Um aplicativo para estimular a doação de alimentos.

Pedro Arthur Santana Espírito Santo
IFBA – Instituto Federal da Bahia
Salvador, Brasil
pedro.a7102@gmail.com

Antonio Carlos Santos Souza
IFBA – Instituto Federal da Bahia
Salvador, Brasil
acsantossouza@gmail.com

Abstract—This article addresses food waste in Brazil, the increase in hunger and its evolution over time. In addition, it features a system developed in Flutter and Python that aims to assist in the act of donating surpluses in an accessible way through mobile devices, which after tests with pré-selected users, proved to be stable and with good usability.

Keywords: hunger; donation; food; mobile app.

Resumo— Este artigo aborda o desperdício de alimentos no Brasil, o aumento da fome e a sua evolução ao longo do tempo. Além disso, apresenta um sistema desenvolvido em Flutter e Python que tem como objetivo, o auxílio no ato de doar excedentes de forma acessível através de dispositivos móveis, que após testes com usuários pré-selecionados, mostrou-se estável e com boa usabilidade.

Palavras-chave: fome; doação; alimento; aplicativo móvel

I Introdução

A fome no Brasil tem aumentado significativamente ao longo do tempo, sobretudo, após a pandemia do Covid-19. Apesar disso, todos os anos, toneladas de alimentos são jogados fora, seja por terem sobrado, ou por terem vencido.

Para amenizar esse problema, o governo brasileiro implantou uma política de incentivos fiscais para estimular a doação de alimentos por parte de redes de distribuição e restaurantes. Entretanto, carece de comunicação efetiva entre as instituições doadoras e beneficiários, visto que não há formas eficientes de saber quando há doações disponíveis, ou donatários interessados.

O aplicativo apresentado nesse artigo sugere uma solução para esse problema através da criação de uma tecnologia social que se baseie nas dores identificadas através da revisão bibliográfica.

O sistema foi desenvolvido com o *framework* Flutter, com API construída em FastAPI e utiliza o PostgreSQL como banco de dados. O seu principal objetivo é estabelecer uma rede de comunicação que possibilite o contato entre doadores de alimentos e donatários e vice-versa.

Alguns dos recursos que possibilitam a concretização desse objetivo são a exibição de alimentos em doação e de perfis de instituições de caridade inscritas na região, bem como formas de se comunicarem e realizarem reservas e retiradas.

O artigo realiza um levantamento do estado da arte e traz a contextualização bibliográfica do tema abordado. Além disso, aborda os requisitos funcionais e não funcionais, arquitetura, casos de uso e apresenta diagramas da solução implementada.

Levando em consideração que o uso de dispositivos móveis está em constante crescimento e a demanda de aplicativos cresce de maneira exponencial, pressupõe-se possível o desenvolvimento de um aplicativo *cross-platform* que tenha os recursos necessários para contribuir no aumento de doações e diminuição nos níveis de fome no Brasil.

A seção II realiza o levantamento do estado da arte, apresentando dados relevantes sobre fome crônica, desperdício e mercado de aplicativos. A seção III apresenta trabalhos correlatos que se enquadram na categoria de doação. A seção IV introduz a solução proposta, seus requisitos, casos de uso, diagramas arquiteturais, diagrama relacional e seu funcionamento. A seção V revela as avaliações e os resultados obtidos através de testes com usuários pré-selecionados. A seção VI conclui o artigo apresentando os resultados obtidos. A seção VII contém

II Fundamentação Teórica

Esta seção tem como objetivo contextualizar o trabalho e apresentar pontos relevantes sobre o tema levantado. A subseção II-A aborda a da fome no Brasil e a sua evolução histórica. A subseção II-B apresenta os números do desperdício no país e as suas consequências sociais e ambientais. A subseção II-C explica como as doações podem ser uma alternativa benéfica e sustentável para todos, trazendo alguns *cases* de sucesso. A subseção II-D caracteriza a tecnologia social e a diferencia da tecnologia convencional. A subseção II-E revela que a demanda de dispositivos móveis cresce constantemente, com projeções de crescimento ainda maiores para o futuro. A subseção II-F enfatiza a importância de aplicativos acessíveis e discorre sobre as diretrizes de acessibilidade e experiência de usuários divulgadas pelos maiores *players* do mercado.

A. *Fome no Brasil*

Diferentemente da fome aguda, que está relacionada com a apetite, a fome crônica ocorre quando a alimentação do indivíduo não é suficiente para fazer manutenção do seu organismo e para sustentar o desempenho das suas atividades rotineiras. Esse é o tipo de fome focalizado neste contexto [1].

Uma das necessidades mais básicas e importantes do ser humano é a alimentação. Ela é a principal responsável pela vida e envolve não apenas razões biológicas, mas também aspectos econômicos, sociais, científicos, políticos, psicológicos e culturais [2]. Sua escassez é a causadora de grande número de doenças.

A fome no Brasil tem suas raízes no período colonial. Naquela época, as suas maiores vítimas eram os membros da classe trabalhadora, que precisariam gastar mais de 70% das suas economias para ter uma alimentação que suprisse parcialmente suas necessidades diárias [3] [4].

Apenas a partir de 1940 surgiu a preocupação de mensurar os níveis de fome da população de forma quantitativa, sendo Josué de Castro, pioneiro na criação do mapa da fome [3].

Em 2006 foi criada o SISAN (Sistema Nacional de Segurança Alimentar e Nutricional), tornando a alimentação um direito social [5].

Hoje, mais de 42 milhões de brasileiros estão em situação de subnutrição. Dentre esses, 19.3 milhões vivem em pobreza extrema e 7.5 milhões sofrem de altíssima insegurança alimentar [6].

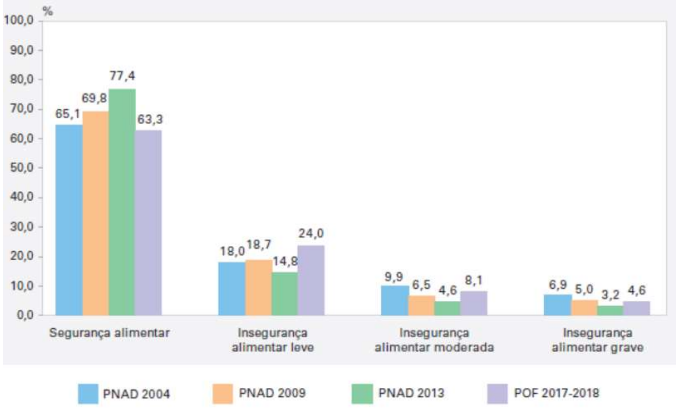


Figura 1 – Evolução da prevalência de segurança alimentar e insegurança alimentar leve, moderada e grave nos domicílios particulares – Brasil – 2004/2018 [7].

Os dados exibidos da Figura 1, mostram a queda nos níveis de segurança alimentar a partir de 2017.

A pandemia do Covid-19 aumentou significativamente os níveis de fome no país devido ao seu impacto econômico, onde milhares de pessoas perderam seus empregos e fontes de renda [8]. O surto do coronavírus, evidencia ainda mais a desigualdade social e tem um impacto significativamente maior em países da América Latina e África [9].

B. *Números do Desperdício*

Apesar de tanta fome, os números do desperdício são alarmantes. A Organização das Nações Unidas para a Alimentação e a Agricultura (FAO) estima que mais de 40% das culturas de raízes, frutas e vegetais são desperdiçados, assim como 35% dos peixes, 30% dos cereais e 20% das sementes oleaginosas, carne e laticínios, acumulando gastos maiores que 1 trilhão de dólares ao ano [10].

Mesmo chegando nas frentes de distribuição em desfalques, os alimentos ainda são desperdiçados. As sobras de alimentos próprios para consumo em estabelecimentos de alimentação coletiva, variam entre 7% e 35%. Grande parte dessas sobras é descartada [11]. Além disso, os alimentos com pequenos defeitos visuais, não são comprados pelos clientes dos supermercados. Quanto maior for o fornecimento, mais exigentes se tornam os consumidores, que consideram a aparência dos alimentos como prioridade acima do sabor [12]. Os alimentos considerados disformes, são descartados, mesmo estando em plenas condições de consumo. Frutas e legumes são desconsideradas pelos consumidores justamente no momento em que estão maduras [13].

Apesar de escolherem os alimentos com melhor aparência, as famílias brasileiras desperdiçam, em média, 353 gramas de comida por dia. A soma é de quase 130 kg ao ano, o que é equivalente a um desperdício anual de 41 kg por pessoa [14].

Estudos realizados em 2007, revelam que nas perdas totais da produção brasileira, 10% ocorrem nas propriedades rurais; 50% no manuseio e no transporte; 30% nas Centrais de Abastecimento (Ceasas) e 10% diluídos entre supermercados e consumidores [15]. Neste mesmo ano, a FAO também relata o desperdício anual de 1.6 gigatoneladas de alimentos no mundo, sendo mais de 80% próprios para consumo [16].

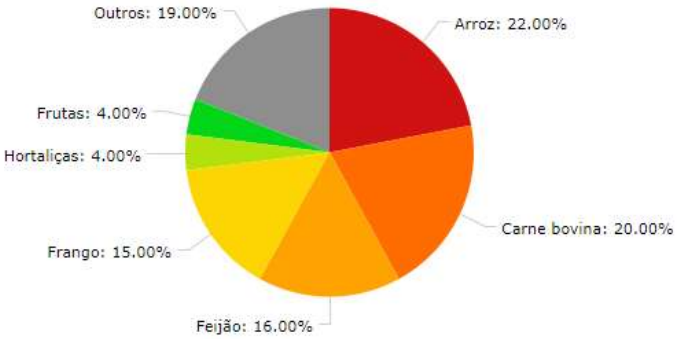


Figura 2 – Alimentos mais desperdiçados pelas famílias brasileiras.

O desperdício de alimentos na América Latina e Caribe, poderiam alimentar cerca de 300 milhões de

pessoas, quantidade equivalente a 10 vezes o número de pessoas subnutridas na região [17].

Em uma pesquisa realizada pela Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA), 43% dos entrevistados afirmaram conhecer pelo menos uma pessoa que desperdiça alimentos com regularidade. O estudo ainda revela que a maioria das pessoas vincula sentimentos negativos com o ato de jogar alimentos fora [14].

O desperdício também traz consequências ambientais. O alimento em decomposição é um dos principais emissores de metano do mundo e é responsável por despejar 3.3 gigatoneladas de dióxido de carbono ao redor do globo. Se esses números pertencessem a um país, este seria o terceiro maior emissor do mundo. A agropecuária também é a responsável pelo consumo de 69% de água doce no Brasil. Quando se fala de desperdício de alimentos, também se fala de desperdício de água [13].

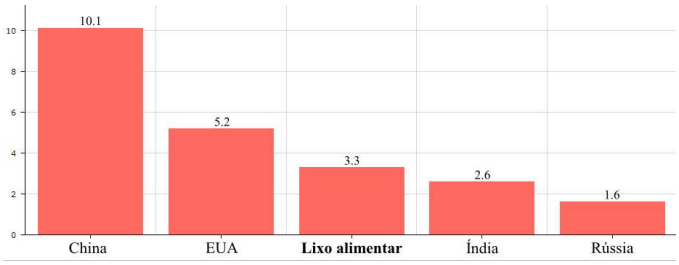


Figura 3 – Maiores emissores de CO2 (em gigatoneladas)

Embora os números do desperdício sejam alarmantes, falar em reduzir o descarte de alimentos não necessariamente significa redução nos níveis de fome, pois trata-se de fenômenos distintos, uma vez que, atualmente, a fome se caracteriza pela baixa renda e dificuldade de acesso aos alimentos. A produção é suficiente [17]. Essa realidade foi vivenciada por Carolina Maria de Jesus, escritora e ex-moradora da favela do Canindé. Em seu livro Quarto de Despejo, ela escreve:

> "Não mais se vê os corvos voando as margens do rio, perto dos lixos. Os homens desempregados substituíram os corvos. […]
>
> Hoje eu fiz arroz e feijão e fritei ovos. Que alegria! Ao escrever isto vão pensar que no Brasil não há o que comer. Nós temos. Só que os preços nos impossibilita de adquirir. Temos bacalhau nas vendas que ficam anos e anos a espera de compradores. As moscas sujam o bacalhau. Então o bacalhau apodrece e os atacadistas jogam no lixo, e jogam creolina para o pobre não catar e comer. Os meus filhos nunca comeu bacalhau." [18].

