

# **PET SOLIDÁRIO: Solução Tecnológica para a Alimentação de Animais de Rua**

**Keiza Analizi da Silva**

kas2@discente.ifpe.edu.br

**Vagner Henrique Silva de França**

vhsf@@discente.ifpe.edu.br

**Orientador: Anderson Luiz Souza Moreira**

anderson.moreira@recife.ifpe.edu.br

**Coorientador: Marco Antonio de Oliveira Domingues**

marcodomingues@recife.ifpe.edu.br

---

## **RESUMO**

Diante do crescente número de animais em situação de rua, vulneráveis à falta de alimentação e hidratação, o projeto "Pet Solidário" busca promover o bem-estar e a saúde desses animais por meio do desenvolvimento de bebedouros e comedouros inteligentes. Esses dispositivos, baseados na tecnologia Arduino, contam com dispensers automatizados e monitoramento contínuo dos níveis de água e ração. Para complementar, foi criado um aplicativo mobile que, em caso de escassez, envia notificações em tempo real, permitindo o acompanhamento remoto e a gestão eficiente dos dispensers. Os resultados obtidos com o Pet Solidário demonstraram que esses dispositivos não apenas beneficiarão inúmeros animais de rua, mas também otimizarão o fornecimento de água e ração, com o apoio de ONGs e pessoas sensibilizadas com a causa, graças à gestão centralizada oferecida pelo aplicativo.

Palavras-chave: Arduino; alimentador automático; mobile; assistividade; animais

## **ABSTRACT**

Given the growing number of stray animals vulnerable to a lack of food and hydration, the "Pet Solidário" project aims to promote the well-being and health of these animals through the development of intelligent water and food dispensers. These devices, based on Arduino technology, feature automated dispensers and continuous monitoring of water and food levels. To complement this, a mobile application was created that, in case of scarcity, sends real-time notifications, allowing remote monitoring and efficient management of the dispensers. The results obtained with Pet Solidário have shown that these devices will not only benefit countless stray animals but also optimize the

supply of water and food, with the support of NGOs and people sensitive to the cause, through the centralized management offered by the application.

Keywords: Arduino; automatic feeder; mobile; assistivity; animals

---

## 1 INTRODUÇÃO

De acordo com uma pesquisa realizada pela (Mars, 2023) em parceria com o State of Pet Homelessness Project em mais de 15 países, cerca de 35% dos cães e gatos encontram-se em situação de rua ou em abrigos à espera de lar. Isso representa aproximadamente 362 milhões de animais. Essa realidade destaca a importância de desenvolver soluções sustentáveis e eficientes para atender essas populações, especialmente no que diz respeito à nutrição, uma das principais necessidades desses animais desamparados.

No Brasil, estima-se que haja cerca de 121,3 milhões de cães e gatos, dos quais 82,1 milhões são cães e 39,2 milhões são gatos. (MVC, 2024). Em relação ao abandono, o estudo revela que aproximadamente 30,2 milhões de cães e gatos são abandonados no país, o que representa 25% do total de animais, colocando o Brasil abaixo da média observada em comparação com os dados de 20 outros países analisados. Dentro desse contexto, uma pequena porcentagem dos animais abandonados encontra-se sob cuidados especiais em abrigos: cerca de 0,07% dos 10 milhões de gatos e aproximadamente 0,88% dos 20,2 milhões de cães.

Considerando as inúmeras vantagens proporcionadas pelo uso da Internet das Coisas (IoT) e a possibilidade de desenvolver alternativas para uma alimentação e hidratação mais eficientes e contínuas, este artigo tem como objetivo apresentar e detalhar o projeto Pet Solidário. Este projeto desenvolveu bebedouros e comedouros automatizados, desenvolvidos com a tecnologia Arduino. Esses dispositivos são equipados com dispensers automatizados e sistemas de monitoramento contínuo dos níveis de água e ração, permitindo uma gestão eficiente dos recursos alimentares e hídricos destinados aos animais de rua. A integração com um aplicativo móvel centraliza todas as informações em tempo real, permitindo o acompanhamento simultâneo de múltiplos pontos de alimentação e hidratação em diferentes regiões. O aplicativo também envia alertas sobre a escassez de recursos, facilitando a logística de reabastecimento e limpeza dos dispositivos e garantindo a disponibilidade contínua de recursos essenciais para a sobrevivência dos animais.

Este artigo está organizado da seguinte forma: na Seção 2, serão apresentadas as fundações teóricas. Na Seção 3, apresentamos os trabalhos relacionados a esse projeto. A Seção 4 detalha a metodologia adotada e oferece uma visão geral dos componentes e implementações necessárias ao protótipo. Na Seção 5, são discutidos os testes e resultados, incluindo os testes e resultados da modelagem do hardware e software. Por fim, a Seção 6 apresenta a conclusão, que resume os resultados e sugere direções futuras.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, são apresentados os conceitos, teorias e modelos que fundamentam as discussões deste trabalho. Além disso, são abordadas as tecnologias, frameworks e padrões de desenvolvimento adotados ao longo do projeto.

### 2.1 Internet das Coisas (IoT)

A Internet das Coisas (IoT – Internet of Things), refere-se a uma rede de dispositivos físicos, veículos, eletrodomésticos e outros objetos físicos que são incorporados com sensores, software e conectividade de rede, permitindo coletar e compartilhar dados em tempo real (IBM, 2024). Essa tecnologia tem revolucionado diversas áreas, desde a automação industrial até aplicações voltadas para o bem-estar social e ambiental, como sistemas inteligentes para monitoramento de recursos e otimização de processos.

De acordo com a (Solutions, 2024), estima-se que, até 2025, mais de 75 bilhões de dispositivos estarão conectados em todo o mundo. Esse crescimento é impulsionado pela crescente demanda por automação e eficiência em diversos setores, como saúde, transporte, manufatura, agricultura e meio ambiente.

A IoT é baseada em três camadas principais: Camadas de Percepção, Rede e Aplicação, que operam em conjunto com seus principais componentes: Sensores e Atuadores, Conectividade, Processamento de dados e interface de usuário.

### 2.2 Armazenamento na Nuvem (Cloud Storage)

O armazenamento em nuvem (Cloud Storage) é um modelo de armazenamento de dados no qual as informações digitais são mantidas em servidores remotos, acessíveis via Internet (Cloud, 2024). Essa tecnologia tornou-se essencial para empresas que buscam maior eficiência, escalabilidade e flexibilidade em suas operações de TI. Segundo dados da (Gartner, 2023), estima-se que até 2027 mais de 70% das empresas utilizarão plataformas em nuvem para impulsionar suas iniciativas de negócios, um aumento significativo em relação aos menos de 15% registrados em 2023.

Com a crescente adoção de soluções em nuvem, tornou-se possível armazenar e sincronizar informações em tempo real entre diferentes dispositivos e plataformas. Um exemplo dessa tecnologia são os bancos de dados NoSQL, como o Cloud Firestore, que permite armazenar, sincronizar e consultar dados facilmente.

Os bancos de dados NoSQL são conhecidos por sua capacidade de processar grandes volumes de dados não estruturados e em constante mudança, ao contrário dos bancos relacionais tradicionais que organizam dados em tabelas com linhas e colunas. Essas tecnologias existem desde a década de 1960, mas se tornaram populares recentemente devido à necessidade de lidar com o grande volume de dados gerados pela nuvem, dispositivos móveis, redes sociais e Big Data (Azure, 2025). Os bancos de dados NoSQL permitem que os desenvolvedores criem sistemas capazes de armazenar e disponibilizar rapidamente esses dados para análise e pesquisa.

O Cloud Firestore é um banco de dados NoSQL baseado em documentos, que

organiza as informações de forma estruturada por meio de coleções e documentos, facilitando o armazenamento e a recuperação de dados (Firebase, 2023). Além disso, ele permite integração com o Firebase Authentication, proporcionando um sistema de autenticação de usuários mais simples do que os modelos tradicionais.

