



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

DANIELLE DE LIMA VIEIRA

**DESENVOLVIMENTO DE UM WEBSITE PARA AUXILIAR NA
BUSCA DE LAR TEMPORÁRIO PARA ANIMAIS
ABANDONADOS**

CAMPINA GRANDE - PB

2023

DANIELLE DE LIMA VIEIRA

**DESENVOLVIMENTO DE UM WEBSITE PARA AUXILIAR NA
BUSCA DE LAR TEMPORÁRIO PARA ANIMAIS
ABANDONADOS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharela em
Ciência da Computação.**

Orientador: Professor Me. Roberto Medeiros de Faria

CAMPINA GRANDE - PB

2023

DANIELLE DE LIMA VIEIRA

**DESENVOLVIMENTO DE UM WEBSITE PARA AUXILIAR NA
BUSCA DE LAR TEMPORÁRIO PARA ANIMAIS
ABANDONADOS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharela em
Ciência da Computação.**

BANCA EXAMINADORA:

Professor Me. Roberto Medeiros de Faria

Orientador – UASC/CEEI/UFCG

Professor Dr. Herman Martins Gomes

Examinador – UASC/CEEI/UFCG

Professor Dr. Tiago Lima Massoni

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 14 de Fevereiro de 2023.

CAMPINA GRANDE - PB

ABSTRACT

The rescue and treatment of abandoned animals is a matter of great importance from a social, animal welfare and public health point of view. In many situations, the interval between the rescue of the animal and its adoption can be quite long, which makes rescue difficult due to overcrowding or lack of physical space to house the animals. An alternative in these situations is to look for a temporary foster care that can take care of the animal until a permanent and responsible home is found. To help with this task, a website was developed in order to facilitate the search for temporary homes, using the React framework for the frontend (the graphical user interface) and BaaS (Backend as a Service) Firebase. Users of this website are ordinary projects/people who support abandoned animals and volunteers who are willing to offer a home. At the end of the development, a satisfaction survey was carried out with potential users. Overall, the site was well received and the survey also pointed to areas where the site needs improvement.

Desenvolvimento de um Website para Auxiliar na Busca de Lar Temporário para Animais Abandonados

Danielle de Lima Vieira¹

Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil

danielle.vieira@ccc.ufcg.edu.br

Roberto Medeiros de Faria

Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil

rfaria@computacao.ufcg.edu.br

RESUMO

O resgate e tratamento de animais abandonados é uma questão de suma importância do ponto de vista social, do bem estar animal e da saúde pública. Em muitas situações o intervalo entre o resgate do animal e sua adoção pode ser bastante longo, o que dificulta o resgate devido a superlotação ou falta de espaço físico para abrigar os animais. Uma alternativa nessas situações é buscar um lar temporário que possa cuidar do animal até que um lar permanente e responsável seja encontrado. Para auxiliar nessa tarefa foi desenvolvido um website com o objetivo de facilitar a busca por lares temporários, utilizando o framework React para o *frontend* (a interface gráfica de usuário) e o BaaS (*Backend as a Service*) Firebase. Os usuários deste website são os projetos/pessoas comuns que prestam apoio a animais abandonados e voluntários que estão dispostos a oferecer um lar. Ao final do desenvolvimento foi feita uma pesquisa de satisfação com potenciais usuários. No geral, o site foi bem aceito e a pesquisa também apontou áreas em que o site deve melhorar.

Keywords

Website, resgate de animais, React, Firebase.

Site

<https://lartemporario.netlify.app/>

1. INTRODUÇÃO

A sociedade carece de informações sobre a forma correta de lidar com os animais, gerando casos frequentes de maus tratos e abandono de animais de companhia [1].

No Brasil, observa-se um grande número de animais em estado de abandono. Dados da Organização Mundial de Saúde (OMS) apontam que há cerca de 10 milhões de gatos e 20 milhões de cães [2].

O resgate e tratamento desses animais abandonados é uma questão de suma importância para o campo da saúde pública, o ambiente e

a sociedade, principalmente em centros urbanos, considerando-se o aumento crescente de abandono de animais, o não respeito ao seu bem-estar e as consequências que eles causam, como a disseminação de enfermidades [3].

Uma das possibilidades geradas pela internet é o trabalho virtual associado ao tema bem-estar animal (BEA) como, por exemplo, as campanhas de adoção de animais pela internet. Apesar dessa abordagem ser recente e, ainda não totalmente explorada, tem-se alguns bem-sucedidos planos de adoções on-line, como por exemplo a campanha “Adotar é Tudo de Bom”, 1ª desenvolvida pela empresa Pedigree® (2015), que promoveu a adoção de 45 mil animais no ano, utilizando os meios digitais [4].

Existem muitos sites e aplicativos voltados para a adoção de animais. Exemplos são os sites “amigo não se compra” [5], “Aliança com a Vida”, ONG (Organização não Governamental) de São Paulo [6], e o aplicativo “Appets” [7] para Android. Porém, o intervalo entre o resgate do animal e sua adoção pode ser bastante longo, principalmente para animais deficientes ou idosos. Esse intervalo dificulta o resgate devido a superlotação ou falta de espaço físico para abrigar os animais. Outras condições específicas como dificuldade de socialização do animal ou doença também são fatores que prejudicam a permanência deste em um abrigo.

Nessas situações é possível buscar um lar temporário. O lar temporário é quando uma pessoa, sozinha ou em parceria com alguma ONG, se compromete a acolher em casa, por um tempo determinado, um animal abandonado.

Em alguns sites de ONGs e outros projetos de resgate de animais é possível preencher um cadastro e se voluntariar como lar temporário, a exemplo do site Aliança com a Vida. No entanto, nem todos oferecem essa possibilidade e geralmente o projeto ou ONG atende uma região específica.

Essa solução não é muito abrangente e a falta de um abrigo pode dificultar ou até impedir o resgate de um animal.

Portanto, como forma de atender essa necessidade, propõe-se o desenvolvimento de um website com o objetivo de facilitar a busca por lares temporários para animais resgatados, aproximando a pessoa que resgata de um lar temporário cadastrado pela sua geolocalização.

Os usuários deste website são os projetos de apoio a animais abandonados e pessoas comuns, que resgataram animais mas não tem condições de cuidar dos mesmos e buscam um lar para recebê-los ou pessoas que estão dispostas a oferecer um lar temporário.

