

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

Etec DA ZONA LESTE

**Ensino Médio com Habilitação Profissional de Técnico em
Desenvolvimento de Sistemas**

Kauan Gallati Vieira Queiroz

Lucas Lima Sant Ana

Nicolas Mauricio Martins Coiado

Théo Kabir Novais de Carvalho

PETS'S PORTAL: Site de adoção de animais de estimação

Trabalho de Conclusão de Curso apresentado ao Curso do Ensino Médio com Habilitação Profissional de Técnico em Desenvolvimento de Sistemas, orientado pelo Prof. Jeferson, como requisito final para obtenção do título de Técnico em Desenvolvimento de Sistemas.

São Paulo

2021

Kauan Gallati Vieira Queiroz
Lucas Lima Sant Ana
Nicolas Mauricio Martins Coiado
Théo Kabir Novais de Carvalho

PET'S PORTAL: Site de animal de estimação

Trabalho de Conclusão de Curso apresentado ao Curso do Ensino Médio com Habilitação Profissional de Técnico em Desenvolvimento de Sistemas, orientado pelo Prof. Jeferson Roberto de Lima, como requisito final para obtenção do título de Técnico em Desenvolvimento de Sistemas.

São Paulo

2021

Dedicamos esse trabalho ao Diego, Chico, Tieta e Link por nos proporcionar o amor inocente de um animal de estimação.

Às ONGs cuidadoras de animais por assumirem esse compromisso e responsabilidade importantíssima na sociedade.

Dedicamos esse projeto a César Augusto Ferreira de Carvalho pelo apoio incondicional mesmo que não mais presente

AGRADECIMENTOS

Nós agradecemos aos nossos pais, familiares e responsáveis pela nossa formação como pessoa e cidadão. Aos professores de nossa escola, por nos instruírem no caminho da aprendizagem, em especial nossos orientadores, Jeferson Roberto de Lima e Rogério Bezerra Costa por nos acompanharem no processo de criação desse projeto. Aos nossos colegas e amigos da turma por sempre nos contagiarem com seu bom humor. Por fim, deixamos aqui registrado nosso imenso agradecimento ao Professor Marlon Marques por sempre nos acolher e aconselhar nos momentos mais turbulento passados por nós.

“Eu acredito que às vezes são as pessoas que ninguém espera nada que fazem as coisas que ninguém consegue imaginar”

ALAN TURING

RESUMO

Este sistema é formado com o fim de diminuir um problema social, abandono de animais na sociedade atual. Através de um site rápido, intuitivo e modernizado, dando a possibilidade de o usuário doar ou adotar um animal, podendo ser um cão ou um gato, atingindo assim interesses mútuos. Com métodos de fácil entendimento qualquer pessoa consegue acessar o site Pets Portal e usufruir de todos os recursos disponíveis.

Sendo composto integralmente de *HyperText Markup Language (HTML)*, *Cascading Style Sheets (CSS)* e *JavaScript* o site é capaz de divulgar animais em doação para outras pessoas, eventos de ONGs registradas e possibilitando a melhoria na problemática do abandono de animais.

Com a integração desse website em meio a sociedade será notório o aumento do costume da adoção animal.

ABSTRACT

This project was made with the objective of decrease one social problem, the animal abandonment in the actual society. With a fast, intuitive and modernized site, We give the possibility to give and adopt animals, can be both dogs or cats, reaching mutual interests between te users. With easy understanding methods, anyone can access the Pets Portal site and use all the available resources.

Being integrally compost by HyperText Markup Language (HTML), Cascading Style Sheets (CSS) and Javascript, the site can spread animals in donation, animal donation events by registered non-governmental organisations (NGOs) and enabling the imprivement in the animal abandonment problematic.

With the integration of this website with the society will be notorious the increase of animal adoption costume.

LISTA DE ILUTRAÇÕES

| | |
|--|----|
| Figura 1 - Estrutura básica do HTML..... | 14 |
| Figura 2 - Estrutura básica do seletor CSS..... | 14 |
| Figura 3 - Importação da biblioteca materialize..... | 15 |
| Figura 4 - Importação da biblioteca materialize via NPM..... | 15 |
| Figura 5 - Importação da biblioteca Axios via NPM..... | 17 |
| Figura 6 - Estrutura Básica do Express..... | 17 |
| Figura 7 - Exemplo de Autor..... | 21 |
| Figura 8 - Exemplo de Interação..... | 21 |
| Figura 9 - Exemplo de Include..... | 21 |
| Figura 10 - Exemplo de Extend..... | 22 |
| Figura 11 - Exemplo de Diagrama de Atividade..... | 22 |
| Figura 12 - Exemplo de Diagrama de Sequência..... | 23 |
| Figura 13 - Diagrama de Casos de Uso..... | 24 |
| Figura 14 - Diagrama de atividade (Cadastrar animal) | 41 |

| | |
|---|----|
| Figura 15 - Diagrama de atividade (Solicitar adoção) | 42 |
| Figura 16 - Diagrama de Atividade (Registrar eventos de doação) | 42 |
| Figura 17 - Diagrama de Sequência (Cadastrar animal) | 43 |
| Figura 18 - Diagrama de Sequência (logar)..... | 43 |
| Figura 19 - Diagrama de Sequência (Cadastrar evento) | 44 |
| Figura 20 - Diagrama de Sequência (Cadastrar usuário) | 45 |
| Figura 21 - Diagrama de Sequência (Doar animal) | 45 |
| Figura 22 - Tela home..... | 46 |
| Figura 23 - Barra de navegação (usuários não logados) | 46 |
| Figura 24 - Barra de navegação (doador/ adotante) | 46 |
| Figura 25 - Barra de navegação (ONGs) | 46 |
| Figura 26 - Barra de navegação (administrador)..... | 47 |
| Figura 27 - Tela Login..... | 47 |
| Figura 28 - Tela Cadastro (doador/ adotante) | 47 |

| | |
|--|----|
| Figura 29 – Tela cadastro (doador/ adotante) | 48 |
| Figura 30 – Tela Perfil..... | 49 |
| Figura 31 – Tela Doação..... | 49 |
| Figura 32 – Tela Adoção..... | 50 |
| Figura 33 – Tela Animal (Visão do dono) | 50 |
| Figura 34 – Tela Animal (Visão adotante) | 51 |
| Figura 35 – Tela Proposta de Adoção..... | 51 |
| Figura 36 – Tela Proposta de Adoções Aceitas..... | 52 |
| Figura 37 – Tela Lista Eventos..... | 52 |
| Figura 38 – Tela Evento (visão ONG responsável) | 53 |
| Figura 39 – Tela Evento (visão doador/ adotante) | 53 |
| Figura 40 – Tela Cadastrar Evento..... | 54 |
| Figura 41 – Tela Validar ONG..... | 54 |
| Figura 42 – Tela Validar Evento..... | 55 |
| Figura 43 – Tela lista de adoções..... | 55 |
| Figura 44 – Tela ver reports..... | 55 |

| | | | | |
|--------------------|-----------|----------|--------------------|--------------|
| Figura | 45 | - | Recuperação | de |
| senha..... | | | | 56 |
| Figura | 46 | - | Termos | de |
| uso..... | | | | 56 |
| Figura | 47 | - | | Saiba |
| mais..... | | | | 57 |
| Figura | 48 | - | Rodapé | da |
| página..... | | | | 57 |

LISTA DE TABELAS

Tabela 1 – Documentação de Caso (Manter animais)

Tabela 2 – Documentação de Caso (Visualizar animal)

Tabela 3 – Documentação de Caso (Registrar doação de animais)

Tabela 4 – Documentação de Caso (Manter usuários)

Tabela 5 – Documentação de Caso (Visualizar usuários)

Tabela 6 – Documentação de Caso (Cadastrar doador/adotante)

Tabela 7 – Documentação de Caso (Acessar perfil)

Tabela 8 – Documentação de Caso (Reportar perfil)

Tabela 9 – Documentação de Caso (Manter eventos de doação)

Tabela 10 – Documentação de Caso (Validar eventos de doação)

Tabela 11 – Documentação de Caso (Fazer login)

Tabela 12 – Documentação de Caso (Adotar animal)

LISTA DE ABREVIATURAS

API - *Application Programming Interface*

CDN - *Content Delivery Network*

CORS – *Cross-Origin Resource Sharing*

CSS - *Cascading Style Sheets*

DOM - *Document Object Model*

HTML - *Hyper Text Markup Language*

HTTP – *HyperText Transfer Protocol*

JS – *JavaScript*

JSON – *JavaScript Object Notation*

JSX - *JavaScript XML*

NPM - *Node Package Manager*

OMS - *Organização Mundial da Saúde*

OMT - *Object Modeling Technique*

ONG - *Organização não Governamental*

OOSE - *Object Oriented Software Engeneering*

SGBD – *Sistema Gerenciador de Banco de Dados*

SO – *Sistema Operacional*

UI - *User Interface*

UML – *Unified Modeling Language*

XML - *Extensible Markup Language*

Sumário

| | |
|--|----|
| LISTA DE TABELAS | 12 |
| 1. INTRODUÇÃO | 12 |
| 2. REFERENCIAL TEÓRICO | 13 |
| 2.1. HTML | 13 |
| 2.2. CSS..... | 14 |
| 2.2.1. Materialize..... | 15 |
| 2.3. Tecnologias JavaScript..... | 16 |
| 2.4. Banco de dados..... | 19 |
| 2.4.1. Relacional e não relacional | 19 |
| 2.4.2. MongoDB | 19 |
| 2.5. UML | 20 |
| 2.5.1. Diagrama de Caso de Uso..... | 21 |
| 2.5.2. Diagrama de Atividade | 22 |
| 2.5.3. Diagrama de Sequência..... | 23 |
| 3. Desenvolvimento | 24 |
| 3.1. Diagrama de Casos de Uso..... | 25 |
| 3.2. Documentação de Casos de Uso | 25 |
| 3.3. Diagrama de Atividade..... | 44 |
| 3.4. Diagrama de Sequência | 46 |
| 3.5. Aplicação WEB | 50 |
| 4. Conclusão..... | 63 |
| REFERÊNCIAS..... | 64 |

1. INTRODUÇÃO

Como descrito pela G1 (2014), PET é uma palavra criada na Escócia, durante o século XIV, para definir “animal domado”. Esses animais veem evoluindo juntamente do homem, numa relação de mais de 20 mil anos. Relação qual começou como um jogo de interesses de ambas as partes. Segundo teorias, lobos seguiam os humanos para se alimentar dos restos de comida. Após notar que outros animais dificilmente atacavam enquanto esses estavam por perto, o humano utilizou inconscientemente da seleção artificial, autorizando os caninos mais dóceis a continuarem por perto.

Entretanto, é citado por D’andretta (2012) que essa relação mudou com o passar do tempo. O animal passou a ser dependente do ser humano para sua sobrevivência, sendo adotado como membro da família. No entanto, é dito pela CNN (2020) que as pessoas desconhecem fatores na hora da adoção, como os gastos de alimentação, saúde, tosa, limpeza e espaço, além das necessidades de tempo para passeios e cuidados específicos. Essa falta de informação antes da adoção resulta num aumento da taxa de abandono animal, sendo dito por Veiga (2020), jornalista da BBC News, que diversas Organizações não Governamentais (ONG) têm tido um aumento de e-mails a procura de um novo dono para seu animal de estimação.

De acordo com Jusbrasil (2013), é estimado que no Brasil há cerca de 30 milhões de animais errantes, sendo 10 milhões de gatos e 20 milhões de cães. Em grandes cidades, há cerca de 1 cão para cada 5 pessoas, entre esses, 10% são abandonados. Não é diferente em cidades menores, algumas chegam a até $\frac{1}{4}$ da população humana.

Devido à falta de informações pré-adoção, é notável a falta de transparência na comunicação entre futuros adotantes com as ONG’s e responsáveis pelos animais. Visando a melhora dessa comunicação, juntamente com a aplicação de uma solução tecnológica, será feito um site de doação e adoção de pets, para que se torne mais comum essa prática, de forma a consolidar a relação entre o dono e sua estimação.

Para a construção dessa solução tecnológica, serão feitas pesquisas bibliográficas, pesquisas de campo, diversos tipos de diagramas e bancos de dados em *MySQL*. E serão utilizadas ferramentas como *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS), *JavaScript* (JS), *ReactJS*, *NodeJs* e *Materialize*. A

metodologia utilizada para gerir o projeto será SCRUM com a organização interna feita pelo modelo *Kanban*.

2. REFERENCIAL TEÓRICO

Este capítulo possui o embasamento teórico das tecnologias que serão utilizadas para a elaboração do projeto de pesquisa do site de doação e adoção de animais Pet's Portal.

2.1. HTML

Segundo Castro e Hyslop (2013) o HTML é uma linguagem começada na década de 90, como um documento pequeno para detalhar todos os elementos utilizados na construção da página *web*.

“Isso também significa que muito do HTML5 funciona em navegadores novos e antigos; ser compatível com versões anteriores é um princípio de design chave do HTML5”, (CASTRO; HYSLOP, 2013, p. xvi).

A estrutura básica do HTML5 pode ser montada de acordo com a figura 1. De acordo com Ferreira e Eis (2011), as funções das tags utilizadas são:

- DOCTYPE – Não é considerado uma tag HTML, e sim uma instrução para o navegador sobre a versão de código e marcação utilizada. É necessário que venha antes da tag `html`;
- `html` – É uma série de elementos onde alguns elementos seguem uma hierarquia, sendo o Principal elemento dessa árvore a tag `html`. Essa tag possui o elemento `lang` que define a linguagem principal do documento;
- `Head` – É onde se localiza a parte inteligente da página, ou seja, é onde fica as informações sobre a página e seu conteúdo publicado;
- `Metatag charset` – Tal *tag* se encontra no cabeçalho do documento, ele especifica a tabela de caracteres que será utilizada no site, a tabela utilizada comumente é a tabela Unicode, que supri a necessidade de diferentes caracteres do mundo inteiro fazendo com que o site seja interpretado e reconhecido por qualquer browser;
- `Link` – A *tag* `link` é a responsável por fazer a conexão entre o documento e outros arquivos que serão utilizados, um tipo de arquivo muito utilizado no

desenvolvimento web é o *Cascading Style Sheets* (CSS), que pode ser importado no arquivo html utilizando o parâmetro *rel* com o valor *stylesheet*, o *type* como *text/css* e o *src* com o caminho até o arquivo;

- Body – Indica o corpo da página. O conteúdo da página ficará nessa tag.

Figura 1 - Estrutura básica do HTML

```
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="UTF-8">
5      <title></title>
6      <link rel="stylesheet" href="style.css">
7  </head>
8  <body>
9
10 </body>
11 </html>
```

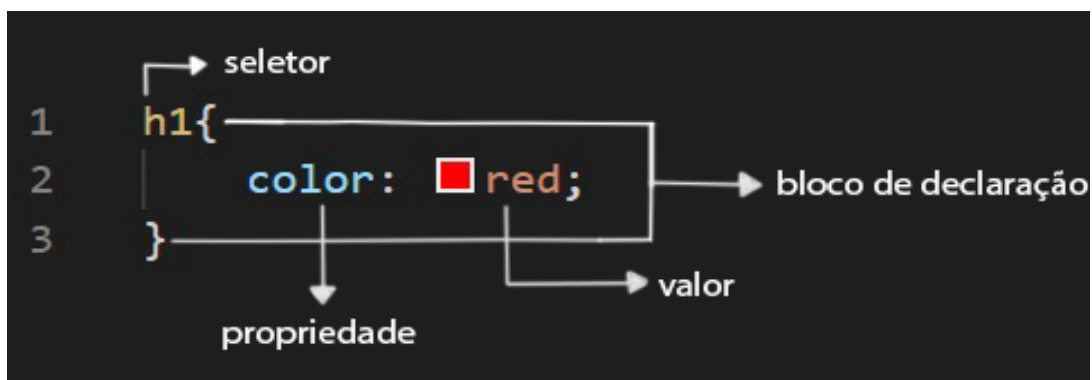
Fonte: Autoria própria, 2021.

