



UNIVERSIDADE DO VALE DO TAQUARI - UNIVATES
CURSO DE ENGENHARIA DE SOFTWARE

**PET AFFECTION: UM APLICATIVO DE AJUDA PARA OS TUTORES
DE PETS**

João Henrique Nunes de Campos

Lajeado/RS, dezembro de 2023



João Henrique Nunes de Campos

PET AFFECTION: UM APLICATIVO DE AJUDA PARA OS TUTORES DE PETS

Pesquisa apresentada no componente curricular de Trabalho de Conclusão de Curso, na linha de formação específica em Engenharia de Software, da Universidade do Vale do Taquari - Univates, como parte da exigência para obtenção do título de Bacharel em Engenharia de Software.

Orientador: Prof. Me. Luis Antônio Schneiders

Lajeado/RS, dezembro de 2023

João Henrique Nunes de Campos

PET AFFECTION: UM APLICATIVO DE AJUDA PARA OS TUTORES DE PETS

A Banca examinadora abaixo aprova a Monografia apresentada no componente curricular Trabalho de Conclusão de Curso II, do Curso de Engenharia de Software, da Universidade do Vale do Taquari - Univates, como parte da exigência para a obtenção do grau de Bacharel em Engenharia de Software:

Prof. Me. Luis Antônio Schneiders - orientador
Universidade do Vale do Taquari – Univates

Profa. Dra. Maria Claudete Schorr
Universidade do Vale do Taquari - Univates

Prof. Me. Edson Moacir Ahlert
Universidade do Vale do Taquari - Univates

Lajeado/RS, dezembro de 2023

RESUMO

Este trabalho apresenta o **Pet Affection** como uma solução inovadora para auxiliar animais abandonados, integrando tecnologia e preocupação com o bem-estar animal. Desenvolvido como parte do Trabalho de Conclusão de Curso em Engenharia de Software na Universidade do Vale do Taquari - Univates, a pesquisa adotou uma abordagem qualitativa para compreender a interação dos usuários com o aplicativo. Utilizando métodos como entrevistas e observação participante, a pesquisa identificou lacunas em soluções existentes, definiu requisitos cruciais para o **Pet Affection** e considerou normativas e estudos relevantes. A metodologia descritiva foi empregada para validar o aplicativo, utilizando métricas específicas para avaliar sua eficácia. Este estudo contribui significativamente para a engenharia de software aplicada ao bem-estar animal, fornecendo diretrizes claras para o desenvolvimento de uma ferramenta eficiente na abordagem da problemática dos animais em situações de vulnerabilidade nas ruas, incorporando práticas modernas como geolocalização e inteligência artificial.

Palavras-chave: Pet Affection; animais abandonados; tecnologia; engenharia de *software*.

ABSTRACT

This work introduces **Pet Affection** as an innovative solution to assist abandoned animals, integrating technology and concern for animal welfare. Developed as part of the Software Engineering Final Project at the University of Vale do Taquari - Univates, the research adopted a qualitative approach to understand user interaction with the application. Employing methods such as interviews and participant observation, the study identified gaps in existing solutions, defined crucial requirements for **Pet Affection**, and considered relevant regulations and studies. Descriptive methodology was employed to validate the application, using specific metrics to assess its effectiveness. This study significantly contributes to software engineering applied to animal welfare, providing clear guidelines for the development of an efficient tool in addressing the challenges faced by animals in vulnerable situations on the streets, incorporating modern practices such as geolocation and artificial intelligence.

Keywords: Pet Affection, abandoned animals, technology, software engineering.

LISTA DE FIGURAS

Figura 1 – Aplicativo de monitoramento	28
Figura 2 – Recursos Cadastro, Mensagens e Map	30
Figura 3 – <i>Layout do chat</i>	31
Figura 4 – Layout do App	32
Figura 5 – Tela do Login (aplicativo)	45
Figura 6 – Tela de Cadastro e Listagem do Pet (aplicativo).....	46
Figura 7 – Arquitetura MVVM.....	47
Figura 8 – Ações disponíveis ao usuário.....	48
Figura 9 – Estrutura do banco de dados	49
Figura 10 – Implementação da Autenticação e Tela de Login ao Fundo	52
Figura 11 – Implementação da Edição do Mapa e de sua Geolocalização.....	53
Figura 12 – Resultado do SQL dos filtros por todos os <i>pets</i> da região em até 100km	54
Figura 13 – Implementação do Widget da Edição do Mapa.....	55
Figura 14 – Implementação da Inteligência Artificial no App.....	56
Figura 15 – Resultado da Implementação da Inteligência Artificial no App.....	57
Figura 16 – Resultado da Implementação com sucesso.....	58
Figura 17 – Resultado da Implementação com erros.....	60
Figura 18 – Resultado da Implementação corrigida	60
Figura 19 – Resultado da Implementação de status	61
Figura 20 – Mais resultados da implementação.....	62

LISTA DE QUADROS

Quadro 1 – Resumo dos Comparativos entre os Trabalhos Relacionados	34
Quadro 2 – Requisitos Funcionais	50
Quadro 3 – Especificações dos Requisitos Funcionais.....	51
Quadro 4 – Requisitos não funcionais.....	51
Quadro 5 – Especificações dos Requisitos Não Funcionais	51

LISTA DE ABREVIATURAS E SIGLAS

ACID	Atomicidade, Consistência, Isolamento e Durabilidade
BSD	Berkeley Software Distribution
BSD-like	Berkeley Software Distribution like
CFMV	Conselho Federal de Medicina Veterinária
CSS	Cascading Style Sheets
GPS	Sistema de Posicionamento Global
HTML	HyperText Markup Language
IA	Inteligência Artificial
iOS	iPhone Operating System
MVVM	Model, View, ViewModel
OMS	Organização Mundial da Saúde
UI	User Interface
UIPA	Sociedade União Internacional Protetora dos Animais
UX	User Experience
WSPA	World Society for the Protection of Animals

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Problema	11
1.2 Objetivos	12
1.2.1 Objetivo geral	12
1.2.2 Objetivos específicos.....	12
1.3 Estrutura	<u>13</u>
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 Teoria do bem-estar	15
2.2 Teoria da conexão social.....	15
2.3 Teoria da antropomorfização	15
2.4 Teoria da tecnologia móvel	16
2.5 Propósito das teorias.....	17
2.6 Aplicativos para dispositivos móveis.....	18
2.6.1 Desenvolvimento Mobile	21
2.6.2 Geolocalização Via Aplicativo Mobile.....	21
2.6.3 Uso da Inteligência Artificial	22
2.6.4 Uso da IA em Servidor Ubuntu.....	23
2.6.5 Relação seres humanos x animais	24
2.7 A Busca por Animais Perdidos e o Combate ao Abandono no mundo.....	24
2.7.1 Desvelando a Triste Realidade: A Epidemia do Abandono de Animais de Estimação no Brasil e Esforços para uma Mudança	25
3 TRABALHOS RELACIONADOS	27
3.1Elaboração de um sistema de localização geográfica para o acompanhamento de animais de estimação.....	27
3.2 Family Tracker	29
3.3 FINDED: Aplicação mobile para rastreamento geográfico	30
3.4 App AI art generator.....	31
3.5 Development of a Mobile Application for Lost and Found Pets.....	33

3.6 Comparativo entre os trabalhos relacionados	33
4 MATERIAIS E MÉTODOS	35
4.1 Tecnologias	36
4.1.1 <i>Framework</i> Flutter	37
4.1.2 Banco de dados PostgreSQL	38
4.1.3 Google Maps API.....	40
4.1.4 Android Studio	40
4.1.5 Inteligência Artificial: Google Cloud Vision API	43
4.1.6 Rodando o aplicativo em máquina Ubuntu	43
4.2 Desenvolvimento de Aplicativo Móvel para Localização de Animais de Estimação: Detalhes de Implementação e Funcionalidades	44
4.2.1 Desenvolvimento das telas	45
4.2.2 Arquitetura do aplicativo	46
4.2.3 Diagrama de estados	47
4.2.4 Estrutura do Banco de Dados	48
4.2.5 Requisitos funcionais	49
4.2.6 Especificação dos requisitos funcionais	50
4.2.7 Requisitos não funcionais.....	50
4.2.8 Especificação dos requisitos não funcionais.....	51
4.2.9 Fases do Desenvolvimento do Aplicativo Móvel: Detalhes e Progresso na Implementação	51
4.2.10 Implementação de Autenticação Multifacetada e Gerenciamento de Sessões em Aplicativo Móvel: Detalhes e Código Fonte	52
4.2.11 Provedor de localização e mapa	53
4.2.12 Desvendando o Poder da Inteligência Artificial: Explorando a Implementação Flexível e Acessível do Cloud Vision API.....	55
5 Testes e análise dos resultados	58
6 CONSIDERAÇÕES FINAIS	64
REFERÊNCIAS.....	67
APÊNDICE.....	71
APÊNDICE A - Repositório do Projeto.....	72

1 INTRODUÇÃO

No mundo contemporâneo, os animais de estimação, conhecidos como *pets*, tornaram-se membros valiosos e queridos de muitas famílias. Seja um cão brincalhão, um gato carinhoso ou qualquer outro animal de convívio doméstico, esses seres peludos proporcionam alegria, afeto e companhia inestimáveis para seus tutores. No entanto, apesar de todo cuidado e atenção dedicados a eles, há momentos em que eles podem vir a se perder.

A perda de um *pet* é uma experiência angustiante para qualquer tutor. A sensação de vazio e preocupação é avassaladora, pois a incerteza paira no ar. Felizmente, a tecnologia moderna tem se mostrado uma aliada valiosa na busca por animais perdidos. É em função da grande quantidade de animais de estimação perdidos em ruas, becos e locais isolados, que o desenvolvimento de um aplicativo ligado a alguma inteligência artificial visa a aumentar as chances desses animais serem encontrados. Acredita-se que essa tecnologia possa ser útil para auxiliar nos resgates dos perdidos ou abandonados. Dessa forma, este trabalho propõe formas de resolver esse problema.

É importante discutir o tema e conscientizar a população quanto à segurança dos *pets* e sua guarda. Segundo Brito (2019), sobre a prevenção do abandono e suporte à adoção de práticas mais éticas e sustentáveis em relação a eles. Além disso, a autora afirma que os *pets* também são relevantes no contexto da sociedade, como fonte de terapia, ajuda emocional e até mesmo na atuação em áreas como segurança e saúde, como no caso de cães-guia e cães farejadores. Por isso, o tema de *pets* abrange uma ampla variedade de aspectos que merecem ser discutidos e abordados

de forma consciente e responsável.

O aplicativo aqui proposto contará com um sistema de geolocalização que permitirá aos usuários registrar e visualizar a última localização conhecida do animal perdido. Isso possibilitará o compartilhamento de informações valiosas que ajudarão na busca e na recuperação dos animais. O sistema também contará com uma funcionalidade de notificação, para que os usuários sejam alertados quando alguém avistar ou tiver informação relevante sobre seu *pet* perdido.

O uso da inteligência artificial permitirá buscar semelhanças entre as imagens gravadas no banco de dados do aplicativo (a partir de fotos tiradas pelos usuários) e os traços da aparência e da cor do animal encontrado. Com isso, o recurso possui um papel muito importante para o funcionamento do aplicativo.

Em resumo, o Pet Affection poderá ser utilizado como solução tecnológica para auxiliar na busca e recuperação de animais de estimação perdidos em todo o mundo. Ao unir a tecnologia móvel, a colaboração da comunidade e a solidariedade entre tutores de *pets*, espera-se minimizar o sofrimento causado pela perda de um animal de estimação e aumentar as chances de reuni-los com suas famílias. Embora as cidades possuam centros de ajudas, como clínicas veterinárias, elas acabam não contemplando um plano de apoio para os *pets* perdidos em regiões pouco movimentadas.

O objetivo principal desta pesquisa, realizada como parte do Trabalho de Conclusão do curso de Engenharia de Software da Universidade do Vale do Taquari - Univates, visa apresentar um *software* que terá como foco principal a localização e a recuperação de animais perdidos, fornecendo uma plataforma intuitiva para que os usuários possam registrar informações detalhadas sobre seus animais de estimação, como descrições, fotos e outras informações de identificação.

1.1 Problema

A relação entre o problema de *pets* abandonados e um aplicativo que possa ajudá-los é bastante relevante, uma vez que um programa de *software* de localização pode ajudar a criar conexões entre os tutores de animais desaparecidos, permitindo que eles compartilhem informações, relatos de avistamentos e ofereçam apoio mútuo. Essas interações podem fortalecer a rede de solidariedade entre amantes de animais e aumentar as chances de localização e retorno dos *pets* perdidos.

Reconhecer a gravidade do problema dos pets perdidos é crucial. Esses animais de estimação se desgarram de seus tutores e frequentemente enfrentam situações de vulnerabilidade, podendo ser encontrados nas ruas, em abrigos ou em condições precárias. Eles enfrentam riscos como doenças, fome, ferimentos e até mesmo a possibilidade de não retornarem aos seus lares. Nesse contexto, um aplicativo dedicado a auxiliar pets perdidos pode desempenhar um papel crucial na amenização desse problema, contribuindo para a rápida localização e reunião desses animais com seus tutores preocupados.

Desse modo, tem-se como pergunta de pesquisa: "Como o desenvolvimento de um aplicativo móvel pode auxiliar na localização e resgate de *pets* abandonados ou perdidos, e de que forma as funcionalidades e usabilidade desse aplicativo podem ser otimizadas para aumentar sua eficácia e impacto positivo?".

1.2 Objetivos

Os objetivos deste trabalho visam alcançar uma compreensão aprofundada da problemática dos animais de estimação perdidos, bem como desenvolver uma solução para mitigar esse problema.

1.2.1 Objetivo geral

O objetivo geral deste estudo é analisar a situação dos *pets* abandonados ou perdidos, e com isso, apresentar um aplicativo para dispositivos móveis justamente para amenizar estes problemas que causam grandes impactos na sociedade e no bem-estar animal. Ao fazer isso, busca-se fornecer informações fundamentais que possam orientar o desenvolvimento de estratégias mais eficazes e abrangentes para lidar com o desafio dos *pets* abandonados ou perdidos.

1.2.2 Objetivos específicos

Com o intuito de alcançar o objetivo principal, alguns objetivos específicos são necessários, incluindo:

- Estudar e identificar as principais funcionalidades e características que um aplicativo móvel pode ter para abordar efetivamente o problema dos *pets*

abandonados ou perdidos;

- Desenvolver um aplicativo, considerando a usabilidade e a experiência do usuário;
- Testar o aplicativo e dar um *feedback*, e com isso, realizar ajustes necessários no aplicativo.

1.3 Estrutura

Este trabalho está estruturado em seis capítulos que delineiam a jornada da pesquisa. O primeiro capítulo introduz de forma sucinta o tema, destacando a problemática central, a evolução do desenvolvimento, o objetivo geral e os objetivos específicos do estudo. O segundo capítulo se dedica a explorar as referências bibliográficas que serviram de alicerce e inspiração para a condução desta pesquisa.

No terceiro capítulo, é realizada uma revisão bibliográfica abrangente, abordando os estudos correlatos ao presente trabalho. Uma análise minuciosa é empreendida para identificar tanto os pontos distintos quanto os convergentes encontrados.

O quarto capítulo detalha os métodos empregados, as tecnologias adotadas e o desenvolvimento prático, delineando os objetivos específicos e a amplitude alcançada durante o procedimento da pesquisa.

No quinto capítulo, são descritas as atividades conduzidas ao longo do desenvolvimento do trabalho, destacando os testes realizados para validar as abordagens adotadas.

Por fim, o sexto capítulo apresenta as considerações finais, consolidando os resultados obtidos durante a pesquisa, as reflexões oriundas da revisão bibliográfica e as conclusões decorrentes dos experimentos conduzidos. Essa seção encerra o trabalho, proporcionando uma visão integrada da pesquisa desenvolvida.

2 FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica deste trabalho abrange diferentes áreas de conhecimento relacionadas ao desenvolvimento do *software* para auxiliar na localização de animais domésticos perdidos.

A popularidade dos aplicativos voltados aos cuidados de *pets* tem crescido rapidamente nos últimos anos, e há várias teorias que podem ser aplicadas para explicar este fenômeno. A seguir, serão apresentados alguns dos principais conceitos e teorias que embasam esta pesquisa, como bem-estar, conexão social, antropomorfização e tecnologia móvel. Além disso, será explicado como funciona a geolocalização e a inteligência artificial no desenvolvimento *mobile*.

2.1 Teoria do bem-estar

Em sua obra "Libertação Animal", o filósofo Singer (1975), um autor contemporâneo renomado por sua promoção a perspectiva dos direitos dos animais sustenta que os eles têm interesses e capacidade de sofrimento, e, portanto, devem ser incluídos na esfera moral. Ele propõe uma ética baseada na consideração dos interesses de todos os seres sencientes¹, independentemente de serem humanos ou animais. Segundo este filósofo, a presença de animais de estimação tem sido associada a inúmeros benefícios para o bem-estar mental e físico, abrangendo a diminuição do nível de estresse e da ansiedade.

¹ O termo "senciente" refere-se à capacidade de experimentar sensações, sentimentos e consciência. Em outras palavras, ser senciente significa ter uma experiência subjetiva de estar consciente e ter uma vida mental.

2.2 Teoria da conexão social

A teoria da conexão social entre humanos e animais é um tema abordado por vários filósofos e estudiosos, como Regan (1986), um filósofo norte-americano que defendeu a teoria dos direitos animais. Em sua obra "The Case for Animal Rights" (O Argumento para os Direitos Animais), ele argumenta que os animais têm direitos inalienáveis, baseados em sua própria experiência subjetiva de vida. Ele defende que os animais merecem consideração moral e devem ser tratados com respeito e dignidade.

Para Silva (2018), o debate acerca dos direitos dos animais é uma questão importante no pensamento jurídico contemporâneo. Existem diferentes perspectivas teóricas nesse debate, com alguns sendo céticos em relação ao reconhecimento de um estatuto jurídico para os animais não humanos, enquanto outros defendem uma postura mais favorável. O objetivo do estatuto é analisar o fundamento adotado por Nussbaum (2011), que se baseia na fragilidade da vida humana, compartilhada com os outros animais, e na necessidade de valores que conduzam as ações humanas e reduzam a exposição arbitrária ao destino.

Nussbaum (2011) desenvolve sua própria abordagem, chamada de "capabilities approach", por meio de um diálogo envolvente que explora o contratualismo e o utilitarismo. Segundo Bennington (2015), ele menciona o filósofo francês Jacques Derrida, associado ao pós-estruturalismo, que abordou a relação entre humanos e animais em sua obra "A Besta e o Soberano". O filósofo explora as complexidades da fronteira entre humanos e animais, questionando a distinção tradicionalmente estabelecida entre os dois, bem como, desafia as concepções antropocêntricas, além de problematizar as hierarquias de poder e dominação presentes na relação entre humanos e animais.

2.3 Teoria da antropomorfização

Segundo a Anat (2021), o filósofo Wittgenstein investigou a linguagem e o significado da tendência humana de atribuir características a objetos ou animais, o que ficou conhecido como antropomorfização. A argumentação usada é de que a antropomorfização é um resultado da forma como usamos a linguagem e que ela

reflete nossa compreensão limitada do mundo.

A antropomorfização é um fenômeno linguístico que surge quando projetamos nossas categorias e conceitos humanos sobre o mundo. Tal projeção é baseada em nossas próprias experiências e formas de vida, e não necessariamente reflete a realidade dos objetos ou seres não humanos. Portanto, isso contribui para a compreensão da antropomorfização ao explorar as complexidades da linguagem, do significado e das formas de representação que utilizamos para entender e interagir com o mundo. Essa teoria sugere que os tutores tendem a projetar características humanas em seus *pets*, tratando-os como membros da família (Anat, 2021).

2.4 Teoria da tecnologia móvel

De acordo com Ling (2005), a teoria da tecnologia móvel reconhece que os dispositivos móveis têm transformado a forma como nos comunicamos, trabalhamos, nos divertimos e nos adaptamos às circunstâncias do ambiente que nos rodeia. Essas tecnologias proporcionam uma conectividade constante e onipresente, permitindo que as pessoas acessem informações, realizem transações, compartilhem conteúdo e interajam virtualmente em qualquer lugar e a qualquer momento.