### 3 TRABALHOS RELACIONADOS

Foram pesquisados trabalhos no mesmo contexto, seguindo o mesmo intuito de automatização voltado à alimentação de pets com dispenser.

O Smart Dog Feeder (Vania; Karyono; Nugroho, 2016) é um alimentador inteligente para cães domésticos, operado via comunicação sem fio e protocolo MQTT, com controle remoto por um aplicativo Android. Utilizando Arduino UNO e NodeMCU, permite ajustar automaticamente porções e horários de alimentação, garantindo praticidade aos tutores. Já o Pet Solidário foca em animais de rua, adotando um sistema dinâmico que ajusta a distribuição de alimentos conforme a necessidade, sem horários fixos. Enquanto o Smart Dog Feeder usa sensores e programações para alimentar os pets sob supervisão humana, o Pet Solidário monitora o consumo em tempo real e opera de forma autônoma. Além disso, o Smart Dog Feeder segue um cronograma de reabastecimento fixo, enquanto o Pet Solidário repõe os alimentos conforme a demanda, tornando-o mais flexível e adaptável. Por atender a um maior número de animais, o Pet Solidário exige uma gestão eficiente dos recursos, proporcionando uma solução escalável e inovadora para o bem-estar animal.

O trabalho de (Cunha et al., 2020) propõe um alimentador automático de pets que distribui ração e água em horários fixos, garantindo a alimentação dos animais mesmo na ausência dos donos. Baseado em um microcontrolador, o sistema é simples e eficiente, ideal para tutores que buscam automatização. Já o Pet Solidário, voltado para animais de rua, adota uma abordagem dinâmica, utilizando sensores ultrassônicos para detectar a presença dos animais e reabastecer água e ração conforme a necessidade, garantindo um suprimento contínuo e adaptável. Outra diferença está nos sensores de nível de água: o sistema de Cunha et al. usa um sensor flutuante com reed switch, que detecta variações pela posição do flutuador magnético, enquanto o Pet Solidário emprega o sensor HR43, que monitora umidade e aciona dispositivos externos para controle automatizado da água.

O KL Pet Care, descrito por (Morais, 2021), utiliza o aplicativo Blynk para monitoramento remoto da alimentação e dos níveis de água dos pets, facilitando o controle para tutores ausentes. O sistema emprega ESP8266 e tecnologias IoT, proporcionando uma solução prática e flexível para o acompanhamento à distância. Seu dispensador de água funciona com um reservatório invertido, onde a água flui por gravidade, seguindo o princípio de Stevin, regulando a pressão conforme a profundidade. Já o Pet Solidário utiliza um aplicativo proprietário, desenvolvido especificamente para monitoramento e gestão de recursos para animais de rua. Ele opera com ESP32, oferecendo maior capacidade de processamento e conectividade. Diferente do KL Pet Care, que depende da gravidade para dispensar água, o Pet Solidário utiliza uma válvula solenóide, permitindo um controle mais preciso do fluxo de água até o pote.

O projeto de (Silva Lombas et al., 2020) desenvolve um alimentador automático

para pets com comunicação via redes e controle remoto por aplicativo web, permitindo o agendamento de refeições e o ajuste das porções de ração, oferecendo mais praticidade aos tutores. Apesar de ambos os projetos utilizarem o mesmo modelo de microcontrolador Arduino ESP32 e empregarem a tecnologia para o bem-estar animal, os projetos possuem públicos e objetivos distintos. Enquanto o Pet Solidário foca no atendimento a animais de rua, utilizando bebedouros e comedouros inteligentes que monitoram e distribuem água e ração com base na demanda, além de enviar alertas em tempo real quando há escassez desses recursos, garantindo um impacto social direto, o projeto de (Silva Lombas et al., 2020) busca atender o público de animais domésticos, nos quais a distribuição desses mesmos recursos ocorre de maneira programada e dentro de ambiente controlado. Outra diferença está no mecanismo de dispensação de ração: o sistema de Lombas et al. utiliza uma rosca transportadora de alumínio fundido, acionada por um motor reaproveitado de impressora e fixado com resina plástica a um redutor de micro-ondas. Já o Pet Solidário adota um funil redutor com um dispensador de madeira acoplado a um Servo Motor TowerPro SG5010.

## 4 METODOLOGIA

Nesta seção, serão abordados os principais aspectos do desenvolvimento de um sistema automatizado projetado para melhorar as condições de vida dos animais em situação de rua, atendendo às suas necessidades básicas, como alimentação e hidratação. Inicialmente, será fornecida uma visão geral dos componentes utilizados na construção dos dispensers e o protótipo do sistema, abordando sua implementação física. Em seguida, será detalhado o processo de desenvolvimento do aplicativo móvel, responsável pelo monitoramento e gerenciamento remoto do sistema. Por fim, será descrita a integração entre os sistemas, destacando como a tecnologia se conecta para oferecer uma solução eficiente e funcional.

### 4.1 Desenvolvimento dos Dispensers

Para a criação de um sistema eficiente, foi realizada uma pesquisa detalhada para identificar os equipamentos mais adequados às principais necessidades do desenvolvimento do dispositivo. Com base nessa análise, os seguintes componentes foram selecionados para a construção do comedouro e bebedouro automatizados:

- Microcontrolador (Figura 1): O ESP32 será um dos componentes centrais do projeto, interpretando sinais dos sensores e controlando sistemas como energização, refrigeração e alertas. Equipado com um processador dual-core Xtensa LX6 de 32 bits (até 240 MHz), oferece alto desempenho para IoT, conectando-se via Wi-Fi, Bluetooth e portas GPIO. Possui regulador de 3,3V, interface Serial-USB, conector micro-USB, LEDs indicadores e botões de RESET e BOOT, facilitando programação e manutenção. Sua robustez e flexibilidade garantem comunicação eficiente e controle preciso, superando plataformas como o Arduino em desempenho, especialmente em redes. (Systems, 2024).

Figura 1 – Microcontrolador ESP32.



Fonte: Arducore (2025)

- Modulo de relés (Figura 2): O Módulo *relé* 12V 1 Canal é essencial para controlar dispositivos de alta potência, como lâmpadas e motores, com um simples pino de controle. Sua principal vantagem é o isolamento entre o microcontrolador e a carga, garantindo segurança contra interferências e curtos-circuitos. Opera com 12V, suporta cargas de até 250 VAC ou 30 VDC e corrente máxima de 10A. Possui LED indicador, pinos de alimentação e controle simplificados, além de borne de saída com parafusos para fácil conexão. Compatível com Arduino, é ideal para automação, IoT e controle remoto de dispositivos. (Tongling, 2024)

Figura 2 – Módulo Relé.



Fonte: Tecnotronics (2025)

- Sensor Ultrassônico (Figura 3): O Sensor Ultrassônico HC-SR04 é utilizado para medir distâncias via ondas ultrassônicas, emite um sinal pelo pino Trig e recebe o retorno pelo pino Echo, calculando o tempo de viagem das ondas. Alimentado com 5V, é compatível com Arduino e outras plataformas. Mede distâncias de 2 cm a 4 m, com precisão de 3 mm, sendo ideal para robótica, automação residencial



torque de 3,2 kgf·cm a 4,8V. Este servo motor é popular em projetos com Arduino, Raspberry Pi e outras plataformas de prototipagem.(RoboCore, 2024b)

Figura 5 – Servo Motor TowerPro-SG5010.