¹ “Os autores retêm os direitos, ao abrigo de uma licença Creative Commons Atribuição CC BY, sobre todo o conteúdo deste artigo (incluindo todos os elementos que possam conter, tais como figuras, desenhos, tabelas), bem como sobre todos os materiais produzidos pelos autores que estejam relacionados ao trabalho relatado e que estejam referenciados no artigo (tais como códigos fonte e bases de dados). Essa licença permite que outros distribuam, adaptem e evoluam seu trabalho, mesmo comercialmente, desde que os autores sejam creditados pela criação original.”

2. SOLUÇÃO

2.1 Soluções conhecidas

Foi feito um levantamento através de pesquisa online e conversa com pessoa responsável por um abrigo com o intuito de obter informações sobre a forma como comumente instituições/voluntários que aloca animais em lares temporários ou permanentes fazem buscas e divulgações de suas necessidades. Observou-se que algumas instituições oferecem formulário próprio para cadastro de voluntários (ONG Aliança com a Vida) ou divulgam a necessidade em suas redes sociais e esperam que alguém entre em contato (GAPACG [8]). Já pessoas que buscam/oferecem lares temporários para animais costumam entrar em contato com instituições da região, outras pessoas de grupos em redes sociais (Lar Temporário [9]), grupos em aplicativos de mensagens ou pessoas conhecidas por atuarem como voluntárias.

2.2 Solução proposta

A solução proposta é um website responsivo (o LarTemporário), desenvolvido seguindo uma abordagem *mobile first*, projetando-se primeiro a interface para dispositivos móveis e depois para outros dispositivos. O design responsivo se adapta ao tamanho da tela, permitindo o uso quer seja a partir de um smartphone ou desktop. Essa escolha foi feita devido ao alto uso de dispositivos móveis para acesso a internet e pela maior precisão da geolocalização do smartphone através do GPS (*Global Positioning System*) desses dispositivos. Além disso, uma aplicação web torna o serviço independente de lojas de aplicativos, podendo ser acessado a partir da maioria dos browsers.

O objetivo do website é permitir a criação de perfis de lares temporários (por pessoas cadastradas que tenham o interesse em oferecer um) e assim construir uma base de dados de lares disponíveis. A partir dessa base de dados, pessoas que buscam um lar podem visualizar uma lista dos lares que se encontram próximos (segmentação por geolocalização) e escolher aquele que melhor se adequa às necessidades do animal resgatado.

Outra possibilidade é filtrar os lares listados por meio de critérios como tipo de animal recebido, tempo para acolhimento, necessidade de auxílio com despesas e se o lar já possui outros animais. Por fim, ao selecionar um lar é exibido o perfil do mesmo com informações mais detalhadas, contatos, redes sociais e imagens. Apenas usuários cadastrados no sistema podem visualizar o conteúdo e criar um perfil de lar.

O website não se compromete a fazer qualquer tipo de verificação de dados ou acompanhamento dos lares, apenas contribui para a divulgação e aproximação das partes interessadas.

O público alvo são pessoas com o desejo de resgatar, acolher e cuidar de animais abandonados, possibilitando uma vida digna, com carinho e atenção, mas que não têm disponibilidade para cuidar dos animais de forma permanente. Instituições que resgatam animais que não possuem sede física ou não têm mais capacidade para novos animais. Essas instituições geralmente fazem a busca do lar permanente enquanto o lar temporário cuida do animal.

O código do website é aberto e possibilita que outros contribuam para aperfeiçoar e expandir as funcionalidades do website, bem como, permite a qualquer pessoa ou instituição copiar, modificar,

estender ou distribuir o software, podendo criar suas próprias versões.

2.3 Funcionalidades

Inicialmente a aplicação contém quatro telas. A primeira tela consiste em uma tela de boas vindas com a logo do website, informações e botões que permitem a funcionalidade de login.

Quando o usuário efetua o login no site pela primeira vez, o sistema de autenticação gera o registro desse novo usuário. É possível fazer login na página inicial do website com uma conta Google e futuramente com uma conta Facebook, conforme as Figuras 1 e 2. Após efetuar o login o usuário é redirecionado para a página principal da aplicação. A funcionalidade de login foi programada utilizando o SDK do Firebase Authentication com provedores do Google e Facebook, contudo a autenticação com o Facebook necessita de configurações extras na API do Facebook e envio de documentação para ser liberada.



Figura 1 - Tela inicial para login do usuário em dispositivos *mobile*



Figura 2 - Tela inicial para login do usuário em desktops

A próxima tela permite buscar, exibir a listagem e filtrar lares temporários ou cadastrar um novo lar.

Ao clicar no botão de busca, o site solicita ao usuário a permissão para acesso a sua geolocalização. Aceitar essa solicitação é obrigatório para seguir com o uso da funcionalidade. Após o usuário aceitar a solicitação, o sistema faz a busca por perfis de lares próximos de acordo com a geolocalização, em um raio fixo de 150 km, e exibe os resultados em uma lista. Cada item da lista possui um botão para mais informações que redireciona para a página de exibição do perfil do lar selecionado. O usuário pode também filtrar os resultados listados de acordo com o tipo de animal recebido, o período de tempo disponível para acolhimento, a necessidade de auxílio com despesas e se o lar já possui outros animais (Figuras 3 e 4).

Nessa tela também é possível fazer o cadastro de um lar temporário, caso o usuário não tenha cadastrado nenhum ainda, ou escolher visualizar o perfil cadastrado. Ao clicar no botão de cadastro, é aberto um modal com um formulário onde devem ser inseridos os dados do lar temporário. Obrigatoriamente o usuário

deve informar um nome para o lar, número para contato, uma breve descrição sobre o lar, tipos de animais aceitos, porte, tempo disponível para cuidar dos animais, quantidade de animais aceitos, necessidade de auxílio com despesas e se já possui outros animais. De forma opcional o usuário pode adicionar suas redes sociais (Figura 5).

A tela do perfil de um lar exibe as informações obrigatórias e opcionais cadastradas (Figuras 6 e 7). Também permite ao usuário responsável pelo perfil adicionar fotos. Para inserir imagens no álbum do perfil basta clicar no botão ao lado do título “Fotos”. Um modal é aberto onde é possível selecionar a imagem a ser adicionada dos arquivos do usuário. Esta tela também possui botões que permitem a edição e exclusão do perfil.