Tal teoria explora como os dispositivos móveis afetam os relacionamentos interpessoais, a privacidade, a identidade digital, o consumo de mídia, as práticas culturais, as estruturas de poder e a economia. Também analisa como essas tecnologias influenciam os padrões de comportamento, as dinâmicas sociais e a construção do conhecimento. A teoria da tecnologia móvel destaca a mobilidade como uma característica fundamental dessas tecnologias e argumenta que ela desempenha um papel central na maneira como elas são adotadas, usadas e incorporadas em diversos contextos.

As teorias abordadas anteriormente oferecem perspectivas valiosas sobre diferentes aspectos da sociedade e das interações humanas. Embora cada uma delas se concentre em áreas específicas, compartilham um objetivo comum, sendo ele compreender e analisar as dinâmicas e os impactos das relações humanas e animais, sejam elas sociais, éticas ou tecnológicas. A teoria da conexão social destaca a importância das interações sociais na formação de identidades individuais e coletivas. Ela examina como os relacionamentos e as conexões sociais moldam as experiências, percepções e comportamentos.

Ao entender a dinâmica dessas conexões, podemos desenvolver estratégias para promover relacionamentos saudáveis, comunidades resilientes e uma sociedade mais coesa. A teoria do bem-estar nos leva a refletir sobre como maximizar o bem-estar individual e coletivo. Ela nos desafia a considerar as consequências éticas de nossas ações, buscando não apenas a felicidade pessoal, mas também o florescimento e a satisfação de todos os seres sencientes. Ao adotar uma abordagem centrada no bem-estar, pode-se buscar um equilíbrio entre nossos interesses pessoais e as necessidades do mundo ao nosso redor, levando em consideração as implicações de nossas escolhas para o bem-estar geral. A teoria da tecnologia móvel nos convida a examinar como os dispositivos móveis e a conectividade constante têm transformado nossas vidas. Ela nos ajuda a entender como essas tecnologias influenciam a comunicação, a interação social, a mobilidade e a construção da identidade. Ao considerar o impacto dessas tecnologias em nossa sociedade, podemos tomar decisões informadas sobre seu uso, promover a inclusão digital e garantir que elas sejam uma ferramenta de melhoria e bem-estar, em vez de uma fonte de alienação ou desigualdade.

Em conjunto, essas teorias fornecem uma base sólida para explorar e compreender os aspectos complexos da sociedade contemporânea. Ao considerar as interseções entre a conexão social, o bem-estar humano e a tecnologia móvel, pode-se trabalhar em direção a um futuro em que as relações humanas sejam fortalecidas, o bem-estar seja promovido e a tecnologia seja usada de maneira consciente e responsável.

2.5 Propósito das teorias

Os objetivos dos aplicativos para pets perdidos são ajudar a encontrar animais que se perderam ou fugiram de suas casas, facilitando a busca e a identificação dos *pets* e aumentando as chances de reencontro com seus tutores.

Segundo Anjo (2022), ao fornecer perfis detalhados dos pets abandonados, com fotos, informações sobre sua personalidade e necessidades, o objetivo acaba sendo encontrar um lar amoroso e responsável para eles.

De acordo com a Pilagallo (2021), o propósito de um aplicativo para encontrar *pets* perdidos é oferecer uma plataforma eficiente e abrangente para ajudar a reunir animais perdidos com seus tutores. Um dos objetivos é facilitar a localização e a

recuperação desses *pets*, proporcionando uma ferramenta de busca que conecte tutores, comunidades locais e voluntários. Esses aplicativos visam a oferecer recursos como a criação de perfis dos animais perdidos, incluindo fotos, descrições e informações de contato dos tutores. Permitem, ainda, que os tutores registrem o desaparecimento de seus *pets* e enviem alertas para a rede de usuários, ampliando as chances de encontrar informações relevantes e avistamentos.

Além disso, esses aplicativos podem disponibilizar um mapa interativo com marcações dos locais onde os *pets* foram vistos pela última vez, fornecendo uma visão geográfica das áreas de busca. O propósito final é reduzir o tempo de separação entre os *pets* e seus tutores, minimizando o sofrimento e a angústia de ambas as partes.

O objetivo de um aplicativo para encontrar *pets* abandonados é facilitar a conexão destes com pessoas interessadas em adotá-los. Além disso, busca conscientizar a sociedade sobre a problemática do abandono de animais, promover a adoção responsável e mobilizar recursos para ajudar *pets* em situação de vulnerabilidade.

Em suma, o propósito do aplicativo é usar a tecnologia móvel e a conectividade para unir esforços, solidariedade e colaboração em prol dos animais perdidos, oferecendo esperança e suporte para aqueles que estão desesperadamente buscando seus *pets* desaparecidos.

2.6 Aplicativos para dispositivos móveis

Segundo Gomes (2021), os aplicativos (ou apps) surgiram juntamente com o desenvolvimento dos *smartphones*. O primeiro *smartphone* a oferecer *software* foi o IBM Simon, lançado em 1993, que permitia o uso de calendário, agenda de contatos e notas. No entanto, foi apenas em 2008, com o lançamento do primeiro iPhone da Apple e a abertura da App Store, que eles se popularizaram. A App Store permitiu que desenvolvedores independentes pudessem criar apps para iOS e disponibilizá-los para *download* em uma plataforma centralizada, e assim tornando-os mais acessíveis.

Com a popularização dos *smartphones* e a crescente demanda por *mobile*, outras plataformas de desenvolvimento de sistemas surgiram, como o Google Play para dispositivos Android, e a Apple App Store, para dispositivos iOS. Isso permitiu que o número de *softwares* disponíveis aumentasse significativamente, oferecendo uma grande variedade de funcionalidades e serviços para os usuários. Atualmente,

eles são uma presença constante em quase todos os aspectos do dia a dia, desde jogos e entretenimento até serviços de saúde, finanças, transporte e muito mais. O mercado de programação é um setor em constante expansão, com muitas oportunidades para desenvolvedores criativos e inovadores.

Os aplicativos móveis têm um impacto significativo na sociedade atual. Eles simplificaram muitas atividades, como fazer compras *on-line*, pedir comida, solicitar serviços de transporte e realizar transações bancárias, tornando-as mais convenientes e acessíveis. Além disso, os aplicativos têm permitido a comunicação instantânea e global, conectando pessoas ao redor do mundo e encurtando distâncias.

Também têm exercido uma função crucial em setores como educação, saúde, negócios e governança. Assim, são utilizados para melhorar a aprendizagem, facilitar o acesso a informações médicas, aumentar a eficiência operacional das empresas e fornecer serviços públicos mais eficientes e transparentes.

No futuro, espera-se que os aplicativos móveis continuem evoluindo e se adaptando às necessidades em constante mudança dos usuários. Algumas tendências emergentes incluem a integração de tecnologias avançadas, como realidade virtual e aumentada, inteligência artificial e aprendizado de máquina. Além disso, a integridade e confidencialidade dos dados dos usuários serão áreas de foco cada vez maiores, à medida que a conscientização sobre a proteção de dados pessoais aumenta.

Em síntese, os aplicativos móveis tornaram-se elementos essenciais em nossas vidas, proporcionando conveniência, entretenimento, produtividade e conexão. Com a constante inovação tecnológica, esses aplicativos continuarão desempenhando um papel fundamental em como vivemos, trabalhamos e nos relacionamos uns com os outros.

2.6.1 Desenvolvimento Mobile

Segundo Rosário (2015), o desenvolvimento mobile é o processo de criação e implementação de aplicativos para dispositivos móveis, como *smartphones* e *tablets*. Essa área da tecnologia torna-se cada vez mais crucial com o aumento do uso de dispositivos móveis cresce rapidamente em todo o mundo.

O desenvolvimento *mobile* envolve diversas etapas, desde a concepção da ideia do aplicativo até o lançamento final. É um processo multidisciplinar que requer

conhecimentos em *design* de interface do usuário (UI), experiência do usuário (UX), programação, teste e otimização. Existem diferentes abordagens para o desenvolvimento *mobile*, incluindo o uso de linguagens de programação nativas, multiplataforma ou híbridas.

Para Rosário (2015), as linguagens de programação mais comuns para o desenvolvimento *mobile* nativo são Java e Kotlin para dispositivos Android e Swift e Objective-C para dispositivos iOS. Essas linguagens são utilizadas em conjunto com os *kits* de desenvolvimento de *software* (SDKs) fornecidos pelos fabricantes dos sistemas operacionais, como o Android SDK e o iOS SDK. O desenvolvimento nativo permite um alto nível de desempenho e acesso total aos recursos do dispositivo, mas requer a criação de aplicativos separados para cada plataforma.

Outra abordagem é o desenvolvimento multiplataforma. Segundo Lima (2019), a utilização dos *frameworks* e tecnologias como React Native, Flutter e Xamarin permitem criar aplicativos que podem ser executados em várias plataformas, como Android e iOS, com a reutilização de grande parte do código. Embora ofereçam eficiência no desenvolvimento, a performance pode não ser tão otimizada quanto o desenvolvimento nativo.

O desenvolvimento híbrido combina elementos do desenvolvimento nativo e multiplataforma. Ele utiliza tecnologias como HTML5, CSS e JavaScript para criar aplicativos que são encapsulados em um container nativo. Esses aplicativos são executados em um navegador embutido dentro do aplicativo nativo. O desenvolvimento híbrido oferece um bom equilíbrio entre eficiência de desenvolvimento e desempenho, mas pode ter algumas limitações em termos de acesso a recursos nativos.

Além da escolha da abordagem de desenvolvimento, os desenvolvedores *mobile* também precisam considerar outros aspectos importantes, como a otimização de desempenho, o *design* responsivo para diferentes tamanhos de tela, a integração com serviços e APIs externas, a segurança do aplicativo e a experiência do usuário.

Ao longo do processo de desenvolvimento, os testes são essenciais para garantir a qualidade e o funcionamento correto do aplicativo. Testes de unidade, testes de integração e testes de usabilidade são algumas das práticas comuns no desenvolvimento *mobile*. A implantação do aplicativo nas lojas de aplicativos, como a Google Play Store e a Apple App Store, também faz parte do processo, envolvendo a criação de listagens, a submissão de versões para revisão e a promoção do aplicativo

para os usuários.

O desenvolvimento mobile constitui um procedimento intrincado que abrange diversas fases, incluindo a concepção da ideia, a seleção da abordagem de desenvolvimento, a escolha das linguagens de programação, o *design* da interface do usuário, a implementação, os testes e a implantação nas lojas de aplicativos. É uma área em constante evolução, com a necessidade de uma resposta para se adaptar às mudanças nas tecnologias e nas preferências dos usuários.

2.6.2 Geolocalização Via Aplicativo Mobile

Conforme Frey (2020), a geolocalização via aplicativo *mobile* é uma funcionalidade que permite identificar a localização geográfica de um usuário por meio de um dispositivo móvel, como *smartphone* ou *tablet*. Essa tecnologia utiliza diferentes recursos, como GPS, redes móveis e torres de celular, e Wi-Fi, para determinar a posição do usuário em tempo real.

O GPS (Sistema de Posicionamento Global) é uma das tecnologias mais comuns usadas para geolocalização em dispositivos móveis. Através de sinais de satélite, o GPS calcula a latitude, longitude e altitude do dispositivo, oferecendo uma localização precisa. Isso possibilita o uso de aplicativos de mapeamento, navegação, serviços de entrega e outras aplicações que requerem informações de localização.

Além do GPS, as redes móveis e as torres de celular também podem ser utilizadas para determinar a localização do dispositivo. Por meio da triangulação das torres de celular próximas, é possível estimar a posição aproximada do usuário. Embora seja menos preciso que o GPS, essa técnica ainda pode fornecer informações úteis, especialmente em áreas urbanas densamente povoadas.

O Wi-Fi é outra tecnologia utilizada para geolocalização em aplicativos móveis. Ao identificar redes Wi-Fi próximas e seus respectivos pontos de acesso, é possível determinar a localização aproximada do dispositivo. Isso é especialmente útil em ambientes internos, onde o sinal GPS pode ser menos confiável.

A geolocalização via aplicativo *mobile* oferece uma série de benefícios e aplicações práticas. Por exemplo, aplicativos de transporte, como Uber e Lyft, utilizam a geolocalização para conectar usuários a motoristas próximos. Aplicativos de mapeamento e navegação, como Google Maps e Waze, fornecem direções e orientações precisas com base na localização do usuário. Além disso, muitos

aplicativos de redes sociais, como Instagram e Facebook, permitem que os usuários compartilhem sua localização com amigos e descubram eventos e lugares interessantes nas proximidades.

É importante destacar que a geolocalização via aplicativo *mobile* levanta questões de privacidade. Os usuários devem estar cientes de como seus dados de localização são coletados, armazenados e utilizados pelos aplicativos. As empresas responsáveis pelos aplicativos devem adotar medidas adequadas para proteger a privacidade do usuário e obter o consentimento explícito antes de acessar e utilizar suas informações de localização.

A geolocalização via aplicativo *mobile* utiliza tecnologias como GPS, redes móveis, torres de celular e Wi-Fi para determinar a localização geográfica de um usuário em tempo real. Essa funcionalidade tem uma ampla variedade de aplicações práticas e oferece benefícios significativos, como navegação precisa, serviços baseados em localização e interação social. No entanto, é essencial considerar questões de privacidade ao usar aplicativos que fazem uso da geolocalização.

2.6.3 Uso da Inteligência Artificial

De acordo com as informações apresentadas por Sichman (2021), a inteligência artificial (IA) é considerada um campo da ciência da computação dedicado à criação de sistemas e algoritmos capazes de realizar tarefas frequentemente associadas à inteligência humana. A finalidade principal desses sistemas é adquirir habilidades de aprendizado, raciocínio e tomada de decisões, reconhecer padrões, processar linguagem natural e interagir de maneira dinâmica com o entorno.

Redes neurais artificiais, modeladas a partir da estrutura do cérebro humano, consistem em camadas interconectadas de neurônios. Esses sistemas têm a capacidade de aprender a reconhecer padrões complexos, desempenhando funções como classificação, identificação de imagem e processamento de linguagem natural.

Conforme descrito por Sichman (2021), os sistemas de inteligência artificial têm uma gama extensiva de aplicações em uma variedade de setores, incluindo campos como saúde, finanças, manufatura, transporte, varejo, entretenimento, e muitos outros. Alguns exemplos práticos incluem assistentes virtuais, veículos autônomos, sistemas de recomendação, detecção de fraudes, diagnóstico médico, chatbots e automação de processos.

Apesar das inúmeras vantagens e benefícios proporcionados pela inteligência artificial, ela também suscita questões éticas e desafios. Temas cruciais a serem ponderados incluem questões de privacidade, viés algorítmico, responsabilidade e impacto no mercado de trabalho.

2.6.4 Uso da IA em Servidor Ubuntu

A inteligência artificial é uma disciplina multidisciplinar que procura desenvolver sistemas capazes de simular ou replicar habilidades humanas, como aprendizado, raciocínio e tomada de decisões. Com suas diversas técnicas e aplicações, a IA está transformando muitos aspectos da nossa sociedade e oferecendo soluções inovadoras para problemas complexos.

Segundo a Documentation Ubuntu (2023), o servidor Ubuntu é uma distribuição de sistema operacional baseada em Linux, projetada especificamente para atender às demandas e aos desafios de ambientes de servidor e infraestrutura de rede. Com base na mesma base sólida e confiável do sistema operacional Ubuntu para *desktops*, o Ubuntu Server é otimizado para oferecer desempenho, estabilidade e recursos de gerenciamento necessários para manter serviços críticos funcionando sem interrupções.

Uma das principais características do servidor Ubuntu é a sua estabilidade. Ele é construído sobre uma fundação robusta que inclui kernel Linux, e recebe atualizações regulares de segurança e correções de *bugs* para garantir um ambiente seguro e confiável. Isso é crucial para evitar tempo de inatividade não planejado e manter os serviços disponíveis para os usuários.

A segurança é uma consideração central no desenvolvimento do Ubuntu Server. O sistema inclui um firewall integrada para proteger o tráfego de rede e suporta a implementação de criptografia para proteger dados sensíveis. As atualizações regulares de segurança ajudam a manter o servidor resistente a ameaças emergentes.

O servidor Ubuntu oferece uma plataforma sólida, versátil e segura para hospedar serviços e aplicações em ambientes de servidor. Sua combinação de estabilidade, gerenciamento simplificado, segurança e suporte ativo o torna uma escolha confiável para empresas e profissionais de TI que buscam uma solução completa para suas necessidades de servidor.

2.7 Relação seres humanos x animais

A convivência harmônica entre seres humanos e o reino animal têm sido complexa e variada ao longo da história, com diferentes culturas e sociedades tendo perspectivas diversas sobre sua natureza e relacionamento. No entanto, existem algumas teorias e abordagens amplamente reconhecidas que podem ajudar a explicar.

Segundo os filósofos e estudiosos Singer (1975) e Regan (1986), o antropocentrismo considera os seres humanos como o centro do universo e a espécie mais importante do planeta. Essa visão tende a ver os animais como objetos a serem usados para atender às suas necessidades e desejos, sendo assim, é cada vez mais questionada, especialmente em relação ao tratamento ético.

Essa visão antropocêntrica tem sido desafiada por diversas correntes de pensamento, como o movimento dos direitos dos animais e a ética animal. Os argumentos dos filósofos e estudiosos, dizem que os animais têm seus próprios interesses e merecem consideração moral. Eles propõem uma abordagem baseada nos direitos dos animais, que visa a proteger seus interesses intrínsecos e evitar sua exploração desnecessária.

2.7.1 A Busca por Animais Perdidos e o Combate ao Abandono no mundo

Infelizmente, não há estatísticas precisas sobre o número de animais perdidos, pois muitos casos não são relatados. No entanto, existem várias organizações e iniciativas que se dedicam a ajudar animais perdidos a encontrar seus tutores. Como exemplo, tem-se a *Missing Pet Partnership*², uma organização sem fins lucrativos que disponibiliza recursos e treinamento para auxiliar na localização de animais perdidos. A *Pet FBI*³ é uma organização sem fins lucrativos que disponibiliza um banco de dados nacional de animais perdidos e encontrados, além de recursos para auxiliar na localização e recuperação para encontrar animais desaparecidos, ou seja, o objetivo desse trabalho.

Conforme informações da Organização Mundial da Saúde (OMS, 2022) e do Conselho Federal de Medicina Veterinária (CFMV, 2022), diversas organizações ao

² Estados Unidos - Missing Pet Partnership. Disponível em: <https://www.spcai.org/>.

³ Estados Unidos - Pet FBI. Disponível em: <https://petfbi.org/>.

redor do mundo estão empenhadas em combater o abandono de animais e oferecer assistência aos animais abandonados. Essas organizações trabalham para resgatar, abrigar, tratar e encontrar lares adotivos para esses animais. Além disso, muitas delas realizam campanhas de conscientização, promovendo a importância da adoção responsável, da esterilização e do cuidado adequado com os animais de estimação.

O abandono de animais é um problema global que afeta inúmeras regiões. Segundo a Bússola (2021), milhões de animais são abandonados anualmente, deixados à própria sorte nas ruas, sem abrigo, comida ou cuidados adequados. Esse fenômeno ocorre por várias razões, incluindo falta de responsabilidade dos tutores, dificuldades financeiras, mudanças nas circunstâncias de vida e falta de conscientização sobre a importância da esterilização e adoção responsável.

Em muitos países, especialmente em regiões urbanas densamente povoadas, é comum ver animais abandonados vagando pelas ruas, buscando alimentos e abrigo. Esses animais enfrentam uma série de desafios, incluindo fome, doenças, ataques de outros animais e exposição a condições climáticas extremas. Além disso, segundo Anjo (2022), o abandono de animais contribui para o aumento da população de animais de rua e pode levar ao surgimento de problemas de saúde pública, como a propagação de doenças.

