



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPOS DE QUIXADÁ
CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

LUCAS ROBSON PEROBA MATIAS

**DESENVOLVIMENTO DE APLICAÇÃO VOLTADO À PRESTAÇÃO DE SERVIÇOS
PARA PETS**

QUIXADÁ

2022

LUCAS ROBSON PEROBA MATIAS

DESENVOLVIMENTO DE APLICAÇÃO VOLTADO À PRESTAÇÃO DE SERVIÇOS PARA
PETS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de Informação
do Campos de Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de em Sistemas de Informação.

Orientador: Prof. David Sena Oliveira

QUIXADÁ

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- M38d Matias, Lucas Robson Peroba.
Desenvolvimento de um aplicativo voltado à prestação de serviços para pets / Lucas Robson Peroba
Matias. – 2022.
53 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá,
Curso de Sistemas de Informação, Quixadá, 2022.
Orientação: Prof. Dr. David Sena Oliveira.
1. Aplicativos móveis. 2. Prestação de serviços. 3. Animais domésticos.

CDD 005

LUCAS ROBSON PEROBA MATIAS

DESENVOLVIMENTO DE APLICAÇÃO VOLTADO À PRESTAÇÃO DE SERVIÇOS PARA
PETS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de Informação
do Campos de Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de em Sistemas de Informação.

Aprovada em:

BANCA EXAMINADORA

Prof. David Sena Oliveira (Orientador)
Universidade Federal do Ceará (UFC)

Profa. Dra. Lívia Almada Cruz
Universidade Federal do Ceará (UFC)

Prof. Dr. Victor Aguiar Evangelista de Farias
Universidade Federal do Ceará (UFC)

RESUMO

O PetWalk é uma inovadora aplicação que conecta tutores de animais de estimação que desejam proporcionar atividades ao ar livre para seus queridos pets com prestadores de serviços dispostos a passeá-los com cuidado e carinho. Esta plataforma oferece a conveniência e segurança de contratar cuidadores para animais de estimação de maneira tão simples e eficiente como encomendar comida ou solicitar um serviço de transporte, resultando na criação de empregos e na melhoria da qualidade de vida tanto para os animais como para seus tutores.

Palavras-chave: PetWalk; aplicativo; tutores de pets; animais.

ABSTRACT

PetWalk is an innovative application that connects pet guardians who wish to provide outdoor activities for their beloved pets with service providers willing to walk them with care and affection. This platform offers the convenience and safety of hiring pet caregivers as easily and efficiently as ordering food or requesting a transportation service, resulting in job creation and an improved quality of life for both the animals and their guardians.

Keywords: PetWalk; application; pet guardians; Animals.

LISTA DE ILUSTRAÇÕES

Figura 1 – Fluxo Geral de funcionamento da <i>WEB</i>	16
Figura 2 – Passos metodológicos	26
Figura 3 – pergunta 1	30
Figura 4 – pergunta 2	31
Figura 5 – pergunta 3	31
Figura 6 – pergunta 4	31
Figura 7 – pergunta 5	32
Figura 8 – pergunta 6	32
Figura 9 – pergunta 7	32
Figura 10 – UML do banco de dados	35
Figura 11 – Rota de pegar o usuário do tipo tutor baseado no id.	36
Figura 12 – Rota de pegar o usuário do tipo passeador baseado no id.	36
Figura 13 – Rota para salvar um endereço.	37
Figura 14 – Rota para salvar um pet.	37
Figura 15 – Rota para criar uma reserva.	37
Figura 16 – Rota para consultar o histórico de reserva	38
Figura 17 – Rota para consultar mensagens de um chat a partir do id de passeio	38
Figura 18 – Rota para enviar uma mensagem	38
Figura 19 – <i>Home</i>	39
Figura 20 – perfilPetWalker	40
Figura 21 – Início da reserva	41
Figura 22 – Processo de reserva	42
Figura 23 – Esperando confirmação	43
Figura 24 – Telas do passeador	44
Figura 25 – Chat	45
Figura 26 – Correção Reserva e Endereço	49
Figura 27 – Correção 02	50

SUMÁRIO

1	INTRODUÇÃO	9
1.1	Objetivos	10
1.1.1	<i>Objetivo Geral</i>	10
1.1.2	<i>Objetivos específicos</i>	10
2	FUNDAMENTAÇÃO TEÓRICA	11
2.1	Processo de Desenvolvimento de Software	11
2.1.1	<i>Kanban</i>	12
2.1.2	<i>Teste de Usabilidade</i>	13
2.2	Sistema WEB	14
2.3	NoSQL e Firebase	17
2.3.1	<i>NoSQL</i>	17
2.3.2	<i>Firebase</i>	18
2.4	Google API	18
2.4.1	<i>Google Maps</i>	19
2.4.2	<i>Google Geolocation</i>	20
3	TRABALHOS RELACIONADOS	21
3.1	HAPPY PET: Plataforma de contratação de <i>Pet Sitters</i>	21
3.2	Passeios Disruptivos: Um Passo para o Futuro	22
3.3	Pet Walk	23
3.4	Análise Comparativa	24
4	METODOLOGIA	26
4.1	Realizar uma pesquisa para entender quais as funcionalidades desejadas pelo público alvo	27
4.2	Checar as possíveis funcionalidades que possam ser implementadas na solução a ser desenvolvida	27
4.3	Definição dos requisitos a serem implementados	27
4.4	Modelar o sistema a fim de atender os requisitos definidos	27
4.5	Implementação dos requisitos definidos	28
4.6	Validação do sistema	28
5	DESENVOLVIMENTO	29

5.1	Pesquisa	29
5.1.1	Formulários	29
5.1.1.1	<i>Pesquisa</i>	29
5.1.1.2	<i>Estudo da arte</i>	33
5.2	Análise do sistema	33
5.2.1	Análise dos requisitos	33
5.2.1.1	<i>Requisitos referentes ao usuário do tipo Tutor</i>	33
5.2.1.2	<i>Requisitos referentes ao usuário do tipo Passeador</i>	34
5.3	Banco de dados	35
5.4	Camada de negócios	35
5.5	Camada de apresentação	39
5.5.1	<i>Perfil do passeador</i>	40
5.5.2	<i>Reserva</i>	41
5.5.3	<i>Chat</i>	45
5.5.4	<i>Geolocalização</i>	46
6	TESTES DE USABILIDADE	47
6.1	Resultados do teste de usabilidade	47
6.2	Correções pós-testes	48
7	CONCLUSÕES E TRABALHOS FUTUROS	51
	REFERÊNCIAS	52

1 INTRODUÇÃO

Com o rápido crescimento da população que adota animais domésticos, é muito comum que tutores tenham diversos objetivos, se envolvem em diferentes atividades e acabam com carência de tempo para atividades cotidianas simples. Na última década, a tecnologia mudou conforme a interação com o mundo, expandiu horizontes físicos, criou possibilidades, conectou pessoas, serviços e necessidades. O surgimento das plataformas agregadoras de serviços como comida e transporte mudou severamente os hábitos das pessoas em relação às suas prioridades na gestão do próprio tempo. Prioridades essas que também precisam alcançar nossos queridos *pets*, que acabam em épocas de contingência das nossas vidas, como trabalho, doença ou cansaço, privados da satisfação de suas reais necessidades de vida.

Uma pesquisa feita pelo IBGE aponta que quase 48 milhões de domicílios no Brasil têm cães ou gatos. 57% dos donos de cães admitem pular caminhadas todas as semanas. Os motivos mais comuns que eles oferecem são: 56% por clima insatisfatório, 32% por pressões de trabalho, 31% por dificuldades em lidar com o cão e 24% por responsabilidades familiares (IBGE, 2019). Analisando a pesquisa, pode-se constatar que a maior parcela dos animais domésticos animais está sofrendo com as consequências de uma vida sedentária e monótona.

Para os cachorros, a prática de exercício físico através do passeio ou corrida é uma necessidade de suma importância para queima de calorias, para prevenir obesidade canina, ansiedade, depressão e stress, além de, socializar, fazer as necessidades fisiológicas, aflorar o instinto do cão, gastar a energia, entre outras vantagens que promovem a qualidade de vida dos animais, tornando-os mais felizes e saudáveis (LOPES, 2019). Atualmente no Brasil existem profissionais que oferecem esse serviço de passeio, porém a tecnologia usada como facilitadora para uma grande parcela são meios de comunicação como o *Whatsapp*, grupos do *Facebook* ou através de indicações em clínicas veterinárias, ou *petshops*. Dessa forma, a divulgação e os processos são feitos de forma manual e muitas vezes sem nenhuma garantia para ambas as partes, resultando num processo custoso e inseguro.

Este trabalho apresenta a proposta do desenvolvimento de uma aplicação chamada *PetWalk*, onde a problemática em foco é a importância do passeio para os *pets* e a carência da oferta de passeadores confiáveis, especializados e de qualidade. A aplicação *PetWalk* tem por objetivo facilitar o encontro de profissionais com o público alvo, garantindo serviços com segurança e praticidade, gerando renda e agregando saúde e qualidade de vida aos animais de estimação.

O presente trabalho apresenta uma pesquisa sobre soluções tecnológicas desenvolvidas para auxiliar a prestação de serviços *pet* através com aplicações que apresentam recursos como chat, geolocalização, atendimento personificado, gerenciamento de agenda entre outras funcionalidades. Na etapa de desenvolvimento, propõe-se um aplicativo voltado para dispositivos móveis especializado no contrato do serviço de passeios com animais domésticos através do uso das tecnologias inclusas nos celulares modernos, alguns recursos-chave para o aplicativo como o uso da câmera para captação de imagens durante o passeio e o monitoramento da localização do animal via geolocalização. Assim, o dono do pet pode se sentir seguro ao acompanhar o passeador durante todo o percurso do passeio. Ao fim do passeio, o aplicativo vai permitir contabilizar o número de vezes que o animal fez suas necessidades fisiológicas, tempo total do passeio e tamanho do percurso percorrido. A aplicação também deve permitir o cadastro dos animais, a geração de um histórico de passeio, instruções para treinamento dos passeadores e aceitar pagamentos online.

1.1 Objetivos

1.1.1 Objetivo Geral

Desenvolver uma aplicação voltada a entrega do serviço de passeio com animais domésticos.

1.1.2 Objetivos específicos

- Implantar fluxo de reserva de passeio;
- Implementar chat e geolocalização;
- Validar a usabilidade da interface.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os principais conceitos para o desenvolvimento deste trabalho disponíveis na literatura. Na Seção 2.1 são tratados os conceitos relacionados a processos de desenvolvimento de um software, bem como o método *Kanban* e testes de usabilidade. Na Seção 2.2 são tratados os conceitos de Sistema *WEB* e uma arquitetura de desenvolvimento desse tipo de sistemas. Na seção 2.3 é apresentado a tecnologias usadas no banco de dados, tais como *noSQL* e *Firestore*. Por último, na Seção 3.4 são tratados os conceitos das *APIs* do *Google* que serão importantes para o desenvolvimento do aplicativo.

2.1 Processo de Desenvolvimento de Software

Pressman e Maxim (2016) definem processo de software como:

Processo é um conjunto de atividades, ações e tarefas realizadas na criação de algum artefato. Uma atividade se esforça para atingir um objetivo amplo (por exemplo, comunicar-se com os envolvidos) sendo utilizada independentemente do campo de aplicação, do tamanho do projeto, da complexidade dos esforços ou do grau de rigor com que a engenharia de software será aplicada. Uma ação (por exemplo, projeto de arquitetura) envolve um conjunto de tarefas resultantes em um artefato de software fundamental (por exemplo, um modelo arquitetural). Dessa forma, a tarefa se concentra em um objetivo pequeno, porém bem definido (por exemplo, realizar um teste de unidades), e produz um resultado tangível. (PRESSMAN; MAXIM, 2016).

De acordo com Sommerville (2019), embora existam diferentes processos de desenvolvimento de software, todos eles devem incluir de alguma forma as quatro atividades fundamentais da engenharia de software:

- **Especificação:** A funcionalidade do software e as restrições sobre sua operação devem ser definidas.
- **Desenvolvimento:** O software deve ser produzido para atender à especificação.
- **Validação:** O software deve ser validado para garantir que atenda ao que o cliente deseja.
- **Evolução:** O software deve evoluir para atender às mudanças nas necessidades dos clientes.

Dentre os processos de desenvolvimento de software mais conhecidos pode-se citar: modelo cascata, modelo RAD, Modelo em V, Modelo Incremental, Modelo Evolutivo, Modelo Espiral, entre outros.

Para a atividade de especificação, a Engenharia de Software dispõe de uma disciplina denominada Engenharia de Requisitos. De acordo com Pressman e Maxim (2016), essa disciplina estabelece uma base sólida para o projeto e construção de um software. Sem ela, o software

resultante tem grande probabilidade de não atender às necessidades do cliente. É nesse processo que ocorre a comunicação entre o cliente e o analista da equipe de desenvolvimento.

O presente trabalho usará as seguintes técnicas da engenharia de requisitos para a etapa de especificação do projeto: Documento de Requisitos, Prototipação de Sistema e Modelagem de Banco de dados.