As imagens de animais que aparecem na Figura 7 foram obtidas de datasets disponíveis na internet [10, 11].

Por último, ao clicar no botão “Ver todas as fotos” no perfil de um lar, o usuário é direcionado para uma página onde são exibidas todas as imagens cadastradas naquele perfil.

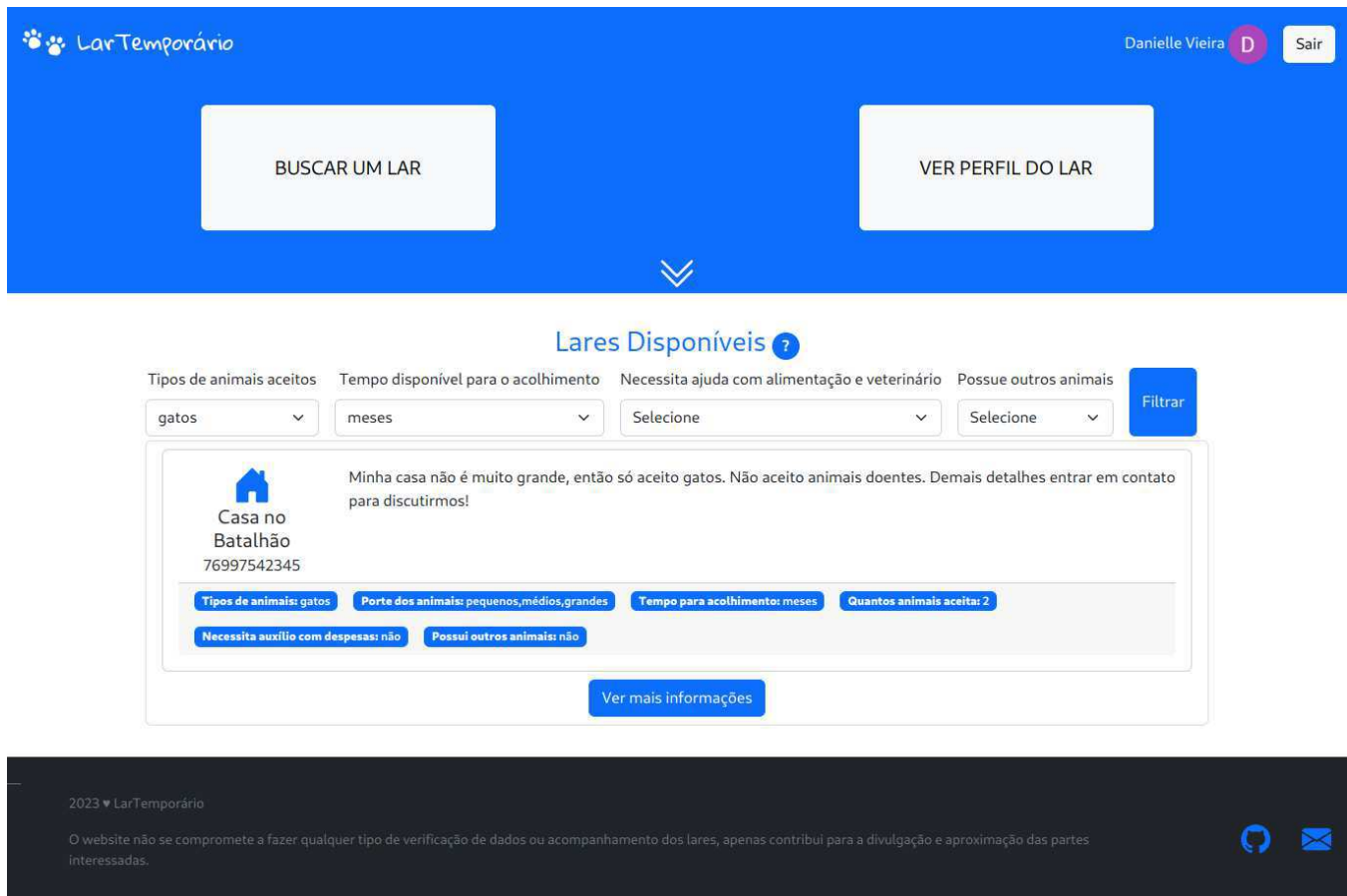


Figura 3 - Tela principal com listagem e filtros de lares disponíveis em *desktops*

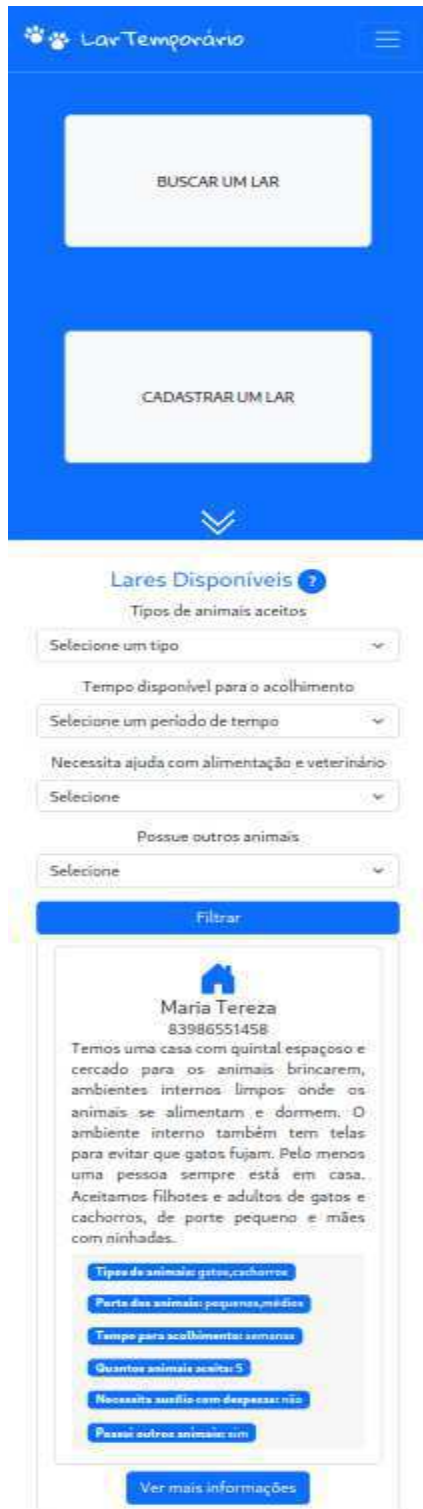


Figura 4 - Tela principal com listagem e filtros de lares disponíveis em dispositivos *mobile*

