

**ESCOLA TÉCNICA ESTADUAL
DEP. ARY DE CAMARGO PEDROSO**

Téc. Desenvolvimento de Sistemas

Davi Almeida Anastacio

Enzo Ramos Moniz

Gabriel Ambrósio de Camargo

Nicolas Cordeiro Rodrigues

Pedro Henrique Caparoz

Robson Passarini

AdotApp

Solução para adoção de animais.

Piracicaba

2022

Davi Almeida Anastacio
Enzo Ramos Moniz
Gabriel Ambrósio de Camargo
Nicolas Cordeiro Rodrigues
Pedro Henrique Caparoz
Robson Passarini

AdotApp

Aplicativo de adoção de animais.

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Desenvolvimento de Sistemas da Etec Dep. Ary de Camargo Pedrosa, orientado pelo Prof. Rodrigo Viviani como requisito parcial para obtenção do título de técnico em Desenvolvimento de Sistemas.

Piracicaba

2022

TERMO DE CONSENTIMENTO

TRABALHO DE CONCLUSÃO DE CURSO – TCC

Nós, alunos abaixo assinados, regularmente matriculados no **Curso Técnico de Desenvolvimento de Sistemas** na **ETEC Deputado Ary de Camargo Pedroso** do Município de Piracicaba/SP declaramos ter pleno conhecimento dos Regulamentos para realização do Trabalho de Conclusão de Curso do Centro Paula Souza. Declaramos, ainda que, o trabalho apresentado é resultado do nosso esforço e autorizamos a divulgação do mesmo pela instituição.

Piracicaba, 13 de dezembro de 2022

DEDICATÓRIA

Dedicamos esta pesquisa aos nossos pais, nossos maiores e melhores orientadores na vida.

AGRADECIMENTOS

Agradecemos a todas as pessoas, familiares e amigos, que nos ajudaram no desenvolvimento deste trabalho, mesmo que indiretamente, com o apoio e paciência ao ouvir nossas dúvidas e aflições.

Ao nosso orientador, Rodrigo Viviani, pela dedicação e paciência nos atendimentos. A todos os professores pelas dicas e orientações prestadas ao grupo, que mesmo não sendo nossos orientadores, disponibilizaram um pouco de seu tempo para nos atender, o que nos levou ao desenvolvimento de muitas ideias.

Aos professores entrevistados, pela disponibilidade e atenção em fornecer as informações necessárias para a realização deste trabalho.

EPÍGRAFE

“Disciplina é a ponte entre metas e realizações” (Jim Rohn)

RESUMO

O projeto de TCC é a realização de um site web para adoção de animais, este foi decidido devido ao alto índice de animais abandonados nas ruas e, conseqüentemente, o aumento de pessoas procurando ajudar esses animais. As pesquisas feitas mostram que a grande maioria das pessoas tem interesse em ajudar e adotar esses animais, logo, o site web teria como principal objetivo disponibilizar ao público uma plataforma acessível, prática e dinâmica, conectando todo o público interessado na adoção ou colaboração direcionada a animais de rua e/ou animais domésticos que também buscam o serviço de adoção. Para atingir esse objetivo, foi pretendido desenvolver um site com características modernas, de fácil acessibilidade e entendimento para até os mais inexperientes usuários de rede social. Por fim, também foi planejado desenvolver um banco de dados de fácil e rápida comunicação com o site web, armazenando as informações dos usuários (CPF, RG, data de nascimento, endereço, gênero, número telefônico, permissões, etc.) e as informações dos animais (espécie, idade, nome, local, referência fotográfica, sexo, castração, histórico de vacinação, descrição).

Palavras-Chave: Web, Site, Animais, Adoção.

ABSTRACT

The TCC project is the realization of a website for the adoption of animals. This was decided due to the high rate of abandoned animals on the streets and, consequently, the increase in people looking to help these animals. The research carried out shows that the vast majority of people are interested in helping and adopting these animals, therefore, the website's main objective would be to provide the public with an accessible, practical and dynamic platform, connecting all the public interested in adoption or collaboration aimed at stray animals and/or domestic animals that also seek the adoption service. To achieve this goal, it is intended to develop a website with modern features, easy to access and understand to even the most inexperienced social network users. Finally, it is also planned to develop a database for easy and quick communication with the website, storing user information (CPF, ID, date of birth, address, gender, telephone number, permissions, etc.) and the animals's information as well (species, age, name, location, photographic reference, sex, castration, vaccination history, description).

Key-Words: Web, Site, Animals, Adoption.

LISTA DE FIGURAS

Figura 1 – Animais abandonados no Brasil	13
Figura 2 – Cronograma	14
Figura 3 – Diagrama de Caso e Uso	15
Figura 4 – Fluxograma de Dados	16
Figura 5 – Fluxograma (Adotar – ampliado)	17
Figura 6 – Fluxograma (Entrar/Cadastrar – ampliado)	18
Figura 7 – Fluxograma (Sobre Nós/Contato – ampliado)	19
Figuras 8 a 13 – Formulário AdotApp	20
Figura 14 – HTML	24
Figura 15 – CSS	25
Figura 16 – Javascript	26
Figura 17 – PHP	27
Figura 18 – SQL	28
Figura 19 – Notepad++	29
Figura 20 – Visual Studio Code	30
Figura 21 – phpMyAdmin	31
Figura 22 – WampServer	32
Figura 23 – MySQL	33
Figura 24 – Canva	34
Figura 25 – Microsoft Word	35
Figura 26 – Lucidchart	36
Figura 27 – Tela de Cadastro	37
Figura 28 – Cadastro (Indeferido)	38
Figura 29 – Cadastro (Deferido)	39
Figuras 30 a 32 – Código de validação (Cadastro)	40
Figura 33 – Tela de Login	43
Figura 34 – Login (Confirmado)	44
Figuras 35 a 37 – Código de validação (Login)	45
Figura 38 – Menu de Adoção	49
Figura 39 – Tela de Adoção	50
Figura 40 – Anúncios do Usuário	51
Figura 41 – Cadastro de Cachorro	52
Figura 42 – Cadastro de Gato	53
Figura 43 – Tabelas do Banco de Dados	54
Figura 44 – Tabela de Animais Cadastrados	54
Figuras 45 e 46 – Código Site Banco de Dados	55
Figura 47 – Criptografia de Senha	57
Figura 48 – Dados da Mensagem de Contato	57
Figura 49 – Página Inicial (Cabeçalho)	58
Figura 50 – Página Inicial (Rodapé)	59
Figura 51 – Menu do Usuário	60
Figura 52 – Formulário de Contato	61

LISTA DE ABREVIATURAS E SIGLAS

Trabalho de Conclusão de Curso (TCC)
Word Wide Web (WEB)
HyperText Markup Language (HTML)
Cascading Style Sheets (CSS)
Hypertext Preprocessor (PHP)
Structured Query Language (SQL)
Universidade de São Paulo (USP)
União Internacional Protetora dos Animais (UIPA)
Organização Mundial da Saúde (OMS)
Instituto Pet Brasil (IPB)
Instituto Brasileiro de Geografia e Estatística (IBGE)
Organizações não-governamentais (ONGs)
Unified Modeling Language (UML)

SUMÁRIO

RESUMO	8
1 – INTRODUÇÃO	9
1.1 – Tema	10
1.2 – Objetivos Geral e Específico	11
1.3 – Justificativa	11
1.4 – Metodologia	11
1.5 – Fundamentação Teórica	12
2 – PLANEJAMENTO	14
2.1 – Cronograma	14
2.2 – Diagrama de Caso de Uso	15
2.3 – Fluxograma de Dados	16
2.4 – Formulários	19
3 – DESENVOLVIMENTO	23
3.1 – Linguagens Utilizadas	24
3.1.1 – HTML	24
3.1.2 – CSS	25
3.1.3 – Javascript	26
3.1.4 – PHP	27
3.1.5 – SQL	28
3.2 – Plataformas Utilizadas	29
3.2.1 – Notepad++	29
3.2.2 – Visual Studio Code	30
3.2.3 – phpMyAdmin	31
3.2.4 – WampServer	32
3.2.5 – MySQL	33
3.2.6 – Canva	34
3.2.7 – Microsoft Word	35
3.2.8 – Lucidchart	36
3.3 – Desenvolvimento do Site	36
3.3.1 – Tela de Cadastro	37
3.3.2 – Tela de Login	43
3.3.3 – Menu de Adoção	48
3.3.4 – Página Inicial	58
4 – RESULTADOS OBTIDOS	62
5 – CONCLUSÃO	63

1 – INTRODUÇÃO

Títulos alinhados à esquerda. Não use o traço 1 INTRODUCAO

É evidente que o problema da situação de rua dos animais no Brasil vem se agravando cada vez mais. Segundo a Gazeta Digital, atualmente o número de animais abandonados está em constante crescimento por todo o Brasil, especialmente durante o período de pandemia, o que faz com que o número de animais que podem ser resgatados aumente também, porém a maioria das pessoas que podem ajudar esses animais abandonados, não tem como colocá-los dentro de sua própria casa, o que faz com que na maioria das vezes apenas ignorem a situação do animal. Para a amenização deste quadro, foi elaborado o *website* AdotApp, que ajudará a diminuição da quantidade de animais sem dono. Ao utilizá-lo, o usuário poderá cadastrar informações de animais em situação de rua, ou, adotar os pets que já estão cadastrados, além disso, é possível ver o funcionamento do sistema de adoção.

espaços

1.1 – Tema

O tema proposto é a criação de um *website* personalizado para a adoção de animais, que irá ajudar na melhora da situação de animais abandonados e de rua. O site é personalizado de acordo com as características estudadas nas pesquisas que foram feitas. Uma parte importante do sistema é a facilidade de comunicação entre os usuários, e promover uma facilidade de adoção desses animais. De acordo com os resultados obtidos por pesquisas produzidas pelo grupo, foi observado que inúmeras pessoas não adotam, ou ajudam animais, por falta de oportunidade ou recursos.

Esse tema tem como base as pesquisas realizadas, que mostraram que a maioria das pessoas têm vontade de adotar algum animal, mas que não o fazem por diversos fatores, o AdotApp vem com o objetivo de facilitar a conexão entre os usuários.