2.2. CSS

Como informa Miletto e Bertagnolli (2014), o CSS, ou folhas de estilo em cascata, é um mecanismo para a formatação da aparência das páginas HTML. Através dessas folhas é possível criar estilos individuais para cada elemento que constitui o site.

Castro e Hyslop (2013) descrevem que o CSS é formado por regras de estilo, que conforme é visto na figura 2, é formado por duas partes principais: o seletor, que determinará os elementos a serem modificados, e o bloco de declarações, composto por uma ou mais declarações, formadas por atributos e valores, que determinam o que será feito nas partes.

Figura 2 - Estrutura básica do seletor CSS



Fonte: Autoria própria, 2021.

2.2.1. Materialize

Scudero (2018) descreve o Materialize como um *framework front-end* que tem o *Material Design* como base de inspiração para cores, ícones e formatos. Segundo ele, o Materialize pode ser utilizado com instalação via *Content Delivery Network* (CDN) ou através da importação de sua biblioteca.

A instalação do Materialize pode ser feita via CDN como indicado na figura 3.

Figura 3 – Importação da biblioteca materialize

```
<!-- Compiled and minified CSS -->
<link rel="stylesheet" href="https://cdn.jsdelivr.net/ajax/libs/materialize/1.0.0/css/materialize.min.css">

<!-- Compiled and minified JavaScript -->
<script src="https://cdn.jsdelivr.net/ajax/libs/materialize/1.0.0/js/materialize.min.js"></script>
```

Fonte: Autoria própria, 2021.

Também é possível importar o Materialize via *Node Package Manager* (NPM), que é o gerenciador de pacotes do Node JS, como indicado na figura 4.

Figura 4 – Importação da biblioteca materialize via NPM

```
C:\Users\NICOLASMAURICIOMARTI\Desktop>npm install materialize-css@next
```

Fonte: Autoria própria, 2021.

Segundo Gasparotto (2017), as vantagens do Materialize sobre outros *frameworks front-end* são:

- O seu visual que permite a criação de sites esteticamente agradáveis com pouco esforço;

- A sua qualidade de seu desenvolvimento, pois poucos bugs foram encontrados em seus serviços;
- O seu *grid* similar à de outros *frameworks*;
- Seus elementos únicos, como o efeito *Parallax* e seus *Toasts*, que não estão disponíveis em *frameworks* concorrentes.

2.3. Tecnologias JavaScript

Como já dito por Silva (2010), para gerar interatividade entre o usuário e a página web, as empresas Netscape e Sun Microsystems criaram o *JavaScript* em 1995. Também é dito que um formulário não pode ser feito apenas com o HTML, pois ele não trata e manipula os dados e nem os utiliza de sozinho, para isso existem linguagens como PHP, ASP, Java, Ruby e Python que, diferente do *JavaScript*, rodam no lado do servidor, o *JavaScript* roda do lado do cliente, a partir de seu navegador.

Como já dito por Mozilla Foundation (2021), o JavaScript é uma linguagem de script feito para o ambiente *web*, projetado para funcionar de forma leve e funcionando como uma linguagem baseada em protótipos, multi-paradigma e dinâmica.

Freires (2019) define o *React JS* como uma biblioteca de *JavaScript* que serve para auxiliar na criação de interfaces para o usuário, que foi desenvolvida e é mantida pelo Facebook. Além disso, ele também explica que *React* foi criado para resolver problemas da camada visual, e assim facilitando o trabalho utilizando componentes interativos e reutilizáveis para a interface do usuário.

E como disse Takagi (2021), o *React JS* é uma solução para a camada visual, pois ele interage diretamente com o seu próprio *Document Object Model* (DOM), sem que o usuário tenha que alterar o que deseja diretamente, pois as alterações são feitas apenas com base em instruções.

Agora sendo mais direto na explicação de como o React funciona, Facebook Inc (2021) explica que o React permite que o usuário faça a divisão da *User Interface* (UI), em componentes independentes e que podem ser reutilizados, para que o desenvolvimento seja pensado em cada pedaço isoladamente.

Como explica Freires (2019), o JavaScript XML (JSX) é derivado da estrutura de *Extensible Markup Language* (XML), e é uma extensão do JavaScript que não é

obrigatória ao escrever códigos no React, porém ela auxilia o desenvolvedor a visualizar melhor suas linhas de código.

Facebook Inc (2021) recomenda o JSX pois ele é uma extensão que descreve como o UI deveria parecer, e ele faz isso pois possui todo o poder do JavaScript, incluindo o de criar “elementos” do React.

Informado por Darski (2016), o Node.js é um ambiente que executa arquivos JavaScript de maneira assíncrona orientada a eventos, de modo que não utiliza do modelo padrão de outras linguagens de fazer múltiplas coisas ao mesmo tempo, os threads do SO (Sistema Operacional), que são ineficientes e difíceis de se usar, por mais que ainda seja possível de utilizá-los no Node, porém o uso não é obrigatório.

Segundo Krefta (2019), o Axios é uma biblioteca *Application Programming Interface* (API), que é fundamentada em *promises*, e está disponível em navegadores ou em ambientes de servidor Node.js.

Segundo Sousa (2021), para instalar o Axios, basta executar um comando de instalação via NPM, como na figura 5.

Figura 5 – Importação da biblioteca Axios via NPM

```
C:\Users\NICOLASMAURICIOMARTI\Desktop\MinhaPasta>npm install axios
```

Fonte: Autoria própria, 2021.

De acordo com Rodrigues (2015), o *Express* é um *framework* minimalista e flexível para o Node.js, ele também conta com um robusto conjunto de recursos e métodos para o desenvolvimento *web* do lado do servidor.

Uma funcionalidade que será muito utilizada durante o projeto é a confecção de *middlewares*, que de acordo com Urban (2017), tem como sua principal função, interceptar as requisições *HyperText Transfer Protocol* (HTTP) feitas para a *Application Programming Interface* (API).

Um exemplo da estrutura do express pode ser vista na figura 6, onde, como já mostrado por Mozilla (2021), o comando *require* que recebe como parâmetro o express, cria a aplicação express, na linha 12 é visto o comando que defini uma rota de solicitação HTTP GET, que, pela linha seguinte, a partir da função *.send*, envia

um objeto JSON como resultado, e na última linha, pela função `.listen`, cria uma conexão de acordo com a variável `port`.

Figura 6 – Estrutura básica do Express

```
1  const express = require('express')
2  const app = express()
3  const routerUsuarios = require("../routes/usuarios")
4  const port = process.env.PORT || 3000
5
6  app.use("/uploads", express.static("uploads"))
7  app.use(express.urlencoded({ extended: true }))
8  app.use(express.json())
9
10 app.use("/usuarios", routerUsuarios)
11
12 app.get("/", (req, res)=>{
13   res.status(200).send({ message: "rota raiz acessada" })
14 })
15
16 app.listen(port, ()=>console.log(`aplicação aberta na porta ${port}`))
```

Fonte: Autoria própria, 2021.

De acordo com Santos (2017), o mongoose é uma biblioteca javascript para Node.JS utilizada para manipular o banco de dados não relacional MongoDB que pode ser facilmente instalado pelo NPM.

De acordo com Anjos (2016), o bcrypt é um algoritmo utilizado para encriptar dados criando hashes. Ele será utilizado no projeto para efetuar a segurança dos dados.

Ribeiro Filho (2018) diz que o bcrypt é uma ótima escolha para tratar principalmente de senhas, pois permite que a senha seja salva encriptada no banco de dados, dificultado o entendimento de um possível invasor, além de todos os dados serem encriptados com uma variável que identifica o poder de processamento para encriptar e decodificar o mesmo, e como o dado é salvo encriptado, o próprio algoritmo faz a verificação da senha fornecida com o *hash* armazenado no sistema.

O multer servirá de ferramenta para armazenar as fotos dos usuários no projeto, e guardará o seu *path* no banco de dados, demonstrado por Teixeira (2021), o multer é usado como *middleware* para lidar com informações enviadas em formatos *multipart* e *form-data*, essa biblioteca é usada principalmente para lidar com uploads de arquivos.

De acordo com, Mozilla (2021), o *Cross-Origin Resource Sharing* (CORS) é uma tecnologia que permite que uma aplicação seja usada a partir de outro domínio. Tal

tecnologia será de extrema importância para o projeto, visto que utilizaremos de uma API que será consumida pelo site do projeto.

O JWT servirá para efetuar logins e manter a segurança dos mesmos, é comum em aplicações o uso de tokens, que contenham os dados do usuário criptografados onde a API possa fazer a verificação para liberar o acesso às rotas, como já explicado por Aylan (2019), tal ferramenta permite que o usuário envie seus dados à API que a retorna um objeto JSON com o token assim que um *login* é feito, dessa forma, sempre que o cliente desejar acessar um alguma rota, ele enviará o token para a API, que fará a verificação do mesmo para liberar o acesso.

2.4. Banco de dados

Explicado por Rezende (2006), um banco de dados é quando uma grande quantidade de informações relacionadas está junta. Para gerenciar esse montante de informação temos sistemas gerenciadores de bancos de dados (SGBD), que são softwares responsáveis por manipular essas informações.

2.4.1. Relacional e não relacional

De acordo com Devmedia (2014) o modelo Relacional de banco de dados foi proposto por Edgar Codd, da IBM, na década de setenta, e desde então segue como a estrutura líder quando se trata de bancos de dados, essa estrutura organiza os dados em tabelas com linhas e colunas, as linhas representam os conjuntos de dados, como em uma tabela de usuário, cada usuário seria uma linha, e as colunas os dados como nome, telefone, *e-mail* e endereço.

Bancos de dados não relacionais, ou NoSQL, são bancos de dados que não seguem a estrutura em tabelas do modelo relacional. Explicado por Tejada e Olprod (2018), os bancos de dados não relacionais armazenam os dados de maneira otimizada para cada caso, tem seus dados armazenados em documentos, geralmente em formato de objetos JavaScript, chamado de *JavaScript Object Notation* (JSON)

2.4.2. MongoDB

Como dito por Hows, Mambrey e Plugge (2015), O MongoDB é um banco de dados não relacional com ótima performance em velocidade e escalabilidade, tem uma configuração simples e facilita o fluxo de trabalho por possibilitar o que o foco do usuário se concentre em tarefas à mão, sem se preocupar com a configuração das estruturas de dados. O banco de dados em MongoDB, por ser um NoSQL, não

segue a estrutura relacional de tabelas com linhas e colunas, guardando os dados nas chamadas *Collections*, que guardam os documentos JSON com os dados. O MongoDB também é muito famoso por ser gerenciado em JavaScript, facilitando a comunicação em os documentos JSON e a aplicação.

2.5. UML

A Unified Modeling Language (UML) é, de acordo com Booch, Rumbaugh e Jacobson (2005), uma linguagem gráfica para visualização, especificação, construção e documentação de sistemas de softwares mais complexos. Ela proporciona um padrão de planejamento da arquitetura dos projetos de sistemas.

Como descrito por Vieira (2003), a UML surgiu em meados da década de 90, com a unificação de métodos já existentes (BOOCH, *Object Oriented Software Engeneering* ou OOSE, e *Object Modeling Technique* ou OMT), sendo seus criadores, respectivamente Grady Booch, Ivar Jacobson, James Rumbaugh. A UML é linguagem líder de modelagem de sistemas orientados a objetos, apresentando diversas ferramentas que a complementam.

Os desenvolvedores da UML, Booch, Rumbaugh e Jacobson (2005), comparam a modelagem com a construção de edifícios, descrevendo que há diferença de porte de projetos e mostrando a importância da busca do equilíbrio entre exigências do cliente e as limitações da tecnologia atual.

“Se quiser construir uma casa de qualidade, que atenda às necessidades de sua família e respeite os códigos de edificação da região, também será preciso desenhar alguns esboços de projetos, com a finalidade de pensar sobre o uso pretendido para cada cômodo e detalhes práticos de energia, circulação e encanamento”. (BOOCH; RUMBAUGH; JACOBSON, 2005)

Conforme Vieira (2003), a UML possui ferramentas próprias tanto para especificação estrutural quanto para especificação do comportamento de um sistema. Tal especificação pode ser feita de forma textual (glossário de termos e caso de uso), ou diagramática (diagramas de classe, de casos de uso, de sequência, ...). Unificando uma série de ferramentas que se complementam a fim de uma maior precisão na especificação do modelo.

2.5.1. Diagrama de Caso de Uso

Segundo Rodrigues (2013), o diagrama de caso de uso demonstra ações que acontecem sequencialmente, assim constituindo o cenário do diagrama, e cenários alternativos, tendo o intuito demonstrar hábitos de um sistema ou parte dele através de interações com autores, o caso de uso mostrará a funcionalidade de um sistema, evidenciando relações externas com outras entidades. Um caso de uso retrata seus procedimentos, incluindo variações, de um sistema inteiro e até suas subclasses, podendo vir a descrever a técnica e suas implementações.

Composição de um diagrama de caso de uso:

- Atores: Agente externo (uma pessoa ou um sistema) que interage com o sistema, dividindo-se em primário que interage diretamente e secundário que somente faz um serviço, representado pela Figura 7;

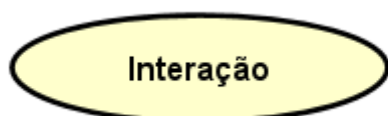
Figura 7 – Exemplo de Autor



Fonte: Autoria própria, 2021.

- Interação: Comunicação dos atores com o sistema, representado na Figura 8;

Figura 8 – Exemplo de Interação

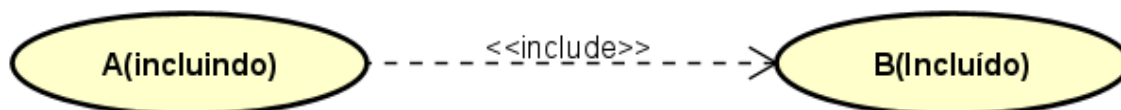


Fonte: Autoria própria, 2021.

- Associação: É a comunicação entre dois sistemas ou agentes com a facilitação de comunicação, podendo ser *Includ*, *Extend* e *Generalization*, sendo esses os mais usados em diagramas de caso de uso.
 - *Include*: Quando o caso de uso A inclui o caso de uso B, significa que sempre que A for executado o caso B iniciará em retrospecto. A

direção do relacionamento é do caso de uso que está incluindo para o caso de uso incluído, representado na Figura 9.

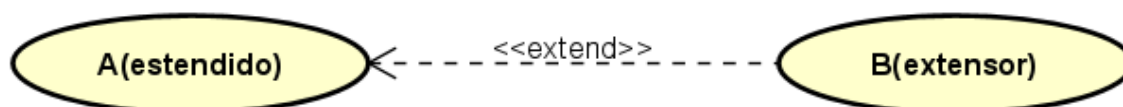
Figura 9 – Exemplo de Include



Fonte: Autoria própria, 2021.

- **Extend:** Quando o caso de uso B estende o caso de uso A, significa que quando A for executado o caso B poderá ser executado, porém não obrigatoriamente. A direção do relacionamento é do caso de uso extensor (aqui o caso de uso B) para o caso de uso estendido (aqui o caso de uso A), representado pela Figura 10.

Figura 10 – Exemplo de Extend

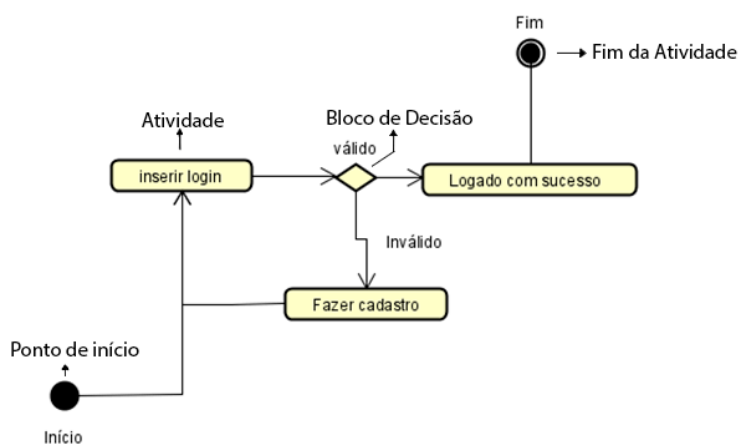


Fonte: Autoria própria, 2021.

