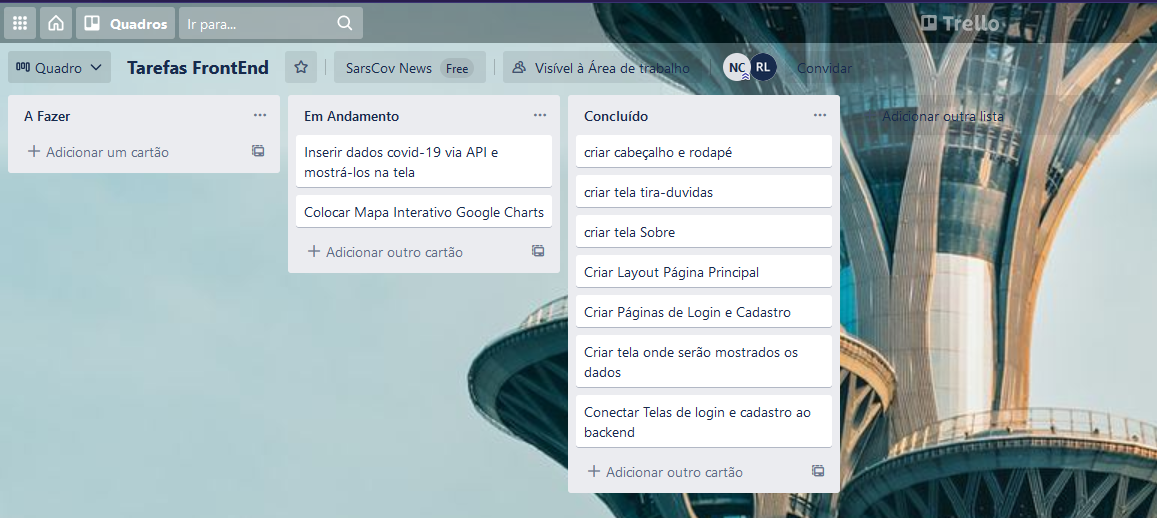


**Fonte:** Compilação do Autor**.**

**2.3 Engenharia de Software**

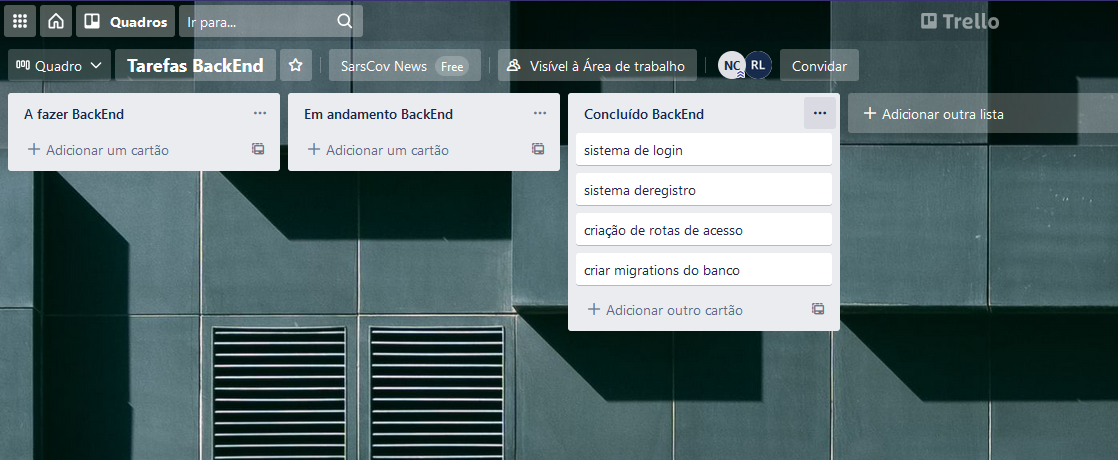
Primeiramente nos organizamos e definimos todas as tarefas que deveríamos fazer neste projeto, dividimos as tarefas em backend, frontend, banco de dados e engenharia. Tudo que deveria ser implementado ou criado estava designado em cada matéria.

Criamos todos os diagramas necessários, modelo de casos de uso, diagrama de classes, diagrama de sequência de estado e atividades. E por fim definimos métricas de qualidade de softwares e criamos um plano de testes para garantir a funcionalidade total do sistema.