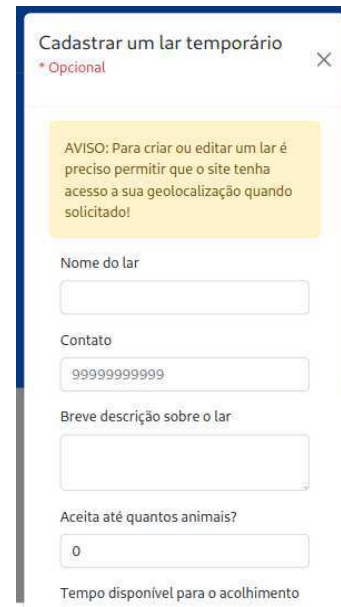


Figura 5 - Modal de cadastro de um lar temporário em dispositivos *mobile*



Figura 6 - Início da tela de perfil do lar em dispositivos *mobile*

2.4 Arquitetura

O LarTemporário foi desenvolvido seguindo a arquitetura cliente-servidor. Esta arquitetura consiste em um sistema de computação distribuída no qual existem dois módulos distintos. Um módulo, o servidor, é responsável por fornecer recursos e serviços que são consumidos pelo módulo cliente. O cliente obtém informações do usuário, envia requisições de recursos/serviços

para o servidor e aguarda o processamento. Esses módulos se comunicam, via de regra, através da rede de computadores (Figura 8).

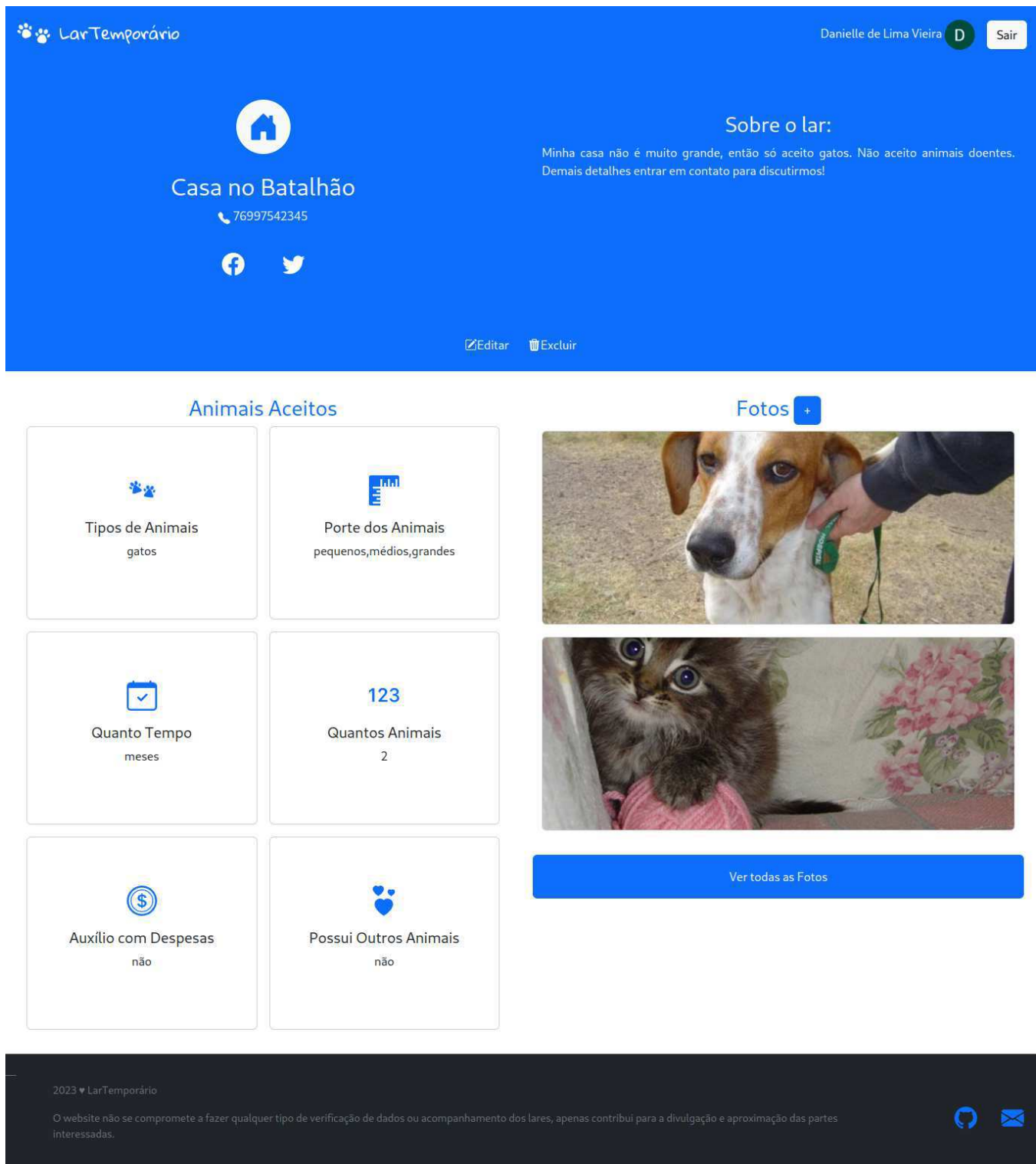


Figura 7 - Tela de perfil do lar no desktop

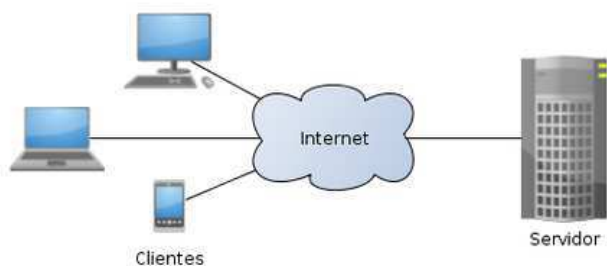


Figura 8 - Modelo cliente-servidor [12]

Em uma aplicação web, o cliente é o frontend no navegador do usuário (a interface gráfica de usuário de um website ou aplicação web, geralmente desenvolvida com HTML, CSS e Javascript). O servidor ou backend trabalha, geralmente, fazendo a ponte entre os dados que vêm do navegador rumo ao banco de dados e vice-versa, sempre aplicando as devidas regras de negócio, validações e garantias num ambiente restrito ao usuário final, ou seja, onde ele não consegue acessar ou manipular algo [13].