2.7.2 Desvelando a Triste Realidade: A Epidemia do Abandono de Animais de Estimação no Brasil e Esforços para uma Mudança

A problemática do abandono de animais de estimação é grave no Brasil. Segundo dados da Organização Mundial da Saúde (OMS, 2022) e do Conselho Federal de Medicina Veterinária (CFMV, 2022), estima-se que existam aproximadamente 30 milhões de casos de abandono no país, envolvendo tanto cães quanto gatos. O desprezo pode acontecer por diversos motivos, como dificuldades financeiras, mudança de residência, falta de espaço, falta de tempo para cuidar, entre outros. No entanto, é importante lembrar que a deserção é uma prática ilegal e cruel, que pode causar sofrimento e morte.

Para combater o abandono, existem diversas iniciativas e organizações que trabalham para resgatar, cuidar e encontrar novos lares para esses pets. Alguns exemplos são as ONGs de proteção como a Sociedade Mundial de Proteção Animal (WSPA), Ampara Animal, Sociedade União Internacional Protetora dos Animais

(UIPA), entre outras.

Se você se preocupa com o bem-estar e quer ajudar a combater o abandono, considere apoiar uma dessas iniciativas ou adotar um em vez de comprá-lo em uma loja ou criador. Além disso, para a Bússola (2021), é importante incentivar a conscientização sobre a importância da guarda responsável e da castração, para evitar a reprodução descontrolada e o aumento do número de animais abandonados.

3 TRABALHOS RELACIONADOS

Os trabalhos relacionados com *apps* de *pets* englobam diferentes áreas e papéis, desde o desenvolvimento de aplicativos especializados até a prestação de serviços complementares. Profissionais como desenvolvedores de *software*, *designers* de interfaces, especialistas em *marketing* e profissionais de saúde animal têm desempenhado um papel fundamental nesse ecossistema, como revelam os aplicativos e dados a seguir:

3.1 Elaboração de um sistema de localização geográfica para o acompanhamento de animais de estimação

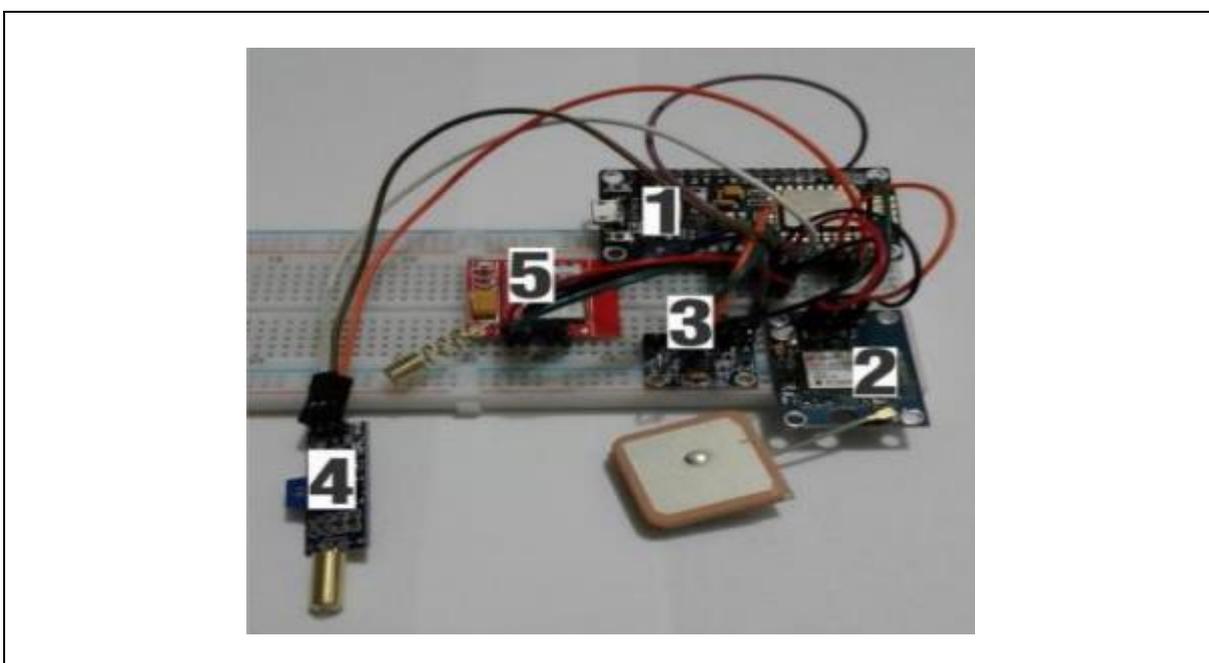
Segundo Silva, Feyh e Roland (2018), o trabalho aborda o desenvolvimento de um sistema de rastreamento de animais de estimação por meio de um aplicativo móvel. O objetivo principal é facilitar a localização dos animais pelos seus proprietários, evitando pesquisas intensivas e eventuais experiências dolorosas. Os donos, cada vez mais preocupados com o bem-estar de seus animais, consideram-nos parte da família.

A evolução da solução ocorre em duas fases, a inicial que envolve a criação de uma coleira especial para os animais, equipada com um dispositivo capaz de adquirir e enviar coordenadas geográficas por meio de conexões Wi-Fi e 3G. A coleira utiliza a placa NodeMCU, que funciona como um escudo para programar o microcontrolador ESP-8266, o qual dispõe módulos Wi-Fi, memória *flash*, sensor de inclinação, acelerômetro com giroscópio, GPS e um módulo de conexão 3G como recursos

principais. A plataforma Arduino é utilizada para controlar os componentes da coleira. Assim que a coleira é ativada, o módulo NodeMCU começa a coletar e enviar os dados.

Na etapa subsequente do desenvolvimento, foi elaborado um aplicativo móvel que utiliza o banco de dados Firebase e o framework Ionic 3 para criar aplicativos híbridos. O processo de desenvolvimento do software incluiu a concepção de diagramas UML, BPMN, análise de casos de uso e a prototipagem das interfaces. Ao iniciar o aplicativo, o usuário é solicitado a efetuar o login ou se cadastrar. Após verificar a existência do registro no banco de dados, o aplicativo realiza o cadastro, se necessário. Após o login, o usuário é redirecionado para a tela principal chamada "Home", onde pode visualizar o Google Maps mostrando a localização das coleiras registradas. Além disso, o aplicativo possibilita o cadastro de animais e seus respectivos números de coleira, conforme demonstrado no modelo de imagem abaixo.

Figura 1 – Aplicativo de monitoramento



Fonte: Adaptado pelo autor com base no Frey (2020).

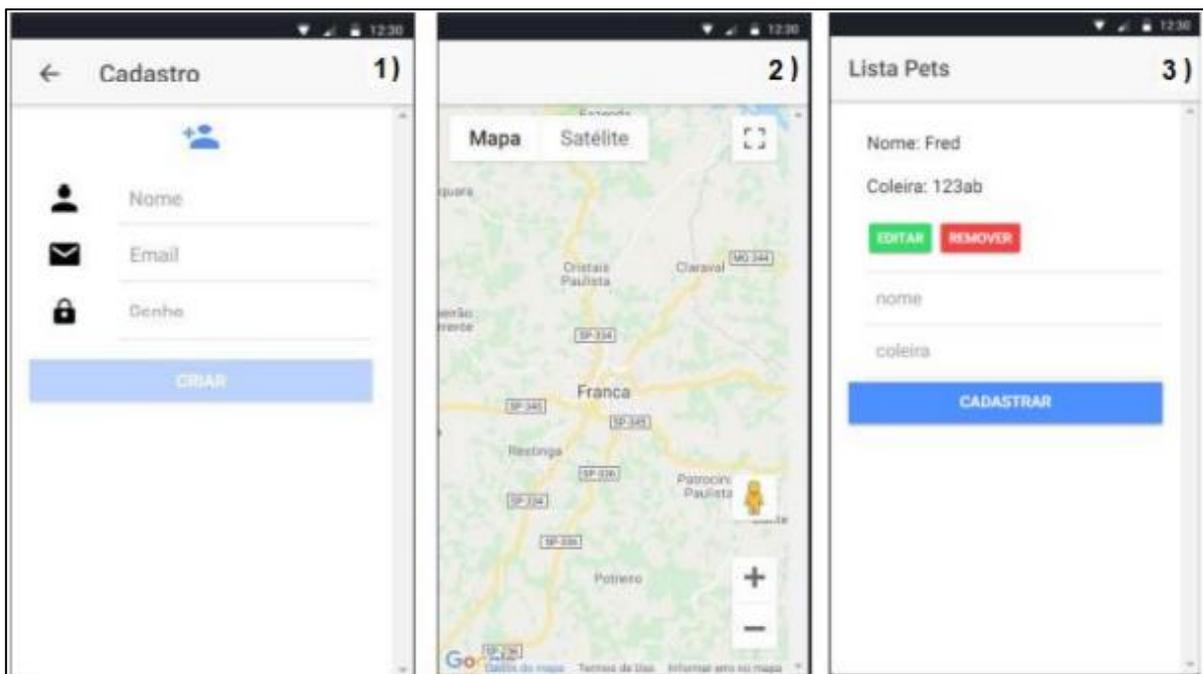
Dessa forma, o sistema de rastreamento de animais de estimação consiste em uma coleira com componentes eletrônicos e um aplicativo móvel para facilitar a localização dos animais, proporcionando mais segurança e tranquilidade aos seus proprietários.

3.2 Family Tracker

Conforme as informações fornecidas no website oficial, o Family Tracker4 trata-se de uma aplicação de rastreamento por GPS disponível para dispositivos Android e iOS mediante pagamento. O aplicativo utiliza diversas tecnologias de localização, como GPS e conexões via Wi-Fi. Com um registro de mais de 10 mil downloads, oferece aos usuários a capacidade de rastrear um número ilimitado de dispositivos. Destaca-se a funcionalidade de troca de mensagens de texto privadas entre usuários por meio do aplicativo, conforme indicado no item 1 da Figura 2. Os Breadcrumbs (migalhas), que registram dados e rotas do dispositivo, permitem ao usuário armazenar ou compartilhar essas informações na versão paga. O item 2 da Figura 2 exemplifica o percurso realizado pelo usuário, enquanto o Geofencing, responsável por enviar notificações quando um dispositivo entra ou sai de uma área geográfica específica, é visualizado no item 3 da Figura 2. É relevante destacar que essa funcionalidade está acessível somente nas versões pagas e em dispositivos que possuam iOS 5 ou posterior, assim como Android 4.4 ou versões mais recentes.

O website oficial do Family Tracker destaca, ainda, que o aplicativo não funciona continuamente em segundo plano, proporcionando, assim, uma utilização mais eficiente da bateria do dispositivo. Além das funcionalidades móveis, o Family Tracker possibilita o acesso via uma interface web para rastrear os trajetos de dispositivos específicos por meio de um código de acesso, conforme exemplificado na Figura 2.

Figura 2 – Recursos Cadastro, Mensagens e Map



Fonte: Adaptado pelo autor com base no Frey (2020).

Esses aplicativos exemplificam o avanço da tecnologia e da digitalização, trazendo benefícios como conveniência, disponibilidade de uma diversidade de produtos e serviços, e oferecendo soluções práticas para os desafios enfrentados pelos proprietários de animais de estimação. Com essas ferramentas, é possível garantir que os *pets* recebam os cuidados necessários, sejam bem atendidos e desfrutem de uma vida saudável e feliz.

3.3 FINDED: Aplicação mobile para rastreamento

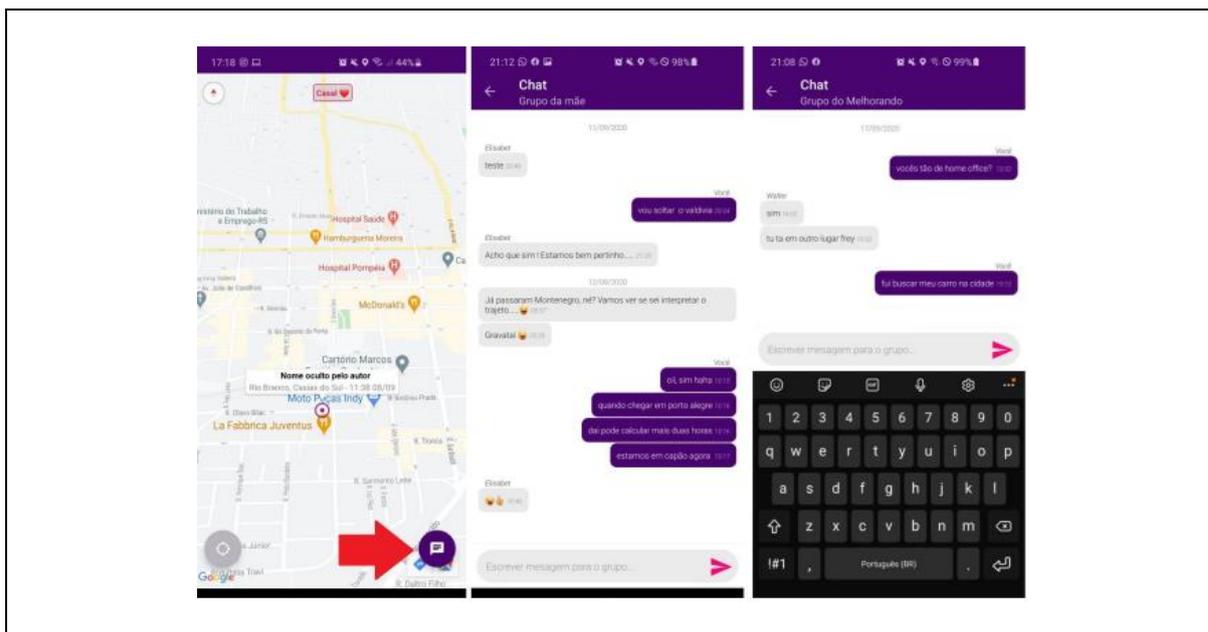
Conforme Frey (2020), o aplicativo permite que os usuários compartilhem sua localização em tempo real, que é projetada em um mapa virtual dentro do aplicativo. Dessa forma, é possível visualizar a geolocalização dos membros do grupo de forma instantânea e precisa. Além disso, o aplicativo inclui um recurso de chat *on-line*, que facilita a comunicação entre os usuários. Com essa opção de troca de mensagens de texto, é possível agilizar a comunicação e evitar mal-entendidos na descrição de localizações.

O trabalho realizado envolveu uma investigação detalhada de técnicas de localização, a plataforma Android e tecnologias voltadas para dispositivos móveis. Adicionalmente, foi conduzida uma pesquisa sobre os aplicativos de geolocalização

disponíveis no mercado e estudos correlatos ao tema, buscando aprimorar a solução proposta.

O processo de desenvolvimento do aplicativo envolveu a elaboração de uma arquitetura eficiente, a implementação de funcionalidades de geolocalização e *chat*, e a validação do aplicativo por meio de testes e avaliações. Os resultados obtidos confirmaram o desempenho da solução em aproximar usuários e simplificar tarefas cotidianas relacionadas à localização, conforme ilustrado abaixo.

Figura 3 – *Layout do chat*



Fonte: Adaptado pelo autor com base em Frey (2020).

Em suma, o aplicativo *mobile* desenvolvido para Android proporciona uma forma rápida e eficaz de compartilhar informações de localização entre usuários, utilizando recursos de geolocalização e um *chat on-line*. O trabalho apresenta um estudo aprofundado, bem como o processo de elaboração, implementação e validação do aplicativo, demonstrando sua eficiência em facilitar a comunicação e agilizar tarefas relacionadas à localização, e baseado na sua geolocalização o aplicativo proposto do trabalho acaba se baseando nele.

3.4 App AI art generator

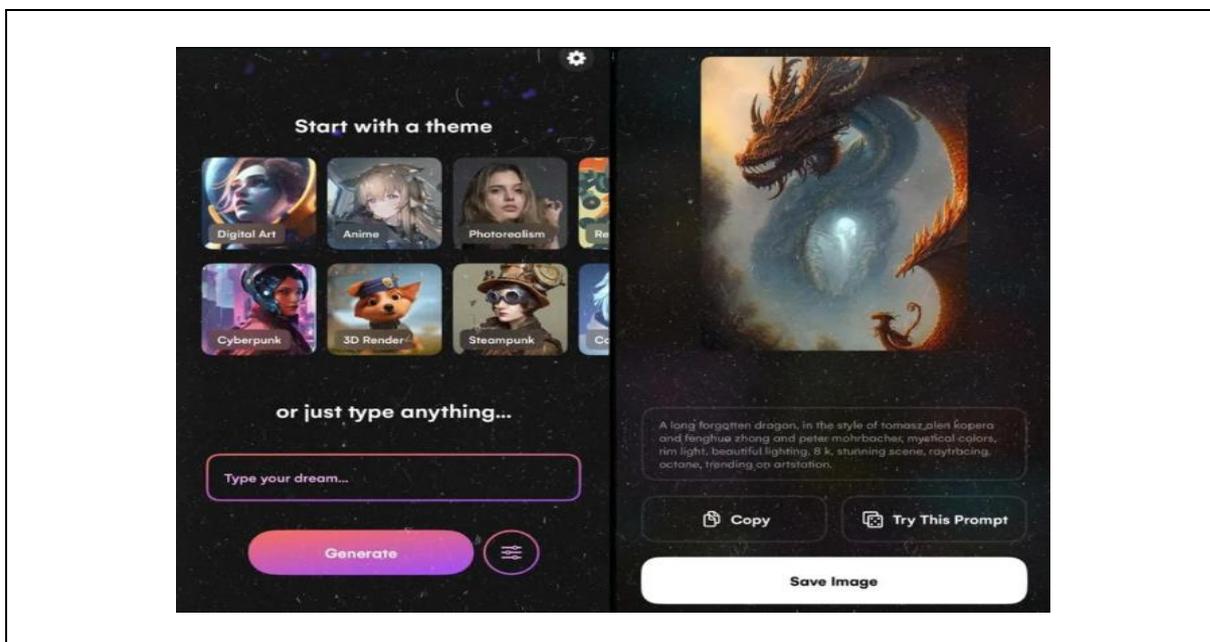
Conforme Souza (2023), o AI Art Generator é um aplicativo inovador que funciona como um gerador de imagens baseado na inteligência artificial. Disponível para ambos os sistemas operacionais, ele permite que os usuários criem obras de

arte com base na imaginação, utilizando temas sugeridos pela própria IA do aplicativo ou *prompts* textuais.

A ferramenta é capaz de produzir uma ampla variedade de estilos artísticos, desde abstratos até ultrarrealistas, de forma simplificada. Os usuários podem explorar temas como anime, arte digital, 3D, retrô, fantasia, futurismo e muito mais, proporcionando uma experiência artística diversificada.

Uma das características notáveis do AI Art Generator é a sua capacidade de carregar arquivos do próprio dispositivo, permitindo que os usuários utilizem suas próprias imagens como base para a criação de novas obras de arte. Essa funcionalidade está disponível exclusivamente no plano vitalício do aplicativo, conforme a imagem abaixo.

Figura 4 – Layout do App



Fonte: Adaptado pelo autor com base em Souza (2023).

Com o AI Art Generator, os usuários podem liberar sua criatividade e explorar diferentes estilos artísticos de forma simples e intuitiva, aproveitando os recursos avançados de inteligência artificial para transformar suas ideias em belas obras de arte, e baseado em sua I.A por seleção de imagens por cores que se assemelha com o aplicativo desenvolvido.

3.5 Development of a Mobile Application for Lost and Found Pets

De acordo com Salcedo, Ortiz e Espitia (2018), o **Development of a Mobile Application for Lost and Found Pets** explora a criação de um aplicativo móvel destinado a auxiliar na localização de animais de estimação perdidos. O estudo aborda a análise de requisitos, o *design* da interface e a implementação do aplicativo, com o objetivo de fornecer uma solução prática e eficiente para reunir *pets* perdidos com seus proprietários.

O objetivo principal do aplicativo é facilitar a busca por animais de estimação perdidos e aumentar as chances de encontrá-los, permitindo que os usuários relatem animais perdidos e encontrados e se comuniquem entre si para facilitar a reunificação dos animais com seus donos.

Possui um registro de animais perdidos ou encontrados onde os usuários poderão criar perfis detalhados para seus animais de estimação perdidos ou relatar animais que encontraram. Isso incluirá informações como descrição do animal (espécie, raça, idade, tamanho, cor, características distintivas), local e data do incidente, além de fotos.