2.5.2. Diagrama de Atividade

Guedes (2018), o diagrama de atividade é usado com a finalidade de modelagem de atividades sequenciais, sendo um dos diagramas mais específicos entre os citados no sistema, o uso do diagrama é essencial na modelagem no caso de uso, especificando etapas e processos. Uma atividade é composta por um conjunto de ações, ou seja, os passos que serão necessários para a conclusão de tal como iniciada.

Figura 11 – Exemplo de Diagrama de Atividade

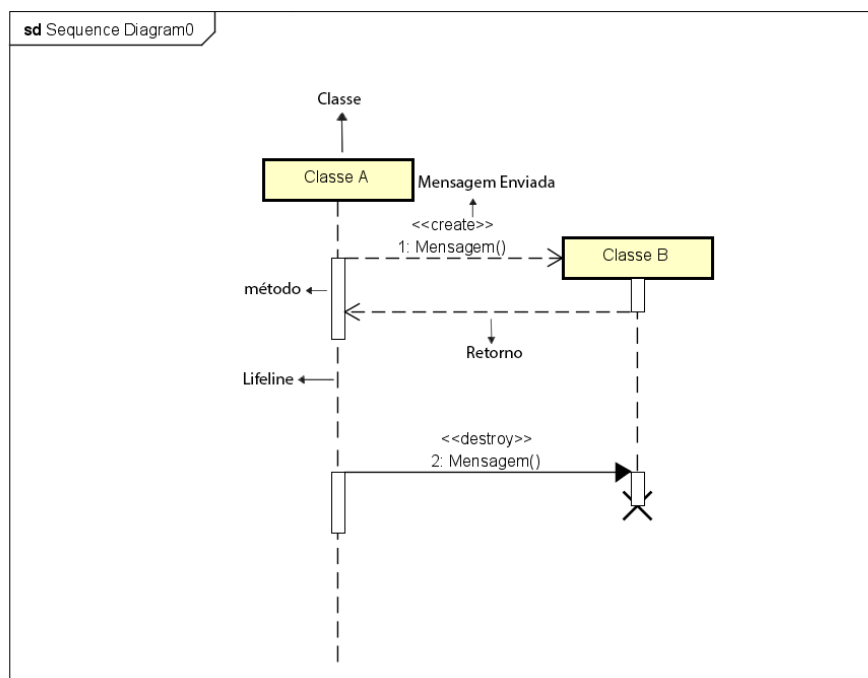


Fonte: Autoria própria, 2021.

2.5.3. Diagrama de Sequência

De acordo com Guedes (2018), o diagrama de sequência é um diagrama comportamental que determina a sequência de eventos que ocorrem em um processo, exibindo as mensagens e os elementos que devem ser disparados e em que ordem. Sendo assim, a principal função desse diagrama é determinar a ordem de eventos, as mensagens a serem enviadas, os métodos que forem chamados e a forma que os objetos performam durante o processo.

Martins (2010) descreve o diagrama da seguinte forma, ilustrada na Figura 12, na parte superior são representadas as classes. As linhas pontilhadas que saem das classes, chamadas de Lifeline, posicionam os métodos no diagrama, representadas pelas caixas brancas. As setas que conectam os métodos são os envios de mensagens e os retornos.

Figura 12 – Exemplo de Diagrama de Sequência

Fonte: Autoria Própria, 2021.

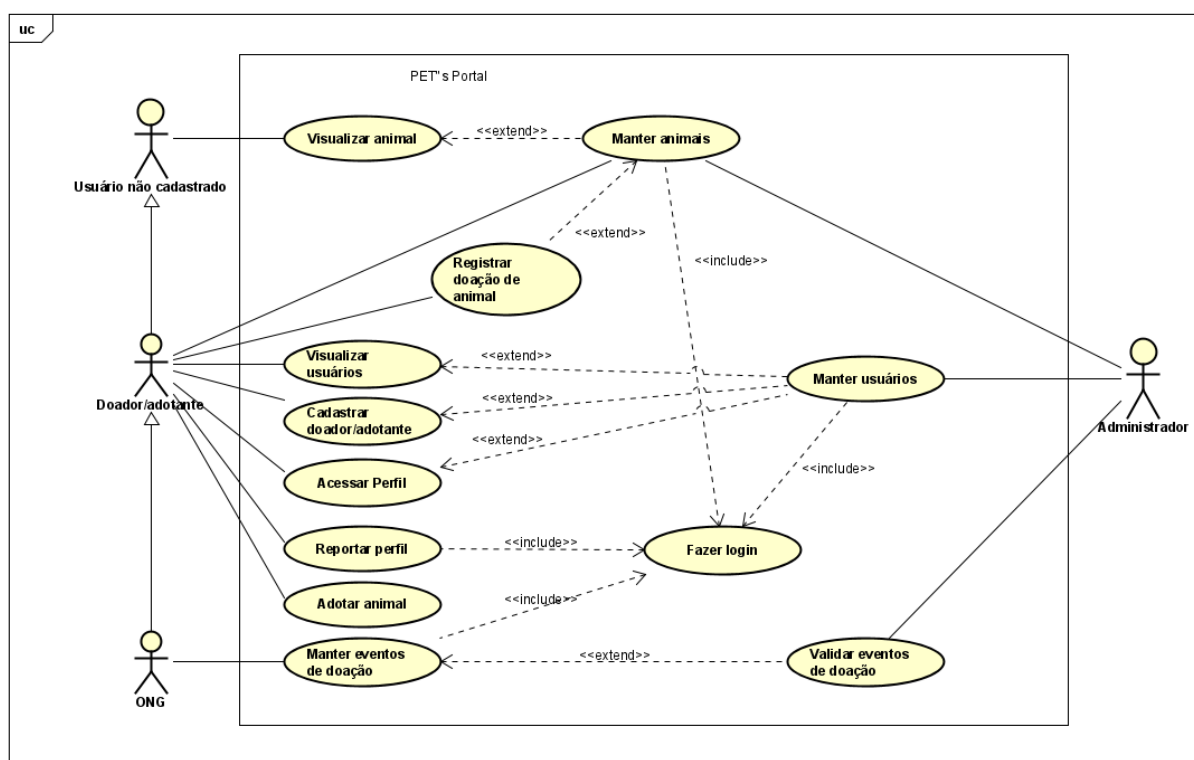
3. Desenvolvimento

Este capítulo possui a documentação e o desenvolvimento utilizados na elaboração do projeto de elaboração do site de doação e adoção de animais Pet's Portal. Será apresentado os diagramas de caso de uso, atividade e sequência do projeto, juntamente a documentação do caso de uso e as telas do sistema.

3.1. Diagrama de Casos de Uso

O diagrama de casos de uso abaixo, na figura 13, exibe a interação dos usuários com a aplicação, ilustrando a acessibilidade e as ações que o usuário poderá realizar através dos casos de uso e suas relações.

Figura 13 – Diagrama de Casos de Uso



Fonte: Autoria Própria, 2021.

3.2. Documentação de Casos de Uso

A documentação de casos de uso abaixo, na tabela 1, especifica as interações dos usuários com a aplicação, situadas no diagrama de casos de uso da figura 13. Ela define como funcionará cada interação, seus atores, condições, permissões e regras de negócio quando necessário.

Tabela 1 – Documentação de Caso (Manter animais)

| Nome do Caso de Uso: | Manter animais |
|----------------------|--------------------------|
| Caso de Uso Geral | |
| Ator Principal | Administrador |
| Atores Secundários | Doador/adotante e ONG's. |

| | |
|---|---|
| Resumo | Este caso de uso determina a criação, leitura, atualização e exclusão dos animais. |
| Pré-Condições | 1. Estar logado |
| Pós-Condições | |
| Permissões | <ol style="list-style-type: none"> 1. Usuários cadastrados (Doador/Adotante e ONG's) poderão registrar animais. 2. Usuários cadastrados poderão editar e excluir somente os animais vinculados a própria conta. 3. Usuários administrador possuem todas as permissões de acesso. |
| Regras de negócio | <ol style="list-style-type: none"> 1. Usuários do tipo Doador/adotante e ONG poderão editar e excluir somente animais vinculados a suas respectivas contas. |
| Fluxo alternativo I – Cadastro de animal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessa a página de doação | |
| | 2. Redireciona para a o formulário de cadastro de animais. |
| | <ol style="list-style-type: none"> 3. Exibe o formulário que requisita os seguintes dados: <ul style="list-style-type: none"> • Nome do animal; • Espécie; • Pelagem; • Idade; • Observação; • Vacinas • Alergias • Doenças |
| 4. Preenche formulário e confirma o cadastro. | |
| | 5. Cadastra o animal no banco de dados e possibilita sua visualização no perfil do doador e na aba de listagem dos animais. |
| Fluxo alternativo II – Visualização de animal (vinculado a sua conta) | |

| Ações do Ator | Ações do Sistema |
|---|--|
| 1. Acessar o seu perfil. | |
| | 2. Redirecionar o usuário para o próprio perfil |
| 3. Selecionar o animal que deseja visualizar na aba de animais doados. | |
| | 4. Abrir a página de visualização do animal. |
| Restrições/confirmações | 1. Somente usuários cadastrados terão uma aba de perfil. |
| Fluxo alternativo III – Visualização de animal (não vinculado a sua conta) | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar aba de adoção | |
| | 2. Redirecionar para aba de animais cadastrados. |
| 3. Selecionar animal que tem interesse de visualizar | |
| | 4. Exibir página de exibição do animal. |
| Restrições/confirmações | 1. Somente usuários cadastrados poderão visualizar as informações de contato dos doadores. |
| Fluxo alternativo IV – Edição de animal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar seu perfil | |
| | 2. Redirecionar o usuário para o próprio perfil |
| 3. Acessar o ícone de opções próxima ao animal que deseja editar e selecionar a opção “editar”. | |
| | 4. Redirecionar para um formulário de edição das informações do animal. |

| | |
|---|---|
| 5. Alterar os dados desejados e confirmar o envio. | |
| | 6. Atualizar os dados do animal no banco de dados. |
| Fluxo alternativo V – Exclusão de animal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar seu perfil | |
| | 2. Redirecionar o usuário para o próprio perfil |
| 3. Acessar o ícone de opções próxima ao animal que deseja excluir e selecionar a opção “excluir”. | |
| | 4. Exibir uma tela de confirmação de desejo de exclusão |
| 5. Selecionar a opção excluir | |
| | 6. Exibir que o animal foi excluído com sucesso. |

Tabela 2 – Documentação de Caso (Visualizar animal)

| Nome do Caso de Uso | Visualizar animal |
|----------------------------|---|
| Caso de Uso Geral | Manter animais. |
| Ator Principal | Usuário não cadastrado |
| Atores Secundários | Doador/adotante, ONG's e Administrador. |
| Resumo | Esse caso de uso permite a visualização de qualquer animal. |
| Pré-Condições | |
| Pós-Condições | |
| Regras de negócio | 1. As informações só aparecerão completa para visualização se o usuário estiver logado. |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |

| | |
|--|--|
| 1. Acessar aba de adoção | |
| | 2. Redirecionar para aba de animais cadastrados. |
| 3. Selecionar animal que tem interesse de visualizar | |
| | 4. Exibir página de exibição do animal. |
| Restrições/confirmações | 1. Somente usuários cadastrados poderão visualizar as informações de contato dos doadores. |
| Fluxo alternativo – Visualização de animais vinculados a sua conta | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar o seu perfil. | |
| | 2. Redirecionar o usuário para o próprio perfil |
| 3. Selecionar o animal que deseja visualizar na aba de animais doados. | |
| | 4. Abrir a página de visualização do animal. |
| Restrições/confirmações | 1. Fluxo exclusivo para usuários cadastrados |

Tabela 3 – Documentação de Caso (Registrar doação de animais)

| Nome do Caso de Uso | Registrar doação de animais |
|----------------------------|---|
| Caso de Uso Geral | |
| Ator Principal | Usuário não cadastrado |
| Atores Secundários | Doador/ adotante, ONG's e Administrador. |
| Resumo | Esse caso de uso permite a visualização de qualquer animal. |
| Pré-Condições | |
| Pós-Condições | |

| | |
|--|--|
| Regras de negócio | 2. As informações só aparecerão completa para visualização se o usuário estiver logado. |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 5. Acessar aba de adoção | |
| | 6. Redirecionar para aba de animais cadastrados. |
| 7. Selecionar animal que tem interesse de visualizar | |
| | 8. Exibir página de exibição do animal. |
| Restrições/confirmações | 2. Somente usuários cadastrados poderão visualizar as informações de contato dos doadores. |
| Fluxo alternativo – Visualização de animais vinculados a sua conta | |
| Ações do Ator | Ações do Sistema |
| 5. Acessar o seu perfil. | |
| | 6. Redirecionar o usuário para o próprio perfil |
| 7. Selecionar o animal que deseja visualizar na aba de animais doados. | |
| | 8. Abrir a página de visualização do animal. |
| Restrições/confirmações | 2. Fluxo exclusivo para usuários cadastrados |

Tabela 4 – Documentação de Caso (Manter usuários)

| Nome do Caso de Uso | Manter usuários |
|---------------------|------------------------|
| Caso de Uso Geral | |
| Ator Principal | Administrador |
| Atores Secundários | Doador/adotante e ONGs |

| | |
|---|--|
| Resumo | Este caso de uso determina a criação, leitura, atualização e exclusão dos Adotantes/Doadores e ONGs |
| Pré-Condições | 1. Estar logado. |
| Pós-Condições | |
| Permissões | <ol style="list-style-type: none"> 1. O Adotante/Doador e ONGs terá a permissão de editar e excluir o próprio perfil cadastrado. 2. O Administrador poderá Adicionar, Editar e Excluir Adotantes/Doadores e ONGs. |
| Fluxo Alternativo I- Cadastrar Adotante/Doador | |
| Ações do Ator | Ações do Sistema |
| 1. Acessa a página de Cadastro. | |
| | <ol style="list-style-type: none"> 2. Exibe o formulário para o usuário preencher os seguintes dados: <ul style="list-style-type: none"> • Nome do Adotante/Doador; • E-mail; • Data de Nascimento; • CPF; • Endereço; • Telefone1; • Telefone2; • Sobre você; |
| 3. Adotante/Doador irá preencher os dados. | |
| | 4. Sistema irá cadastrar o Adotante/Doador no sistema. |
| Fluxo Alternativo II- Visualizar Usuário | |
| Ações do Ator | Ações do Sistema |
| 1. O usuário irá acessar a página de Adoção. | |
| | 2. O sistema levará o usuário para a página de animais em adoção. |
| 3. Usuário visualizará o animal e ver o perfil do usuário logo abaixo das especificações do animal. | |

| | |
|---|--|
| | 4. Redirecionará para a página do perfil do Doador para a visualização. |
| Fluxo Alternativo III – Cadastrar ONG | |
| Ações do Ator | Ações do Sistema |
| 1. Acessa a página de Cadastro. | |
| | 2. Exibe o formulário para o usuário preencher os seguintes dados: <ul style="list-style-type: none"> • Nome; • Endereço; • E-mail; • Telefone; • Descrição; • Estatuto Social; |
| 3. A Organização não Governamental irá preencher os campos que foram apresentados. | |
| | 4. A ONG será cadastrada no sistema. |
| Fluxo Alternativo IV - Edição dos dados Doador/adotante | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar o próprio perfil. | |
| | 2. Redirecionar o Adotante/Doador para o próprio perfil. |
| 3. O Adotante/Doador irá acessar o ícone de Edição de perfil para alterar a informações que deseja. | |
| | 4. Exibe o formulário que requisita os seguintes dados: <ul style="list-style-type: none"> • Nome do Adotante/Doador; • E-mail; • Data de Nascimento; • CPF; • Endereço; • Telefone1; • Telefone2; • Sobre você; |
| 5. Editar os Campos desejados. | |

| | |
|--|--|
| | 6. Salvar as alterações que o Doador/Adotante alterou. |
| Fluxo Alternativo V - Edição dos dados de ONGs | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar o próprio perfil | |
| | 2. Redirecionar a ONG para o próprio perfil. |
| 3. A ONG irá acessar o ícone de Edição de perfil para alterar a informações que deseja | |
| | 4. Exibe o formulário que requisita os seguintes dados: <ul style="list-style-type: none"> • Nome; • Endereço; • E-mail; • Telefone; • Descrição; • Estatuto Social; |
| 5. A ONG irá editar os Campos desejados. | |
| | 6. Salvar as alterações que a ONG alterou. |
| Fluxo alternativo VI - Exclusão do usuário (doador/adotante e ONG) | |
| 1. Acessar o próprio perfil. | |
| | 2. Redirecionar o usuário para o próprio perfil. |
| 3. O usuário irá acessar o ícone de excluir o perfil para apagar a conta. | |
| | 4. Exibir um aviso de que a conta será excluída permanentemente |
| 5. Confirmar a exclusão | |
| | 6. Excluir a conta do banco de dados. |