O documento de requisitos de software é uma declaração oficial do que os desenvolvedores do sistema devem implementar. Ele pode incluir os requisitos de usuário para um sistema e uma especificação detalhada dos requisitos de sistema. O nível de detalhe que deve ser incluído em um documento de requisitos depende do tipo de sistema que está sendo desenvolvido e do processo de desenvolvimento utilizado (SOMMERVILLE, 2019).

De acordo com Vazquez e Simões (2016), a prototipação é uma técnica que busca simular para o usuário o funcionamento dos seus requisitos antes que o produto final esteja pronto. É utilizado para, entre outras coisas, melhorar a experiência do usuário, avaliar opções de design e montar uma base para o desenvolvimento final do produto. A prototipação pode ser classificada como sendo de **baixa fidelidade**, onde os protótipos são desenhados na forma de esboço e sem muitos detalhes, não havendo a preocupação prioritária com as características estéticas, e de **alta fidelidade**, onde é apresentado um protótipo de alta fidelidade e com um nível alto de detalhes do ponto de vista da aparência, sendo o usuário capaz de enxergar de modo mais realista possível como será o produto final.

A Modelagem de Banco de dados é a etapa aonde, a partir dos requisitos do sistema, são definidos as entidades a serem criadas no banco de dados e seus relacionamentos entre si. O artefato gerado nessa modelagem pode ser um Modelo Entidade-Relacionamento (MER) ou Diagrama Entidade-Relacionamento (DER).

2.1.1 Kanban

Kanban é uma palavra de origem japonesa que significa cartão visual. Baseia-se em uma ideia muito simples: o número total de atividades em andamento deve ser limitado, e as atividades só devem ser lançadas quando puderem ser lançadas, ou seja, quando uma atividade em andamento for liberada, ou caso os limites impostas esteja disponível. Ele usa um mecanismo visual para acompanhar o fluxo de atividades. Utilizamos o quadro kanban como metodologia de gerenciamento das atividades do time.

Kanban Systems for Software Development é uma filosofia de gestão que pode ser

aplicada a qualquer tipo de processo para aprimorá-lo gradativamente. *Kanban* não tem regras. Você pode alterar o *Kanban* livremente, mas ele possui propriedades que podem melhorar efetivamente sistemas complexos, gerar processos e coordenar, mesmo entre várias equipes. Esses atributos são:

- Visualizar o fluxo de trabalho.
- Limitar o trabalho em andamento.
- Mede e gerencia o tráfego.
- Define políticas.
- Usa de modelos para melhorar.

Pontes e Arthaud (2018) também mencionou que essas medidas simples são poderosas para que a equipe de trabalho possa explorar melhor seu potencial e limitações, fornecer indicadores claros de problemas e leva a empresa para as mudanças certas de direção, de forma incremental, destinam-se principalmente a ter um impacto emocional mínimo na organização.

Segundo Vale (2022), com a introdução dos métodos ágeis, a técnica passou a ser utilizada no desenvolvimento de software. Quadros brancos com cartões e notas adesivas são usados para indicar a situação do trabalho em andamento. Por exemplo, quando uma equipe move uma carta para a área de atividades completas, significa que ela está pronta para a próxima etapa. Isso permite a criação de fluxos em iterações e evita que os requisitos sejam enviados para a equipe em massa.

As vantagens do *Kanban* são: permite o controle visual das atividades existentes em andamento ou em fila para iniciar, permite identificar vários problemas no processo de desenvolvimento, promove equipes multifuncionais, incentiva a melhoria contínua (*Kaizen*) e a redução de todos os tipos de desperdício (*Lean*), reduz a burocracia nos métodos de desenvolvimento e por fim, incentiva que as atividades sejam feitas no momento certo (*Just In Time*).

2.1.2 Teste de Usabilidade

No Teste de Usabilidade, o sistema interativo é avaliado a partir da experiência de usuário ao utilizá-lo. A avaliação é feita observando a utilização do sistema pelo usuário e adquirindo métricas mensuráveis, observadas durante a utilização do sistema pelo usuário, que são feitas a partir dos objetivos da avaliação. Essas métricas podem ser a quantidade de cliques que o usuário deu para terminar uma atividade, o tempo que ele levou ou a quantidade de vezes que ele pediu por ajuda. Este teste é feito em um ambiente controlado para que se tenha como

adquirir as medições. Para conduzir um Teste de Usabilidade primeiramente é necessário ter em mente qual o objetivo em fazer este teste e após vem os procedimentos: preparar o teste, coletar os dados, interpretar os dados, consolidar e relatar os resultados (BARBOSA; SILVA, 2010).

Durante a preparação, define-se o perfil dos participantes que farão o teste, o termo de consentimento, as atividades que os participantes devem realizar, os materiais e locais necessários para a realização do teste para a obtenção de dados, e deve ser realizado um teste piloto. Testes piloto são testes que testam testes, ou seja, verificam se o plano de teste estão planejados corretamente.

Na coleta de dados é feita a execução do Teste de Usabilidade, em um ambiente controlado, segundo o planejamento. Os participantes devem ser observados e ter suas ações registradas pelos condutores do estudo, podendo ter auxílio de meios de gravação de vídeo ou áudio para análise posterior.

Na interpretação e consolidação os dados do teste são reunidos e analisados. Utilizam-se gráficos e cálculos estatísticos para apresentar os dados quantitativos e tirar conclusões sobre eles

No relato devem ser apresentados todos detalhes desde o objetivo do porquê está fazendo o teste até os resultados e as interpretações deles.

2.2 Sistema WEB

A *WEB* foi concebida em 1989, buscando criar um meio de troca de dados utilizando a infraestrutura da internet e se caracteriza como uma vasta coleção mundial de documentos, geralmente chamados páginas *WEB* e tendo como principal característica o hipertexto, que significa que cada página pode conter links para outras páginas em qualquer lugar do mundo e os usuários podem seguir esses links através de um simples clique (TANENBAUM; WETHERALL, 2011).

A partir da sua criação, a *WEB* se popularizou e passou em poucos anos a estar presente no cotidiano das pessoas, fazendo com que na maioria das atividades desenvolvidas por um indivíduo, a *WEB* se encontre atuante (VIDOTTI *et al.*, 2019). Se comunicar com outras pessoas, assistir filmes, escutar músicas, ler livros, participar de reuniões de trabalho, tudo isso são exemplos de atividades que hoje em dia podem ser desenvolvidas por um indivíduo através do uso de sistemas *WEB*.

Segundo Noletto (2020), um sistema *WEB* é um tipo de aplicação que pode ser

executado diretamente no navegador de internet do usuário, como, por exemplo, *Google Chrome*, *Mozilla Firefox*, *Microsoft EDGE*, *Safari*, entre outros, não havendo a necessidade de uma instalação daquela aplicação no computador do usuário. Esses navegadores são responsáveis por buscarem a página solicitada, interpretar seu conteúdo e exibir a página, formatada de modo apropriado na tela do dispositivo do usuário (TANENBAUM; WETHERALL, 2011).

A principal diferença conceitual de um sistema *WEB* para um *site*, é que naquele, há uma possibilidade de um usuário ter uma iteração com o sistema, enquanto um *site* tende a ter uma informação estática, sendo apenas um conjunto de páginas que normalmente informam o usuário sobre algo (NOLETO, 2020).

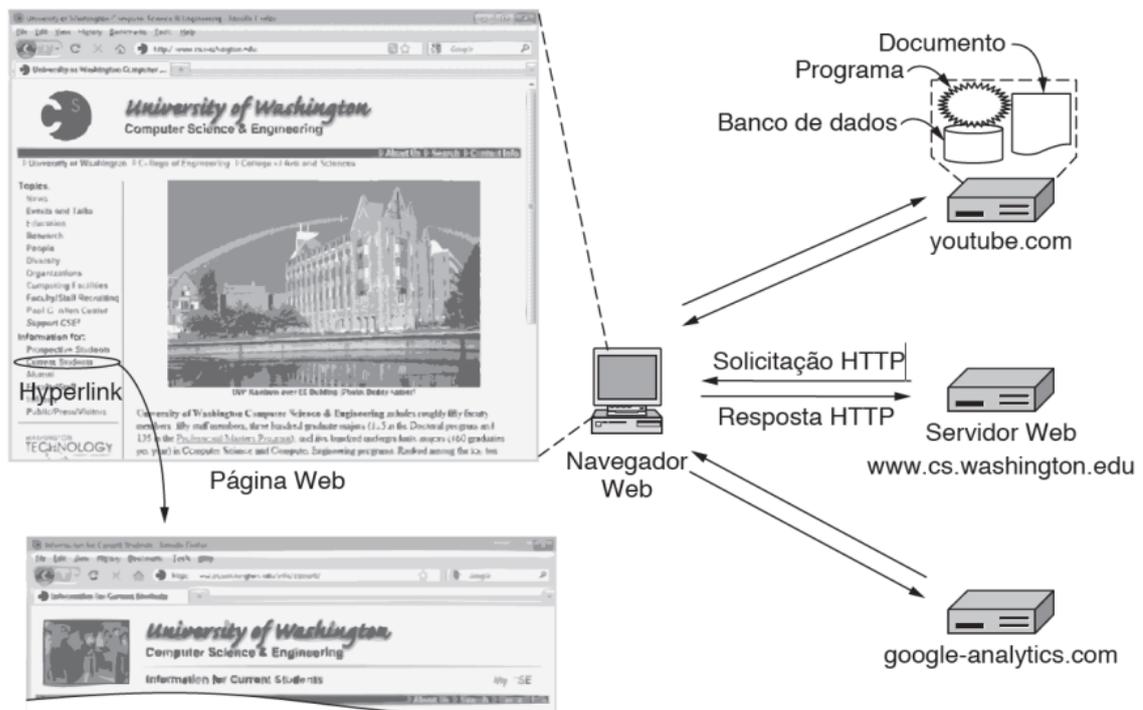
As principais tecnologias necessárias para a construção de um sistema *WEB* são:

- **HTML:** *HyperText Markup Language* (HTML) (Linguagem de marcação de hipertexto), foi introduzida com a *WEB* que permite que sejam produzidas páginas que incluem texto, gráficos, vídeos, links para outras páginas na *WEB* e muito mais. Essa linguagem, por ser de marcação, permite descrever como os documentos (páginas) devem ser formatados. O navegador então é responsável por interpretar os comandos de marcação e aplicá-los ao conteúdo, renderizando o conteúdo de uma forma amigável ao usuário final. (TANENBAUM; WETHERALL, 2011).
- **CSS:** *Cascading Style Sheets* (CSS) (Folhas de estilo em cascata) são estilos que definem a aparência dos elementos *HTML*. Foi criada para dar suporte a *HTML*, separando, de modo adequado, o estilo do conteúdo da página. Os estilos podem ser armazenados em folhas de estilos dentro do próprio arquivo *HTML* ou em folhas de estilos externas: arquivos *CSS*. O nome *Cascading* foi utilizado porque um estilo definido é aplicado para todos os elementos dentro de um documento (CASAS, 2008).
- **Javascript:** linguagem de programação interpretada que é responsável por adicionar interatividade às páginas *HTML* e que são muito utilizados, por exemplo, para a validação de formulários no cliente, antes de serem enviados ao servidor (CASAS, 2008). Com a criação do *Node.js*, agora é possível a execução de *javascript* fora de uma página *WEB*, sendo possível o desenvolvimento de um sistema *WEB* de ponta a ponta - do cliente ao banco de dados - apenas utilizando essa tecnologia.
- **Banco de dados:** responsável por agrupar dados sobre um mesmo assunto, armazenando assim diversos dados como documentos, endereços, serviços, clientes, entre outros. Desta forma, um banco de dados serve para que estes dados sejam armazenados e consumidos

posteriormente. Para seu gerenciamento, é necessário a utilização de um Sistema de Gerenciamento de Banco de Dados (SGBD), são eles os softwares responsáveis por armazenar, listar, pesquisar, remover e qualquer outra manipulação feita em um banco de dados. Existem dois tipos de *SGBD*: Os relacionais e os não relacionais (*NoSQL*). Os relacionais modelam os dados no formato de tabelas, que podem se relacionar entre si, e os não relacionais modelam utilizando diversos modelos de dados, incluindo documentos, gráficos, chave-valor e colunares. (ANDRADE, 2021)

Em um sistema *WEB*, é necessário um servidor para efetuar a comunicação entre as requisições e as respostas a essas requisições, possibilitando que o usuário solicite um recurso, por um navegador, que retorne para ele documentos em formato PDF, imagens e gráficos ou páginas em *HTML* após o servidor encontrar e responder o recurso requisitado (BORGES, 2013). Essa comunicação é realizada através de um protocolo chamado de *Hyper Text Transfer Protocol (HTTP)* e as páginas *HTML* retornadas podem ser estáticas, quando apresentam o mesmo documento toda vez que for exibida, ou dinâmicas, quando a página é gerada sob demanda por um programa ou se contém um programa (TANENBAUM; WETHERALL, 2011). A Figura 1 ilustra esse fluxo geral de funcionamento da *WEB*.

Figura 1 – Fluxo Geral de funcionamento da *WEB*



Fonte: Tanenbaum e Wetherall (2011).

Visando um maior alcance e praticidade no primeiro acesso de tutores e servidores

pet, foi escolhido o uso de sistemas WEB, pois o mesmo, consegue suprir todas as necessidades do projeto.