Fonte: MarketHero (2025)

- Sensor de nível de água (Figura 6): O Sensor de Chuva e Nível de Água HR43 é usado em automação e monitoramento ambiental, ideal para irrigação e drenagem. Detecta chuva ou água, acionando dispositivos externos via saída digital. Opera com 5V e é compatível com Arduino. Mede o nível de água por resistência elétrica, sendo útil em monitoramento climático, automação residencial e IoT, permitindo controle preciso de dispositivos conforme as condições ambientais.(Saravati, 2024)

Figura 6 – Sensor de Nível da Água.



Fonte: Mult Comercial (2025)

- Sensor de Peso 50Kg - Célula de Carga (Figura 7): Sensor de Peso 50Kg - Utilizado para medir força ou peso, convertendo-os em sinal elétrico. Possui um strain gauge que altera sua resistência conforme a carga aplicada, garantindo alta precisão. Ideal para balanças digitais, pesagem industrial e automação IoT. Opera com 5V, sendo compatível com Arduino e outras plataformas. Suporta até 50kg e, para maior precisão, é geralmente utilizado com um amplificador de sinal, como o HX711.(Curtocircuito, 2024b)

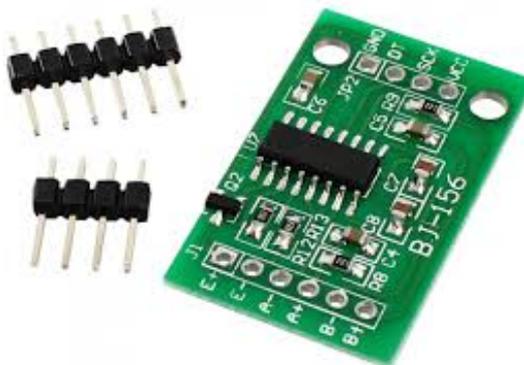
Figura 7 – Célula de Carga 50kg.



Fonte: RoboCore (2025)

- Módulo Conversor Célula de Carga - HX711 (Figura 8): O Módulo Conversor Célula de Carga - HX711 é um amplificador de sinal de alta precisão para células de carga, convertendo sinais analógicos em digitais para microcontroladores como Arduino. Opera com 5V, possui resistor de referência interno e entradas diferenciais de alta precisão. Ideal para balanças digitais, sistemas de pesagem e automação IoT, garantindo medições precisas em projetos industriais e de monitoramento de carga.(Curtocircuito, 2024a)

Figura 8 – Módulo Conversor HX-711.



Fonte: Usinainfo (2025)

#### 4.1.1 Protótipo

O desenvolvimento do protótipo considerou animais de pequeno e médio porte. A estrutura foi montada utilizando blocos de madeira e envelopada para aumentar ainda mais a sua resistência e durabilidade.

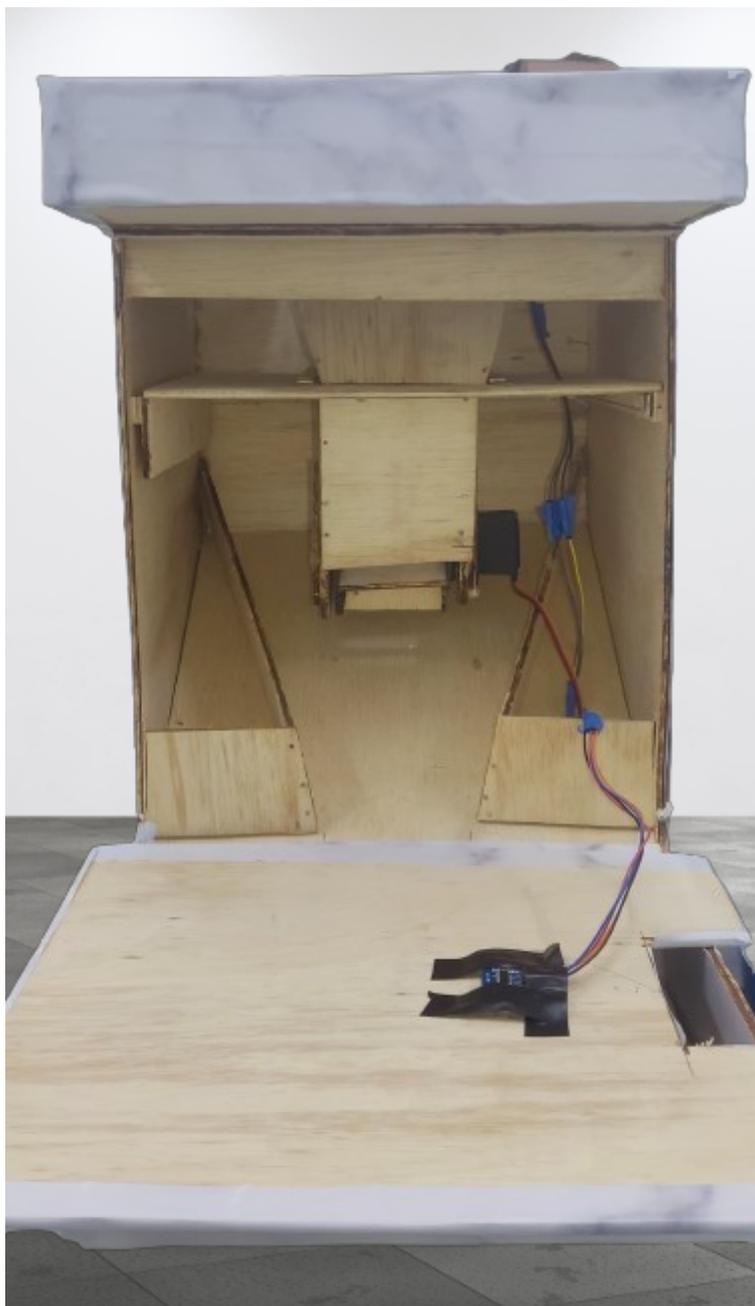
O dispenser de ração (Figura 9) tem capacidade para aproximadamente 10 porções. Nele, há um reservatório em formato de funil, cujo volume de alimento disponível é monitorado por um sensor ultrassônico. Esse formato foi projetado para utilizar a gravidade como meio de direcionar e controlar o fluxo de insumos até o dispensador de madeira, que será acionado pelo Servo Motor responsável pela liberação da ração para o pet. O alimento liberado cai em uma pequena esteira, posicionada de maneira angular em formato de funil, abaixo do mecanismo de controle da ração, garantindo que o alimento seja direcionado corretamente para dentro da tigela.

Figura 9 – Dispenser de ração.



Fonte: Próprio autor (2025)

Figura 10 – Estrutura do dispenser de ração.



Fonte: Próprio autor (2025)