Tabela 5 – Documentação de Caso (Visualizar usuários)

| | |
|----------------------------|----------------------------|
| Nome do Caso de Uso | Visualizar usuários |
|----------------------------|----------------------------|

| | |
|--|---|
| Caso de Uso Geral | Manter usuários |
| Ator Principal | Doador/Adotante |
| Atores Secundários | Administrador e ONG's |
| Resumo | Esse caso de uso permite a visualização dos usuários. |
| Pré-Condições | 1. Estar logado. |
| Pós-Condições | |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar página principal | |
| | 2. Redirecionar o Doador/Adotante para a página principal. |
| 3. O Doador/Adotante irá selecionar no animal que deseja visualizar. | |
| | 4. O site irá ampliar o animal mostrando seus dados respectivamente, o ícone do responsável irá ficar logo após o animal. |
| 5. Usuário irá clicar no perfil do responsável. | |
| | 6. O perfil do responsável ficará disponível para visualização. |

Tabela 6 – Documentação de Caso (Cadastrar doador/adotante)

| Nome do Caso de Uso | Cadastrar doador/adotante |
|----------------------------|---|
| Caso de Uso Geral | Manter usuários |
| Ator Principal | Doador/adotante |
| Atores Secundários | |
| Resumo | Esse caso de uso permite o cadastro dos doadores/adotantes. |
| Pré-Condições | |
| Pós-Condições | |

| Fluxo Principal | |
|--|---|
| Ações do Ator | Ações do Sistema |
| 1. Acessa a página de Cadastro. | |
| | 2. Exibe o formulário para o usuário preencher os seguintes dados: <ul style="list-style-type: none"> • Nome do Adotante/Doador; • E-mail; • Data de Nascimento; • CPF; • Endereço; • Telefone1; • Telefone2; • Sobre você; |
| 3. Adotante/Doador irá preencher os dados. | |
| | 4. Sistema irá cadastrar o Adotante/Doador no sistema. |

Tabela 7 – Documentação de Caso (Acessar Perfil)

| Nome do Caso de Uso | Acessar Perfil |
|--|--|
| Caso de Uso Geral | |
| Ator Principal | Doador/adotante |
| Atores Secundários | ONG, administrador |
| Resumo | Esse caso de uso permite que o ator visualize seu perfil juntamente com os dados nele presente |
| Pré-Condições | 1. Estar logado |
| Pós-Condições | |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. O ator seleciona a aba em que há sua foto e seu nome. | |

| | |
|---|--|
| | <p>2. O sistema redireciona para a página do perfil do usuário. Nesta página é possível visualizar os seguintes dados do ator:</p> <ul style="list-style-type: none"> • Nome do Adotante/Doador; • E-mail; • Data de Nascimento; • CPF; • Endereço; • Telefone1; • Telefone2; • Sobre você; <p>Haverá nessa página opções de edição dos dados, exclusão da conta e os animais que usuário colocou para doação.</p> |
| Fluxo Alternativo – Acessar o perfil de outros usuários | |
| Ações do Ator | Ações do Sistema |
| <p>1. O usuário seleciona um animal de sua preferência em uma das duas páginas disponíveis que há animais:</p> <ul style="list-style-type: none"> • Home; • Adotar; | |
| | <p>2. Redireciona para a página de um animal, que apresentará algumas informações do animal e o perfil do doador.</p> |
| <p>3. O usuário seleciona o perfil do doador.</p> | |
| | <p>4. O site redireciona o usuário para o perfil do doador, que haverá informações de contato e a opção de report do perfil.</p> |

Tabela 8 – Documentação de Caso (Reportar Perfil)

| Nome do Caso de Uso | Reportar Perfil |
|---------------------|-----------------|
| Caso de Uso Geral | |

| | |
|--|--|
| Ator Principal | Doador/adoptante |
| Atores Secundários | ONG, administrador |
| Resumo | Esse caso de uso permite que o ator reporte perfis de outros usuários. |
| Pré-Condições | 1. Estar logado |
| Pós-Condições | |
| Regra de negócio | 1. Caso o usuário receba três <i>reports</i> , sua conta será bloqueada |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. Visitar o perfil que deseja reportar. | |
| | 2. Redireciona para a página do perfil do usuário selecionado. |
| 3. O Ator reporta seleciona o ícone de report. | |
| | 4. Será apresentado caixas de seleção de possíveis motivos para o report e logo abaixo um campo de texto para o usuário descrever melhor o report. |
| 5. Preencher o formulário e enviar o report. | |
| | 6. Adicionar um report à conta reportada. |

Tabela 9 – Documentação de Caso (Manter eventos de doação)

| Nome do Caso de Uso | Manter eventos de doação |
|---------------------|---|
| Caso de Uso Geral | |
| Ator Principal | ONG |
| Atores Secundários | Administrador |
| Resumo | Esse caso de uso permite que o ator crie, edite, visualize e exclua eventos que serão divulgados no site. |

| | |
|--|---|
| Pré-Condições | 1. Estar logado |
| Pós-Condições | 2. Ser validado pelo administrador. |
| Fluxo Alternativo I – Cadastrar evento | |
| Ações do Ator | Ações do Sistema |
| 1. O ator selecionará a aba de eventos em sua barra de navegação, logo após selecionar a opção “cadastrar evento”. | |
| | 2. Redirecionará para uma página com um formulário a ser preenchido com as seguintes informações: <ul style="list-style-type: none"> • Nome do evento; • Local; • Data; • Horário; • Contato; • Organizador; • Imagem; • Informações extra; |
| 3. O ator preenche o formulário e confirma o envio para o administrador. | |
| Restrições/permissões | 1. O evento precisará ser confirmado por um administrador. |
| Fluxo Alternativo II – Visualizar evento | |
| Ações do Ator | Ações do Sistema |
| 1. O ator selecionará a aba de eventos em sua barra de navegação, logo após selecionar a opção “buscar evento”. | |
| | 2. O site redirecionará o usuário para a página de exibição dos eventos, havendo opções de filtragem deles. |
| 3. Selecionar o evento de interesse | |
| | 4. Exibir informações extras do evento. |
| Fluxo Alternativo III – Editar evento | |
| Ações do Ator | Ações do Sistema |

| | |
|--|---|
| 1. O ator selecionará a aba de eventos em sua barra de navegação, logo após selecionar a opção “meus eventos”. | |
| | 2. Redirecionar para a página que exibe os eventos criados pela ONG logada. |
| 3. Selecionar o evento que deseja editar as informações | |
| | 4. Expandir as informações do evento selecionado, disponibilizando os dados: <ul style="list-style-type: none"> • Nome do evento; • Local; • Data; • Horário; • Contato; • Organizador; • Imagem; • Informações extra; E as opções de edição e deleção do evento. |
| 5. Selecionar a opção de edição das informações do evento. | |
| | 6. Disponibilizar um formulário de edição dos dados. |
| 7. Preencher os dados editados e confirmar o envio. | |
| Fluxo Alternativo IV – Excluir evento | |
| Ações do Ator | Ações do Sistema |
| 1. O ator selecionará a aba de eventos em sua barra de navegação, logo após selecionar a opção “meus eventos”. | |
| | 2. Redirecionar para a página que exibe os eventos criados pela ONG logada. |

| | |
|---|---|
| 3. Selecionar o evento que deseja editar as informações | |
| | <p>4. Expandir as informações do evento selecionado, disponibilizando os dados:</p> <ul style="list-style-type: none"> • Nome do evento; • Local; • Data; • Horário; • Contato; • Organizador; • Imagem; • Informações extra; <p>E as opções de edição e deleção do evento.</p> |
| 5. Selecionar a opção de deleção do evento. | |
| | 6. Exibir um aviso de deleção. |
| 7. Confirmar a deleção do evento. | |
| | 8. Excluir o evento. |

Tabela 10 – Documentação de Caso (Validar eventos de doação)

| Nome do Caso de Uso | Validar eventos de doação |
|----------------------------|---|
| Caso de Uso Geral | Manter eventos de doação |
| Ator Principal | Administrador |
| Atores Secundários | ONG |
| Resumo | Esse caso de uso permite que o ator principal autorize a publicação do evento no site. |
| Pré-Condições | <ol style="list-style-type: none"> 1. Estar logado 2. Ter recebido uma solicitação de criação de evento |

| | |
|---|--|
| Pós-Condições | |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. O administrador entrará na aba eventos, e logo após selecionará a opção “eventos a validar”. | |
| | 2. Redireciona o administrador para a aba dos eventos, onde serão apresentados os eventos ainda não validados. |
| 3. Usuário seleciona o evento a ser visualizar. | |
| | 4. Informações exibidas em formato de tabela: <ul style="list-style-type: none"> • Nome do evento; • Local; • Data; • Horário; • Contato; • Organizador; • Imagem; • Informações extra; Apresenta botões de validar ou recusar o evento. |
| 5. Aceita ou recusa evento com os botões próprios. | |

Tabela 11 – Documentação de Caso (Fazer login)

| Nome do Caso de Uso | Fazer Login |
|---------------------|---|
| Caso de Uso Geral | Manter usuários |
| Ator Principal | Doador/adotante, ONG, administrador |
| Atores Secundários | |
| Resumo | Esse caso de uso permite ao ator fazer login em sua conta no site |

| | |
|---|---|
| Pré-Condições | 1. Ter se cadastrado no PET's Portal |
| Pós-Condições | |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar a página PET's Portal. | |
| | 2. Redireciona o usuário para a página principal do site. |
| 3. Clica em Fazer login | |
| | 4. Exibirá os campos para validar o Login (E-mail e Senha). |
| 5. Preenche os campos exigidos (E-mail e Senha.) | |
| | 6. Caso o Login esteja correto o sistema redirecionará para a página Home com o Login executado. |
| Fluxo Alternativo I- Cadastro de usuário não cadastrado (Doador/Adotante) | |
| Ações do Ator | Ações do Sistema |
| 1. Clica em fazer cadastro | |
| | 2. Exibirá a tela de cadastro com os seguintes campos: <ul style="list-style-type: none"> • Nome do Adotante/Doador; • E-mail; • Data de Nascimento; • CPF; • Endereço; • Telefone1; • Telefone2; • Sobre você; |
| 3. Preenche os campos solicitados pelo sistema. | |
| | 4. Cadastra Doador/Adotante |
| 5. Acessa a página de Login | |
| Fluxo Alternativo II- Cadastro de usuário não cadastrado (ONGs) | |
| 1. Clica em fazer cadastro | |

| | |
|---|--|
| | <p>2. Exibirá a tela de cadastro com os seguintes campos:</p> <ul style="list-style-type: none"> • Nome; • Endereço; • E-mail; • Telefone; • Descrição; • Estatuto Social; |
| 3. Preenche os campos solicitados pelo sistema. | |
| | 4. Cadastra Doador/Adotante |
| 5. Acessa a página de Login | |

Tabela 12 – Documentação de Caso (Adotar animal)

| Nome do Caso de Uso | Adotar Animal |
|-----------------------------------|---|
| Caso de Uso Geral | Manter Animal |
| Ator Principal | Doador/adotante, ONG, administrador |
| Atores Secundários | |
| Resumo | Esse caso de uso permite que o Doador/Adotante valide ou envie, respectivamente, um pedido de adoção no seu perfil. |
| Pré-Condições | Ter se cadastrado no PET's |
| Pós-Condições | |
| Fluxo Principal | |
| Ações do Ator | Ações do Sistema |
| 1. Acessar a página PET's Portal. | |
| | 2. Redireciona o Adotante/Doador para a página principal |
| 3. Clica em Fazer login. | |
| | 4. Exibirá os campos para validar o Login (E-mail e Senha). |

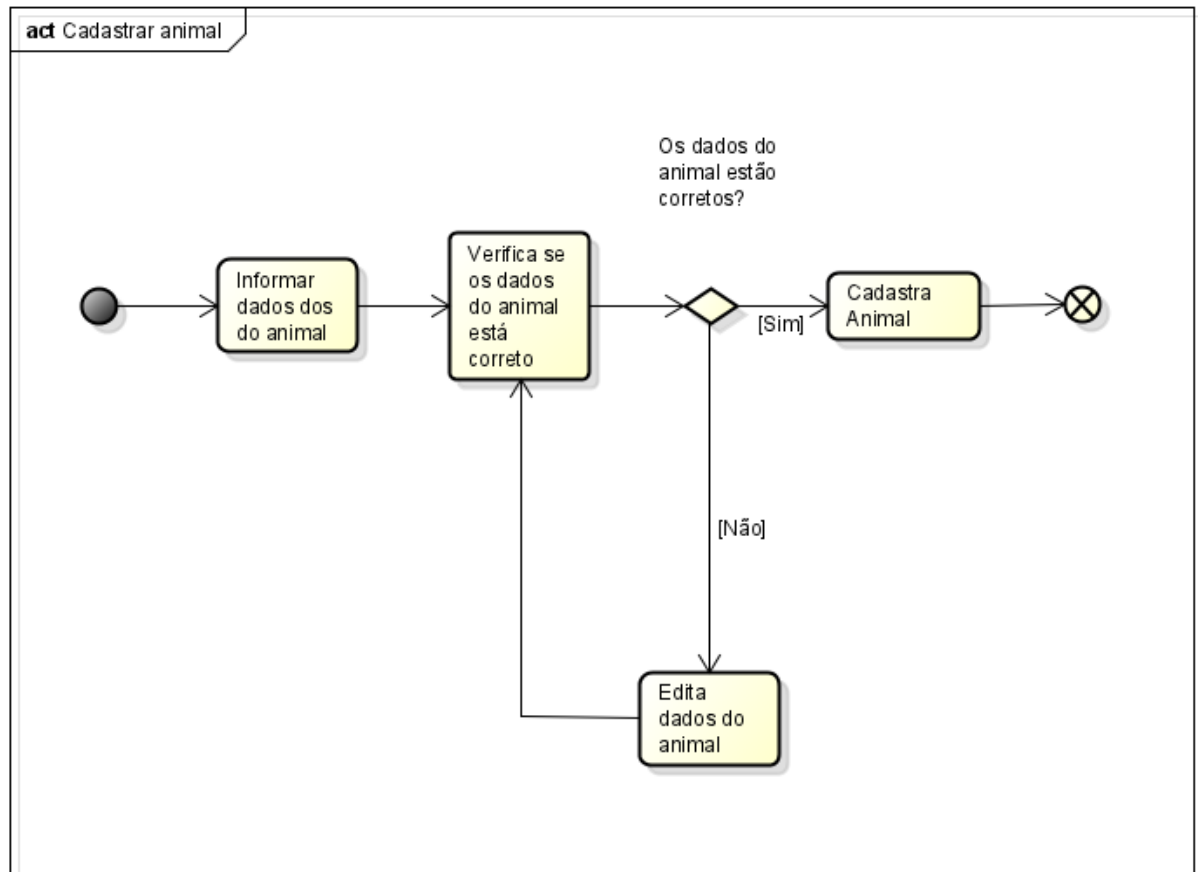
| | |
|---|---|
| 5. Preenche os campos exigidos (E-mail e Senha.) | |
| | 6. Apresentará os pedidos de Adoção que adotantes realizaram. |
| 7. O Adotante/Doador irá aceitar ou recusar o pedido de adoção do Adotante. | |
| | 8. Caso o Doador aceite o pedido de adoção o sistema exibirá o telefone do Doador em questão. |

3.3. Diagrama de Atividade

Os diagramas foram divididos em três, que ilustram as atividades cadastrar animal, solicitar adoção e registrar evento de doação.