Carolina traz à tona a realidade feia, nua e crua que a sociedade tem medo de olhar. De um lado seres humanos que são forçados a se comportar como corvos para encontrar o que comer, do outro, toneladas de alimentos sendo jogados fora.

O aproveitamento de alimentos em iminência de descarte tem potencial de causar impacto positivo e melhorar realidades de pessoas como Carolina, uma vez que a ação tenha alcance e repercussão [13] [17].

Diminuir a quantidade de alimentos jogados no lixo significa reduzir o custo de gerir resíduos sólidos. Os recursos obtidos com essa ação podem ser utilizados para promover uma redução cada vez maior, através da implementação de novas políticas sustentáveis. [13].

C. *Efetividade de Doações*

Apesar de todas as medidas de redução de sobras que podem ser tomadas, elas sempre existirão, principalmente em restaurantes e supermercados. Uma boa forma de lidar com esse problema, é a doação.

Um estudo de caso realizado em restaurantes localizados em San Diego, revela que as doações de excedentes, apesar de seu alto impacto social, podem ter um baixo custo. Os resultados também mostram que essa atitude também pode trazer retornos financeiros, uma vez que os estabelecimentos se tornam elegíveis para receber incentivos fiscais [19]. No Brasil, os incentivos são possíveis através da Lei Nº 14.016, sancionada em 2020 [20].

Alguns outros exemplos de casos de sucesso são [21]:

- Carrefour: Mais de 2.5 toneladas de alimentos doados para programas de combate a fome e redução de preço em itens próximos do vencimento.
- Veneza: Doações semanais de 400 quilos de frutas e verduras que resultaram em uma redução de 70% do lixo orgânico.
- Mondelêz Brasil: Desde 2003 impacta mais de 1.2 milhão de pessoas com sua política de doações.

Organizados pelo Comitê Gestor, os bancos de alimentos contam com mais de 2 mil toneladas de alimentos arrecadados e doados. A participação na rede é voluntária e exige da unidade a comprovação de atividades previstas na Instrução Normativa nº 01, de 15 de maio de 2017 e assinatura de termo de compromisso. Em troca, o governo brasileiro oferece a oportunidade de participar de eventos e programas de capacitação [22] [23].

D. *Tecnologia Social*

Antes de abordar a tecnologia social, é importante entender o que se caracteriza como tecnologia convencional.

A tecnologia convencional foca principalmente no

aspecto da automatização ou facilitação do trabalho. Tal tecnologia atua no sentido de poupar o trabalho humano [24].

Tal tecnologia têm suas propriedades determinadas pelos mercados de alta renda dos países avançados. Dessa forma, ela é elaborada para satisfazer as demandas e interesses das mais altas classes sociais [24].

Por ser ditada pelos interesses capitalistas, a tecnologia convencional tende a ser ambientalmente insustentável, pois desconsidera a deterioração do meio ambiente [24].

Além disso, essa tecnologia é alienante e é elaborada de forma hierarquizada. Demanda a figura do proprietário, do chefe, etc [24].

Em contraponto com as características da tecnologia convencional, a tecnologia social foca em resolver problemas da sociedade e remete para propostas inovadoras de desenvolvimento, considerando a participação coletiva no processo de organização, desenvolvimento e implementação. O foco dessa tecnologia é apresentar soluções para problemas voltados a demandas de alimentação, educação, energia, habitação, renda, recursos hídricos, saúde, meio ambiente, dentre outras [24] [25].

As tecnologias sociais podem aliar saber popular, organização social e conhecimento técnico-científico. Importa essencialmente que sejam efetivas e reaplicáveis, propiciando desenvolvimento social em escala [25].

O uso da tecnologia social tem sido altamente difundido. Em novembro de 2004, foi realizada a I Conferência Internacional e Mostra de Tecnologia Social, na cidade de São Paulo [26].

Nesse evento estiveram presentes gestores de instituições públicas e privadas, empresários, lideranças comunitárias, empreendedores sociais, representantes governamentais e de organizações de pesquisa, entre outros, incluindo representantes de outros países que também trabalham com a lógica de tecnologia social [26] [27].

Ainda nessa conferência, foi proposta a criação da Rede de Tecnologia Social (RTS), que tem o propósito de promover o desenvolvimento sustentável mediante a difusão e a reaplicação em escala de tecnologias sociais [26].

E. *Demanda de Aplicativos Móveis*

Entre 2016 e 2020 o número de usuários de *smartphones* aumentou em 40%, atingindo 3.8 bilhões de dispositivos. A previsão é que em 2026, esse número chegue a 7.5 bilhões [28]. Além disso, estima-se que em 2025 72% dos usuários utilizarão apenas o celular para acessar a internet [29].

Já em 2016, um relatório divulgado pelo Pew Research Center revelou que 61% dos brasileiros entre 18 e 34 anos possuíam um *smartphone*. Usuários esses que cada vez mais, fazem uso de aplicativos. Em escala global, entre 2015 e 2016 o tempo gasto em aplicativos cresceu 25% [30].

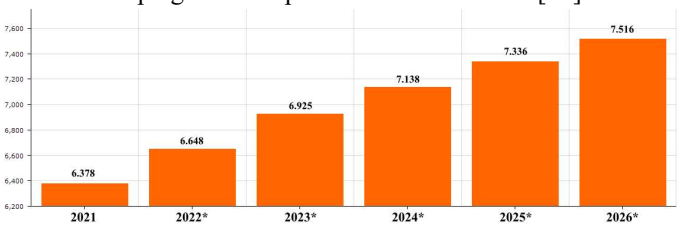


Figura 4 – Projeção do número de usuários de *smartphone* até 2026

Uma pesquisa realizada pelo CNDL (Confederação Nacional de Dirigentes Lojistas) em parceria com o SPC (Serviço de Proteção de Crédito) revela que 97% das pessoas da terceira idade estão conectadas à internet, 81% delas, através do *smartphone* [31].

54% dos idosos que usam a internet, buscam por informações sobre produtos e serviços e 72% deles, utilizam redes sociais.

F. *Acessibilidade e Experiência de Usuário*

A acessibilidade é fundamental para a democratização do acesso à tecnologia. A inclusão de deficientes e pessoas com limitações físicas é essencial. Tanto a Google quanto a Apple se preocupam com essas questões e disponibilizam *guidelines* sobre como desenvolver aplicativos acessíveis para suas plataformas. Melhorar a acessibilidade, significa melhorar a experiência dos usuários em geral, mas principalmente daqueles com baixa visão, cegueira, deficiência auditiva, prejuízos cognitivos e com deficiências motoras ou situacionais.

Segue abaixo uma descrição dos principais requisitos para uma interface *mobile* acessível, segundo Apple e Google [32][33]

1. Hierarquia: Os elementos principais precisam ser destacados. Uma boa forma de fazer isso, é os colocar no topo. Os conteúdos apresentados também precisam seguir ordem lógica, de forma que faça sentido para o usuário a leitura horizontal das informações;
2. Contraste: Textos com letras pequenas precisam ter a taxa de contraste de cores de pelo menos 4,5:1. Para textos maiores, a taxa de contraste de cores deve ser de pelo menos 3,0:1.
3. Alvos de toque: Estes precisam possuir área de toque de pelo menos 44 dp (*density pixel*).
4. Agrupamento: Itens semelhantes devem ser agrupados juntos.
5. Tipografia: Os usuários devem poder aumentar o tamanho da fonte pra melhorar a legibilidade do conteúdo. A interface deve ter o seu *layout* preservado.

6. Texto de acessibilidade: Os elementos da interface devem ter suas ações descritas para aprimorar o uso dos leitores de tela.
7. Identificadores: Os itens não devem ser identificados apenas por cores

Um levantamento realizado pela Big Data Corp, apurou que dentre os 2068 aplicativos mais populares no Brasil, 63,46% descrevem menos de 10% dos elementos de interface visual e apenas 0.34% apresentam a descrição para mais de 75% dos seus componentes [34].

Uma alternativa que pode melhorar a visibilidade de informações para alguns usuários é o modo escuro. Embora esse tipo de tema não seja o preferido pela maioria dos usuários [35], um experimento realizado por [36] revela que os participantes tinham uma acuidade visual significativamente maior para o modo escuro do que para o modo claro. Os usuários submetidos ao teste também cometeram menos erros de percepção ao utilizar o modo escuro.

A alta luminosidade das telas utilizando o tema claro pode ser incômodo para alguns usuários e também tem a capacidade de aumentar a fadiga visual devido a alta luminescência emitida.

Tanto o sistema Android como iOS disponibilizam alternativas de tema escuro para seus usuários. Por isso, é importante que os aplicativos também contem com versões com baixa luminescência de seus temas principais, tornando a experiência do usuário livre de quebras de padrões.

Don Norman, em seu livro Design do Dia a Dia fala sobre a importância da consistência. Uma interface considerada consistente não é somente aquela que implementa os temas de forma uniforme, mas também a que atribui elementos similares a operações similares [37].

Ainda em seu livro, Norman aborda outros conceitos que quando implementados, aprimoram a experiência do usuário.

1. Os componentes de maior importância devem ter maior visibilidade
2. A interface deve conceder *feedbacks* ao usuário. Áudio, tátil, visual ou uma combinação deles.
3. Os limites e restrições devem ser definidos de forma que impeça o usuário de seguir um caminho inadequado,
4. Os elementos visuais devem ter correspondência ao mapeamento cerebral natural dos usuários.
5. Os elementos da interface devem ser auto-descritivos, comunicando ao usuário aquilo que significam, ou a operação que realizam.

III Trabalhos Correlatos

Entre os trabalhos correlatos, destaca-se "Comida Invisível", o aplicativo que mais está relacionado ao que foi desenvolvido nesse projeto e pode ser encontrado em versões WEB e Android.

Apoiado pela cozinheira Paola Carosella e pelo time de futebol Santos FC, a plataforma tem como objetivo principal o combate do desperdício de alimentos através de doações. O sistema permite o cadastro de produtos e o requerimento deles por pessoas e instituições. As principais funcionalidades disponíveis são:

- Visualização no mapa de alimentos em doação.
- Requerimento de doação.
- Verificação de restaurantes (dependendo da disponibilidade da equipe)
- Avaliação de doadores e donatários após a retirada do alimento requerido.
- Indicadores de não desperdício de água, energia, gás, baseados nas informações coletadas de outros usuários da região.
- Chat após requerimento ser aceito pelo doador.