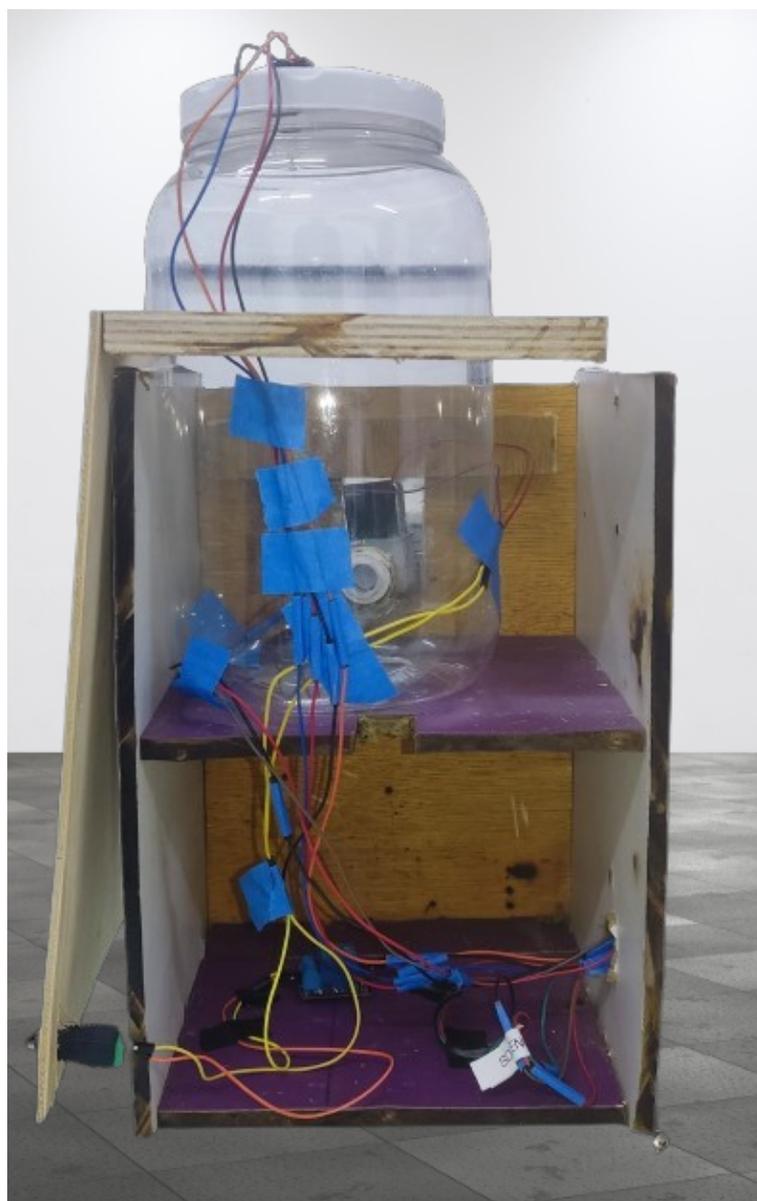
Para o dispenser de água (Figura 11) foi utilizada uma suqueira de plástico com capacidade para 7 litros, alocada em um reservatório de material resistente para aumentar sua durabilidade e proteção. Um sensor ultrassônico monitora constantemente o volume de água disponível no dispenser. O nível de água no pote é detectado por um sensor de nível, que, ao identificar um valor abaixo do ideal, aciona o módulo relé. Esse, por sua vez, libera a válvula solenoide, permitindo o fluxo de água para reabastecer o pote automaticamente.

Figura 11 – Dispenser de água.



Fonte: Próprio autor (2025)

Figura 12 – Estrutura do dispenser de água.



Fonte: Próprio autor (2025)

## 4.2 Desenvolvimento do Aplicativo Móvel

O aplicativo móvel do sistema prototipado foi desenvolvido com a linguagem de programação Kotlin, conhecida por sua produtividade, segurança e total interoperabilidade com Java. O objetivo principal do aplicativo é facilitar a interação dos apoiadores da comunidade, oferecendo uma interface intuitiva para monitoramento em tempo real dos comedouros e bebedouros instalados.

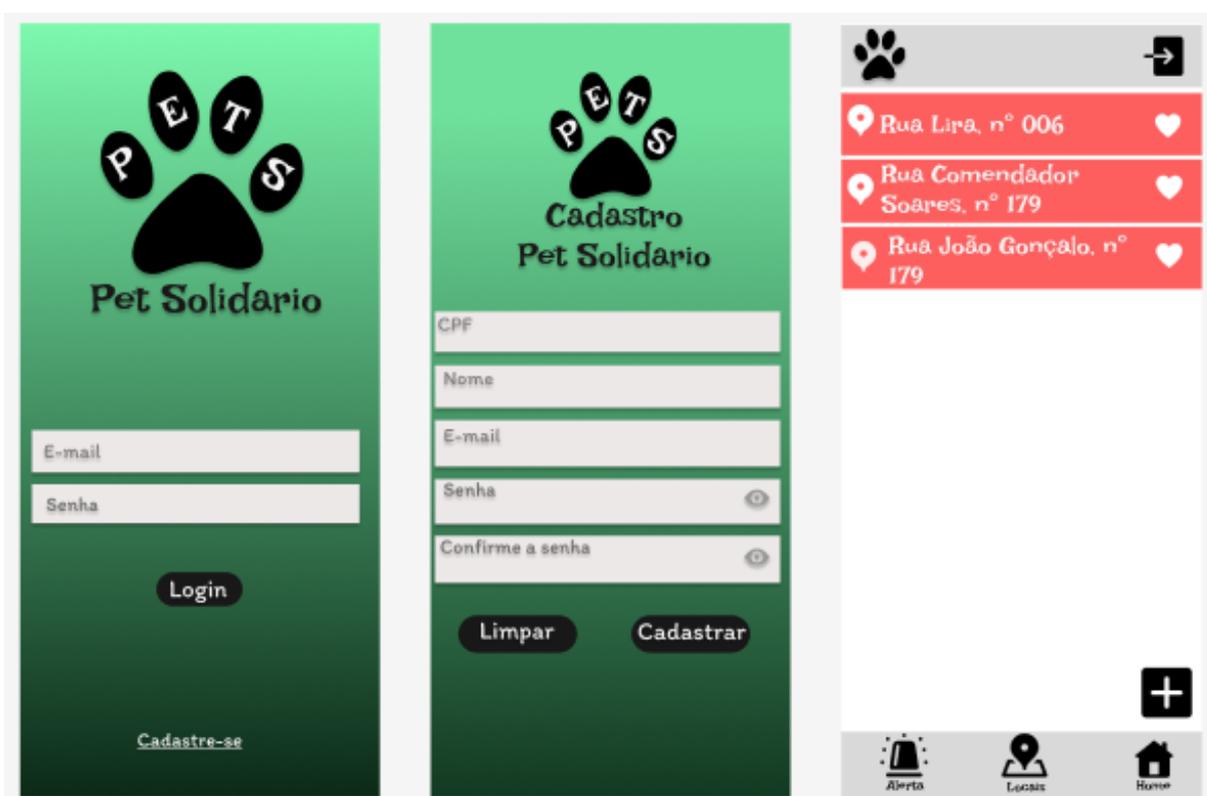
Para o armazenamento de dados, utilizou-se o Cloud Firestore, um banco de dados NoSQL hospedado na nuvem, que garante a sincronização automática dos dados entre dispositivos. Essa funcionalidade possibilita a visualização em tempo real do status dos potes, como os níveis de água e ração. Além disso, o aplicativo foi integrado à API do

Google Maps, permitindo a exibição da localização geográfica dos dispositivos. Essa funcionalidade facilita a gestão e aumenta a visibilidade dos pontos de alimentação e hidratação, auxiliando os apoiadores na logística de reabastecimento e cuidado com os animais de rua.

Além disso, para garantir a organização e o controle de versão do código, foi utilizado o GitHub, uma plataforma colaborativa de repositório de código-fonte baseada em Git.