O aplicativo permite que os usuários pesquisem animais perdidos ou encontrados com base em diferentes critérios, como localização geográfica, raça, cor, idade, entre outros. Recursos de filtragem avançada também podem ser adicionados para refinar ainda mais os resultados da pesquisa.

Esses são apenas alguns detalhes sobre o trabalho envolvido no desenvolvimento de um aplicativo móvel para perdidos e achados de animais de estimação. O processo de desenvolvimento pode variar dependendo dos requisitos específicos do projeto, recursos desejados e tecnologias escolhidas.

3.6 Comparativo entre os trabalhos relacionados

Os estudos acadêmicos têm desempenhado um papel fundamental na compreensão e no aprimoramento do setor de *pet shops*. Diversos pesquisadores têm se dedicado a investigar diferentes aspectos relacionados a esse mercado em constante crescimento, visando fornecer *insights* valiosos e informações relevantes para aprimorar a qualidade dos serviços e produtos oferecidos, bem como entender as demandas e as preferências dos consumidores.

Por meio do Quadro 1, é possível realizar uma análise comparativa entre os elementos essenciais dos estudos relacionados e a proposta do presente trabalho.

Quadro 1 – Resumo dos Comparativos entre os Trabalhos Relacionados

Característica	Trab. 3.1	Trab. 3.2	Trab. 3.3	Trab. 3.4	Trab 3.5	Proposta
Sistema operacional	Android e iOS	Android, iOS e Web	Android	Android e iOS	Android	Android
Contém Chat	Sim	Sim	Sim	Não	Não	Não
Pago	Sim	Sim	Sim	Não	Não	Não

Fonte: Do autor (2023).

Observa-se que, apesar da existência de uma ampla gama de aplicações de rastreamento disponíveis no mercado móvel e com essa tabela, podemos ver características diferentes, como por exemplo uso da I.A e da geolocalização, sendo um aplicativo não pago que não contém chat, justamente por ele ser mais objetivo e intuitivo ao usuário para chegar logo no seu objetivo.

4 MATERIAIS E MÉTODOS

Os aplicativos de procura de *pets* perdidos geralmente utilizam uma combinação de diferentes métodos para ajudar a localizar animais de estimação. Alguns dos métodos mais comuns utilizados nesses aplicativos incluem os cadastros de informações dos tutores dos *pets*, que podem fornecer informações detalhadas sobre eles, como nome, raça, idade, características físicas, fotos, informações de contato e local onde foram vistos pela última vez.

Os *apps* geralmente usam recursos de geolocalização para permitir que os tutores dos *pets* marquem no mapa o local em que o animal foi visto pela última vez. Isso ajuda a criar uma referência espacial para os usuários e facilita a busca. Com isso, o envio de notificações e alertas para os usuários torna-se essencial, pois alertará em áreas próximas os *pets* que foram perdidos, permitindo que outras pessoas estejam cientes do desaparecimento e possam ajudar na busca.

A pesquisa realizada neste trabalho utiliza uma abordagem qualitativa. Segundo Prodanov e Freitas (2013), essa abordagem metodológica é notável por sua simplicidade e pela abertura à interpretação e análise por parte do autor em relação aos dados coletados. Diferentemente da pesquisa quantitativa, a pesquisa qualitativa não se baseia na mensuração de aspectos através de estatísticas e números, mas sim na compreensão dos significados, experiências e percepções dos participantes envolvidos no estudo. Na pesquisa qualitativa, o foco está na coleta de dados descritivos e detalhados, buscando entender os fenômenos estudados em sua complexidade e profundidade. Para isso, utiliza métodos como entrevistas, observação participante, exame de documentos e análise de conteúdo. Seu objetivo

é compreender os significados e as experiências dos participantes, possibilitando uma compreensão mais aprofundada dos fenômenos estudados.

Nesse contexto, a atual pesquisa se enquadra no domínio qualitativo, uma vez que o aplicativo passa por validação, e sua qualidade é avaliada por meio de métricas específicas.

Quanto à natureza da pesquisa, este trabalho possui um caráter descritivo. Conforme Prodanov e Freitas (2013), a pesquisa descritiva é um tipo de pesquisa que busca descrever fenômenos, características ou experiências relacionadas a um estudo específico. Nessa abordagem, o objetivo principal é obter uma visão clara e detalhada do objeto de pesquisa, identificando suas características e qualidades. A pesquisa descritiva é uma abordagem que busca descrever fenômenos, características ou experiências relacionadas a um estudo específico. Seu propósito é obter uma visão clara e detalhada do objeto de pesquisa, identificando suas características, relações e manifestações. Utiliza técnicas de coleta de dados para descrever e documentar aspectos relevantes do fenômeno, permitindo a análise de variáveis, a identificação de padrões e a construção de perfis. A pesquisa descritiva contribui para a compreensão do fenômeno e auxilia na tomada de decisões. Nesse contexto, durante o desenvolvimento do projeto, a pesquisa descritiva está ligada ao conjunto de informações coletadas e registradas, de acordo com Prodanov e Freitas (2013).

No que concerne às metodologias utilizadas, este estudo se baseia em pesquisa bibliográfica e experimental. A pesquisa bibliográfica visa familiarizar o pesquisador com o material relacionado ao tema de estudo, enquanto a pesquisa experimental envolve a observação do comportamento de um objeto de estudo sob a influência de variáveis manipuladas, com o intuito de testar hipóteses e identificar relações de causa e efeito, conforme delineado por Silveira (2019).

A revisão bibliográfica foi empregada para fundamentar teoricamente o trabalho, abordando os principais tópicos pertinentes, como os sistemas de localização, a estrutura da plataforma Android e a inteligência artificial.

4.1 Tecnologias

A seguir, serão apresentadas as tecnologias empregadas na construção do aplicativo proposto.

4.1.1 Framework Flutter

Segundo Corazza (2018), o *framework* Flutter, distribuído sob a Licença BSD modificada⁴, oferece uma ampla gama de recursos e bibliotecas que facilitam a criação de interfaces de usuário atraentes e responsivas. Ele também possibilita a construção de apps nativos compatíveis com iOS e Android a partir de um único código-fonte, utilizando a linguagem de programação Dart.

Com isso, justifica-se a escolha, sendo ela o desenvolvimento multiplataforma, pois sua principal vantagem é a capacidade de criar aplicativos nativos para iOS e Android a partir de um único código-base. Isso reduz significativamente o esforço contínuo e o tempo necessário para planejar e implementar melhorias e manter aplicativos em diferentes plataformas, proporcionando maior eficiência e produtividade no desenvolvimento do aplicativo.

De acordo com Neves e Junior (2021), o Flutter é um framework de desenvolvimento de aplicativos móveis de código aberto desenvolvido pelo Google. Esse framework possibilita aos desenvolvedores criarem aplicativos nativos para iOS e Android a partir de um único código-fonte, resultando em uma considerável economia de tempo e recursos.

A arquitetura do Flutter é baseada em *widgets*, que são os blocos de construção fundamentais dos aplicativos. Os *widgets* são altamente personalizáveis e permitem a criação de interfaces de usuário atraentes e responsivas. O Flutter possui uma vasta biblioteca de *widgets* prontos para uso, incluindo widgets para texto, botões, imagens, listas e muito mais.

Uma característica distintiva do Flutter é o seu mecanismo de renderização próprio, chamado Skia. Esse mecanismo permite que os aplicativos Flutter ofereçam desempenho de alta qualidade e visualizações suaves.

Além disso, o Flutter oferece uma ampla gama de recursos e APIs para acesso a recursos do dispositivo, como câmera, geolocalização, sensores e armazenamento. Os desenvolvedores também podem integrar facilmente serviços em nuvem, bancos de dados e outras APIs de terceiros aos seus aplicativos.

O Flutter utiliza a linguagem de programação Dart, que é fácil de aprender e

⁴ Essa licença é conhecida por ser permissiva e oferecer liberdade aos desenvolvedores para modificar.

possui recursos modernos, como tipagem estática e coleta de lixo. Essa linguagem combinada com o Hot Reload, uma funcionalidade do Flutter, permite que os desenvolvedores façam alterações no código em tempo real e vejam imediatamente os resultados, agilizando o processo de desenvolvimento e teste.

Esse framework é uma plataforma poderosa para o desenvolvimento de aplicativos móveis nativos, permitindo que os desenvolvedores criem interfaces de usuário incríveis, ofereçam desempenho de alta qualidade e acessem facilmente recursos do dispositivo. Com sua abordagem de desenvolvimento rápido e eficiente, o Flutter se tornou uma escolha popular entre os desenvolvedores para a criação de aplicativos móveis modernos e elegantes.

Com o Flutter, é possível desenvolver aplicativos com aparência nativa⁵, garantindo uma experiência de usuário de alta qualidade. Esse *framework* possui uma comunidade de desenvolvedores ativa e crescente. Isso significa que há uma vasta quantidade de recursos, tutoriais, bibliotecas e soluções disponíveis para ajudar no desenvolvimento do aplicativo.

4.1.2 Banco de dados PostgreSQL

A escolha do banco de dados PostgreSQL, distribuído sob a licença BSD-like, baseia-se no fato de ser conhecido por sua capacidade de integração com diversas tecnologias. Ele possui suporte para vários formatos de dados, como JSON, e oferece extensibilidade para adicionar novas funcionalidades personalizadas. Além disso, existem bibliotecas e pacotes disponíveis para o Flutter que facilitam a interação, tornando o processo de desenvolvimento mais eficiente.

Segundo Carlos (2007), o PostgreSQL é conhecido por sua confiabilidade e estabilidade, o que reforça a justificativa da escolha. Ele é projetado para lidar com cargas de trabalho pesadas e oferece um desempenho sólido mesmo em bancos de dados grandes. Isso o torna adequado para aplicativos Flutter que exigem um banco de dados robusto e rápido.

Seus recursos avançados acabam oferecendo uma ampla gama de funcionalidades, como suporte a transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade), chaves estrangeiras, consultas complexas, funções

⁵ Interfaces de usuário que se assemelham visualmente e comportamentalmente às interfaces nativas de cada plataforma específica.

armazenadas e muito mais. Esses recursos podem ser aproveitados para criar aplicativos Flutter sofisticados que exigem manipulação de dados complexa.

Conforme mencionado por Costa (2013), o PostgreSQL é um sistema de gerenciamento de banco de dados que segue o modelo relacional (SGBDR) de código aberto, reconhecido por sua confiabilidade, flexibilidade e recursos avançados. Desenvolvido para atender a uma variedade de requisitos de armazenamento e manipulação de dados em aplicações de grande porte.

O PostgreSQL suporta a linguagem SQL padrão e proporciona uma extensa gama de funcionalidades, incluindo suporte a transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade), controle de concorrência, indexação avançada, gatilhos, procedimentos armazenados, replicação e extensibilidade. Isso torna o PostgreSQL adequado para uma ampla variedade de aplicações, desde pequenos projetos até sistemas empresariais complexos.

Além disso, o PostgreSQL é altamente customizável e permite aos desenvolvedores estenderem suas funcionalidades por meio de linguagens de programação como PL/pgSQL, PL/Python, PL/Perl, entre outras. Isso possibilita a criação de funções personalizadas e procedimentos armazenados complexos dentro do próprio banco de dados.

O desempenho do PostgreSQL também é notável, com otimizações avançadas para consultas complexas e grandes volumes de dados. Ele oferece recursos como índices avançados, otimização de consultas, paralelismo e suporte a dados geoespaciais.

Outra vantagem do PostgreSQL é sua comunidade ativa e suporte contínuo. A comunidade PostgreSQL é composta por desenvolvedores e usuários dedicados, que contribuem com melhorias, correções de *bugs* e suporte através de fóruns, grupos de discussão e recursos online.

O PostgreSQL é uma robusta plataforma de gerenciamento de banco de dados relacional, conhecido por seus recursos avançados, alta customização e desempenho otimizado. Sua extensa lista de funcionalidades, estabilidade e suporte ativo o tornam uma escolha popular para aplicações que demandam armazenamento e manipulação de dados confiáveis e eficientes.

4.1.3 Google Maps API

O Google Maps API é uma plataforma disponibilizada pelo Google que permite que os desenvolvedores incorporem recursos avançados de mapas interativos em seus aplicativos e *sites*. Essa API oferece um conjunto abrangente de serviços e funcionalidades para visualização de mapas, obtenção de informações geográficas, cálculos de rotas, pesquisa de lugares, entre outros recursos relacionados a mapas. Através do Google Maps API, os desenvolvedores podem integrar mapas interativos em suas aplicações, permitindo que os usuários visualizem e interajam com informações geográficas de maneira intuitiva. Eles podem exibir mapas personalizados, marcadores, polígonos e linhas, além de adicionar interatividade, como permitir que os usuários arrastem e soltem marcadores ou explorem a área em diferentes níveis de *zoom*.

Além da visualização de mapas, a API também permite a obtenção de informações geográficas detalhadas, como coordenadas de latitude e longitude, endereços, nomes de lugares, além de oferecer recursos avançados de pesquisa. Os desenvolvedores podem utilizar essas informações para exibir dados relevantes no mapa, como pontos de interesse, endereços de empresas, informações de trânsito, entre outros, conforme o exemplo abaixo.

Outro recurso importante fornecido pelo Google Maps API é a capacidade de calcular rotas e fornecer direções de navegação. Os desenvolvedores podem usar a API para calcular a rota mais rápida ou otimizada entre dois pontos, exibir a rota no mapa e fornecer informações de direção passo a passo para os usuários.

O funcionamento do Google Maps API é baseado na interação entre os desenvolvedores e os servidores do Google. Os desenvolvedores acessam a API por meio de uma chave de API exclusiva, que autentica e controla o acesso ao serviço. Eles fazem solicitações à API para obter os dados e recursos desejados, e a API responde com as informações solicitadas, como mapas renderizados, dados geográficos ou resultados de pesquisa (Documentation Google Maps API, 2023).

4.1.4 Android Studio

O sistema operacional Android adota uma abordagem de código aberto concebido para capacitar desenvolvedores na criação de aplicativos móveis que

exploram ao máximo as capacidades dos smartphones (Lecheta, 2013).

Conforme mencionado por Pereira e Da Silva (2009), a origem do Android remete a 2003, quando a Android Inc. iniciou seu processo de desenvolvimento. Em 2005, o Google adquiriu a Android Inc. com o propósito de ingressar no mercado de smartphones. Como indicado por Lecheta (2013), o sistema Android oferece uma personalização robusta, integração eficaz com GPS, possui uma variedade de aplicações pré-instaladas e um ambiente de desenvolvimento que é poderoso, inovador e flexível.

Embora tenha sido inicialmente projetado para dispositivos de telefonia móvel, como ressaltado por Bohrer (2011), o Android tem sido amplamente empregado em uma variedade de dispositivos portáteis, incluindo tablets, televisões e até mesmo micro-ondas. No entanto, neste projeto, a ênfase está particularmente voltada para o setor de dispositivos móveis, como telefones e tablets.

No âmbito do desenvolvimento, utiliza-se o Android Studio como uma IDE oficialmente suportado pelo Google para criação de aplicativos Android. Ele apresenta uma ampla diversidade de ferramentas e funcionalidades que facilitam o desenvolvimento, teste e depuração de aplicativos Android.

Ele é baseado na plataforma JetBrains IntelliJ IDEA⁶ e foi projetado especificamente para o desenvolvimento de aplicativos Android. Oferece suporte completo para todas as etapas do processo de desenvolvimento, desde a criação do projeto até a publicação do aplicativo na Play Store.

O Android Studio também fornece um emulador de dispositivo Android integrado, permitindo que os desenvolvedores testem seus aplicativos em diferentes dispositivos virtuais antes de implantá-los em dispositivos reais. Isso facilita a identificação e correção de erros e problemas de compatibilidade.

Além disso, o Android Studio oferece suporte a uma ampla variedade de bibliotecas, *frameworks* e APIs do Android, permitindo que os desenvolvedores acessem recursos avançados do sistema operacional. Ele também é compatível com a linguagem de programação Kotlin, que se tornou oficialmente uma linguagem suportada pelo Android, além do Java.

A integração com o Google Play Services também é um recurso essencial do Android Studio. Ele permite que os desenvolvedores acessem e integrem serviços do

⁶ Ambiente de desenvolvimento integrado (IDE) para programação de software.

Google, como mapas, autenticação, notificações *push*, análises, entre outros, em seus aplicativos Android (De Araújo, 2018).

Conforme Lecheta (2013), a plataforma Android possui uma arquitetura robusta e flexível que facilita a criação de aplicativos para dispositivos móveis eficientes e de alta qualidade. A arquitetura do Android é baseada em camadas, cada uma com sua função específica. Vamos explorar essa arquitetura em detalhes:

- **Aplicativos:** a camada de aplicativos é onde os usuários interagem diretamente com os aplicativos desenvolvidos para Android. Esses aplicativos podem ser instalados e executados em dispositivos Android e são a interface principal para os usuários utilizarem os recursos do sistema operacional;
- **Frameworks de Aplicativos:** essa camada fornece um conjunto de bibliotecas e APIs (interfaces de programação de aplicativos) que permitem o desenvolvimento de aplicativos Android. O Android SDK (Software Development Kit) disponibiliza uma vasta variedade de ferramentas e recursos para os desenvolvedores criarem aplicativos com funcionalidades avançadas;
- **Bibliotecas Android:** as bibliotecas Android são conjuntos de código pré-construído que fornecem recursos e funcionalidades específicas para os desenvolvedores. Elas incluem bibliotecas para gráficos 2D e 3D, manipulação de dados, conectividade de rede, gerenciamento de banco de dados e muito mais. Essas bibliotecas facilitam o desenvolvimento de aplicativos eficientes e otimizados;
- **Máquina Virtual Dalvik:** o Android utiliza uma máquina virtual especial chamada Dalvik. Ela é responsável por executar o código dos aplicativos Android. A Dalvik VM otimiza o desempenho dos aplicativos, convertendo o código-fonte Java em um formato otimizado para execução eficiente em dispositivos móveis;
- **Kernel do Linux:** a camada inferior da arquitetura do Android é o kernel do Linux. Ele fornece os serviços básicos do sistema operacional, como gerenciamento de recursos (memória), gerenciamento de energia, drivers de dispositivo e comunicação com o *hardware* do dispositivo. O kernel do Linux é a base sobre a qual o sistema operacional Android é construído.

Essa arquitetura em camadas permite uma separação clara de responsabilidades e modularidade no desenvolvimento de aplicativos para Android. Ela oferece flexibilidade aos desenvolvedores, permitindo que criem aplicativos com recursos avançados e personalizados, ao mesmo tempo em que garantem a compatibilidade e o desempenho do sistema operacional como um todo.

4.1.5 Inteligência Artificial: Google Cloud Vision API

O Google Cloud Vision API é um serviço oferecido pela Google que utiliza recursos avançados de Inteligência Artificial (IA) para analisar e compreender o conteúdo de imagens. Ele permite que os desenvolvedores realizem tarefas complexas, como reconhecimento de objetos, detecção de rostos, identificação de textos e muito mais. O funcionamento do Google Cloud Vision API é fundamentado em princípios de aprendizado de máquina treinados em um grande conjunto de dados de imagens. Esses algoritmos são capazes de extrair informações valiosas a partir das imagens, analisando pixels, padrões e características visuais.

Ao utilizar o Google Cloud Vision API, os usuários podem enviar imagens para a API e receber resultados com informações detalhadas sobre o conteúdo dessas imagens. Por exemplo, é possível identificar objetos específicos presentes na imagem, como carros, animais, alimentos, e obter informações adicionais sobre eles. Além disso, a API pode detectar e reconhecer rostos, identificar características faciais como olhos, narizes e sorrisos, e até mesmo atribuir estimativas de idade e emoções. Ela também é capaz de identificar textos presentes nas imagens, permitindo a extração de palavras-chave ou até mesmo a tradução de idiomas (Documentation Cloud Vision API, 2023).