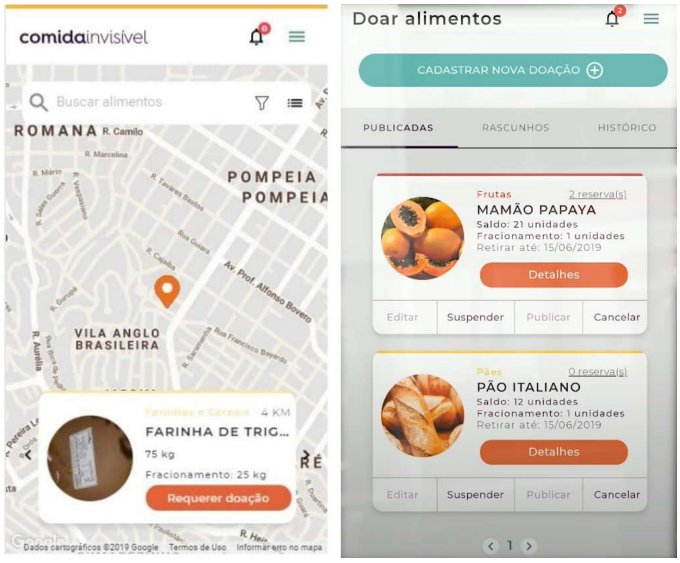


Figura 5 – Telas principais do aplicativo "Comida Invisível".

Apesar de ótima proposta, o aplicativo não se apresenta funcional, e isso pode ser constatado ao observar as avaliações disponíveis na loja de aplicativos do Android. O aplicativo acumulou mais avaliações negativas do que positivas e não recebe atualizações desde setembro de 2019. A versão web por sua vez, apresenta-se funcional.

Alguns outros aplicativos que lidam com o tema de forma indireta são:

- Joyz Doação: Possibilita apoiar ONGS, abrigos e outras instituições de apoio à causas nobres. A

plataforma permite trocar moedas internas por doações.

- Doador: Permite a doação de diversos itens de diversas categorias diferentes. Remédios, moveis, alimentos, etc.
- Ribon: Exibe uma lista de diversas instituições de caridade e permite que doações sejam feitas de forma gratuita.
- Doação do Bem: Permite que o usuário realize uma doação para instituições pré-selecionadas.
- Small Doações: Segmenta instituições de caridade por categoria e conecta o usuário a elas.

IV A SOLUÇÃO DESENVOLVIDA

A presente seção tem como objetivo destrinchar a solução desenvolvida, apresentando as linguagens utilizadas em sua construção, arquitetura e respectivos diagramas.

A subseção IV-A apresenta o *framework* utilizado para construir o aplicativo mobile, diagramas de sua construção e aborda o fluxo do seu funcionamento. A subseção IV-B apresenta o sistema *backend, a* sua arquitetura, o *framework* web onde foi desenvolvido, e aspectos importantes sobre componentes e conectores internos. A subseção IV-C apresenta o banco de dados responsável por armazenar os dados da aplicação e seu diagrama relacional. Por fim, a subseção IV-D revela as funcionalidades do aplicativo, apresentando seus requisitos, casos de uso, *screenshots* e principais vantagens de utilizar o sistema.

A. *Mobile*

O cliente mobile foi elaborado para atender aos seguintes requisitos não funcionais:

Identificador	**Nome**
MRNF01	Usabilidade: O aplicativo deve favorecer a experiência do usuário**.**
MRNF02	Compatibilidade: O aplicativo ser compatível com a maioria dos *smartphones*.
MRNF03	Segurança: O aplicativo deve estar de acordo com as políticas de segurança contidas nas *guidelines* do sistema onde será implantado.
MRNF04	Manutenibilidade: O aplicativo deve ter arquitetura bem definida que favoreça a sua manutenção.
MRNF05	Internacionalização: O aplicativo deve ter estrutura que suporte a adição de novos idiomas.

Tabela 1 – Requisitos não funcionais do aplicativo.

O aplicativo foi desenvolvido utilizando Flutter[1], *framework open source* criado sobre a linguagem Dart e mantido pela Google.

Flutter possibilita o desenvolvimento multiplataforma. Dessa forma, um único código é capaz de ser executado por sistemas diferentes, como Android, iOS e WEB. O processo de construção de aplicativos com essa tecnologia se torna muito mais fluído, ágil e produtivo, sem que haja perdas significativas de performance no produto final [38].

Flutter oferece diversas ferramentas que auxiliam o desenvolvedor durante o processo de desenvolvimento, como o *hot reload* e *hot refresh*. Além disso, a construção de layouts com essa ferramenta é mais simplificada, quando comparada ao desenvolvimento nativo. Flutter usa *widgets* para criação de componentes, que são estruturados como árvore.

Nos últimos anos Flutter se tornou muito popular, sendo usada por grandes empresas, como Nubank, Ebay, BMW, Alibaba e Capital One.

Para o desenvolvimento dessa aplicação, foi escolhida uma arquitetura em camadas, focada no desacoplamento de código, de forma que cada módulo seja independente e interoperante.

A arquitetura em camadas favorece a modularidade do código, a simplicidade de implementação, a manutenibilidade, a flexibilidade, escalabilidade, portabilidade e estabilidade geral [39].

Uma boa arquitetura simplifica a compreensão, o desenvolvimento e a implantação do sistema. O objetivo final de uma boa arquitetura deve ser minimizar o custo de manutenção durante a vida útil do *software* e maximizar a produtividade dos desenvolvedores [40].

A arquitetura elaborada para o aplicativo *mobile* foca na separação dos seguintes conceitos:

1. Camada de Regras de Negócio
2. Camadas de Regras da Aplicação
3. Adaptadores de Interface
4. *Plugins* e bibliotecas externas.

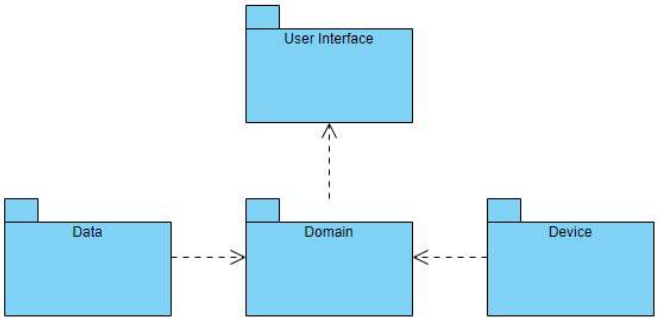


Figura 6 – Diagrama de pacotes do aplicativo.

1 - Site da ferramenta: flutter.dev

A Figura 6 demonstra a forma que os pacotes do aplicativo estão organizadas.

O Data é o responsável pela comunicação http com o *web service,* implementando todas as regras de conexão.

O Device representa a camada de dados interna e contém os *plugins* e as bibliotecas externas que podem ser substituídas ao longo do tempo.

A camada de apresentação corresponde a interface de usuário e contém os componentes responsáveis pela exibição e lógica de exibição;

O Domain representa o *core* do sistema sendo responsável por coordenar e definir todos os processos e especificando as interfaces que precisam ser implementadas. Os componentes desse pacote realizam a comunicação entre o Data e o Device com os componentes da interface do usuário.

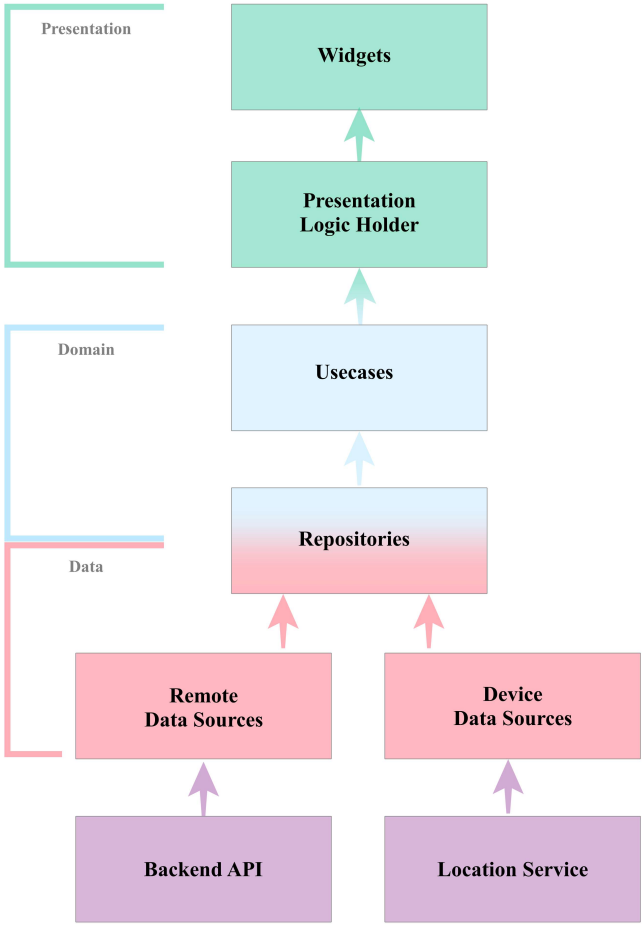


Figura 7 – Arquitetura em camadas do aplicativo.

A Figura 7 demonstra a arquitetura geral do sistema mobile. A conexão entre os componentes de apresentação se dá através do conector de eventos e de chamadas de métodos. Já a comunicação entre o Repository e a API se dá através do conector Distributor. A camada "Remote Data Sources" é responsável por se conectar com a API através da rede. Ele implementa os protocolos http e *websocket*.

Há uma comunicação em dois lados envolvendo os *widgets* e os componentes responsáveis por implementar a lógica de negócio. Tal comunicação ocorre em cenários de I/O. Os *widgets* captam a entrada de dados que partem do usuário e o "Presentation Holder" emite eventos que notificam os *listeners* quando precisam se atualizar. Flutter implementa o conceito de *widgets* imutáveis. Dessa forma, um *widget* nunca pode ser alterado. Caso haja necessidade de atualização, o *widget* tem o seu lugar na árvore substituído por outro *widget* que contém os dados atualizados.

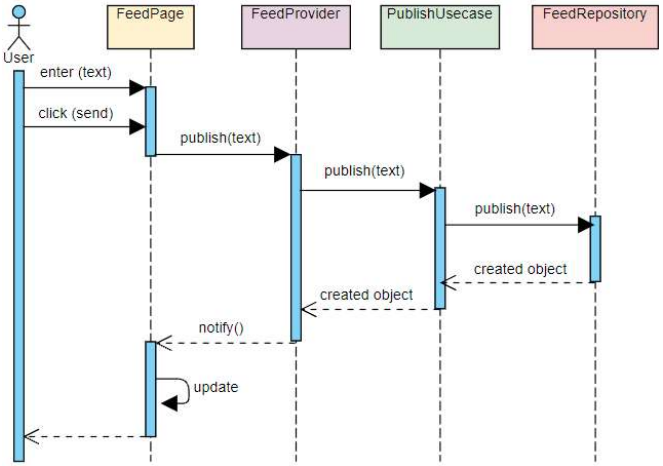


Figura 8 – Diagrama de sequência do processo de envio de uma publicação no aplicativo.

A Figura 8 que apresenta o fluxo de comunicação entre as camadas do sistema usando como exemplo o processo de postagem de uma publicação no mural do aplicativo.

O aplicativo foi publicado na Play Store em modo de teste interno. O tamanho do arquivo de instalação é de 20 *megabytes* e após instalado ocupa 27 *megabytes* da memória do dispositivo.

O Aplicativo é compatível com versões Android maiores ou iguais a 5.0 Lolipop, totalizando 15.622 dispositivos.

Não foi realizada a compilação do aplicativo em iOS devido à ausência de dispositivos da Apple necessários para realizar tal operação durante o desenvolvimento.

B. *Backend*

O *backend* foi elaborado para atender aos seguintes requisitos não funcionais:

Identificador	**Nome**
BRNF01	Disponibilidade: O *backend* deve estar disponível na maior parte do tempo.
BRNF02	Integridade: O *backend* deve permitir o

	acesso apenas de usuários autenticados.
BRNF03	Segurança: O *backend* deve implementar políticas de para proteger credenciais dos usuários.
BRNF04	Manutenibilidade: O *backend* deve ter arquitetura bem definida que favoreça a sua manutenção.
BRNF05	Escalabilidade: A arquitetura *backend* deve permitir o crescimento do sistema.