2.4.1 Tecnologias do frontend

O frontend do LarTemporário trata-se de uma *Single Page Application* (SPA). Uma aplicação web ou website que carrega apenas um único documento/página web, contendo as informações de HTML, JavaScript e CSS para responder a todas as possíveis interações do usuário, e depois atualizar o conteúdo desse documento de forma assíncrona via APIs (*Application Programming Interface*) Javascript, como XMLHttpRequest ou Fetch, reescrevendo dinamicamente a página com novos dados a partir do servidor. Como a estrutura da página permanece estática e não precisa carregar páginas completas a partir do servidor, há um ganho de performance e experiência mais dinâmica para os usuários pois a interface fica mais rápida, comparável àquela encontrada em apps nativos. Os *frameworks* JavaScript mais famosos para criar SPA são Angular, React e Vue.js. Para esse trabalho foi escolhido o *framework* React.

O React é um *framework* Javascript de código aberto que permite construir interfaces de usuários para web. Ele é baseado em componentes encapsulados que gerenciam seu próprio estado, melhorando o reaproveitamento de código e facilitando a manutenção, pois os componentes podem ser combinados para criar interfaces complexas.

Além do React utilizou-se outro *framework* de componentes *frontend* para desenvolvimento de websites responsivos, o Bootstrap. Este framework combina HTML, CSS e Javascript com Sass (uma linguagem de extensão do CSS), sistemas de *grid* responsivo, componentes pré-construídos e plugins Javascript. Pode ser adicionado ao projeto utilizando *npm* (*Node Package Manager*). O React-Bootstrap é uma adaptação do Bootstrap Javascript para trabalhar com componentes React. Permite um desenvolvimento *mobile first* com design responsivo suportado pelos navegadores mais populares e uma curva de aprendizagem baixa para quem tem algum conhecimento de CSS, HTML e Javascript.

Para lidar com os formulários da aplicação foi utilizada a biblioteca Formik.

2.4.1.1 Estrutura do frontend

O React não define um padrão de estruturação para os projetos, porém existem algumas abordagens comuns no mercado. A abordagem escolhida foi separar os componentes de acordo com o papel que desempenham na aplicação (Figura 9).

Na pasta *assets* são armazenados os arquivos estáticos (logos, imagens e ícones da aplicação). Em *pages* estão os componentes em nível de rota (as telas que são renderizadas no browser). *Components* contém os componentes globais, ou seja, reaproveitados em várias partes do código. Na pasta *routes* estão os arquivos que definem as rotas da aplicação. *Contexts* contém os *states* (variáveis que registram algum estado da aplicação) que precisam ser compartilhados entre componentes e a pasta *services* possui os serviços externos com os quais a aplicação interage.

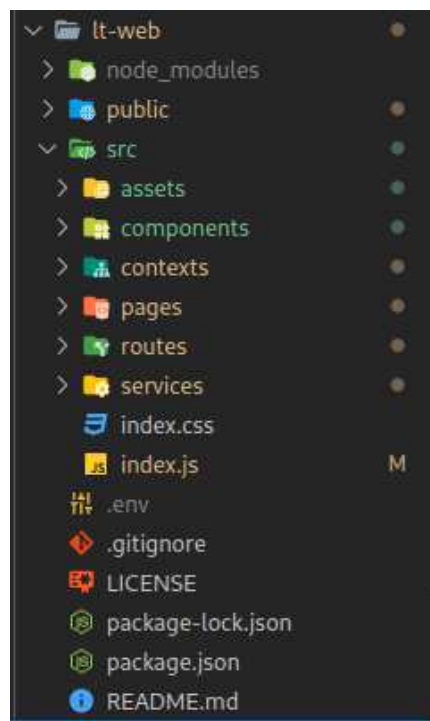


Figura 9 - Estruturação do projeto no frontend

2.4.2 Tecnologias do backend

Neste trabalho, decidiu-se pelo uso de um BaaS (*Backend as a Service*), o Firebase, fornecido pelo Google. Backend como serviço é um modelo de serviço em nuvem que oferece soluções modulares para funções comuns de uma infraestrutura. Autenticação de usuários, armazenamento, serviço de notificações (*push notifications*), integração com serviços externos, escalabilidade automática e *analytics* estão entre os básicos. Esses serviços são fornecidos por meio de SDKs (*software development kits*) e APIs (*application programming interfaces*).

O Firebase fornece uma plataforma básica e unificada para muitos aplicativos, juntamente com uma série de outros recursos do

Google incluídos no serviço. Ele lida com a maior parte do trabalho do lado do servidor quando se trata do desenvolvimento de aplicativos [14].

Todos os aplicativos do Firebase registrados no mesmo projeto compartilham os mesmos backends, como Firebase Hosting, Authentication, Realtime Database, Cloud Firestore, Cloud Storage e Cloud Functions [15] (Figura 10).