4.1.6 Rodando o aplicativo em máquina Ubuntu

Rodar um aplicativo Flutter para gravar em um banco de dados PostgreSQL em uma máquina Ubuntu envolve uma interação complexa entre várias camadas tecnológicas. O Flutter é um framework para o desenvolvimento de aplicativos móveis que faz uso da linguagem Dart para construir interfaces de usuário atrativas e responsivas. Ao utilizar uma biblioteca de acesso a banco de dados, como o pacote

"postgres" em Dart, o aplicativo pode se conectar ao banco de dados PostgreSQL para realizar operações de gravação.

No processo, o aplicativo Flutter envia solicitações para o PostgreSQL, que é um sistema de gerenciamento de banco de dados relacional. O PostgreSQL processa essas solicitações, executa consultas SQL para inserir dados e mantém a consistência e integridade dos dados armazenados. O desempenho desse fluxo depende de vários fatores, incluindo a eficiência das consultas SQL, a velocidade da conexão entre o aplicativo e o banco de dados, a carga de trabalho do servidor PostgreSQL e a otimização do código do aplicativo.

A eficiência do aplicativo Flutter ao gravar em um banco de dados PostgreSQL em uma máquina Ubuntu é influenciada pela qualidade da implementação do código, pela otimização das consultas SQL e pela capacidade do servidor PostgreSQL de lidar com as operações de gravação.

O desempenho pode ser melhorado por meio da indexação adequada de tabelas, configurações otimizadas do servidor PostgreSQL e ajustes na comunicação entre o aplicativo e o banco de dados. A escolha da máquina Ubuntu como sistema operacional pode oferecer estabilidade e flexibilidade, mas também é importante considerar a capacidade de recursos, como CPU, RAM e armazenamento, para garantir um desempenho ideal do aplicativo e do banco de dados (Documentation Ubuntu, 2023).

4.2 Desenvolvimento de Aplicativo Móvel para Localização de Animais de Estimação: Detalhes de Implementação e Funcionalidades

O aplicativo móvel a ser desenvolvido servirá para ajudar na busca por animais de estimação. Ele permitirá que os usuários relatem animais perdidos e forneçam informações, como fotos, descrição, local e data em que o animal foi visto pela última vez. Além disso, o *app* fornecerá recursos de busca e notificação para ajudar na localização dos animais e de seus proprietários.

A seguir, serão fornecidos os detalhes da implementação que atendem aos requisitos do sistema. Esses detalhes incluem informações sobre a arquitetura escolhida, diagrama de estados, estrutura do banco de dados, implementações e testes.

4.2.1 Desenvolvimento das telas

Para este projeto de 5 a 6 telas, as principais funcionalidades serão o cadastro de usuário, que poderá criar uma conta no aplicativo, fornecendo informações básicas, como nome, endereço de e-mail e senha, além do registro dos animais perdidos.

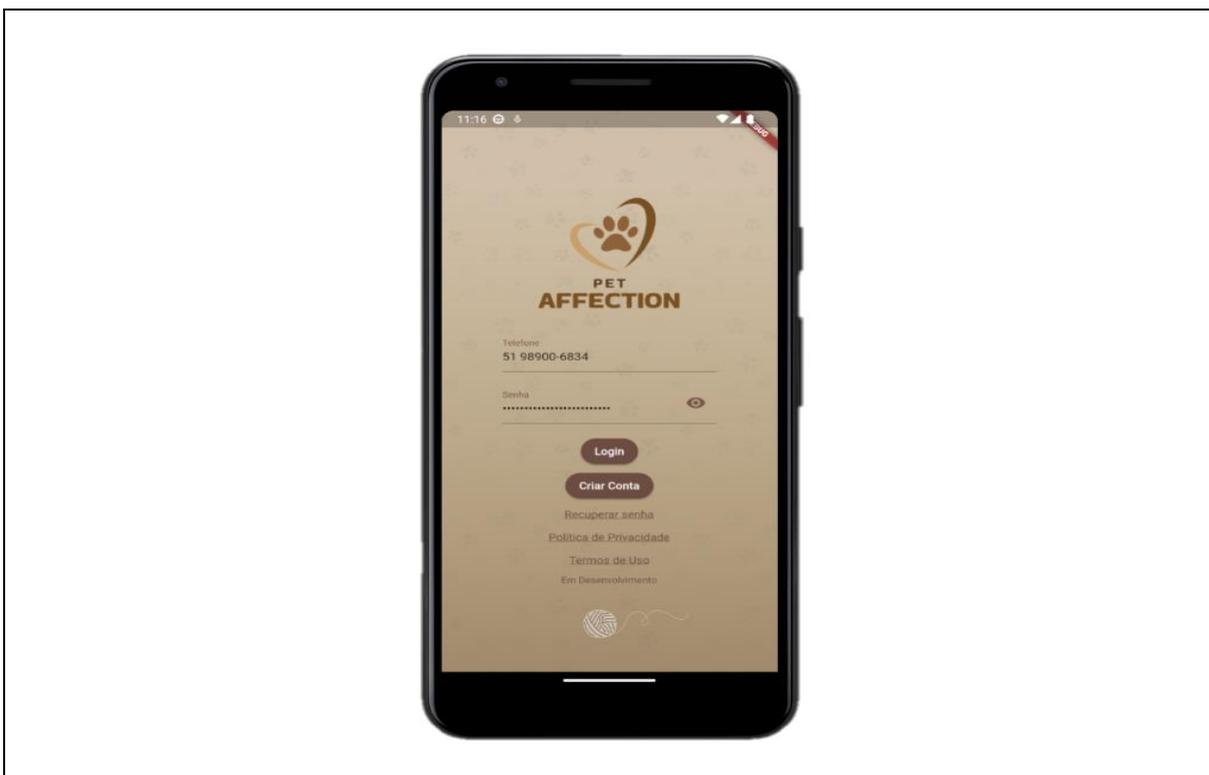
O aplicativo conta com o fornecimento de informações sobre o animal, incluindo foto, raça, cor, porte, local e data do desaparecimento. Além disso, haverá a pesquisa por animais perdidos, que poderá ser feita com base em critérios como localização, raça ou características.

Os detalhes completos de um animal perdido poderão ser visualizados no *app*, incluindo foto, descrição e informações de contato do proprietário.

O aplicativo contará com a notificação de animais encontrados, ou seja, quando um *pet* perdido for encontrado próximo à sua localização, o aplicativo emitirá um alerta.

A seguir, serão fornecidos alguns detalhes da implementação das telas.

Figura 5 – Tela do Login (aplicativo)

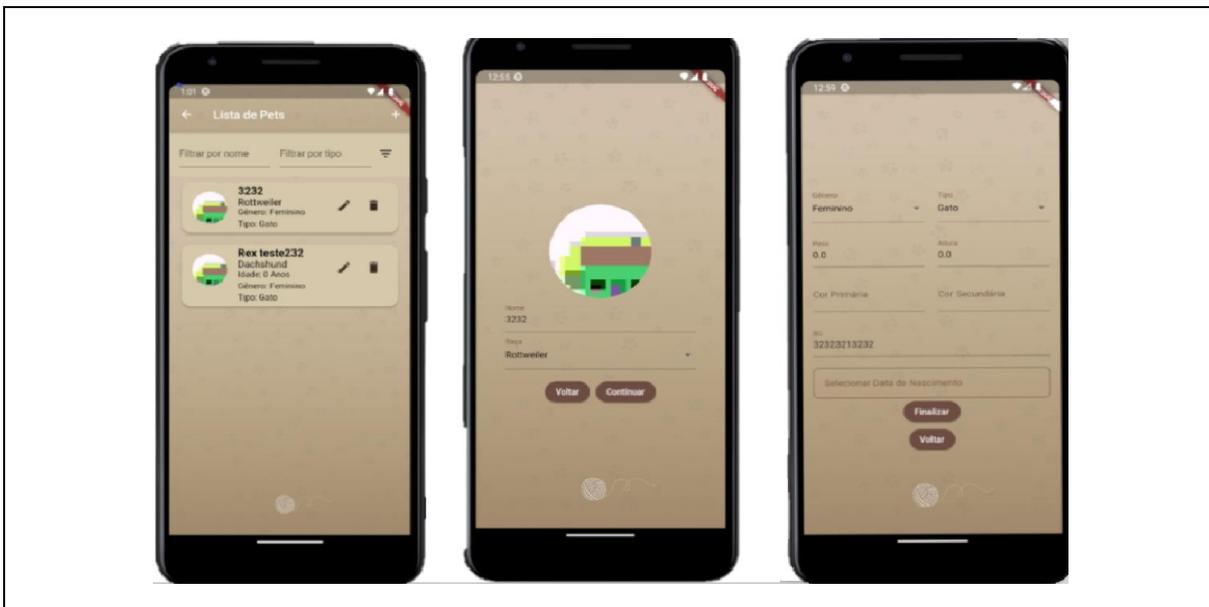


Fonte: Do autor (2023).

O *layout* e o *design* visam a proporcionar uma experiência simples e eficiente, facilitando o acesso dos usuários à plataforma. A tela procura garantir a segurança

das informações, implementando medidas de proteção de dados e seguindo as melhores práticas de segurança *on-line* para proteger a privacidade dos usuários.

Figura 6 – Tela de Cadastro e Listagem do Pet (aplicativo)



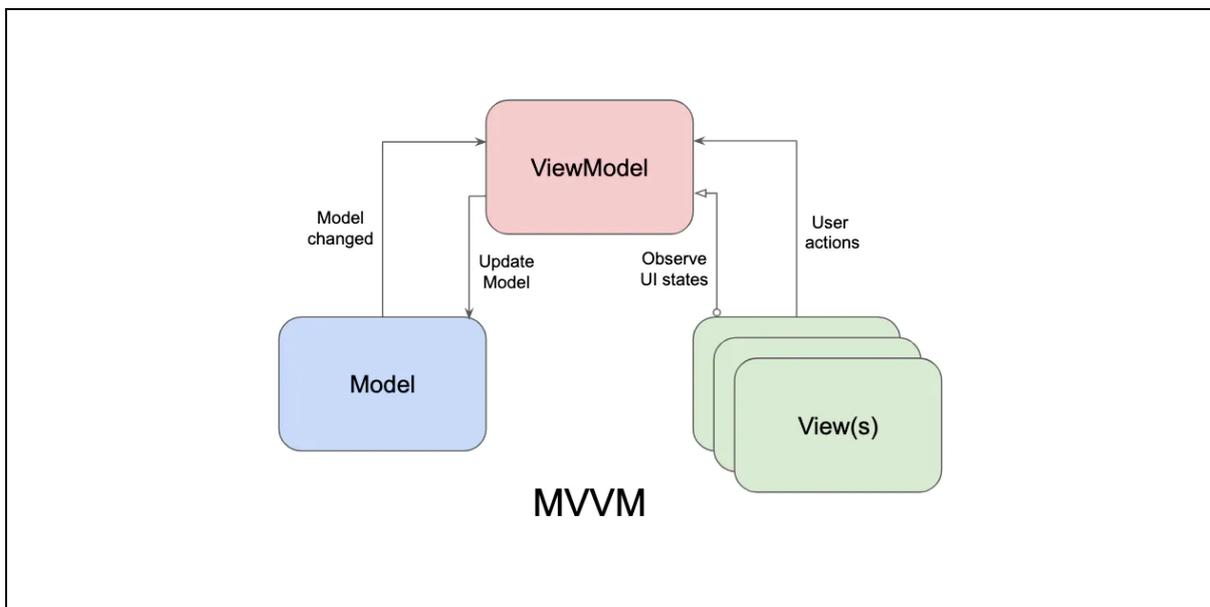
Fonte: Do autor (2023).

Ambas as telas, de Cadastro e de Listagem do Pet, são essenciais para os usuários do aplicativo. Elas permitem que donos de animais desaparecidos forneçam informações detalhadas, além de facilitar a busca e a visualização de *pets* perdidos para aqueles que estão tentando ajudar a localizar esses animais.

4.2.2 Arquitetura do aplicativo

Para a arquitetura e o desenvolvimento do aplicativo, será utilizada a arquitetura MVVM (Model-View-ViewModel). De acordo com Cadu (2010), essa arquitetura divide o aplicativo em três camadas principais: Model: representa a camada de dados e lógica de negócios. Aqui, as informações sobre os animais perdidos e os usuários serão armazenadas e gerenciadas. View: representa a camada de interface do usuário. As telas do aplicativo serão desenvolvidas nesta camada, permitindo que os usuários interajam com o aplicativo. ViewModel: atua como intermediário entre a camada Model e a camada View. Contém a lógica de apresentação e fornece os dados necessários para exibir nas telas. Podemos ver o modelo abaixo sobre o fluxo da arquitetura.

Figura 7 – Arquitetura MVVM



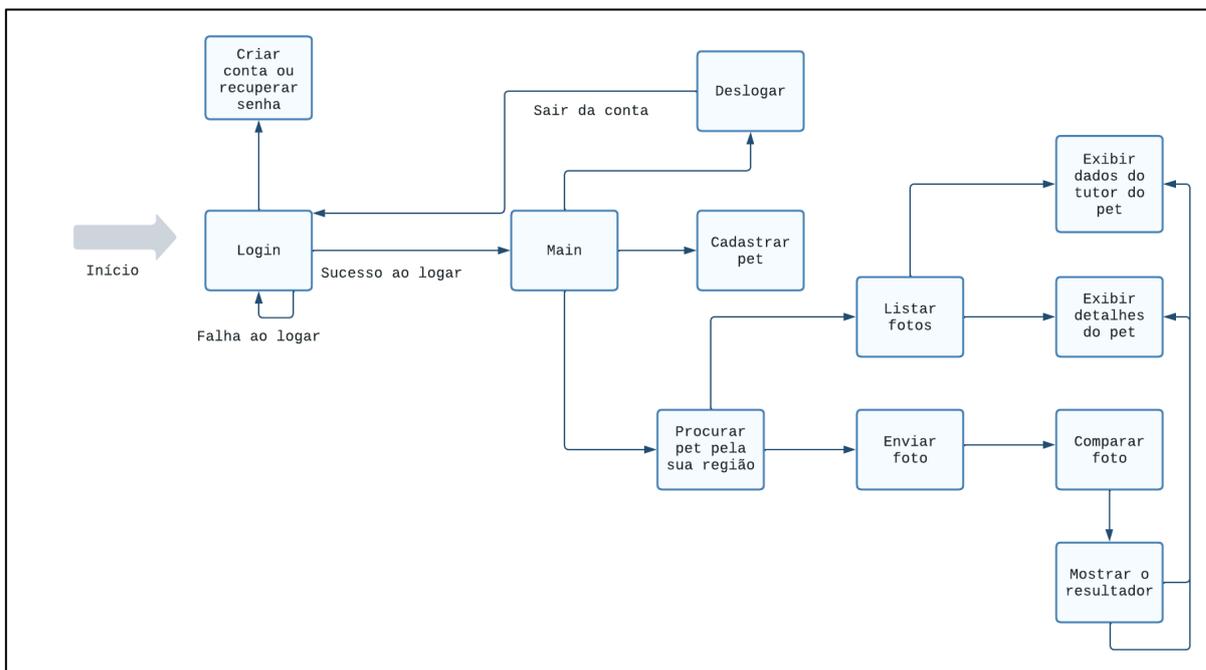
Fonte: Adaptado pelo autor com base no Cadu (2023).

A arquitetura MVVM visa a separação clara de preocupações, permitindo um desenvolvimento mais organizado, escalável e testável. Ao adotar esse padrão, os desenvolvedores podem criar aplicativos mais robustos, com lógica de negócios bem definida, interfaces de usuário dinâmicas e componentes facilmente testáveis.

4.2.3 Diagrama de estados

Com o objetivo de esclarecer as principais operações acessíveis ao utilizador, é empregada a abordagem de máquina de estados, evidenciada na representação gráfica designada como Figura 8. Para compreender tais ocorrências, é fundamental possuir familiaridade com os cenários de utilização da aplicação. A seguir, podemos ver o modelo do diagrama utilizado.

Figura 8 – Ações disponíveis ao usuário



Fonte: Do autor (2023).

Ao iniciar o aplicativo, o primeiro passo consiste em efetuar o login ou criar uma conta. Após isso, o usuário será redirecionado para a tela principal, onde encontrará três opções. A primeira possibilita sair da conta, encaminhando o usuário à tela de login. A segunda escolha possibilita cadastrar seu animal de estimação.

A terceira opção permite procurar o *pet* dentro da sua região, onde serão exibidas as fotos e a descrição de outros *pets* que, quando acessadas, exibirão uma tela com os dados de contato do tutor do *pet* perdido. Na mesma tela da terceira opção, o usuário poderá enviar uma foto do seu *pet* e, logo em seguida, compará-la com outras fotos no banco de dados, sendo que as mesmas exibirão os dados dos *pets* semelhantes e a existência de um tutor.

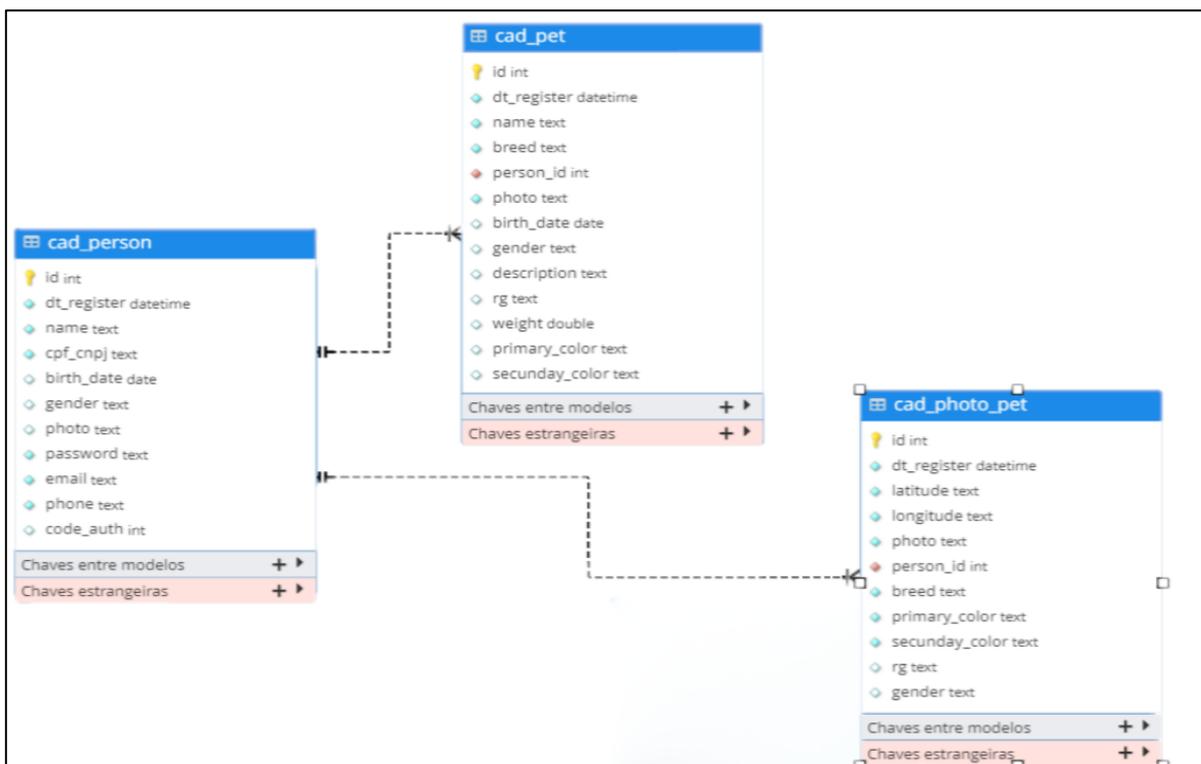
4.2.4 Estrutura do Banco de Dados

O Banco de Dados será *on-line*, pois precisará possibilitar grande número de buscas para encontrar imagens semelhantes.

Considerando o uso do PostgreSQL para armazenamento de dados, a estruturação foi realizada conforme ilustrado na Figura 9. A seguir, apresenta-se o modelo da

estrutura do banco de dados.

Figura 9 – Estrutura do banco de dados



Fonte: Do autor (2023).

Nota-se que há três tabelas principais para segmentar informações dos *pets* e dos usuários. A tabela de usuário é basicamente formada pelos dados do cliente; com isso, ela se liga às demais tabelas, uma que registra o seu histórico chamada *usuario_historico* e a outra para cadastrar o seu animal de estimação, chamada de *pet*.

