

**UNIVERSIDADE REGIONAL INTEGRADA DO ALTO URUGUAI E DAS MISSÕES
PRÓ-REITORIA DE ENSINO, PESQUISA E PÓS-GRADUAÇÃO
CÂMPUS DE ERECHIM
DEPARTAMENTO DE ENGENHARIAS E CIÊNCIA DA COMPUTAÇÃO
CURSO DE ENGENHARIA ELÉTRICA**

EVANDRO LUIZ JACOBOSKI

**DISPOSITIVO DE SUPERVISÃO ALIMENTAR PARA
ANIMAIS DOMÉSTICOS**

TRABALHO DE CONCLUSÃO DE CURSO

**ERECHIM - RS
2021**

EVANDRO LUIZ JACOBOSKI

**DISPOSITIVO DE SUPERVISÃO ALIMENTAR PARA
ANIMAIS DOMÉSTICOS**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica como requisito parcial à obtenção do título de Bacharel em Engenharia Elétrica, Departamento de Engenharias e Ciência da Computação da Universidade Regional Integrada do Alto Uruguai e das Missões – Câmpus de Erechim.

Orientador: Prof. Cássio Luciano Baratieri

**ERECHIM - RS
2021**

EVANDRO LUIZ JACOBOSKI

**DISPOSITIVO DE SUPERVISÃO ALIMENTAR PARA
ANIMAIS DOMÉSTICOS**

**Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia Elétrica como
requisito parcial à obtenção do título de
Bacharel em Engenharia Elétrica,
Departamento de Engenharias e Ciência da
Computação da Universidade Regional
Integrada do Alto Uruguai e das Missões –
Câmpus de Erechim.**

Erechim, 09 de julho de 2021.

BANCA EXAMINADORA

Prof. Dr. Cássio Luciano Baratieri (Orientador)
URI - Erechim

Prof.^a. Ma. Camila Sampaio dos Reis (Examinadora)
URI - Erechim

Prof. Me. Glênio Luis de Vasconcellos Rigoni (Examinador)
URI - Erechim

“[Em menção a invenção do motor de indução]

A ideia veio como um relâmpago e num instante a verdade foi revelada.”

Nikola Tesla.

AGRADECIMENTOS

À minha esposa Franciele, pelo apoio incondicional, por suportar a minha ausência e serenidade durante todo o curso de Engenharia Elétrica, pela compreensão e amor que me fizeram ter ainda mais empenho.

À toda minha família, pelo apoio e motivação que me fizeram continuar.

Aos meus colegas que permaneceram ao meu lado durante toda a Graduação, principalmente aos que participaram do nosso grupo de estudos, onde podemos aprender uns com os outros.

Ao meu orientador Prof. Cássio Luciano Baratieri pelos ensinamentos e pela disponibilidade em me atender, sanando minhas dúvidas e contribuindo para este trabalho.

A Marciele e a todos os funcionários da Universidade Regional Integrada do Alto Uruguai e das Missões - Campus de Erechim que me auxiliaram na trajetória acadêmica.

Por fim, meu sincero agradecimento aos professores do Curso de Engenharia Elétrica que contribuíram para a minha formação acadêmica e profissional.

RESUMO

Este trabalho apresenta o desenvolvimento de um protótipo de produto com automação *pet* para alimentação de animais domésticos. O protótipo tem como objetivo automatizar o processo de alimentação de animais domésticos, de forma autônoma e/ou remota, assim como o monitoramento e registro das variáveis atribuídas ao sistema. Para tanto, considerou-se uma forma prática, rápida e segura para o tutor ter este controle. O projeto de automação tem o *Single Board Computer Raspberry Pi 2* como principal dispositivo, um *System-on-Chip* utilizando *RaspberryOS*. A programação da lógica da automação está na linguagem Python e SQL. As variáveis de operação são controladas e monitoradas via interface gráfica web. A programação da lógica da interface gráfica está na linguagem PHP, HTML e SQL. Foi idealizado um protótipo em 3D, considerando a necessidade de alimento e água. Por fim foram apresentados os resultados das simulações em ambiente computacional e dos testes realizados em bancada experimental. Através deste protótipo é possível, determinar a quantidade de refeições, agendar horários para refeição, determinar a quantidade de ração a ser liberada, sendo consideradas variáveis como os tipos de rações, e suas dimensões. Foram feito teste de acertividade comprovando a eficácia do produto, demonstrando que o alimentador possibilita o manejo meticuloso da alimentação.

Palavras-chave: Alimentador automático. Automação. Animais de estimação. Sistema em um Chip.

ABSTRACT

This work presents a product prototype with pet automation for feeding domestic animals. The prototype aims to automate the process of feeding domestic animals, autonomously and / or remotely, as well as monitoring and recording the variables attributed to the system. For this, it was considered a practical, fast and safe way for the tutor to have this control. The automation project has the Single Board Computer *Raspberry Pi 2* as its main device, a System-on-Chip using *RaspberryOS*. The programming of the automation logic is in Python and SQL. The operating variables are controlled and monitored via a graphical web interface. The logic programming of the graphical interface is in PHP, HTML and SQL. A 3D prototype was designed, considering the need for food and water. Finally, the results of simulations in a computer environment and tests carried out on an experimental bench were presented. Through this prototype it is possible to determine the amount of meals, schedule times for meals, determine the amount of ration to be released, considering variables such as the types of rations and their dimensions. An accuracy test was carried out, proving the effectiveness of the product, demonstrating that the feeder enables meticulous feeding management.

Keywords: Automatic feeder. Automation. Pet. System-on-Chip.

LISTA DE ILUSTRAÇÕES

Figura 1 – Pirâmide de automação.	21
Figura 2 – Instruções “Se-Então”.	23
Figura 3 – Encapsulamentos de Chips.	24
Figura 4 – 12 Fatores determinantes da IoT.	26
Figura 5 – SOC- Diagrama de blocos do <i>Raspberry Pi 3</i>	27
Figura 6 – <i>Raspberry Pi</i>	28
Figura 7 – Alimentador 1.	33
Figura 8 – Alimentador 2.	33
Figura 9 – Alimentador 3.	33
Figura 10 – Alimentador 4.	34
Figura 11 – Alimentador 5.	34
Figura 12 – Fluxograma da Operação.	38
Figura 13 – Simbologia <i>RaspberryOS</i> , GPIO e Python.	39
Figura 14 – Especificação de <i>Hardware Raspberry Pi</i>	40
Figura 15 – Célula de carga.	40
Figura 16 – Sensor de distância ultrassônico HC-SR04.	41
Figura 17 – Motor 28BYJ-48.	42
Figura 18 – Protótipo Completo (50% transparente), vista isométrica.	44
Figura 19 – Protótipo Bloco 1 (50% transparente), vista isométrica.	44
Figura 20 – Protótipo Bloco 2 (50% transparente), vista isométrica.	45
Figura 21 – Protótipo Bloco 3, vista isométrica.	45
Figura 22 – Protótipo Completo, vista superior.	46
Figura 23 – Protótipo, detalhe acoplamento motor a rosca helicoidal.	46
Figura 24 – Protótipo em escala Completo, vista isométrica.	47
Figura 25 – Construção final do protótipo, vista superior.	47
Figura 26 – Protótipo em escala Completo, funcional, vista superior.	48
Figura 27 – PCB Projeto, visualização em 3D.	49
Figura 28 – Projeto interface Comando/Potência – Detalhe isolamento comando motor.	50
Figura 29 – Projeto interface Comando/Potência – CI Driver e Conexões externas.	50
Figura 30 – Projeto interface Comando/Potência – CI Célula de carga e sensor ultrassônico.	51
Figura 31 – Interface entre <i>Raspberry Pi</i> e Sensores/Atuadores.	52
Figura 32 – Detalhe esquema elétrico.	55

Figura 33 – Interface gráfica – Tela inicial (Home).....	65
Figura 34 – <i>E-mail</i> de alerta.	66
Figura 35 – Vista frontal do produto.	67
Figura 36 – Interface gráfica – Tela inicial (Home).....	82
Figura 37 – Interface gráfica – Tela parâmetros.....	83
Figura 38 – Interface gráfica – Tela agendador.....	84
Figura 39 – Interface gráfica – Tela Cadastro Usuário	84
Figura 40 – Interface gráfica – Tela Cadastro <i>Pet</i>	85
Figura 41 – Interface gráfica – Tela consulta e edição agenda	86
Figura 42 – Interface gráfica – Tela consulta e edição usuário	86
Figura 43 – Interface gráfica – Tela consulta e edição <i>Pet</i>	86
Figura 44 – Interface gráfica – Tela consulta e edição parâmetros	86
Figura 45 – Interface gráfica – Tela consulta histórico	86
Quadro 1 – Descrição/Parâmetros para o sistema proposto.	37

LISTA DE TABELAS

Tabela 1 – Pinos GPIO	60
Tabela 2 – Lista dos itens	61
Tabela 3 – Descrição dos custos.....	62
Tabela 4 – Custos do Produto e Importação.....	63
Tabela 5 – Custos do produto e preço de venda.....	64
Tabela 6 – Estatística Amostral	69
Tabela 7 – Medições de corrente.....	71

LISTA DE ABREVIATURAS E SIGLAS

ABINPET	<i>Associação Brasileira de Indústria de Produtos para Animais de Estimação</i>
AOS	Arquitetura Orientada a Serviços
CAD	<i>Computer-Aided Design</i>
CEDOM	<i>Asociación Española de Domótica e Inmótica</i>
CMOS	<i>Complementary Metal-Oxide Semiconductor</i>
CNPJ	Cadastro Nacional de Pessoa Jurídica
COFINS	Contribuição para o Financiamento da Seguridade Social
CPF	Cadastro de Pessoa Física
CPS	<i>Cyber Physical System</i>
DIFAL	Diferencial de Alíquota de ICMS entre Unidades Federativas do Brasil
DIY	<i>Do It Yourself</i>
EEPROM	<i>Electrically-Erasable Programmable Read-Only Memory</i>
EPROM	<i>Erasable Programmable Read-Only Memory</i>
FLASH	<i>Flash Memory</i>
FPGA	<i>Field Programmable Gate Array</i>
GPIO	<i>General Purpose Input/Output</i>
HTML	<i>HyperText Markup Language</i>
IBAMA	Instituto Brasileiro do Meio Ambiente
ICMS	Imposto sobre Circulação de Mercadorias e Serviços
IDC	<i>International Data Corporation</i>
ISS	Imposto Sobre Serviços
IOS	<i>Iphone Operating System</i>
IoT	<i>Internet of Things</i>
IR	<i>Infrared</i>
LAMP	Linux, Apache, MySQL, PHP
LED	<i>Light-Emitting Diode</i>
MySQL	Sistema de Gerenciamento de Banco de Dados, Proprietário da Oracle Corporation.
MEI	Micro Empreendedor Individual
NPN	<i>Negativo Positivo Negativo</i>
OS	<i>Operating System</i>
PCB	<i>Printed Circuit Board</i>

PIS	Programa de Integração Social
PHP	<i>Hypertext Preprocessor</i>
SBC	<i>Single-Board Computers</i>
SBM	<i>Single-Board Microcontroller</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
SMD	<i>Surface Mounted Device</i>
SQL	<i>Structured Query Language</i>
SOC	<i>System On Chip</i>
STL	<i>Standard Triangle Language ou Standard Tessellation Language</i>
RAM	<i>Random Access Memory</i>
RFID	<i>Radio-Frequency IDentification</i>
ROM	<i>Read-Only Memory</i>
RTOS	<i>Real-Time Operating System</i>
TTL	<i>Transistor-Transistor Logic</i>
USB	<i>Universal Serial Bus</i>
WIBS	<i>Web Based Information Systems</i>

SUMÁRIO

1	INTRODUÇÃO.....	15
1.1	Justificativa.....	16
1.2	Objetivos.....	16
1.3	Estrutura do trabalho.....	17
2	REVISÃO DE LITERATURA.....	18
2.1	Automação <i>pet</i>	18
2.2	Produtos Comerciais Relacionados.....	20
2.3	Pirâmide da automação.....	21
2.3.1	Projeto de Automação.....	22
2.3.2	Sensores.....	23
2.3.3	Atuadores.....	23
2.3.4	Circuitos Integrados.....	24
2.3.5	Sistemas físicos cibernéticos.....	24
2.3.5.1	Internet das coisas.....	25
2.3.5.2	Internet de serviços.....	26
2.4	System-on-Chip.....	27
2.4.1	Arquitetura <i>Single Board Computers – Raspberry Pi</i>	27
2.4.2	Linguagem de programação Python.....	29
2.4.3	Linguagem de programação HTML.....	29
2.4.4	Linguagem de programação PHP.....	29
2.4.5	Linguagem de programação SQL.....	30
2.4.6	Linguagem de programação Terminal ou <i>Shell</i>	30
2.4.7	<i>Firmware</i>	31
2.4.8	Sistema de Informação Baseado na WEB.....	32
2.5	Prototipagem Rápida.....	32
3	METODOLOGIA.....	35
3.1	Descrição do fluxo.....	35
3.1.1	Requisitos da Automação.....	35
3.2	Descrição do sistema proposto.....	36
3.2.1	Fluxograma da automação.....	37
3.2.2	Infraestrutura de Gerenciamento.....	38
3.2.3	SBC – <i>Raspberry Pi</i>	39
3.2.4	Sensor de peso – balança da tigela.....	40
3.2.5	Sensor de distância ultrassônico – água e ração.....	41
3.2.6	Motor de passo – transportadora.....	41
3.3	Descrição do Projeto.....	42
3.3.1	Optoacoplador - CI PC817.....	42
3.3.2	Circuito Integrado - CI ULN2003A.....	43
3.3.3	Amplificador Operacional – CI HX711.....	43
3.3.4	Descrição do Protótipo.....	44
3.4	Impressão do Protótipo.....	47
3.5	<i>Printed Circuit Board</i> - Interface de Comando/Potência.....	49
3.6	Projeto Interface de Potência.....	49
3.7	Descrição do Funcionamento.....	53
3.7.1	Sensor Ultrassônico HC-SR04.....	53
3.7.2	Motor de Passo 28BYJ-48.....	55
3.7.3	CI ULN2003A.....	57

3.7.4	Célula de Carga.....	58
3.7.5	CI HX711.....	59
3.7.6	<i>RASPBERRY PI</i>	59
3.7.7	Parâmetros Gerais.....	60
3.7.8	Componentes Eletrônicos Auxiliares.....	61
3.8	Materiais utilizados.....	61
3.9	Orçamento do Projeto.....	62
3.10	Planejamento Empresarial.....	64
4	RESULTADOS E DISCUSSÕES.....	65
4.1	Configuração da interface gráfica.....	65
4.2	Envio de <i>E-mail</i> ao tutor.....	65
4.3	Construção Física em 3D do Protótipo.....	66
4.4	Testes e medições.....	67
4.4.1	Teste em bancada da comunicação entre os dispositivos.....	68
4.4.1.1	Teste de Falha de energização.....	68
4.4.1.2	Teste de Seccionamento de cabos.....	68
4.4.1.3	Teste de Inversão das conexões dos <i>plugs</i> dos sensores e atuadores.....	68
4.4.2	Precisão da Balança.....	69
4.4.3	Precisão do Sensor dos Reservatórios.....	70
4.4.4	Teste de Desempenho de Comandos.....	70
4.4.5	Teste de Assertividade de Comandos.....	71
4.4.6	Medições.....	71
4.5	Simulação/cadastros.....	71
4.5.1	Registro das informações.....	72
4.6	Falhas, Erros e Desistências.....	72
4.7	Possibilidades de melhorias.....	73
5	CONSIDERAÇÕES FINAIS.....	74
5.1	Sugestão para Continuação do Trabalho.....	75
	REFERÊNCIAS.....	76
	APÊNDICE A – TELAS.....	82
	Tela Inicial.....	82
	Telas de Gerenciamento.....	82
	Telas de Cadastro – Usuário e Pet.....	84
	Telas de Consulta e Edição.....	85

1 INTRODUÇÃO

O Brasil é o terceiro maior país em população total de animais de estimação, com um total de 139,3 milhões de *pet*, movimentando em 2019, um montante de R\$ 22,3 bilhões de reais (ABINPET, 2020). Assim, é possível perceber um significativo potencial à inserção de novas tendências, tecnologias e inovações, em um mercado em ascensão. Um mercado que tem mudado o estilo de vida de muitas pessoas, introduzindo o *pet* em muitas casas, sendo muitas vezes, considerado como um “membro familiar”.