**Figura 8 – Tarefas Frontend**

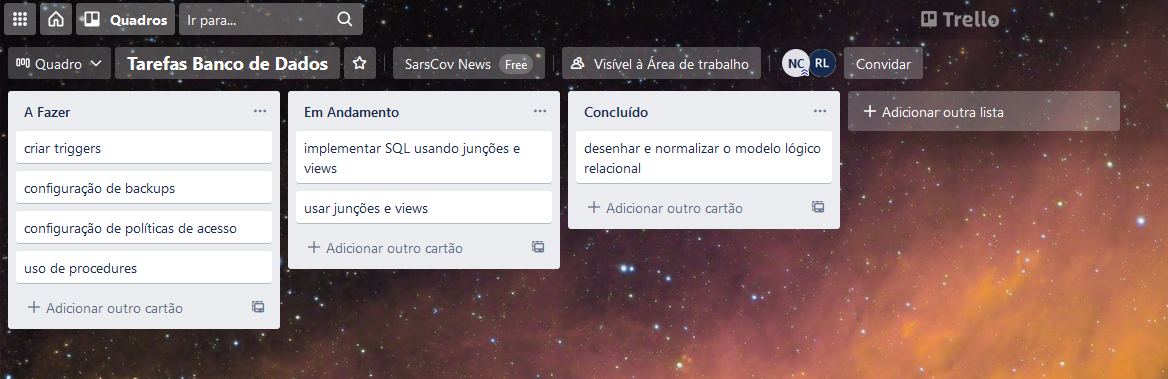
**Fonte:** Compilação do Autor**.**

**Figura 9 – Tarefas Backend**



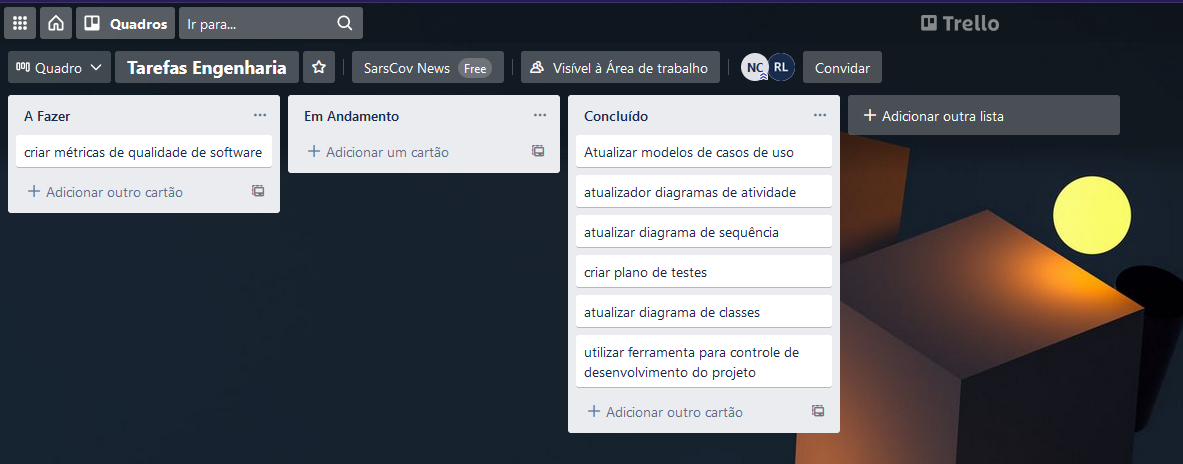