Tabela 2 – Requisitos não funcionais do *backend*.

O *backend* foi desenvolvido utilizando o *framework* web FastAPI[2]. Escrito em Python, essa ferramenta tem um alto desempenho e fornece uma interface intuitiva e robusta para o desenvolvedor. FastAPI é uma estrutura de aplicação web que fornece uma série de funcionalidades como validação de dados, serialização e documentação.

Essa ferramenta utiliza o Starlette[3] como *framework* ASGI (Asynchronous Server Gateway Interface). Starlette fornece um ambiente *production-ready* e dá suporte a *websockets*, o que é importante para o sistema desenvolvido.

Atualmente, FastAPI é usado por empresas como: Microsoft, Uber, Netflix, Hug e SpaCy.

A segurança do sistema é baseada em JWT (JSON Web Token). O *token* é concedido após o *login* e precisa ser fornecido a cada requisição para garantir a autorização de visualização de dados.

Para evitar que um possível vazamento comprometa outras contas de um usuário, as senhas são criptografadas antes de serem armazenadas. Para isso, é utilizado o método Bcrypt, que utiliza criptografia do tipo *hash.* Quando o usuário realiza o *login*, o *hash* da senha digitada é comparado com o *hash* da senha armazenada no banco de dados.

O *backend* usa recursos disponibilizados pelo Firebase[4]. Cloud Messaging e Storage, para enviar notificações e armazenar fotos, respectivamente.

O Firebase é uma plataforma desenvolvida pela Google que auxilia no desenvolvimento de aplicativos móveis e web. Possui uma fácil integração com diversos sistemas e disponibiliza generosos limites de uso em seu plano gratuito.

Os links públicos para as imagens, são temporários e são renovados a cada hora. Para evitar sobrecarga e acessos desnecessários ao Firebase, foi implementado um serviço de *cache* de imagens utilizando Redis[5], um armazenamento de dados em memória, usado como um banco de dados temporário.

O sistema desenvolvido tem a necessidade de se comunicar com o banco de dados e realizar consultas complexas que sejam executadas em um ambiente que esteja em consonância com o ACID.

Para satisfazer todos os princípios requeridos pelo ACID, é necessária a existência de um objeto que coordene todos os outros e possibilite que as operações sejam sincronizadas, possibilitando que os objetos resultantes de uma tabela sempre a representem. Para esse objetivo seja atingido, foi escolhida a ORM SQLAlchemy[6].

SQLAlchemy satisfaz esses princípios da seguinte forma:

- Atomicidade: Em caso de falha, todas as transações relacionadas com a transação que falhou sofrem *rollback*.
- Consistência: Todos os dados a serem persistidos passam por verificações que constatam se as *constraints* definidas no *schema* foram atendidas.
- Isolamento: As transações são executadas isoladas umas das outras de forma concorrente.
- Durabilidade: Após o *commit*, todos dados são presentes no Identity Map são armazenados no banco de dados de forma segura

SQL Alchemy implementa o conceito de sessão. A sessão possibilita que todas as transações relacionadas a um banco de dados envolvendo um determinado contexto, sejam retidas até que haja uma instrução de persistência ou reversão [41].

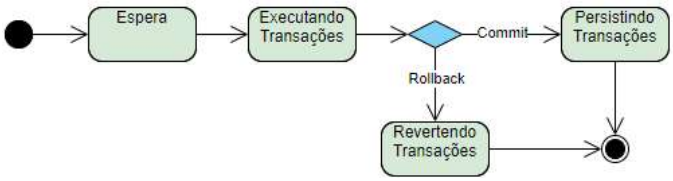


Figura 9 – Diagrama de estados do SQLAlchemy.

Em uma sessão, cada objeto atua como *proxy* de uma linha no banco de dados, de tal maneira que as informações contidas no objeto sempre estarão atualizadas. Quando a transação é encerrada, os objetos relacionados a ela são marcados como “expirados”. Quando uma nova consulta é feita àquele objeto, a sessão se encarrega de atualizá-lo com as informações mais recentes contidas no banco de dados [41].

Além disso, em cada consulta ao banco de dados, a sessão se encarrega de certificar que não existam transações em andamento que estejam relacionadas às linhas requeridas. Caso haja, impede que as informações sejam consultadas antes que as transações pendentes sejam persistidas. Quando a transação bloqueante é finalizada, tendo ela sido bem-sucedida ou não, a sessão permite que a

2 - Site da ferramenta: fastapi.tiangolo.com

3 - Site da ferramenta: starlette.io

4 - Site da ferramenta: firebase.google.com

5 - Site da ferramenta: redis.io

6 - Site da ferramenta: sqlalchemy.org

transação pendente seja executada. Para atingir esse objetivo, o *pattern* Unit of Work é utilizado [41].

Esse *pattern* é utilizado para manter uma lista dos objetos afetados por uma transação e para coordenar a leitura e escrita de novos dados. Dessa forma, tudo que é feito durante as transações é acompanhado pelo *pattern,* que por fim, realiza todas as modificações necessárias para que os novos dados sejam persistidos ou lidos.

> *"The key thing about Unit of Work is that, when it comes time to commit, the Unit of Work decides what to do. It opens a transaction, does any concurrency checking […], and writes changes out to the database. Application programmers never explicitly call methods for database updates. This way they don't have to keep track of what's changed or worry about how referential integrity affects the order in which they need to do things.*
>
> *With caller registration, the user of an object has to remember to register the object with the Unit of Work for changes. Any objects that aren't registered won't be written out on commit.*" [42]

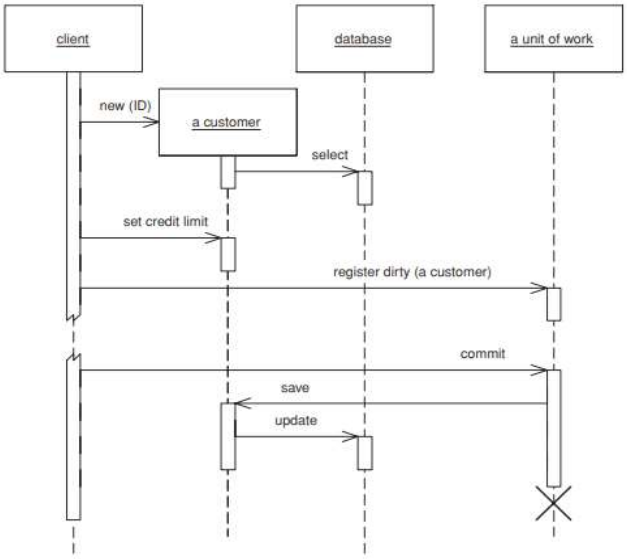


Figura 10 – Diagrama de sequência do Unit of Work [42].

A Figura 10 demonstra o funcionamento do *pattern* em um diagrama de sequência. Após o *commit* da transação, o Unit of Work se encarrega de realizar os trâmites necessários para que o dado seja salvo de forma correta.

A ORM usa o Identity Map para armazenar o estado dos objetos envolvidos em uma transação.

O Identity Map é utilizado para mapear todos objetos provenientes do banco de dados e para assegurar que cada objeto carregado seja único. Dessa forma, cada linha resultante de uma consulta precisa ser verificada.

Esse *pattern* atua como um *cache,* evitando que novas consultas ao banco sejam feitas sem necessidade. Caso o objeto já esteja em memória, ele é retornado. Sendo assim, se em uma mesma sessão a mesma linha do banco de dados for consultada múltiplas vezes, o mesmo *proxy* será retornado, o que favorece o princípio do isolamento.

Os objetos são armazenados enquanto a transação estiver em andamento. Uma vez que haja o *commit* ou *rollback*, todos os dados expiram [42], [43].

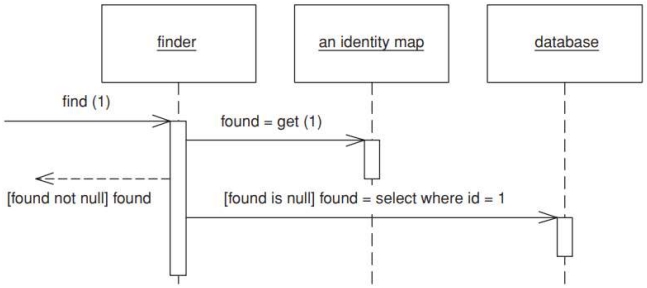


Figura 11 – Diagrama de sequência do Identity Map [42].

A Figura 11 demonstra como o *pattern* funciona. O método *find* recebe como parâmetro o identificador do objeto consultado, que caso seja encontrado no Identity Map, o método retorna sem realizar alterações. Caso não seja encontrado, o novo objeto é adicionado ao *map*.

Todo o processamento exigido por esse *pattern* pode ter um alto custo performático, devido à necessidade de verificar se cada linha retornada por uma consulta já possui um objeto correspondente armazenado no *map*. Entretanto, esse comportamento mantém a sessão com objetos únicos, eliminando a redundância no carregamento de dados e reafirmando o princípio do isolamento. A redução no carregamento de dados pode compensar o custo de verificação.

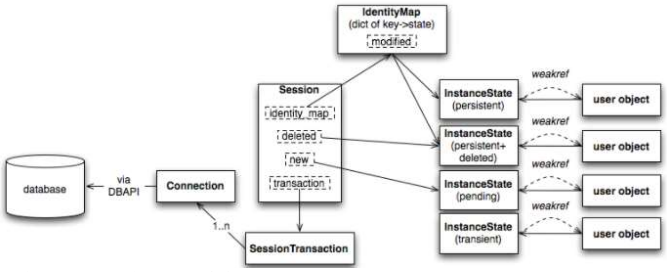


Figura 12 – Visão geral da sessão [43].

A Figura 12 representa as relações que envolvem a sessão, tendo o Identity Map como responsável por armazenar o estado dos objetos.

Para esse projeto, a sessão foi atrelada às requisições http através de um *middleware.* Dessa forma, a cada nova requisição, uma nova sessão é iniciada, garantindo o isolamento e confiabilidade dos resultados. Se torna impossível que dados sejam parcialmente salvos em casos de inserções em múltiplas tabelas.

A arquitetura do sistema é uma arquitetura em camadas que se baseia no MVC.

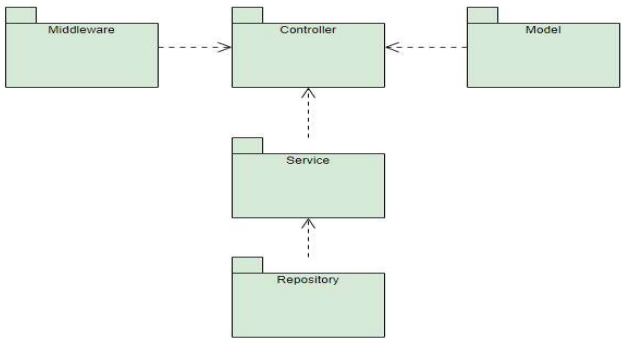


Figura 13 – Diagrama de pacotes do backend.

O diagrama de pacotes exibido na Figura 13, demonstra como os pacotes do sistema estão organizados.

- Controller: é responsável pela definição dos *endpoints* e coordenação dos processos.
- Service tem como papel executar a lógica de negócio e realizar conversões nos modelos.
- Repository: estrutura as consultas utilizando a ORM e as executa.
- Model, define a representação orientada a objetos das tabelas contidas no banco.
- Middleware: Contém implementação de *middlewares* que são executados antes do Controller.