O conceito de animal de estimação é amplo. Para o IBAMA (1998), todos aqueles animais que, através de processos tradicionais e sistematizados de manejo e/ou melhoramento zootécnico, tornaram-se domésticos, com características biológicas e comportamentais em estreita dependência do homem e, características variáveis, diferente da espécie silvestre que os originou, são denominados animais de estimação.

Não há como definir a evolução exata de todos os animais domésticos, já que esse processo foi iniciado muitos anos antes das civilizações antigas, sendo muitos desenhos encontrados em cavernas, indicando que nossos ancestrais possuam conhecimento sobre os hábitos dos animais. Por muitas vezes os tinham como animais de companhia demonstrando cuidado e afeto, o que se perpetua até os dias atuais, aonde a tecnologia, permeia o cotidiano (DALY, 2020).

Em geral, busca-se no animal de estimação, a companhia para diminuir os efeitos do estresse, solidão e apoio psicológico em situações mais extremas, porém a rotina agitada do cotidiano acaba por intercalar as horas de afeto e cuidado com viagens e compromissos. Para muitos o afeto e a dedicação para cuidar dos *pets* se torna primordial, muitas vezes utilizando-se de tecnologias modernas no âmbito *pet*.

Com a disseminação da tecnologia, ela deixa de ser uma exclusividade dos setores privados e invade os ambientes domésticos, proporcionando mais conforto, sofisticação e praticidade. Esse ambiente tecnológico é cada vez mais almejado pelas famílias. Segundo dados do IDC (2020), o segmento de *Internet of Things* (IoT) deve movimentar US\$ 745 bilhões no mundo em 2019, com potencial para ultrapassar US\$ 1 trilhão em 2022.

Neste contexto tecnológico, insere-se a proposta deste trabalho que trata sobre um alimentador automatizado que auxilia na prevenção de possíveis descuidos com relação ao fornecimento do alimento aos animais e ao mesmo tempo conciliar os compromissos do cotidiano.

Este tipo de equipamento permite agendamento dos horários de alimentação, monitoramento dos níveis de consumo dos alimentos e ajuste automático da programação de acordo com a preferência do animal, tutor ou recomendação do veterinário, além de ser possível controlá-lo, por meio da internet, via página web.

1.1 Justificativa

Com o constante crescimento da IoT aplicada a *Smart Home*, surge uma oportunidade para inserção de produtos para automatização doméstica destinados aos pets. Este produto atenderá a um nicho de mercado, sendo o dos consumidores com animais domésticos, que ficam fora de casa por longos períodos de tempo, ou aqueles que desejam ter um sistema que os auxilie no processo de reger a alimentação de seus *pets*.

1.2 Objetivos

O objetivo geral deste trabalho é o desenvolvimento de um alimentador *pet* de baixo custo baseado em *System-on-Chip* (SOC), funcionando de forma autônoma e/ou remota, via página web, adotando sistema operacional *RaspberryOS*. Para alcançar o objetivo geral, foram realizados os seguintes objetivos específicos:

- Projetar um protótipo de alimentador automático/remoto para o *pet* com desenvolvimento de eletrônica embarcada;
- Desenvolver uma página web para controle e monitoramento remoto;
- Elaborar um servidor com o sistema operacional *RaspberryOS*, instalação do *Apache Web Server*, instalação do PHP (*Hypertext Preprocessor*) e modelagem de um banco de dados em MySQL;
- Desenvolver o protótipo em 3D com acoplamento de sensores e atuadores;
- Efetuar a impressão do protótipo 3D em escala reduzida;
- Elaborar a placa de circuito eletrônico de instrumentação e controle;
- Realizar testes do protótipo em bancada.

1.3 Estrutura do trabalho

A estruturação deste trabalho está dividida em cinco capítulos. No primeiro capítulo é descrita a introdução, que aborda em si a apresentação do tema, determinando a devida importância que a mesma desempenha na indústria moderna e objetiva, que detalha em si os passos da automação e determina a justificativa do trabalho.

No segundo capítulo, é contextualizada a revisão bibliográfica que se utilizou como suporte e amparo para a realização do estudo, por intermédio de publicações de pares, automação residencial, acionamentos elétricos, bem como artigos e trabalhos acadêmicos de especialização com o mesmo assunto abordado.

O terceiro capítulo é voltado à metodologia de estudo aplicada para a obtenção dos resultados, através das avaliações realizadas e levantamentos de dados.

No quarto capítulo, se abordam os resultados e discussões sobre o estudo realizado, em que se apresentam os resultados obtidos com as avaliações e levantamentos, assim como as discussões sobre os mesmos.

O último capítulo trata das considerações finais sobre os resultados obtidos com a realização do Trabalho de Conclusão de Curso.

2 REVISÃO DE LITERATURA

Este capítulo apresenta uma revisão de literatura acerca de alimentadores automáticos, fundamentando conceitos e propostas de pares e diversos produtos comercializados ou em financiamento coletivo, bem como projetos *Do It Yourself* (DIY), sendo, aqueles que a própria pessoa monta, melhora ou cria um determinado dispositivo, mediante alguma necessidade.

2.1 Automação *pet*

Considerando-se que a automação residencial é o conjunto de tecnologias aplicadas na conexão e integração entre sistemas inteligentes para viabilizar a execução de atividades de forma autônoma, permitindo a gestão eficiente. Seus principais benefícios são o aumento de conforto, bem-estar, praticidade e segurança, além da comunicação entre o usuário e o sistema. Nos últimos anos, passando a abranger também, automações voltadas aos *pets*.

Com o surgimento de produtos domésticos inteligentes para animais de estimação, está mais fácil ser um tutor, pois mesmo quando ocupado, ainda poderá ver e alimentar remotamente o *pet*. Além disso, algumas dessas tecnologias inteligentes de monitoramento para animais de estimação também podem mantê-los seguros e saudáveis.

É possível automatizar sua casa e controlar grande parte dela remotamente através de um aplicativo, estendendo-se também para tarefas relacionadas ao *pet*. Para muitos, os animais não são apenas animais de estimação, eles são membros da família. Muitas famílias gastam milhares de reais por ano para garantir que seus animais de estimação recebam os melhores produtos, cuidados e atenção. Onde há demanda, fabricantes inovadores estão lá para fornecer.

Considerando-se que há demanda, são descritas algumas das possibilidades existentes para usar a tecnologia em gerenciar animais de estimação. Faz sentido ter algum tipo de capacidade de gerenciamento de animais de estimação em uma casa inteligente. Então serão listados alguns dos produtos atualmente disponíveis, com o intuito de facilitar a vida dos *pets* e tutores:

- *PetChatz Smart Pet Camera* afirma ser uma “creche digital” para seu animal de estimação. É um produto para interagir com seus cães ou gatos quando está fora. *PetChatz* vem com áudio bidirecional e câmera de vídeo HD, detecção de movimento,

som, um recurso de distribuição de guloseimas, aromaterapia, gravação de vídeo e entretenimento (PETCHATZ, 2020).

- *Whistle Go* Coleira inteligente é um rastreador de animais de estimação que pode ser preso a uma coleira. Notifica quando o animal de estimação deixa uma zona segura designada. Pode-se ver a localização do animal de estimação em qualquer lugar em que o serviço de celular 3G esteja disponível. Além disso, ajuda a monitorar a saúde do seu animal de estimação, fornecendo informações sobre as atividades diárias e como sua saúde está em comparação com outros cães da mesma raça, peso e idade (TRICKDROID, 2020).
- *Litter-Robot 3 Connect* é uma caixa de areia com autolimpeza habilitada para *Wi-Fi*. Além da autolimpeza acompanha um aplicativo ajudando a monitorar os hábitos de banheiro dos seus gatos. Equipado com informações sobre o nível da gaveta de resíduos e histórico de uso recente. O aplicativo também permite que você gerencie as configurações do dispositivo (LITTER-ROBOT, 2020).
- *PetSafe Smart Feed* é um comedouro para animais de estimação com *Wi-Fi*. Com o aplicativo, podem-se dispensar as refeições a qualquer hora e lugar. Permite agendar horários de refeições para seus animais de estimação facilmente. Ele vem com recursos de refeição personalizáveis, onde é possível escolher uma opção de alimentação lenta (PETS SAFE, 2020).
- *iFetch* é um lançador de bola automático, que mantém o cão ativo (IFETCH, 2020).
- *Sureflap* é uma porta para o animal de estimação, abrirá a porta apenas quando o dispositivo ler o microchip do *pet* (PETSATHOME, 2020).
- *Wonderwoof* é rede social de atividade de cachorro, uma rede social para cães com rastreamento de atividades (SOFTWAREBROTHERS, 2020).
- *Go Pro Fetch*: Arnês para animais de estimação, se prende ao seu cão para permitir que você grave vídeos através dos olhos do seu animal de estimação. À prova d'água também (PROAVENTURA, 2020)
- *Hipster Harness* é destinado para cães com displasia de quadril, é um dispositivo que ajuda os cães a estabilizar e fortalecer os músculos do quadril e evitar mais lesões (PEDEN, 2020).
- O *Clever PET* oferece uma maneira para os humanos interagirem com animais de estimação remotamente (CLEVERPET, 2020).
- *Tailio* é um sistema de monitoramento do peso de gatos, hábito alimentar e caixa de areia, conectado sob uma caixa de areia (TAILIO, 2020).

- *PupPod* é um quebra-cabeça de vários níveis que estimula a curiosidade dos cães e direciona a energia deles por meio de uma série de jogos habilitados para aplicativos e guloseimas (PUPPOD, 2020).
- *Pet Camera Smart Ball* é uma câmera tudo-em-um e bola de jogo conectada (PLAYDATE, 2020).
- *Felix e Fido PetBot*: O robô de estimação que cuida do *pet* (FIDO, 2020).
- *PetSafe Zoo Rotating Zoo Cat Toy*: é um laser giratório para animais de estimação que mantém o gato ocupado (LASERTOY, 2020).

Esses diversos produtos direcionados a *pets*, demonstram a vasta quantidade de dispositivos que estão presentes do mercado, para atender as mais diversas necessidades dos tutores. Além disso, evidencia que este mercado está em crescimento, indicando uma preocupação com a saúde mental e física dos *pets*, o que o torna muito atrativo a investidores, como um local de desenvolvimento de tecnologias e produtos.

2.2 Produtos Comerciais Relacionados

Esta seção apresenta soluções disponíveis no mercado que utilizam princípios e tecnologias que de alguma forma são semelhantes aos adotados neste trabalho.

O primeiro exemplo é o "*PetSafe Smart Feed*", com venda na Amazon, (USD 170.00), possui as seguintes funcionalidades "Utilizando-se do *smartphone* é possível alimentar o *pet* a qualquer hora, de qualquer lugar. Com o aplicativo em seu *smartphone*, é possível conectar ao alimentador e ao *Wi-Fi* da casa, podendo-se agendar, ajustar e monitorar as refeições do animal de estimação. Este alimentador inteligente funciona com o dispositivo Amazon Echo para que se possa pedir rapidamente a Alexa para alimentar o animal de estimação com um lanche sem usar as mãos. O sensor de pouca comida enviará uma notificação para o telefone quando o alimentador estiver com pouca ração e quando estiver vazio. Com a ajuda do sensor, é possível usar o *Amazon Dash Replenishment* para adquirir automaticamente a comida do animal de estimação" (PETS SAFE, 2020).

Outro alimentador que existe no mercado internacional é o CatFi. O CatFi Pro é o alimentador inteligente com tecnologia de reconhecimento facial para gatos, que permite o acompanhamento das estatísticas de consumo dos gatos (peso, alimentação e entradas de água) com uma função de alerta. Utiliza-se de reconhecimento facial e vale destacar que projeto do CatFi foi financiado através de doações no valor de US\$ 240.680,00, cuja campanha de arrecadação finalizou em 14 de agosto de 2014. (P3, 2020).

Por fim, cita-se o SuperFeeder, criado em 1992 com a primeira produção em 1994. A especialidade foi a fabricação e venda de alimentadores automáticos de qualidade. Segundo os fabricantes que buscavam desenvolver um alimentador verdadeiramente confiável, para alimentos em flocos, que não só poderia ser usado em uma breve temporada de férias, mas também poderia ser instalado de forma permanente. Contudo possui a limitação do alimentador com a falta de um dispositivo de agendamento de horários (SUPERFEEDER, 2020).

2.3 Pirâmide da automação

A automação de um equipamento de uso residencial pode ser baseada em alguns elementos da automação para fins industriais. A automação industrial exige a realização de muitas funções e os diferentes níveis de automação estão representados em forma de pirâmide na Figura 1.

Figura 1 – Pirâmide de automação.



Fonte: Adaptado de Silva (2017).