A figura 13 descreve, através das atividades, o fluxo para doar um animal, com a inserção de dados e a validação para autenticar o usuário.

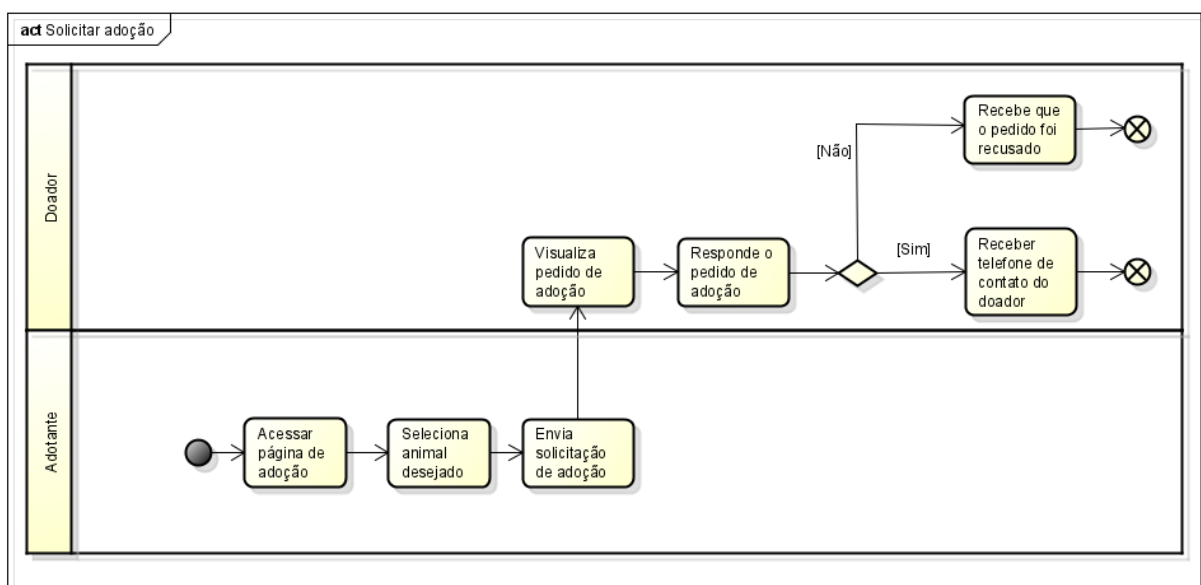
Figura 14 – Diagrama de atividade (Cadastrar animal)



Fonte: Autoria Própria, 2021.

A figura 15 é o diagrama de atividades para solicitar doação de animais, ilustrando seu fluxo de atividades.

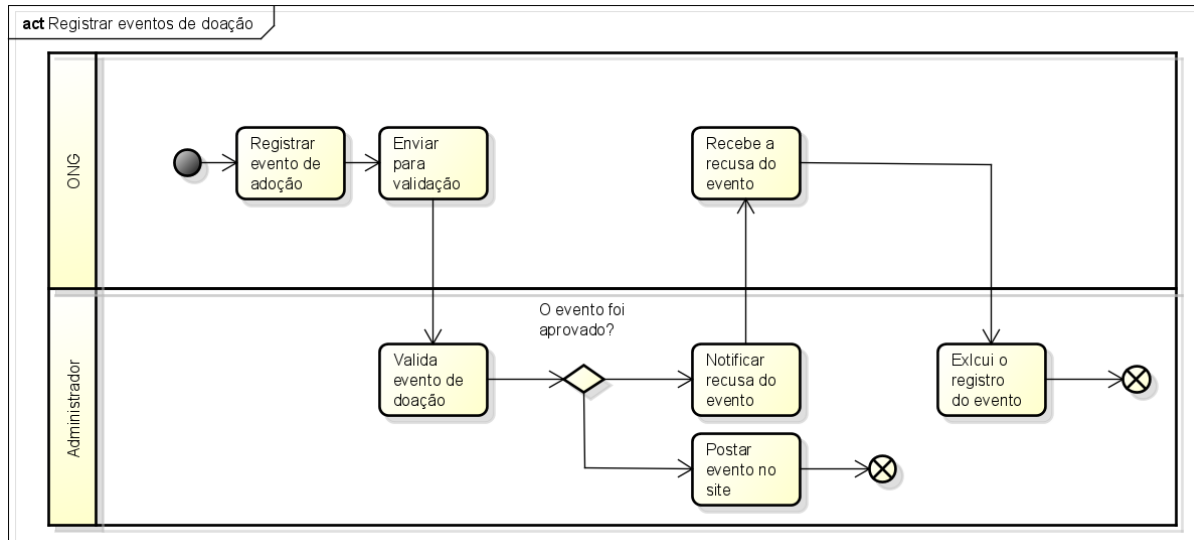
Figura 15 – Diagrama de atividade (Solicitar adoção)



Fonte: Autoria Própria, 2021.

Na figura 16, o diagrama de atividades registrar eventos de doação exhibe o processo de requisição de divulgação de um evento para um administrador.

Figura 16 – Diagrama de atividade (Registrar eventos de doação)



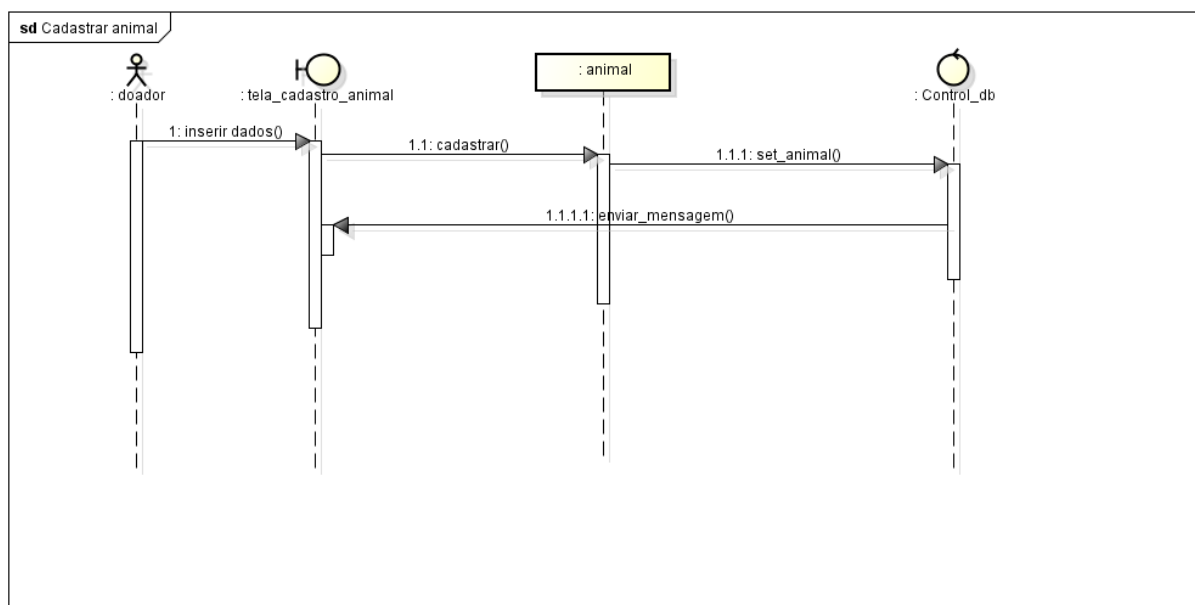
Fonte: Autoria Própria, 2021.

3.4. Diagrama de Sequência

Os diagramas de sequência compõem no total cinco diagramas, os quais são cadastrar animal, logar, cadastrar evento, cadastrar usuário e doar animal.

Na figura 17, o diagrama descreve a sequência da realização do cadastro do animal no sistema, ilustrando como esse processo ocorre desde a tela de cadastro, passando pela classe animal até chegar ao banco.

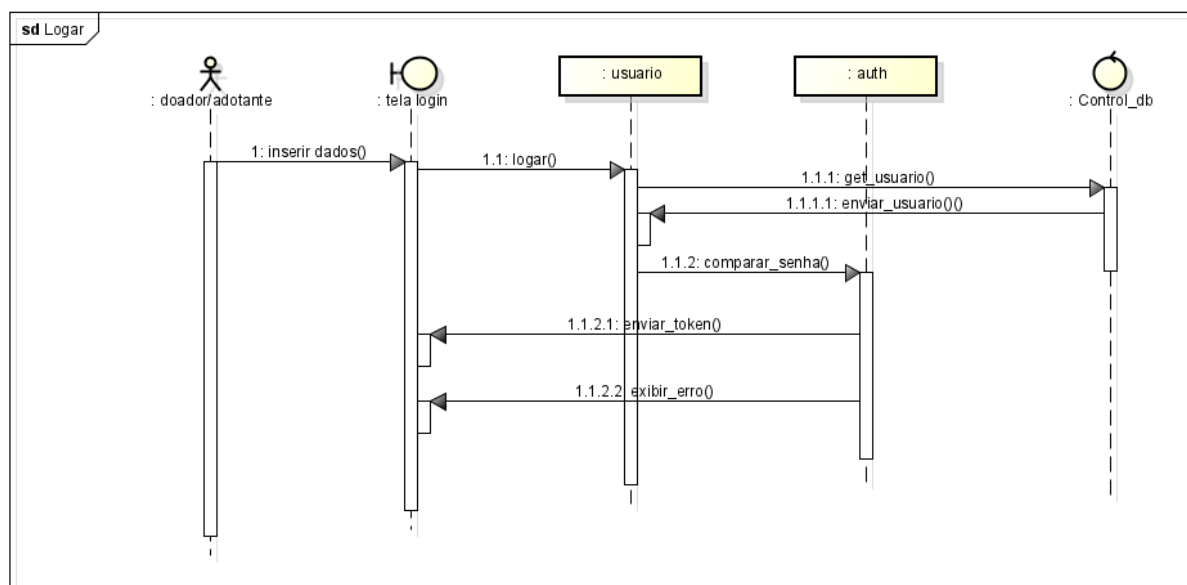
Figura 17 – Diagrama de sequência (Cadastrar animal)



Fonte: Autoria Própria, 2021.

Na figura 18, o diagrama ilustra o processo de login dos usuários, desde a tela de login, passando pelas classes de usuário e de autenticação, até chegar no banco de forma encriptada.

Figura 18 – Diagrama de sequência (Logar)

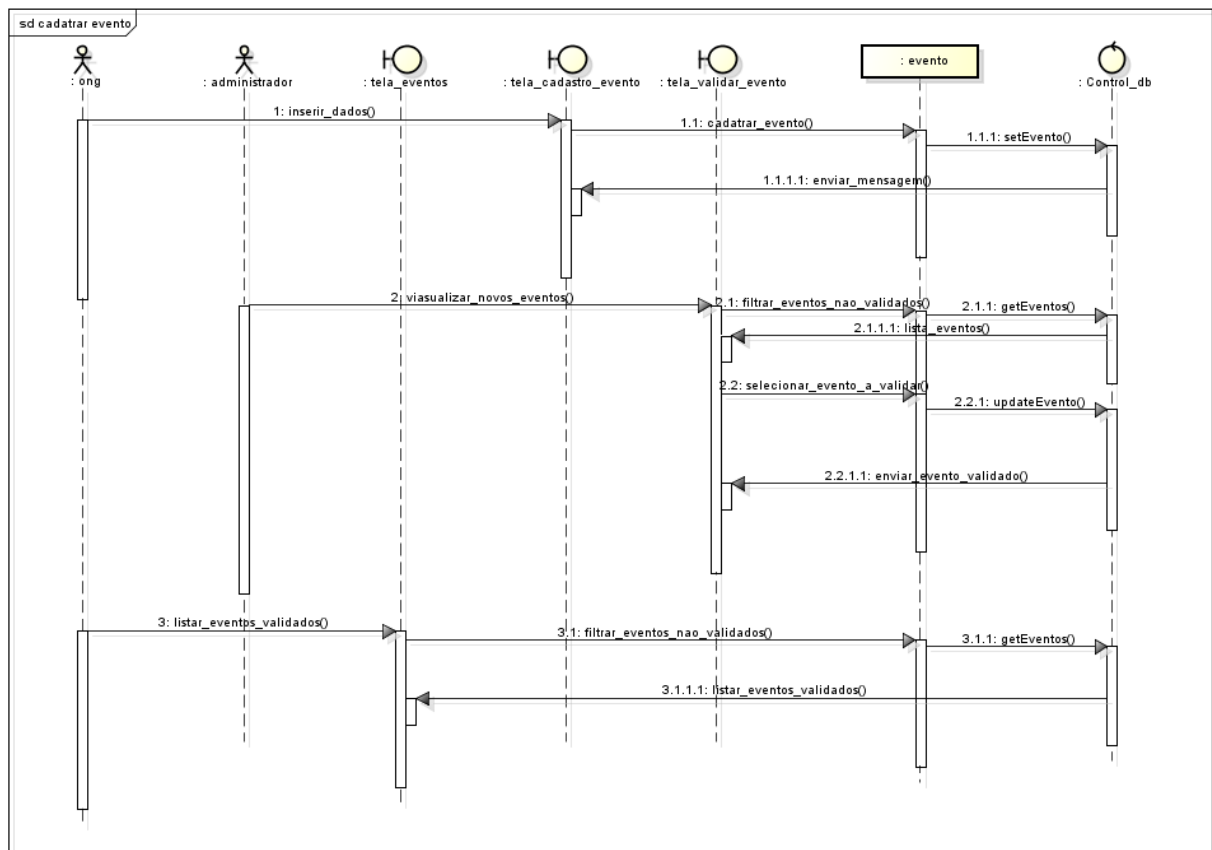


Fonte: Autoria Própria, 2021.

Na figura 19, a sequência de cadastro de eventos. O processo é dividido em três fases, o cadastro do evento pela ONG, que passa pela classe evento e é registrada no banco, mas ainda como um evento inválido. A segunda fase é da aprovação evento pelo administrador, que acessa uma página exclusiva para seu uso que

busca no banco de dados todos os eventos que não foram validados ainda, em seguida o administrador muda o status do evento para válido. Por fim, a última fase é a visualização do evento que já foi validado, que passa pela interface de visualização que busca tais eventos no banco.