**Fonte:** Compilação do Autor**.**

**Figura 10 – Tarefas Banco de Dados**

****

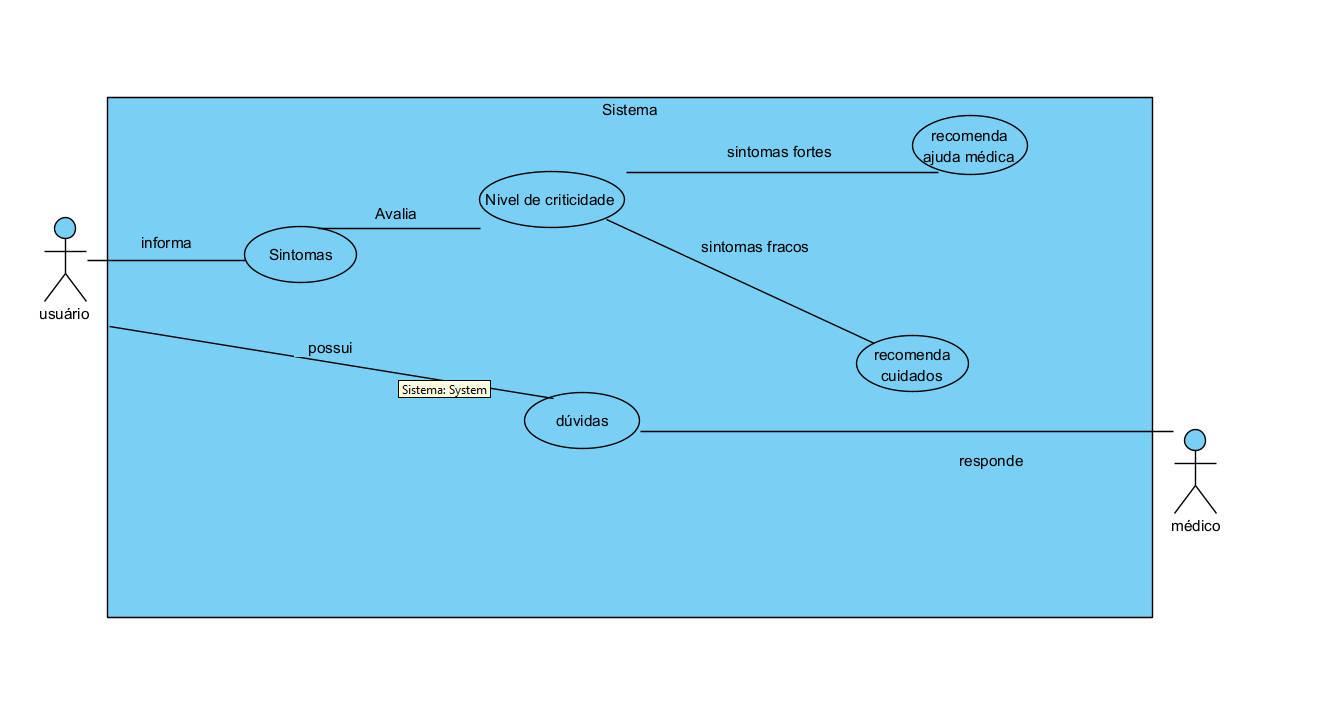
**Fonte:** Compilação do Autor**.**

**Figura 11 – Tarefas Engenharia de Software**



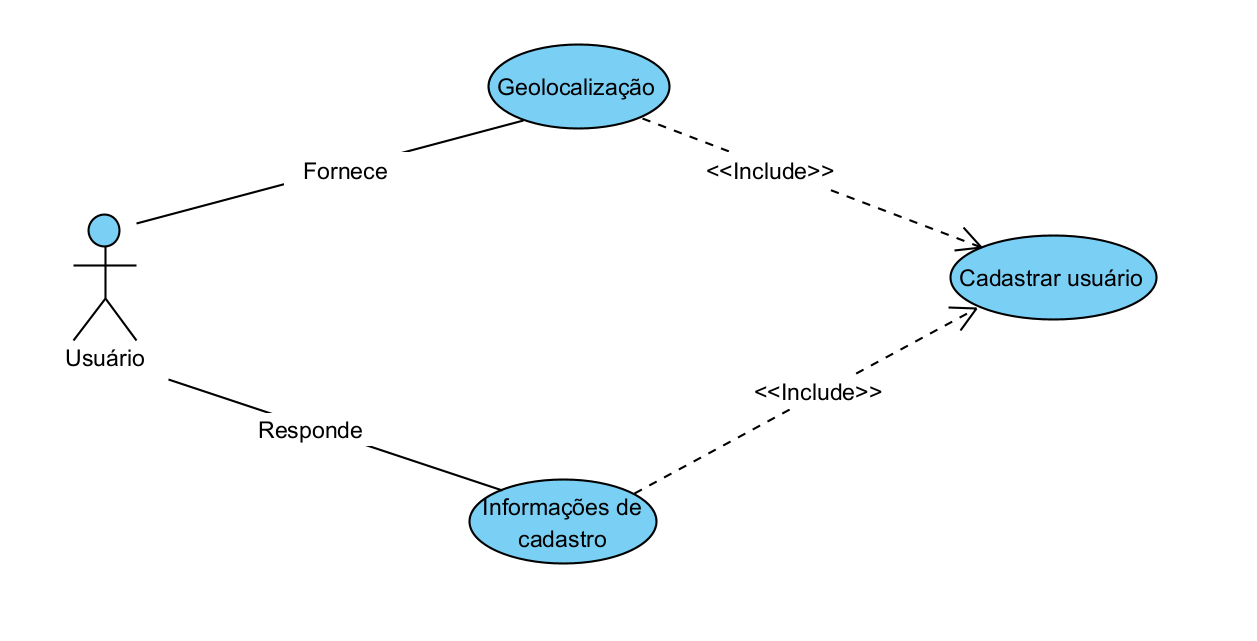