Conforme SILVA (2017), a pirâmide estabelece os níveis do processo produtivo, desde sua base até o topo de forma hierárquica:

- Nível 1: Na base da pirâmide, estão os equipamentos mais próximos ao produto: sensores, botoeiras, atuadores, entre outros. O montante de informação é pequeno, no entanto, a velocidade do processamento é alta (CAPELLI, 2013).
- Nível 2: É o nível dos controladores digitais, dinâmicos e lógicos, e de algum tipo de supervisão associada ao processo. Neste nível, se encontram concentradores de

informações sobre o Nível 1, e as Interfaces Homem-Máquina (IHM) (MORAES e CASTRUCCI, 2010).

- Nível 3: Inicia-se com a supervisão e o controle dos recursos e a otimização do processo. Além disso, é responsável pelo controle geral de todo o processo produtivo (CAPELLI, 2013).
- Nível 4: É onde ocorre a programação e o planejamento da produção, realizando o controle e a logística dos estoques (MORAES e CASTRUCCI, 2010).
- Nível 5: No topo da pirâmide que acontece a gestão estratégica da empresa. Os computadores trabalham no controle corporativo, desde a entrada de material até a gestão de recursos (CAPELLI, 2013).

Como pode ser observado, a pirâmide tem foco no contexto corporativo, ou industrial, porém de forma análoga, pode-se aplicar a mesma pirâmide, mantendo-se as peculiaridades ao contexto residencial.

Neste caso a pirâmide estabelece os níveis da operação, desde sua base até o topo de forma hierárquica:

- Nível 1: Na base da pirâmide, estão os sensores, botões, atuadores, entre outros.
- Nível 2: É o nível dos controladores digitais, aonde se encontram concentradores de informações sobre o Nível 1.
- Nível 3: Inicia-se com a supervisão, informando ao usuário.
- Nível 4: É onde ocorre a programação, mediante a informação provida pela Nível 3.
- Nível 5: No topo da pirâmide que acontece a gestão doméstica.

Recentemente a internet das coisas está sendo utilizada como o canal de conexão de cada elemento de um sistema interligado, gerando a interconectividade. Deste modo apresentando vantagens promissoras, por permitir aos componentes se comportarem como uma “coisa”, independentemente de suas características originais construtivas.

2.3.1 Projeto de Automação

A automação de projetos é um processo executado em segundo plano, ou independente de interação humana, sem esforço manual. Esse processo é feito usando instruções "se-então", conforme pode ser verificado na Figura 2. Assim é possível usar a tecnologia a favor de melhorias, otimizações, ou como também tornar aquele processo livre de erros humanos.

Figura 2 – Instruções “Se-Então”.



Fonte: WILCOX (2020)

2.3.2 Sensores

Segundo o dicionário Michaelis (2020), sensor é um “dispositivo ou equipamento que, sensível a estímulos magnéticos, motores, de calor, de luz, de pressão, de som etc., é capaz de converter essa energia e transmitir um impulso correspondente”. Portanto, a função primordial de um sensor é a de monitorar algum estímulo conforme sua construção.

“Um elemento essencial da maioria dos sistemas de controle são os sensores de entrada. Sensores são os componentes que detectam mudanças físicas ou eventos e os convertem em sinais elétricos para serem processados.” (FRENZEL, 2018, p.295, tradução nossa).

De forma complementar Frenzel (2014, p.4, tradução nossa), afirma que “Monitorar significa observar e medir estados físicos como temperatura, pressão, posição mecânica, nível de líquido ou intensidade de luz. Sensores convertem essas características físicas em sinais elétricos que podem ser processados por circuitos eletrônicos.”

Nos produtos relacionados, os sensores mais utilizados são os de *Radio-Frequency Identification* (RFID), sensor de presença, temperatura, nível e peso. De modo geral a combinação dos sensores traz uma precisão maior ao produto, tornando-o mais confiável.

2.3.3 Atuadores

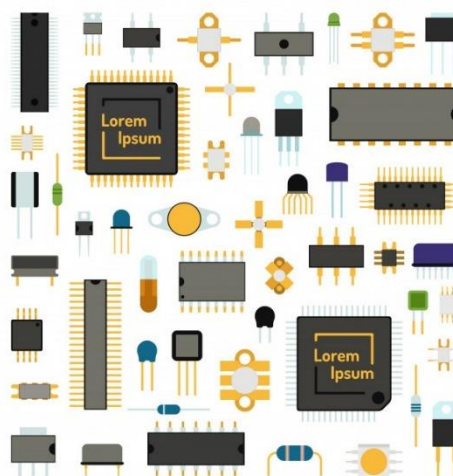
“Um dispositivo de saída ou um atuador é o mais frequentemente controlado em um sistema de controle. Os dois tipos mais comuns de atuadores são relés e motores.” (FRENZEL, 2018, p.303, tradução nossa)

Nestes alimentadores, normalmente, são utilizados motores de passo, ou motores CC com redução nestes alimentadores, pelo fácil controle e baixo custo. O motor de passo sempre sendo utilizado nos dispositivos que necessitam de um controle maior da dosagem da ração.

2.3.4 Circuitos Integrados

Circuito integrado é uma pequena pastilha (chip), que contém em seu interior milhares de componentes eletrônicos (transistores, diodos, resistências, capacitores e suas interligações). Estes componentes eletrônicos são formadores de portas lógicas que, interligadas, formam um determinado circuito digital. A pastilha (chip) é encapsulada em um pacote de cerâmica ou plástico e as conexões com o exterior são soldadas aos pinos externos para completar o dispositivo (NETO, 2018). Na Figura 3, é destacada alguns dos encapsulamentos disponíveis comercialmente dos circuitos integrados e alguns outros componentes.

Figura 3 – Encapsulamentos de Chips.



Fonte: VSSSTUDIO (2020)

2.3.5 Sistemas físicos cibernéticos

Os sistemas físicos cibernéticos são usados onde sistemas físicos complexos necessitem se comunicar com o *software* para permitir que seu desempenho seja otimizado e sua eficiência, melhorada, desempenhando um papel importante no processo industrial e no controle de produção. “Os sistemas físicos cibernéticos combinam capacidades cibernéticas com capacidades físicas para resolver problemas que nenhuma das partes poderia resolver sozinha.” (PLATZER, 2018, p.1, tradução nossa)

Para Vogel-Heuser (2014), os sistemas físicos cibernéticos podem monitorar o ambiente diretamente com seus sensores correspondentes, avaliar os dados com a ajuda de serviços disponíveis em rede e executar tarefas no mundo físico com a ajuda de atuadores.

De acordo com Schlick et al. (2014), sistema físico cibernético (CPS) é o espaço cibernético capaz de realizar a fusão entre o mundo virtual e físico por meio da internet e rede de dados sem precedentes.

Como os sistemas físicos cibernéticos combinam capacidades cibernéticas e físicas, precisamos entender ambos para entender o CPS. No entanto, não é suficiente entender os dois recursos apenas isoladamente, porque também precisamos entender como os elementos cibernéticos e físicos funcionam juntos, ou seja, o que acontece quando eles se conectam e interagem, porque é disso que se tratam os CPS's. (PLATZER, 2018, p.2, tradução nossa)

Os CPS estão sendo cada vez mais utilizados em aplicações industriais e em sistemas que demandam grande capacidade de processamento e análise, tais como *machine learning* e indústria 4.0.

2.3.5.1 Internet das coisas

Para Gilchrist, (2016), a internet industrial é uma derivação da IoT que oferece uma maneira de obter melhor visibilidade e percepção das operações e dos ativos da empresa por meio da integração de sensores de máquina, *software*, sistemas de computação e armazenamento em nuvem. Assim sendo, fornece um método para transformar processos operacionais de negócios usando como *feedback* os resultados obtidos ao se utilizar grandes conjuntos de dados por meio de análises avançadas. Os ganhos de negócios são alcançados através de ganhos de eficiência operacional e produtividade acelerada, o que resulta em redução do tempo de inatividade não planejado e eficiência otimizada, resultando em lucros.

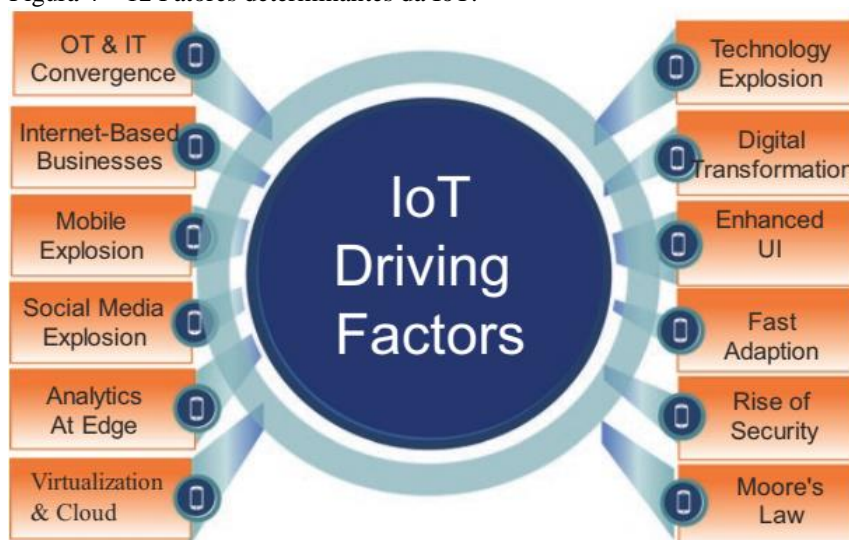
A Internet das coisas, segundo Schlick et al. (2014), estende o conceito da internet clássica, que se limita puramente ao mundo virtual, enquanto a internet das coisas pode ter suas informações oriundas do mundo físico como lugar, posição e condição e destrói a ideia que há uma separação do mundo físico e virtual.

Assumimos que a Internet seja bem conhecida e não tenha nenhuma definição adicional. A questão é o que realmente queremos dizer com 'Coisas'? Bem, as coisas são na verdade 'qualquer coisa' e 'tudo', de eletrodomésticos a edifícios, carros, pessoas, animais, árvores a plantas, etc. Portanto, a IoT em sua forma mais simples pode ser considerada como a interseção da Internet, coisas e dados. (RAYES; SALAM, 2019, p.3, tradução nossa)

“Uma definição mais completa, acreditamos, também deve incluir ‘Padrões’ e ‘Processos’ permitindo que ‘Coisas’ sejam conectadas pela ‘Internet’ para trocar ‘Dados’ usando ‘Padrões’ da indústria que garantem a interoperabilidade e possibilitando úteis e principalmente automatizados ‘Processos’.” (RAYES; SALAM, 2019, p.3, tradução nossa)

Na Figura 4, poderá ser observado os 12 fatores determinantes para impulsionar o crescimento exponencial da Internet das coisas, nos últimos anos.

Figura 4 – 12 Fatores determinantes da IoT.



Fonte: Rayes; Salam (2019).

2.3.5.2 Internet de serviços

As abordagens de Arquitetura Orientada a Serviços (AOS) estão atualmente se adequando aos dispositivos que apresentam processamento e comunicação incorporados. Como resultado, esses serviços obtêm a capacidade de ser hospedados em máquinas de ponta e dispositivos de recursos sem fio, implementados em qualquer equipamento (BALAKRISHNAN; SANGAIAH, 2016).

Para Kangermann (2014), a computação em nuvem cria virtualmente qualquer recurso de TI, aplicativos de *software*, serviços on-line ou até mesmo processos de negócios que podem ser acessados a qualquer momento e eles serão a base para novos serviços inovadores. Infraestruturas de serviços estão surgindo agregando serviços inteligentes para todas as áreas da vida e das redes de negócios.

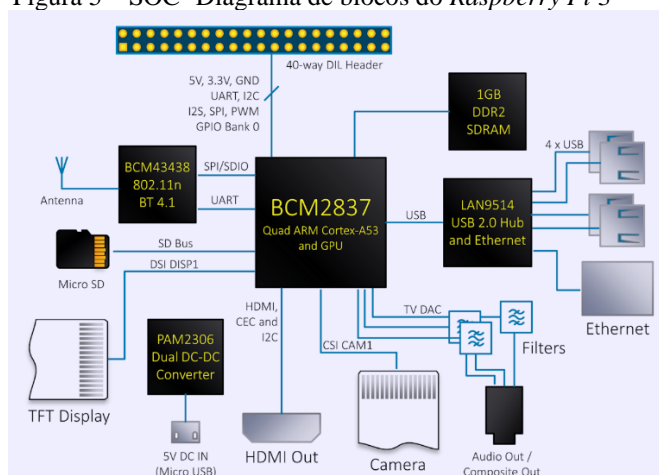
2.4 System-on-Chip

O conceito de *system-on-chip* é relativamente novo, considerando-se que o primeiro componente eletrônico tem aproximadamente 180 anos, na Figura 5 pode-se verificar a construção lógica deste sistema.

Os blocos básicos de construção de equipamentos eletrônicos são circuitos individuais compostos de transistores, resistores, capacitores e, às vezes, transformadores, indutores ou outros componentes. Costumava ser que os circuitos eram feitos de componentes discretos individuais que eram ligados entre si em uma placa de circuito impresso (PCB). Mas hoje, a maioria dos circuitos eletrônicos está na forma de circuito integrado (CI) ou, mais provavelmente, vários circuitos já estão combinados em um único chip de silício para criar um circuito, subsistema ou sistema em um chip. (FRENZEL, 2018, p.63, tradução nossa)

A Figura 5 demonstra o diagrama de blocos do *Raspberry Pi 3*, com os principais blocos.

Figura 5 – SOC- Diagrama de blocos do *Raspberry Pi 3*



Fonte: KPC TEAM (2020)

2.4.1 Arquitetura *Single Board Computers – Raspberry Pi*

Um SBC é um computador completo, ou seja, contém um microprocessador que é construído em uma única placa de circuito com todos os recursos exigidos de um computador funcional, e geralmente com uma interface de entrada/saída, conhecida como *General Purpose Input/Output* (GPIO). Originalmente utilizados como sistemas de demonstração ou de desenvolvimento, para sistemas educacionais ou para uso como controladores de computador embarcados. Atualmente largamente utilizado em projetos *do it yourself*, desde *firewalls* domésticos, sistemas de segurança até *clusters* mineradores de *bitcoins*.

O *Raspberry Pi* é um computador de baixo custo do tamanho de um cartão de crédito que se conecta a um monitor de computador ou TV e usa um teclado e mouse padrão. É um pequeno dispositivo capaz de permitir que pessoas de todas as idades explorem a computação e aprendam a programar em linguagens como Scratch e Python. Ele é capaz de fazer tudo o que você espera que um computador desktop faça, desde navegar na Internet e reproduzir vídeos em alta definição até criar planilhas, processamento de texto e jogos...tem a capacidade de interagir com o mundo exterior e tem sido usado em uma ampla gama de projetos de criadores digitais. (Raspberry Pi Foundation, 2020, tradução nossa)

A Figura 6 demonstra o aspecto físico do *Raspberry Pi*.