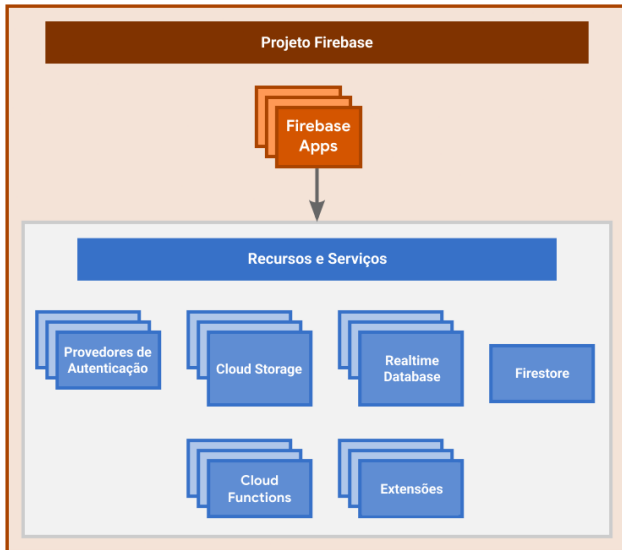


Figura 10 - Recursos e serviços do Firebase (adaptado) [15]

Apesar da necessidade de dedicar um tempo para o aprendizado dos backends do Firebase, as funcionalidades e facilidades, que estes disponibilizam, compensam o tempo gasto inicialmente no estudo. O LarTemporário faz uso dos seguintes serviços:

- Cloud Firestore - É um banco de dados em nuvem NoSQL, flexível e escalável, para armazenar e sincronizar dados. O Cloud Firestore armazena dados em "documentos", que estão contidos em "coleções". Por sua vez, os documentos também podem conter coleções [16]. Por exemplo, para o LarTemporário armazena-se uma lista de documentos de perfis dos lares em uma coleção chamada "shelters". Os dados são armazenados em JavaScript Object Notation (JSON).
- Cloud Storage - O Firebase Storage foi projetado para desenvolvedores de aplicações que precisam armazenar e servir conteúdo gerado pelo usuário, por exemplo, fotos ou qualquer outro arquivo [14]. Utilizou-se para permitir o armazenamento de fotos dos perfis de lares temporários. Para cada foto armazenada no Cloud Storage é gerado um link de download, o qual pode ser armazenado no documento do perfil de um lar. Quando o documento é recuperado pelo frontend, o link pode ser utilizado em uma tag HTML que faz o carregamento da imagem.
- Firebase Authentication - Oferece serviços de backend, SDKs fáceis de usar e bibliotecas de UI instantâneas para autenticar clientes em um aplicativo. Os usuários podem fazer *login* em um aplicativo Firebase, usando

FirebaseUI como uma solução de autenticação completa ou usando o SDK para integrar manualmente uma ou mais técnicas de login [14]. No presente trabalho, foi empregado o Firebase UI em conjunto com os provedores de autenticação do Google e do Facebook.

2.4.2.1 Rules do Firebase

As *rules* (regras de segurança) do Firebase permitem controlar o acesso a documentos e coleções no banco de dados. As regras básicas consistem em uma declaração de *match* especificando um caminho de documento e uma expressão de *allow*, detalhando quando a leitura dos dados especificados é permitida [17]. Também é possível definir regras para a validação dos dados antes que estes sejam salvos no banco de dados (Figura 11).

```
function verifyFields(fields) {
  return request.resource.data.keys().hasAll(fields) &&
    request.resource.data.keys().hasOnly(fields);
}

function validateFields() {
  return request.resource.data.name is string &&
    request.resource.data.contact is string &&
    request.resource.data.description is string &&
    request.resource.data.images is list &&

    request.resource.data.social is map &&
    request.resource.data.social.twitter is string &&
    request.resource.data.social.facebook is string &&
    request.resource.data.social.instagram is string &&

    request.resource.data.location is map &&
    request.resource.data.location.geohash is string &&
    request.resource.data.location.lat is number &&
    request.resource.data.location.lng is number &&

    request.resource.data.acceptConditions is map &&
    request.resource.data.acceptConditions.type is list &&
    request.resource.data.acceptConditions.size is list &&
    request.resource.data.acceptConditions.number is number &&
    request.resource.data.acceptConditions.time is string &&
    request.resource.data.acceptConditions.havePets is string &&
    request.resource.data.acceptConditions.payment is string;
}

match /shelters/{userID} {
  allow read: if isSignIn();
  allow create, update: if isSignIn() && isOwner(userID) &&
    verifyFields(
      ["name", "contact", "description", "acceptConditions", "social", "location"]
    ) && validateFields();
  allow delete: if isSignIn() && isOwner(userID);
}
```

Figura 11 - Regras de segurança para a coleção de lares temporários (*shelters*)

2.4.3 GeoFire e API Geolocation

Para o desenvolvimento da funcionalidade de busca por lares a partir da geolocalização do usuário, foram utilizadas a API Javascript Geolocation, nativa em muitos browsers, e a biblioteca GeoFire.

A API Geolocation permite a aplicação determinar a geolocalização (latitude, longitude) do dispositivo do usuário, usando GPS, WIFI e geolocalização por IP, entre outros. Por motivos de privacidade, é necessário que o usuário conceda a permissão para a localização do dispositivo.

A biblioteca auxiliar do Firebase GeoFire permite utilizar Geohashes para armazenar e consultar documentos por posição no Cloud Firestore, tornando possível exibir apenas lares em uma

distância fixa, próxima do usuário que está fazendo uma busca, e portanto acessível para a entrega do animal.

O Geohash é um sistema para codificar uma localização (latitude, longitude) em uma única string no padrão Base32. No sistema Geohash, o mundo é dividido em uma grade retangular. Cada caractere de uma string Geohash especifica uma das 32 subdivisões do prefixo hash. Quanto maior o prefixo compartilhado entre dois hashes, mais próximos eles estão um do outro [18].

3. SISTEMA EM USO

Com o intuito de avaliar o site LarTemporário, foi feita uma pesquisa com 4 potenciais usuários da aplicação, utilizando um formulário Google enviado por WhatsApp [19]. O formulário consiste na proposição de 2 cenários de uso do site. O primeiro envolve a busca de um lar temporário e o segundo envolve a criação de um perfil de lar temporário, para que os participantes pudessem conhecer o website e suas funcionalidades. Em seguida os participantes deveriam responder a um questionário no mesmo formulário.

O questionário tem como objetivo avaliar a satisfação e percepções dos usuários sobre o site. Para tanto foi feita uma adaptação do CSUQ (*The Computer System Usability Questionnaire*) [20] desenvolvido pela IBM para avaliação de métricas de usabilidade. O CSUQ consiste em questões que avaliam a satisfação do usuário com a usabilidade do sistema, mensagens de erro, documentação, funcionalidades, interface e satisfação no geral.

No questionário adaptado algumas afirmações são feitas sobre a satisfação do usuário com respeito aos tópicos citados acima. Estas deveriam ser respondidas em uma escala de 1 a 7, com 1 indicando discordância total da afirmação e 7 concordância total. Em seguida, o usuário deveria marcar as alternativas que indicavam os motivos de sua satisfação ou descrever os motivos que resultaram na insatisfação.