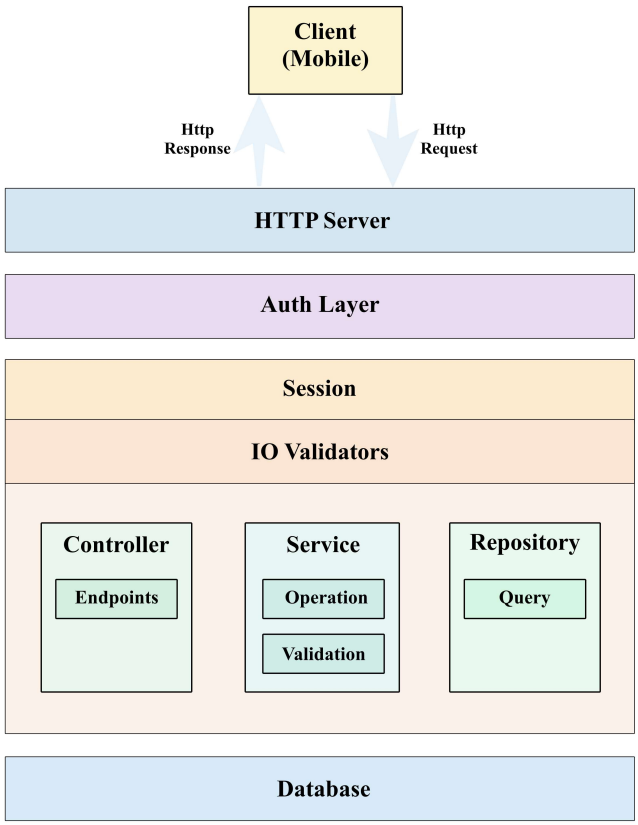


Figura 14 – Arquitetura do *backend*.

A Figura 13 contém o diagrama arquitetural do *backend*. Após a conexão http, existe uma camada de autenticação que valida a requisição. Após isso, é iniciada a sessão. Os dados de *input* e *output* são validados conforme os *schemas* definidos. Em caso de incompatibilidades, os processos são abortados.

O sistema usa conectores do tipo "Procedure Call" para realizar a comunicação interna e usa a ORM SQL Alchemy, (que pode ser abordada como uma combinação dos conectores Procedure Call, Data Access, Distributor e Adaptor) para se comunicar com o banco de dados através da rede.

C. *Banco de Dados*

O banco de dados escolhido para esse projeto foi o PostgreSQL[7]. Esse SGBD (Sistema Gerenciador de Banco de Dados) é *open source* e oferece um sistema inovador de gerenciamento de dados conhecido por sua confiabilidade e robustez.

Esse sistema se beneficia de mais de 25 anos de desenvolvimento de código aberto de uma comunidade global de desenvolvedores e tornou-se o banco de dados relacional de código aberto preferido para organizações de todos os tamanhos.

O diagrama relacional construído para a aplicação apresentada pode ser visualizado no Anexo I.

D. *Funcionalidades*

Após identificar as necessidades de comunicação entre doadores e donatários, foram definidas os requisitos para que o aplicativo desenvolvido fosse eficiente em atender a esse público.

A tabela a seguir apresenta os requisitos funcionais da aplicação:

Identificador	Nome
RF01	O sistema deve permitir a visualização de doações no mapa em um raio de 60 km.
RF02	O sistema deve realizar diferenciação visual de perfis no mapa.
RF03	O sistema deve permitir a visualização de perfis de ONGs e estabelecimentos no mapa.
RF04	O sistema deve permitir o início de conversas com ONGs e estabelecimentos.
RF05	O sistema deve permitir a adição de alimentos para doar.
RF06	O sistema deve permitir a reserva dos

7 - Site da ferramenta: postgresql.org

	alimentos se assim for desejado pelo doador.
RF07	O sistema deve permitir o início de uma conversa após a realização ou recebimento de reserva.
RF08	O sistema deve permitir a ONG ou estabelecimento decidir se quer receber mensagens sem reserva anterior.
RF09	O sistema deve suportar o gerenciamento de múltiplos endereços.
RF10	O sistema deve notificar o usuário sobre novas reservas recebidas.
RF11	O sistema deve notificar o usuário sobre reservas canceladas.
RF12	O sistema deve exibir uma lista com todos os chats ativos.
RF13	O sistema deve exibir uma lista com todas as doações em um raio de 60 km.
RF14	O sistema deve permitir o filtro de alimentos por nome ou categoria.
RF15	O sistema deve permitir o gerenciamento de doações.
RF16	O sistema deve permitir o gerenciamento de reservas recebidas.
RF17	O sistema deve permitir o gerenciamento de reservas realizadas.
RF18	O sistema deve permitir a avaliação do perfil doador após a retirada da reserva.
RF19	O sistema deve permitir a visualização das avaliações recebidas por um perfil.
RF20	O sistema deve permitir a visualização das avaliações realizadas.
RF21	O sistema deve permitir a alteração de dados do perfil do usuário.
RF22	O sistema deve suportar a alteração de tamanhos de fonte.
RF23	O sistema deve ter suporte a tema escuro.
RF24	O sistema deve permitir ao usuário selecionar as notificações que deseja receber.
RF25	O sistema deve permitir ao usuário escrever uma publicação no mural.
RF26	O sistema deve exibir todas as publicações contidas no mural.
RF27	O sistema deve permitir curtidas em uma publicação.
RF28	O sistema deve permitir comentários em uma publicação.
RF29	O sistema deve exibir todos os comentários em uma publicação.
RF30	O sistema deve exibir a quantidade de curtidas e comentários em uma publicação.
RF31	O sistema deve permitir a exclusão de uma publicação.
RF32	O sistema deve permitir a exclusão de um comentário.
RF33	Os componentes do sistema devem ter descrição compatível com “Talk Back” e “Voice Over”.

Tabela 3 – Requisitos funcionais do sistema.

Os requisitos estabelecidos para o sistema, possibilitam que haja uma comunicação eficiente entre doadores e donatários. Além disso a acessibilidade e usabilidade não foi deixada de lado, tendo requisitos que focam especificamente nesses aspectos.

Os casos de uso do sistema podem ser visualizados no Anexo II.

O aplicativo funciona com base na localização do usuário, por esse motivo, permitir o acesso ao local é necessário para o acesso ao sistema.

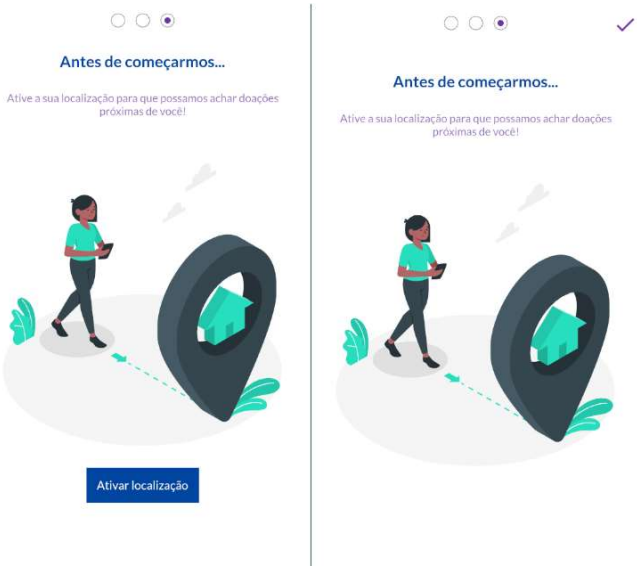


Figura 15 – Tela de boas vindas.

A Figura 15 exibe a tela de boas vindas, onde é feita uma introdução do usuário ao aplicativo e em seguida solicita o acesso à sua localização. Essa permissão precisa ser concedida apenas uma vez. Após concedida, os dados de latitude e longitude são armazenados na memória interna do dispositivo para uso posterior.

Em seguida, o usuário é direcionado para a tela de login, para se autenticar no aplicativo, ou fazer o cadastro caso ainda não tenha conta.

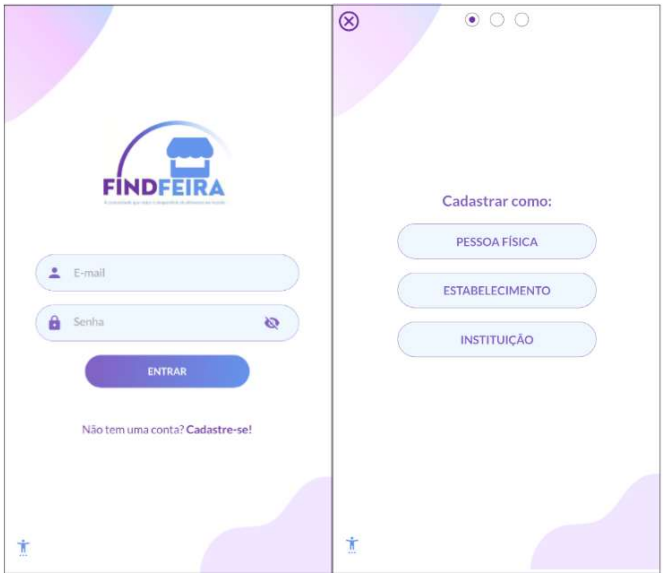


Figura 16 – Tela de login.

Na Figura 16 pode-se visualizar a tela de login e uma das etapas do cadastro. O cadastro possui 3 etapas, envolvendo dados da pessoa física ou jurídica e o endereço principal.

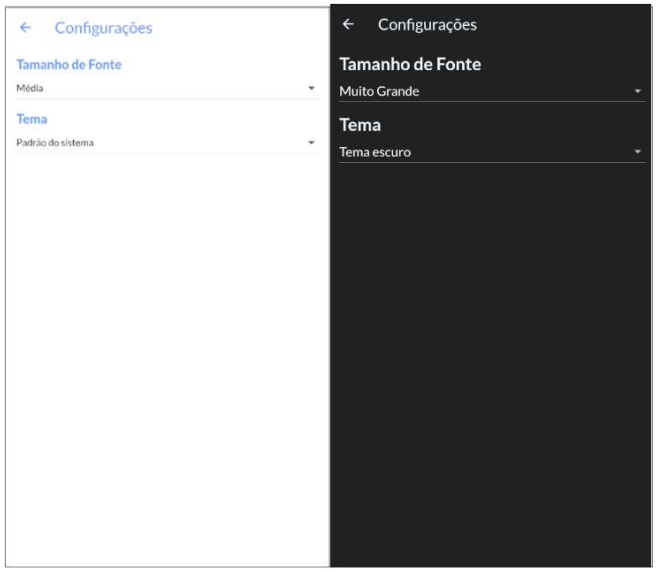


Figura 17 – Tela de configurações de acessibilidade.

Durante o login, o usuário também pode ativar opções de acessibilidade, como alterar o tamanho das letras e o tema do aplicativo. Essas opções também podem ser alteradas após o login.

Após ter se autenticado, o usuário é redirecionado para a tela principal, onde contém o mapa. No mapa podem ser visualizadas as doações ao redor da localização.

Os estabelecimentos, instituições e pessoas físicas são identificadas no mapa através de ícones. Estabelecimentos e instituições ficam sempre visíveis no mapa, mesmo que não tenham doações ativas. O objetivo é informar o usuário sobre os perfis ativos na plataforma e permitir a comunicação com as instituições de apoio, fornecendo o chat para resolução de dúvidas e a localização.