Figura 6 – *Raspberry Pi*.



Fonte: RASPBERRY PI (2020)

Por ser um SBC, este computador possui apenas uma fração da capacidade de processamento, armazenagem ou velocidade de um *desktop*, contudo para projetos *do it yourself*, está mais que satisfatório, tendo em vista o seu baixo custo e fácil programação/utilização.

Com a popularização destes computadores, começaram a ser fabricados e comercializados inúmeros acessórios, tais como câmeras, *displays*, *cases* além de vários *shields* com as mais diversas funções.

Atualmente existem centenas de *Single-Board Computers* (SBC's), para citar alguns deles, *DragonBoard*, *Raspberry Pi*, *Tinker Board*, *ODROID*, *Up Board*, *Orange Pi*, *Banana Pi*, *BeagleBone*, Intel NUC, além de outras inúmeras SBM's, tais como Arduino, ESP32, Universal Solder, Linduino One, dentre outras.

2.4.2 Linguagem de programação Python

O Python tem ganhado espaço entre desenvolvedores profissionais, principalmente utilizado para processamento de *Big Data* e mineração de dados, e entre amadores, como *hobby*. A linguagem Python é:

"Uma linguagem de programação interpretada, orientada a objetos e de alto nível com semântica dinâmica. Suas estruturas de dados embutidas de alto nível, combinadas com tipagem dinâmica e vinculação dinâmica, tornam-no muito atraente para o desenvolvimento rápido de aplicativos, bem como para uso como linguagem de script ou cola para conectar componentes existentes. A sintaxe simples e fácil de aprender do Python enfatiza a legibilidade e, portanto, reduz o custo de manutenção do programa. Python oferece suporte a módulos e pacotes, o que incentiva a modularidade do programa e a reutilização de código. O interpretador Python e a extensa biblioteca padrão estão disponíveis na forma de código-fonte ou binário gratuitamente para todas as plataformas principais e podem ser distribuídos gratuitamente." (Sumário Executivo da Python, 2020, tradução nossa)

Esta é a linguagem utilizada para a escrita do *firmware*, que fará a comunicação entre os sensores e o banco de dados.

2.4.3 Linguagem de programação HTML

O *HyperText Markup Language* (HTML) é a linguagem vastamente utilizada para a escrita de páginas Web, aliando-se a outras linguagens, tais como PHP e Java Script, assim fornecendo a estrutura de visualização de conteúdos mais comum.

A definição disponível na Mozilla, (2021) é o código que você usa para estruturar uma página web e seu conteúdo. O HTML não é uma linguagem de programação; é uma linguagem de marcação, usada para definir a estrutura do seu conteúdo. Consiste de uma série de elementos, que você usa para delimitar ou agrupar diferentes partes do conteúdo para que ele apareça ou atue de determinada maneira. As *tags* anexas podem transformar uma palavra ou imagem em um *hiperlink*, pode colocar palavras em itálico, pode aumentar ou diminuir a fonte e assim por diante.

2.4.4 Linguagem de programação PHP

Segundo a documentação disponível no site da PHP, "este é uma linguagem de *script open source* de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML. Em vez de muitos comandos para mostrar

HTML, as páginas PHP contêm HTML em código mesclado que faz "alguma coisa". O código PHP é delimitado pelas instruções de processamento (*tags*) de início e fim `<?php` e `?>` que permitem que você entre e saia do "modo PHP".

O que distingue o PHP de algo como o JavaScript no lado do cliente é que o código é executado no servidor, gerando o HTML que é então enviado para o navegador. O navegador recebe os resultados da execução desse script, mas não sabe qual era o código fonte. Você pode inclusive configurar seu servidor web para processar todos os seus arquivos HTML com o PHP, então ocultando o processo do usuário.

A melhor coisa em usar o PHP é que ele é extremamente simples para um iniciante, mas oferece muitos recursos avançados para um programador profissional. Apesar do desenvolvimento do PHP ser focado nos scripts do lado do servidor, você pode fazer muito mais com ele.

2.4.5 Linguagem de programação SQL

O SQL é a linguagem utilizada em todos os bancos de dados, para efetuar consultas, logo, independente do banco de dados, a estrutura do SQL permanece a mesma.

O *Structured Query Language* (SQL) significa Linguagem Estruturada de Consulta, é utilizada de maneira parecida entre os principais bancos de dados relacionais do mercado. Comumente usada por cientista de dados que usa Python para agregar os dados das diferentes fontes de informações.

Para o site de treinamento Alura, o SQL é uma linguagem padrão para trabalhar com bancos de dados relacionais. Ela é uma linguagem declarativa e que não necessita de profundos conhecimentos de programação para que alguém possa começar a escrever *queries*, as consultas e pedidos, que trazem resultados de acordo com o que você está buscando.

2.4.6 Linguagem de programação Terminal ou *Shell*

Todo o sistema operacional independente do desenvolvedor, possui um terminal, a área gráfica (ou interface do usuário), nada mais é do que um facilitador, para que este não necessite conhecer comandos, pois a interface faz este “tradução”.

A maioria das operações em *Linux/Windows* podem ser feitas usando a linha de comando. Há majoritariamente ferramentas gráficas para a maioria dos programas, embora por vezes não sejam suficientes, tornando-se útil a linha de comando. Até pouco tempo, esta

era a única maneira que o usuário interagiu com o computador, atualmente os usuários avançados preferem o uso do *shell* pode ser mais rápido do que um método gráfico. O uso original do terminal era como um navegador de arquivos, e ainda é usado certamente usado como um navegador de arquivos.

2.4.7 *Firmware*

“*Firmware* é um *software* integrado a um produto de sistema embarcado e armazenado em memória não volátil, como ROM, EPROM, EEPROM ou FLASH. A memória pode estar localizada no chip ou fora do chip. O *firmware* também é conhecido na indústria como “*software* incorporado” ou “*software* de baixo nível”. Os principais componentes do *firmware* incluem opcionalmente um sistema operacional (SO), *kernel*, *drivers* de dispositivo e código do aplicativo. O termo “*firmware*” possui outros significados na indústria que não são usados aqui. Por exemplo, para algumas pessoas, *firmware* refere-se ao microcódigo rodando em processadores que é executado por causa de uma instrução em linguagem *assembly*. Em outras seções da indústria, *firmware* é o conteúdo baixado em um *Field Programmable Gate Array* (FPGA) para programá-lo. O *firmware* pode conter um sistema operacional, pode ser um *Real-Time Operating System* (RTOS). O sistema operacional pode ser um produto comercial, como sistemas embarcados Windows, Linux e VxWorks, ou pode ser desenvolvido internamente. Alguns sistemas embarcados leves não usam um sistema operacional, mas, em vez disso, executam o *firmware* diretamente.” (STRINGHAM, 2010, p.6, tradução nossa)

Para Stringham (2010), a interface *de hardware / firmware* é a junção onde o *hardware* e o *firmware* se encontram e se comunicam. No lado do *hardware*, é uma coleção de registradores endereçáveis que são acessíveis ao *firmware* por meio de leituras e gravações. Isso inclui as interrupções que notificam o *firmware* de eventos. No lado do *firmware*, são os *drivers* de dispositivo ou *software* de baixo nível que controlam o *hardware* gravando valores nos registradores, interpretando as informações lidas dos registradores e respondendo às solicitações de interrupção do *hardware*. Claro, há mais no *hardware* do que registros e interrupções, e mais no *firmware* do que *drivers* de dispositivo, mas esta é a interface entre os dois e onde os engenheiros de ambos os lados devem se preocupar para obter uma integração bem-sucedida.

O *firmware* é gravado permanentemente em um circuito de memória durante o processo de fabricação do componente. Tratando-se de um *software* específico que está associado a um dispositivo de *hardware*. Consiste em um ou mais programas de computador, basicamente são instruções operacionais. São instalados diretamente em um componente de *hardware*, o microchip.

2.4.8 Sistema de Informação Baseado na WEB

É um sistema de informação que usa tecnologias da internet para fornecer informações e serviços aos usuários. Esta tecnologia é um sistema de *software* e é usada para publicar e manter dados pelo princípio do hipertexto. O sistema de informação baseado na Web é a combinação de um ou mais aplicativos da Web, componentes específicos orientados a funcionalidade. Basicamente, neste tipo de sistema de informação, o navegador da web é usado como *front end* (páginas em que o usuário tem acesso) e todos os bancos de dados são usados como *back end* (lógicas, procedimentos e processos desenvolvidos pelo programador).

Estes sistemas evoluíram significativamente nos últimos anos com seu aprimoramento. Os aplicativos baseados na Web têm várias vantagens sobre os aplicativos tradicionais baseados em *software*. A maioria destes aplicativos é compatível em plataformas diferentes do *software* tradicional instalado. O requisito mínimo seria um navegador da web (Internet Explorer, Firefox, Netscape, dentre outros), e é indiferente ao sistema operacional.

O WBIS só precisa ser instalado no servidor, colocando requisitos mínimos na estação de trabalho do usuário final, o que torna o sistema mais fácil de manter e atualizar, já que normalmente tudo pode ser feito no servidor. Os aplicativos baseados na Web podem ser usados por vários usuários ao mesmo tempo. Não é necessário compartilhar a tela ou enviar uma captura de tela quando vários usuários veem e até mesmo editam o mesmo documento ao mesmo tempo.

2.5 Prototipagem Rápida

A prototipagem rápida é uma tecnologia de manufatura aditiva, utilizando largamente o apoio da impressão 3D, na qual o início do processo se dá através da criação de um *design* virtual em *Computer-Aided Design* (CAD).

Para Gorni (2001) o termo prototipagem rápida designa um conjunto de tecnologias usadas para se fabricar objetos físicos diretamente a partir de fontes de dados gerados por sistemas de projeto auxiliado por computador. Tais métodos são bastante peculiares, uma vez

que eles agregam e ligam materiais, camada a camada, de forma a constituir o objeto desejado. Eles oferecem diversas vantagens em muitas aplicações quando comparados aos processos de fabricação clássicos baseados em remoção de material, tais como fresamento ou torneamento. Estes métodos estão em franca expansão, devido a velocidade em que os produtos estão sendo desenvolvidos e lançados ao mercado.

Para a confecção deste protótipo, foram pesquisados vários modelos, entre projetos *do it yourself*, produtos comerciais e transportadores de cereais, abaixo estão selecionadas as Figuras 7, 8, 9, 10 e 11 serviram de inspiração, com partes funcionais ou estéticas.

Figura 7 – Alimentador 1.



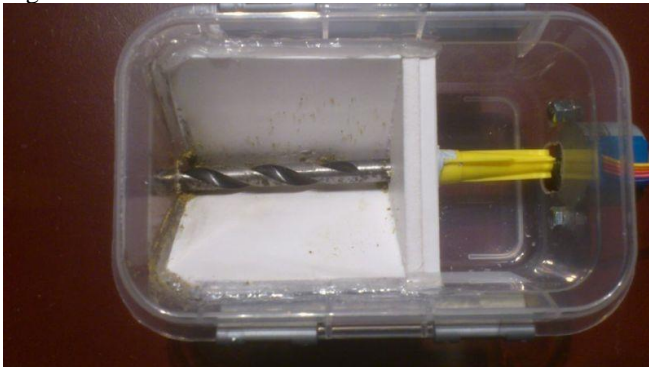
Fonte: DANEKSHEA

Figura 8 – Alimentador 2.



Fonte: KRICHBAUMER (2020)

Figura 9 – Alimentador 3.



Fonte: BREMER (2020)

Figura 10 – Alimentador 4.



Fonte: PETSAFE (2020)

Figura 11 – Alimentador 5.



Fonte: WOPET (2020)

Como pode ser visto nas Figuras 7, 8 e 9, em todos os projetos *do it yourself*, desde os mais simples até os mais completos, seguem a mesma abordagem: um funil com uma rosca helicoidal, para transporte da ração e um funil aonde fica armazenada a ração. Embora a aparência destes projetos possam parecer exóticas, todos se mostraram funcionais segundo as pessoas que reproduziram os projetos, pois vários deles, têm os fontes “STL” liberados para *download* e impressão, sob a licenças *Creative Commons*.

Por outro lado, os produtos comerciais seguem uma linha mais esteticamente definida, porém com valores substancialmente maiores que os projetos *do it yourself*, pelo fato de serem produtos comerciais, não foi possível localizar demonstrações, ou mesmo imagens de suas partes internas, por razões axiomáticas. Entretanto, todos os produtos, funcionam de forma similar, levando a crer, que o processo de funcionamento dos itens comerciais, deva ser muito semelhante aos de código aberto.

3 METODOLOGIA

Este capítulo descreve os procedimentos metodológicos utilizados para a elaboração do projeto proposto. Objetivando os resultados necessários, foram definidas as etapas de funcionamento do protótipo. Em seguida, determinou-se quais partes desse sistema seriam controlados, para assim especificar os dispositivos para esse controle. Com as definições citadas acima, foi elaborado um fluxograma da automação a ser executado. Por fim, foram realizados os testes no código para verificação e funcionamento.

3.1 Descrição do fluxo

Para contextualizar, o dispositivo inicia o sistema operacional ao ser energizado, ao terminar o processo, entrará em modo de execução, aonde através do programa "cron" (que é um agendador de tarefas baseado em tempo em sistemas operacionais de computador semelhantes ao Unix), irá executar os comandos agendados.

Tendo em vista que os parâmetros de operação estarão armazenados em um banco de dados, é necessário que o *firmware* acesse o banco de dados e através de um SQL, busque as informações necessárias, tais como hora de efetuar a alimentação e quantidade de gramas a disponibilizar, para então executar sua lógica, varrendo (*pooling*), os sensores e atuadores, efetuando as operações necessárias. Durante a execução as informações coletadas dos sensores e atuadores são armazenadas no banco de dados, de onde poderão ser visualizadas pelo sistema WEB. As informações coletadas estarão armazenadas na tabela "historico" (em banco de dados, tabela é um conjunto de dados dispostos em número infinito de colunas e número ilimitado de tuplas).

Há 7 tabelas, sendo elas agendamento, atuador, historico, parametro, pet, sensor, usuário. Em cada tabela será gravado informações pertinentes, que unidas dão parâmetros para que o *firmware* opere corretamente. Todas as tabelas podem ser editadas pelo tutor, através do sistema WEB, com a exceção da tabela "historico".

3.1.1 Requisitos da Automação

Todo o produto, seja ele *hardware* ou *software*, foi projetado para atender a dois tipos de demandas: acionistas do negócio e as do usuário/cliente. Este deve oferecer algo ao usuário final que melhore sua experiência, como simplificar tarefas e criar novas oportunidades de interação.