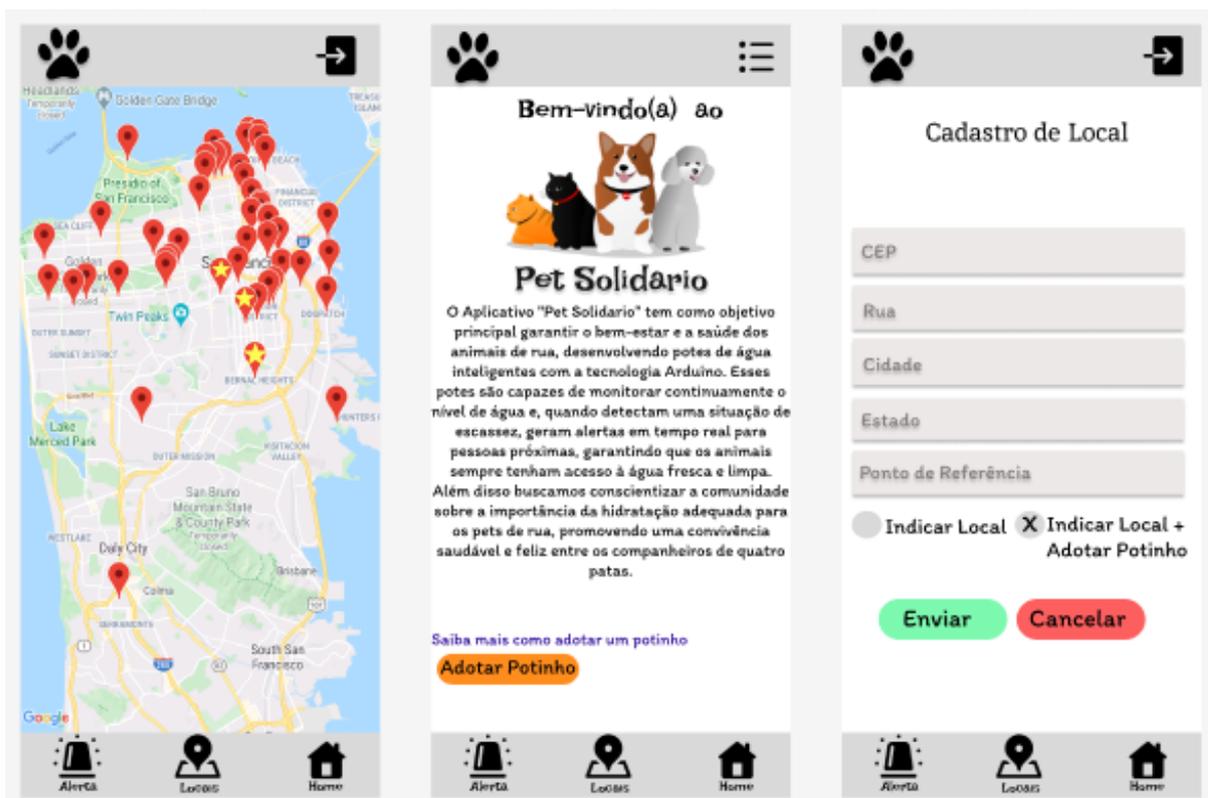
Como primeiro passo, foi realizada a criação de um protótipo para o desenvolvimento da aplicação. Para essa etapa, utilizou-se o Figma, uma ferramenta de prototipagem gráfica que permite a criação de wireframes e o planejamento da interface do usuário. A Figura 13 e 14 ilustra o protótipo desenvolvido para as telas do aplicativo mobile, servindo como base para a estrutura e o projeto do sistema.

Figura 13 – Telas do aplicativo protótipo.



Fonte: Próprio autor (2025)

Figura 14 – Telas do aplicativo protótipo.



Fonte: Próprio autor (2025)

### 4.3 Integração dos Sistemas

Por meio da integração entre o Cloud Firestore e o ESP32, é possível conectar os dispositivos ao aplicativo mobile, permitindo a atualização em tempo real do status dos potes. Para isso, foi utilizada a biblioteca FirebaseESP32, desenvolvida por Mobizt e disponível para download no GitHub.

Figura 15 – Arquitetura da Integração.

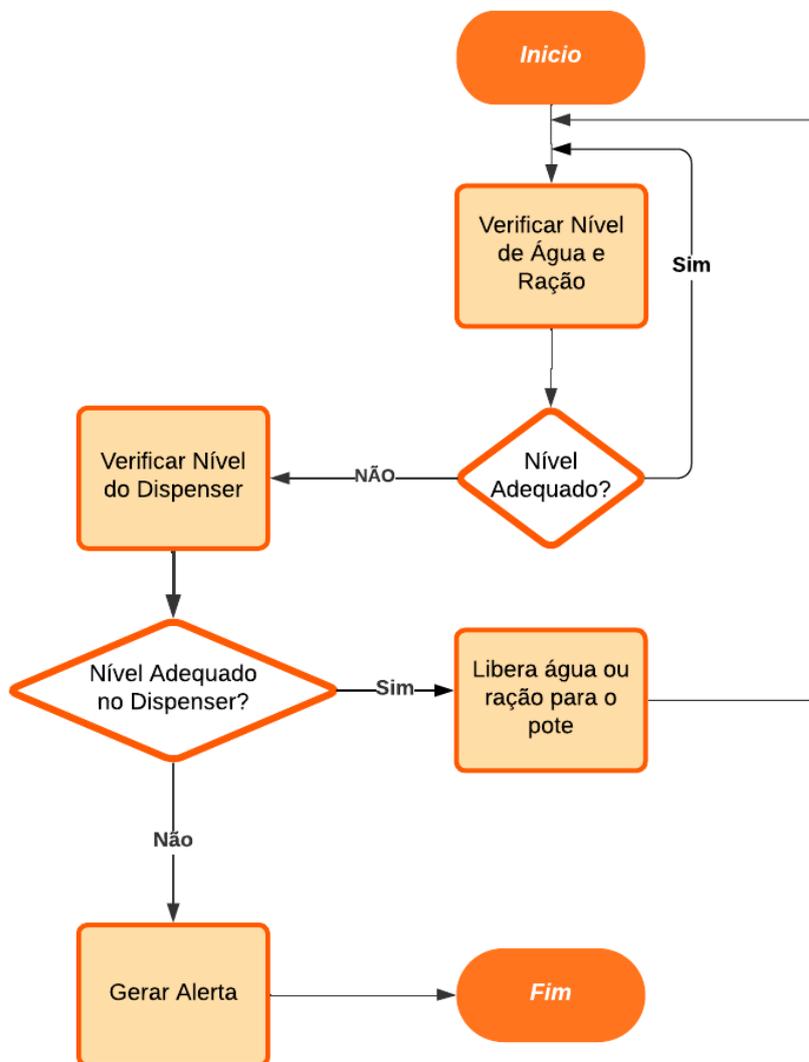


Fonte: Próprio autor (2025)

A Figura 16 apresenta o fluxo de monitoramento contínuo dos níveis de água e ração nos potes por meio de sensores. Quando um nível crítico é detectado, o sistema verifica a disponibilidade no dispenser e, se houver suprimento suficiente, realiza a reposição automaticamente. Caso contrário, o status dos potes é atualizado no Firebase

para "FALSE", e uma notificação é enviada ao aplicativo, alertando os responsáveis sobre a necessidade de reposição manual.

Figura 16 – Representação das etapas.



Fonte: Próprio autor (2025)

## 5 TESTES E RESULTADOS

Nesta seção, são apresentados os testes realizados para validar a funcionalidade do sistema *Pet Solidário* como um todo, com foco na integração dos componentes de hardware e software. A avaliação foi realizada para garantir que todos os dispositivos (sensores, atuadores, microcontrolador e aplicativo) trabalhassem de forma coordenada, executando a automação proposta com eficiência. Os testes envolveram a interação entre o microcontrolador ESP32, os sensores, válvulas solenoides, o módulo de relé e o aplicativo móvel, que juntos formam o sistema automatizado de monitoramento e controle.

## 5.1 Objetivos dos Testes Funcionais

Os testes funcionais tiveram como objetivo verificar os seguintes aspectos do sistema:

- **Integração entre Hardware e Software:** Como os componentes físicos (sensores, atuadores) se comunicam com o microcontrolador e como essas informações são processadas pelo aplicativo.
- **Desempenho na Automação de Alimentação e Hidratação:** Avaliar a capacidade do sistema em fornecer ração e água automaticamente, com base nas leituras dos sensores de nível.
- **Respostas e Alertas em Tempo Real:** Testar a emissão de alertas via o aplicativo móvel quando os níveis de água ou ração atingem um limite crítico.
- **Precisão na Leitura de Sensores:** Verificar a precisão dos sensores de peso, nível de água e distância para garantir que as informações enviadas ao aplicativo sejam confiáveis.