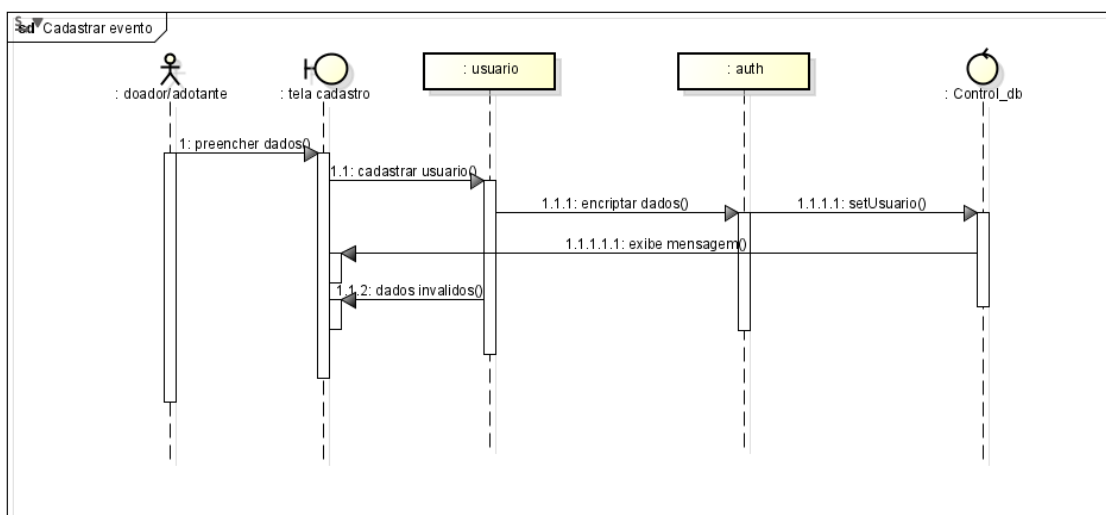
Figura 19 – Diagrama de sequência (Cadastrar evento)



Fonte: Autoria Própria, 2021.

Na figura 20, o diagrama descreve a sequência da realização do cadastro de usuários no site, ilustrando como esse processo ocorre da interface, passando pelas classes ao banco.

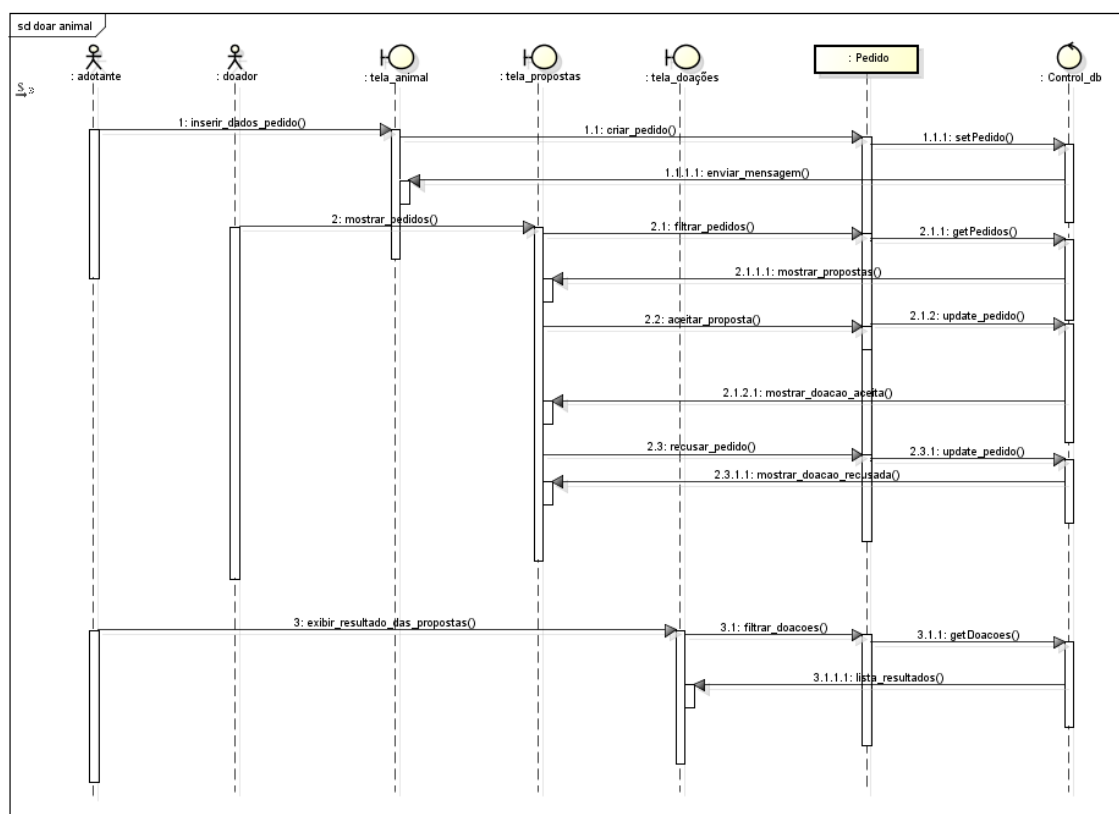
Figura 20 – Diagrama de sequência (Cadastrar usuário)



Fonte: Autoria Própria, 2021.

Na figura 21, o diagrama ilustra o processo de doação de um animal, sendo dividido nas fases de envio da proposta de adoção, análise e devolutiva da proposta pelo dono do animal e visualização da resposta da proposta, passando por três diferentes interfaces, uma classe e pelo banco.

Figura 21– Diagrama de sequência (Doar animal)



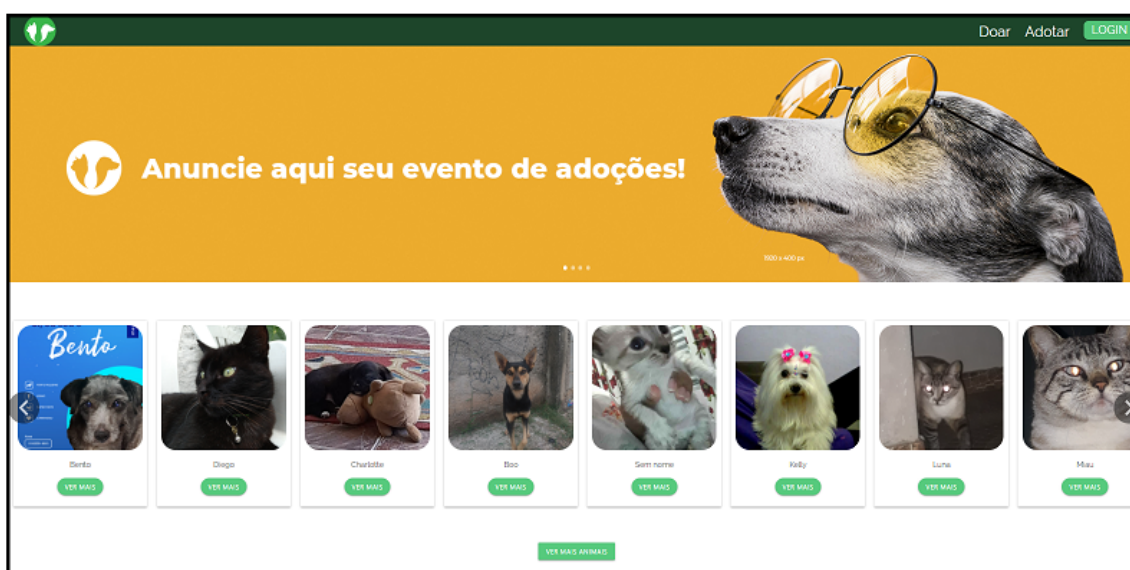
Fonte: Autoria Própria, 2021.

3.5. Aplicação WEB

O aplicativo é composto por um total de 20 telas, que possui diversas adaptações de acordo com o tipo de usuário.

A tela home, figura 22, é a tela principal do site, nela é exibida uma barra de navegação, um carrossel com eventos de doação de animais, e animais disponíveis para adoção.

Figura 22 – Tela Home



Fonte: Autoria Própria, 2021.

A barra de navegação é o principal fator de mudança entre os diferentes tipos de usuários. Ela se adapta a usuários não logados, conforme figura 23, doador/ adotante, figura 24, ONG's, figura 25, e ao usuário administrador, figura 26.

Figura 23 – Barra de navegação (usuários não logados)



Fonte: Autoria Própria, 2021.

Figura 24 – Barra de navegação (doador/adotante)



Fonte: Autoria Própria, 2021.

Figura 25 – Barra de navegação (ONGs)



Fonte: Autoria Própria, 2021.

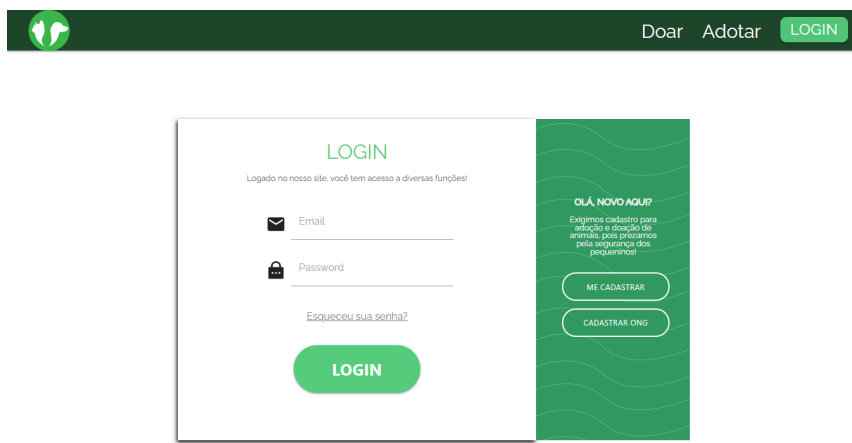
Figura 26 – Barra de navegação (Administrador)



Fonte: Autoria Própria, 2021.

Para logar-se, o usuário acessará a tela login, figura 27, que pede seu e-mail e senha como forma de verificação.

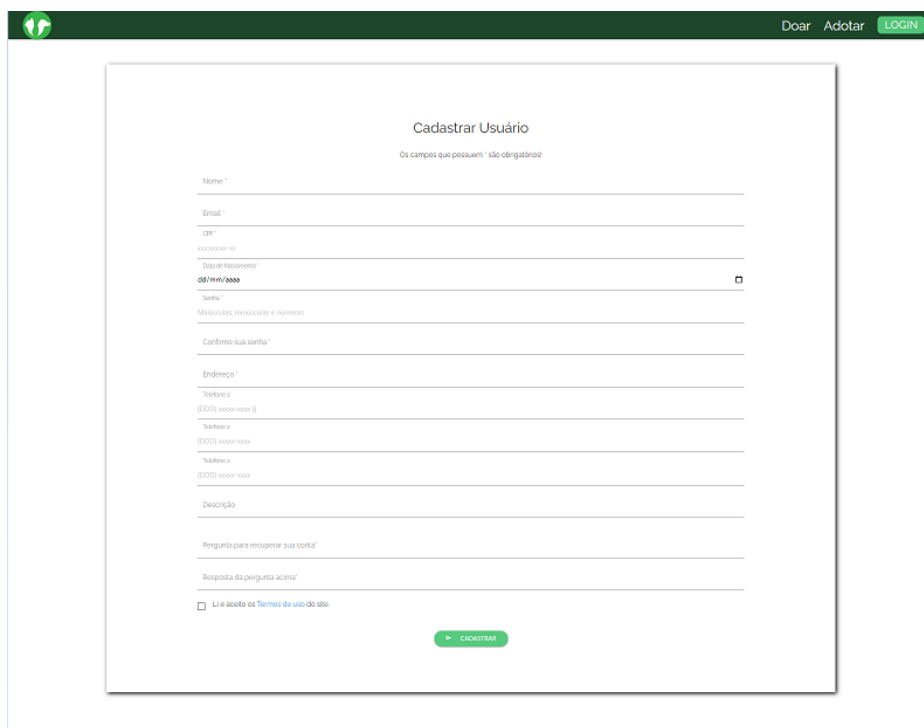
Figura 27 – Tela Login



Fonte: Autoria Própria, 2021.

Caso o usuário não possua cadastro, ele poderá acessar a tela cadastro doador/ adotante, figura 28, caso seja um usuário comum.

Figura 28 – Tela Cadastro Doador/adotante



Fonte: Autoria Própria, 2021.

No caso do usuário que estiver se cadastrando seja uma ONG, se inscreverá através da tela cadastro ONG, figura 29.

Figura 29 – Tela Cadastro ONG

Cadastrar ONG

Os campos que possuem * são obrigatórios

Nome *

E-mail *

Senha *

Másculas, minúsculas e números

Confirme sua senha *

Endereço *

Número *

(DDD) xxxxx-xxxx

Telefone *

(DDD) xxxxx-xxxx

Descrição

Pergunta para recuperar sua conta *

Resposta da pergunta acima *

Li e aceito os [Termos de uso do site](#).

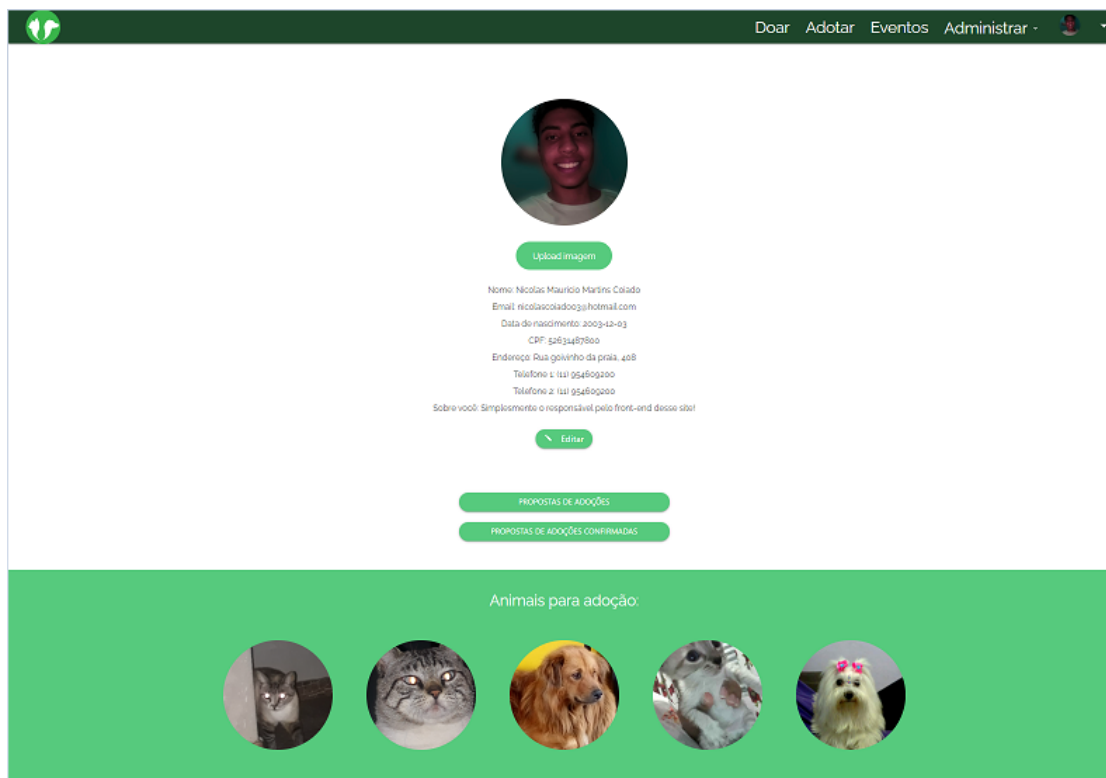
Estatuto Social [Arquivo](#)

[CADASTRAR ONG](#)

Fonte: Autoria Própria, 2021.

Após executar o login, o usuário será redirecionado para a tela perfil, figura 30, e poderá revisar suas informações e adicionar uma foto para seu perfil.

Figura 30 – Tela Perfil



Fonte: Autoria Própria, 2021.