É demasiado simples identificar os resultados desejáveis, porém iniciar um novo projeto é uma tarefa arriscada. A possibilidade de ultrapassar o orçamento, seja por aumento do escopo ou por deficiências descobertas muito tarde na fase de teste, são relativamente altas. E mesmo que o produto chegue às mãos do usuário/cliente a tempo e sem muitos problemas, sempre há a possibilidade de que o usuário/cliente simplesmente não o aceite.

Muitos projetos não devem apenas atender às necessidades dos usuários finais, mas também cumprir as obrigações legais e regulamentações que adicionam um nível extra de complexidade ao desafio ao projeto. A definição de requisitos deve ajudar o projeto a ter um mapa claro.

Quando é possível ver o quadro geral e os detalhes finos juntos, fica mais simples a contextualização de todos os pontos a serem desenvolvidos em um projeto.

Então para este protótipo foi proposto 04 escopos:

- *Hardware*: Neste ponto é contemplado a interface e o *Raspberry Pi*. A interface é responsável pela comunicação entre os sensores e atuadores ao *Raspberry Pi*.
- *Software*: É atendida, toda a infraestrutura de armazenagem de dados, comunicação de rede e com o tutor.
- *Firmware*: Está contemplada, toda a programação que irá efetuar o monitoramento dos sensores e comando dos atuadores.
- Protótipo: É considerada toda impressão 3D do produto.

3.2 Descrição do sistema proposto

O produto/protótipo é um alimentador automático/remoto de animais domésticos, fornecendo alimento seguindo agendamento do tutor. Os agendamentos são livres, e no Quadro 1, é possível verificar quais são as restrições, modo de operação, e construção física do protótipo.

Quadro 1 – Descrição/Parâmetros para o sistema proposto.

Usabilidade	Físico	Restrições	Modo de Operação
Ser de fácil uso (configuração) pelo tutor e pelo <i>pet</i> ;	Reservatório de Água e Ração	Somente será disponibilizado alimentador (funil) para ração seca granulada (pelo fato de ração úmida / ração úmida caseira não poder ficar muito tempo aberta e/ou necessitar de refrigeração)	O tutor deverá logar na interface web;
Ser de fácil higienização;	Tigela para Água e Ração	É obrigatório o dispositivo estar ligado à rede LAN da casa, para poder efetuar a comunicação com o(s) dispositivo(s), caso não esteja ligado, o sistema irá operar somente em modo automático, após a configuração dos horários pelo tutor.	Definir um usuário para si; (opcional)
Fácil reposição de ração e água;	Rosca helicoidal para transporte da ração	A tigela de água será reabastecida por gravidade, não sendo necessários sensores, válvulas e afins para dosagem.	Definir o tipo do <i>pet</i> ;
=	Sensor de comida na tigela	Não é a prova d'água	Definir Agendamentos.
=	Indicador de falta de ração	Cada agendamento terá somente a possibilidade de definição de uma quantidade em gramas da ração.	Consultar Relatórios
=	Indicador de falta de água	É obrigatório o dispositivo estar ligado à rede elétrica.	

Fonte: Elaborado pelo Autor

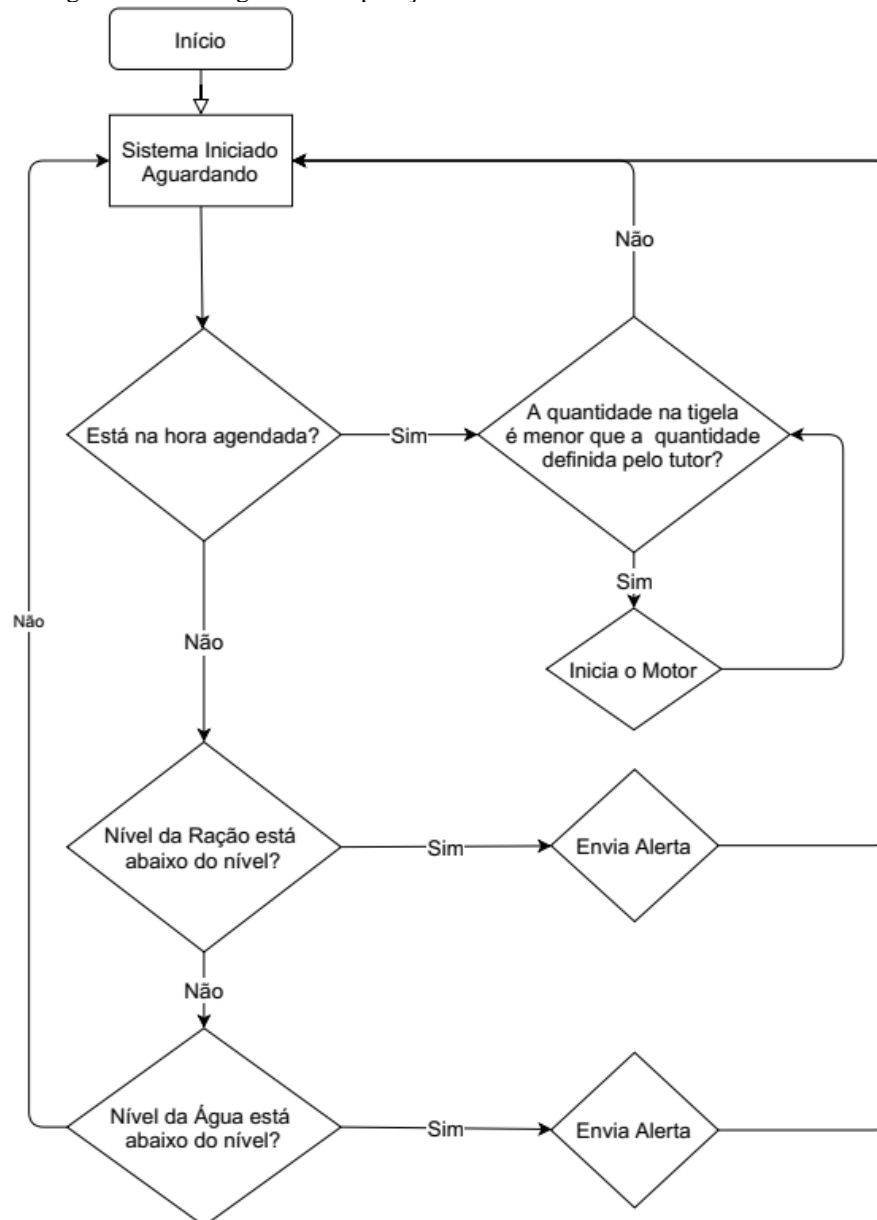
3.2.1 Fluxograma da automação

O fluxograma apresentado na Figura 12 é uma representação gráfica que descreve os passos e etapas sequenciais de um determinado processo.

Como o processo descrito no item “Descrição do Fluxo” finalizado, inicia-se a rotina normal, que consiste em:

- Sistema aguardando: O sistema ficará aguardando o momento da próxima execução, o agendamento será executado a cada minuto.
- Hora agendada: quando o "cron", executar e estiver na hora agendada, será verificado o peso (quantidade de ração), disponível na tigela. Caso seja maior, o sistema não irá disponibilizar mais ração. Caso seja menor, o sistema irá acionar o motor do transportador, que ao girar, irá movimentar a rosca helicoidal, com este movimento. O alimento será transportado da tremonha contendora, para a tigela. Quando a quantidade de ração for maior ou igual, o motor será desligado, finalizando o processo.
- Nível da ração baixo: Quando o nível da ração estiver abaixo será enviado um e-mail informando. Um e-mail deverá ser cadastrado.
- Nível da água baixo: Quando o nível da água estiver abaixo será enviado um e-mail informando. Um e-mail deverá ser cadastrado.

Figura 12 – Fluxograma da Operação.



Fonte: Elaborado pelo autor.

3.2.2 Infraestrutura de Gerenciamento

Para este protótipo foram usadas duas interfaces. A primeira é o *Raspberry Pi*, com os *softwares* conhecidos como Linux, Apache, MySQL, PHP (LAMP) e o *firmware*, que gerencia o barramento GPIO, para a comunicação com os sensores/elementos. Já a segunda é uma interface "intermediária" que receberá os comandos provenientes do *firmware* (oriundas do *Raspberry Pi*) e controla os sensores/elementos, bem como receberá sinais dos sensores/elementos e tratará os sinais destinados aos pinos do GPIO.

O *firmware* foi escrito em Python, utilizando-se de bibliotecas internas da linguagem Python e em especial uma biblioteca obtida na internet que emula o chip "HX711", para que

seja possível a interpretação do sinal enviado por este chip ao *firmware*. A biblioteca foi obtida no repositório “GitHub”, do desenvolvedor nomeado “Tatobari” (TATOBARI, 2020)

Na Figura 13, é demonstrado os símbolos do *RaspberryOS*, Python e as portas GPIO.

Figura 13 – Simbologia *RaspberryOS*, GPIO e Python.

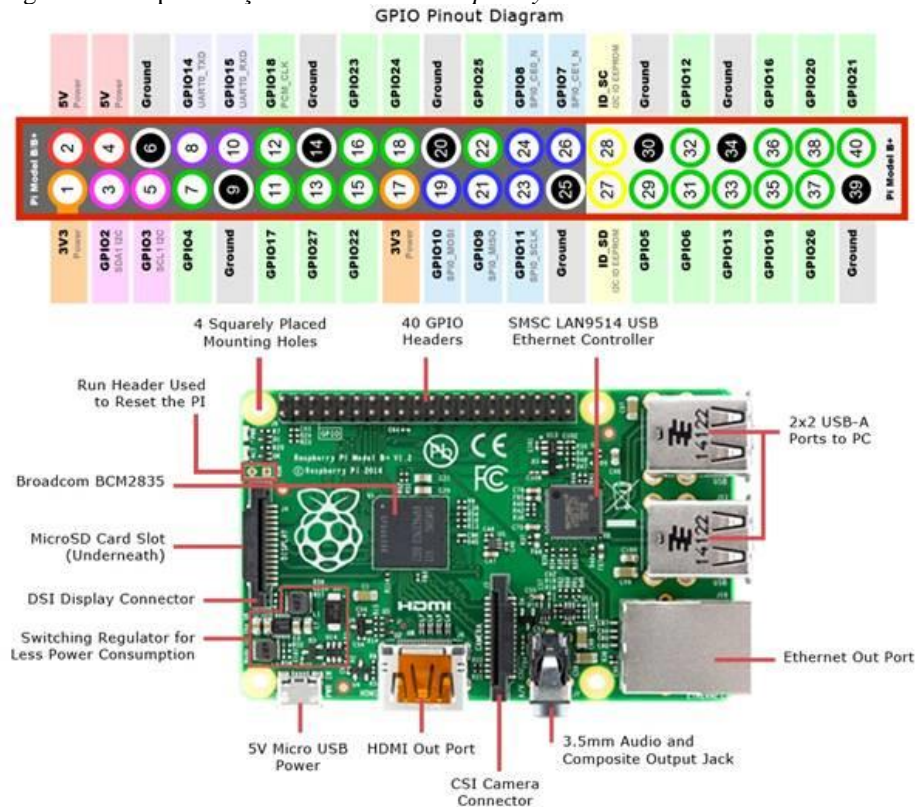


Fonte: BUENO (2020)

3.2.3 SBC – *Raspberry Pi*

A definição do *Raspberry Pi* foi embasada no suporte de vários sistemas operacionais disponíveis, bem como uma comunidade gigantesca com habilidade técnica na utilização deste SBC. Ainda, por ser um distro Linux, a manutenção é extremamente parecida a um *desktop* comum usando uma distribuição Ubuntu, não sendo necessário infindáveis horas de treinamento a um técnico. Por tratar-se de um sistema operacional, é deveras simples desenvolver programas/funções que podem ser rapidamente/facilmente atualizados ou corrigidos “*on-the-fly*”, sem a necessidade de que o produto seja enviado a uma assistência técnica.

Com relação a custos, a versão padrão do *Raspberry Pi* custa USD35.00, enquanto a versão padrão do Arduino custa USD20.00, sem nenhuma das interfaces de rede. Destaca-se algumas especificações técnicas do *Raspberry Pi 2*: Model B V1.1; CPU: Broadcom BCM2836 900MHz quad-core ARM Cortex-A7 processor; RAM: 1 GB SDRAM; Portas USB: 4 USB 2.0 *ports*; *Network*: 10/100 Mbit/s *Ethernet*; Classificações de potência: 600 mA (3.0 W); Fonte de energia: 5V Micro USB); Tamanho: 85.60 mm × 56.5 mm; Peso: 45 g. A especificação dos pinos e os componentes gerias do *Raspberry Pi 2* são apresentados na Figura 14.

Figura 14 – Especificação de *Hardware Raspberry Pi*.

Fonte: WATELECTRONICS (2020)

3.2.4 Sensor de peso – balança da tigela

O sensor de peso é o responsável por detectar a presença de ração (e a quantidade) que ainda está disponível na tigela para questões de registro. Além disso, tem a função de proporcionar o depósito de somente o que está faltando para a quantidade pré-definida pelo tutor. As especificações técnicas do sensor definido, Figura 15, são: Modelo: Sensor de Peso 50Kg Célula de Carga; Módulo HX711 (conversor); Tensão de operação: 4,8 à 5,5V DC; Corrente de operação: 1,6mA; Temperatura de operação: -20 à 85°C; Interface SPI; Dimensões: 29 x 17 x 4mm (sem os pinos);

Figura 15 – Célula de carga.



Fonte: FILIPEFLOP (2020)

3.2.5 Sensor de distância ultrassônico – água e ração

Inicialmente o sensor a ser utilizado era um sensor de proximidade, contudo no decorrer do projeto, identificou-se um sensor de distância melhor e muito mais barato, o sensor HC-SR04.

O Sensor de Distância Ultrassônico HC-SR04 é capaz de medir distâncias de 2 cm a 4 m com precisão. É um módulo possuindo um circuito pronto com emissor e receptor acoplados (Figura 16). O sensor de distância ultrassônico é responsável por detectar o nível da ração. As especificações técnicas do sensor utilizado são: Tensão de operação: 5VDC; Corrente de operação: 2mA; Alcance: 2cm ~ 4m; Ângulo de efeito: 15°; Precisão: 3mm

Figura 16 – Sensor de distância ultrassônico HC-SR04.