1.2 – Objetivo Geral e Específico

O objetivo é facilitar a adoção e resgate de animais. Fazendo com que o número de animais abandonados nas ruas diminua, através de um ambiente de adoção segura que cria uma confiança e incentiva essas ações

1.3 – Justificativa

Pretende-se realizar este projeto devido ao grande crescimento de abandonados no país inteiro. O número de animais abandonados no Brasil durante a pandemia aumentou em quase 60%, alguns protetores declararam aumento de abandono de 300%, de 150%, outros de 30%. Esses dados se tornam ainda mais agravantes quando é possível ver que o número de doações também diminuiu por causa da pandemia, em que quase não teve eventos de adoção. Há a necessidade de pessoas que podem ter um animal em casa, mas que por algum fator, não conseguem achar uma pessoa que esteja disposta a doar, ou não conseguem resgatar da rua. Por outro lado, há também pessoas que, por alguma dificuldade, precisam colocar animais para adoção, e veem como forma mais fácil, simplesmente abandonar o animal na rua, ou em lugares desertos.

Este é o principal motivo da criação do AdotApp, um site para facilitar esse tipo de situação, colaborando com os dois tipos de público, os que querem doar, e os que querem adotar.

1.4 – Metodologia

Foi utilizado no projeto os programas e linguagens Sublime Text, HTML, CSS, JavaScript, PHP, SQL, Visual Studio Code, MySQL, PHPMyAdmin, Wampserver, Canva, e Microsoft Word.

1.5 – Fundamentação Teórica

Segundo o Jornal da USP e os dados da Uipa (União Internacional Protetora dos Animais), mostram que aumentou em 400% a procura de animais para adoção. Isso é um reflexo da pandemia, que fez com que as pessoas ficassem mais reclusas em suas casas. Porém, esse entusiasmo para adotar os animais, faz com que muitas pessoas esqueçam da dificuldade que é criar um animal em casa, fazendo com que se torne gigante o número de abandonos, o que só contribui para a elevação do número de animais nos abrigos ou nas ruas.

A professora Deise Dellova, chefe do Hospital Veterinário de Pequenos Animais da Faculdade de Zootecnia e Engenharia de Alimentos da USP de Pirassununga, lembra que é preciso ter muita consciência, não agir por impulso, para não se arrepender depois, pois há vários fatores envolvidos nesse processo. “O animal não é um objeto, ele precisa de cuidados e atenção e, quando isso se tornou um peso, a solução foi devolver ao abrigo de animais ou soltar nas ruas.

Dados da Organização Mundial da Saúde (OMS) apontam que, no Brasil, existem cerca de 30 milhões de animais abandonados; desse total, 10 milhões são gatos, e 20 milhões, cachorros. Os traumas de um abandono deixam várias cicatrizes na vida de um pet, muitas vezes é necessário a ajuda de um veterinário ou adestrador para reverter essa situação.

“A gente tem o número de que 2,1% dos animais abandonados vieram de uma situação de vulnerabilidade, evoluíram pra uma situação pior”, afirma o presidente do Instituto Pet Brasil (IPB), Caio Villela.

“É observado que os números da vulnerabilidade acompanham a situação econômica do país. Se uma população empobrece, a vulnerabilidade desses animais aumenta. A gente consegue relacionar isso porque o IBGE indica que mais de 50% dos lares brasileiros têm animais”, destaca Villela.

Outro indicador importante é o número de ONGs, que aumentou no período. Em 2018, o IPB monitorava 370 instituições e agora já são mais de 400. A maioria (45%) está localizada no Sudeste e é de médio porte, abrigando de 100 a 500 animais.

Combater o problema é fundamental. Mais importante ainda é não deixar que ele aconteça.

Figura 1 – Animais abandonados no Brasil



Fonte: Bicho Legal, 2014

2 – PLANEJAMENTO

Devido ao grande número de animais abandonados no Brasil – que segue somente aumentando –, foi planejado fazer este projeto destinado para aqueles que querem adotar animais e aqueles que acabam por encontrar animais perdidos e/ou abandonados, e querem colocá-los para adoção para encontrarem um novo lar. O projeto será um site que contará com cadastro de doadores, de adotantes, de animais, chat para os adotantes conversarem com os doadores.

2.1 – Cronograma

Para designar de maneira ordenada as tarefas e funções de cada um dos componentes do grupo no desenvolvimento do TCC, foi feito o cronograma correspondente para o ano todo, onde é indicado o responsável ou os responsáveis por cada processo pelo qual seria passado ao longo do ano na confecção do projeto.

Figura 2 – Cronograma



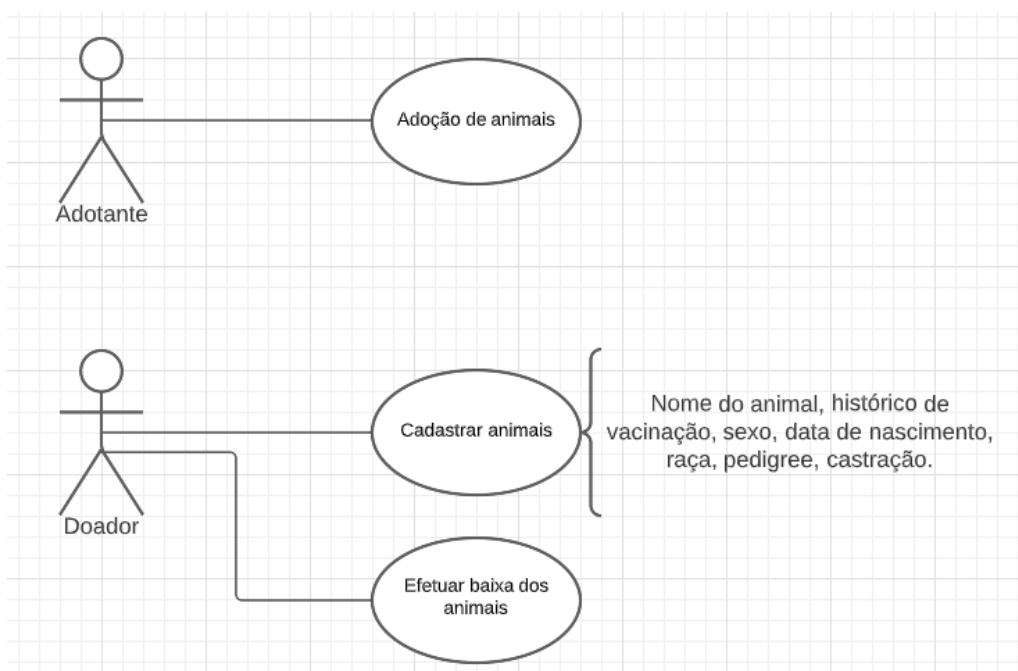
Fonte: Autores, (2022)

2.2 – Diagrama de Caso e Uso

O diagrama de caso e uso é de suma importância para o levantamento de requisitos funcionais para o sistema em questão. Na Linguagem de Modelagem Unificada (UML), esse diagrama descreve e denota os detalhes dos usuários do seu sistema e as formas as quais podem interagir com a aplicação.

Com sua elaboração, logo ao início do projeto, é facilitada a compreensão entre os mesmos a respeito de como deve se suceder o sistema na prática e como deve ser estruturada sua programação.

Figura 3 – Diagrama de Caso e Uso



Fonte: Autores, (2022)

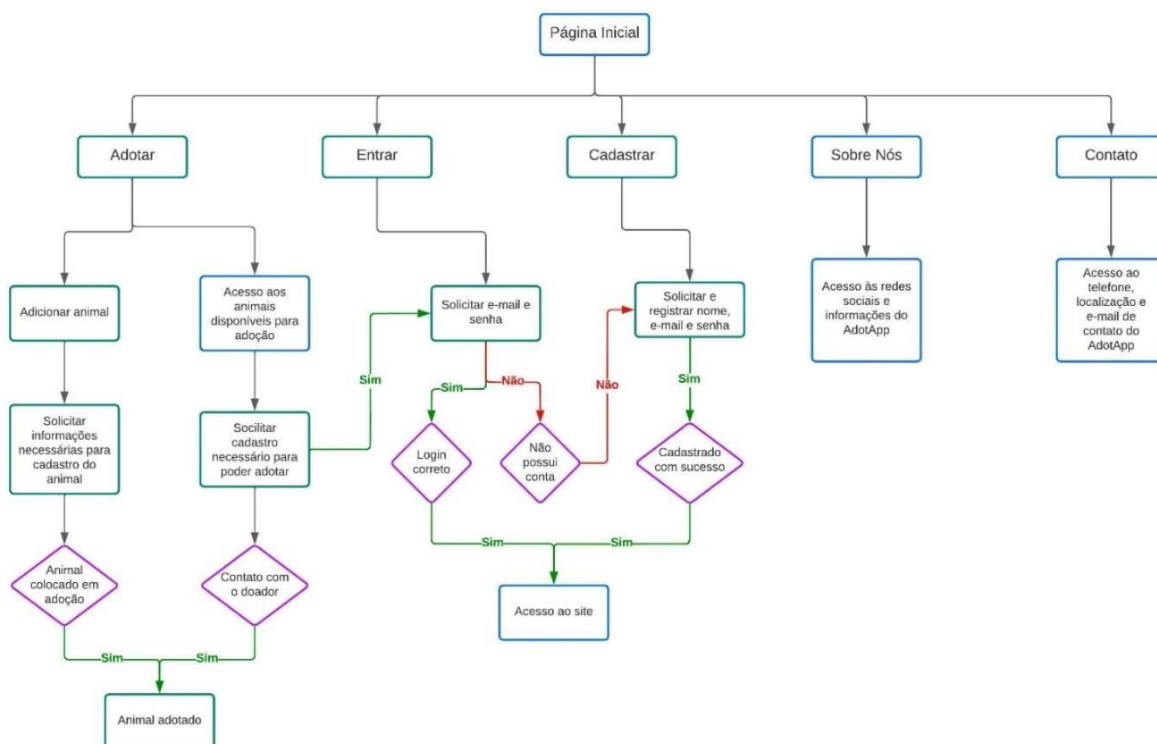
2.3 – Fluxograma de Dados

Um fluxograma de dados é um diagrama em fluxo que mapeia as informações decorridas em qualquer tipo de processo para qualquer tipo de sistema. Através de retângulos, círculos, flechas, losangos, entre outros, é feita a representação gráfica do fluxo de entradas e saídas de dados, do armazenamento e seus respectivos destinos finais.

É ideal para o mapeamento de como irá ser o funcionamento e os fluxos dos dados dentro de um sistema.