**Fonte:** Compilação do Autor**.**

**Figura 12 – Caso de uso 1**

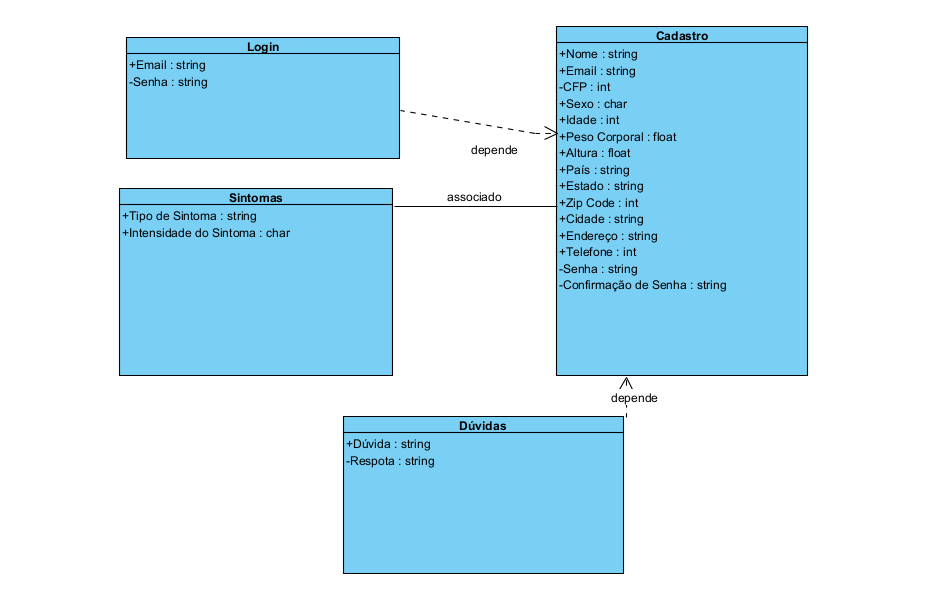
****

**Fonte:** Compilação do Autor**.**

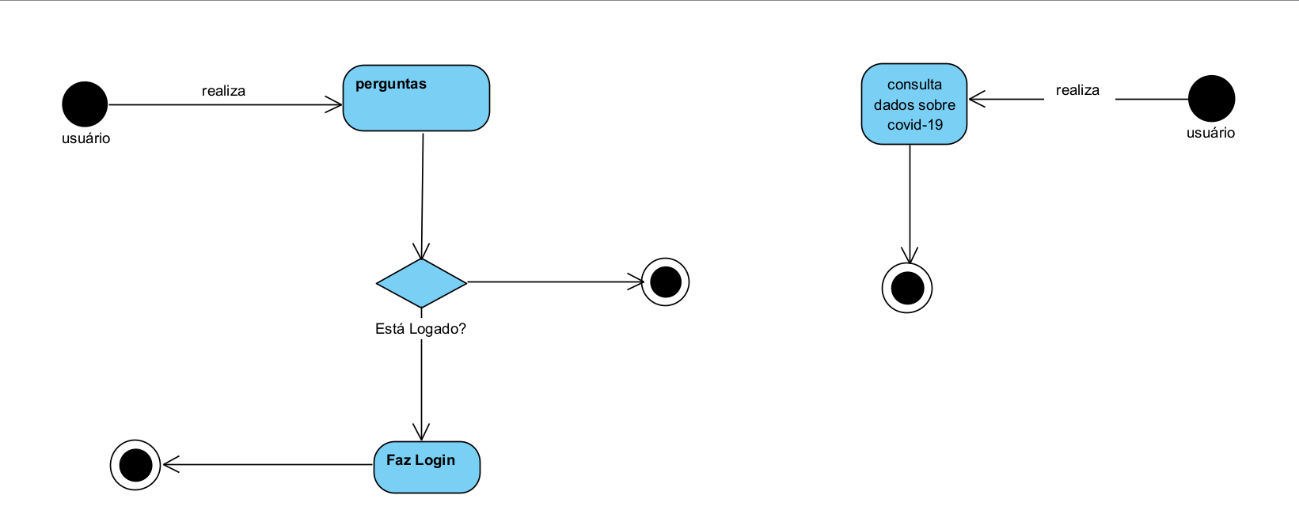