2.3 NoSQL e Firebase

2.3.1 NoSQL

O termo *Not Only Standard Query Language (NoSQL)* surgiu em 2009 em uma conferência promovida por Johan Oskarsson para discutir bancos de dados não relacionais, distribuídos e de código aberto. O termo foi usado posteriormente para descrever soluções de armazenamento de dados não relacionais (SADALAGE; FOWLER, 2019). O *NoSQL* foi criado devido ao aumento do volume e variedade de dados (CUER, 2015).

Segundo Fernandes *et al.* (2014) a sociedade cada vez mais conectada passa a enviar e receber dados pela Internet, como vídeos, fotos, mensagens de texto, dados do Sistema de Posicionamento Global (GPS), redes sociais, etc., que fazem parte do grande banco de dados não estruturado.

O *NoSQL* pode ser dividido em quatro categorias principais: base de dados chave – valor, base de dados de documentos, base de dados orientadas a colunas e base de dados orientada a gráficos. Onde o primeiro possui uma chave com um valor atribuído a ela, a segunda é usar o método de armazenamento do modo coleção, utilizando documentos padronizados em formatos como *Javascript Object Notation (JSON)*, o terceiro se caracteriza por armazenar e processar um grande volume de dados distribuídos, já o último utiliza um modelo com três componentes, nós, relacionamentos e as propriedades, onde cada nó se conecta com diversas arestas, sendo muito utilizado para realizar consultas complexas (MUNIZ, 2018).

Alguns dos recursos do *NoSQL* incluem: escalabilidade horizontal, sem esquema, suporte a replicação nativa e *API* simples para acesso a dados. A ausência de esquema fornece a conveniência de escalabilidade e maior disponibilidade, mas não garante a integridade dos dados. As soluções *NoSQL* não se preocupam com como os dados são armazenados, mas sim recuperam os dados de forma eficiente para alta disponibilidade e escalabilidade, por isso *APIs* simples são desenvolvidas para facilitar o acesso rápido e eficiente aos dados por qualquer aplicativo (CUER, 2015).

2.3.2 *Firebase*

O *Firebase* é uma plataforma digital para facilitar o desenvolvimento de aplicativos web e mobile, um serviço fornecido pelo *Firebase* chamado *Banking as a Service (BaaS)*. O *Firebase* foi fundado por James Tamplin e Andrew Lee em 2011 e adquirido pelo *Google* *Coodesh* em 2014 (2021).

O *BaaS* é definido por (BATSCHINSKI, 2021) como: "Um serviço de computação em nuvem que serve como *middleware*. O mesmo fornece aos desenvolvedores uma forma para conectar suas aplicações *mobile* e *web* a serviços na nuvem a partir de *APIs* e *SDKs* (BATSCHINSKI, 2021)."

Segundo ANDRADE (2020), o *BaaS* é um tipo de serviço que provê a infraestrutura e o *back-end* de uma aplicação de forma simplificada, assim não é preciso o desenvolvimento dessa solução manualmente. Algumas das funcionalidades mais comuns oferecidas pelo *BaaS* são: autenticação, armazenamento, notificações, escalabilidade e outros. Com isso, os desenvolvedores podem focar apenas na construção do *front-end*.

O *Firebase* possui diversos recursos. Dentre estes, destacamos o *realtime database* e o *cloud storage* que são utilizados nesse trabalho. O *realtime database* é um banco de dados *NoSQL* flexível e escalonável, com sincronia com os aplicativos clientes em tempo real, além de oferecer suporte *off-line*. Já o *cloud storage* é um serviço de armazenagem, fotos e vídeos, simples e econômico criado para a escala do *Google*. Com ele é possível que o usuário final possa fazer *download* e *upload* com a segurança do *Google Firebase*.

Para o desenvolvimento *back-end* da aplicação será empregada um serviço de *serverless*, na qual, foi optado a ferramenta *Firebase* por simplificar tempo e recurso, visando ter seu principal foco no *front-end* abstraindo o *back-end* através de tecnologias *BaaS* com *real time database*.

2.4 Google API

As *APIs* do *Google* são um conjunto de interfaces de programação de aplicativos (*APIs*) desenvolvidas pelo *Google* que permitem a comunicação com os serviços do *Google* e a integração com outros serviços. As *APIs* fornecem recursos como análise, aprendizado de máquina como serviço ou acesso aos dados do usuário (quando recebem permissão para ler os dados). Exemplos desses serviços incluem pesquisa, *Gmail*, tradutor ou *Google Maps*. Aplicati-

vos de terceiros podem usar essas *APIs* para alavancar ou estender a funcionalidade de serviços existentes a fim de agregar valor ao aplicativo, mas em outros casos elas são completamente necessárias para eles, portanto, sem essas *APIs*, o aplicativo não existiria (LECHETA, 2013).

As *APIs* utilizadas no desenvolvimento da ferramenta desenvolvida foram *Google Maps* e *Google Geolocation*.

A *API* do *Google Maps* permite que os usuários utilizem mapas e informações de mapeamento personalizado em suas aplicações, possibilitando aos mesmos explorarem o mundo através dos mapas fornecidos pela *Google*.

2.4.1 *Google Maps*

A *API* do *Google Maps* permite que os usuários usem mapas e informações de mapas personalizados em seus aplicativos, permitindo que eles explorem o mundo por meio de mapas fornecidos pelo *Google*.

Os usuários podem escolher o tipo de visualização do mapa (Satélite, Terreno e Híbrido), personalizá-lo adicionando marcadores no mapa para indicar pontos de interesse especiais e realizar efeitos de câmera como rotação, zoom e inclinação com base na área que podem ser exibidos em mapa tridimensional. O *Google Maps* é gratuito e pode ser usado no aplicativo em qualquer plataforma, entretanto é necessário obter uma chave de autenticação para usar o serviço (MAPS, 2016).

As *Google Maps APIs* são categorizadas por plataforma: *WEB*, *Android* e *iOS*. Essas *APIs* de plataformas nativas são complementadas pelo conjunto de serviços *WEB HTTP* da *Google* (MAPS, 2016). Uma vez que o desenvolvimento da aplicação deste estudo de caso é baseado em tecnologia de plataforma *WEB*, é utilizada a *API JavaScript* do *Google Maps*, que pertence à categoria de *API* de rede. Nesse aspecto da *API*, como em todo o resto, todo o mapa é personalizável. Os estilos personalizados de mapa são definidos no formato *JSON*, portanto, a definição do mesmo mapa é compatível com a *WEB*, *Android* e *iOS*.

Usando o formato *JSON* é possível apresentar o mapa da maneira que quiser, com a capacidade de aplicar estilos dinamicamente ao seu aplicativo em tempo de execução. Além da documentação, o *Google* também fornece o *Style Assistant*, uma plataforma de visualização que gera definições *JSON* de componentes de visualização de mapas (rótulos, cores, tipos de estradas e suas densidades, etc.).

Esse trabalho usará dos mapas fornecidos pela *Google* para mostrar a localização do

passador em tempo real durante o passeio.

2.4.2 Google Geolocation

O *Google Geolocation* é um serviço que fornece aos usuários as informações necessárias para obter a localização de seu dispositivo. Ele fornece uma lista de provedores de localização disponíveis e registra como essas informações de localização são recebidas, definindo a localização por latitude e longitude. O provedor de localização também é responsável por otimizar o uso da bateria do dispositivo.

A *Google Geolocation API* retorna uma localização e um raio de precisão com base em informações de torres de celular e pontos de acesso *Wifi* que o cliente móvel pode detectar (GEOLOCATION, 2016).

A comunicação é realizada por *HTTPS* usando *POST*. Tanto a solicitação quanto a resposta têm formatação *JSON*. Usaremos essas informações para localizar o passageiro durante o passeio.

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados alguns trabalhos relacionados encontrados na literatura. Foram selecionados três trabalhos cujos objetivos se assemelham ao objetivo do *PetWalk*, com aplicações que oferecem serviços voltados à animais domésticos. Por fim, é apresentada uma seção de comparação com as diferenças entre os trabalhos apresentados e a solução proposta.

Como os trabalhos utilizam diferentes termos para representar os fornecedores e os consumidores dos serviços de *pet*, os termos tutores e clientes se referem aos donos de *pet* que atuam como consumidores do sistema e os termos passeador e prestador de serviço se referem às pessoas que oferecem seus serviços através dos sistemas.

3.1 HAPPY PET: Plataforma de contratação de *Pet Sitters*

O trabalho apresentado por Morais e Pinheiro (2021) se passa em um contexto de pandemia na Faculdade de Tecnologia Baixada Santista Rubens Lara no curso de Tecnologia em Sistemas para Internet. O autor, aproveitando-se de um mercado *pet* em constante crescimento, propõe uma aplicação que possibilite que profissionais do ramo de *pet* possam divulgar seus serviços, suas capacitações e gerenciar seus atendimentos. O trabalho apresenta o desenvolvimento de um aplicativo com um *chat* para que os tutores possam conversar diretamente com os profissionais de sua preferência.

A aplicação foi desenvolvido para que os profissionais ofereçam serviços de: banho, tosa, corte de unha, hospedagem, passeio, adestramento e veterinário. Os tutores podem filtrar suas preferências de busca tais como: cidade, animal, porte, tendo como resultado a lista de profissionais que oferecem aquele serviço, ranqueado iniciando com o mais bem avaliado. Ao selecionar um prestador de serviço, o tutor poderá ver suas avaliações e a forma de trabalho que estará na descrição do profissional. Caso se interesse, o tutor poderá entrar em contato com o profissional via *WhatsApp* para trocarem informações de forma mais detalhada sobre o serviço e agendamento.

Por fim, o trabalho apresentou um protótipo que ofereceu as seguintes funcionalidades: filtrar, visualizar, adicionar e excluir serviço, adicionar, visualizar e excluir *pet* e editar perfil. Foram realizados testes de usabilidade com 6 participantes. Todos os participantes afirmaram conseguir encontrar o que queriam e que se sentiram no controle durante a navegação no aplicativo.

3.2 Passeios Disruptivos: Um Passo para o Futuro

No trabalho de Arruda e Facchini (2021) é apresentada a ideia de uma aplicação chamada *Dogger* voltada ao passeio com animais domésticos. Nele, é abordada a necessidade da sociedade para esse tipo de serviço, a necessidade de uma tecnologia disruptiva para o serviço e o que a mesma pode proporcionar.

Para validar a relevância da solução proposta, é relatado e apresentado o atual papel do cachorro como animal doméstico nos locais em que ele reside. No trabalho, é usado de contextos históricos para explicar a atual tendência em que cachorros e gatos estão cada vez mais ocupando um espaço de relevância como um membro da família.

Com essa contextualização, é introduzido o tema da saúde desses animais, sendo ressaltada a importância do passeio no cotidiano dos mesmos para uma agregação na qualidade de vida, usando-se de citações que expõem as necessidades físicas e psicológicas em torno do cotidiano canino.

Arruda e Facchini (2021) fez uma pesquisa para validar a relevância da aplicação usando de análises quantitativas. Na mesma, ele consta o número de passeios diários dos tutores com os seus *pet*, se os tutores estão satisfeitos com as quantidades de passeios diários do seu *pet*, qual o porte desses animais, se o *pet* só faz suas necessidades fora de casa, entre outras perguntas-chave para a validação de uma oportunidade de aplicação da tecnologia para auxiliar esses passeios.

Arruda e Facchini (2021) defende a importância do passeio diário. Então é introduzida a ideia do uso de uma tecnologia disruptiva para mudar um cenário da prestação de serviços de passeios para *pets*. Assim, o aplicativo *Dogger* vem para solucionar esse problema ao oferecer a oportunidade de o responsável pelo cachorro solicitar passeios pelo aplicativo, independentemente de onde estiver, quer seja no seu trabalho ou em viagem. E para isso, necessita apenas haver alguém que entregue o seu cachorro para o passeador.

O modelo do aplicativo *Dogger* se parece ao operacional com o da *Uber*. O serviço de geolocalização agrega a segurança e a garantia que o passeio está sendo feito conforme o combinado, mostrando a localização de todo percurso do passeio com o *pet*. A forma de pagamento é via aplicativo utilizando cartão de crédito ou comprando créditos no aplicativo via boleto bancário. O trabalho não realiza a implementação de protótipo, apenas a idealização dessas funcionalidades.

3.3 Pet Walk

O projeto de Castillo *et al.* (2021), busca mostrar um aplicativo de celular de sucesso que oferece o serviço de caminhada e treinamento canino. O público alvo destacado são os estudantes universitários que estudam e trabalham e os idosos que trabalham e não têm tempo para passear com seus animais de estimação. Como diferencial, eles buscam se destacar de seus concorrentes com um atendimento de qualidade, utilizando um sistema de geolocalização que permite saber onde estão seus cães em tempo real como diferencial tecnológico.

Para isso, foram realizadas pesquisas e experimentos que permitiram conhecer as necessidades do usuário que poderiam contratar o serviço. Foi realizada uma análise financeira para mostrar que o projeto também é rentável e tem grande potencial de crescimento no futuro.

O trabalho foca na elaboração de uma documentação que demonstre a viabilidade da aplicação. Sua finalização, são apresentadas as seguintes conclusões e recomendações:

- Usuários são mais atraídos por um aplicativo se ele oferece maior segurança em termos de serviços e dados pessoais.
- Existem espaços publicitários melhores que outros para esse tipo de negócio. Redes sociais apresentaram melhor retorno em vendas do que *e-mail* ou publicidade do *Google*.
- Um sistema *gamificado* pode ser implementado dentro do aplicativo que para aumentar retenção de clientes por meio de benefícios para ele, gerando assim mais fidelidade e mais recomendações.