## 5.2 Descrição dos Testes Funcionais

Para garantir que o sistema funcionasse como esperado, os seguintes testes foram realizados:

- **Teste 1: Monitoramento e Controle de Ração**

O sensor de peso, conectado à célula de carga, foi instalado no comedouro (pote) para monitorar a quantidade de ração disponível. O módulo HX711, integrado ao ESP32, foi testado para garantir que as medições de peso fossem precisas e enviadas corretamente ao aplicativo. Quando a quantidade de ração alcançava um valor crítico, o sistema acionava automaticamente o dispenser de ração para reabastecer o comedouro.

  - **Calibração da Célula de Carga:**

Para calibrar as células de carga em conjunto com o amplificador HX711, foi utilizada a biblioteca HX711.h na IDE Arduino. A calibração foi realizada a partir da determinação do fator de calibração (calibration factor), utilizando um peso de referência conhecido. O código abaixo disponibilizado pelo autor (`celula_carga_Hx711`) foi empregado para encontrar o valor ideal de calibração, garantindo medições precisas. Durante a execução, o sistema permitiu ajustes no fator de calibração por meio da interface serial, garantindo a correta aferição do sensor de peso. O valor obtido para o fator de calibração foi -38789.00.

Figura 17 – Código de calibração part 1

```

1  #include "HX711.h" /*Biblioteca do HX711.h */
2
3  HX711 balanca(3, 2); /* SCK= pino 2 e DT= pino 3 */
4
5  float calibration_factor = 48011.00; /* Fator de calibração para ajuste da célula */
6  float peso; /* variável peso */
7
8  void setup() { /* rotina de configurações */
9      Serial.begin(9600); /* Baud rate da comunicação */
10     Serial.println("Remova todos os pesos da balança"); /* Printa "Remova todos os pesos da balança" na COM */
11     delay(1000); /* atraso de 1000ms = 1s */
12     Serial.println("Após estabilização das leituras, coloque o peso conhecido na balança"); /* Printa "Após estabilização das leituras, coloque o peso conhecido na balança" na COM */
13     delay(1000); /* atraso de 1000ms = 1s */
14     Serial.println("Pressione + para incrementar o fator de calibração"); /* Printa "Pressione + para incrementar o fator de calibração" na COM */
15     Serial.println("Pressione - para decrementar o fator de calibração"); /* Printa "Pressione - para decrementar o fator de calibração" na COM */
16     delay(1000); /* atraso de 1000ms = 1s */
17     balanca.set_scale(); /* seta escala */
18     balanca.tare(); /* escala da tara */
19
20     long zero_factor = balanca.read_average(); /* Realizando a leitura */
21 }
22
23 void loop() { /* chama função de loop */
24
25     balanca.set_scale(calibration_factor); /* a balança está em função do fator de calibração */
26
27     Serial.print("Peso: "); /* Printa "Peso:" na COM */
28     peso = balanca.get_units(), 10; /* imprime peso */

```

Fonte: Curto Circuito (2025)

Figura 18 – Código de calibração part 2

```

28     peso = balanca.get_units(), 10; /* imprime peso */
29     if (peso < 0) /* se a unidade for menor que 0 será considerado 0 */
30     {
31         peso = 0.00; /* Para o caso do peso ser negativo, o valor apresentado será 0 */
32     }
33     Serial.print(peso); /* Printa o peso na serial */
34     Serial.print(" kg"); /* Printa "kg" na serial */
35     Serial.print(" Fator de calibração: "); /* Printa "Fator de calibração:" na serial */
36     Serial.print(calibration_factor); /* Printa o fator de calibração na serial */
37     Serial.println(); /* Pula linha no serial */
38     delay(500); /* atraso de 500ms = 0.5s */
39
40     if(Serial.available()) /* caso sejam inseridos caracteres no serial */
41     {
42         char temp = Serial.read();
43         if(temp == '+') /* Se o + for pressionado */
44             calibration_factor += 1; /* incrementa 1 no fator de calibração */
45         else if(temp == '-') /* Caso o - seja pressionado */
46             calibration_factor -= 1; /* Decrementa 1 do fator de calibração */
47     }
48 }

```

Fonte: Curto Circuito (2025)

### • Teste 2: Monitoramento e Controle de Água

O módulo de relé foi utilizado no bebedouro para controlar o fornecimento de água. O sensor de nível de água HR43 foi instalado no bebedouro para medir os níveis de água. Quando o nível caía abaixo da capacidade ideal, o ESP32 acionava a válvula solenoide, liberando mais água até o nível ideal ser atingido. O aplicativo também recebia um alerta sobre a necessidade de reabastecimento.

### • Teste 3: Resposta e Alertas em Tempo Real

Os sensores de nível (água e ração) foram testados em conjunto com o aplicativo Pet's. Quando os sensores atingiam os níveis críticos, o aplicativo enviava notificações para os usuários, informando sobre a necessidade de reabastecimento.

### • Teste 4: Conectividade e Sincronização com o Aplicativo

A sincronização entre os dispositivos de hardware e o aplicativo foi verificada. A

atualização dos dados de consumo em tempo real no aplicativo, como o status de cada bebedouro e comedouro, foi testada para garantir que a interface exibisse informações precisas e atualizadas sem atrasos significativos.

### 5.3 Resultados Obtidos

Os testes realizados mostraram que o sistema *Pet Solidário* foi eficaz na execução das funções automáticas de controle e monitoramento, com a integração dos componentes funcionando conforme o esperado. Abaixo estão os resultados dos testes:

Teste	Componente	Descrição	Resultado
Teste 1	Sensor de Peso e Célula de Carga	Monitoramento da quantidade de ração no comedouro	Precisão baixa na medição. Resultado contrário ao esperado
Teste 2	Módulo Relé (Bebedouro)	Controle do fornecimento de água nos bebedouros	Alta precisão no controle
Teste 3	Integração Sensor-ESP32-Aplicativo	Envio de alertas via aplicativo	Alertas enviados em tempo real, sem falhas
Teste 4	Conectividade e Sincronização com o App	Comunicação entre hardware e aplicativo	Sincronização de dados em tempo real

Tabela 1 – Desempenho Funcional dos Componentes

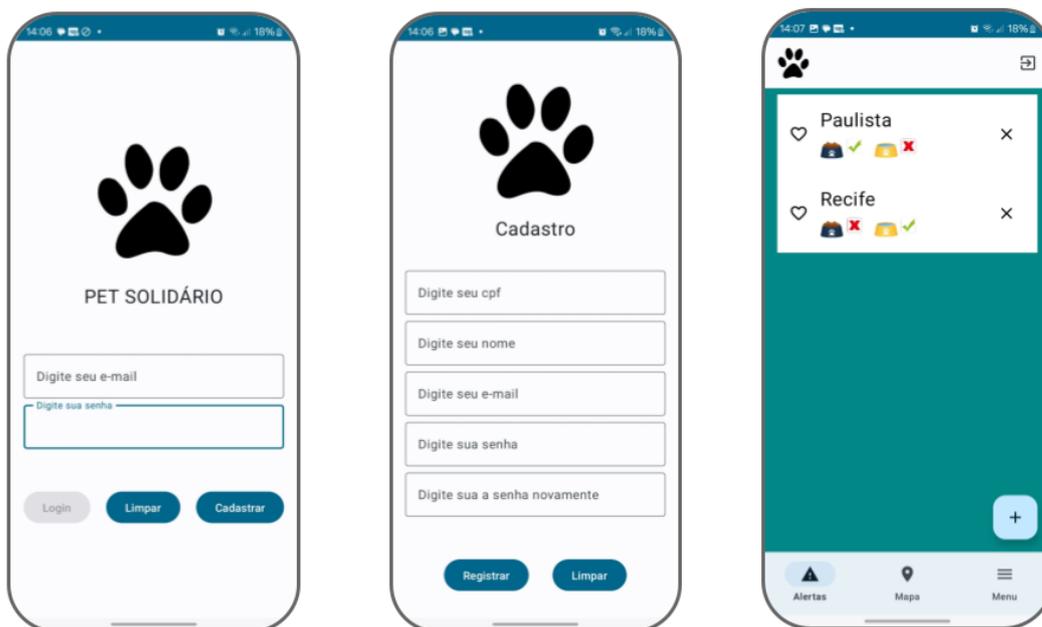
Teste	Observações
Teste 1	O sistema acionou corretamente o dispenser de ração quando o nível caiu abaixo do mínimo. No entanto, a imprecisão das medições da célula de carga resultou em falhas esporádicas na captura do peso.
Teste 2	O bebedouro foi reabastecido automaticamente quando o nível de água caiu para o nível crítico%.
Teste 3	O aplicativo foi atualizado com precisão e tempo adequado para cada evento de reabastecimento.
Teste 4	O aplicativo manteve a atualização sem falhas, com informações de consumo precisas.