**Figura 13 – Caso de Uso 2**

****

**Fonte:** Compilação do Autor**.**

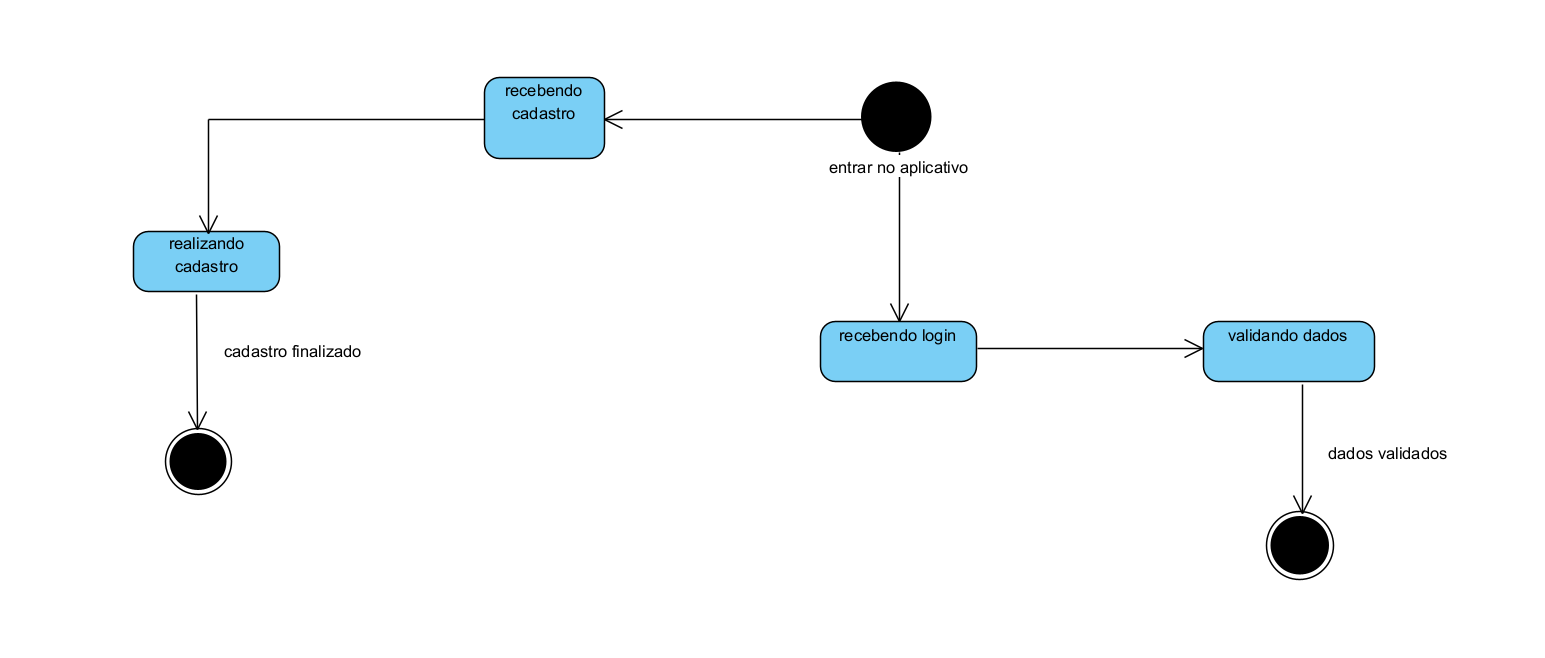
**Figura 14 – Diagrama de Classes****Fonte:** Compilação do Autor**.**