O trabalho não realiza a implementação de protótipo, apenas a idealização dessas funcionalidades.

3.4 Análise Comparativa

Este capítulo apresenta uma análise comparativa entre os trabalhos apresentados neste capítulo e a solução proposta neste trabalho. A Quadro 1 apresenta o resumo dessa análise.

Quadro 1 – Comparação entre trabalhos

Autor	Trabalho	Desenvolve uma aplicação	Idealiza funcionalidades de passeio	Funcionalidade da geolocalização em tempo real	Funcionalidade do chat
(MORAIS; PINHEIRO, 2021)	Happy pet	React Native	Não idealiza	Não oferece	Não oferece
(ARRUDA; FACCHINI, 2021)	Dogger	Não desenvolve	Geolocalização e chat	Via Whatsapp	Via Whatsapp
(CASTILLO <i>et al.</i> , 2021)	Petwalk	Não desenvolve	Geolocalização	Não implementa	Não oferece
Autor	Presente trabalho	React	Geolocalização e chat	Funcionalidade interna	Funcionalidade interna

Fonte: Elaborado pelo autor.

Os trabalhos apresentados tem em comum a ideia de um aplicativo de reserva de serviços voltados ao público *pet*. Cada um desses trabalhos, apesar de resolverem problemas parecidos, usam diferentes abordagens, soluções com diferentes funcionalidades, sendo algumas delas semelhantes aos do presente trabalho. Dessa forma foram definidos alguns critérios de comparação entre esses trabalhos e o presente trabalho:

- **Desenvolve uma aplicação:** utilizado para verificar se o trabalho apresenta uma aplicação como resultado.
- **Idealiza a funcionalidade de geolocalização em tempo real:** utilizado para identificar se o trabalho apresenta a ideia de geolocalização durante a prestação do serviço de passeio.
- **Implementa a geolocalização em tempo real:** utilizado para verificar se o trabalho entrega a implementação de um serviço de geolocalização.
- **Implementa um chat interno:** Utilizado para verificar se o trabalho entrega a implementação de um chat interno a aplicação.
- **Válida a ideia de uma aplicação pet:** utilizado para verificar se o trabalho faz uma validação de ideia proposta.

O trabalho de Morais e Pinheiro (2021) e o presente trabalho desenvolvem uma aplicação. Com exceção do trabalho de Arruda e Facchini (2021) e Castillo *et al.* (2021) cujo trabalho apresenta a ideia de uma aplicação e sua validação.

Com exceção do trabalho do Morais e Pinheiro (2021), os demais trabalhos, assim como o presente trabalho, apresentam a idealização da funcionalidade de geolocalização para garantir maior segurança aos tutores de *pet*.

Diferentemente dos trabalhos de Morais e Pinheiro (2021), Arruda e Facchini (2021) e Castillo *et al.* (2021), este trabalho apresenta a implementação da funcionalidade de geolocali-

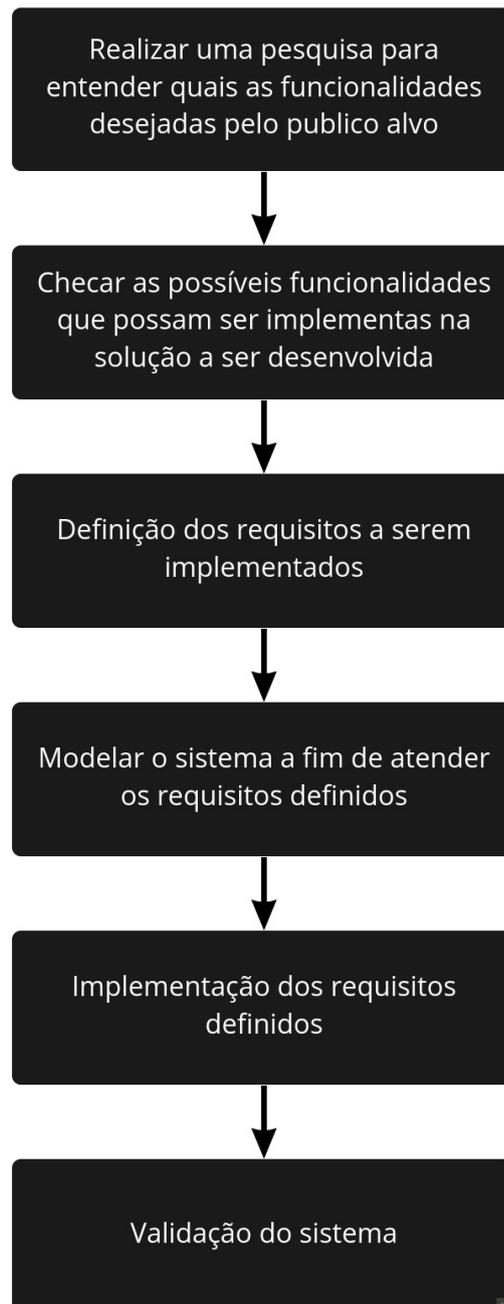
zação em tempo real para promover maior conforto e segurança durante a prestação do serviço de passeio.

Por fim, é importante destacar que o presente trabalho visa apresentar uma aplicação voltada a prestação de serviços de passeios, e para isso busca implementar funcionalidades que possam trazer segurança e reter os usuários na plataforma proposta. Apesar de não ser a única que idealizou um chat interno, é o único que implementa a funcionalidade que é de suma importância para tornar a aplicação independente, facilitar a permanência do usuário na plataforma e proporcionar escalabilidade de futuras funcionalidades que agregam na qualidade do serviço

4 METODOLOGIA

Neste capítulo são apresentadas as etapas necessárias para a execução do trabalho, além de um planejamento de condução dessas tarefas, para que a solução presente nesse trabalho possa ser modelada, desenvolvida e validada. A figura 2 apresenta a sequência das etapas.

Figura 2 – Passos metodológicos



Fonte: Elaborado pelo autor.

4.1 Realizar uma pesquisa para entender quais as funcionalidades desejadas pelo público alvo

Sabendo que o serviço de passeio existe sendo oferecido por aplicações, a primeira grande tarefa será a de pesquisar quais soluções outras plataformas utilizam para a prestação do serviço em questão, de modo que seja analisado pontos como:

- Quais estratégias adotadas para garantir a segurança do tutor durante o serviço.
- Quais as principais funcionalidades requeridas nesse meio.
- Como funciona o fluxo para realizar a reserva de um passeio.

Além disso, vamos realizar uma pesquisa em *petshops* e parques da região a fim de encontrar tutores, passeadores e profissionais da área e realizar entrevistas e passar formulários para entender quais as reais necessidades do público em questão.

4.2 Checar as possíveis funcionalidades que possam ser implementadas na solução a ser desenvolvida

Uma vez que as soluções utilizadas em outros aplicativos tiverem sido levantadas, a próxima etapa é realizar uma análise dessas soluções visando identificar possíveis funcionalidades que possam ser aplicáveis no contexto da PetWalk. Essas funcionalidades serão listadas para serem validadas nas etapas seguintes.

4.3 Definição dos requisitos a serem implementados

Nessa etapa é realizada a junção das funcionalidades úteis e aplicáveis que foram identificadas na etapa anterior, com aquelas que serão identificadas com as partes interessadas envolvidas no processo de levantamento de requisitos. Os participantes no processo de definição de requisitos são o time de produção da PetWalk, professores, orientadores, tutores de pet e passeadores.

4.4 Modelar o sistema a fim de atender os requisitos definidos

Com o conjunto de funcionalidades definidas, o próximo passo é desenhar uma arquitetura de sistema que atenda essas funcionalidades através da realização de uma prototipação do sistema, criação de diagramas *UML*, escrita de um documento de requisitos e realização da

modelagem de banco de dados.

4.5 Implementação dos requisitos definidos

Uma vez que a modelagem estiver pronta, a próxima etapa é a de implementação do sistema com base nesses artefatos gerados na etapa anterior. Conforme fundamentado no Capítulo 3, é implementado um sistema *WEB*.

4.6 Validação do sistema

Após a primeira etapa de prototipagem, teremos um produto minimamente funcional apto a ser avaliado. Esse produto, conhecido por Produto Mínimo Viável ou (*MVP*), permitirá a validação das partes interessadas envolvidas no processo (tutores de pet e passeadores), para se identificar se a solução desenvolvida é apropriada e orientar as etapas futuras do projeto.

5 DESENVOLVIMENTO

5.1 Pesquisa

Neste tópico é apresentada a metodologia utilizada durante a fundamentação do projeto. Para isso foram realizadas duas pesquisas. Uma por formulários para entender melhoras o público na região e um estudo da arte com duas aplicações de serviços *pet*.

5.1.1 Formulários

Para criação dos formulários foi necessário traçar seus objetivos. Que para o presente trabalho foi: entender como os tutores e prestadores de serviços *pet* atendem as necessidades dos animais atualmente, as necessidades desse público e se existe demanda para esse tipo de serviço. Tendo em mãos os objetivos, foi feito um formulário que serviu de apoio para validar as percepções recolhidas. Para fazer parte do grupo apto a participar da pesquisa, foi exigido os seguintes pré-requisitos: ser tutor de *pet* ou atuar no mercado *pet*.

Foi utilizado a ferramenta Google Forms como um centralizador das cento e cinquenta respostas obtidas. Inicialmente, foi realizado um trabalho em *petshops*, clínicas veterinárias e parques, onde passamos o formulário e entrevistamos pessoas do mundo *pet* presencialmente para entender como os mesmos reagem às perguntas a fim de identificar pontos de melhoria no formulário e ter percepções de melhorias nas perguntas.

5.1.1.1 Pesquisa

Nesse tópico são listados as sete perguntas e um breve análise das respostas obtidas através do uso do formulário.

Na Figura 3 é apresentada a primeira pergunta do formulário. O objetivo dessa pergunta é entender por onde os tutores buscam por serviços *pet* atualmente. E foi constatado que 69,8% dos tutores estão a merce de indicações em *petshops* e clínicas veterinárias. Além disso, o gráfico aponta que 53,7% dos tutores usam de rede social para encontrar prestadores de serviços *pet*.

Na Figura 4 é apresentada a segunda pergunta do formulário, criada visando entender quais só serviços *pet* mais relevantes para os tutores. Foi constatado que a maior preocupação do tutor é a vacinação, estando em 88,6% das respostas obtidas. No entanto, vale destacar que

passeio apareceu com 55,7%. Lazer, banho e higiene apareceram com 69,8%.

Na Figura 5 é apresentada a terceira pergunta do formulário. Nela buscamos entender se os tutores conseguem identificar as consequências da negligência das necessidades dos seus animais de estimação. Foi constatado que o comportamento de hiperatividade aparece em 41,3% das respostas. E esse comportamento é um dos principais sintomas da negligência dos passeios diários.

Na Figura 6 é apresentada a quarta pergunta do formulário. Nela buscamos entender a quem os tutores costumam recorrer quando precisão de suporte nos cuidados dos seus animais de estimação. Confirmando que 71,2% dos tutores recorrem a parentes pedindo ajuda quando vão viajar. Caso não tenham esse suporte, os tutores levam o animal e quando não podem viajar com seu bichinho, os mesmos acabam desistindo da viagem. Das respostas, 54,3% dos tutores ou levam o animal, ou não viajam.

Na Figura 7 é apresentada a quinta pergunta do formulário. Nela buscamos entender quais as principais demandas de serviços para *pets*. Foi constatado que o serviço de veterinária se destacou com 76% das respostas. No entanto, os serviços de passeio, veterinária, babá pet, adestramento, hotelaria *pet*, salão *pet* também têm demandas relevantes.

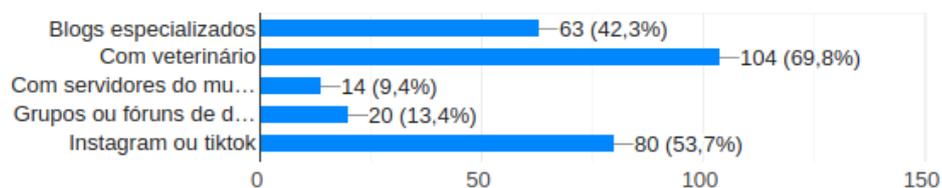
Na Figura 8 é apresentada a sexta pergunta do formulário. Esta pergunta foi criada com o intuito de entender se atualmente os tutores têm acesso a serviços pet. Nela podemos constatar que a maioria das respostas foi direcionada a serviços comuns em clínicas e *petsshops*. Desta forma, podemos afirmar que os serviços *pet* propostos pela *petwalk* ainda é um serviço emergente na região da pesquisa.

Na Figura 9 é apresentada a sétima pergunta do formulário. Seu objetivo foi compreender quais funcionalidades de segurança podem ser agregadas em nosso aplicativo. E com suas respostas pudemos perceber que chamada de vídeo e geolocalização trazem conforto ao tutor.