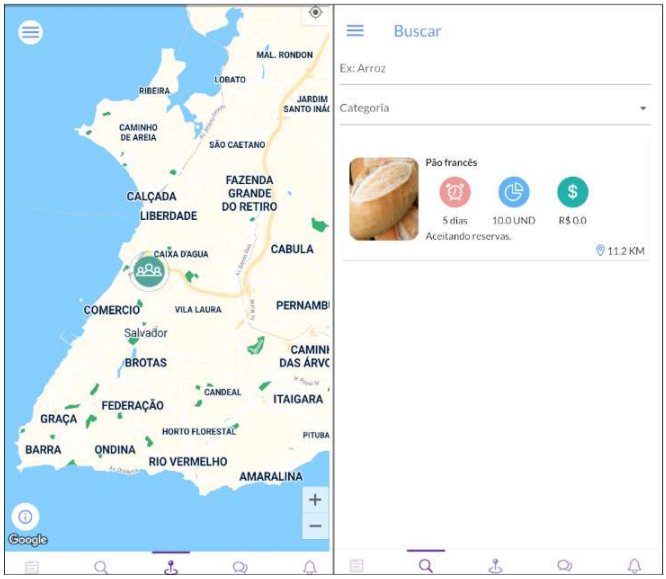


Figura 18 – Mapa e lista de alimentos.

Na aba de busca, se pode visualizar as doações em forma de lista, podendo filtrá-las por categoria ou nome. Os alimentos são listados com *badges* que indicam até que quando o alimento estará disponível, a quantidade total de itens e um *badge* indicando a gratuidade.

Também é indicado se o alimento pode ser reservado através do aplicativo e a distância em linha reta até o endereço do doador.

Na tela de detalhes de um alimento é possível abrir um aplicativo externo para visualizar as rotas até o endereço onde ele está armazenado.

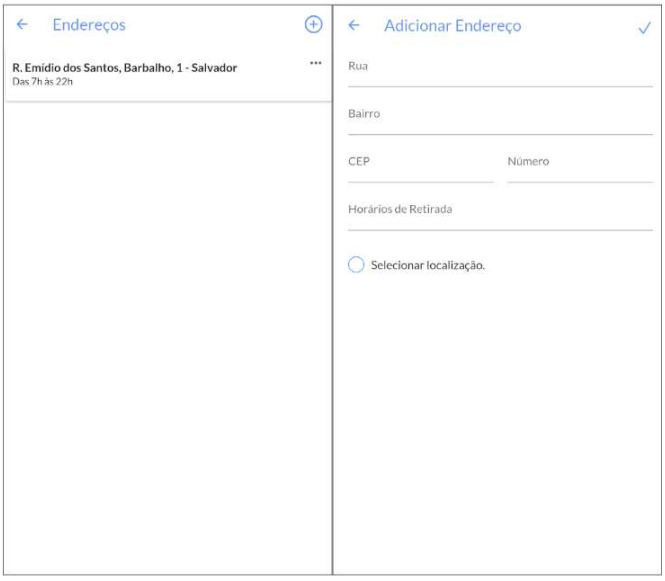


Figura 19 – Tela de endereços.

O usuário também pode adicionar ou editar seus endereços. Para isso, precisa preencher os campos correspondentes ao endereço que deseja adicionar e selecionar a localização do endereço no mapa.

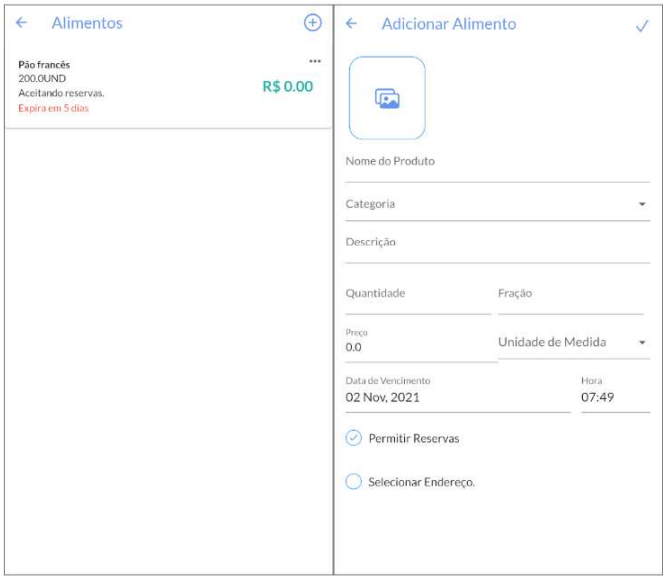


Figura 20 – Tela de alimentos.

Para criar um alimento, o usuário precisa informar o nome, a categoria e uma breve descrição. Também é preciso informar a quantidade disponível, a fração, a unidade de medida, até quando o produto estará disponível e se ele poderá ser reservado pelo aplicativo. Por fim, é necessário vincular um endereço previamente criado a um produto.

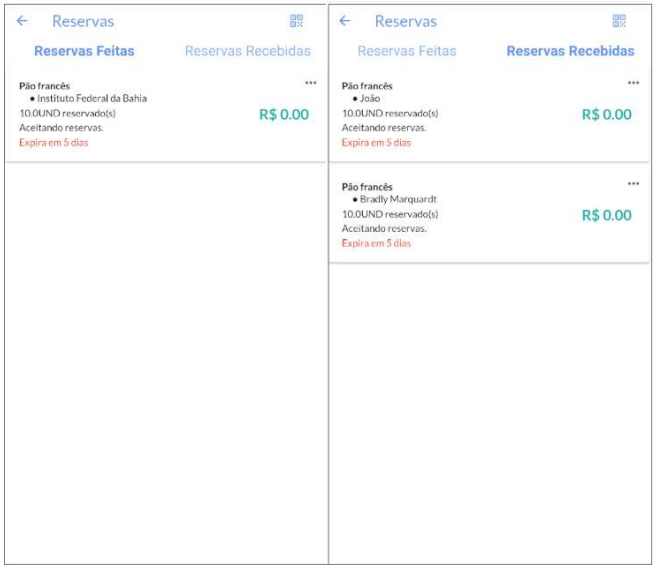


Figura 21 – Tela de reservas.

O usuário pode visualizar as reservas que realizou e que recebeu. A partir dessa tela é possível iniciar uma nova conversa e cancelar as reservas recebidas ou realizadas,

Para os alimentos reservados é possível gerar um código de resposta rápida (QR) que pode ser apresentado no momento de retirada, para facilitar o processo. Após a leitura ter sido feita, uma caixa de diálogo é exibida com os itens reservados e a reserva é marcada como retirada. O processo também pode ser feito de forma manual.

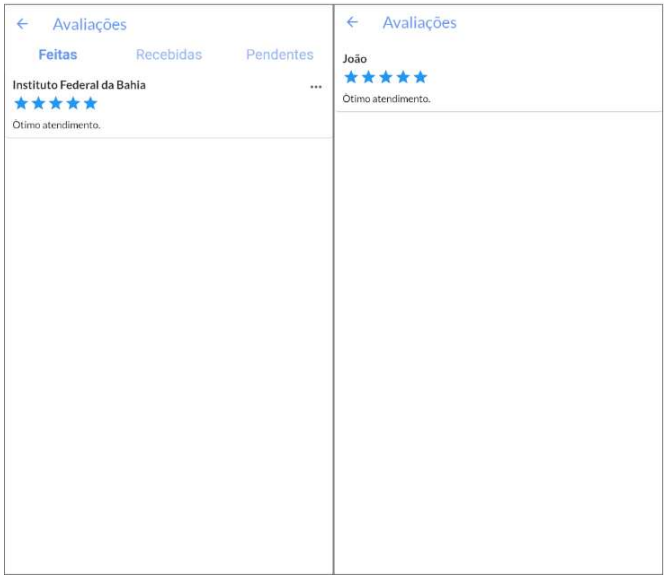


Figura 22 – Telas de avaliação do aplicativo.

Após uma reserva ser retirada, os usuários podem avaliar o doador. A avaliação pendente pode ser visualizada na aba correspondente e após ser efetuada é adicionada à aba “Feitas”. O usuário que recebeu avaliação pode visualizá-la na aba “Recebidas”.

A média e a quantidade das avaliações ficam visíveis publicamente no perfil de quem recebeu. A lista de avaliações pode ser visualizada após o clique do usuário nos *badges* de estrelas.

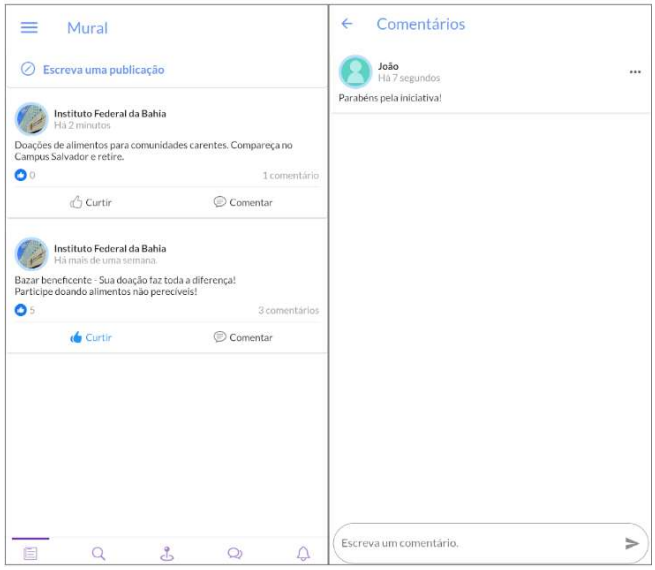


Figura 23 – Mural do aplicativo.

O aplicativo tem um mural de publicações que podem ser visualizadas por todos usuários cadastrados. O mural pode ser usado para publicar notícias relevantes ou compartilhar eventos e ações beneficentes. Os usuários

também podem interagir curtindo ou comentando as publicações.

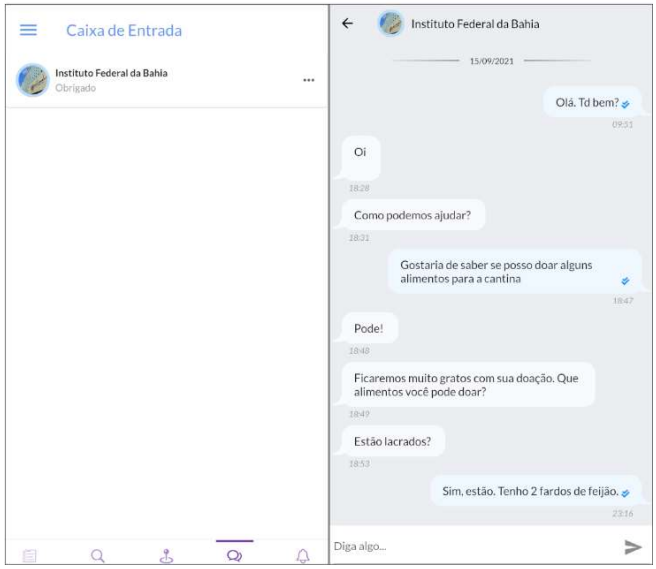


Figura 24 – Chat do aplicativo.

Outra forma de interagir é através do chat. O chat é ativado sempre que há uma reserva e os usuários podem entrar em contato com estabelecimentos e instituições sem reserva prévia, se assim for permitido por elas.

O chat permite saber quando a mensagem foi enviada e se foi visualizada pelo destinatário.