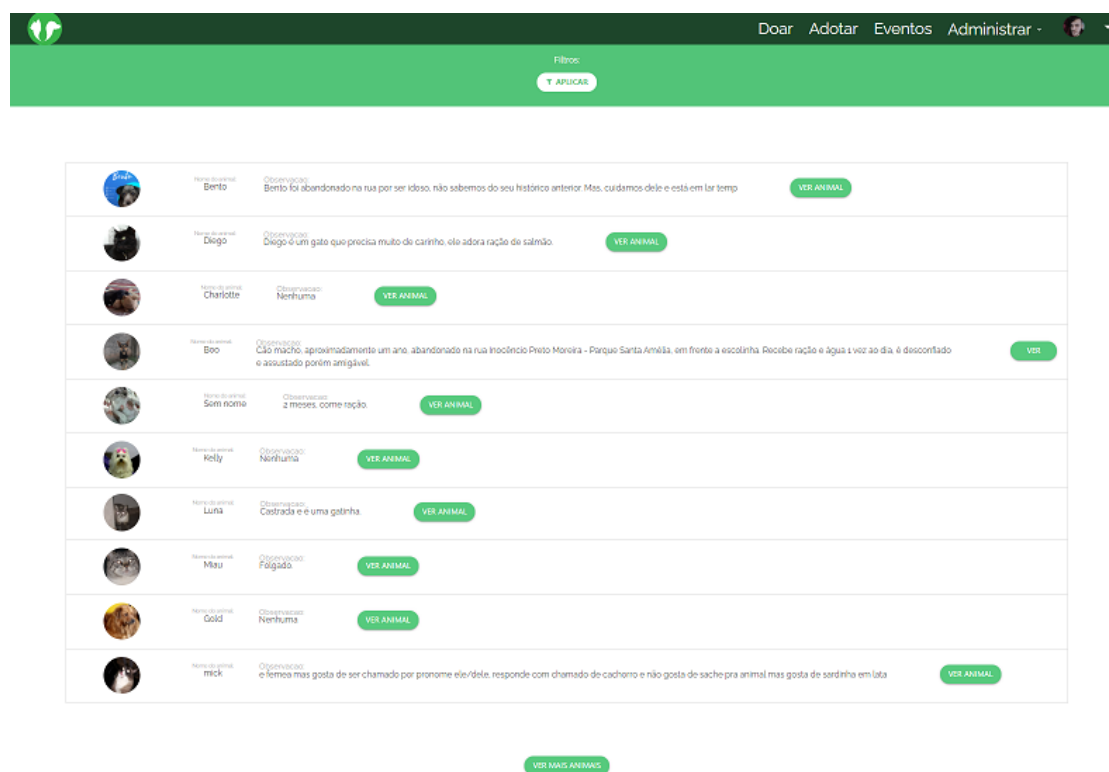
Caso queira registrar seu animal no site para doação, o usuário acessará a tela doação, figura 31, que possuirá um formulário para cadastro do seu animal.

Figura 31 – Tela Doação

Fonte: Autoria Própria, 2021.

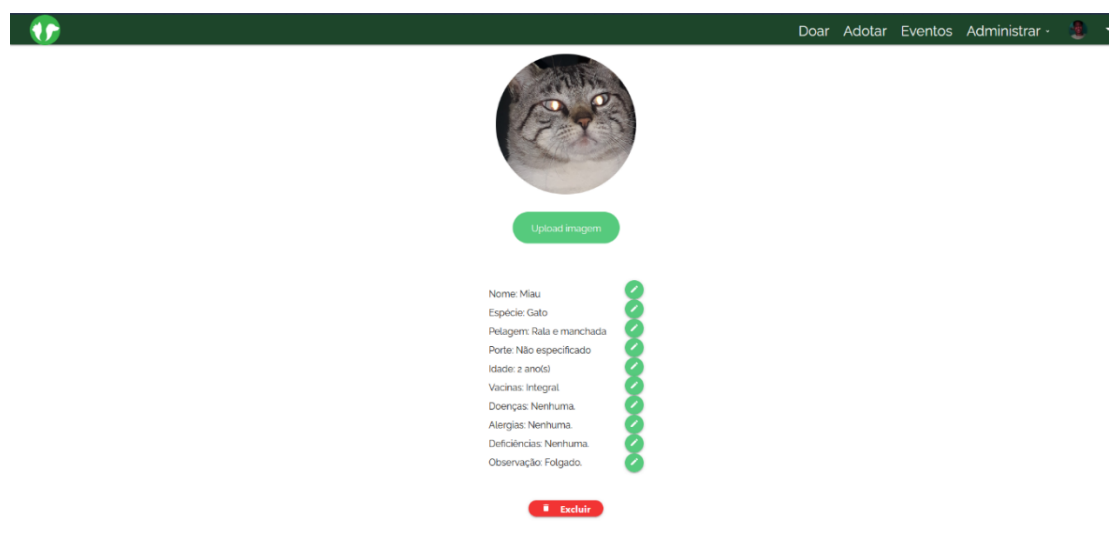
Já se for uma pessoa interessada em adotar um animal, acessará a tela adoção, imagem 32, que listará os animais cadastrados no site. Ao selecionar o animal, será redirecionado a tela animal, que possui duas vertentes, caso o usuário acessando seja o responsável pelo animal, figura 33, ou se for qualquer outro usuário, figura 34.

Figura 32 – Tela Adoção



Fonte: Autoria Própria, 2021.

Figura 33 – Tela Animal (Visão dono)



Fonte: Autoria Própria, 2021.

Figura 34 – Tela Animal (Visão adotante)

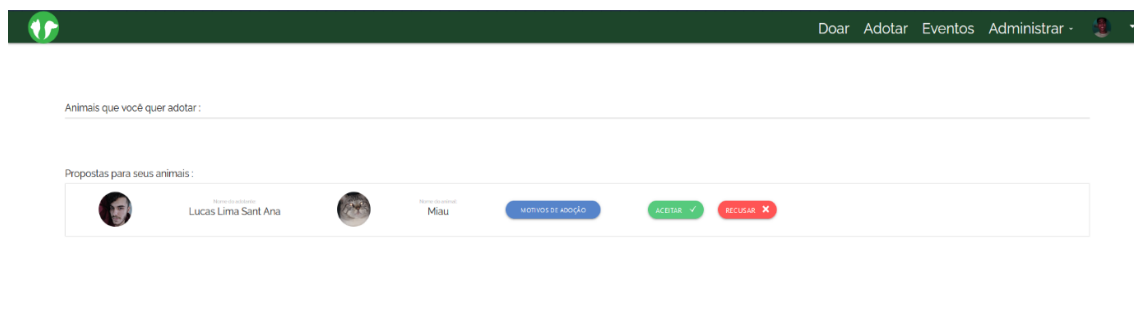
Fonte: Autoria Própria, 2021.

Na tela perfil, figura 30, quando acessada pelo próprio usuário, é possível acessar suas propostas de adoção, figura 35, que exibirá todas suas propostas em andamento e aceitas. É possível acessar também a tela pedidos de adoção, figura 36, que exibirá todos os pedidos que foram recebidos pelos animais cadastrados pelo usuário.

Figura 35 – Tela Propostas de Adoção

Fonte: Autoria Própria, 2021.

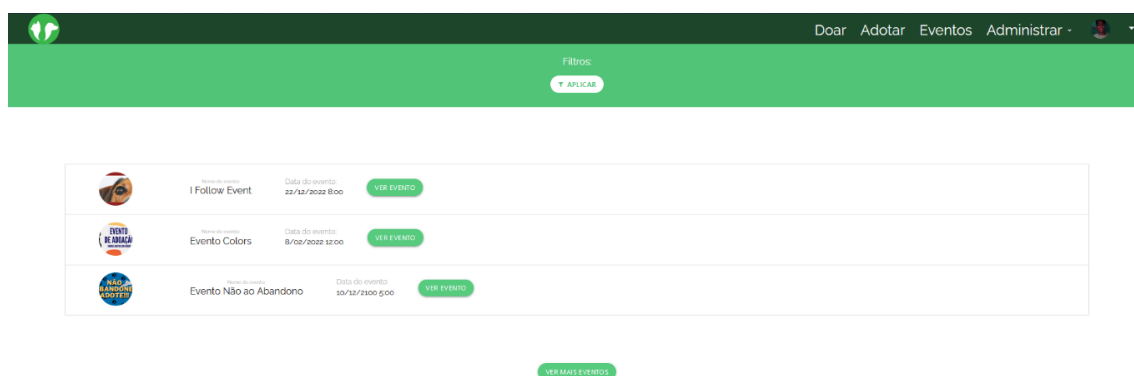
Figura 36 – Tela Propostas de Adoção Aceitas



Fonte: Autoria Própria, 2021.

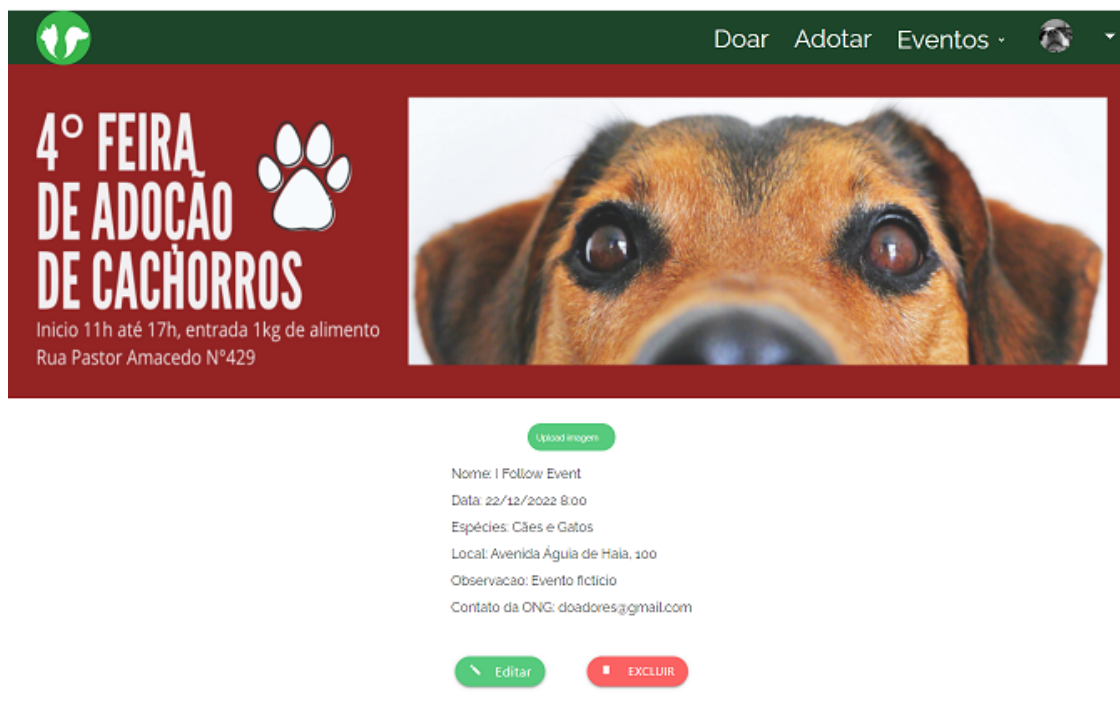
O site divulgará eventos de doação de animais, ao acessar a tela lista eventos, figura 37, será listado todos os eventos cadastrados no site, essa é uma tela disponível somente para usuários cadastrados, que poderão acessar a tela evento, que possui visões diferentes para a ONG que cadastrou o evento, figura 38, e para o doador/adotante, figura 39.

Figura 37 – Tela Lista Eventos



Fonte: Autoria Própria, 2021.

Figura 38 – Tela Evento (visão ONG responsável)



Fonte: Autoria Própria, 2021.

Figura 39 – Tela Evento (visão doador/adotante)



Fonte: Autoria Própria, 2021.

Esses eventos serão registrados pelas ONG's na tela cadastrar evento, figura 40, que exibirá um formulário com as informações necessárias para o registro do evento. As ONG's só conseguirão logar quando tiverem suas contas validadas pelos administradores na tela validar ONG, figura 41.

Figura 40 – Tela Cadastrar Evento

The screenshot shows a web interface for registering an event. At the top, there is a dark green navigation bar with a logo on the left and links for 'Doar', 'Adotar', and 'Eventos' on the right. The main content area is a white box titled 'Cadastrar Evento'. It contains several input fields: 'Nome', 'Local do evento', 'Data e Horário do Evento' (with a calendar icon), 'Observações', and 'Geral'. Below these fields, a note states: 'A arte de divulgação deve possuir as dimensões de 1520px (largura) X 400px (altura)'. At the bottom of the form, there are two green buttons: 'Banner de divulgação' and 'CADASTRAR EVENTO'.

Fonte: Autoria Própria, 2021.

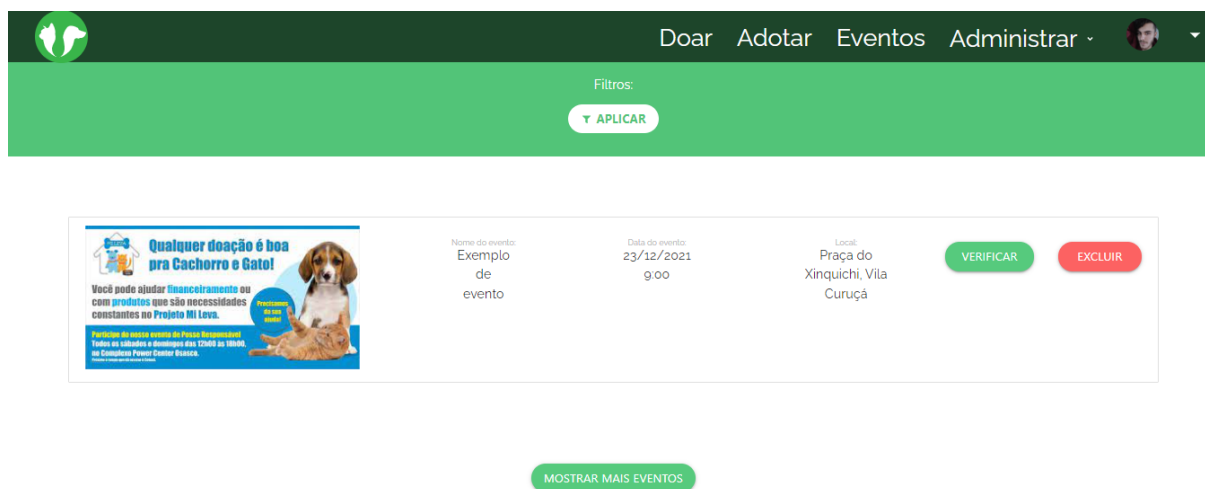
Figura 41 – Tela Validar ONG

The screenshot shows a web interface for validating an NGO. At the top, there is a dark green navigation bar with a logo on the left and links for 'Doar', 'Adotar', 'Eventos', and 'Administrar' on the right. Below the navigation bar, there is a green bar with the text 'Filtros:' and a button 'APLICAR'. The main content area shows a card for an NGO with a profile icon, the text 'Nome do animal: ONG', and two buttons: 'INFORMAÇÕES DA ONG' and 'VALIDAR ONG'.

Fonte: Autoria Própria, 2021.

Para que o evento seja exibido na tela home os eventos cadastrados passarão por uma verificação dos usuários administradores. Essa verificação será feita na tela validar evento, figura 42.

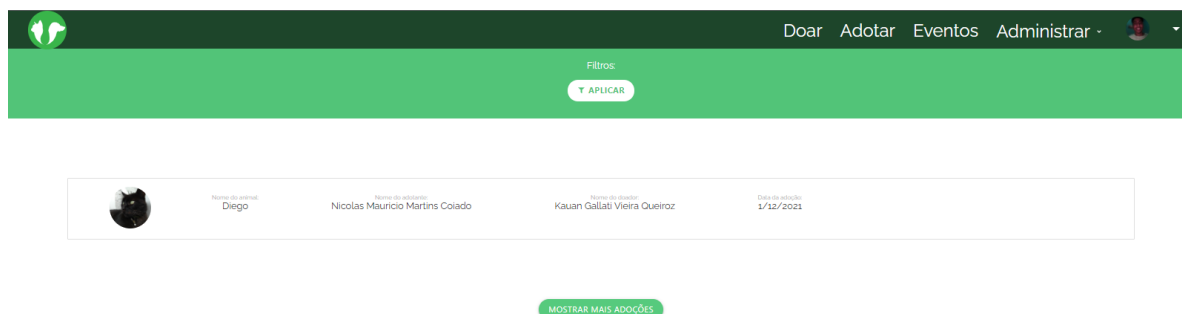
Figura 42 – Tela Validar Evento



Fonte: Autoria Própria, 2021.