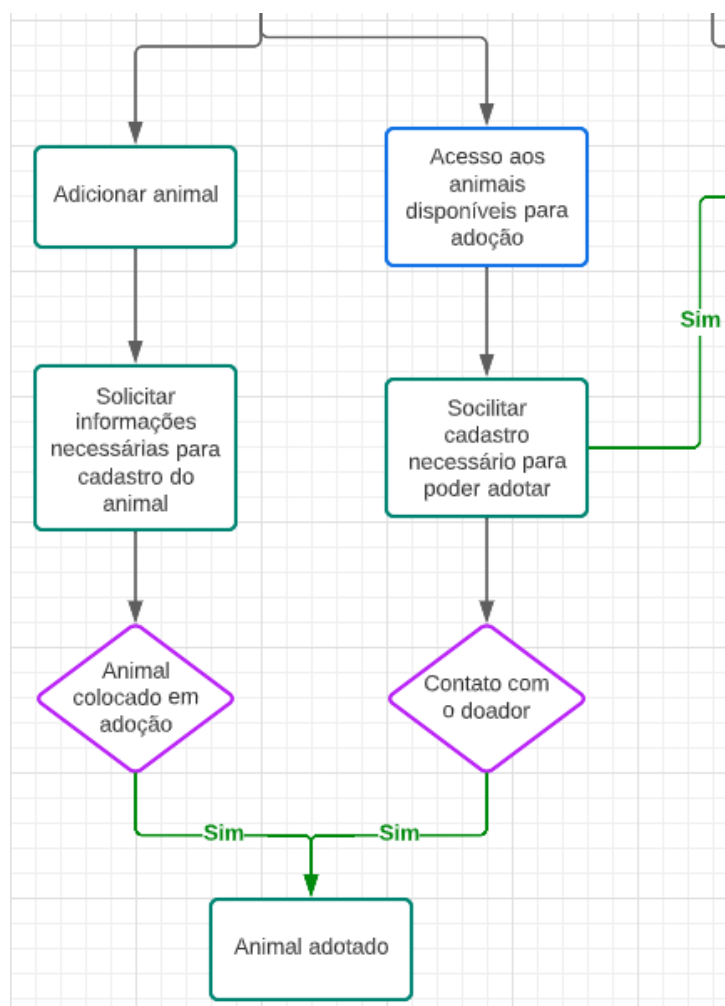
Fora desenvolvido, pela plataforma Lucidchart, este diagrama para o planejamento do site, banco de dados e a conexão entre ambos, esquematizando então, como iria se desenvolver a funcionalidade do site.

Figura 4 – Fluxograma de Dados



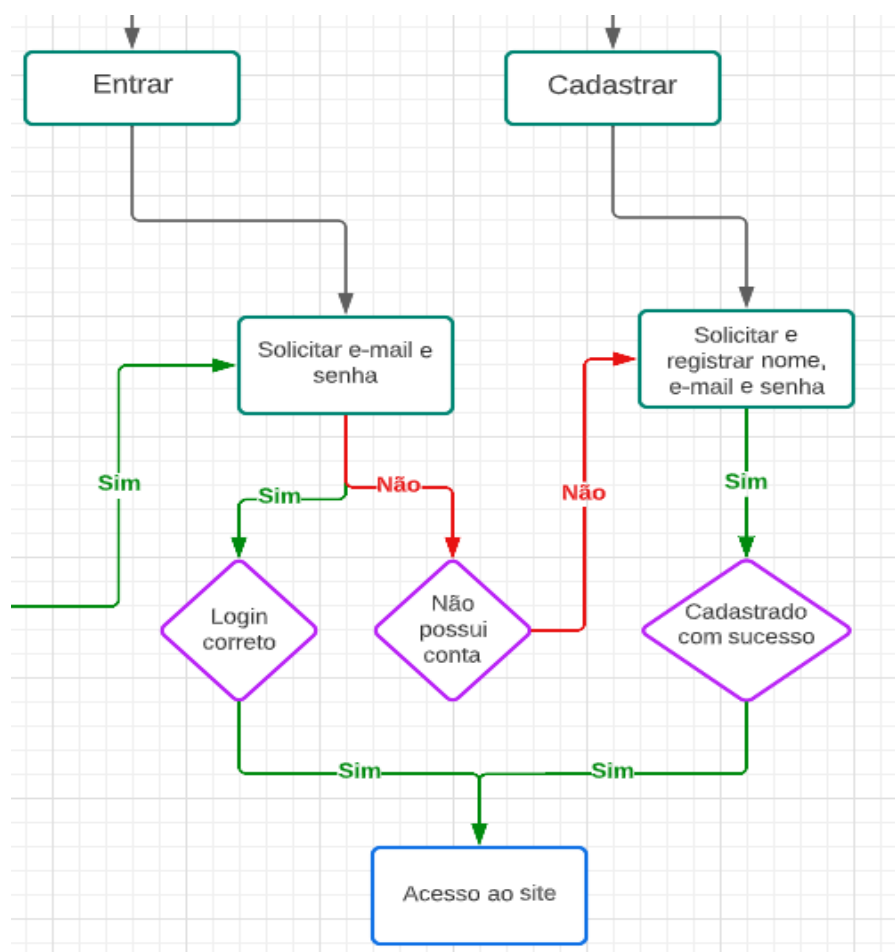
Fonte: Autores, (2022)

Figura 5 – Fluxograma (Adotar – ampliado)



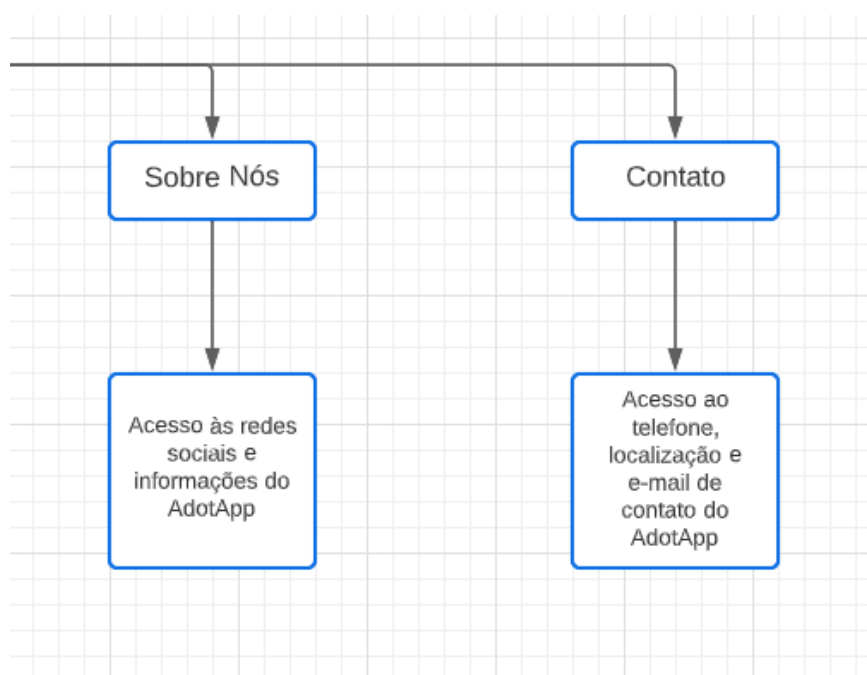
Fonte: Autores, (2022)

Figura 6 – Fluxograma (Entrar/Cadastrar – ampliado)



Fonte: Autores, (2022)

Figura 7 – Fluxograma (Sobre Nós/Contato – ampliado)



Fonte: Autores, (2022)

2.4 – Formulários

Com o propósito de embasar mais profundamente o projeto e, reforçar as causas pelo qual o AdotApp é motivado e tenta contribuir, foi feito um formulário com perguntas e respostas de múltipla escolha, através da plataforma Google Forms. O formulário foi disponibilizado já à época do desenvolvimento inicial do projeto.

A divulgação foi feita por parte da equipe e, graças a ela, foram recebidas 74 respostas. Um dos objetivos com esse questionário era de se ter um contato mais próximo com possíveis usuários que estariam interessados em usufruir da plataforma, assim reunindo um feedback por parte do público, captando as diferentes visões, situações e condições dos usuários, para, também, conhecer e entender melhor para quem seria nossa plataforma e para quais caminhos deveríamos seguir.

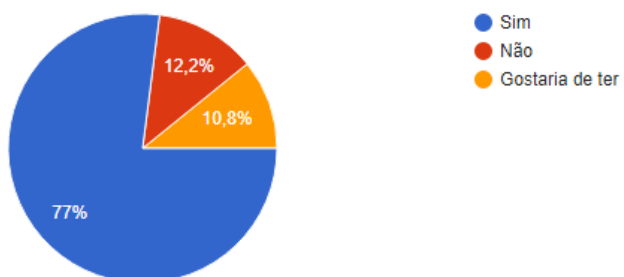
Vale-se ressaltar a importância do formulário em divulgar o projeto e a ideia em si, funcionando como uma ótima forma de captação de usuários interessados em adoção de animais. Abaixo, através dos gráficos gerados pelo Google Forms, pode-

se ver as respostas dos usuários, dadas pelas porcentagens em cada uma das perguntas (Figuras 8 a 13):

Figura 8 – Formulário AdotApp

Você tem animal em casa?

74 respostas

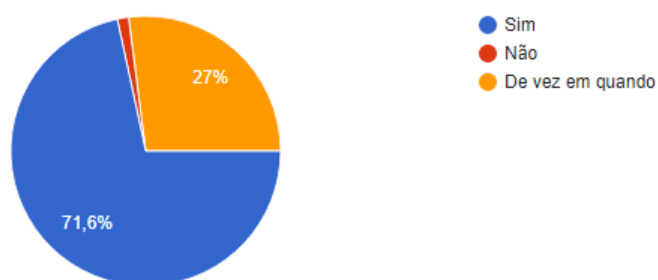


Fonte: Autores, (2022)

Figura 9 – Formulário AdotApp

Você vê muitos animais abandonados ?

74 respostas

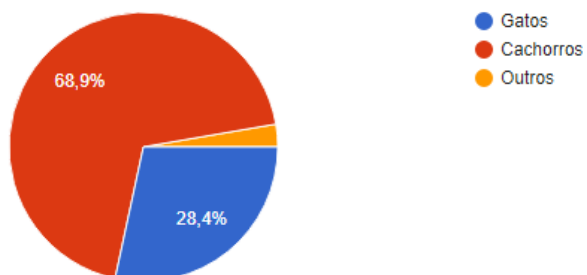


Fonte: Autores, (2022)

Figura 10 – Formulário AdotApp

Qual tipo de animal você prefere?

74 respostas

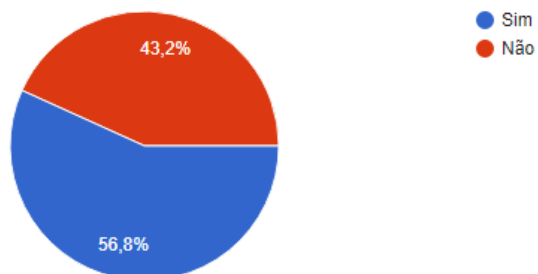


Fonte: Autores, (2022)

Figura 11 – Formulário AdotApp

Você ajuda animais abandonados na rua?