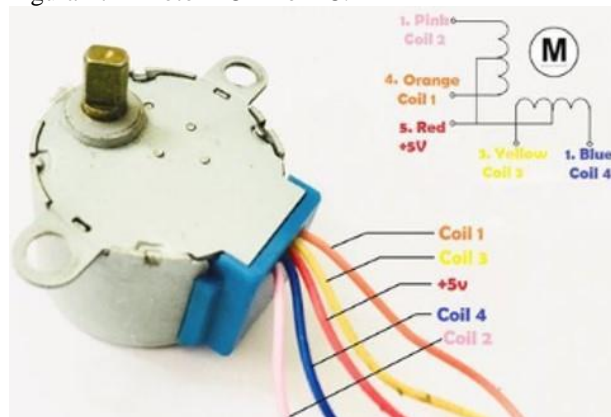


Fonte: FILIPEFLOP (2020)

3.2.6 Motor de passo – transportadora

O Motor de passo é acoplado a rosca helicoidal, que ao ser acionado transportará a ração do reservatório ao canal que levará a tigela de ração, Figura 17. As especificações técnicas são: Modelo: 28BYJ-48; Driver: ULN2003; Tensão: 5VDC; Número de fase: 4; Número de vias: 5; Caixa de Redução: 1/64; Diâmetro do eixo: 5mm; Ângulo do Passo: $5,625^\circ/64 \sim 0,088^\circ$; Frequência: 100Hz; Resistência DC: $50\Omega \pm 7\%(25^\circ\text{C})$; Torque: 34,3 mN.m; Peso: 40g.

Figura 17 – Motor 28BYJ-48.



Fonte: MICROCONTROLLERSLAB (2020)

3.3 Descrição do Projeto

O alimentador consiste em uma tremonha contidora de ração e um recipiente para água. Na base da tremonha, está instalado uma rosca helicoidal, que ao ser girada, por meio do motor de passo, desloca o alimento até o fim do transportador, de onde cairá no recipiente para alimentar o *pet*.

O motor é acionado por um controlador de passo, não deixando o motor sempre energizado. O controlador de passo é controlado pelo SBC *Raspberry Pi* que possui um servidor web.

Neste servidor há um *software* de gerenciamento de banco de dados, aonde são configuradas as informações relativas aos parâmetros de controle do usuário, tais como: agendamentos, quantidade de ração a ser fornecida e dados do *pet*.

Do gerenciador as informações são convertidas para comandos que o *Raspberry Pi* possa interpretar, e comandar os dispositivos, monitorar os sensores e executar os comandos.

3.3.1 Optoacoplador - CI PC817

É um circuito integrado de apenas um canal isolado. É muito utilizado por ter uma isolamento interna entre emissor e receptor de 5 kV e ainda possuir grande imunidade a ruídos e interferências eletromagnéticas. Possui alta frequência de comutação e a segurança de acionamento, pois seu LED interno produz luz infravermelha e somente esse tipo de feixe luminoso pode disparar o fototransistor.

Utilizando o efeito fotoelétrico, através de um feixe de luz infravermelha (LED IR), polariza a base do fototransistor, forçando a condução entre base e emissor. Com esta operação, é consumida a tensão base-emissor a 0,7V, saturação a base do transistor, permitindo a passagem de corrente através de seu coletor e emissor, assim acionando a carga.

Ele é utilizado para efetuar a comunicação entre o SBC e a interface de força, assim isolando a parte de comando lógico da parte de comando de força, para que caso algum problema ocorra não ocorram danos ao SBC.

3.3.2 Circuito Integrado - CI ULN2003A

O circuito integrado ULN2003A é normalmente utilizado em circuitos digitais na tecnologia TTL e CMOS que operam com sinais de entrada e saída de 5 e 12V em corrente contínua, possuindo uma matriz de sete pares de transistores *Negativo Positivo Negativo* (NPN) na configuração de *Darlington* de forma independente, que fornecem um ganho de saída capaz de controlar dispositivos com corrente máxima de até 500mA.

Eles são compostos de dezesseis terminais, oito de cada lado, sendo eles sete entradas e sete saídas diretas, o terminal GND e o terminal comum, na parte interna possuem diodos de supressão que protegem o CI de correntes reversas fornecidas a partir de circuitos indutivos, dispensando o uso de diodos externos como proteção. Possuem também um resistor de 2,7 k Ω em série com cada entrada para trabalhar com circuitos *Transistor-Transistor Logic* (TTL) e *Complementary Metal-Oxide Semiconductor* (CMOS).

Sua aplicação mais comum é o controle de motores de passo, servindo de interface entre o motor e o controlador. Para ligação das cargas ao componente, por se tratar de saídas com transistores NPN, faz-se necessário interligar o negativo de cada dispositivo nas saídas correspondentes do ULN2003A pois o mesmo irá chavear o negativo. O positivo de cada dispositivo é interligado ao positivo da fonte externa, que por sua vez é interligado ao terminal comum.

3.3.3 Amplificador Operacional – CI HX711

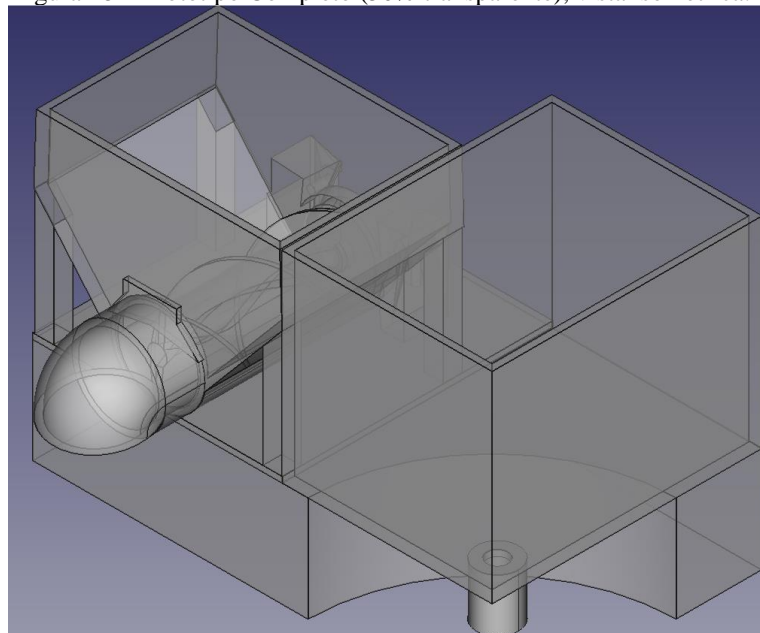
É um módulo amplificador operacional de 24 bits utilizado para converter o sinal analógico em digital, além de amplificador de saída. Ele foi criado com a intenção de atuar com sensores de peso ligados em ponte, sendo que o multiplexador interno seleciona a entrada diferencial para o amplificador de ganho.

3.3.4 Descrição do Protótipo

O protótipo foi desenhado em dois *softwares* de CAD, NX11 e FreeCAD. Está separado em três blocos. No bloco 1, localiza-se a área aonde serão instalados o *Raspberry Pi* e a interface de potência. O bloco 2 é o tremonha e a rosca helicoidal, e no bloco 3 aonde está o reservatório de água.

Pode ser visto na Figura 18, o projeto unido, e nas próximas figuras os blocos. Na Figura 20 é possível ver a vista isométrica do protótipo completo, estando esse com 50% de transparência para facilitar a visualização de toda estrutura.

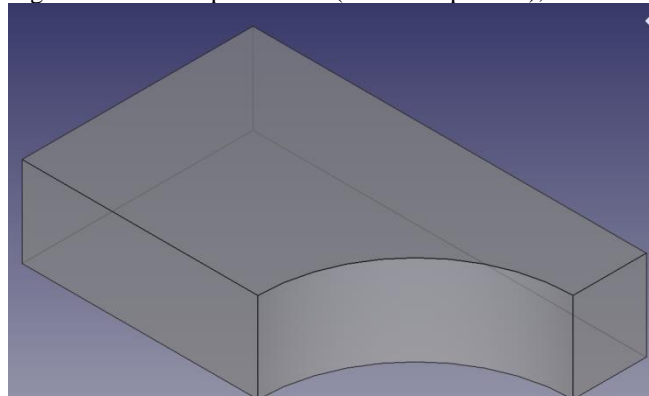
Figura 18 – Protótipo Completo (50% transparente), vista isométrica.



Fonte: Elaborado pelo autor.

A Figura 19 mostra o bloco 1, em vista isométrica, com 50% de transparência, área aonde é instalado o *Raspberry Pi* e a interface de potência.

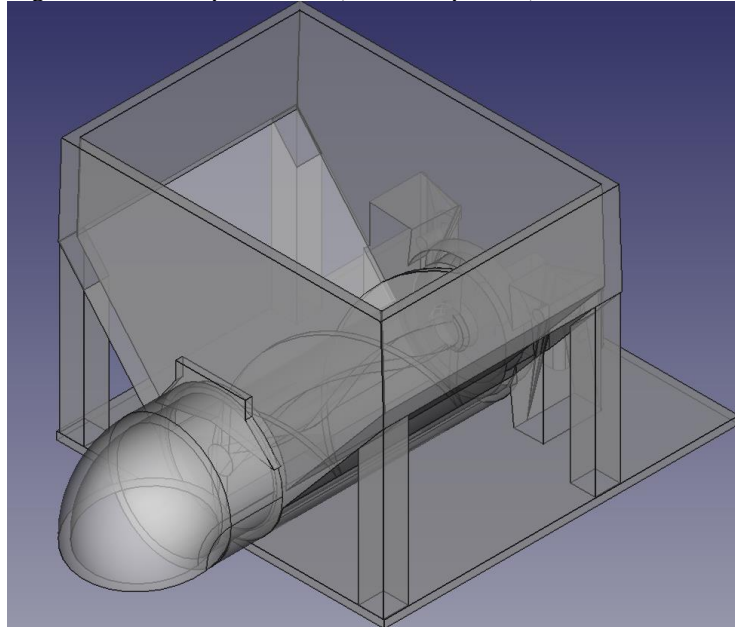
Figura 19 – Protótipo Bloco 1 (50% transparente), vista isométrica.



Fonte: Elaborado pelo autor.

A Figura 20 evidencia o bloco 2, em vista isométrica, com 50% de transparência, detalhando a tremonha e a rosca helicoidal, local aonde será depositada a ração.

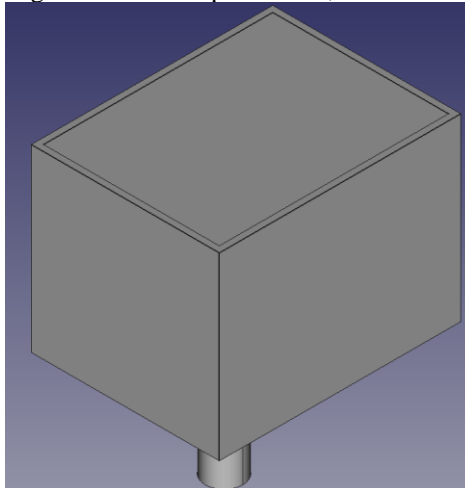
Figura 20 – Protótipo Bloco 2 (50% transparente), vista isométrica.



Fonte: Elaborado pelo autor.

A Figura 21 apresenta o bloco 3, em vista isométrica, detalhando o reservatório de água, com capacidade de aproximadamente 550ml.

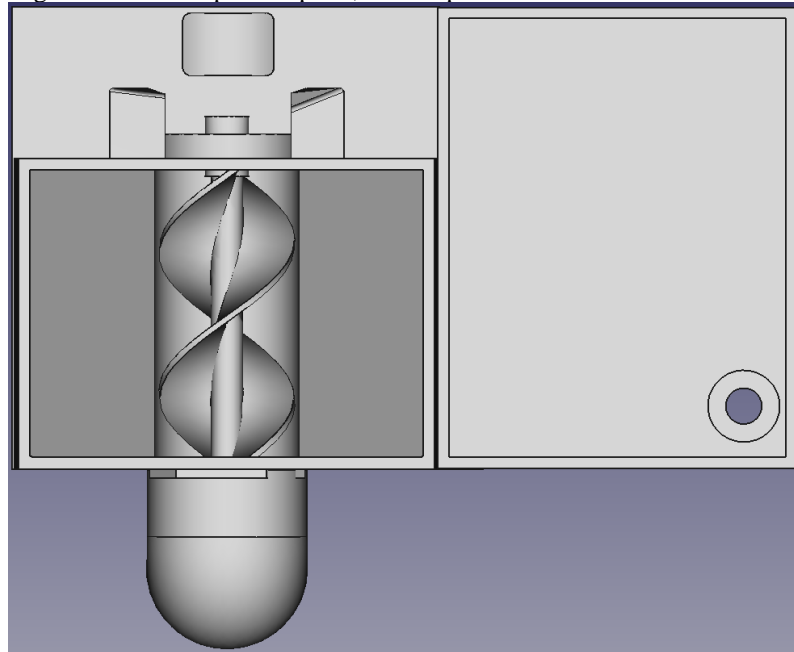
Figura 21 – Protótipo Bloco 3, vista isométrica.



Fonte: Elaborado pelo autor.

A Figura 22, mostra os blocos 2 e 3, em vista superior, sem transparência, detalhando a tremonha, rosca helicoidal e reservatório de água.

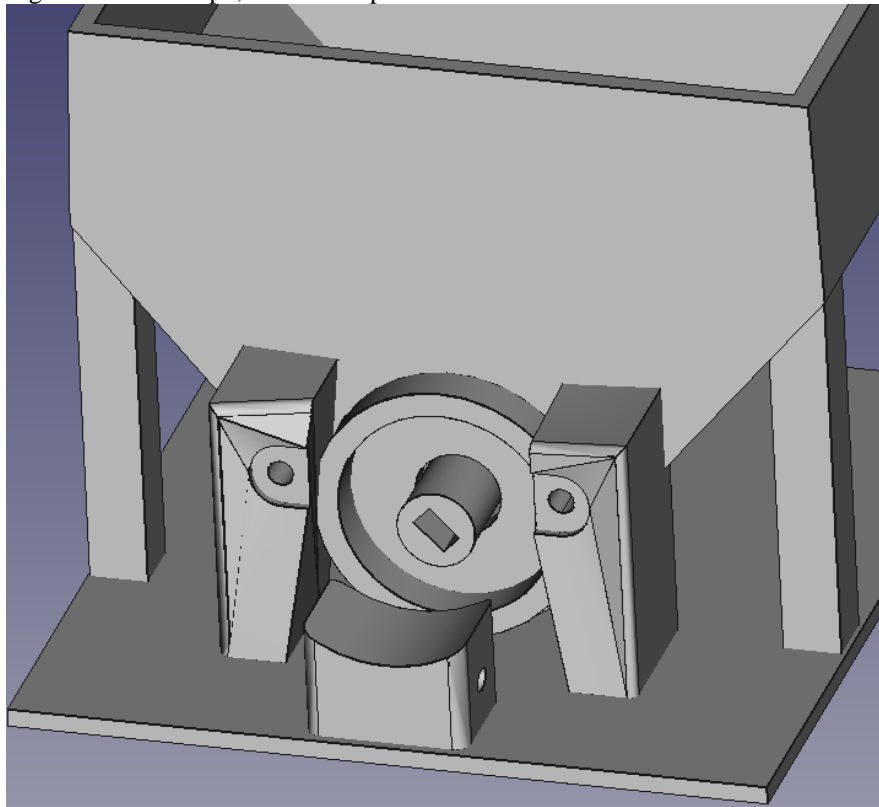
Figura 22 – Protótipo Completo, vista superior.