Figura 3 – pergunta 1

Por quais meios você obtém informações sobre pets? (pode marcar mais de um)

149 respostas

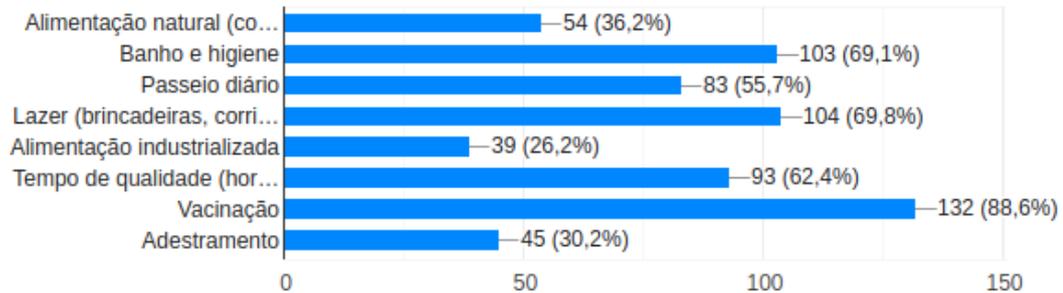


Fonte: Elaborado pelo autor.

Figura 4 – pergunta 2

Das opções abaixo, quais você considera prioridade no cuidado do seu pet? (pode marcar mais de um)

149 respostas

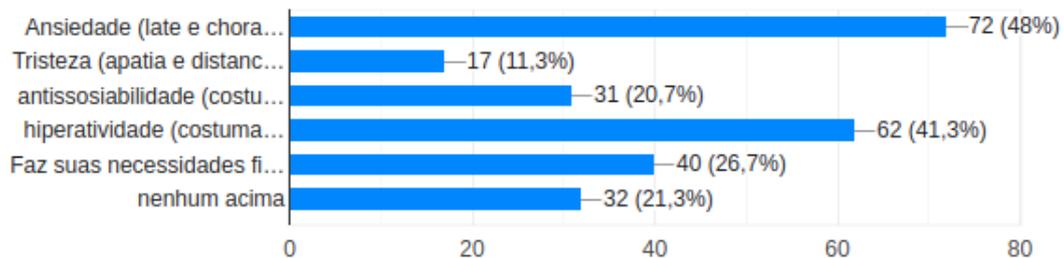


Fonte: Elaborado pelo autor.

Figura 5 – pergunta 3

Você consegue identificar algum desses comportamentos no seu Doguinho? (pode marcar mais de um)

150 respostas

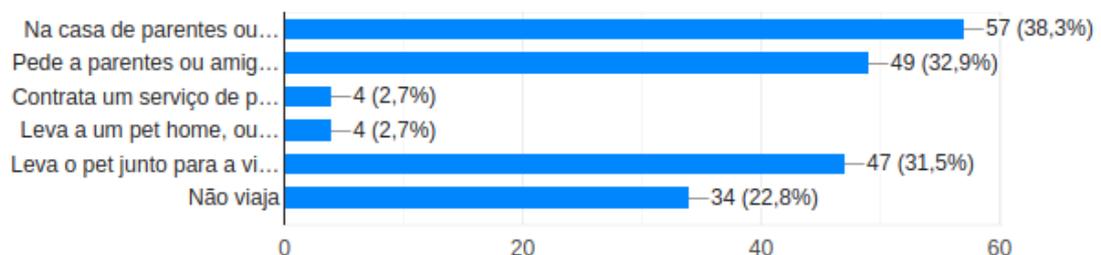


Fonte: Elaborado pelo autor.

Figura 6 – pergunta 4

Quando você viaja, como você costuma deixar seu pet? (pode marcar mais de um)

149 respostas

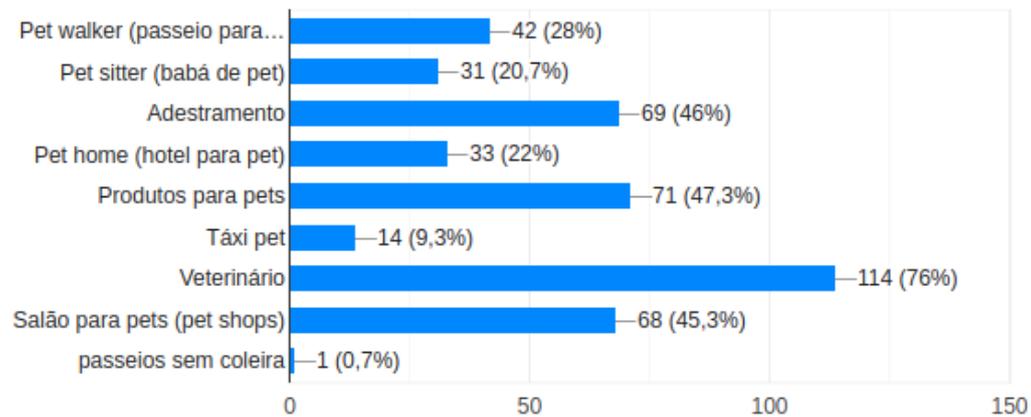


Fonte: Elaborado pelo autor.

Figura 7 – pergunta 5

Que serviços você contrataria para aumentar seu conforto e a qualidade de vida dos seus bichinhos? (pode marcar mais de um)

150 respostas

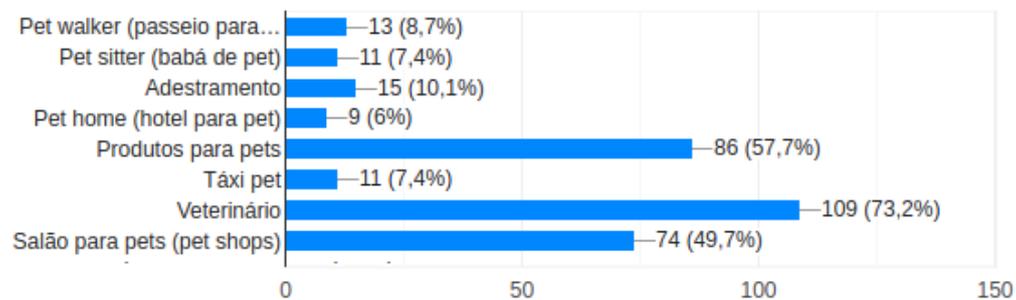


Fonte: Elaborado pelo autor.

Figura 8 – pergunta 6

Quais desses serviços você tem acesso fácil? (pode marcar mais de um)

149 respostas

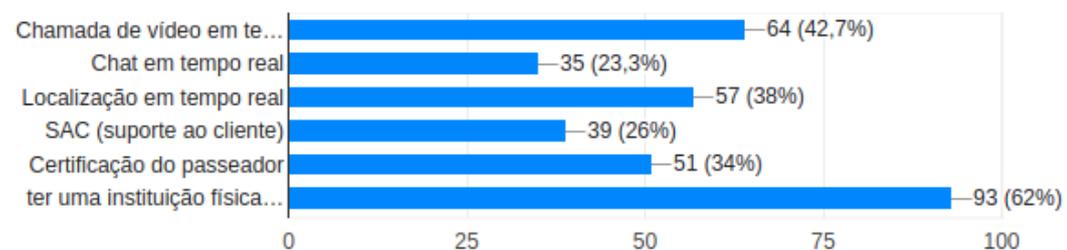


Fonte: Elaborado pelo autor.

Figura 9 – pergunta 7

O que deixa você mais seguro ou confortável ao receber um serviço para seu peludinho?

150 respostas



Fonte: Elaborado pelo autor.

5.1.1.2 *Estudo da arte*

Utilizamos como base duas plataformas encontradas que possuem ideias semelhantes à nossa, a Pet Anjo e Dog Hero.

A plataforma Pet Anjo serve para contratação de passeadores e de outros serviços como baba, hospedagem, creche, adestramento, banho e tosa, veterinário entre outros. E a Dog Hero é outra plataforma de contratação de passeadores e de outros serviços, como baba, hospedagem, creche e veterinário.

Foi feito uma análise das principais funcionalidades, tais como o fluxo de reserva, cadastro pet, e página de perfil, entre outros recursos oferecidos em aplicações de serviços pet. A partir da análise nas aplicações semelhantes, foi feita uma análise minuciosa para entender quais funcionalidades são fundamentais para criação de um produto viável mínimo.

5.2 **Análise do sistema**

Neste tópico são apresentadas todas as etapas e processos realizados durante o desenvolvimento do projeto.

5.2.1 *Análise dos requisitos*

Nesse tópico são listados os requisitos referentes ao usuário do tipo Tutor e Passeador. Os requisitos foram ordenados por ordem de implementação.

5.2.1.1 *Requisitos referentes ao usuário do tipo Tutor*

- **[RF01] Cadastrar tutor:** O sistema deverá permitir que o tutor realize cadastro informando seus dados pessoais tais como: nome, cidade, endereço, e-mail, senha, telefone, entre outros.
- **[RF02] Login tutor:** O sistema deverá permitir que o tutor realize login informando seus dados pessoais tais como: e-mail e senha ou com sua conta Google.
- **[RF03] Filtrar por região:** O sistema deverá permitir que o tutor filtre a lista de passeadores de uma região.
- **[RF04] Visualizar passeador:** O sistema deverá permitir que o tutor visualize todos os passeadores resultantes do filtro.

- **[RF05] Selecionar passeadores:** O sistema deverá permitir que o tutor selecione determinado perfil de passeador de seu interesse.
- **[RF06] Visualizar perfil de passeador selecionado:** O sistema deverá permitir que o tutor visualize suas informações do passeado.
- **[RF07] Realizar uma reserva de passeio:** O sistema deverá permitir que o tutor informe todos os dados necessários para marcar uma reserva com o passeador.
- **[RF08] Avaliar passeador:** O sistema deverá permitir que o tutor avalie o perfil do passeador de 0 a 5 estrelas e escrever um comentário se desejar.
- **[RF09] Acompanhar serviço através da geolocalização:** O sistema deverá permitir que o tutor conseguir acompanhar o passeio através da geolocalização durante o passeio.
- **[RF10] Adicionar pet:** O sistema deverá permitir que o tutor adicione seus pets.
- **[RF11] Exibir perfil do pet:** O sistema deverá permitir que o tutor veja os dados dos pets.
- **[RF12] Excluir pet:** O sistema deverá permitir que o tutor exclua seus pets.

5.2.1.2 *Requisitos referentes ao usuário do tipo Passeador*

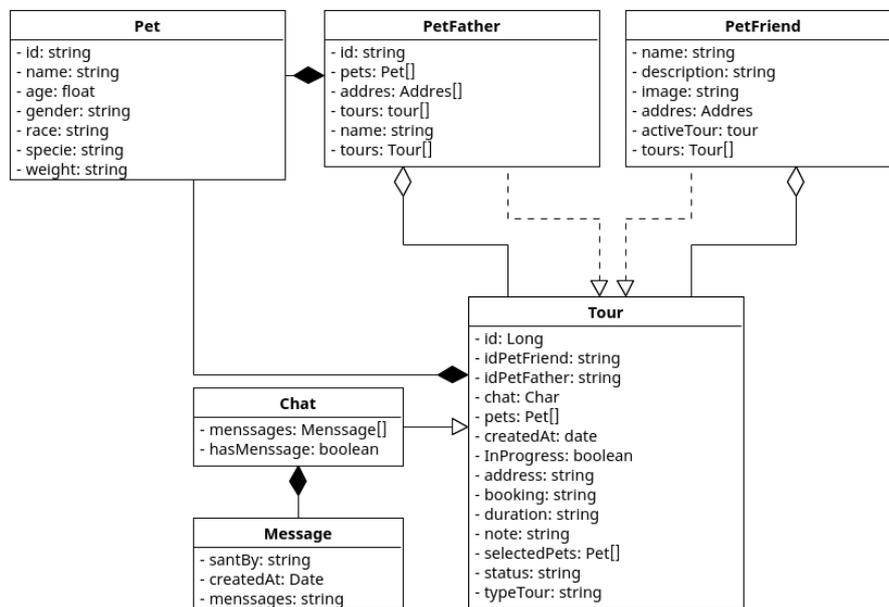
- **[RF01] Cadastrar passeador:** O sistema deverá permitir que o passeador realize cadastro informando seus dados pessoais tais como: nome, cidade, endereço, e-mail, senha, telefone, entre outros.
- **[RF02] Login passeador:** O sistema deverá permitir que o passeador realize login informando seus dados pessoais tais como: e-mail e senha ou com sua conta Google.
- **[RF03] Visualizar serviço:** O sistema deverá permitir que o passeador visualize todos os seus serviços requisitados a ele.
- **[RF04] Responder requisições de serviços:** O sistema deverá permitir que o passeador possa aceitar ou negar serviços solicitados a ele.
- **[RF05] Iniciar um serviço:** O sistema deve permitir que o passeador possa iniciar um serviço.
- **[RF06] Compartilhar sua localização durante um serviço:** O sistema deve permitir que o passeador compartilhe sua localização durante o serviço.
- **[RF07] Editar perfil:** O sistema deverá permitir que o passeador modifique seus dados pessoais tais como: descrição, fotos, informações pessoais entre outros.
- **[RF08] Histórico de passeio:** O sistema deverá permitir que o passeador consiga ter acesso a um histórico de passeio realizados em sua conta.

5.3 Banco de dados

Neste t3pico ser3o apresentados os relacionamentos das tabelas do banco de dados do aplicativo. A Figura 10 apresenta o diagrama *UML* do aplicativo PetWalk, o digrama permite visualizar como as informa33es ser3o armazenadas no sistema.