4.2.5 Requisitos funcionais

No Quadro 2, serão apresentados os requisitos funcionais empregados na construção do aplicativo proposto.

Quadro 2 – Requisitos Funcionais

Nº	Descrição do Requisito	Prioridade
1	Cadastrar usuário	Obrigatório
2	Cadastrar animais	Obrigatório
3	Pesquisar animais perdidos	Obrigatório
4	Detalhar os dados dos animais	Importante
5	Cadastrar o histórico de usuários	Importante
6	Notificar o usuário de animais encontrados	Importante

Fonte: Do autor (2023).

4.2.6 Especificação dos requisitos funcionais

Neste quadro, serão apresentadas as especificações dos requisitos funcionais empregados na construção do aplicativo proposto.

Quadro 3 – Especificações dos Requisitos Funcionais

Nº	Especificação do Requisito
RF 1	Cadastrar os usuários, com nome, endereço de e-mail, idade, telefone e senha como os campos obrigatórios. Validar o endereço de e-mail para não repetir.
RF 2	Cadastrar os animais, contendo a cor, foto, raça, local e data do desaparecimento como campos obrigatórios. Os campos como o nome não serão obrigatórios.
RF 3	Pesquisar animais perdidos com base em critérios como localização, raça, foto, cor e aspecto, se tiver.
RF 4	Detalhar os animais perdidos por completo, incluindo foto, descrição e informações de contato do proprietário.
RF 5	Cadastrar o histórico de usuários contendo a descrição e a data de registro como campos obrigatórios.
RF 6	Notificar o usuário de animais encontrados próximos à localização do usuário.

Fonte: Do autor (2023).

4.2.7 Requisitos não funcionais

Nesta seção, serão apresentados os requisitos não funcionais empregados na construção do aplicativo proposto.

Quadro 4 – Requisitos não funcionais

Nº	Descrição do Requisito	Prioridade
1	Eficiente (Rápido)	Importante
2	Robusto (Seguro)	Obrigatório
3	Usabilidade	Importante
4	Manutenibilidade	Importante
5	Escalabilidade	Importante

Fonte: Do autor (2023).

4.2.8 Especificação dos requisitos não funcionais

Nesta seção, serão delineadas as especificações dos requisitos não funcionais adotados na elaboração do aplicativo proposto.

Quadro 5 – Especificações dos Requisitos Não Funcionais

Nº	Especificação do Requisito
RNF 1	O sistema deve ser capaz de suportar pelo menos 1000 usuários simultâneos sem apresentar degradação significativa no tempo de resposta.
RNF 2	As senhas dos usuários devem ser armazenadas de forma criptografada no banco de dados.
RNF 3	A interface do usuário deve ser intuitiva e de fácil aprendizado, para que usuários novatos possam realizar tarefas sem a necessidade de treinamento extenso.
RNF 4	As atualizações de <i>software</i> devem ser realizadas de forma incremental e sem afetar negativamente a operação do sistema.
RNF 5	A capacidade de armazenamento do banco de dados deve ser facilmente ampliada à medida que o número de registros cresce.

Fonte: Do autor (2023).

4.2.9 Fases do Desenvolvimento do Aplicativo Móvel: Detalhes e Progresso na Implementação

Uma vez que a arquitetura do sistema, o diagrama de estados e a estrutura do banco de dados foram definidos, o desenvolvimento do aplicativo foi iniciado, dividindo-se em etapas, que são abordadas detalhadamente nas subseções seguintes. Cada subseção descreve o progresso do desenvolvimento em uma fase específica, fornecendo uma visão clara do processo de implementação em andamento.

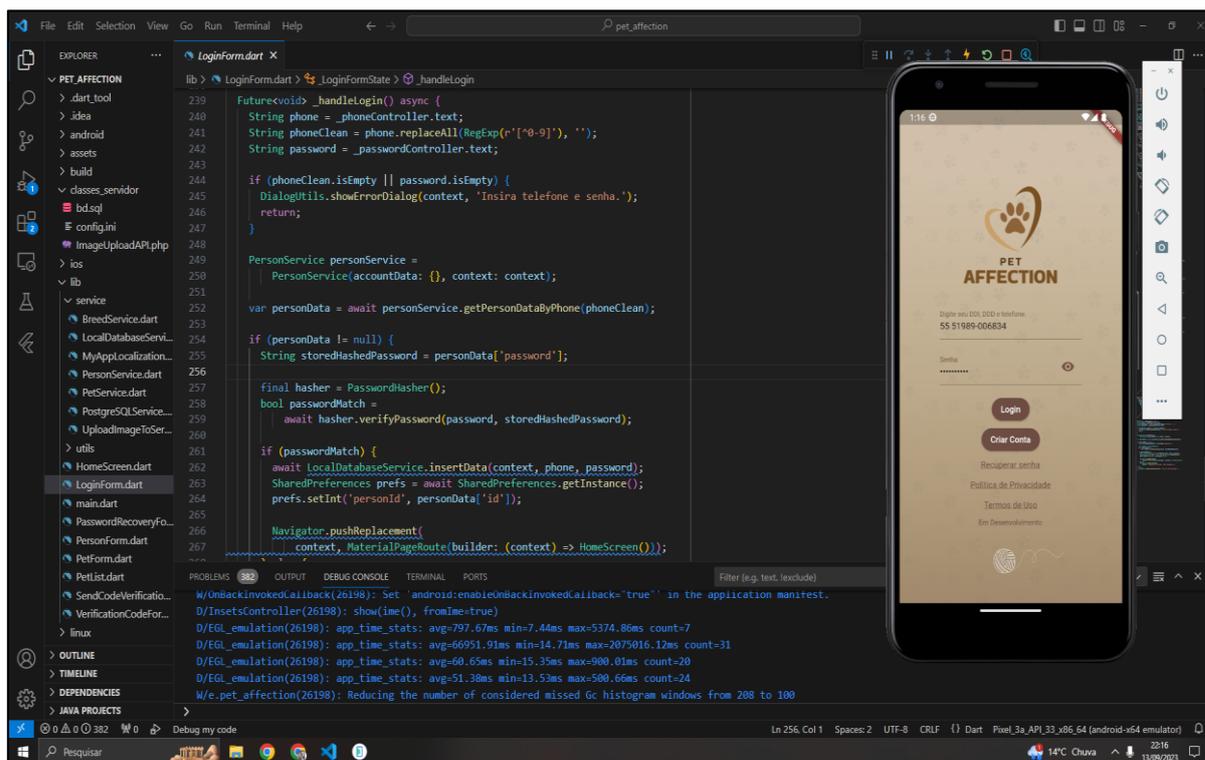
4.2.10 Implementação de Autenticação Multifacetada e Gerenciamento de Sessões em Aplicativo Móvel: Detalhes e Código Fonte

Com base na função criada, foram incorporadas alternativas para autenticar usuários, criar novas contas e recuperar senhas por meio de e-mail, SMS e gerenciamento de sessões de login no aplicativo.

A primeira alternativa possibilita a autenticação de usuários, a criação de novas contas e o gerenciamento das sessões de login.

Após a criação da conta, a função *handlelogin* é utilizada para verificar se a autenticação foi bem-sucedida. Se a autenticação falhar, uma mensagem será exibida ao usuário (tais como senha inválida, conta inexistente, conta não verificada, e outras causas potenciais). A seguir, é possível verificar o código fonte que valida essas informações, onde ele verifica os campos e os dados para ver se existe algum dado incorreto para ser salvo como por exemplo o telefone, ou se existe algum campo vazio, além de verificar a segurança da senha.

Figura 10 – Implementação da Autenticação e Tela de Login ao Fundo



Fonte: Do autor (2023).

A combinação eficiente da implementação da autenticação com uma tela de *login* bem projetada e funcional é essencial para oferecer uma experiência de usuário

avançados, permitindo que os usuários visualizem mapas, editem informações geográficas e interajam com base na localização em tempo real. Esses recursos são frequentemente usados em aplicativos de navegação, entrega, redes sociais e muito mais.

Figura 12 – Resultado do SQL dos filtros por todos os *pets* da região em até 100km

```

petaffection=# SELECT *
FROM cad_photo_lost_pet
WHERE (
  6371 * 2 * ASIN(
    SQRT(
      SIN(
        RADIANS(CAST(latitude AS NUMERIC) - -29.391046650429317) / 2
      ) * SIN(
        RADIANS(CAST(latitude AS NUMERIC) - -29.391046650429317) / 2
      ) + COS(RADIANS(-29.391046650429317)) * COS(
        RADIANS(CAST(latitude AS NUMERIC))
      ) * SIN(
        RADIANS(CAST(longitude AS NUMERIC) - -51.95779703557491) / 2
      ) * SIN(
        RADIANS(CAST(longitude AS NUMERIC) - -51.95779703557491) / 2
      )
    )
  ) <= 100;
 id | dt_register | latitude | longitude | photo | person_id | breed_id | primary_color | rg | gender
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
  6 | 2023-09-22 07:54:17 | -29.391046650429317 | -51.95779703557491 | /var/www/html/repository/650d72571482c.jpg | 2 | 5 | hhsuus | | 
  9 | 2023-09-25 19:53:34 | -29.3911358 | -51.9586388 | /var/www/html/repository/65120f68893db.jpg | 2 | 6 | asssh | | 
 10 | 2023-09-25 20:03:03 | -29.3909374 | -51.9589552 | /var/www/html/repository/651211a19154d.jpg | 2 | 6 | shshzgzg | | 
(3 rows)

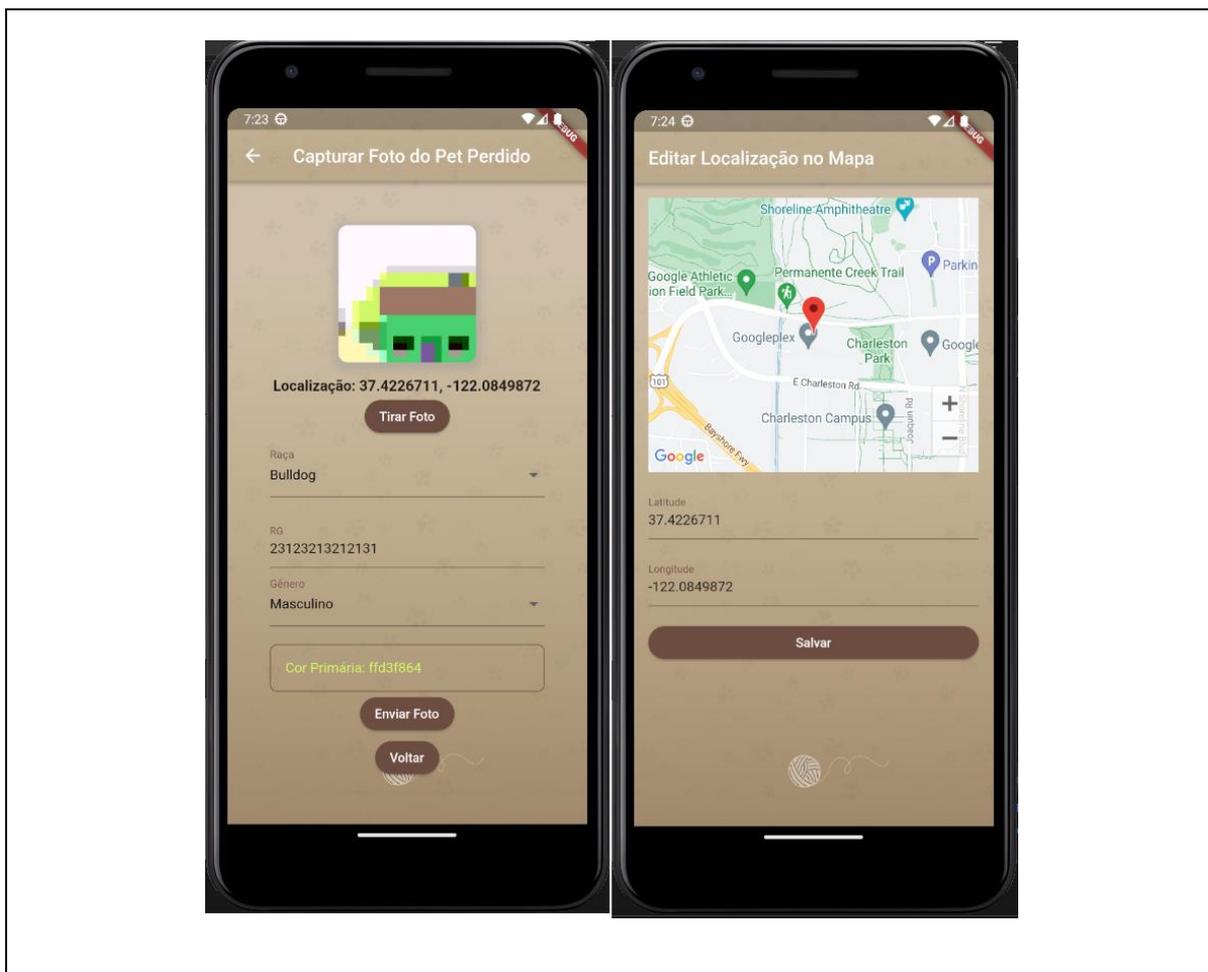
```

Fonte: Do autor (2023).

O resultado de todas essas implementações é visualizado na Figura 12, onde o teste para pegar automaticamente a localização e, logo após, colocá-la no mapa.

Nessa Figura 13, ele pega e manda para a tela de edição para o usuário confirmar se realmente é aquela sua localização. Na Figura 13, podemos ver o resultado sendo exibido em tela para o usuário.

Figura 13 – Implementação do Widget da Edição do Mapa



Fonte: Do autor (2023).

A implementação de um *widget* de edição de mapa é um processo desafiador, pois requer a combinação de interações de usuário complexas, integração com dados geospaciais e a capacidade de manipular elementos gráficos em tempo real. A escolha da biblioteca certa e a compreensão das funcionalidades específicas desejadas são fundamentais para o sucesso na implementação deste tipo de recurso.

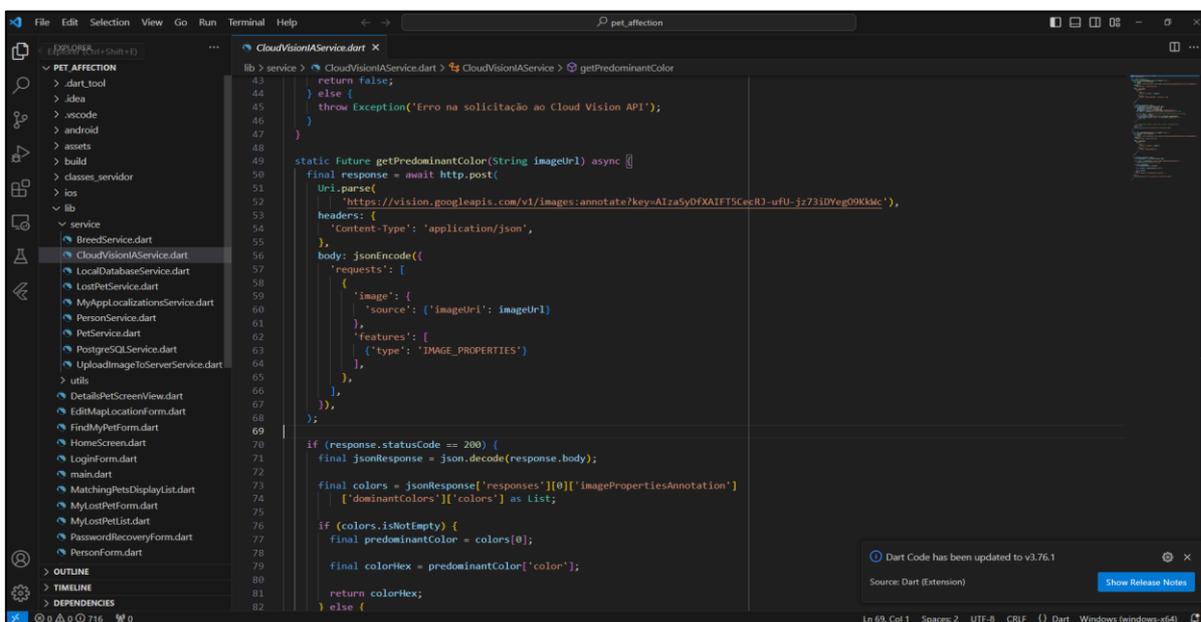
4.2.12 Desvendando o Poder da Inteligência Artificial: Explorando a Implementação Flexível e Acessível do Cloud Vision API

A implementação e uso da inteligência artificial do Cloud Vision API são bastante acessíveis e flexíveis. Através de uma simples integração com a plataforma Google Cloud, os desenvolvedores podem enviar solicitações HTTP para a API, permitindo o processamento de imagens para análise de conteúdo visual. A IA do Cloud Vision oferece recursos como detecção de rostos, leitura e reconhecimento de

textos, identificação de objetos e cenários, além da capacidade de categorizar conteúdo explícito. Os resultados são retornados em formato JSON, possibilitando uma fácil interpretação e utilização dentro de aplicativos e sistemas. Essa implementação é extremamente útil em diversos setores, desde a automação de tarefas rotineiras até a análise avançada de imagens em aplicações de segurança, e-commerce, saúde, entre outros.

Na imagem abaixo, podemos ver o código fonte implementado no aplicativo, onde ele se comunica com a API do Google e verifica os dados se estão coerentes, pois ele valida se na foto é um animal, se for ele vai verificar a cor predominante do animal colocando-a como cor primária e logo em seguida a sua forma e assim validando se o pet é semelhante as fotos que estão na base, e se estão em na geolocalização adequada. (Documentation Cloud Vision API, 2023).

Figura 14 – Implementação da Inteligência Artificial no App



```

lib > service > CloudVisionService.dart > CloudVisionService > getPredominantColor
43  return false;
44  } else {
45    throw Exception('Erro na solicitação ao Cloud Vision API');
46  }
47  }
48  }
49  }
50  static Future<Color> getPredominantColor(String imageUrl) async {
51    final response = await Http.post(
52      Uri.parse(
53        'https://vision.googleapis.com/v1/images:annotate?key=AIzaSy0FXAIFT5CecR3-ufU-jz73iDyep09Kkic',
54      ),
55      headers: {
56        'Content-Type': 'application/json',
57      },
58      body: jsonEncode({
59        'requests': [
60          {
61            'image': {
62              'source': {'imageUrl': imageUrl}
63            },
64            'features': [
65              {'type': 'IMAGE_PROPERTIES'}
66            ],
67          },
68        ],
69      }));
70  if (response.statusCode == 200) {
71    final jsonResponse = json.decode(response.body);
72    final colors = jsonResponse['responses'][0]['imagePropertiesAnnotation']
73      ['dominantColors']['colors'] as List;
74    if (colors.isNotEmpty) {
75      final predominantColor = colors[0];
76      final colorHex = predominantColor['color'];
77      return colorHex;
78    } else {
79      return null;
80    }
81  }
82  }

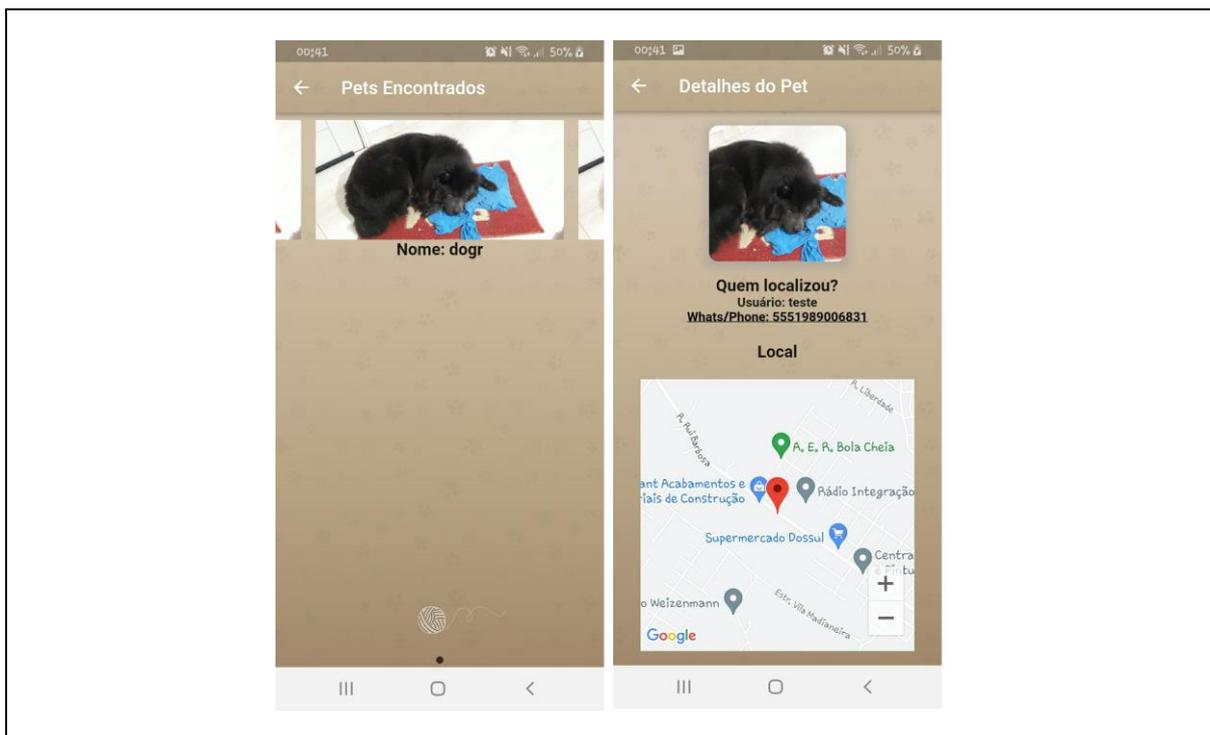
```

Fonte: Do autor (2023).