**Figura 15 – Diagrama de Atividades**

****

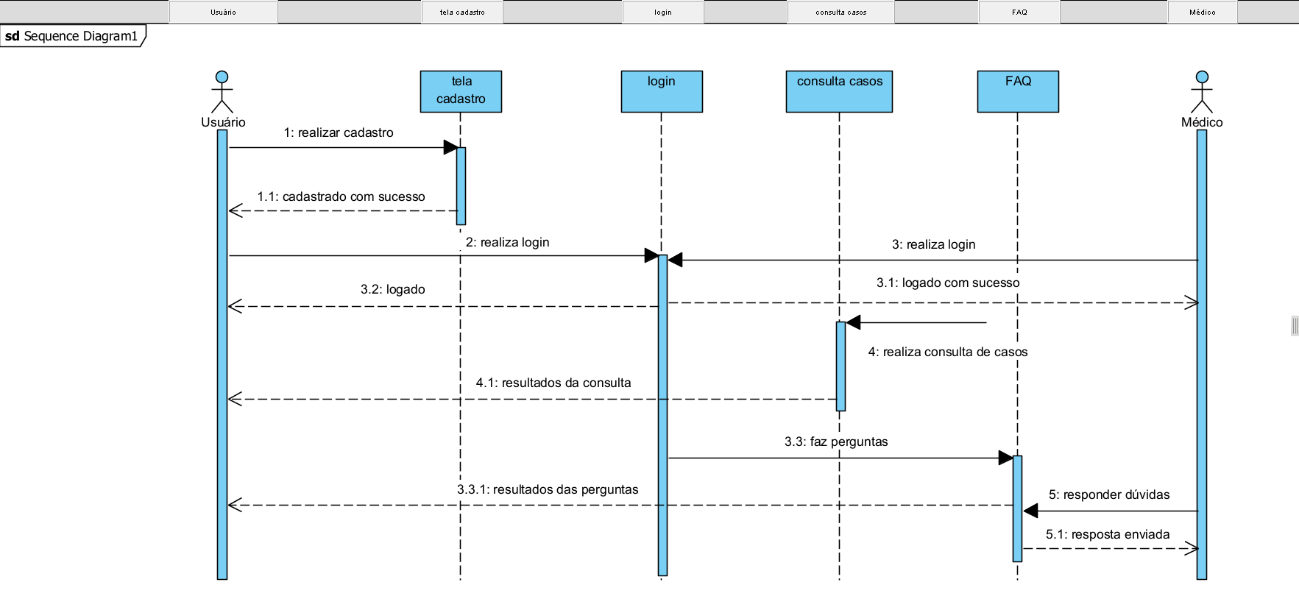
**Fonte:** Compilação do Autor**.**

**Figura 16 – Diagrama de Estado**

****

**Fonte:** Compilação do Autor**.**

**Figura 17 – Diagrama de Sequência**

****

**Fonte:** Compilação do Autor**.**

**3 CONCLUSÃO**

Sobre o desenvolvimento desde projeto durante este semestre, percebemos que evoluímos muito nossos conhecimentos sobre programação banco de dados e engenharia de softwares, e criamos um sistema funcional que pode servir para muitos. Podemos ainda atualizá-lo, implementando mais funções como quantidade de pessoas vacinadas no mundo todo, agora que está ocorrendo a distribuição de vacinas, além de melhorar a filtragem de casos e adicionar filtros por cidades e estados de cada país do mundo.

Ainda esse projeto mostrou que podemos criar sistemas/sites muito melhores e nos inserirmos no mercado de trabalho com facilidade.