As principais tabelas s3o Pet, os PetFather (Tutor), PetFriend (Passeador), Tour(Caminho a ser percorrido), Chat e Message (guardam as trocas de dados entre passeador e tutor).

Figura 10 – UML do banco de dados



Fonte: Elaborado pelo autor.

5.4 Camada de neg3cios

Neste t3pico ser3o apresentados os c3digos desenvolvidos para trabalhar com a *API* do *Firebase Realtime Database* no aplicativo. Os c3digos *Javascript* das abstra33es das principais funcionalidades s3o apresentados aqui para mostrar como a intera333o entre as tecnologias funciona.

- A Figura 11 apresenta a rota para consultar no banco as informa333es do usu3rio do tipo tutor baseado no id.
- A Figura 12 apresenta a rota para consultar no banco as informa333es do usu3rio do tipo passeador baseado no id.
- A Figura 13 apresenta a rota utilizada para fazer o cadastro de endere3o enviando via *JSON* os dados necess3rios para armazenar o endere3o no banco.

- A Figura 14 apresentada a rota utilizada para fazer o cadastro de um pet enviando via JSON os dados necessários para armazenar o pet no banco.
- A Figura 15 apresenta a rota utilizada para criar um serviço enviando um JSON com as informações necessárias de um serviço.
- A Figura 16 apresenta a rota utilizada para consultar dentro do banco os serviços daquele usuário do tipo tutor, baseado no seu id.
- A Figura 17 apresenta a rota utilizada para consultar dentro do banco, todas mensagens de um chat baseado no id do serviço.
- Por fim, a figura 18 apresenta a rota utilizada para enviar uma mensagem para o chat de um passeio.

Figura 11 – Rota de pegar o usuário do tipo tutor baseado no id.

```
function getPetFather(id: string) {
  const db = getDatabase();
  get(ref(db, `petFather/${id}`))
    .then((snapshot) => {
      if (snapshot.exists()) {
        setUser(snapshot.val());
      } else {
        console.log("No data available");
      }
    })
    .catch((error) => {
      console.error(error);
    });
}
```

Fonte: Elaborado pelo autor.

Figura 12 – Rota de pegar o usuário do tipo passeador baseado no id.

```
function getPetFriend(id: string) {
  const db = getDatabase();
  get(ref(db, `petFriend/${id}`))
    .then((snapshot) => {
      if (snapshot.exists()) {
        let petFriend = snapshot.val();
        setPetFriend(petFriend);
      } else {
        console.log("No data available");
      }
    })
    .catch((error) => {
      console.error(error);
    });
}
```

Fonte: Elaborado pelo autor.

Figura 13 – Rota para salvar um endereço.

```
function registerAddress(address: Address, callback: ()=>{}) {
  const db = getDatabase();
  set(
    ref(db, "petFather/" + getIdUser(data) + "/address/" + id),
    address
  ).then(() => {
    callback();
  });
}
```

Fonte: Elaborado pelo autor.

Figura 14 – Rota para salvar um pet.

```
export function registerPet(pet: PetProps, idPetFather: string) {
  const db = getDatabase();
  set(ref(db, "petFather/" + idPetFather + "/pets/" + pet.id),
    pet
  ).then(() => {
    handleBack();
  });
}
```

Fonte: Elaborado pelo autor.

Figura 15 – Rota para criar uma reserva.

```
export function handleReservation(reservation: ReservationProps) {
  const id = uuidv4();
  const db = getDatabase();
  const { idPetFather, idPetFriend, data } = reservation;
  set(ref(db, "tours/" + id), {
    idPetFather,
    idPetFriend,
    data,
  }).then(() => {
    push(ref(db, "petFather/" + idPetFather + "/tours"), {id});
    push(ref(db, "petFriend/" + idPetFriend + "/tours"), {id});
    Router.push("/tour/" + id);
  });
}
```

Fonte: Elaborado pelo autor.

Figura 16 – Rota para consultar o histórico de reserva

```
function getHistoric(id: string) {
  if (!id) return;
  let dataTours: ReservationProps[] = [];
  const db = getDatabase();
  get(ref(db, `petFather/${id}/tours`))
    .then((snapshot) => {
      if (snapshot.exists()) {
        dataTours(snapshot.val())
      } else {
        console.log("No data available");
      }
    })
    .catch((error) => {
      console.error(error);
    });
}
```

Fonte: Elaborado pelo autor.

Figura 17 – Rota para consultar mensagens de um chat a partir do id de passeio

```
function getMessages() {
  const db = getDatabase();
  onValue(ref(db, "chats/" + idTour + "/messages"), (snapshot) => {
    if (snapshot.exists()) {
      setMessages(
        Object.keys(snapshot.val()).map((key) => {
          return snapshot.val()[key] as Message;
        })
      );
    }
  });
}
```

Fonte: Elaborado pelo autor.

Figura 18 – Rota para enviar uma mensagem

```
function getMessages() {
  const db = getDatabase();
  onValue(ref(db, "chats/" + idTour + "/messages"), (snapshot) => {
    if (snapshot.exists()) {
      setMessages(
        Object.keys(snapshot.val()).map((key) => {
          return snapshot.val()[key] as Message;
        })
      );
    }
  });
}
```

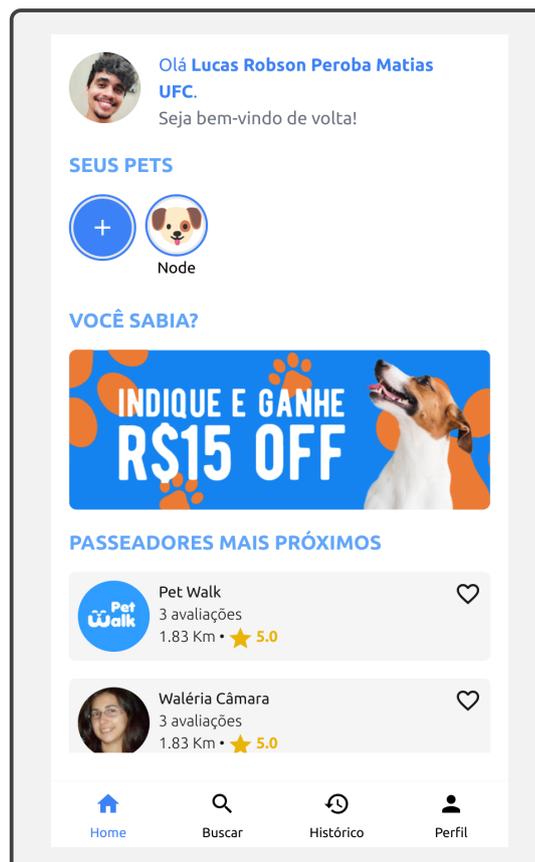
Fonte: Elaborado pelo autor.

5.5 Camada de apresentação

Neste tópico serão apresentadas as principais telas com suas funcionalidades para o funcionamento do aplicativo. Para o seu desenvolvimento, foi utilizado um misto de referências advindas de aplicações atuantes no mercado. E para isso foi realizado uma pesquisa para entender os fluxos e funcionalidades em aplicativos de serviços pet.

A Figura 19 apresenta a tela inicial do tutor. Ela apresenta brevemente os pets que ele já possui cadastrados em seu perfil, opção de cadastrar novos pets, um cupom de indicação para indicar o aplicativo, uma lista de passeadores para serem selecionados e um menu inferior podendo levar às opções **Home**, **Buscar**, **Histórico** e **Perfil**. **Home** sempre leva o usuário de volta para tela inicial, **Buscar** permite ao usuário procurar por um passeador, **Histórico** mostra todos os passeios já realizados e **Perfil** permite ao usuário visualizar e modificar o próprio perfil.

Figura 19 – Home



Fonte: Elaborado pelo autor.

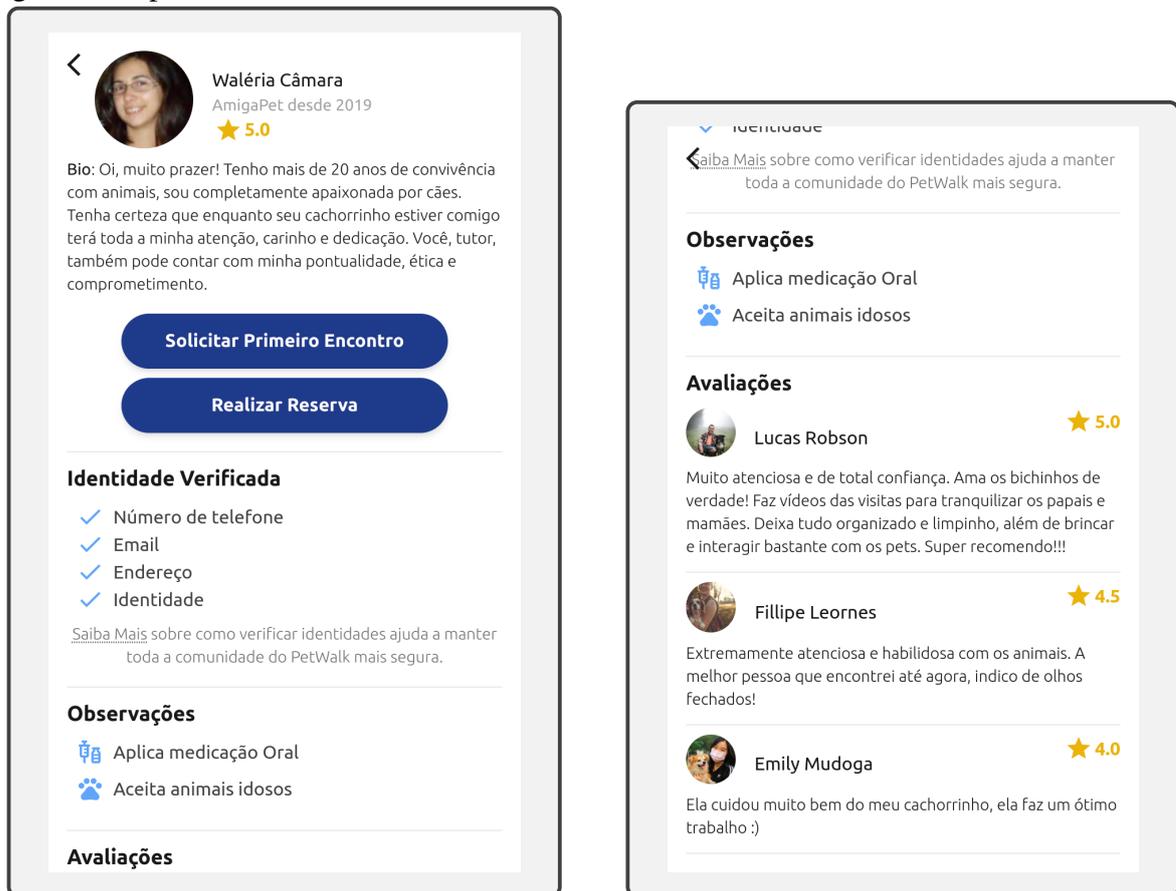
O link para indicação foi inserido tanto para melhorar a composição visual da tela como para criar um espaço para divulgar informações e promoções vinculadas ao aplicativo. Atualmente, o aplicativo está mostrando na tela inicial todos os passeadores cadastrados no sistema.

Posteriormente, serão apresentados na tela inicial apenas os passeadores que se cadastraram para aquela área específica de atuação.

5.5.1 Perfil do passeador

As Figuras 20a e 20b apresentam a tela das informações do passeador selecionado. Como apresentado nos trabalhos relacionados, a confiabilidade é um dos valores críticos para a implementação de um serviço de cuidados de *Pet*. Com base nisso, a tela de apresentação de perfil do passeador busca apresentar elementos críticos que gerem segurança ao tutor, tais como as camadas de validação da identidade do passeador, sendo atualmente feitas de forma manual, e as avaliações dos últimos passeios.

Figura 20 – perfilPetWalker



(a) Elaborado pelo autor.

(b) Elaborado pelo autor.

As avaliações visualizadas são obtidas através das melhores avaliações realizadas pelos tutores após a finalização dos passeios. Não existe ainda nenhum critério ou ferramenta para filtrar ou excluir avaliações ruins ou depreciativas, devendo tais remoções serem feitas, se necessário, diretamente pela equipe de gestão da ferramenta. As observações atualmente são

apenas de cunho ilustrativo e futuramente são ser implementadas de maneira a ser determinadas por um *checklist* marcado pelo passeador na hora do cadastro.

5.5.2 Reserva

A Figura 21a e 21b são apresentadas a tela inicial do fluxo de reserva. Nela o tutor escolhe se vai querer apenas um serviço avulso ou semanal, em seguida a perguntado quem vai participar do passeio, podendo escolher um *pet* ou mais de um. Caso o tutor não tenha nenhum pet cadastrado, ele é obrigado a cadastrar um pet para passar dessa etapa da reserva. Na escolha do tipo de passeio, ele pode escolher entre marcar um passeio avulso ou fazer uma reserva semanal. O passeio avulso é útil quando é o primeiro contato entre tutor e passeador, ou quando o tutor quer testar ou conhecer a ferramenta. O passeio semanal deixa as informações registradas e cria um evento recorrente.