74 respostas

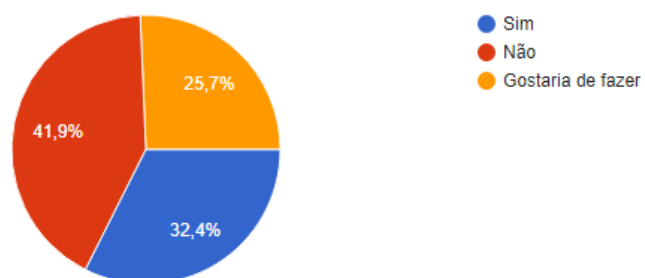


Fonte: Autores, (2022)

Figura 12 – Formulário AdotApp

Você conhece ou faz parte de alguma ONG?

74 respostas

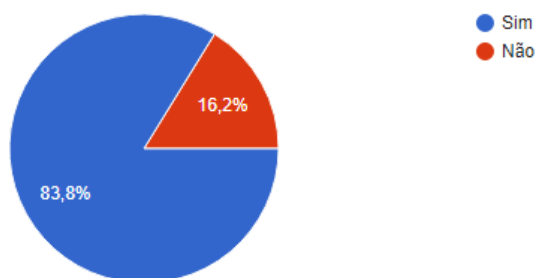


Fonte: Autores, (2022)

Figura 13 – Formulário AdotApp

Você usaria o AdotApp?

74 respostas



Fonte: Autores, (2022)

3 – DESENVOLVIMENTO

A princípio, a ideia inicial era de se fazer um aplicativo destinado a adoção de animais, porém como fora escolhido o Android Studio como banco de dados, mas a escola não o possuía instalado nas máquinas logo foi abandonada a ideia, e houve uma tentativa de utilização de um banco de dados chamado Firebase, a plataforma foi estudada, e começou-se a programação, mas também ocorreu uma mudança de ideia, devido ao curto prazo, o pouco conhecimento e o conflito que estava dando quando se era feito em máquinas diferentes, porém foi mantida a ideia de criar algo relacionado a animais abandonados, já que o número de animais abandonados vem crescendo muito.

Com essa necessidade então foi resolvido de se criar um site para que as pessoas possam cadastrar e adotar os animais abandonados

Começa-se a pôr em prática a partir da divisão de tarefas e alguns componentes do grupo ficaram encarregados em realizar a criação do site e outros faziam a documentação.

Por ser um projeto que exigia tempo para produzi-lo, existiram algumas dificuldades, porém com muitas pesquisas, ajuda de professores e colegas de classe, foi possível que o projeto fosse desenvolvido.

3.1 – Linguagens Utilizadas

As linguagens utilizadas na programação do TCC são:

3.1.1 – HTML

HTML é uma linguagem de marcação utilizada no desenvolvimento de páginas na web, podendo ser interpretada por praticamente todos os navegadores utilizados atualmente.

Essa linguagem é composta por uma série de marcações que dizem para os servidores da web qual é o estilo e a estrutura de um documento. Com o HTML, é possível estruturar seções, parágrafos e links usando elementos, tags e atributos.

Figura 14 – HTML



Fonte: HTML 5.2, (2017)

3.1.2 – CSS

CSS é uma linguagem de estilo que funciona como um mecanismo para determinar o estilo dos elementos de um documento em HTML ou XML.

O CSS define toda a parte visual da página, alterando as cores do texto e do fundo, fonte, espaçamento entre os parágrafos, ajustando imagens, dentre as mais diversas variações de layout.

Figura 15 – CSS

CSS



Fonte: CSS3, (2012)

3.1.3 – Javascript

Javascript é uma linguagem de programação interpretada estruturada, com tipagem dinâmica fraca e multiparadigma. Aplicada principalmente para desenvolvimento Web e desenvolvimento de softwares.

Também responsável pela parte visual da aplicação, o Javascript permite que ocorra a interação entre o usuário e a página e é responsável por sua dinamicidade.

Figura 16 – Javascript



Fonte: ECMAScript, (2022)

3.1.4 – PHP

PHP é uma linguagem de programação utilizada por programadores e desenvolvedores para construir sites dinâmicos, extensões de integração de aplicações e para acelerar o desenvolvimento de um sistema. Essa linguagem é uma das mais utilizadas pela facilidade em aprendê-la, manuseá-la, além de ser compatível com quase todos os sistemas operacionais que existem.

Figura 17 – PHP



Fonte: PHP 8.1.5, (2022)

3.1.5 – SQL

SQL é uma linguagem de programação para lidar com banco de dados relacional, baseado em tabelas. Foi criado para que vários desenvolvedores pudessem acessar e modificar dados de uma empresa simultaneamente, de maneira descomplicada e unificada.

Figura 18 – SQL



Fonte: SQL:2016, (2016)

3.2 – Plataformas Utilizadas

As plataformas utilizadas na composição de alguns elementos do TCC são os programas:

3.2.1 – Notepad++

O Notepad++ é um editor de texto e de código fonte, funcionando para várias linguagens de programação populares. É um software livre de código aberto, sendo bastante utilizado por ser multiuso e sua intuitividade.

Figura 19 – Notepad++

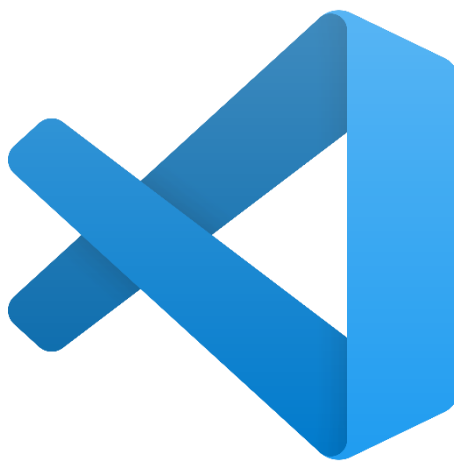


Fonte: Notepad-Plus-Plus, (2022)

3.2.2 – Visual Studio Code

O Visual Studio Code é um editor de código aberto, disponível tanto para Windows, quanto para Mac OS e Linux e atende a uma gama enorme de projetos, não apenas ASP.NET, como também Node.js. Adicionalmente, o editor possui suporte à sintaxe de diversas linguagens como Python, Ruby, C++.

Figura 20 – Visual Studio Code



Fonte: Code Visual Studio, (2022)

3.2.3 – phpMyAdmin

O phpMyAdmin é uma ferramenta de suporte à criação e ao acesso simplificado a bancos de dados de aplicações. Por meio de uma interface simples, diretamente do navegador, qualquer alteração pode ser feita, eliminando a necessidade de linhas de códigos.

Figura 21 – phpMyAdmin



Fonte: phpMyAdmin, (2022)

3.2.4 – Wampserver

WampServer é uma aplicação que instala um ambiente de desenvolvimento web no Windows. Com ele pode-se criar aplicações web com Apache2, PHP e banco de dados MySQL. Além disso, é possível gerenciar facilmente seus bancos de dados com a ferramenta phpMyAdmin que faz parte do pacote.

Figura 22 – WampServer



Fonte: WampServer, (2022)

3.2.5 – MySQL

O MySQL é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O MySQL utiliza a linguagem SQL, que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados.

Figura 23 – MySQL



Fonte: MySQL, (2018)

3.2.6 – Canva

O Canva é um editor gráfico gratuito que permite criar artes de forma fácil, usando modelos prontos ou criando os próprios layouts. O serviço é útil para criar posts para redes sociais ou para sites, cartões para impressão, materiais gráficos como cartazes, folderes, currículos, entre várias outras opções.

Figura 24 – Canva



Fonte: Canva, (2022)

3.2.7 - Microsoft Word

O Microsoft Word é um editor de texto produzido pela Microsoft, suas funções variam das mais simples como escrever um texto comum, até produzir grandes documentos e formulários.

Ele é um programa muito utilizado, pois é o mais comum editor/criador de textos para as plataformas de computador.

Figura 25 – Microsoft Word



Fonte: Microsoft, (2016)

3.2.8 – Lucidchart

O Lucidchart é uma solução online com foco na criação de diagramas e wireframes. Com uma interface baseada em arrastar e soltar elementos, tem uma seleção vasta de recursos para equipes de diversas áreas aproveitarem. Possuindo até vários templates já prontos.

Figura 26 – Lucidchart



Fonte: Lucidchart, (2022)

3.3 – Desenvolvimento do Site

O projeto foi programado em PHP, Javascript, HTML e CSS, utilizando, principalmente, o Visual Studio Code como editor de código, além disso o banco de dados foi majoritariamente desenvolvido através do MySQL e, posteriormente, integrado e conectado à programação do site, fixando assim, a base de dados.

Como um site que trabalha necessariamente com informações pessoais dos usuários, se não possuir uma conta e não a criar, a navegação pelo site fica limitada apenas à página inicial onde se contém dizeres básicos sobre o projeto e informações pertinentes sobre o mesmo. Porém, ao se registrar preenchendo corretamente os campos necessários, o usuário, a partir de então, possuirá um cadastro e uma conta na qual, sob ela, poderá navegar livremente pelo site e, sobretudo, acessar a seção de adoção.

Ao clicar em “Adotar”, o usuário irá se deparar com fotos dos animais que foram ali colocados para adoção por outros usuários, podendo entrar no anúncio e visualizar as informações mais específicas e detalhadas do animal como nome, raça, sexo, vacinas, etc. Para assim, poder entrar em contato com o doador e adotar o animal, será disponibilizado, após clicar na foto do animal, na tela, junto às informações do animal, um telefone e/ou e-mail para contato com o doador em questão.

3.3.1 – Tela de Cadastro

O usuário que não possui uma conta, deverá clicar no texto, presente na página de login, que o direciona para o cadastro, onde deve-se preencher corretamente os campos, contendo nome, e-mail, senha e tendo de digitar novamente a senha para confirmá-la. Após isso, o usuário somente deverá clicar em “Cadastrar” e seguir para o site, agora já registrado.