Fonte: Elaborado pelo autor.

A Figura 23, apresenta o bloco 2, em vista posterior, sem transparência, detalhando a tremonha, acoplamento do motor de passo à rosca helicoidal e suporte do motor de passo.

Figura 23 – Protótipo, detalhe acoplamento motor a rosca helicoidal.



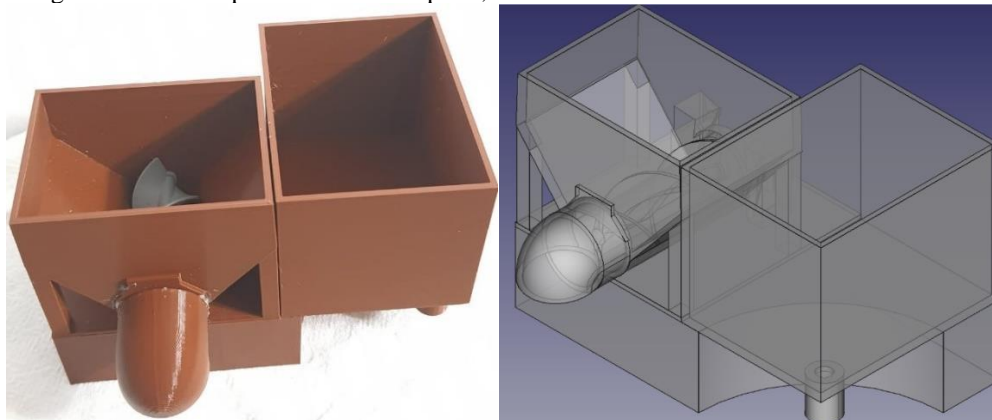
Fonte: Elaborado pelo autor.

3.4 Impressão do Protótipo

A impressão do protótipo foi feita em escala 1:4, tendo em vista seu caráter experimental e de análise da estrutura. Assim fica evidenciado que a estrutura em escala 1:1 também funcionará como planejado.

Pode ser visto na Figura 24, uma foto do protótipo impresso, sem os componentes eletrônicos instalados.

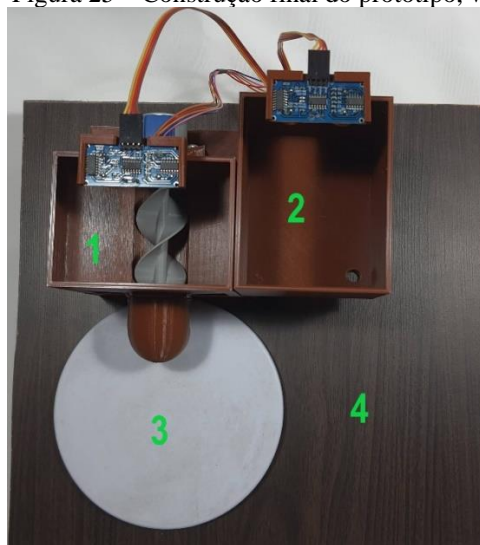
Figura 24 – Protótipo em escala Completo, vista isométrica.



Fonte: Elaborado pelo autor.

A Figura 25 evidencia a construção final. A numeração informa a função de cada parte do protótipo. Sendo: (1) funil e tremonha (nesta área ficará disponível a ração); (2) reservatório de água (funcionará com uma caixa d'água, que abastecerá a tigela); (3) suporte para tigela de ração (abaixo da estrutura branca está fixada a célula de carga, efetuando a pesagem da ração disponível para o *pet*); e (4) base de madeira aonde toda a estrutura está fixada.

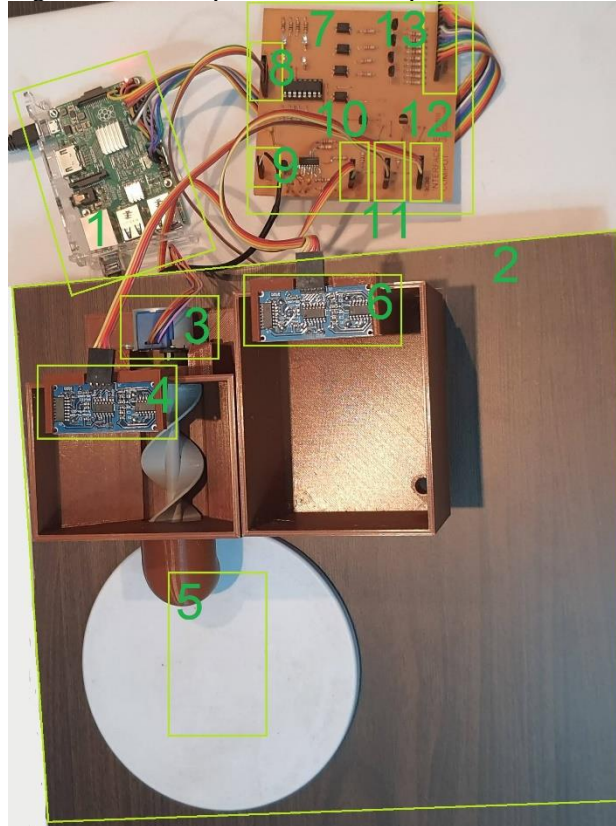
Figura 25 – Construção final do protótipo, vista superior.



Fonte: Elaborado pelo autor.

A Figura 26 demonstra o produto com toda a estrutura, interface e controle instalados.

Figura 26 – Protótipo em escala Completo, funcional, vista superior.



Fonte: Elaborado pelo autor.

A numeração delimitada por retângulos na Figura 26 representa:

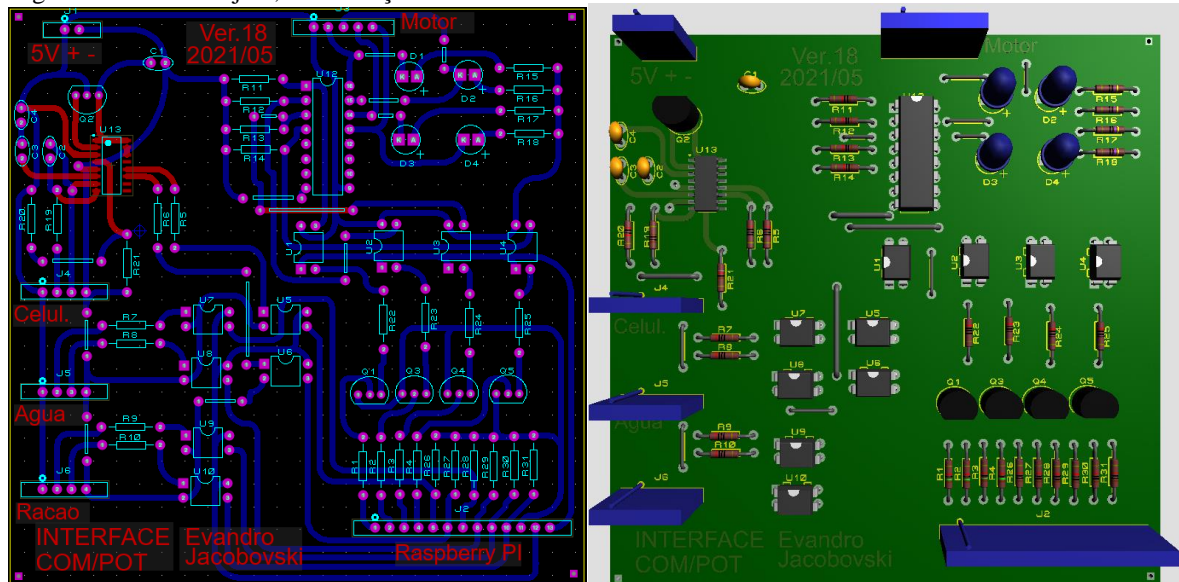
- 1: *Raspberry Pi*
- 2: Base em madeira do protótipo
- 3: Motor de passo instalado
- 4: Sensor ultrassônico HC-SR04 da ração
- 5: Localização da célula de carga
- 6: Sensor ultrassônico HC-SR04 da água
- 7: Interface comando/potência
- 8: Cabo flat de comando do motor de passo
- 9: Cabo de alimentação 5 V
- 10: Cabo flat de comando da célula de carga
- 11: Cabo flat de comando do sensor da água
- 12: Cabo flat de comando do sensor da ração
- 13: Cabo flat de comunicação com o *Raspberry Pi*

Como apresentado na Figura 26, aonde é demonstrado o protótipo em escala funcional, julga-se que o produto está adequado a proposta inicial.

3.5 Printed Circuit Board - Interface de Comando/Potência

A interface de comando/potência, vista na Figura 27 apresenta o projeto (esquerda), visualização em 3D (direita), é o resultado do projeto eletrônico desenvolvido para gerenciar os sensores e atuadores especificados. Assim, o circuito gerencia os comandos provenientes do *Raspberry Pi*, interpreta os sinais enviados pelos sensores, e executa os comandos enviando-os aos atuadores, desta forma mantendo o dispositivo operacional.

Figura 27 – PCB Projeto, visualização em 3D.



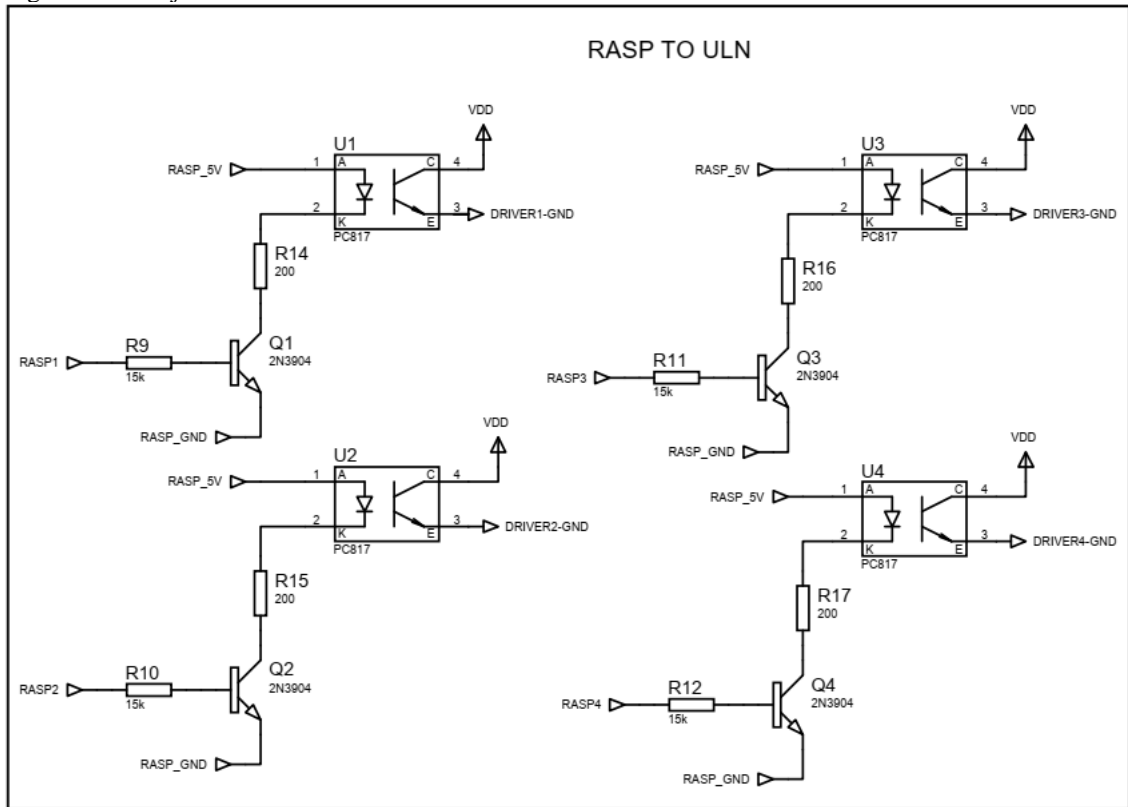
Fonte: Elaborado pelo Autor.

3.6 Projeto Interface de Potência

O projeto possui uma interface de potência. Esta interface que faz a comunicação de dados e alimentação dos sensores e atuadores, capturando as informações. Ao capturar, essas informações são transmitidas às portas GPIO do *Raspberry Pi*, que fará o tratamento via *firmware*. As Figuras 28, 29 e 30 apresentam o projeto eletrônico que foi elaborado e simulado do no *Proteus*.

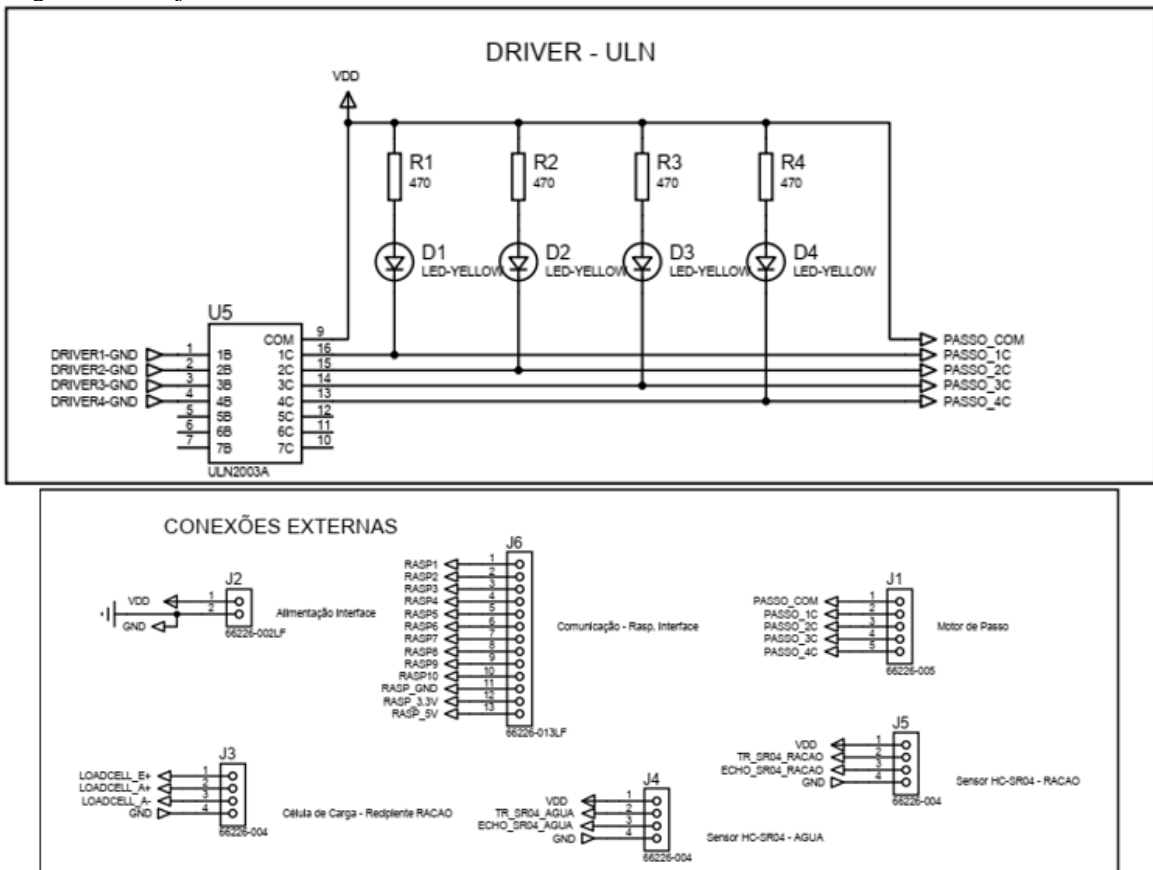
O projeto está dividido nas 3 figuras citadas acima, para uma melhor visualização de sua construção, sendo eles, detalhe isolamento comando motor, CI driver e conexões externas e CI célula de carga e sensor ultrassônico.