Figura 21 – Início da reserva



A Figura 21 b apresenta o segundo passo da reserva, nela é perguntado se o tutor

tem alguma observação sobre sua reserva. Nesse campo podem ser inseridas informações sobre preferências ou problemas do *pet*, sugestões de trajeto ou tratamento, entre outros.

A Figura 22 *a* apresenta o terceiro passo da reserva, nela é perguntado quanto tempo de passeio, a data e a hora do mesmo. O preço é apresentado nessa hora e está vinculado à duração dos passeios. Atualmente, o preço está implementado de forma fixa, apenas relacionado à duração do passeio, não estando vinculado à região, à demanda, ou ao passeador escolhido. No futuro, semelhante ao que acontece em outros aplicativos de corrida, deve ser possível estudar melhor os modelos de negócio e criar um preço mais orgânico e realista que se adapte as diversas variáveis que regulam o mercado.

Ainda trabalhando o valor de confiabilidade, nessa tela são apresentadas instruções sugerindo ao tutor que aproveite os primeiros passeios para ir com o passeador, conhecê-lo, ver como é o tratamento oferecido por ele ao seu *pet*. A Figura 22 *b* apresenta o quarto passo da reserva, nela é perguntado onde o passeador deve ir buscar o pet.

Figura 22 – Processo de reserva

Passeio
Aproveite este passeio para conhecer o PetWalker. Seu pet vai adorar!

Duração do passeio?

30 Minutos.
R\$: 19,90 por passeio.

60 Minutos.
R\$: 26,90 por passeio.

Quando será o passeio?

dd/mm/aaaa --:-- 📅

[+ Adicionar outro passeio.](#)

Você pode aproveitar esse passeio para conhecer melhor um passeador e conversar sobre detalhes desse passeio e talvez outros futuros.

Próximo

Voltar

Onde o passeador deve buscar seu pet para o passeio ?

📍 Rua caetano alvares, 123
Centro, Fortaleza/CE

Cadastrar Outro Endereço

Próximo

Voltar

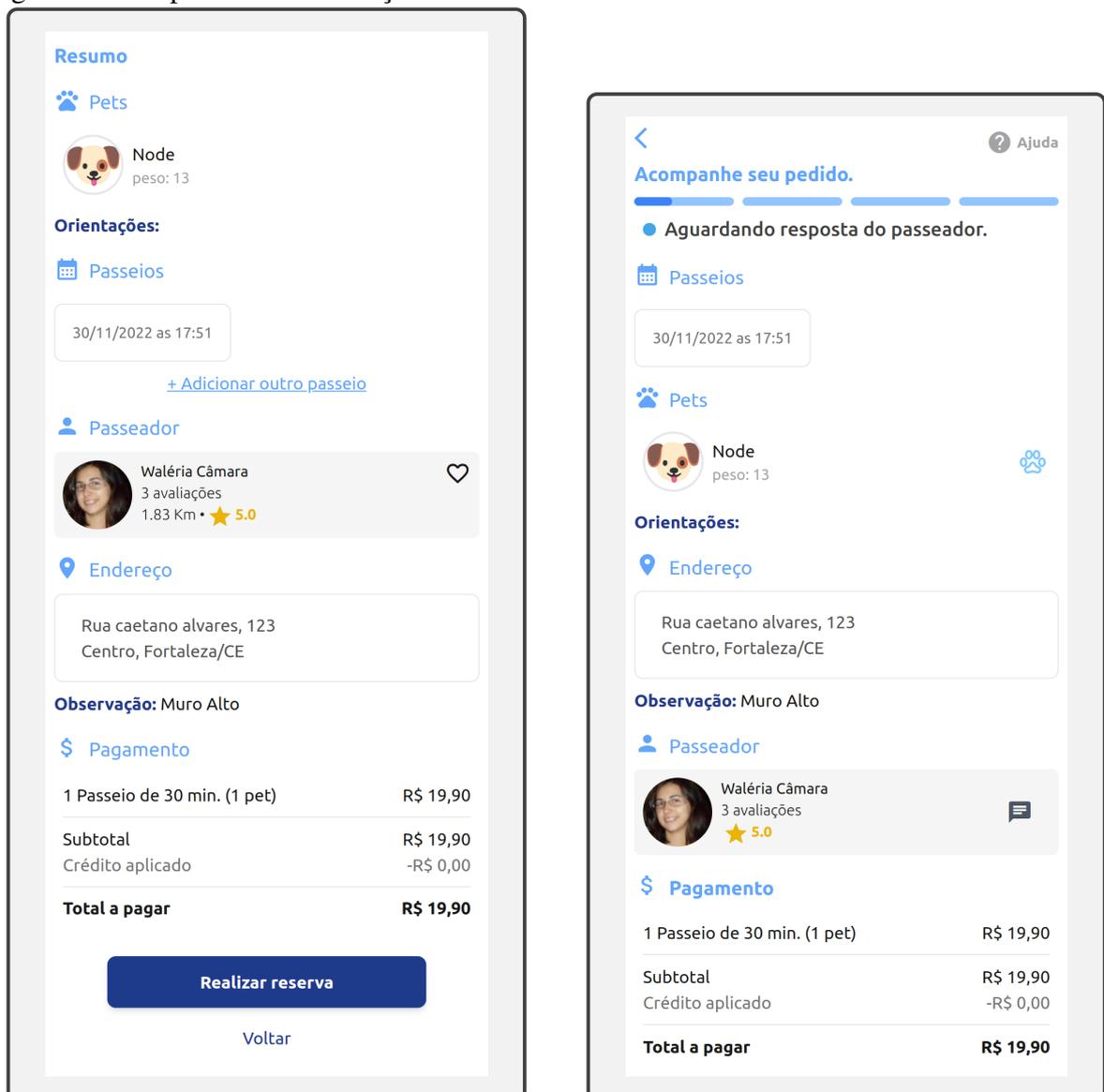
(a) Elaborado pelo autor.

(b) Elaborado pelo autor.

A Figura 23 *a* apresenta o quinto passo da reserva, nela é apresentado um resumo das informações coletadas durante o processo de reserva. Nela o tutor pode conferir os *pets* selecionados para o passeio, a data e hora do passeio, o passeador selecionado, o endereço de retirada do pet e um resumo do valor a ser pago pelo serviço. Por fim um botão realizar reserva. Esse botão encaminha a requisição do passeio ao passeador que ainda precisa confirmar disponibilidade.

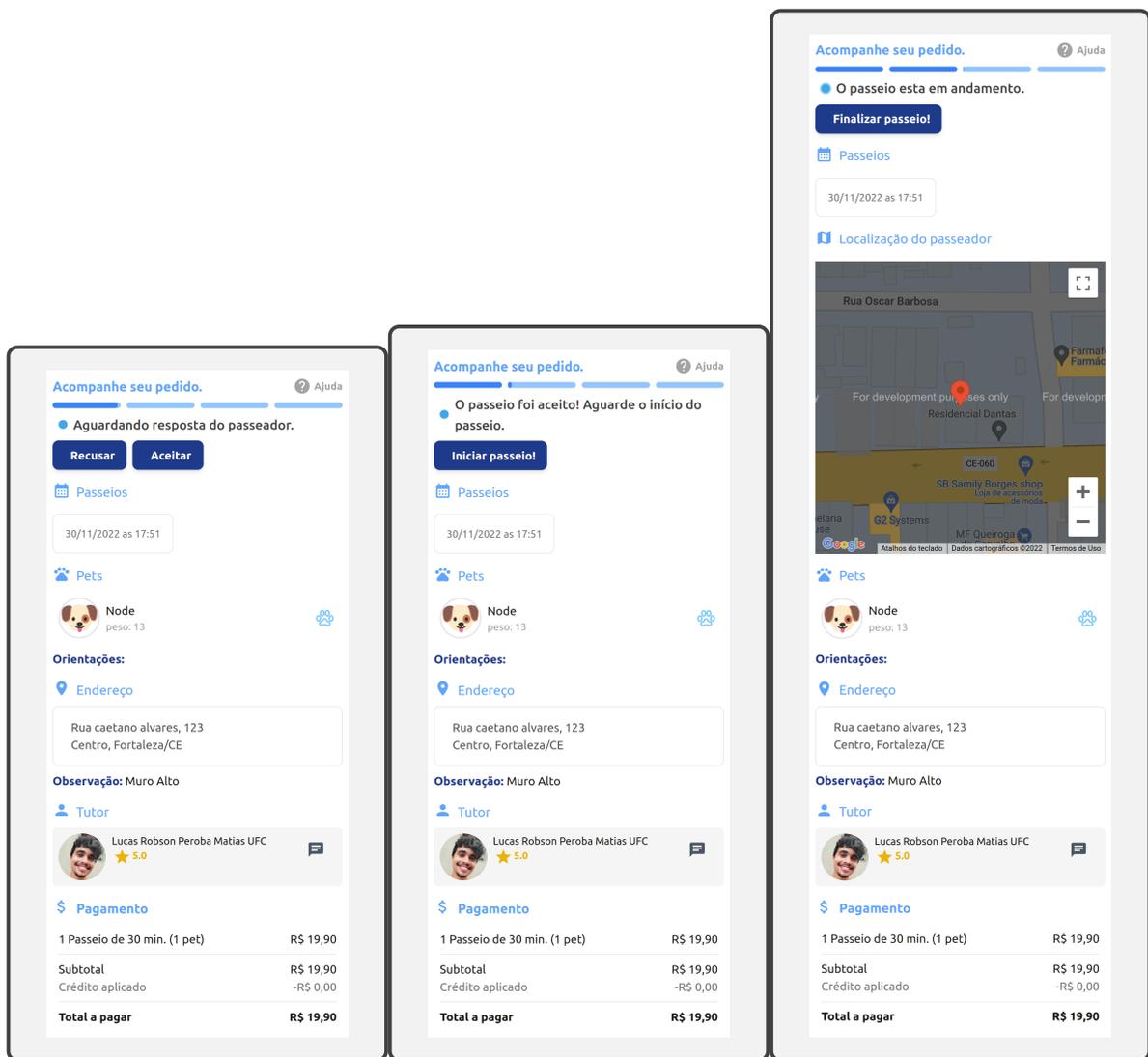
A Figura 23 *b* apresenta uma reserva com o estado "Aguardando a resposta do passeador". Nessa tela o tutor tem acesso as mesmas informações da tela anterior e está aguardando a resposta do passeador. Além disso, ele tem acesso a um chat que pode ser acessado através do ícone de mensagem ao lado do perfil do passeador.

Figura 23 – Esperando confirmação



Na Figura 24 *a* é apresentada a tela da requisição de uma reserva ao passeador. Nessa tela o passeador tem acesso as mesmas informações do passeio solicitado com a opção de aceitar ou negar o serviço. Se aceitar o serviço, ele terá à sua disposição a tela apresentada na Figura 24 *b*, que lhe dará opção para iniciar o passeio. Após iniciado o passeio, na tela da Figura 24 *c* é apresentada e nela fica disponível o botão de finalizar passeio. O elemento visual de quatro etapas ao topo da página informa a etapa do passeio em que se está. Em todas as telas, está disponível o ícone do *chat* ao lado da foto do tutor. Nas pesquisas realizadas durante a fase de requisitos, foi visto que em alguns lugares é comum o passeador pegar ao mesmo tempo, pets de diferentes tutores e levá-los juntos para passear. Atualmente, apenas é possível um vínculo contratual de passeio por vez por janela de tempo, não possuindo a possibilidade ainda de fechar passeio com vários tutores da mesma área para passear na mesma janela de tempo.

Figura 24 – Telas do passeador



(a) Elaborado pelo autor.

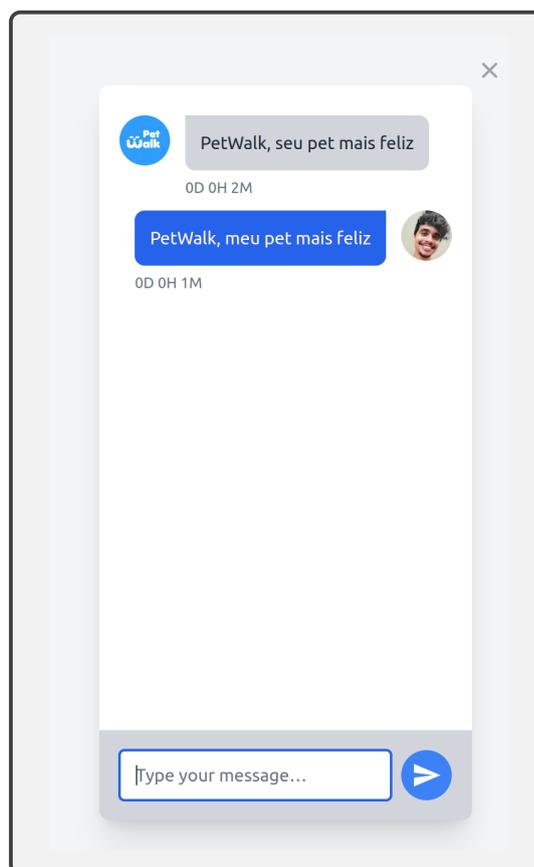
(b) Elaborado pelo autor.

(c) Elaborado pelo autor.