Se já possuir uma conta, tem de clicar no texto que leva à página de login. A figura abaixo apresenta esses recursos (Figura 27):

Figura 27 – Tela de Cadastro



A imagem mostra a interface de usuário para o cadastro. No topo, há um ícone circular com um coração e silhuetas de um gato e um cachorro. Abaixo, há quatro campos de entrada de texto rotulados: 'Nome', 'Email', 'Senha' e 'Confirme sua senha'. Cada campo é seguido por um botão de seta para a direita. Na base, há um botão verde com o texto 'Cadastrar' e um link azul que diz 'Já tem uma conta? [Entre Aqui.](#)'

Fonte: Autores, (2022)

Se o usuário não preencher os campos da maneira correta ou deixar de preenchê-los e tentar, mesmo assim, se cadastrar, será exibido um texto abaixo dos campos incompletos, “Preencha esse campo”, expondo a obrigatoriedade das informações necessárias para manter-se um controle e segurança sobre os registros, e para poder seguir livremente para o site. Também se preenchidos incorretamente, podem ser exibidos textos como “e-mail inválido” ou “A senha não é a mesma”, além das indicações visuais do erro no cadastro – contorno vermelho ao redor do campo incorreto. – Conforme é visto na figura abaixo (Figura 28):

Figura 28 – Cadastro (Indeferido)



A captura de tela mostra um formulário de cadastro com o seguinte layout:

- Logo no topo: um círculo verde com um gato e um cachorro.
- Campos de entrada: Nome, Email, Senha e Confirme sua senha.
- Mensagens de erro: Abaixo de cada campo vazio, o texto "Preencha esse campo" é exibido em uma fonte menor.
- Botão: Um botão verde com o texto "Cadastrar".
- Link: Abaixo do botão, o texto "Já tem uma conta? [Entre Aqui.](#)".

Fonte: Autores, (2022)

Ao preencher corretamente todos os campos com as informações exigidas e clicar em “Cadastrar”, será visualizado um sinal de “correto” nos campos, confirmando que o cadastro foi registrado com sucesso no sistema e assim terá acesso permitido ao site. Essa situação pode ser observada na figura a seguir (Figura 29):

Figura 29 – Cadastro (Deferido)



A imagem mostra um formulário de cadastro com um ícone de perfil no topo. Abaixo do ícone, há quatro campos de entrada de texto, cada um com um ícone de checkmark verde à direita, indicando que os dados foram validados com sucesso. Os campos são: 'Nome' com o valor 'Nicolas', 'Email' com o valor 'nicolasteste@gmail.com', 'Senha' com caracteres ocultos por pontos, e 'Confirme sua senha' também com caracteres ocultos por pontos. Abaixo dos campos, há um botão verde com o texto 'Cadastrar' e um link azul com o texto 'Já tem uma conta? Entre Aqui'.

Fonte: Autores, (2022)

Na programação desse processo foi, em resumo, utilizado o Javascript para fazer uma validação de formulário, ou seja, impedir o usuário de enviar o formulário vazio ou com informações faltando, e para assegurar que as informações estão todas corretas, – senhas correspondendo igualmente, cadastro dentro das normas, – como método de segurança, para o usuário não inserir uma senha que possivelmente esquecerá depois. A ideia, é de validar um formulário em si, como é visto no código a seguir (Figuras 30 a 32):

Figura 30 – Código de validação (Cadastro)

```
validacadcompleto.js
1 var form = document.getElementById('form')
2 const username = document.getElementById('uname')
3 const email = document.getElementById('uemail')
4 const password = document.getElementById('psw')
5 const passwordtwo = document.getElementById('confirmpsw')
6
7 function validaOninput(self, type) {
8     const value = self.value.trim()
9
10    if(value === '' && type == "username" ) {
11        // mostrar erro
12        // add classe
13        setErrorFor(self, 'Preencha esse campo')
14    } else if (type == "username") {
15        // adicionar a classe de sucesso
16        setSuccessFor(self)
17    }
18
19    if(value === '' && type == "emailValue") {
20        // mostrar erro
21        // add classe
22        setErrorFor(self, 'Preencha esse campo')
23    } else if (!isEmail(value) && type == "emailValue") {
24        setErrorFor(self, 'Email inválido')
25    } else if (type == "emailValue") {
26        // adicionar a classe de sucesso
27        setSuccessFor(self)
28    }
29
30    if(value === '' && type == "passwordValue") {
31        // mostrar erro
32        // add classe
33        setErrorFor(self, 'Preencha esse campo')
34    } else if (value.length < 8 && type == "passwordValue") {
35        setErrorFor(self, 'Senha deve ter mais que 8 caracteres')
36    } else if (type == "passwordValue") {
37        // adicionar a classe de sucesso
38        setSuccessFor(self)
39    }
}
```

Fonte: Autores, (2022)

Figura 31 – Código de validação (Cadastro)

```

41  const confpsw = document.querySelector('input#psw')
42  if(value === '' && type == "passwordtwoValue") {
43      // mostrar erro
44      // add classe
45      setErrorFor(self, 'Preencha esse campo')
46
47  } else if(value !== confpsw.value.trim() && type == "passwordtwoValue") {
48      setErrorFor(self, 'Senhas não estão iguais')
49  } else if(type == "passwordtwoValue") {
50      // adicionar a classe de sucesso
51      setSuccessFor(self)
52  }
53  }
54
55  form.addEventListener('submit', (e) => {
56      const send = checkInputs()
57      console.log(send);
58      if(send) {
59          e.preventDefault()
60      }
61  })
62
63  function setErrorFor(input, message) {
64      const formControl = input.parentElement;
65      const small = formControl.querySelector('small')
66
67      small.innerText = message
68
69      formControl.className = 'form-control error';
70  }
71
72  function setSuccessFor(input) {
73      const formControl = input.parentElement;
74      formControl.className = 'form-control success';
75  }
76
77  function isEmail(email) {
78      return /^(?:[a-z0-9!#$%&'*/+=?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*/+=?^_`{|}~-]+)*|
79  }
80

```

Fonte: Autores, (2022)

Figura 32 – Código de validação (Cadastro)

```
81 function checkInputs() {
82
83     var key = true;
84     const usernameValue = username.value.trim()
85     const emailValue = email.value.trim()
86     const passwordValue = password.value.trim()
87     const passwordtwoValue = passwordtwo.value.trim()
88
89     if(usernameValue === '') {
90         // mostrar erro
91         // add classe
92         setErrorFor(username, 'Preencha esse campo')
93         key = false;
94     } else {
95         // adicionar a classe de sucesso
96         setSuccessFor(username)
97     }
98
99     if(emailValue === '') {
100         // mostrar erro
101         // add classe
102         setErrorFor(email, 'Preencha esse campo')
103         key = false;
104     } else if (!isEmail(emailValue)) {
105         setErrorFor(email, 'Email inválido')
106         key = false;
107     } else {
108         // adicionar a classe de sucesso
109         setSuccessFor(email)
110     }
111
112     if(passwordValue === '') {
113         // mostrar erro
114         // add classe
115         setErrorFor(password, 'Preencha esse campo')
116         key = false;
117     } else if(passwordValue.length < 8) {
118         setErrorFor(password, 'Senha deve ter mais que 8 caracteres')
119         key = false;
120     } else {
121         // adicionar a classe de sucesso
122         setSuccessFor(password)
123     }
124
125     if(passwordtwoValue === '') {
126         // mostrar erro
127         // add classe
128         setErrorFor(passwordtwo, 'Preencha esse campo')
129         key = false;
130
131     } else if(passwordValue !== passwordtwoValue) {
132         setErrorFor(passwordtwo, 'Senhas não estão iguais')
133         key = false;
134     } else {
135         // adicionar a classe de sucesso
136         setSuccessFor(passwordtwo)
137     }
138     if (key) {
139         return false;
140     } else {
141         return true;
142     }
143 }
144 }
```

Fonte: Autores, (2022)

3.3.2 – Tela de Login

O usuário que já está cadastrado, deverá clicar no texto, se na tela de cadastro, para direcioná-lo para a tela de login. Nessa tela, existem somente os campos de e-mail e senha, pois outras informações como por exemplo o nome, não são necessárias para se efetuar o login. Após digitar os dados exigidos, é necessário somente clicar no botão “Entrar” para visualizar o site, já sob sua conta.

Se não possuir uma conta, pode-se ser direcionado a tela de cadastro através do texto logo abaixo do botão “Entrar”. Notam-se esses elementos na imagem a seguir (Figura 33):

Figura 33 – Tela de Login

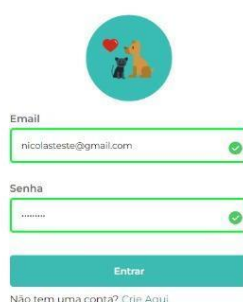


A imagem mostra a interface de login. No topo, há um ícone circular com um coração e um animal. Abaixo dele, há dois campos de entrada: "Email" e "Senha". Abaixo dos campos, há um botão verde com o texto "Entrar". Na base, há o texto "Não tem uma conta? [Crie Aqui.](#)".

Fonte: Autores, (2022)

Assim que se clicar em “Entrar”, se preenchido corretamente, será indicado a validação do login através de elementos visuais como o sinal de “correto” ao lado dos dados e o contorno em verde envolta do campo. Abaixo, essa situação é observada (Figura 34):

Figura 34 – Login (Confirmado)



A imagem mostra um formulário de login com o seguinte conteúdo:

- Logo: Um círculo verde com um ícone de um gato e um cachorro.
- Etiqueta "Email": Um campo de texto contendo "nicolasteste@gmail.com" com um ícone de checkmark verde à direita.
- Etiqueta "Senha": Um campo de texto contendo "....." com um ícone de checkmark verde à direita.
- Botão "Entrar": Um botão retangular verde.
- Link: "Não tem uma conta? [Crie Aqui.](#)"

Fonte: Autores, (2022)

Para o desenvolvimento, essa ideia se trata da função do Javascript, não se trata da tela em si, possuindo objetos, campo input ou botões, na verdade, se trata da função por trás disso. Essa função faz com que quando o usuário clicar, embaixo estará colocado um elemento funcionando como validador automático quando se estiver no formulário, conforme for colocado determinados valores, a validação acontecerá, verificando se aquilo está correto ou incorreto.

É uma função que, continuamente, permanece verificando a function, os if e else, por exemplo, existe o “isEmail” – válido ou não – que se trata de algo predefinido para verificar se os valores inseridos possuem arroba, um gmail, etc. É verificado se está correta a formatação. Com isso, é enviado indicadores visuais, deixando como vermelho, se incorreto, ocorre uma transformação no CSS da página. “setError”, quando colocado, exhibe o erro e define a classe, normalmente, no tocante a questão de deferir os sinais visuais vermelho, para incorreto, e verde, para correto, é colocado “setSuccess”, ao contrário do que é indicado na situação em que “setError” aparece.