Tabela 2 – Observações dos Testes

#### 5.3.1 Resultado final do Hardware e Software

Apresenta as interfaces de entrada no aplicativo, incluindo as telas de login, cadastro de usuário e a tela principal, onde o usuário acessa as funcionalidades do sistema.

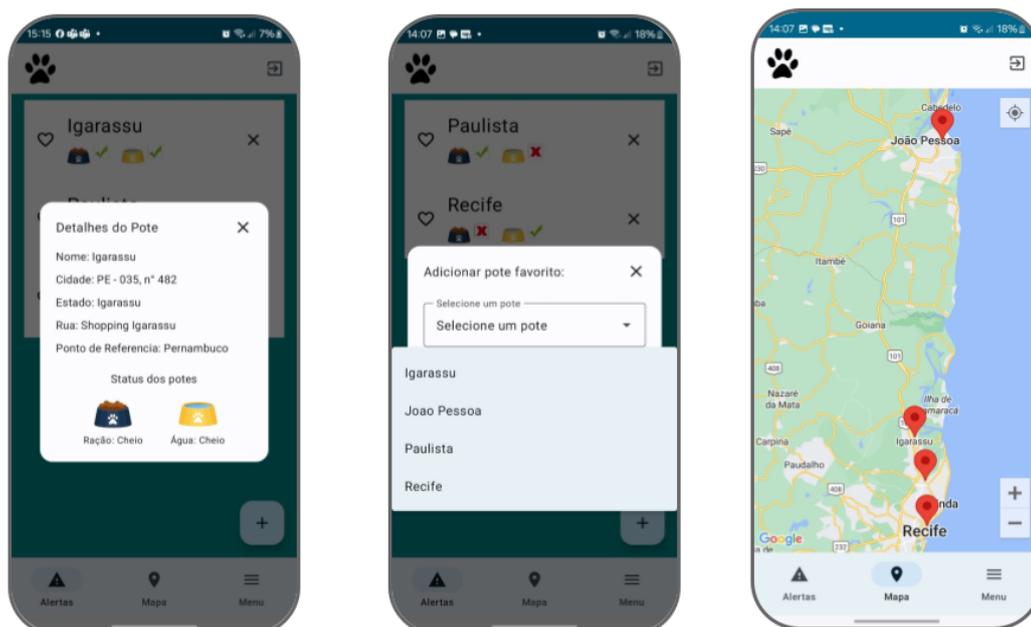
Figura 19 – Telas de Login, Cadastro de Usuário e Tela Principal.



Fonte: Próprio autor (2025)

Demonstra as principais funcionalidades disponíveis na tela inicial, incluindo a opção de cadastrar um pote como favorito, permitindo que o usuário receba notificações relacionadas a esse pote. Além disso, destaca a integração com o Google Maps, que exibe a localização dos dispositivos instalados, facilitando o monitoramento em tempo real.

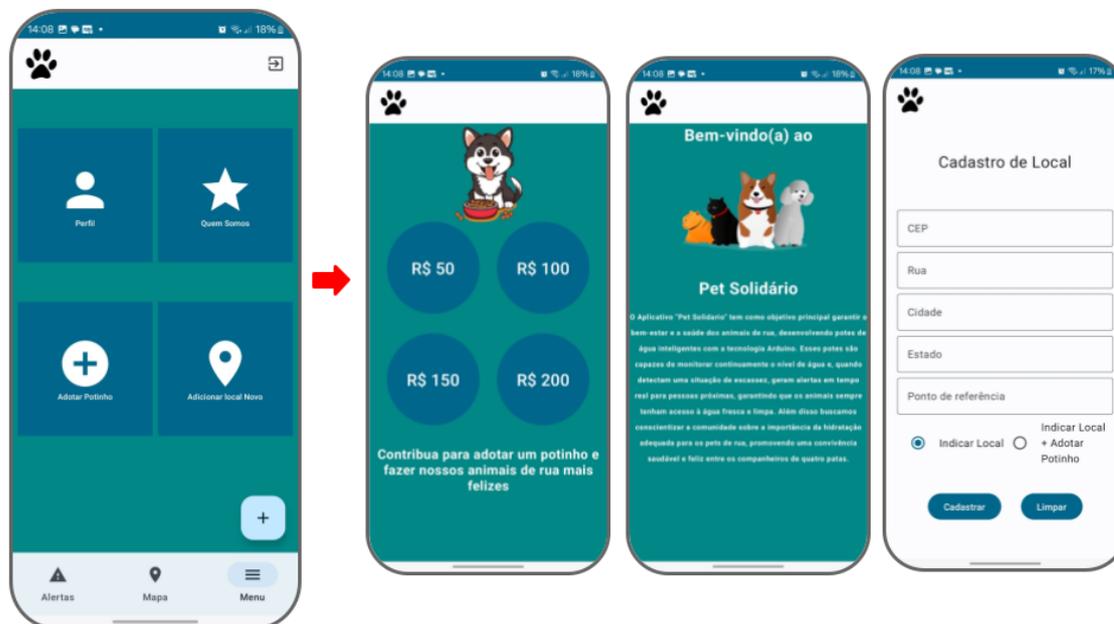
Figura 20 – Funcionalidades da tela principal e tela do mapa.



Fonte: Próprio autor (2025)

Exibe o menu de navegação do aplicativo, proporcionando acesso a diferentes seções, como o cadastro de novos locais para instalação dos potes, a opção de doação para a aquisição de novos dispositivos e a visualização das informações do usuário.

Figura 21 – Telas do menu.



Fonte: Próprio autor (2025)

Ilustra o protótipo físico dos dispositivos inteligentes desenvolvidos para o projeto, responsáveis pelo fornecimento automatizado de ração e água aos animais de rua.

Figura 22 – Comedouro e bebedouro.



Fonte: Próprio autor (2025)

#### 5.4 Análise dos Resultados

Os testes funcionais indicam que o sistema *Pet Solidário* opera de maneira eficiente e integrada. A precisão na medição dos níveis de água foi excelente, com a automação operando sem falhas durante os testes. A comunicação entre os sensores e o microcontrolador ESP32 foi estável, e a atualização dos dados no aplicativo foi realizada em tempo real, permitindo aos usuários o recebimento de informações instantâneas sobre o estado dos comedouros e bebedouros.

No entanto, um problema significativo foi identificado no processo de medição do peso da ração. A imprecisão na captura da deformação da célula de carga, medida pelo HX711, resultou em leituras inconsistentes, especialmente no monitoramento da quantidade de ração no comedouro. Essa falha foi mais evidente quando o peso registrado na tigela não correspondia à quantidade real de ração disponível, compro-

metendo a automação do abastecimento. Para solucionar esse problema, a célula de carga foi removida e substituída por um sensor ultrassônico, que libera uma quantidade fixa de ração sempre que detecta a presença do pet.