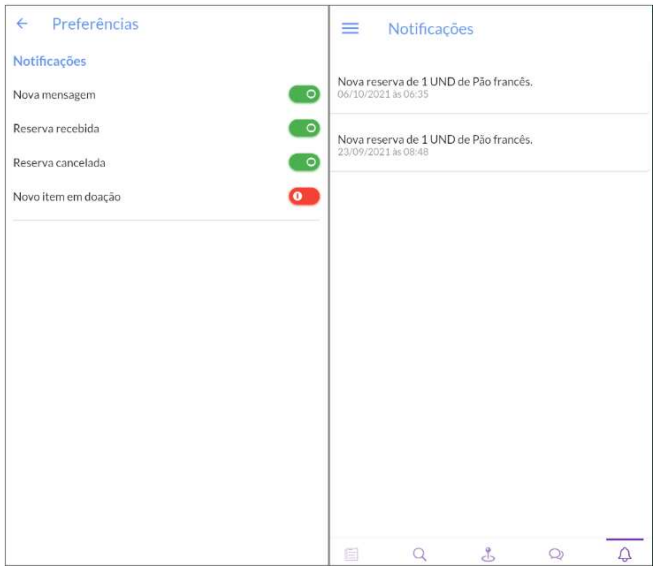


Figura 25 – Tela de preferências e notificações

O usuário também pode escolher quais notificações deseja receber ao acessar as preferências de sua conta. As notificações permitidas pelo usuário são entregues por meio de *push notifications* e também podem ser visualizadas dentro do aplicativo.

V AVALIAÇÕES E RESULTADOS

Tendo em vista que o sistema desenvolvido tem requisitos promissores quanto a proposta, realizou-se um teste com usuários para atestar a usabilidade do aplicativo.

Os testes foram realizados por usuários com experiência profissional em desenvolvimento *frontend* entre os dias 02 de outubro e 06 de novembro. O acesso ao aplicativo foi disponibilizado através da “Play Store”.

Foram definidas metas para que os usuários cumprissem, sendo elas:

1. Cadastrar-se.
2. Deslogar do aplicativo.
3. Realizar login.
4. Reservar um alimento.
5. Publicar um alimento.
6. Editar um alimento.
7. Retirar um alimento.
8. Enviar mensagem para uma instituição.
9. Alterar o tema do aplicativo.
10. Alterar preferências de notificações.
11. Interagir com uma publicação no mural.

Após a conclusão das metas, foi solicitado que fosse preenchido um formulário de *feedback* para avaliar a relevância percebida das funcionalidades e medir dificuldades de usabilidade. O formulário pode ser visualizado no Anexo III e no Anexo IV.

Como resultado, todos os usuários acharam as funcionalidades relevantes.

No formulário, foi aberto um campo opcional para que os usuários comentassem de forma livre sobre as funcionalidades. Foi possível colher os seguintes *feedbacks:*

> “Essa área [mural de publicações] foi fundamental, pois facilita tanto a divulgação quanto a exibição de doações.”
>
> “[O mapa] facilita achar a localização das doações.”
>
> “Muito interessante que [a tela de busca] tem a parte de categorias, deixando mais específica a busca, sem rodeios.”
>
> “[A retirada de alimentos é] muito prática, pois apenas com o QR já registra a ação do usuário, o tipo de doação e quando foi feita.”
>
> “Esse aplicativo aborda um tema que cresceu no último ano, principalmente com a pandemia e ações do governo, que é a fome. As ONG's existentes pode utilizar este aplicativo e usar como um facilitador para as suas ações, pois há aqueles que não possui um celular, ou algo relacionado, que não saberão

onde pode pegar os alimentos, então as ONG's poderiam agir nesse quesito. Além do mais, as pessoas podem comunicar as outras e, assim, chegar aqueles desprovidos de tecnologias."

"O aplicativo é muito intuitivo, então não tive problema nenhum durante uso. A questão da mudança dos temas de claro para escuro ficou muito bom, principalmente para quem tem problemas de visão, mostrando que o desenvolvedor se preocupou com o usuário, melhorando sua experiência. Esse aplicativo tem tudo para dar certo. A ideia foi genial."

50% dos usuários que realizaram o teste afirmaram sentir motivação para reservar alimentos através do aplicativo e 50% mostraram-se em dúvida. 100% afirmaram estar dispostos a realizar doações utilizando a plataforma.

VI Conclusão

A fome crônica no Brasil atinge milhões de pessoas. Tal fome não tem como causa a escassez na produção de alimentos mas sim, a falta de recursos que abrange a população carente. Por outro lado, existe o desperdício desenfreado de alimentos, que causa uma série de prejuízos econômicos e ambientais.

Embora o desperdício não tenha relação direta com a fome, a redução no desperdício pode ser usada para diminuir os níveis de fome.

A Lei Nº 14.016, sancionada em 2020 reduz a barreira de entrada para doadores, dividindo a responsabilidade da retirada de alimentos entre doador e donatário.

Para que a doação de alimentos tenha relevância estatística é necessário que haja a implementação de ações que tenham alto potencial de difusão. Sob esse contexto foi criado um aplicativo que possibilita a comunicação entre doadores e donatários.

O aplicativo tem boa usabilidade e conta com recursos de acessibilidade. O sistema consome uma API que utiliza o conceito de sessão para lidar com as requisições, tornando-a sólida e confiável.

VII Trabalhos Futuros

Alguns novos recursos podem tornar-se convenientes após o aumento do alcance da solução desenvolvida. Entre eles destacam-se:

- Criação de aplicação WEB.
- Criação de painel administrativo na WEB.
- Possibilitar envio de imagens no chat.
- Criptografar mensagens do chat.
- Adicionar a possibilidade de realizar denúncias de um perfil.
- Assim como no aplicativo "Comida Invisível", adicionar calculadora de gastos que baseia-se nos resultados coletados na região.
- Adicionar inglês à lista de idiomas suportados.

Referências

[1] C. Augusto. *A dimensão da pobreza, da desnutrição e da fome no Brasil: implicações para as políticas públicas. Estudos e Pesquisas*. Brasília: Inae, no 53. 2003.

[2] R. P. C. Proença. *Alimentação e globalização: algumas reflexões*. Ciência e Cultura, Campinas, v. 62, n. 4, p.

[3] J. L. Silva da. A Fome no Brasil: do Período Colonial até 1940. Revista de Geografia, Recife, v. 23, n.3, p. 31-36, 2006.

[4] F. A. G. Vasconcelos. *Combate à fome no Brasil: uma análise histórica de Vargas a Lula*. Revista de Nutrição 18(Supl. 4):439-457, 2005

[5] Governo Brasileiro, Lei Nº 11.346. 2006.

[6] I. C. Neves das. *Fome no Brasil é drama diário de 19.3 milhões*. TCE MS. 2021.

[7] *POF - Pesquisa de Orçamentos Familiares.* IBGE. 2018.

[8] J. A. Neves, M. L. Machado, L. D. A. Oliveira, Y. M. F. Moreno, M. A. T. Medeiros and F. A. G. Vasconcelos de. *Unemployment, poverty, and hunger in Brazil in Covid-19 pandemic times.* Revista de Nutrição. 34. 2021

[9] M. E. Sipioni, M. R. L. Riquieri, J. P .M. Barbosa, D. B. Biscotto, T. D. Sarti, and M. A. C. Andrade. *Máscaras cobrem o rosto, a fome desmascara o resto: COVID-19 e o enfrentamento à fome no Brasil*. Pre Print Scielo. 2020.

[10] Organização das Nações Unidas para a Alimentação e a Agricultura. *Boosting food security in cities through better markets, reduced food waste.* 2015.

[11] J. S. Ribeiro. *Indicadores de desperdício de alimentos em restaurantes comerciais [Brasil.* Rosa dos Ventos, vol. 12, núm. 2, 2. 2020.

[12] C. C. Moraes. *Mitigação do desperdício de alimentos: práticas e causas na díade fornecedor-supermercado*. UFSCar. 2020.

[13] UnBTV. *Documentário: Quanto vale um terço?* . May. 13, 2016. Accessed on: Nov 1, 2021. [Video file]. Available: https://www.youtube.com/watch?v=1__rhYpDiqE

[14] G. P. Araújo de, C. E. Lourenço, C. M. L. Araújo de and A. Bastos. *Intercâmbio Brasil-União Europeia sobre desperdício de alimentos: relatório final*. 2019.

[15] *Desperdício – O vilão de todos nós.* Revista Ideias na Mesa 1º 2013.

[16] Food and Agriculture Organization of the United Nations. *Food wastage footprint. Impacts on natural resources*. Summary report. Rome. 2013.

[17] Governo brasileiro. Estudo destaca desperdício de alimentos no mundo. 2017

[18] C. M. Jesus de. *Quarto de Despejo – Diário de uma favelada.* São Paulo: Francisco Alves, 1963.

[19] San Diego Food System Alliance. Case Study: Prepared Food Donation. 2019.

[20] Governo Brasileiro, Lei Nº 14.016, 2020.

[21] S. Kafruni. *Desperdício de alimentos traz prejuízos também ao meio ambiente.* Correio Braziliense. 2018.

[22] Governo brasileiro. *Rede brasileira de bancos de alimentos.*

[23] ONG Banco de alimentos. Available: https://bancodealimentos.org.br/

[24] R. Dagnino. *A tecnologia social e seus desafios.* Fundação Banco do Brasil. Tecnologia social: uma estratégia para o desenvolvimento. Rio de Janeiro: 2004. p. 187-210.

[25] Fundação Banco do Brasil. *Tecnologia Social.* 2021.

[26] I. Rodrigues and J. C. Barbieri. *A emergência da Tecnologia Social: revisitando o movimento da tecnologia apropriada como estratégia*

de desenvolvimento sustentável. Rev. Adm. Pública, v. 42, n. 6, p. 1069-94, nov./dez. 2008.

[27] GIFE. *Conferência internacional aborda uso de tecnologias sociais.* 2004.

[28] S. GmbH. *Number of smartphone users from 2016 to 2021 (in millions).*

Available: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/

[29] L. Handley. *Nearly three quarters of the world will use just their smartphones to access the internet by 2025.* CNBC.. 2019.

[30] M. Mandel and E. Long. A Economia de Aplicativos no Brasil. Progressive Policy Institute. Washington-DC, 2017.

[31] CNDL Brasil. *Número de idosos que acessam internet cresce de 68% para 97%.* 2021.

[32] Apple. *Human Interface Guidelines.* Available: https://developer.apple.com/design/human-interface-guidelines/accessibility/overview/content/

[33] Google. *Criar apps mais acessíveis.* Available: https://developer.android.com/guide/topics/ui/accessibility/apps

[34] *[Estudo] Acessibilidade em apps com mais de 10 milhões de downloads no Brasil.* BigDataCorp.

[35] R. H. Hall and P. Hanna. T*he Impact of Web Page Text-background Colour Combinations on Readability, Retention, Aesthetics and Behavioural Intention.* . 2004.

[36] K. Kim, A. Erickson, A. Lambert, G. Bruder and G. Welch. *Effects of Dark Mode on Visual Fatigue and Acuity in Optical See-Through Head-Mounted Displays.* Symposium on Spacial User Interaction. 2019.

[37] A. D. Norman. *The Design of Everyday Things*. MIT Press, 2013.

[38] J. Fayzullaev. *Native-like Cross-Platform Mobile Development Multi-OS Engine & Kotlin Native vs Flutter.* South Eastern Finland University of Applied Sciences. 2018.