Basicamente, ocorre a validação dos valores enviados. Abaixo, encontra-se o código para tal processo (Figuras 35 a 37):

Figura 35 – Código de validação (Login)

```
validalog.js x
1 var form = document.getElementById('form')
2 const email = document.getElementById('uemail')
3 const password = document.getElementById('psw')
4
5 function validaOninput(self, type) {
6     const value = self.value.trim()
7
8     if(value === '' && type == "emailValue") {
9         // mostrar erro
10        // add classe
11        setErrorFor(self, 'Preencha esse campo')
12    } else if (!isEmail(value) && type == "emailValue") {
13        setErrorFor(self, 'Email inválido')
14    } else if (type == "emailValue") {
15        // adicionar a classe de sucesso
16        setSuccessFor(self)
17    }
18
19    if(value === '' && type == "passwordValue") {
20        // mostrar erro
21        // add classe
22        setErrorFor(self, 'Preencha esse campo')
23    } else if (type == "passwordValue") {
24        // adicionar a classe de sucesso
25        setSuccessFor(self)
26    }
27
28 }
29
30
31 function setErrorFor(input, message) {
32     const formControl = input.parentElement;
33     const small = formControl.querySelector('small')
34
35     small.innerText = message
36
37     formControl.className = 'form-control error';
38 }
39
40 function setSuccessFor(input) {
41     const formControl = input.parentElement;
42     console.log("clizou");
43     formControl.className = 'form-control success';
44 }
45
```

Fonte: Autores, (2022)

Figura 36 – Código de validação (Login)

```
46 function isEmail(email) {
47     return /^[^:~!@#$%^&*+\/=?^`{|}~-]+(?:\
48 }
49
50 function setErrorForBut(button, msg) {
51     const butControl = button.parentElement;
52     button.innerText = msg
53
54     butControl.className = 'but-control error';
55 }
56
57
58 function setNormalForBut(button, msg) {
59     const butControl = button.parentElement;
60     button.innerText = msg
61
62     butControl.className = 'but-control';
63 }
```

Fonte: Autores, (2022)

Figura 37 – Código de validação (Login)

```
66 function checkInputs() {
67
68     var key = true;
69     const emailValue = email.value.trim()
70     const passwordValue = password.value.trim()
71
72     if(emailValue === '') {
73         // mostrar erro
74         // add classe
75         setErrorFor(email, 'Preencha esse campo')
76         key = false;
77     } else if (!isEmail(emailValue)) {
78         setErrorFor(email, 'Email inválido')
79         key = false;
80     } else {
81         // adicionar a classe de sucesso
82         setSuccessFor(email)
83     }
84
85     if(passwordValue === '') {
86         // mostrar erro
87         // add classe
88         setErrorFor(password, 'Preencha esse campo')
89         key = false;
90     } else if(passwordValue.length < 8) {
91         setErrorFor(password, 'Senha deve ter mais que 8 caracteres')
92         key = false;
93     } else {
94         // adicionar a classe de sucesso
95         setSuccessFor(password)
96     }
97
98     if (key) {
99         return false;
100     } else {
101         return true;
102     }
103
104 }
```

Fonte: Autores, (2022)

3.3.3 – Menu de Adoção

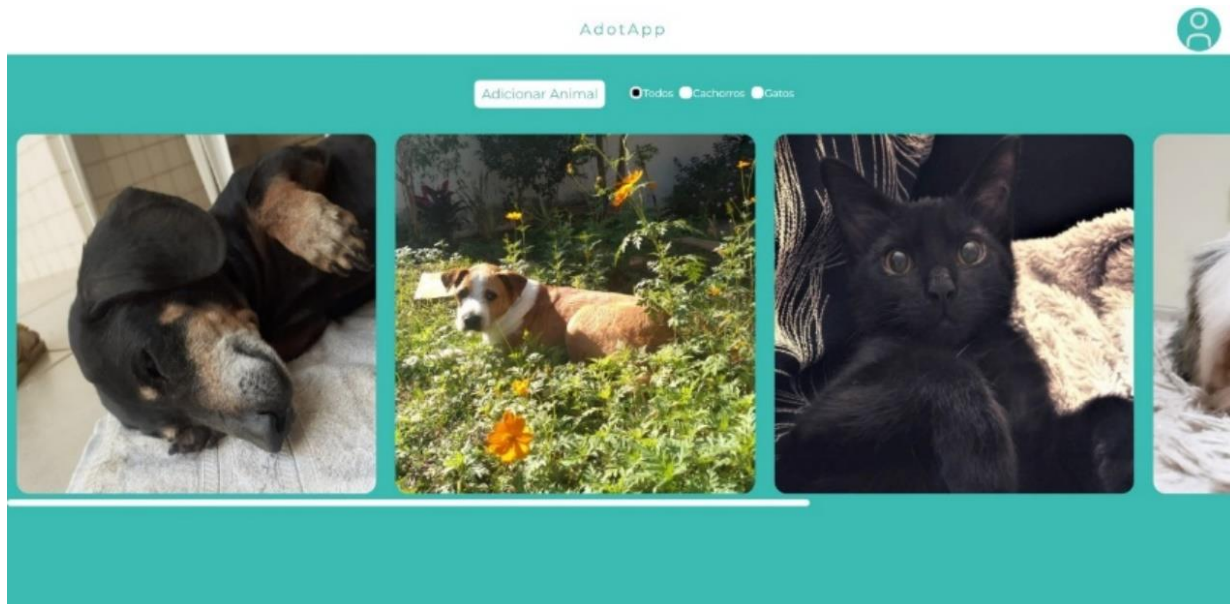
O usuário corretamente identificado terá acesso, ao clicar na aba “Adotar” presente na página inicial do site, à principal ferramenta e proposta do projeto: o menu de adoção, em si. Nele, visualiza-se fotos dos animais colocados para adoção por outros usuários, em uma espécie de feed, que segue à direita conforme o usuário colocar o mouse no lado em questão da tela.

Nessa tela, existem os marcadores de “Cachorros”, “Gatos” e “Todos”, para assim, filtrar os animais exibidos na tela conforme desejar o usuário, a depender de qual o mesmo está interessado em adotar.

O botão “Adicionar Animal” serve, basicamente, para aqueles que desejarem colocar algum animal para adoção. Sua função é simplesmente cadastrar o animal em questão no sistema, para assim ser exibido seu anúncio com sua foto no menu de adoção, para usuários interessados em adotar. Ao clicar no botão de adicionar, o usuário será levado para uma tela onde será requisitado todas as informações de importância que são necessárias para se colocar um animal para adoção com segurança.

O interessado em adotar pode clicar na foto do animal escolhido e assim, seguir para uma tela mais detalhada sobre o animal, com todas suas informações (sexo, idade, se é castrado ou não, etc), contendo, sobretudo, o telefone para contato, fazendo a ponte entre o adotante com o doador. Segue abaixo, imagem do menu de adoção do site (Figura 38):

Figura 38 – Menu de Adoção



Fonte: Autores, (2022)

Após clicar na foto do animal, o usuário é redirecionado para a tela de adoção, propriamente dita, onde ele pode encontrar as informações do animal e o contato com seu doador para poder dar sequência ao processo adotivo. As informações contidas nessa tela são fornecidas pelo próprio doador ao fazer o cadastro do animal na plataforma. Essas informações são obrigatórias para se concluir o cadastro, sendo dados simples e fáceis de serem conseguidos pelo doador. Contém também, nessa tela, o botão de “Adotar Animal”, aonde, ao clicar, o usuário é levado para uma página em uma conversa de WhatsApp, com base no número informado pelo doador, para assim dialogar diretamente com o mesmo. Todos os elementos citados podem ser observados na figura a seguir (Figura 39):

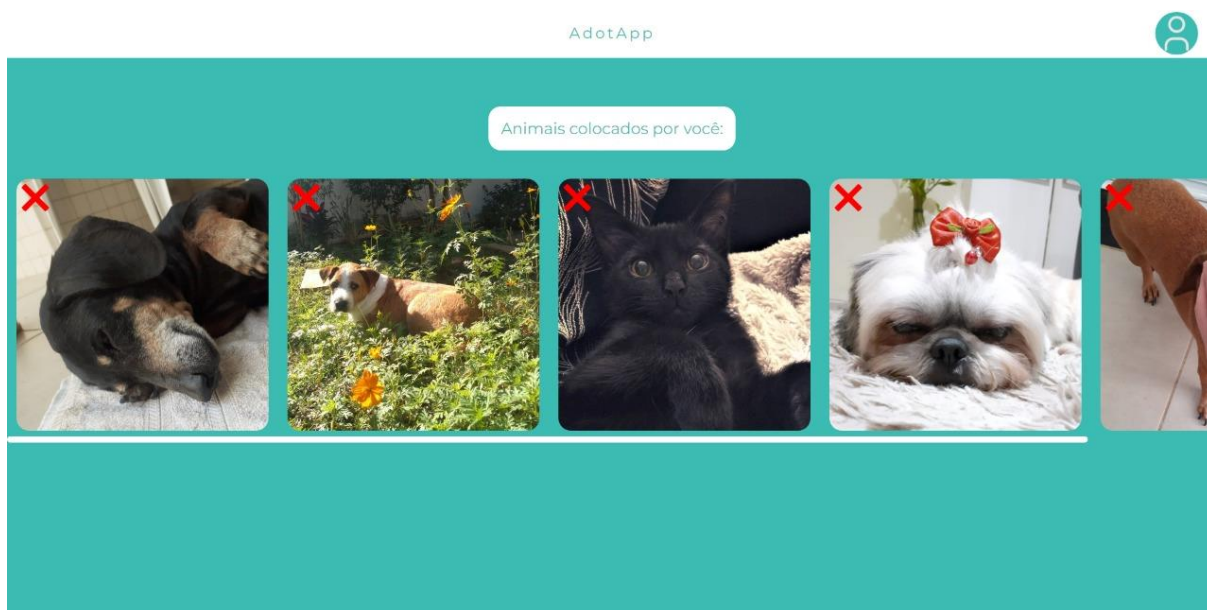
Figura 39 – Tela de Adoção



Fonte: Autores, (2022)

O usuário que possuir anúncios de animais para adoção colocados no site, poderá visualizar uma outra tela, onde lhe é dado um feed semelhante, porém, contendo seus animais que foram colocados para adoção. Nessa tela, o usuário pode remover os animais que já foram ou estão no processo de adoção. A figura, em seguida, ilustra essa tela (Figura 40):

Figura 40 – Anúncios do Usuário



Fonte: Autores, (2022)

Ao clicar em “Adicionar Animal”, no menu de adoção, o usuário é encaminhado para a tela de cadastro do animal, onde pode registrar o animal em questão e assim colocá-lo para adoção. Para tal processo, são requeridas diversas informações como forma de assegurar um processo de adoção seguro e transparente. Informações como: o tipo de animal (podendo selecionar as opções nos marcadores “Cachorro” ou “Gato”), nome, idade, telefone para contato com o doador, as vacinas e condições ligadas a saúde do animal, sexo e também, uma foto do animal para ser exibida no anúncio de adoção do mesmo no menu de adoção. O cadastro de cachorro é visto na imagem abaixo (Figura 41):