5.5.3 Chat

Na Figura 25 é apresentada a tela de *chat*. Nessa tela, o tutor e o passeador podem se comunicar a partir da solicitação inicial do serviço, mas também durante e depois do serviço. A funcionalidade de *chat*, foi implementado utilizando a tecnologia do *Firestore real time database*. Ao enviar uma mensagem, a mensagem é armazenada no banco de dados e carregada automaticamente na tela do destinatário, mas apenas se este estiver acessando a tela do *chat*. Não foi implementada a funcionalidade de notificações para que o passeador ou tutor saiba que existe uma nova mensagem. Também não foram inseridos elementos visuais na tela que avisem se a mensagem foi visualizada. Desde que ambos estejam acessando a página de *chat*, o *chat* em tempo real funciona normalmente como esperado em aplicativos de mensagens instantâneas.

Figura 25 – Chat



Fonte: Elaborado pelo autor.

5.5.4 Geolocalização

- Na Figura 24 c é apresentada a tela da reserva de passeio em andamento, nela é apresentado a geolocalização. Na geolocalização, é enviada uma requisição ao dispositivo do passeador que deve autorizar o envio da sua localização. Após a primeira autorização, a localização é atualizada a cada certo intervalo de tempo. A geolocalização só está disponível durante o momento do passeio. O passeador inicia o passeio após pegar o animal e clicar em "iniciar passeio!". Após iniciar, o sistema de geolocalização fica habilitado para consulta pelo tutor. Nesse momento, na tela de reserva do tutor também é exibido a geolocalização do passeador.

6 TESTES DE USABILIDADE

Neste tópico serão apresentados os resultados dos testes de usabilidade no sistema realizado no dia 19/10/2022. Os participantes: 20 tutores foram orientados a realizar uma tarefa pré-definida, sem determinação de tempo para finalização.

- Cadastrar um pet
- Cadastrar um endereço
- Escolher determinado passeador
- Realizar uma reserva

O teste foi realizado em uma sala apropriada no campo UFC Quixadá, esse ambiente permite a aplicação de testes com o mínimo de interferências externas. A sala é dividida em duas partes, em um lado o usuário do teste tem acesso ao sistema e um aparelho celular que está fazendo uma transmissão de toda interação para um computador que fica na sala do operador de testes. Além disso, o operador pode se comunicar com o usuário através de um microfone ligado a caixas de som na sala do usuário. Durante o teste, os usuários foram gravados e lhes foi pedido para narrar seus pensamentos durante todo o processo.

6.1 Resultados do teste de usabilidade

Neste tópico serão apresentados o perfil dos participantes e algumas observações ditas durante a interação. Fizemos um total de 20 testes, e após uma análise, foi selecionado as 7 mais relevantes, dessa forma eliminando testes com resultados repetidos. Por fim, conseguimos entender os pontos positivos e negativos no sistema.

1ª Interação

Numero de interações semelhantes: 2

Observações: O propósito do sistema ficou claro, gostou do design, durante a realização do pedido, é possível recarregar a página, causando a perda de algumas informações já fornecidas anteriormente por mim

2ª Interação

Numero de interações semelhantes: 5

Observações: Achou o tema do projeto ótimo, o objetivo claro, por ser tutora, atende suas necessidades. Analisou todas as telas do aplicativo, realizou todas as tarefas e não encontrou

dificuldades em utilizar a plataforma. Não deu nenhuma sugestão de melhoria.

3ª Interação

Numero de interações semelhantes: 1

Observações: Gostei em geral do design, e deixou uma ressalva: após selecionar um serviço avulso, não percebi qual seria o próximo passo. Sugeri separar a primeira etapa da reserva em duas partes.

4ª Interação

Numero de interações semelhantes: 8

Observações: Realizou todas as tarefas rapidamente sem dificuldades. Não fez nenhuma observação.

5ª Interação

Numero de interações semelhantes: 1

Observações: Realizou todas as tarefas rapidamente sem dificuldades. No entanto, pontuou que no fluxo de reserva, se passar pelo cadastro pet, o sistema esquece a escolha do tipo de passeio.

6ª Interação

Numero de interações semelhantes: 2

Observações: Realizou todas as tarefas rapidamente sem dificuldades, no entanto, apontou que os campos de data e peso poderia ser teclado numérico.

7ª Interação

Numero de interações semelhantes: 1

Observações: Gostei do design, ao finalizar o cadastro de um endereço. Não entendi que teria que selecionar o endereço cadastrado

6.2 Correções pós-testes

Neste tópico serão apresentadas as correções realizadas no aplicativo após os testes tendo como base os resultados obtidos na seção 6.1, com o intuito de melhorar a qualidade de

uso do sistema.

Na figura 26 *b* é apresentada a correção realizada na tela do "fluxo de reserva". Nela é apresentado a primeira tela do fluxo após criar um pet, a opção avulso foi persistida em memória. Essa alteração foi feita para prevenir a perda dos dados já preenchido pelo usuário. Dessa forma corrigindo 2 problemas, relatados nos testes. A volatilidade dos dados quando entra na tela de cadastro pet ou quando recarregamos a página no fluxo de reserva.

Na figura 26 *b* é apresentado a correção na tela do "Fluxo de reserva na etapa de seleção do endereço". Agora o sistema vai lembrar qual o último endereço selecionado e caso o usuário só tenha um endereço ou nunca tenha selecionado endereços antes, o sistema seleciona o primeiro endereço da lista.

Figura 26 – Correção Reserva e Endereço



(a) Autor, 2022

(b) Autor, 2022

Na figura 27 é apresentada a correção realizada na tela "cadastro pet". Foi feita alterações no campo peso e data de nascimento. Agora os campos são apenas para números e caso o usuário esteja acessando por um dispositivo móvel, é apresentado um teclado numérico.

Figura 27 – Correção 02

The image shows a mobile browser interface for a pet registration form. The browser's address bar displays "petwalks.com.br/pets". The form is titled "Pet" and contains the following sections:

- Pet:** Two radio button options: "Cachorro" (Dog) and "Gato" (Cat).
- Gênero:** Two radio button options: "Macho" (Male) and "Fêmea" (Female).
- Raça:** A text input field for the breed.
- Weight:** A text input field with a blue border and a blue slider handle, containing the text "_kg".
- Data de nascimento:** A text input field for the birth date.

A numeric keypad is visible at the bottom of the screen, with a teal arrow key on the right.

Fonte: Elaborado pelo autor.

7 CONCLUSÕES E TRABALHOS FUTUROS

Concluimos que, o objetivo proposto de desenvolver um aplicativo para que tutores de animais domésticos pudessem contratar serviços oferecidos por profissionais prestadores de serviço pet foi alcançado.

As seguintes funcionalidades foram implementadas e testadas: realizar login de tutor e passeador, cadastro de pets, visualizar um perfil, realizar a reserva de um serviço. Além disso, o sistema contempla outras funcionalidades não testadas com o chat, geolocalização em tempo real, sistema de avaliação de serviço.

Os testes de usabilidade realizados com 20 participantes sugerem que o aplicativo PetWalk teve boa avaliação. No total, 90% por cento dos participantes conseguiram encontrar o que queriam e que se sentiram no controle durante a navegação no aplicativo. Também foram recebidos comentários positivos, afirmado que as telas são atrativas e funcionais.

Sobre a perda de dados durante a reserva, foi feita uma correção assertiva de modo a prevenir esse problema. Além disso, foram resolvidos problemas de usabilidade como o campo de peso e data de nascimento, solucionando então o problema em questão.

Para futuras implementações pretendemos aumentar a área de alcance do aplicativo e adicionar novas funcionalidades tais como: videochamada durante o passeio, implementação de um sistema de notificações, adaptação da solução *web* para aplicativo *mobile*, melhorar o campo de pesquisa, cadastro do tutor e passeador, gerenciamento de agenda, filtro de passeadores por região, possibilitar a edição de perfil, possibilidade de atender a múltiplos tutores na mesma janela de tempo, gestão melhorada de preços.

REFERÊNCIAS

- ANDRADE, A. P. **O que é flutter?** 2020. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-flutter/>. Acesso em: 18 dez. 2022.
- ANDRADE, A. P. **O que é banco de dados?** 2021. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-banco-de-dados>. Acesso em: 18 dez. 2022.
- ARRUDA, G.; FACCHINI, E. Passeios disruptivos: um passo para o futuro. **South American Development Society Journal**, v. 7, n. 20, p. 126, 2021.
- BARBOSA, S.; SILVA, B. **Interação humano-computador**. [S.l.]: Elsevier Brasil, 2010.
- BATSCHINSKI. **GraphQL o que É? Visão Geral, prós e Contras**. 2021. Disponível em: <https://blog.back4app.com/pt/graphql-o-que-e/>. Acesso em: 18 dez. 2022.
- BORGES, A. C. Monografia, **Desenvolvendo um sistema Web com EXTJS**. [S.l.]: Faculdade de Tecnologia de Americana, 2013.
- CASAS, R. S. L. Monografia, **Tecnologia WEB aplicadas ao ensino de engenharia de estruturas**. [S.l.]: Universidade Federal de Minas Gerais, 2008.
- CASTILLO, K. R.; FERNANDEZ, L. A. M.; HUARACA, A. B. R.; UCULMANA, R. E. T.; PEÑA, C. B. V. Monografia, **Pet walk**. [S.l.]: Universidad Peruana de Ciencias Aplicadas (UPC), 2021.
- CUER, E. d. S. Monografia, **Comparação de desempenho de bancos de dados sql e nosql**. [S.l.]: Centro Universitário Eurípides de Marília (CUEM), 2015.
- FERNANDES, A. de A.; AMORIM, P. R. dos S.; BRITO, C. J.; MOURA, A. G. de; MOREIRA, D. G.; COSTA, C. M. A.; SILLERO-QUINTANA, M.; MARINS, J. C. B. Measuring skin temperature before, during and after exercise: a comparison of thermocouples and infrared thermography. **Physiological measurement**, IOP Publishing, v. 35, n. 2, p. 189, 2014.
- GEOLOCATION. **The Google Maps Geolocation API**. [S.l.]: Google, 2016. Disponível em: <https://developers.google.com/maps/documentation/geolocation/>. Acesso em: 18 dez. 2022.
- IBGE. **PNS 2019 V4 - IBGE | portal do ibge | IBGE**. 2019. Disponível em: <https://biblioteca.ibge.gov.br/visualizacao/livros/liv101764.pdf>. Acesso em: 18 dez. 2022.
- LECHETA, R. R. **Google Android: aprenda a criar aplicações para dispositivos móveis com o android sdk**. [S.l.]: Novatec, 2013.
- LOPES, L. **Passear com Cachorro: 9 motivos para não deixar o passeio de lado**. 2019. Disponível em: https://www.patasdacasa.com.br/noticia/passear-com-cachorro-9-motivos-para-nao-deixar-o-passeio-de-lado_a504/1. Acesso em: 18 dez. 2022.
- MAPS. **The Google Maps API**. [S.l.]: Google, 2016. Disponível em: <https://developers.google.com/maps>. Acesso em: 18 dez. 2022.
- MORAIS, B. S. d.; PINHEIRO, V. B. M. **Happy Pet: plataforma de contratação de pet sitters**. [S.l.]: 005, 2021.

MUNIZ, D. M. C. B. Monografia, **Visualização de dados no facebook com o uso do software GEPHI**: análise de rede sociais da biblioteconomia no maranhão. [S.l.]: Universidade Federal do Maranhão, 2018.

NOLETO, C. **Aplicações web**:: entenda o que são e como funcionam! 2020. Disponível em: <https://blog.betrybe.com/desenvolvimento-web/aplicacoes-web/>. Acesso em: 18 dez. 2022.

PONTES, T. B.; ARTHAUD, D. D. B. Metodologias ágeis para o desenvolvimento de softwares. **Ciência E Sustentabilidade**, v. 4, n. 2, p. 173–213, 2018.

PRESSMAN, R.; MAXIM, B. **Engenharia de Software**. [S.l.: s.n.], 2016. Disponível em: <https://books.google.com.br/books?id=wexzCwAAQBAJ>. Acesso em: 18 dez. 2022. ISBN 9788580555349.

SADALAGE, P. J.; FOWLER, M. **NoSQL essencial**: um guia conciso para o mundo emergente da persistência poliglota. [S.l.]: Novatec, 2019.

SOMMERVILLE, I. **Engenharia De Software**. [S.l.]: PEARSON BRASIL, 2019. Disponível em: <https://books.google.com.br/books?id=tfivwwEACAAJ>. Acesso em: 18 dez. 2022. ISBN 9788543024974.

TANENBAUM, A. S.; WETHERALL, D. J. **Redes de computadores**. [S.l.]: Pearson Prentice Hall, 2011.

VALE. **Alisson Vale - O Fundador do Software Zen**. 2022. Disponível em: <https://softwarezen.me/alisson-vale/>. Acesso em: 18 dez. 2022.

VAZQUEZ, C. E.; SIMÕES, G. S. **Engenharia de Requisitos**: software orientado ao negócio. [S.l.]: Brasport, 2016.

VIDOTTI, S. A. B. G.; CONEGLIAN, C. S.; ROA-MARTÍNEZ, S. M.; VECHIATO, F. L.; SEGUNDO, J. E. S. Web, web semântica e web pragmática:: um posicionamento da arquitetura da informação. **Informação & Sociedade**, Universidade Federal da Paraíba-Programa de Pós-Graduação em Ciência da . . . , v. 29, n. 1, 2019.