**Apêndice A - Script Criação de Base de Dados**

CREATE DATABASE sarscov;

CREATE TABLE users (

id BIGSERIAL NOT NULL,

name varchar(50) NOT NULL,

cpf varchar(15) NOT NULL UNIQUE,

age int4 NOT NULL,

height int4 NOT NULL,

phone varchar(255) NOT NULL,

weight float4 NOT NULL,

sex char(1) NOT NULL,

adress varchar(255) NOT NULL,

zip\_code varchar(50) NOT NULL,

city\_id int8 NOT NULL, PRIMARY KEY (id)

);

COMMENT ON COLUMN users.id IS 'Numero de identificação do usuário';

COMMENT ON COLUMN users.name IS 'Nome do usuário';

COMMENT ON COLUMN users.cpf IS 'CPF do usuário';

COMMENT ON COLUMN users.age IS 'Idade do usuário';

COMMENT ON COLUMN users.height IS 'Altura do usuário em centimetros';

COMMENT ON COLUMN users.fone IS 'Contato de telefone do usuario';

COMMENT ON COLUMN users.weight IS 'Peso do usuário em Kg';

COMMENT ON COLUMN users.sex IS 'Sexo do usuário';

COMMENT ON COLUMN users.adress IS 'Rua usuario/paciente';

CREATE TABLE acesses (

id BIGSERIAL NOT NULL,

email varchar(255) NOT NULL UNIQUE,

password varchar(32) NOT NULL,

level int4 DEFAULT 1 NOT NULL,

user\_id int8 NOT NULL,

PRIMARY KEY (id)

);

COMMENT ON COLUMN acesses.id IS 'id de acesso';

COMMENT ON COLUMN acesses.email IS 'Usuário para acesso ao sistema';

COMMENT ON COLUMN acesses.password IS 'Senha de acesso de até 32 caracteres';

COMMENT ON COLUMN acesses.nivace IS 'Nivel de acesso ao sistema';

CREATE TABLE countries (

id BIGSERIAL NOT NULL,

name varchar(50) NOT NULL,

initials varchar(5) NOT NULL,

PRIMARY KEY (id)

);

COMMENT ON COLUMN countries.id IS 'Código do pais';

COMMENT ON COLUMN countries.name IS 'Nome por extenso do pais';

CREATE TABLE states (

id BIGSERIAL NOT NULL,

name varchar(50) NOT NULL,

country\_id int8 NOT NULL,

PRIMARY KEY (id)

);

COMMENT ON COLUMN states.id IS 'Código de identificação do estado';

COMMENT ON COLUMN states.name IS 'Nome por extenso do estado';

COMMENT ON COLUMN states.country\_id IS 'Código do pais';

CREATE TABLE cities (

id BIGSERIAL NOT NULL,

name varchar(50) NOT NULL,

state\_id int8 NOT NULL,

PRIMARY KEY (codcid)

);

COMMENT ON COLUMN cities.codcid IS 'Código da cidade';

COMMENT ON COLUMN cities.name IS 'Nome da cidade';

CREATE TABLE monitorings (

id BIGSERIAL NOT NULL,

degree int4 NOT NULL,

timestamps timestamp(255) NOT NULL,

user\_id int8 NOT NULL,

sympton\_id int8 NOT NULL,

created\_at timestamp NOT NULL,

updated\_at timestamp,

PRIMARY KEY (id)

);

COMMENT ON COLUMN monitorings.id IS 'Identificação do local de prontoatendimento';

COMMENT ON COLUMN monitorings.degree IS 'gravidade do sintoma';

CREATE TABLE symptoms (

id BIGSERIAL NOT NULL,

name varchar(50) NOT NULL,

PRIMARY KEY (id)

);

COMMENT ON COLUMN symptoms.id IS 'Código de identificação do registro do sintoma';

COMMENT ON COLUMN symptoms.name IS 'Nome por extenso do sintoma';

ALTER TABLE states ADD CONSTRAINT FKstates294072 FOREIGN KEY (country\_id) REFERENCES countries (id);

ALTER TABLE cities ADD CONSTRAINT FKcities434231 FOREIGN KEY (state\_id) REFERENCES states (id);

ALTER TABLE users ADD CONSTRAINT FKusers227988 FOREIGN KEY (city\_id) REFERENCES cities (codcid);

ALTER TABLE acesses ADD CONSTRAINT FKacesses906999 FOREIGN KEY (user\_id) REFERENCES users (id);

ALTER TABLE monitorings ADD CONSTRAINT FKmonitoring647781 FOREIGN KEY (user\_id) REFERENCES users (id);

ALTER TABLE monitorings ADD CONSTRAINT FKmonitoring749248 FOREIGN KEY (sympton\_id) REFERENCES symptoms (id);