Figura 41 – Cadastro de Cachorro

Selecione o Animal:
 Cachorro
 Gato

Nome do Animal:

Idade do Animal:

Telefone para Contato:

Vacinação e Informações Complementares:
 Vacina V8
 Vacina V10
 Vacina V12
 Vermífugado
 Castrado

Sexo:
 Macho
 Fêmea

Coloque uma Foto do Animal:


Cadastrar

Fonte: Autores, (2022)

A tela de cadastro para gatos se trata da mesma tela de cadastro para cachorros. Mudam-se apenas o marcador que ficará selecionado, sendo o tipo de animal que se tratará o anúncio, e as vacinas do animal que são possíveis selecionar. Como é possível ver na imagem a seguir (Figura 42):

Figura 42 – Cadastro de Gato

Selecione o Animal:
 Cachorro
 Gato

Nome do Animal:

Idade do Animal:

Telefone para Contato:

Vacinação e Informações Complementares:
 Vacina V3
 Vacina V4
 Vacina V5
 Vermifugado
 Castrado

Sexo:
 Macho
 Fêmea

Coloque uma Foto do Animal:


Cadastrar

Fonte: Autores, (2022)

Os dados de usuários e de animais colocados para adoção, ficarão gravadas em duas tabelas: “usuarios” e “animais_adocao”, respectivamente. Também existe a tabela “ajuda”, onde é salva as informações enviadas pelo usuário através do formulário de contato, que serve para identificar algum erro ou relatar alguma dúvida. Essas tabelas são vistas na imagem abaixo (Figura 43):

Figura 43 – Tabelas do Banco de Dados



Fonte: Autores, (2022)

Todas as informações inseridas pelo usuário no cadastro do animal serão gravadas no banco de dados na tabela “animais_adocao”. Para assim, poder exibir essas informações na tela de adoção, na qual o usuário interessado em adotar acessa. Abaixo, figura da tabela em questão (Figura 44):

Figura 44 – Tabela de Animais Cadastrados

aid	nome	sexo	telefone_contato	tipo_animal	castrado	vermifugado	idade_animal	vacinav3	vacinav4	vacinav5	vacinav8	vacinav10	vacinav12	foto	donor
1	Belinha	FemSexo	(19) 98185-3251	CachorroCad	Castrado	Vermifugado	13	null	null	null	null	v10	v12	101752652.png	1
2	Nero	MacSexo	(19) 98185-3251	CachorroCad	Castrado	Vermifugado	10	null	null	null	v8	v10	v12	720175115.jpeg	1
3	Sebastião	MacSexo	(19) 98185-3251	GatoCad	null	Vermifugado	4	v3	v4	v5	null	null	null	930654599.jpg	1
4	Piper	FemSexo	(19) 98185-3251	CachorroCad	null	Vermifugado	3	null	null	null	v8	v10	v12	619787263.png	1
5	Mel	FemSexo	(19) 98185-3251	CachorroCad	Castrado	Vermifugado	11	null	null	null	null	v10	v12	904733758.png	1

Fonte: Autores, (2022)

Esses são os códigos em PHP para cadastrar as informações inseridas do animal, pelo usuário, no banco de dados.

Consiste em um arquivo PHP que cadastra essas informações dos animais no banco de dados, basicamente são coletadas as informações enviadas no formulário e encaminhadas para o banco de dados, que as grava na tabela “animais_adocao”.

Também, existe um elemento em que ele coleta a imagem do animal que fora enviada pelo usuário e salva em uma pasta do site chamada “imganimais”. De maneira simplória, é colocado um nome para a imagem em questão e assim, ela é salva no banco de dados. A seguir, os códigos para tal processo (Figuras 45 e 46):

Figura 45 – Código Site Banco de Dados

```
<?php
include_once("conectabd.php");
include('../php/protect.php');

//Verifica se existe foto animal dentro dos arquivos, se não existir ele mata na hora
if (!array_key_exists('fotoanimal', $_FILES)) {
    die("Foto do animal não informada");
}

$filename = $_FILES["fotoanimal"]["name"];
$tmpname = $_FILES["fotoanimal"]["tmp_name"];
$type = $_FILES["fotoanimal"]["type"];
$dot = explode(".", $filename);

//Verifica os arquivos suportados
//&& $type != "image/formato"
if ($type != "image/png" && $type != "image/jpg" && $type != "image/jpeg" && $type != "image/webp") {
    die("Formato de arquivo não suportado");
}

//valida se o arquivo não tem mais de 1 ponto no nome(o windows impede, mas o linux não).
if (count($dot) > 2) {
    die("Nome do arquivo insuportável");
}

$format = "." . $dot[count($dot) - 1];
$newname = rand(100000000, 999999999) . $format; // 100% aleatório, mas tem a chance de sobreporem mas é menos de 0.00001%
$folder = "../imganimais/" . $newname;

if (!rename($tmpname, $folder)) {
    die('Não foi possível salvar o arquivo no servidor');
}

$tipoanimal = filter_input(INPUT_POST, 'tipanimal', FILTER_SANITIZE_EMAIL);
$anome = filter_input(INPUT_POST, 'anome', FILTER_SANITIZE_EMAIL);
$idadeanimal = filter_input(INPUT_POST, 'idadeanimal', FILTER_SANITIZE_EMAIL);
$telefone = filter_input(INPUT_POST, 'telefone', FILTER_DEFAULT);
$vacinav8 = filter_input(INPUT_POST, 'V8', FILTER_SANITIZE_EMAIL);
$vacinav10 = filter_input(INPUT_POST, 'V10', FILTER_SANITIZE_EMAIL);
$vacinav12 = filter_input(INPUT_POST, 'V12', FILTER_SANITIZE_EMAIL);
$vacinav3 = filter_input(INPUT_POST, 'V3', FILTER_SANITIZE_EMAIL);
$vacinav4 = filter_input(INPUT_POST, 'V4', FILTER_SANITIZE_EMAIL);
$vacinav5 = filter_input(INPUT_POST, 'V5', FILTER_SANITIZE_EMAIL);
$vermifugado = filter_input(INPUT_POST, 'Vermifugado', FILTER_SANITIZE_EMAIL);
$castrado = filter_input(INPUT_POST, 'Castrado', FILTER_SANITIZE_EMAIL);
$sexo = filter_input(INPUT_POST, 'animsex', FILTER_SANITIZE_EMAIL);
$dono = $_SESSION['id'];
```

Fonte: Autores, (2022)

Figura 46 – Código Site Banco de Dados

```

$dono = $_SESSION['id'];

if (!isset($vacinav3)) {
    $vacinav3 = "null";
}
if (!isset($vacinav4)) {
    $vacinav4 = "null";
}
if (!isset($vacinav5)) {
    $vacinav5 = "null";
}
if (!isset($vacinav8)) {
    $vacinav8 = "null";
}
if (!isset($vacinav10)) {
    $vacinav10 = "null";
}
if (!isset($vacinav12)) {
    $vacinav12 = "null";
}
if (!isset($castrado)) {
    $castrado = "null";
}
if (!isset($vermifugado)) {
    $vermifugado = "null";
}

$result_animal = "INSERT INTO animais_adocao" .
    "(tipo_animal, nome, idade_animal, telefone_contato, vacinav8," .
    " vacinav10, vacinav12, vacinav3, vacinav4," .
    " vacinav5, vermifugado, castrado, sexo, foto, dono)" .
    " VALUES " .
    "('$tipoanimal', '$anome', '$idadeanimal', '$telefone'," .
    " '$vacinav8', '$vacinav10', '$vacinav12', '$vacinav3'," .
    " '$vacinav4', '$vacinav5', '$vermifugado', '$castrado'," .
    " '$sexo', '$newname', '$dono')";

$resultado_animal = mysqli_query($mysqli, $result_animal);

if (mysqli_insert_id($mysqli)) {
    $_SESSION['msg'] = "<p style='color:green;'>Animal cadastrado com sucesso</p>";
    header("Location: ../painel.php");
} else {
    $_SESSION['msg'] = "<p style='color:red;'>Animal não foi cadastrado com sucesso</p>";
    header("Location: ../index.php");
}

```

Fonte: Autores, (2022)

Os dados fornecidos pelo usuário no cadastro ficaram salvos e criptografados na tabela “usuarios”, a criptografia da senha é essencial por questões de segurança e abaixo é possível ver essa criptografia (Figura 48):

Figura 47 – Criptografia de Senha

id	uname	uemail	psw
1	Admin	admin@admin.com	\$2y\$10\$hC5/UkfFsa0MAygz4xyPDecpcp5MpbRZbejnfAgSioo...

Fonte: Autores, (2022)

Ficarão salvos no banco de dados, na tabela “ajuda”, todas as informações que forem escritas e enviadas pelo usuário para o e-mail de contato. Servindo para relatar algum problema ou dúvida aos desenvolvedores do site. É possível ver essa tabela a seguir (Figura 49):

Figura 48 – Dados da Mensagem de Contato

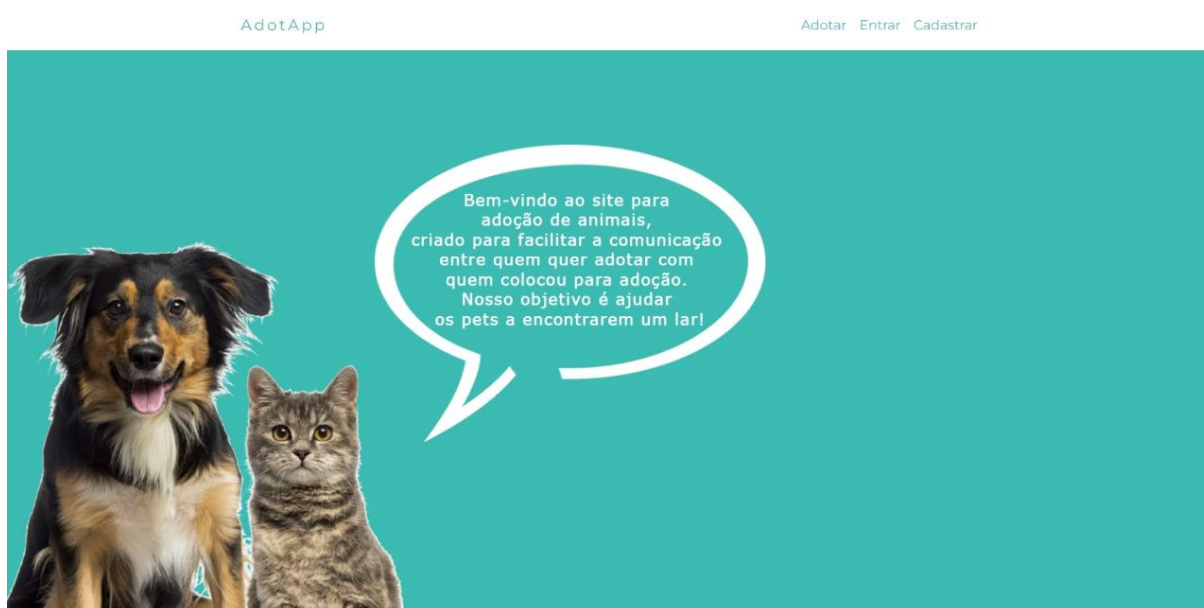
id	name	message	email
1	Admin	Estou tentando entrar em contato para uma adoção m...	admin@admin.com