Para utilizar a Cloud Vision API, é necessário integrá-la ao aplicativo por meio de chamadas de API. Os desenvolvedores podem enviar imagens para a API e receber as respostas estruturadas com as informações extraídas da imagem.

A Cloud Vision API permite uma ampla gama de funcionalidades, facilitando a análise e compreensão de imagens, sendo extremamente útil para aplicativos que necessitam de recursos de visão computacional, reconhecimento e interpretação de imagens de forma rápida e precisa. Na imagem a seguir podemos ver o resultado sendo exibido em tela.

Figura 15 – Resultado da Implementação da Inteligência Artificial no App



Fonte: Do autor (2023).

O resultado obtido na imagem mostra o sucesso da implementação em tela para o usuário.

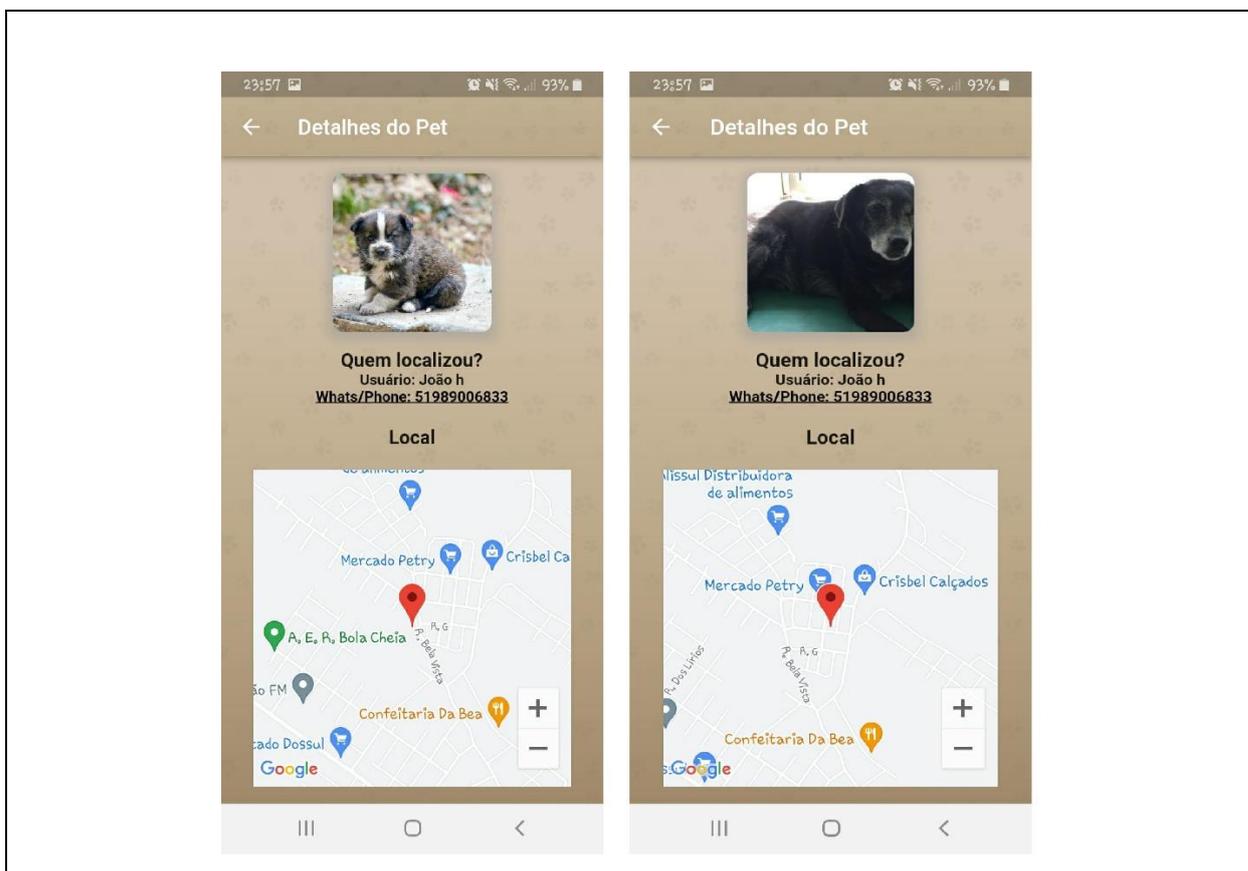
5 TESTES E ANÁLISE DOS RESULTADOS

Conforme Black (2011), o planejamento da sistemática de testes e análise dos resultados é um processo crucial para o sucesso de um aplicativo para *pets*.

A relevância do aplicativo de ajuda a *pets* perdidos foi avaliada por meio de testes práticos em colaboração com a Associação de Proteção aos Animais de Arroio do Meio (APAAM), corroborando a importância destacada na conclusão deste estudo.

Os testes foram meticulosamente planejados e realizados para avaliar as funcionalidades cruciais do aplicativo. Os cenários de teste abrangeram desde a facilidade de navegação até a precisão da identificação por inteligência artificial (IA), passando pela segurança dos dados e a capacidade de reunir informações sobre animais perdidos. Como na imagem abaixo.

Figura 16 – Resultado da Implementação com sucesso

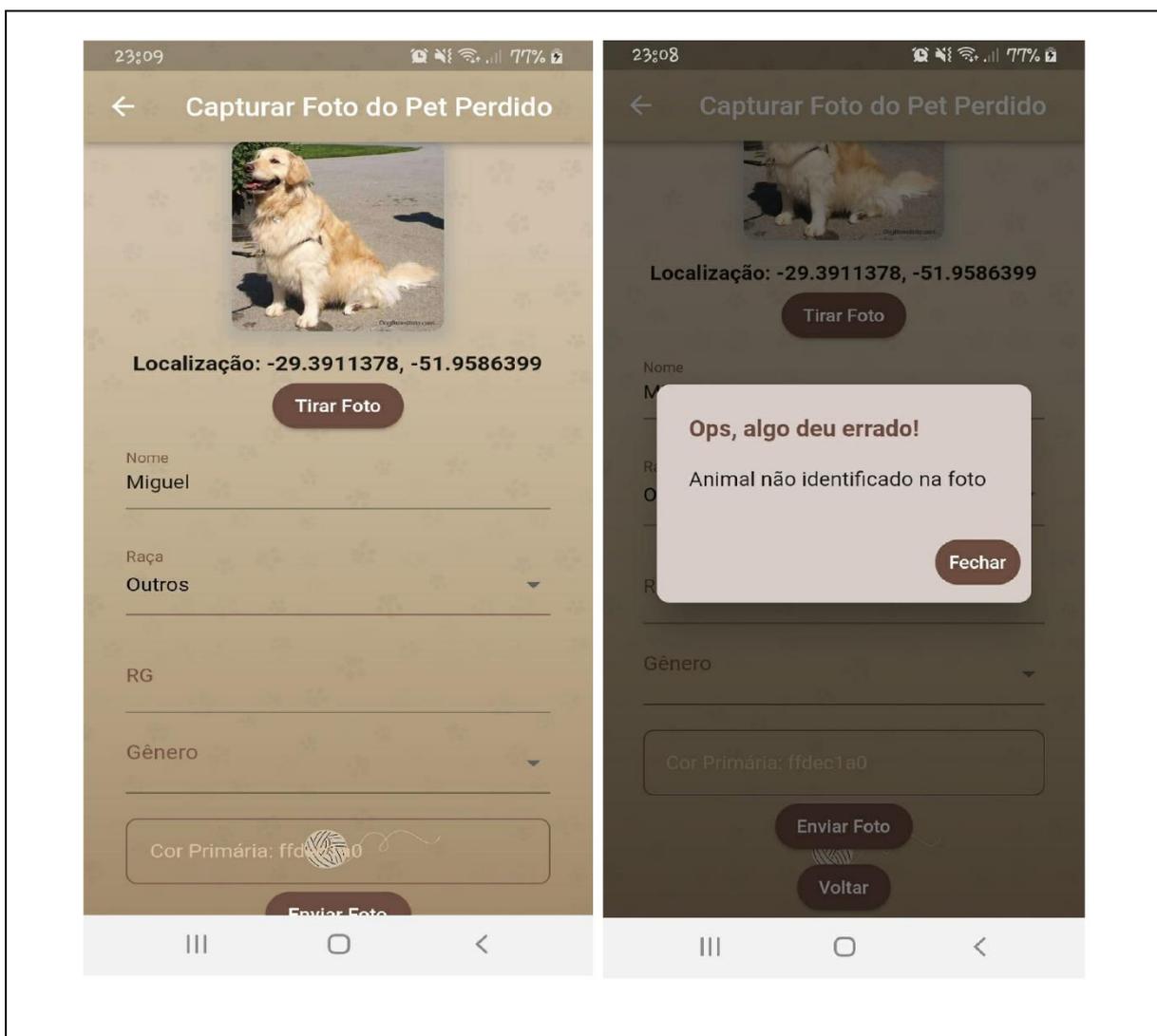


Fonte: Do autor (2023).

Nessa imagem podemos ver o resultado da comparação de um pet preto, onde o mesmo localizou dois próximos a região e semelhantes em aspectos de cores.

A metodologia adotada incluiu testes de unidade e integração, refletindo uma abordagem multifacetada para avaliar o desempenho do aplicativo em dispositivos variados. A execução dos testes seguiu os cenários predefinidos, com resultados registrados detalhadamente, enfatizando falhas identificadas e a análise criteriosa dos resultados obtidos. Algumas das falhas foi registrado com prints, como as fotos abaixo.

Figura 17 – Resultado da Implementação com erros

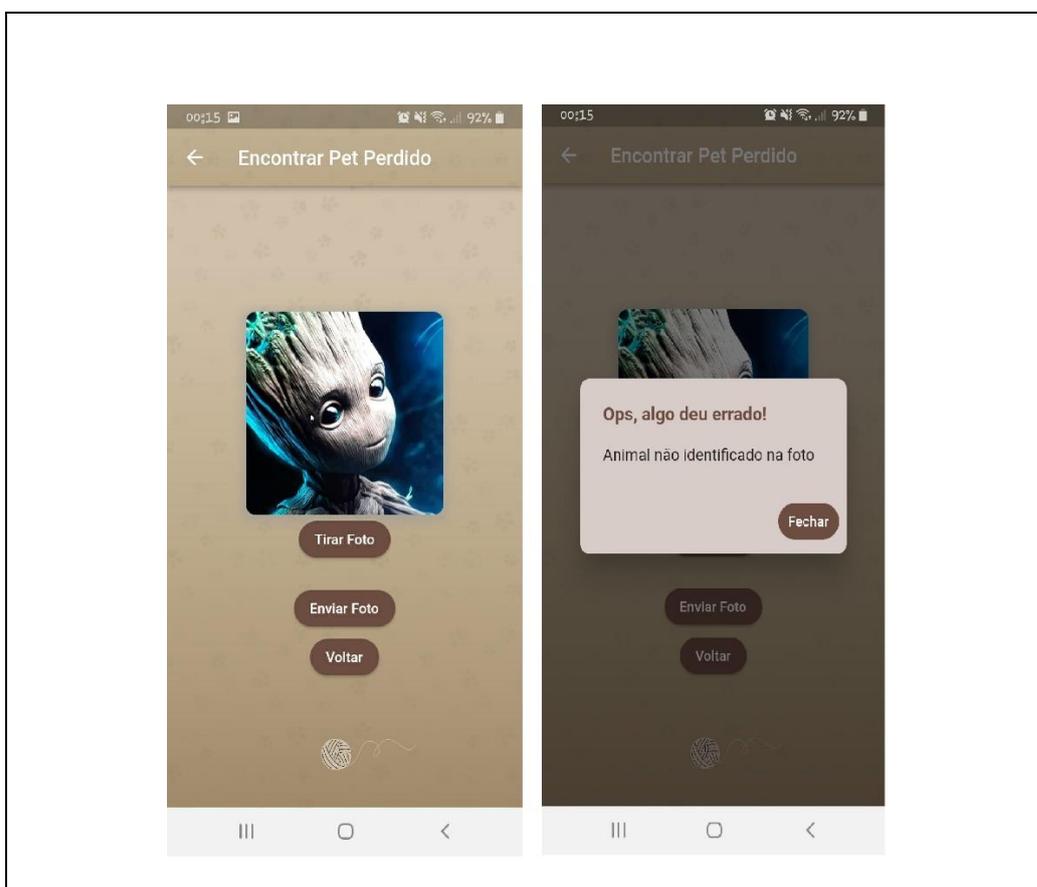


Fonte: Do autor (2023).

Na figura 17, podemos ver um erro de implementação que mais para a frente foi corrigido, onde o animal não foi identificado na foto, justamente por estar distante ou até mesmo pela falta de iluminação.

Os testes na APAAM permitiram identificar áreas de aprimoramento e desafios a serem superados. Embora a aplicação da IA para identificação de animais perdidos tenha mostrado resultados promissores, particularmente em relação à precisão, desafios foram evidenciados, ressaltando a necessidade contínua de melhorias nessa área. Como na imagem abaixo, onde podemos ver a implementação corrigida.

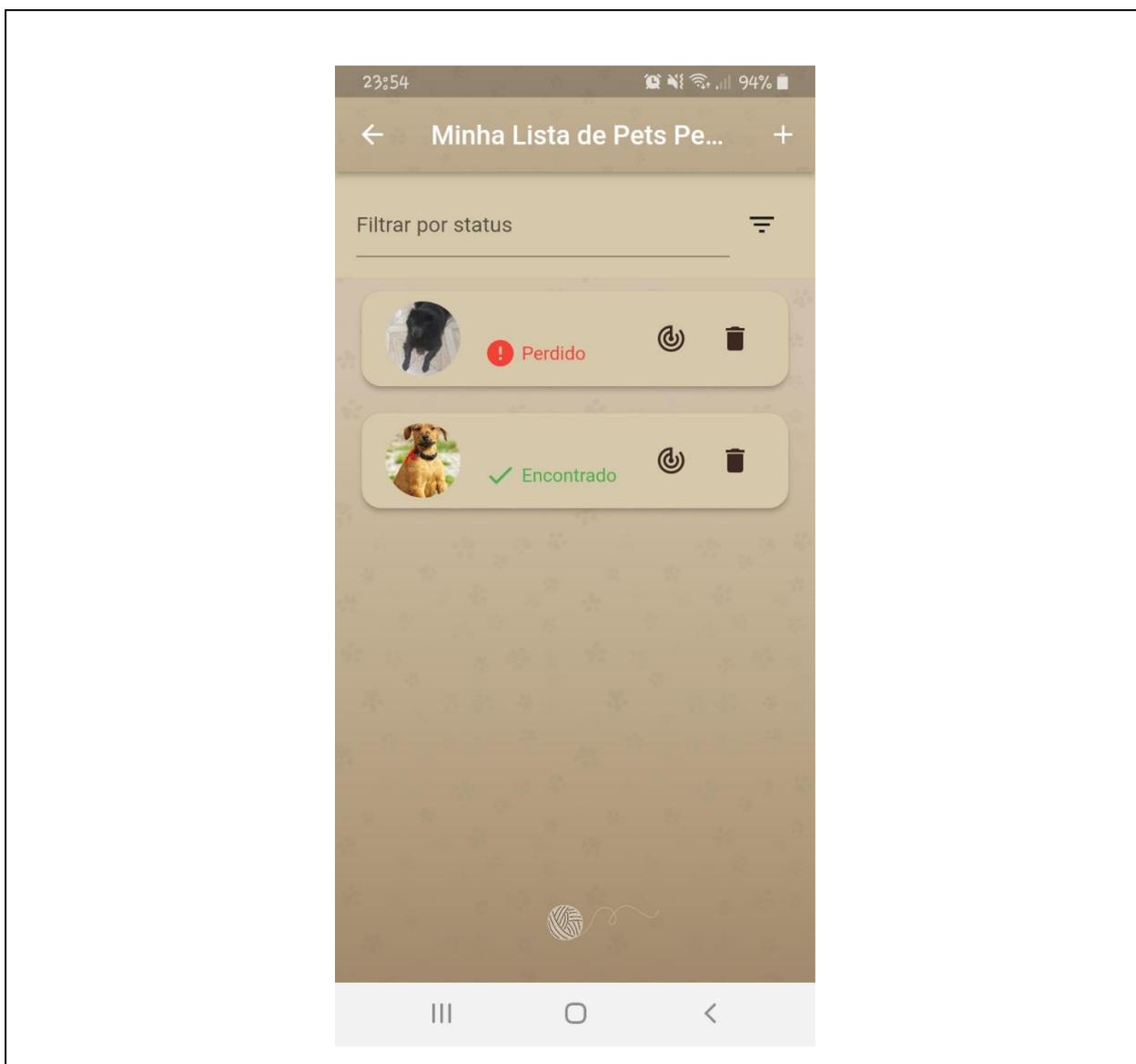
Figura 18 – Resultado da Implementação corrigida



Fonte: Do autor (2023).

A implementação de ajustes e melhorias foi baseada nos resultados dos testes, buscando corrigir falhas e aprimorar a funcionalidade do aplicativo, garantindo uma experiência mais precisa e confiável para os usuários. Podemos ver na figura 18 o resultado sendo efetivo para identificação dos animais, além desse filtro e validação também é possível melhorar o filtro para facilitar o sucesso da I.A em achar os pets perdidos, como por exemplo na figura abaixo.