Por último, é solicitado aos participantes que listem pontos negativos e positivos da aplicação.

A afirmação sobre a satisfação dos usuários com a usabilidade do sistema recebeu uma média de 6,5 pontos. Dos motivos de satisfação com essa afirmação, os mais votados foram a facilidade de usar o sistema e a facilidade de aprender a usar o sistema. Um motivo de insatisfação apontado foi que quando se fecha a aba aberta com o site, ou redireciona para outra página, ao voltar para o site é necessário fazer login novamente.

Quanto a afirmação sobre a satisfação com as mensagens de erro apresentadas pelo sistema, a média obtida foi de 6,5 pontos. O motivo de satisfação mais votado para essa questão foi “sempre que cometo um erro ao usar o sistema, me recupero com facilidade e rapidez”. Não foram descritos motivos de insatisfação.

A afirmação da satisfação do usuário com as informações (como ajuda on-line, mensagens na tela e outras documentações) fornecidas com este sistema, recebeu uma média de 6,5. Os motivos “as informações fornecidas pelo sistema são de fácil compreensão” e “as informações são eficazes para me ajudar a concluir as tarefas e cenários” foram os mais votados. Nos motivos de insatisfação com esse tópico foi sugerido que o site deveria ter uma página FAQ (*Frequently Asked Questions*).

Sobre a satisfação dos usuários com a interface do site, obteve-se uma média de 6,5. O motivo mais votado foi “a interface deste sistema é agradável”. Foi apontado como motivo de insatisfação o tamanho dos botões da tela principal nos dispositivos móveis.

As funcionalidades do sistema e a satisfação geral receberam uma média de 6,5 e 6,7 respectivamente.

No final é solicitado aos participantes elencar pontos negativos e positivos do sistema. Alguns pontos negativos apontados foram o tamanho dos botões de busca e cadastro nas telas dos dispositivos móveis e a falta de persistência na seção do site. O principal ponto positivo mencionado foi a simplicidade de uso do site.

As mensagens de erro do sistema, a documentação e a interface foram os únicos tópicos que receberam alguma nota 5 (nota mais baixa), apontando que estes devem receber especial atenção e serem reavaliados e melhorados em atualizações futuras.

Embora a amostra dos participantes na avaliação seja pequena, a predominância de notas 5 ou superior fornece um indicativo da satisfação dos usuários com a aplicação.

4. EXPERIÊNCIAS E LIÇÕES APRENDIDAS

4.1 Processo de Desenvolvimento

Para o levantamento dos requisitos do website foi feita uma pesquisa online sobre o tema para identificar como pessoas e instituições fazem uso da internet nos trabalhos de resgate de animais abandonados. Também foi distribuído um formulário [21] para pessoas que fazem ou já fizeram busca/oferecimento de lar temporário, além de uma conversa com uma pessoa responsável por um abrigo de animais. Um ponto que ficou bastante claro desses levantamentos é que a disponibilidade de imagens dos animais e dos lares é algo de grande importância para esses clientes, por isso decidiu-se permitir que os usuários adicionem imagens ao perfil do lar temporário, embora esta funcionalidade não tenha sido inicialmente considerada. Os clientes também apontaram informações importantes para a escolha de um lar temporário. As informações apontadas mais frequentes foram selecionadas para serem filtros da listagem de lares e são exibidas com destaque nas páginas de perfil, sendo elas: tipos de animais, porte, quantidade de animais aceitos, necessidade de auxílio financeiro, tempo disponível para o acolhimento e se o lar já possui outros animais.

Com um melhor conhecimento do domínio do problema e dos requisitos dos clientes delimitou-se as principais funcionalidades para uma primeira versão do website, seguido de ordenação dessas funcionalidades de acordo com o seu grau de importância. A criação e exibição do perfil de um lar temporário, a busca e listagem de lares, o login de usuário, a edição do perfil de um lar e a exclusão de um perfil são as funcionalidades mais importantes para a obtenção de um mínimo produto viável que possa ser submetido ao teste de usuários. O desenvolvimento seguiu *sprints* (período de tempo no qual o trabalho deve ser concluído) planejadas de acordo com essa ordenação e a prototipação das telas e criação da logo do site foram feitas na ferramenta Figma.

As tecnologias foram escolhidas com base na familiaridade e uso prévio, nas funcionalidades oferecidas para atender as necessidades do projeto e facilitar o processo de desenvolvimento e no reconhecimento pelo mercado.

Devido a sua maior facilidade de configuração para projetos React o *deploy* do frontend encontra-se no Netlify. O Netlify permite a integração com o Github e escolha de uma *branch* que é monitorada. Assim, sempre que uma atualização é feita nessa *branch*, o Netlify automaticamente faz um novo *deploy* do site com as alterações recentes.

4.2 Desafios

Dada a pouca experiência com o design de interfaces, principalmente aquelas destinadas a dispositivos móveis, foi um grande desafio criar uma interface agradável e responsiva, sendo necessária uma ampla pesquisa sobre modelos e layouts recomendados.

Aprender novas ferramentas (Bootstrap, Firebase, Formik), levou mais tempo do que o previsto, porém as facilidades de desenvolvimento que estas ferramentas entregam fez delas uma escolha adequada. Principalmente o Firebase Authentication, pois reduz bastante a complexidade de programação da autenticação de usuários e o Bootstrap que oferece uma boa variedade de componentes responsivos.

Manter-se no tempo planejado para o desenvolvimento de cada *sprint* também foi um desafio constante. Algumas funcionalidades precisaram de mais tempo para o desenvolvimento do que o previsto inicialmente, como a funcionalidade de geolocalização. Pretendia-se desenvolver a geolocalização utilizando apenas as informações de latitude e longitude fornecidas pela *API Geolocation*, que seriam armazenadas no documento do perfil de um lar. Uma equação seria utilizada para o cálculo da distância entre dois pares de latitude e longitude, um do usuário que está buscando e outro do perfil do lar. Essa distância calculada seria o filtro para a busca de lares. No entanto, não é possível implementar essa consulta diretamente no Firestore. As alternativas encontradas foram utilizar o serviço Cloud Functions do Firebase como intermediário para o cálculo das distâncias e filtragem, utilizar uma API de geolocalização de terceiros ou utilizar uma biblioteca que trabalha com Geohash. Esta última foi a alternativa escolhida pois era uma solução sugerida pelo Firestore e próxima ao inicialmente pretendido. A biblioteca GeoFire gera o geohash que é então armazenado no documento do lar juntamente com as informações de latitude e longitude. A filtragem dos lares pode então ser feita diretamente no Firestore utilizando o campo de geohash.