Fonte: Autores, (2022)

3.3.3 – Página Inicial

Na página inicial do site, logo no centro da tela, contém um balão, nele escrito uma mensagem de boas-vindas e uma breve explicação da ideia do site. Acima, no canto mais à direita, na perspectiva de quem acessa, tem-se três seções: “Adotar” – onde se pode visualizar os animais disponibilizados por outros usuários para adoção, todos possuindo fotos e informações a respeito, – “Entrar” – tela na qual o usuário, já registrado previamente, consegue-se fazer o login informando, corretamente, seu e-mail e senha, – “Cadastrar” – seção em que o usuário que ainda não fora cadastrado, pode se registrar facilmente, efetuando-o de maneira correta seguindo as condições impostas e preenchendo todos os campos obrigatórios. A página inicial do site é visualizada na imagem a seguir (Figura 49):

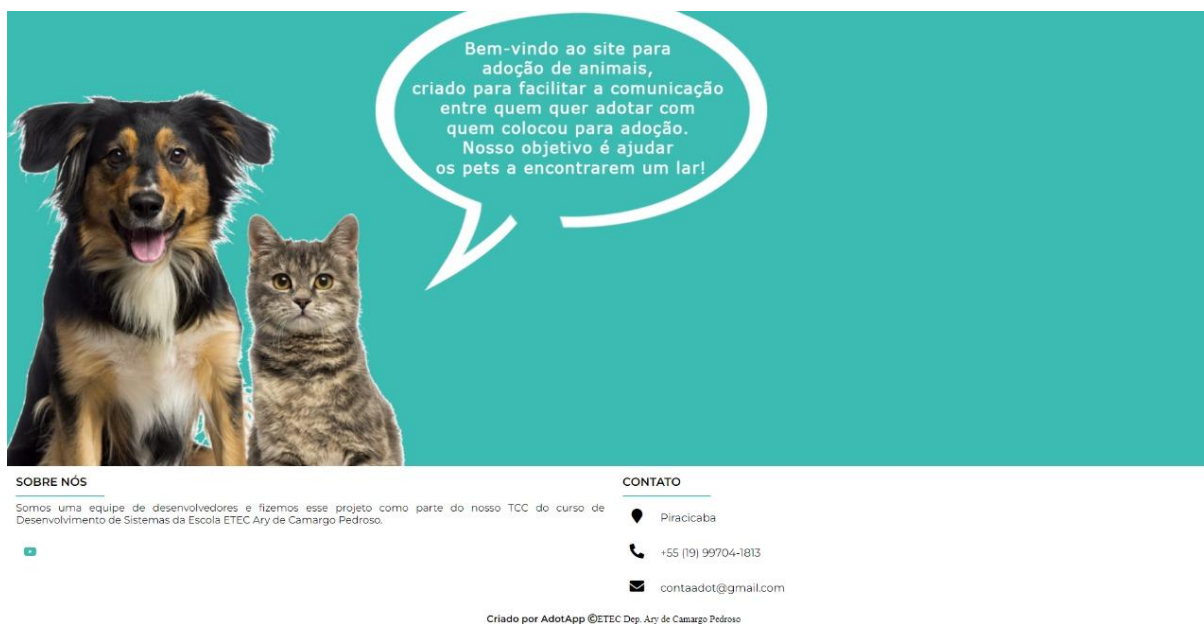
Figura 49 – Página Inicial (Cabeçalho)



Fonte: Autores, (2022)

Ainda na home page do site, mais abaixo na tela, estão informações para o usuário conhecer ainda melhor o projeto. São apresentados o telefone e e-mail para contato, a localização e ícones de redes sociais que, ao clicar, encaminha o usuário para as mesmas. A imagem abaixo destaca essas informações (Figura 50):

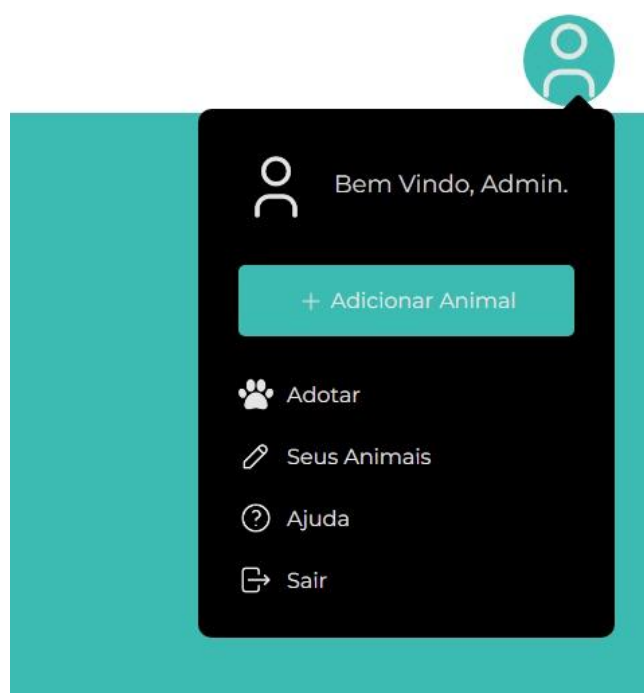
Figura 50 – Página Inicial (Rodapé)



Fonte: Autores, (2022)

Em todas as páginas, no canto superior direito, estará disponível um ícone referente a conta do usuário. Se já cadastrado corretamente, será visível esse ícone, no qual ao usuário clicar nele será aberto um pequeno menu de opções a respeito do login e da conta do usuário. Junto a mensagem de “Bem-vindo”, tem-se o botão de “Adicionar Animal” – para assim ir direto a essa opção –, “Adotar” – para acesso direto ao menu de adoção –, “Seus Animais” – aonde visualiza-se todos os animais que foram colocados para adoção pelo usuário –, “Ajuda” – na qual leva para a tela que possui o formulário de contato, para enviar uma mensagem aos desenvolvedores a respeito de qualquer dúvida ou problema do site –, e a opção “Sair” – para deslogar o usuário de sua conta e assim, sair do site. A imagem abaixo exhibe o menu do usuário (Figura 51):

Figura 51 – Menu do Usuário



Fonte: Autores, (2022)

Se o usuário possuir alguma dúvida ou problema a relatar, ele pode acessar a página de ajuda em que se tem o formulário de contato, onde ele deve inserir seu nome e e-mail para contato, e escrever sua mensagem que será encaminhada à equipe do projeto. Para assim, poder solucionar qualquer seja o problema ou sanar a dúvida em questão. Imagem, a seguir, do formulário de contato (Figura 52):

Figura 52 – Formulário de Contato

A imagem mostra um formulário de contato em um fundo verde-água. No canto superior direito, há um ícone de perfil. O formulário é centralizado e contém o título "Formulário de Contato" em branco. Abaixo do título, há três campos de entrada: "Nome:" com um campo de texto branco; "E-mail para contato:" com um campo de texto branco; e "Mensagem/Dúvida:" com um campo de texto maior e branco. Abaixo dos campos, há um botão verde-água com o texto "ENVIAR" em branco.

Fonte: Autores, (2022)

4 – RESULTADOS OBTIDOS

Os resultados obtidos foram extremamente satisfatórios a todos. Atendendo o que havia sido planejado e projetado para o TCC.

Com a finalização do site, o acabamento e aperfeiçoamento de todas suas páginas e seções, sem dúvidas, o principal objetivo do trabalho foi alcançado. O site, quando funcionamento como normalmente, deverá contar: a página inicial, com informações de importância no rodapé da página a respeito dos idealizadores e do que se trata o projeto, os botões de entrar, cadastrar e adotar onde se dá a principal função do site, que é, de fato conectar usuários interessados em adotar e usuários interessados em colocar um animal para adoção, reunindo então, uma grande comunidade de diversos adotantes e pessoas com animais que necessitam de um lar.

5 – CONCLUSÃO

Devido a grave questão do grande número de animais abandonados no Brasil e seu crescimento nos últimos anos, foi optado por abordar este tema como projeto para o TCC. A finalidade proposta é de se projetar uma plataforma onde seria possível conectar pessoas que acabam por encontrar animais abandonados que precisam de um lar e pessoas interessadas em adotar e cuidar desses animais, um problema, de fato, recorrente e que necessita de atenção. A partir da programação, sobretudo em PHP e CSS, das páginas HTML, através do software Visual Studio Code, decorreu o desenvolvimento do site AdotApp. O site se trata de um ambiente seguro de adoção de animais, onde os usuários, após cadastrados, podem colocar animais para adoção e, também, adotar animais disponibilizados por outros usuários. Sendo uma maneira ordenada de se encontrar um lar para animais abandonados.

Por meio de pesquisas, sobretudo formulários disponibilizados pela plataforma Google Forms, foram obtidas respostas essenciais que ajudaram no desenvolvimento do projeto. Pelos gráficos gerados após o recolhimento das respostas do formulário, foi possível ver com clareza a importância em tratar da problemática de animais abandonados, foi possível ver como grande parte das respostas foram sinalizações positivas, de apoio e solidariedade a causa, serviram de exemplo para demonstrar como existem muitas pessoas preocupadas com esse problema e, também, pessoas interessadas em adotar e ajudar os animais, da maneira com que fosse possível.

Alguns problemas foram enfrentados pelo grupo, que foram resolvidos com o passar do tempo, como por exemplo a dificuldade em se fazer o site, onde foram utilizados informações de sites especializados e buscas na internet para se obter informação necessária para o desenvolvimento do mesmo.

O projeto ajudou no aprendizado de se trabalhar em grupo, se relacionar com profissionais da área em busca de informações e também enfrentar dificuldades já previstas ou até mesmo imprevistas. Os objetivos foram alcançados aos poucos e isso é motivo de felicitação porque com muito trabalho foi possível chegar nessa etapa e finalizar o TCC para futura apresentação.

Onde estão as referências