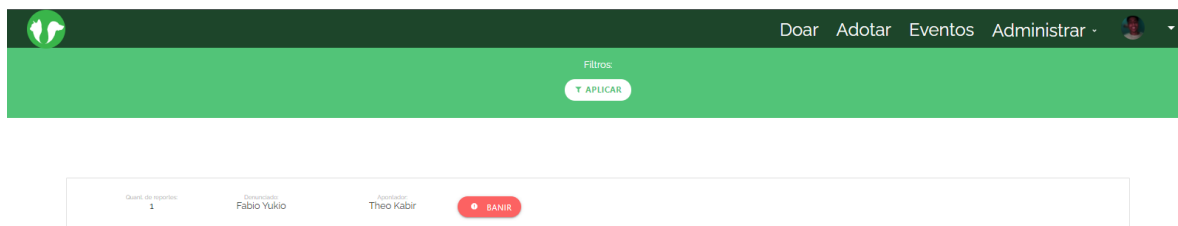
Os administradores terão acesso à uma lista com os processos de adoção realizadas pelos usuários, que será exibida na tela lista de adoções, figura 43.

Figura 43- Tela lista de adoções



Fonte: Autoria Própria, 2021.

Em caso do ferimento dos termos de uso do site, os perfis poderão ser reportados por outros usuários, e posteriormente banidos pelos administradores, que verão esses perfis na tela ver reports, figura 44.

Figura 44- Tela ver reports

Fonte: Autoria Própria, 2021.

Os usuários que eventualmente esquecerem sua senha de acesso ao perfil, poderão utilizar a tela recuperação de senha, figura 45, e responderão uma pergunta de segurança para a redefinição da senha.

Figura 45- Recuperação de senha

Fonte: Autoria Própria, 2021.

Para cadastrar-se em nosso site será necessária a leitura e aceitação dos termos de uso, a partir disso o usuário poderá ter acesso aos recursos do site. A tela termos de uso, figura 46, está disponível no rodapé e durante o cadastro.

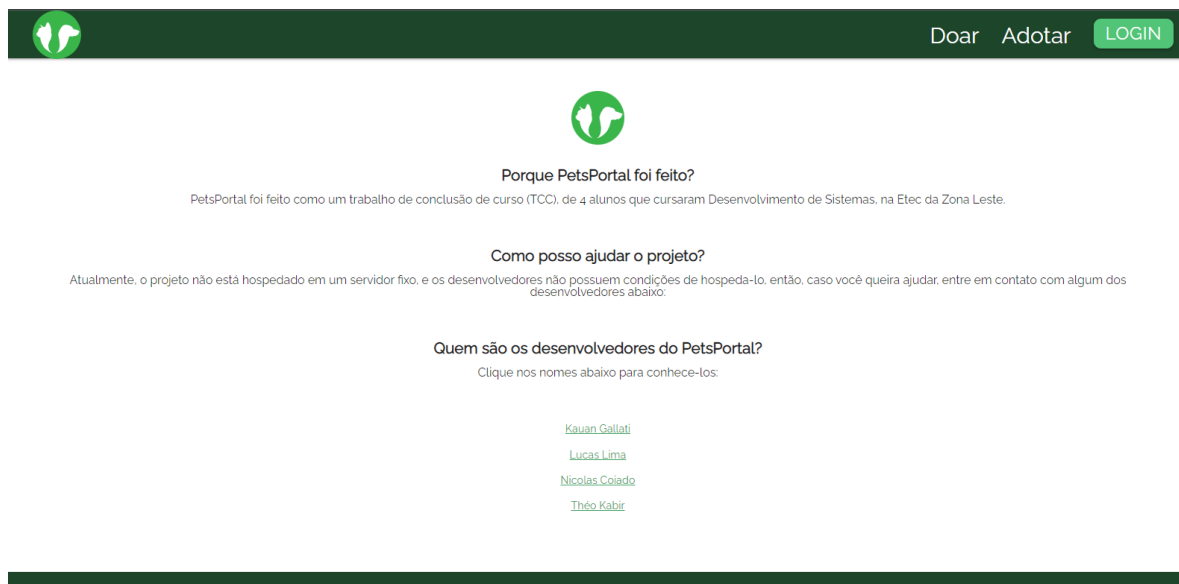
Figura 46- Tela termos de uso



Fonte: Autoria Própria, 2021.

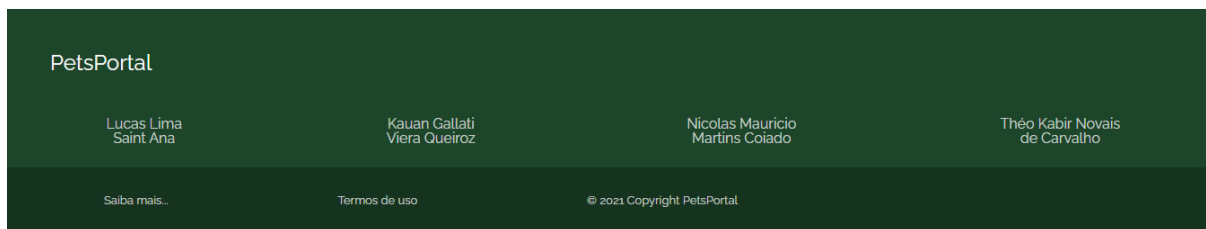
Na tela saiba mais, figura 47, conterà uma tela com as principais informações do projeto Pets Portal, com isso usuários terão acesso a informações sobre quem somos, como ajudar o site, os desenvolvedores e suas redes de contato e o que é o Pets Portal.

Figura 47- Tela saiba mais



Fonte: Autoria Própria, 2021.

Tanto o saiba mais e os termos de uso estarão presentes no rodapé da nossa página, figura 48, que está presente em todas as telas do sistema.

Figura 48- Rodapé da página

Fonte: Autoria Própria, 2021.

4. Conclusão

O estudo e utilização da UML para a modelagem do sistema garantiu um bom alicerce para o desenvolvimento do sistema. Os diagramas de caso de uso, atividade e de sequência foram suficientes para o planejamento das interações dos usuários, lógica de programação e no algoritmo em si.

A linguagem Javascript e suas tecnologias, principalmente o react, supriram completamente as necessidades de nosso site.

Cumprindo o propósito, a aplicação permite o cadastro de animais por usuários que se registrarem, e também ONGs, que podem utilizar da plataforma para divulgação de seus eventos de adoção. A plataforma possui interface intuitiva e pode ser acessado através de qualquer dispositivo com um navegador atualizado a fim de estabelecer comunicação entre os doadores de um animal e seus possíveis adotantes.

REFERÊNCIAS

ANJOS, Leandro Terra Versiani dos. **Segurança da Informação na Programação com uso de Criptografia e Certificação Digital**. 2016. 19 f. TCC (Graduação) - Curso de Tecnologia em Sistemas de Computação, Universidade Federal Fluminense, Niterói- Rj, 2016.

AYLAN. **Como o JWT funciona**. 2019. Disponível em: <https://www.devmedia.com.br/como-o-jwt-funciona/40265>. Acesso em: 27 jul. 2021.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: Guia do usuário**. 2. ed. Rio de Janeiro: Elsevier, 2005. 552 p.

CASTRO, Elizabeth; HYSLOP, Bruce. **HTML5 e CSS3: guia prático e visual**. 7. ed. Rio de Janeiro: Alta Books, 2013

CNN (São Paulo). **Adoção de cães e gatos cresce durante a quarentena**. 29/07/2020. Elaborado por CNN Brasil. Disponível em: <https://www.cnnbrasil.com.br/nacional/2020/07/29/adocao-de-caes-e-gatos-cresce-durante-a-quarentena>. Acesso em: 03 jun. 2021.

D'ANDRETTA, João Paulo Marques. **Análise de fatores que contribuíram para a guarda responsável ou abandono de cães e gatos em área de São Paulo**. 12. 108 f. Tese (Doutorado) - Curso de Epidemiologia Experimental Às Zoonoses, Medicina Veterinária Preventiva e Saúde Animal, USP, São Paulo, 2012.

DARSKI, Reinaldo. **Busca de Indicadores de Eficiência no Uso em Recursos em Aplicações de Transmissão por Upload em um Estudo comparativo de servidor Node.js para Middleware versus um Apache Tomcat**. 2016. 30 f. TCC (Graduação) - Curso de Especialização em Qualidade de Software, Universidade do Vale do Rio dos Sinos, São Leopoldo - Rs, 2016.

DEVMEDIA. **Comparando o NoSQL ao modelo relacional**. 2014. Disponível em: <https://www.devmedia.com.br/comparando-o-nosql-ao-modelo-relacional/30917>

FACEBOOK INC. **Introduzindo JSX**. Disponível em: <https://pt-br.reactjs.org/docs/introducing-jsx.html>. Acesso em: 21 jul. 2021.

FACEBOOK INC. **React Top-Level API**. Disponível em: <https://pt-br.reactjs.org/docs/react-api.html>. Acesso em: 21 jul. 2021.

FERREIRA, Elcio; EIS, Diego. **HTML5: Curso W3C Escritório Brasil**. São Paulo: W3C Escritório Brasil, 2011. 106 p.

FREIRES, Naélio. **Um Guia para usar Node.JS**. 2019. Disponível em: <https://blog.geekhunter.com.br/um-guia-para-usar-react-js/>. Acesso em: 23 jun. 2021.

G1 (São Paulo). **A evolução dos cães até se tornarem animais de estimação**. 29/12/2014. Disponível em: <http://g1.globo.com/sp/bauru-marilia/mundo-pet/2014/noticia/2014/12/mundo-pet-evolucao-dos-caes-ate-se-tornarem-animais-de-estimacao.html>. Acesso em: 03 jun. 2021.

GASPAROTTO, Henrique Machado. **Bootstrap ou Materialize: Conheça as principais diferenças**. 2017. Disponível em: <https://www.devmedia.com.br/bootstrap-ou-materialize-conheca-as-principais-diferencas/37830>. Acesso em: 23 jun. 2021.

GUEDES, Gilleanes. **UML 2 Uma abordagem prática**. 3. ed. São Paulo: Novatec, 2018. 350 p.

HOWS, David; MAMBREY, Peter; PLUGGE, Eelco. **Introdução ao MongoDB**. São Paulo: Novatec, 2015. 168 p.

JUSBRASIL (São Paulo). **Brasil tem 30 milhões de animais abandonados**. 2013. Publicado por Agência de Notícias de Direitos Animais - ANDA. Disponível em: <https://anda.jusbrasil.com.br/noticias/100681698/brasil-tem-30-milhoes-de-animais-abandonados#:~:text=A%20Organização%20Mundial%20da%20Saúde,Destes%2C%2010%25%20estão%20abandonados...> Acesso em: 03 jun. 2021.

KREFTA, Darley Leonardo. **PLATAFORMA PARA AUXÍLIO NO DESENVOLVIMENTO DE RACIOCÍNIO LÓGICO DE CRIANÇAS**. 2019. 53 f. TCC (Graduação) - Curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Universidade Tecnológica Federal do Paraná, Pato Branco, 2019. Disponível em: http://riut.utfpr.edu.br/jspui/bitstream/1/24337/1/PB_COADS_2019_1_01.pdf. Acesso em: 24 jul. 2021.

MARTINS, José Carlos Cordeiro. **Gerenciando Projetos de Desenvolvimento de Software com PMI, RUP e UML**. 5. ed. Rio de Janeiro: Bras, 2010. 316 p.

MILETTO, Evandro Manara; BERTAGNOLLI, Silvia de Castro (org.). **Desenvolvimento de software II: introdução ao desenvolvimento web com html, css, javascript e php**. Porto Alegre: Bookman, 2014. 276 p. (Tekne).

Mozilla Foundation (EUA). **Introdução Express/Node**. 22/06/2021. Disponível em: https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Express_Nodejs/Introduction. Acesso em: 23 jun. 2021.

Mozilla Foundation (EUA). **JavaScript**. 22/06/2021. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>. Acesso em: 23 jun. 2021.

Mozilla Foundation (EUA). **JavaScript**. 23/07/2021. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/CORS>. Acesso em: 24 jul. 2021.

REZENDE, Ricardo. **Conceitos Fundamentais de Banco de Dados**. 2006. Disponível em: <https://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>.

RIBEIRO FILHO, Luiz Felipe Basile. **RU DIGITAL: UM SISTEMA PARA GERENCIAMENTO DE SALDO FINANCEIRO DE CLIENTES DE RESTAURANTES UNIVERSITÁRIOS**. 2018. 70 f. TCC (Graduação) - Curso de Tecnologia em Sistemas Para Internet, Universidade Tecnológica Federal do Paraná, Guarapuava, 2018.

RODRIGUES, Rennan Martini. **CONSTRUINDO UMA API PARA EXIBIR HORÁRIOS DE TRANSPORTE PÚBLICO**. 2015. 111 f. TCC (Graduação) - Curso de Desenvolvimento Web, Universidade Tecnológica Federal do Paraná, Londrina, 2015.

SANTOS, Guilherme Oliveira dos. **Desenvolvimento de um Gerenciador de Smart Devices Usando o Protocolo MQTT e uma Stack em Javascript para Modelar um Cenário de IoT**. 2017. 41 f. Monografia (Especialização) - Curso de Ciência da Computação, Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

SCUDERO, Erick. **Os 6 frameworks front-end mais amados no mundo (segundo o GitHub)**. 2018. Disponível em: <https://becode.com.br/frameworks-front-end-mais-amados-segundo-github/>. Acesso em: 23 jun. 2021.

SILVA, Maurício Samy. **JavaScript - Guia do Programador: guia completo das funcionalidades da linguagem javascript**. São Paulo: Novatec, 2010. 608 p.

SOUSA, Romulo. **Consumindo uma API com React.JS e Axios**. 2021. Disponível em: <https://www.devmedia.com.br/consumindo-uma-api-com-react-js-e-axios/42900>. Acesso em: 24 jul. 2021.

TAKAGI, Jennifer Camila dos Reis. **ESPECIFICAÇÃO DE PARÂMETROS PARA O USO DA FERRAMENTA ESLINT EM PROJETOS REACT JS PARA MANUTENÇÃO DE CÓDIGO LIMPO**. 2021. 216 f. TCC (Graduação) - Curso de Gestão da Tecnologia da Informação, Departamento Acadêmico de Saúde e Serviços, Instituto Federal de Santa Catarina, Florianópolis - Sc, 2021.

TEIXEIRA, Pedro Henrique Faria. **Uma API REST para a contratação de profissionais na aplicação Severino**. 2021. 68 f. Monografia (Doutorado) - Curso de Ciência da Computação, Faculdade de Computação, Universidade Federal de Uberlândia, Uberlândia, 2021. TEIXEIRA, Pedro Henrique Faria. Uma API REST para a contratação de profissionais na aplicação Severino. 2021. 68 f. Monografia (Doutorado) - Curso de Ciência da Computação, Faculdade de Computação, Universidade Federal de Uberlândia, Uberlândia, 2021.

TEJADA, Zoiner; OLPROD. **Dados não relacionais e NoSQL**. 2018. Disponível em: <https://docs.microsoft.com/pt-br/azure/architecture/data-guide/big-data/non-relational-data>. Acesso em: 30 jul. 2021.

URBAN, Diego Leonardo. **RESOURCE OPTIMIZER API: otimização de recursos de páginas web**. 2017. 79 f. Monografia (Especialização) - Curso de Ciência da Computação, Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2017.

VEIGA, Edison. **A 'epidemia de abandono' dos animais de estimação na crise do coronavírus**. 30/07/2020. Disponível em: <https://www.bbc.com/portuguese/brasil-53594179>. Acesso em: 03 jun. 2021.

VIEIRA, Ricardo. **Uso da UML na Especificação do Sistema de Informação da área de infraestrutura da UFRGS**. 2003. 86 f. TCC (Graduação) - Curso de Especialização Web e Sistemas da Informação, Informática Aplicada, Universidade do Rio Grande do Sul, Rio Grande do Sul, 2003.