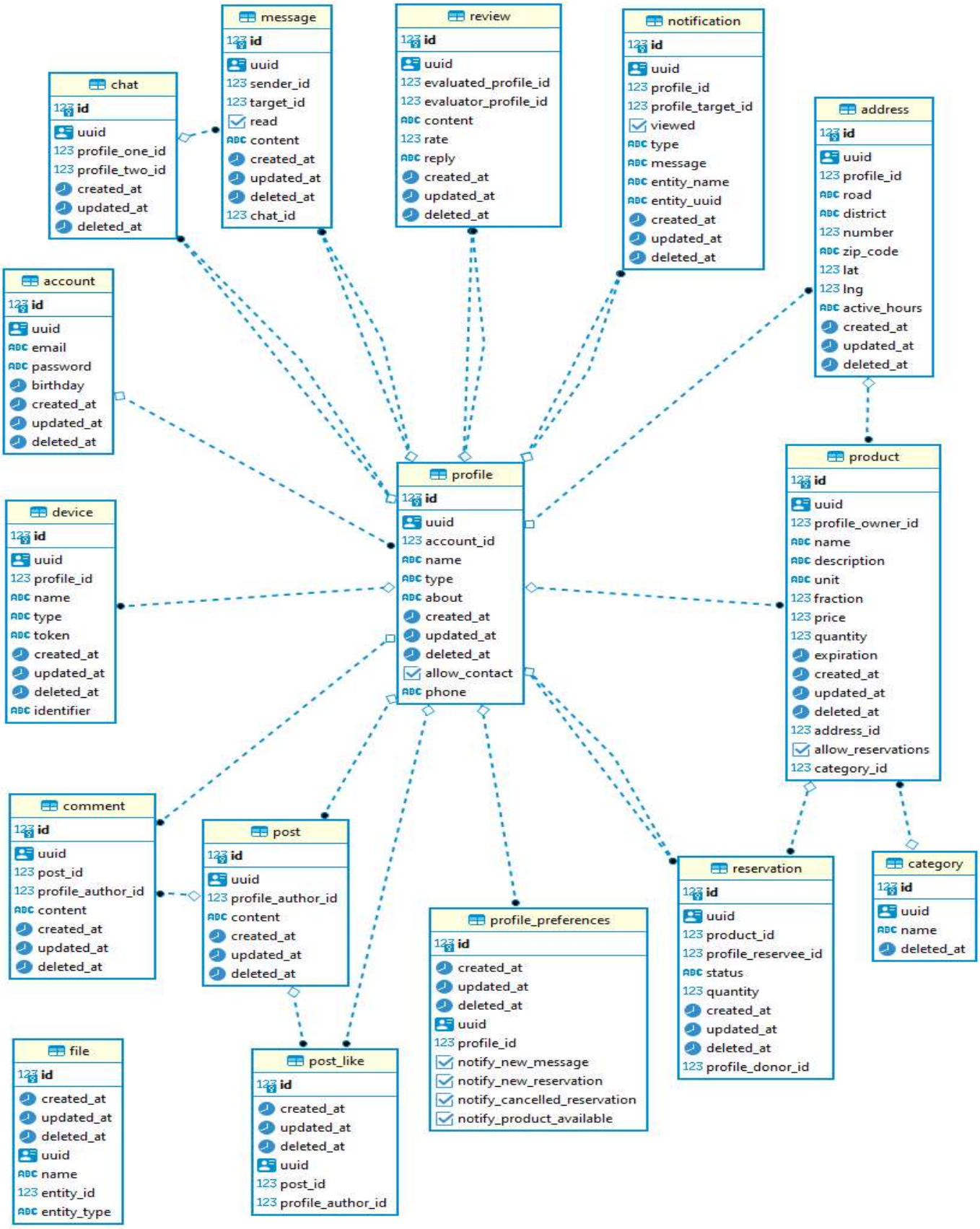
[39] G. Vivekananda and P. Reddy. *Critical analysis of Cross-layer approach.* ICGCIoT. 2015

[40] R. C. Martin. *Clean architecture: a craftsman's guide to software structure and design. Prentice Hall*. 2018.

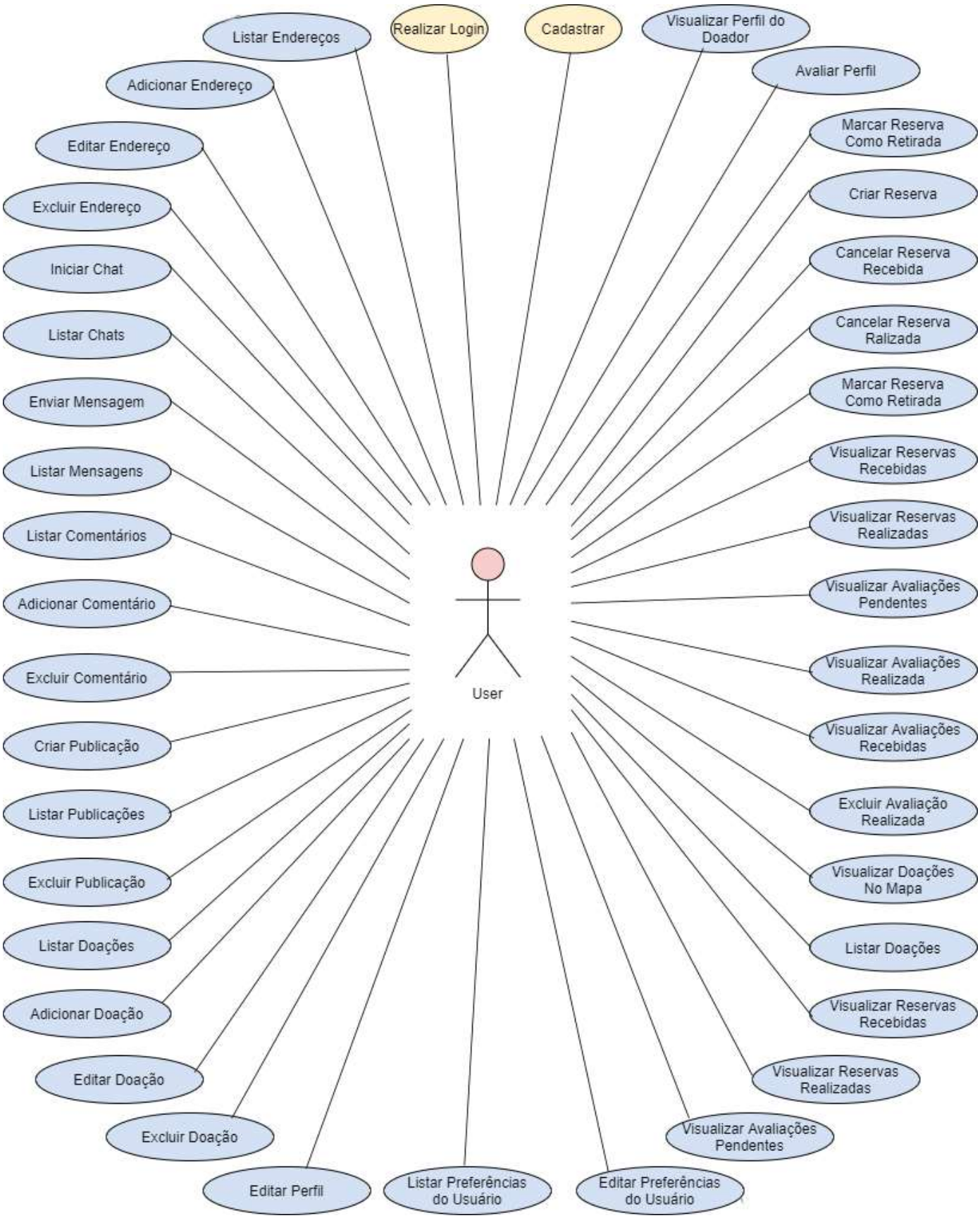
[41] M. Bayer, SQLAlchemy 1.4 Documentation. 2021

[42] M. Fowler et. all, Patterns of Enterprise Application Architecture, vol. 1. pp 184-199. 2002

[43] M. Bayer, The Architecture of Open Source Applications vol. 2.


message
id
uuid
sender_id
target_id
read
content
created_at
updated_at
deleted_at
chat_id
review
id
uuid
evaluated_profile_id
evaluator_profile_id
content
rate
reply
created_at
updated_at
deleted_at
notification
id
uuid
profile_id
profile_target_id
viewed
type
message
entity_name
entity_uuid
created_at
updated_at
deleted_at
chat
id
uuid
profile_one_id
profile_two_id
created_at
updated_at
deleted_at
address
id
uuid
profile_id
road
district
number
zip_code
lat
lng
active_hours
created_at
updated_at
deleted_at
account
id
uuid
email
password
birthday
created_at
updated_at
deleted_at
profile
id
uuid
account_id
name
type
about
created_at
updated_at
deleted_at
allow_contact
phone
product
id
uuid
profile_owner_id
name
description
unit
fraction
price
quantity
expiration
created_at
updated_at
deleted_at
address_id
allow_reservations
category_id
device
id
uuid
profile_id
name
type
token
created_at
updated_at
deleted_at
identifier
comment
id
uuid
post_id
profile_author_id
content
created_at
updated_at
deleted_at
post
id
uuid
profile_author_id
content
created_at
updated_at
deleted_at
reservation
id
uuid
product_id
profile_reservee_id
status
quantity
created_at
updated_at
deleted_at
profile_donor_id
category
id
uuid
name
deleted_at
profile_preferences
id
created_at
updated_at
deleted_at
uuid
profile_id
notify_new_message
notify_new_reservation
notify_cancelled_reservation
notify_product_available
file
id
created_at
updated_at
deleted_at
uuid
name
entity_id
entity_type
post_like
id
created_at
updated_at
deleted_at
uuid
post_id
profile_author_id

Anexo I


Listar Endereços
Realizar Login
Cadastrar
Visualizar Perfil do Doador
Avaliar Perfil
Adicionar Endereço
Editar Endereço
Marcar Reserva Como Retirada
Excluir Endereço
Criar Reserva
Iniciar Chat
Cancelar Reserva Recebida
Listar Chats
Cancelar Reserva Ralizada
Enviar Mensagem
Marcar Reserva Como Retirada
Listar Mensagens
Visualizar Reservas Recebidas
Listar Comentários
Visualizar Reservas Realizadas
Adicionar Comentário
Visualizar Avaliações Pendentes
Excluir Comentário
Visualizar Avaliações Realizada
User
Criar Publicação
Visualizar Avaliações Recebidas
Listar Publicações
Excluir Avaliação Realizada
Excluir Publicação
Visualizar Doações No Mapa
Listar Doações
Listar Doações
Adicionar Doação
Visualizar Reservas Recebidas
Editar Doação
Visualizar Reservas Realizadas
Excluir Doação
Visualizar Avaliações Pendentes
Editar Perfil
Listar Preferências do Usuário
Editar Preferências do Usuário

Anexo II

Avaliação do Aplicativo

Formulário para coletar as considerações dos usuários sobre o aplicativo.

Sinta-se à vontade para comentar sobre suas respostas, elogiar ou sugerir melhorias.

*Obrigatório

1. O quão importante você achou o MURAL DE PUBLICAÇÕES para o objetivo do app? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Importante	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Importante

2. O quão importante você achou o MAPA DE DOAÇÕES para o objetivo do app? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Importante	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Importante

3. O quão importante você achou a TELA DE BUSCA DE DOAÇÕES para o objetivo do app? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Importante	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Importante

4. O quão importante você achou as AVALIAÇÕES para o objetivo do app? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Importante	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Importante

Anexo III

5. O quão eficiente é a proposta de RETIRADA DE ALIMENTOS proposta? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Eficiente	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Eficiente

6. O que você achou da ACESSIBILIDADE do aplicativo? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Acessível	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Acessível

7. De modo geral, o quão útil você considera esse aplicativo? *

Marcar apenas uma oval.

	1	2	3	4	5	6	7	8	9	10	
Pouco Útil	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Muito Útil

8. Você se sentiu motivado (a) à doar alimentos através do aplicativo? *

Marcar apenas uma oval.

- ◯ Sim
- ◯ Não
- ◯ Talvez

9. Você se sentiu motivado (a) à reservar alimentos através do aplicativo? *

Marcar apenas uma oval.

- ◯ Sim
- ◯ Não
- ◯ Talvez

Anexo IV