O sistema também apresentou um bom desempenho na geração de alertas, com a funcionalidade de notificações operando de forma confiável sempre que os níveis de água ou ração atingiam limites críticos. Além disso, a integração com a API do Google Maps permitiu que os pontos de alimentação fossem localizados de forma fácil, facilitando a gestão logística.

## 5.5 Discussão dos Resultados

Os resultados confirmam que a integração dos componentes do sistema, incluindo o microcontrolador ESP32, sensores, válvulas solenoides e o aplicativo Pet's, foi bem-sucedida. O sistema demonstrou uma automação eficiente e inteligente, proporcionando uma gestão contínua e precisa da alimentação e hidratação dos animais.

Contudo, a imprecisão observada na medição do peso da ração precisa ser endereçada. Considerando a importância da precisão nas medições para garantir o fornecimento adequado de ração e evitar desperdícios, ajustes na calibração do HX711 ou a utilização de uma célula de carga diferente devem ser considerados. Além disso, será necessário investigar a possibilidade de interferências externas que possam estar afetando as medições.

## 6 CONCLUSÃO

Este trabalho propôs o desenvolvimento de comedouros e bebedouros inteligentes integrados a um aplicativo móvel, com o objetivo principal de otimizar a alimentação e hidratação de animais de rua. Para atingir esse propósito, foram desenvolvidos dois dispositivos equipados com dispensers para água e ração, que utilizam tecnologia IoT para monitoramento em tempo real. Isso permite que o aplicativo móvel receba atualizações constantes sobre o status de cada pote. Além disso, o aplicativo oferece opções de doação para a produção de novos dispositivos e permite a indicação de locais com grande movimentação de animais em situação de rua.

A implementação obteve resultados positivos em relação aos objetivos inicialmente propostos, contribuindo significativamente para a melhoria das condições de vida de animais em situação de vulnerabilidade. Além disso, fortalece o engajamento de comunidades e organizações dedicadas à causa animal. A conexão com o aplicativo móvel possibilita um acompanhamento mais eficiente, permitindo a rápida identificação da necessidade de reabastecimento dos comedouros e bebedouros, tornando o processo mais ágil e preciso.

Como parte dos trabalhos futuros, propõe-se a implementação de algoritmos de inteligência artificial capazes de identificar o porte do animal e liberar automaticamente a quantidade ideal de ração para seu tamanho. Além disso, a adoção de painéis solares pode tornar o sistema mais sustentável e autônomo. Outra melhoria planejada é o aprimoramento da interface do aplicativo, com funcionalidades específicas voltadas para as pessoas responsáveis pela administração e manutenção dos potes distribuí-

dos, além da reimplantação da medição do peso da ração, com ênfase na melhoria da calibração do HX711 para garantir maior precisão.

Dessa forma, espera-se que o Pet Solidário continue expandindo sua rede de voluntários e parcerias, ampliando a cobertura e o impacto social do projeto, proporcionando melhores condições para os animais de rua e fortalecendo ações voltadas ao bem-estar animal.

## REFERÊNCIAS

ARDUCORE. **Válvula de Vazão Solenoide 12VDC 1/2 Polegadas**. 2024.

<https://www.arducore.com.br/valvula-de-vazao-solenoide-12v-1-2-polegadas>. Acesso em: 21 mar. 2025.

AZURE, Microsoft. **O que é um banco de dados NoSQL?** 2025.

<https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary>. Acesso em: 2 mar. 2025.

CLOUD, Google. **O que é o Cloud Storage?** 2024.

<https://cloud.google.com/learn/what-is-cloud-storage?hl=pt-BR>. Acesso em: 30 jan. 2025.

CUNHA, Camila Oliveira da et al. **Alimentador Automático**. 2020. Jorge Street.

<https://www.jorgestreet.com.br/wp-content/uploads/2020/03/TCC-ALIMENTADOR-AUTOMATICO.pdf>. Acesso em: 21 mar. 2025.

CURTOCIRCUITO. **Módulo Conversor Célula de Carga - HX711**. 2024.

<https://curtocircuito.com.br/modulo-conversor-celula-de-carga-hx711.html>. Acesso em: 21 mar. 2025.

\_\_\_\_\_. **Sensor de Peso 50Kg - Célula de Carga**. 2024.

<https://curtocircuito.com.br/sensor-celula-de-carga-50kg.html>. Acesso em: 21 mar. 2025.

FIREBASE. **Cloud Firestore**. 2023. <https://tinyurl.com/firebase-site>. Acesso em: 30 jan. 2025.

GARTNER. **O que são plataformas do setor na nuvem?** 2023.

<https://tinyurl.com/plataformas-setor-de-nuvem>. Acesso em: 30 jan. 2025.

IBM. **O que é a Internet das Coisas (IoT)?** 2024.

<https://www.ibm.com/br-pt/topics/internet-of-things>. Acesso em: 30 jan. 2025.

MARS. **1 em cada 3 animais de estimação está desabrigado**. 2023.

<https://www.mars.com/pt-br/news-and-stories>. Acesso em: 21 mar. 2025.

MORAIS, Luciana Alves de. **KI Pet Care**

- **Alimentador Automático para Pets**. 2021. Pontifícia Universidade Católica de Goiás.

[https://repositorio.pucgoias.edu.br/jspui/bitstream/123456789/5923/1/TCC\\_II\\_LucianaAlvesDeMoraes.pdf](https://repositorio.pucgoias.edu.br/jspui/bitstream/123456789/5923/1/TCC_II_LucianaAlvesDeMoraes.pdf). Acesso em: 21 mar. 2025.

MVC, Instituto. **Índice de Abandono no Brasil**. 2024.

<https://institutomvc.org.br/site/index.php/2024/04/04/indice-de-abandono-no-brasil/>. Acesso em: 20 fev. 2025.

ROBOCORE. **Sensor de Distância Ultrassônico HC-SR04**. 2024.

<https://www.robocore.net/sensor-robo/sensor-de-distancia-ultrassonico-hc-sr04>.

Acesso em: 21 mar. 2025.

\_\_\_\_\_. **Servo Motor TowerPro SG5010**. 2024.

<https://www.robocore.net/servo-motor/servo-sg5010>. Acesso em: 21 mar. 2025.

SARAVATI. **Sensor de Chuva e Nível de Água HR43**. 2024.

<https://www.saravati.com.br/sensor-de-chuva-e-nivel-de-agua-para-arduino-hr43.html>.

Acesso em: 21 mar. 2025.

SILVA LOMBAS, Jaqueline da et al. **Alimentador Automático para Pet**. 2020.

Centro Paula Souza.

[https://ric.cps.sp.gov.br/bitstream/123456789/8456/1/ALIMENTADOR\\_AUTOMATICO\\_PARA\\_PE](https://ric.cps.sp.gov.br/bitstream/123456789/8456/1/ALIMENTADOR_AUTOMATICO_PARA_PE)

Acesso em: 21 mar. 2025.

SOLUTIONS, MBM. **IoT e Sustentabilidade: Como Dispositivos Conectados Podem Salvar o Planeta**. 2024.

<https://blog.mbmsolutions.com.br/iot-e-sustentabilidade/>. Acesso em: 30 jan. 2025.

SYSTEMS, Espressif. **ESP32 - Overview**. 2024.

<https://www.espressif.com/en/products/socs/esp32>. Acesso em: 21 mar. 2025.

TONGLING. **Relé 12V 1 Canal**. 2024.

<https://www.byteflop.com.br/modulo-rele-1-canal-12v>. Acesso em: 21 mar. 2025.

VANIA; KARYONO, Kanisius; NUGROHO, I. Hargyo Tri. Smart dog feeder design using wireless communication, MQTT and Android client. In: 2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA).

[S.l.: s.n.], 2016. P. 191–196. DOI: 10.1109/IC3INA.2016.7863048.