Um dos requisitos indicado pela pessoa responsável por um abrigo que foi entrevistada no início do projeto não teve tempo de ser desenvolvido. Esse requisito consistia na possibilidade de cadastrar campanhas de doação no perfil de um lar.

Outra dificuldade foi lidar com APIs de terceiros. Por exemplo, ainda que a programação da funcionalidade de login com o provedor do Facebook tenha ocorrido sem problemas, funcionando normalmente com uma conta de desenvolvedor, o Facebook exige algumas configurações e documentos extras para liberar o login de outras contas que não sejam a de desenvolvedor, não tendo sido possível completar o envio desses documentos até a data de entrega do trabalho.

5. TRABALHOS FUTUROS

Futuramente espera-se evoluir as funcionalidades do sistema e corrigir os problemas apontados pela avaliação dos usuários.

Inicialmente é importante a complementação das funcionalidades já existentes. Por exemplo, para permitir que os usuários possam

não só adicionar imagens ao perfil como também excluí-las, que os usuários escolham o raio de busca de um lar, que seja possível a exclusão e edição da conta de usuário completa e liberar o login usando o Facebook.

Uma funcionalidade apontada no momento do levantamento dos requisitos do website foi a possibilidade de adicionar campanhas de doação aos perfis. Porém, devido a limitação de tempo esse requisito não foi implementado. Sendo esta uma questão de contribuição importante no auxílio aos animais abandonados, é importante desenvolver esta funcionalidade em trabalhos futuros.

Além disso, os pontos que apresentaram resultados mais baixos no questionário de avaliação do site devem ser revistos e aprimorados, resultando em melhorias no tratamento de erros da aplicação e na documentação fornecida aos usuários.

Outra providência importante é aumentar o público alvo do formulário de avaliação do site, para obter resultados estatísticos mais consistentes.

Algumas outras ideias propostas para o futuro da aplicação são a adição de perfis para os usuários que buscam um lar e um sistema de *match* ou sugestão de perfis baseado nas necessidades dos usuários que procuram um lar e nas condições de aceitação dos lares temporários. Também há a possibilidade de permitir aos usuários trocarem mensagens dentro do próprio site.

6. REFERÊNCIAS

- [1] Oliveira, A.B., Lourenção, C. e Belizario, G.D. Índice Estatístico de Animais Domésticos Resgatados da Rua vs Adoção. Revista Dimensão Acadêmica 1, 2 (Jul-Dez 2016).
- [2] Silva, D.O., *et al.* Solução de design digital para o problema dos animais abandonados. Revista Sistemas e Mídias Digitais (RSMD) - Experimentos em Design Digital 4, 2 (Out. 2019). Edição Especial.
- [3] Rodrigues, J. H. G. S. Reflexão Bioética sobre o Resgate e Tratamento de Animais Abandonados. Dissertação de Mestrado, Universidade de Brasília, Brasília, DF, 2022.
- [4] Evangelista, A.G., *et al.* As Mídias Sociais como promotoras de adoção de cães e gatos abandonados. In Proc. 7º Congresso Brasileiro de Extensão Universitária, 07-09 Setembro, 2016, Ouro Preto, Minas Gerais.
- [5] Amigo não se compra. 2023. Disponível em: <https://www.amigonaosecompra.com.br/>.
- [6] Aliança com a Vida. 2023. Disponível em: <https://www.aliancacomavida.com.br/>.
- [7] Appets. 2023. Appets, adote um animal de est. Disponível em: https://play.google.com/store/apps/details?id=com.myappets.apps&hl=pt_BR&gl=US&pli=1.
- [8] gapacg. 2023. Disponível em: <https://www.instagram.com/gapacg/>.
- [9] Lar temporário. 2023. Disponível em: https://www.facebook.com/groups/5439988492708718/?mib_extid=6NoCDW.
- [10] Zhang, W., Sun, J., and Tang X. Cat Head Detection - How to Effectively Exploit Shape and Texture Features. In *Proc.*

- of *European Conf. Computer Vision*, 2008, vol. 4, pp.802-816.
- [11] Khosla, A., *et al.* Novel dataset for Fine-Grained Image Categorization. In *Proceeding of First Workshop on Fine-Grained Visual Categorization (FGVC)*, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.
- [12] O Modelo Cliente Servidor. Disponível em: https://www.gta.ufrj.br/ensino/eel878/redes1-2016-1/16_1/p2p/modelo.html.
- [13] Souto, M. 2022. Front-end, Back-end e Full Stack. Disponível em: <https://www.alura.com.br/artigos/o-que-e-front-end-e-back-end>.
- [14] Chatterjee, N., *et al.* Real-time Communication Application Based on Android Using Google Firebase. *IJARCSMS* 6, 4 (April 2018) , 74-79.
- [15] Firebase. 2022. Entenda os projetos do Firebase. Disponível em: <https://firebase.google.com/docs/projects/learn-more?authuser=0&hl=pt>.
- [16] React Native Firebase. 2023. Cloud Firestore. Disponível em: <https://rnfirebase.io/firestore/usage>.
- [17] Firebase. 2022. Condições de gravação para regras de segurança do Cloud Firestore. Disponível em: <https://firebase.google.com/docs/firestore/security/rules-conditions?hl=pt&authuser=0>.
- [18] Firebase. 2022.Consultas geográficas. Disponível em: <https://firebase.google.com/docs/firestore/solutions/geoqueries?hl=pt-br>.
- [19] Vieira, D.L. 2023. Pesquisa de Avaliação do Website LarTemporário. Disponível em: <https://forms.gle/RDTNfw7bLjPwEY198>.
- [20] James R. Lewis. IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use, *International Journal of Human-Computer Interaction* 7, 1 (1995), 57-78.
- [21] Vieira, D.L. 2022. Projeto de desenvolvimento de um site para auxiliar na busca de lar temporário para animais abandonados. Disponível em: <https://forms.gle/RUaGDHwo5M1BTMp49>.