Figura 19 – Resultado da Implementação de status



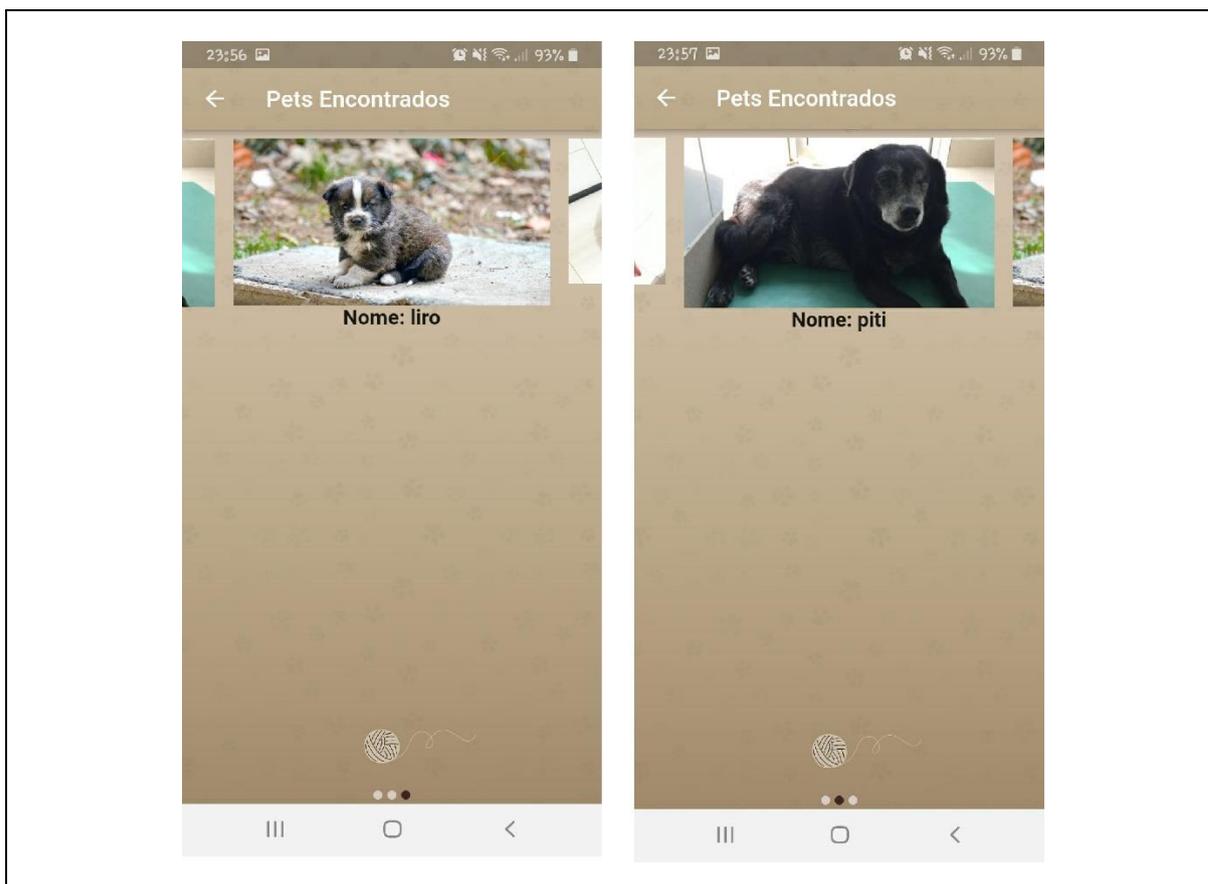
Fonte: Do autor (2023).

Na figura 19 podemos ter uma tela de status para o usuário marcar se o seu pet foi encontrado ou se ainda está perdido, e com base nesse status a I.A vai pegar da tabela do banco de dados e verificar se ele está perdido ou não, pois se ele estiver com o status de “Encontrado” a I.A vai ignorar a foto.

Com isso a repetição dos testes permitiu verificar a eficácia das modificações implementadas, reforçando a importância da continuidade na verificação e validação do sistema, especialmente no que tange à identificação precisa por IA. Na imagem abaixo podemos ver os pets encontrados e clicando neles podemos acessar dados específicos e até mesmo conversar com o usuário que o localizou conforme na

imagem 16 e 20.

Figura 20 – Mais resultados da implementação



Fonte: Do autor (2023).

A imagem da figura 20 mostra várias fotos com os resultados semelhantes tirados da figura 19, onde podemos ver um pet marcado como “Perdido” nela, e com isso, os testes práticos desempenharam um papel fundamental ao evidenciar a importância intrínseca de uma abordagem sistemática na avaliação da eficácia do aplicativo. A sistemática aplicada não apenas permitiu uma análise abrangente das funcionalidades, mas também proporcionou uma compreensão mais profunda do desempenho do sistema em cenários do mundo real. A interação direta com a Associação de Proteção aos Animais de Arroio do Meio (APAAM) durante os testes reforçou a relevância prática do aplicativo, demonstrando sua utilidade tangível no apoio à causa da proteção animal.

Além disso, a abordagem sistemática não se limitou à mera identificação de problemas; ela se estendeu à formulação de soluções tangíveis. Os resultados dos testes não apenas ressaltaram áreas de aprimoramento, mas também forneceram uma base sólida para a implementação de ajustes e melhorias substanciais. A

colaboração estreita entre os desenvolvedores e a APAAM, como evidenciado pelos resultados positivos da implementação corrigida, reflete um compromisso genuíno com a qualidade e a eficácia do aplicativo.

A identificação contínua de áreas de melhoria durante o ciclo de testes reforça a necessidade de um comprometimento constante com o aprimoramento contínuo do sistema. A Figura 20, que apresenta mais resultados da implementação, é emblemática desse compromisso, destacando a natureza iterativa do processo de teste e validação. Esses resultados subsequentes não apenas validam as modificações implementadas, mas também lançam luz sobre a importância de uma vigilância constante para garantir a excelência operacional do aplicativo.

Em resumo, os testes práticos não apenas validaram a eficácia do aplicativo, mas também ressaltaram que a abordagem sistemática é um pilar essencial para o desenvolvimento contínuo. A compreensão das complexidades do mundo real, aliada à implementação proativa de melhorias, solidifica a posição do aplicativo como uma ferramenta valiosa no suporte à proteção animal, reforçando seu impacto positivo na sociedade.

6 CONSIDERAÇÕES FINAIS

Um aplicativo de ajuda a *pets* pode desempenhar um papel fundamental na unificação de animais de estimação perdidos com seus tutores. O aplicativo proposto neste estudo pode ter um impacto significativo na vida de pessoas que perderam seus *pets*. Ele oferece uma plataforma centralizada para relatar, pesquisar e compartilhar informações, aumentando as chances de reencontro dos animais com seus proprietários.

Além das funcionalidades básicas de relato e pesquisa de animais perdidos, é interessante explorar recursos avançados que possam aumentar a eficácia do aplicativo, pois a inteligência artificial (IA) vai trabalhar para achar fotos semelhantes que se encontram em seu banco de dados. Por meio dessa plataforma, abrigos e resgatistas podem aumentar sua visibilidade e alcançar um público maior, facilitando o processo de adoção e a captação de recursos para oferecer cuidados adequados aos animais resgatados.

O desenvolvimento de um aplicativo para auxílio a *pets* perdidos demonstra um impacto substancial no setor de reunificação animal e no suporte aos tutores que enfrentam a angústia de perder um animal de estimação. Ao longo desta pesquisa e do processo de prototipagem, evidenciou-se que a tecnologia desempenha um papel vital na ampliação das chances de localizar animais perdidos e devolvê-los a seus donos.

O aplicativo oferece uma plataforma centralizada e acessível para reportar, pesquisar e compartilhar informações sobre animais perdidos. A integração de IA para filtrar e identificar fotos semelhantes dentro do banco de dados do aplicativo é um

avanço promissor, aumentando as possibilidades de encontrar animais desaparecidos de forma mais rápida e eficiente. Essa funcionalidade representa uma contribuição significativa para a missão do aplicativo.

Além das funcionalidades básicas de relato e pesquisa de animais perdidos, a implementação de recursos avançados como a IA permite uma melhoria considerável na precisão e na velocidade de identificação de animais perdidos por meio de fotografias. Este avanço promove não só a reunião entre tutores e animais perdidos, mas também serve como uma ferramenta de apoio para abrigos e resgatistas, ampliando sua visibilidade e permitindo um alcance mais amplo de potenciais adotantes.

Durante o processo de teste e prototipagem, algumas dificuldades foram identificadas, tais como desafios na otimização da precisão da IA em identificar fotos semelhantes, necessidade contínua de um banco de dados abrangente e garantia da segurança dos dados dos usuários. Essas dificuldades mostraram-se como áreas críticas que demandam atenção contínua para aprimoramento e aperfeiçoamento do aplicativo.

No que diz respeito à viabilidade técnica, os testes realizados sugerem que a solução é promissora e viável para ser implementada em larga escala. Contudo, considerando a manutenção, a atualização do banco de dados e o desenvolvimento contínuo da IA, os custos operacionais e de manutenção são pontos a serem cuidadosamente considerados para garantir a sustentabilidade do aplicativo a longo prazo.

Sobre o público-alvo do aplicativo, ele acaba abrangendo uma ampla gama de usuários, incluindo tutores de animais de estimação, abrigos de animais, resgatistas e potenciais adotantes. Tutores que perderam seus animais podem se beneficiar diretamente da plataforma ao relatar e procurar por seus pets, enquanto abrigos e resgatistas encontram no aplicativo uma ferramenta valiosa para aumentar a visibilidade de animais disponíveis para adoção.

Além disso, a proposta do aplicativo cria oportunidades para pesquisadores e profissionais do campo da inteligência artificial e desenvolvimento de software se envolverem em projetos relacionados. A implementação da IA para identificação de animais perdidos abre portas para aprimoramentos e pesquisas adicionais nesse domínio, contribuindo para o desenvolvimento de tecnologias mais avançadas.

Outros trabalhos que poderiam se vincular a este aplicativo incluem pesquisas sobre a eficácia de tecnologias de reconhecimento de imagem em contextos específicos, estudos sobre a psicologia e o impacto emocional de tutores de animais que perderam seus pets e análises da efetividade de aplicativos semelhantes em outras regiões ou para diferentes espécies de animais.

O aplicativo não só incita a inovação no campo da reunificação animal, mas também inspira trabalhos futuros que podem explorar novas funcionalidades, aprimorar a precisão da IA e adaptar a tecnologia para atender a diversas necessidades dentro da comunidade de cuidadores de animais. Essa interconexão entre o aplicativo e pesquisas complementares pode contribuir significativamente para o avanço e a evolução contínua no campo do bem-estar animal e da tecnologia aplicada.

Portanto, o aplicativo de ajuda a *pets* perdidos apresenta uma proposta promissora para a reunificação de animais com suas famílias, fornecendo uma solução inovadora e tecnologicamente avançada. A contínua pesquisa, desenvolvimento e aprimoramento são essenciais para maximizar o potencial do aplicativo e seu impacto positivo na comunidade de tutores de animais de estimação e organizações de resgate.

REFERÊNCIAS

ANAT, Biletzki; ANAT, Matar. **Ludwig Wittgenstein**. [S./], 20 out. 2021. Disponível: <https://plato.stanford.edu/entries/wittgenstein/>. Acesso em: 01 mai. 2023.

ANJO. **Abandono de animais**: entenda qual é o impacto na vida do pet. [S./], 16 dez. 2022. Disponível: <https://petanjo.com/blog/paredeabandonar-conscientizacao-sobre-abandono-de-animais/>. Acesso em: 21 mai. 2023.

BENNINGTON, Geoffrey. **A besta e o soberano**: três notas para Derrida. [S./], 17 jan. 2015. Disponível: <https://www.scielo.br/j/alea/a/dDXHRZv7xn8ZV7dfcw6rnFM/?lang=pt>. Acesso em: 05 mai. 2023.

BLACK, Rex. **Advanced Software Testing**. 3. ed. Santa Barbara, 2011.

BOHRER, Fernando José. **Serviço de geolocalização para plataforma Android**. 2011. Trabalho de Conclusão de curso (Monografia) - Centro Universitário Univates, Lajeado, RS, 2011.

BRITO, Katia. **Influência e benefícios dos animais de estimação na vida das pessoas**. [S./], 16 dez. 2019. Disponível: <https://www.psicologoeterapia.com.br/blog/infuencia-e-beneficios-dos-animais-de-estimacao-na-vida-das-pessoas/>. Acesso em: 26 mai. 2023.

BÚSSOLA. **Abandono de animais aumentou cerca de 60% durante a pandemia**. [S./], 27 dez. 2021. Disponível: <https://exame.com/bussola/abandono-de-animais-aumentou-cerca-de-60-durante-a-pandemia/>. Acesso em: 26 mai. 2023.

CADU. **Entendendo o Pattern Model View ViewModel MVVM**. [S./], 2010. Disponível: <https://www.devmedia.com.br/entendendo-o-pattern-model-view-viewmodel-mvvm/18411>. Acesso em: 05 mai. 2023.

CARLOS. **Artigo SQL Magazine 37 - PostgreSQL**. [S./], 2007. Disponível: <https://www.devmedia.com.br/artigo-sql-magazine-37-postgresql/6891>. Acesso em: 18 jul. 2023.

CONSELHO FEDERAL DE MEDICINA VETERINÁRIA – CFMV. ORGANIZAÇÃO MUNDIAL DA SAÚDE - OMS. **A Proteção Animal Mundial premia as melhores iniciativas de cuidado com cães e gatos nas cidades da América Latina**. 31 out. 2022. Disponível: <https://www.cfmv.gov.br/protacao-animal-mundial-premia-as-melhores-iniciativas-de-cuidado-com-caes-e-gatos-nas-cidades-da-america-latina/comunicacao/noticias/2019/05/20/>. Acesso em: 21 mai. 2023.

CORAZZA, Paulo. **Um aplicativo multiplataforma desenvolvido com Flutter e NoSQL para o cálculo da probabilidade de apendicite**. [S./], 2018. Disponível: <https://www.lume.ufrgs.br/handle/10183/190147>. Acesso em: 05 mai. 2023.

COSTA, Marcelo. **PostgreSQL: Armazenamento de dados em formato "schemaless"**. [S./], 22 out. 2013. Disponível: <https://www.infoq.com/br/articles/postgresql-schemaless/>. Acesso em: 18 jul. 2023.

DE ARAÚJO, Gregory Eilert. POLITER – Aplicativo android voltado a interesses políticos. **Projeto Aplicado para Projetista-Projeto Integrador**, [S./], v. 1, n. 1, 2018.

DOCUMENTATION CLOUD VISION API. **Documentação da Pesquisa de produtos da API Vision**. 2023. Disponível em: <https://cloud.google.com/vision/product-search/docs/?hl=pt-br>. Acesso em: 19 jul 2023.

DOCUMENTATION GOOGLE MAPS API. **Google Maps Plataform**. 2023. Disponível em: <https://mapsplatform.google.com/>. Acesso em: 19 jul. 2023.

DOCUMENTATION UBUNTU. **Official Ubuntu Documentation**. 2023. Disponível em: https://help.ubuntu.com/?_ga=2.252282057.1145112606.1693015981-2015953045.1693015981. Acesso em: 25 ago. 2023.

FREY, Matheus. **FINDED**: Aplicação mobile para rastreamento geográfico e comunicação via chat entre membros de um grupo de interesse. [S./], nov. 2020. Disponível em: <https://www.univates.br/bduserver/api/core/bitstreams/2f2b3ebd-b378-4af7-bdd4-dc4c5ff18939/content>. Acesso em: 17 jul. 2023.

GIL, Antonio C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2002. Acesso em: 18 jul. 2023.

GOMES, Lucas. **A história dos apps, uma breve linha do tempo**. [S./], 25 mai. 2021. Disponível: <https://www.showmetech.com.br/a-historia-dos-apps/>. Acesso em: 19 mai. 2023.

LECHETA, Ricardo R. Google Android. 3. ed. **Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. [S./]: Novatec Editora, 2013.

LIMA, Fernando. **Avaliação de Frameworks para o Desenvolvimento de Aplicações Híbridas**. [S.l.], 24 jun. 2019. Disponível: <https://dspace.unipampa.edu.br/bitstream/riu/4224/1/Fernando%20Fortunato%20de%20Lima%20-%202019.pdf>. Acesso em: 16 jul. 2023.

LING, Rich. **The Mobile Connection: The Cell Phone's Impact on Society**. 1. ed. Califórnia: San Francisco, 2004.

NEVES, Jonathan; JUNIOR, Vilmar. **Uma análise comparativa entre Flutter e React Native como Frameworks para desenvolvimento híbrido de aplicativos Mobile: estudo de caso visando produtividade**. Tubarão, SC, 2021. Disponível: <https://repositorio.animaeducacao.com.br/bitstream/ANIMA/15960/1/Artigo%20final.pdf>. Acesso em: 18 jul. 2023.

NUSSBAUM, Martha. Creating Capabilities: The Human Development Approach. **Hypatia**, [S.l.], v. 24. n. 3, 2011.

PILAGALLO, Sofia. App ajuda a achar pets perdidos a partir de reconhecimento facial. **Noticias.r7**, 25 set. 2021. Disponível: <https://noticias.r7.com/tecnologia-e-ciencia/app-ajuda-a-achar-pets-perdidos-a-partir-de-reconhecimento-facial-25092021>. Acesso em: 26 mai. 2023.

PRADO, Juan; ANTONIO, José. **Aplicación móvil para ayudar a localizar animales perdidos**. 28 de jul. 2022. Disponível: <https://rua.ua.es/dspace/handle/10045/125655>. Acesso em: 21 mai. 2023.

PRODANOV, Cleber Cristiano; DE FREITAS, Ernani Cesar. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico**. 2. ed. [S.l.]: Editora Feevale, 2013.

REGAN, Tom. **The Case for Animal Rights**. [S.l.], 1986. Disponível: https://www.wellbeingintlstudiesrepository.org/cgi/viewcontent.cgi?article=1003&context=acwp_awap. Acesso em: 01 mai. 2023.

ROSÁRIO, Fábio. **Desenvolvimento de aplicativos móveis multiplataforma**. [S.l.], 16 nov. 2015. Disponível: https://repositorio.utfpr.edu.br/jspui/bitstream/1/13394/1/MD_COADS_2015_2_05.pdf. Acesso em: 16 jul. 2023.

SALCEDO, Octavio; ORTIZ, Maria; ESPITIA, Miguel. Development of a Mobile application that allows the search of lost pets in bogota using geographical location and image recognition. **International Journal of Applied Engineering Research**, [S.l.], v. 13, n. 15, p. 12055-12062, 2018. Disponível: https://www.ripublication.com/ijaer18/ijaerv13n15_50.pdf. Acesso em: 18 mai. 2023.

SICHMAN, Jaime. Inteligência Artificial e sociedade: avanços e riscos. **Estudos Avançados**, [S.l.], v. 35, n. 101, abr. 2021. Disponível em: <https://www.scielo.br/j/ea/a/c4sqqrthGMS3ngdBhGWtKhh#ModalTutors>. Acesso em: 15 jul. 2023.

SILVA, Antonio. Continuum de vulnerabilidades e capabilities approach: o fundamento comum de Martha C. Nussbaum à promoção dos direitos humanos e dos direitos dos animais não humanos. **Revista do Programa de Pós-Graduação em Direito da UFBA**, [S.l.], v. 28, n. 2, p. 121-146, jul./ dez. 2018. Disponível: <https://periodicos.ufba.br/index.php/rppgd/article/view/29019>. Acesso em: 05 mai. 2023.

SILVA, Caio Poli; FEYH, Pedro Gustavo Ramm; ROLAND, Carlos Eduardo de França. PETS: desenvolvimento de sistema de geolocalização para monitoramento de animais de estimação. **Revista eletrônica de sistemas de informação e gestão tecnológica**, [S.l.]: Uni-FACEF, v. 9, n. 3, 2018.

SILVEIRA, Denise Tolfo; CÓRDOVA, Fernanda Peixoto. Unidade 2 – A pesquisa científica. **Métodos de pesquisa**, [S.l.], v. 1, 2009. Disponível em: https://www.cesadufs.com.br/ORBI/public/uploadCatalago/11315818082016Pratica_de_Pesquisa_I_Aula_2.pdf. Acesso em: 18 jul. 2023.

SINGER, Peter. **Libertação Animal**. [S.l.], 1975. Disponível: <https://olhequenao.files.wordpress.com/2011/12/peter-singer-libertac3a7c3a3o-animal.pdf>. Acesso em: 01 mai. 2023.

SOUZA, Giesele. Aplicativo de inteligência artificial. **Techtudo**, 2023. Disponível: <https://www.techtudo.com.br/listas/2023/07/aplicativo-de-inteligencia-artificial-para-foto-veja-6-opcoes-edapps.ghtml>. Acesso em: 17 jul. 2023.

APÊNDICE

APÊNDICE A - Repositório do Projeto

Repositório do Projeto. 2023. Disponível em: <https://github.com/J-H-N-Campos/Pet-Affection>



UNIVATES

R. Avelino Talini, 171 | Bairro Universitário | Lajeado | RS | Brasil
CEP 95914.014 | Cx. Postal 155 | Fone: (51) 3714.7000
www.univates.br | 0800 7 07 08 09