Figura 28 – Projeto interface Comando/Potência – Detalhe isolamento comando motor.



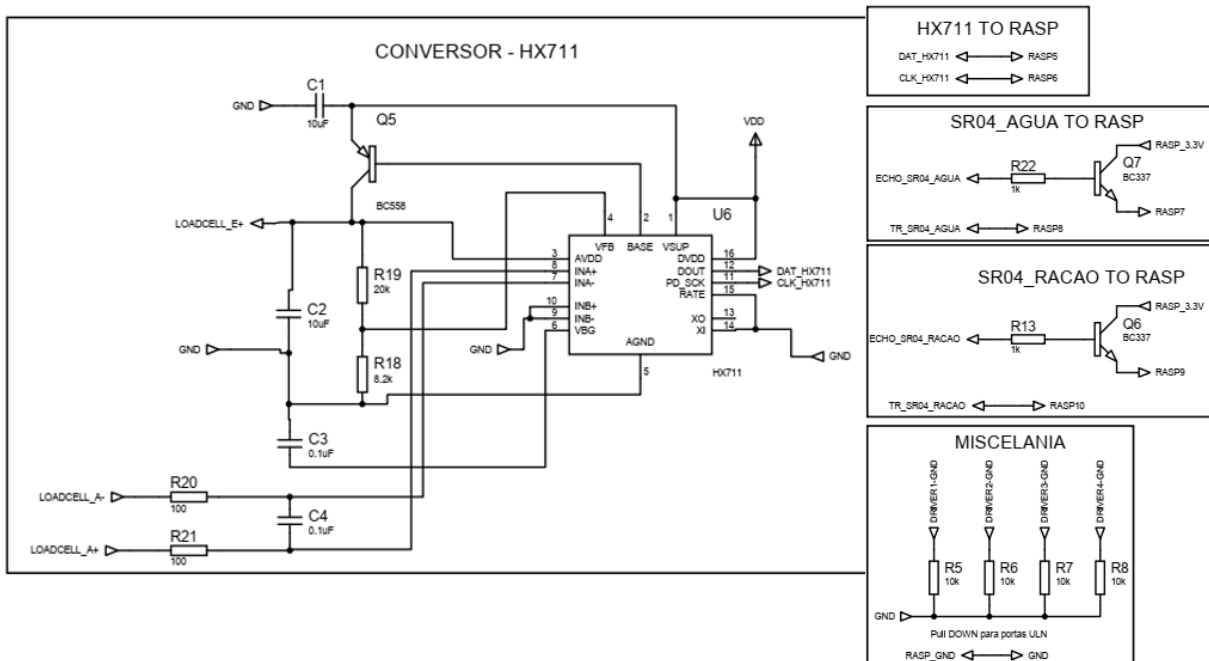
Fonte: Elaborado pelo Autor.

Figura 29 – Projeto interface Comando/Potência – CI Driver e Conexões externas.



Fonte: Elaborado pelo Autor.

Figura 30 – Projeto interface Comando/Potência – CI Célula de carga e sensor ultrassônico.



Fonte: Elaborado pelo Autor.

Os CI's PC817 foram usados como proteção entre o circuito de comando e o circuito de potência. A isolação proporcionada pelo CI impedirá que qualquer problema relativo ao motor (e seu CI de comando), danifique as portas GPIO do *Raspberry Pi*.

Todos os resistores alocados entre os dispositivos e os componentes tem também o objetivo de proteção. Os resistores de 200Ω , entre os CI's (PC817) e o *Raspberry Pi*, estão assim definidos, pois as portas do GPIO, disponibilizam apenas 16mA de corrente e 3,3V de tensão em CC.

O circuito integrado ULN2003A possui uma característica de matriz de sete pares de transistores NPN na configuração de *Darlington* de forma independente, que fornecem um ganho de saída capaz de controlar dispositivos. É o CI utilizado para o controle de motor de passo, servindo de interface entre o motor e o controlador. Para ligação das cargas ao componente, faz-se necessário interligar o negativo de cada dispositivo nas saídas correspondentes, pois o mesmo irá chavear o negativo. O positivo de cada dispositivo é interligado ao positivo da fonte externa, que por sua vez é interligado ao terminal comum.

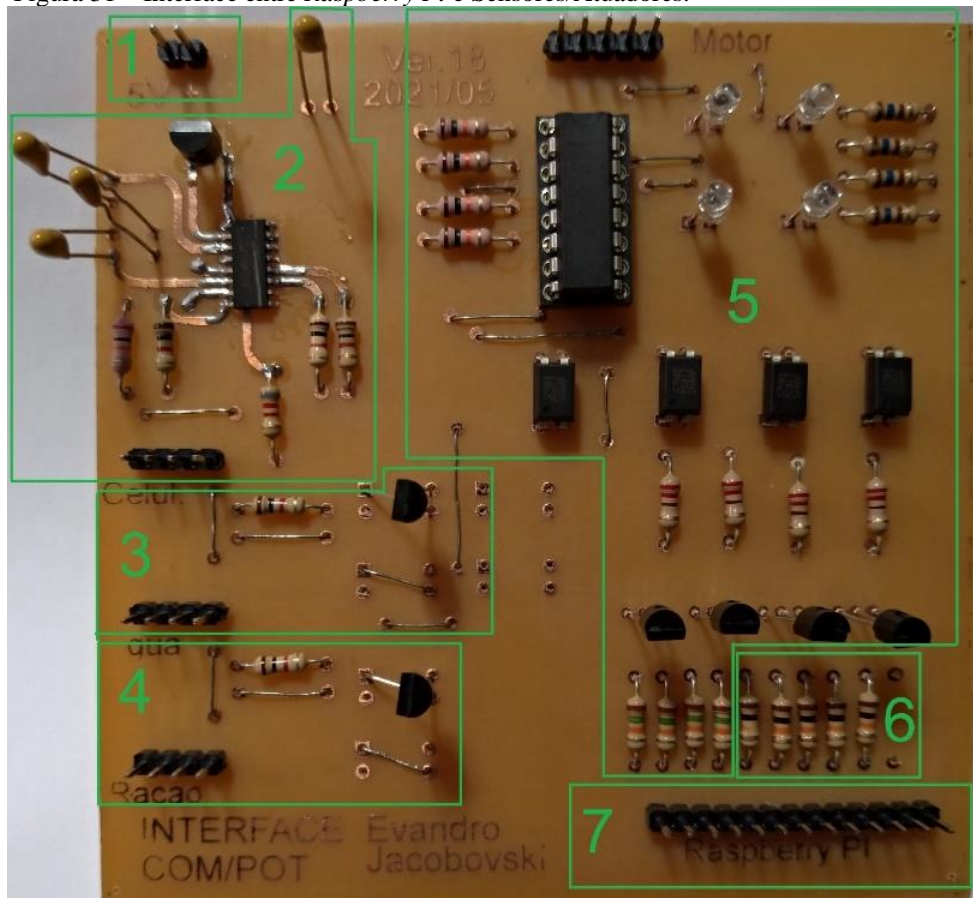
Utilizando o CI ULN2003A, o motor de passo é digitalmente controlado, trabalhando na forma unipolar. Construtivamente se caracteriza por possuir um fio (*center-tape*) entre o enrolamento de suas bobinas, sendo este o terminal positivo, aonde o motor é controlado aterrando-se as extremidades dos enrolamentos. A alimentação deve ser na forma sequencial e

repetida, bobina a bobina, então cada passo corresponderá a $1,8^\circ$, tendo como vantagem o aumento do torque do motor.

O circuito integrado HX711 é um amplificador operacional de 24 bits utilizado para converter o sinal analógico em digital, além de amplificador de saída. Atua como controlador dos sensores de peso ligados em ponte, sendo que o multiplexador interno seleciona a entrada diferencial para o amplificador de ganho.

A Figura 31 apresenta a *Printed Circuit Board* (PCB) discriminando através de retângulos cada uma de suas partes. 1: pinos de alimentação. 2: CI HX711, seus componentes auxiliares e pinos de comunicação com a célula de carga. 3: pinos de comunicação com o sensor ultrassônico medidor de água, e componentes auxiliares à comunicação. 4: pinos de comunicação com o sensor ultrassônico medidor de ração, e componentes auxiliares à comunicação. 5: CI ULN2003A, componentes auxiliares, LED's de indicação de operação, optoacopladores. 6: resistores de *pull-down*. 7: interface de comunicação entre comando e placa de potência.

Figura 31 – Interface entre *Raspberry Pi* e Sensores/Atuadores.



Fonte: Elaborado pelo Autor.

3.7 Descrição do Funcionamento

A descrição de funcionamento e a função dos componentes utilizados na construção da interface de potência/comando é explanada neste tópico.

3.7.1 Sensor Ultrassônico HC-SR04

O sensor ultrassônico é um dispositivo que utiliza o som para medir distância até um obstáculo, inspirado no "sonares" dos morcegos. Este sensor utiliza ondas sonoras de frequência acima da faixa audível do ser humano. É constituído de dois elementos principais, um transmissor e um receptor de som, posicionados lado a lado. Ao ativa-lo, o transmissor produz uma série de pulsos sonoros que viajam pelo espaço entre o sonar e um obstáculo na velocidade do som (normalmente o meio percorrido é o ar). Então quando as ondas sonoras atingem um obstáculo, parte delas sofre reflexão, essas ondas refletidas são então captadas pelo receptor. A diferença de tempo entre a transmissão dos pulsos sonoros e a captação pelo receptor é usada para calcular a distância até o objeto. A equação (1) é usada para obter a distância (d), em metros, a partir da velocidade do som (v_s) em metros por segundo (será utilizado a v_s igual a 344 m/s amplamente utiliza à 25 °C), e do seu tempo de ida e volta (Δt), em segundos:

$$d = \frac{1}{2} v_s \Delta t \quad (1)$$

O sensor ultrassônico HC-SR04, possui 4 pinos: VCC, GND, TRIG, ECHO e exige uma alimentação de 5 V. TRIG é o pino de ativação do transmissor, que ao recebe um pulso de 10 microssegundos, ativa o transmissor do módulo. O pino ECHO vai para o nível lógico alto logo após o transmissor finalizar o envio de seus pulsos ultrassônicos e permanece em alto até o momento em que o receptor capta as ondas refletidas por um obstáculo. O intervalo em que ECHO está em nível lógico alto é o tempo de ida e volta dos pulsos ultrassônicos.

Os pinos digitais do *Raspberry Pi* trabalham com níveis lógicos de 0 a 3.3 V. Assim, não se pode conectar diretamente o pino ECHO a um pino de entrada do *Raspberry Pi*, pois quando estes pinos forem ativados pelo módulo disponibilizarão 5 V, danificando o GPIO sendo este um dos motivos da conexão ser intermediar por CI PC817.

O sensor consome até 15 mA de corrente, e em sua saída "Echo", disponibiliza uma corrente muito menor, então foi elaborado um circuito para limitar esta corrente,

O resistor, está sendo usado para limitar a corrente do pino "Echo" ao máximo de 15 mA e aplica-la a base do transistor a equação (2) apresenta o cálculo da tensão que estará sob o resistor na base. Sendo 5 V a tensão do pino "Echo" e 0,7 V a queda de tensão na base do transistor.

$$V_r = 5 - 0,7 = 4,3 \text{ V} \quad (2)$$

Após obter a tensão que será aplicada, é necessário localizar no datasheet do transistor o ganho (β). Neste caso será de aproximadamente 100. Aplicando a equação (3), obtém-se a corrente de base necessário para acionar o transistor.

$$I_B = \frac{I_C}{\beta} \quad (3)$$

$$I_B = \frac{0,015}{100} = 0,000150 \text{ A} \quad (4)$$

Portanto será consumido efetivamente 150uA. Assim, é possível calcular a resistência que deverá ser conectada a base do transistor, conforme evidenciado na equação (6).

$$R = \frac{V}{I} \quad (5)$$

$$R = \frac{4,3}{0,000150} = 28.666,67 \Omega \quad (6)$$

Entretanto em seu datasheet, a corrente de base pode ser de até 100mA, então foi utilizado um resistor de 1k Ω , facilmente encontrado, e que atende os parâmetros, conforme equação (7).

$$I = \frac{4,3}{1000} = 0,0043 \text{ A} \quad (7)$$

No coletor deste transistor, está aplicada a tensão de 3,3 V proveniente do *Raspberry Pi*, e seu emissor conectado a um GPIO de comando, quando o transistor for saturado, irá comandar um pino do GPIO.

3.7.2 Motor de Passo 28BYJ-48

O motor de passo é unipolar, possui quatro enrolamentos ou fases. Este tipo de motor possui 4 modos de operação. O passo completo com alto torque (*Full step*), o passo completo com baixo torque (*Wave Step*), o meio passo (*Half Step*) e micro-passo (*Micro Stepping*).

Será abordado somente o modo *full step*, pois este que será utilizado. No modo supra citado, duas fases são acionadas ao mesmo tempo, ou seja, para cada passo duas fases são ativadas simultaneamente. O motor será acionado mediante comandos do CI ULN2003A, através das portas 13, 14, 15 e 16.

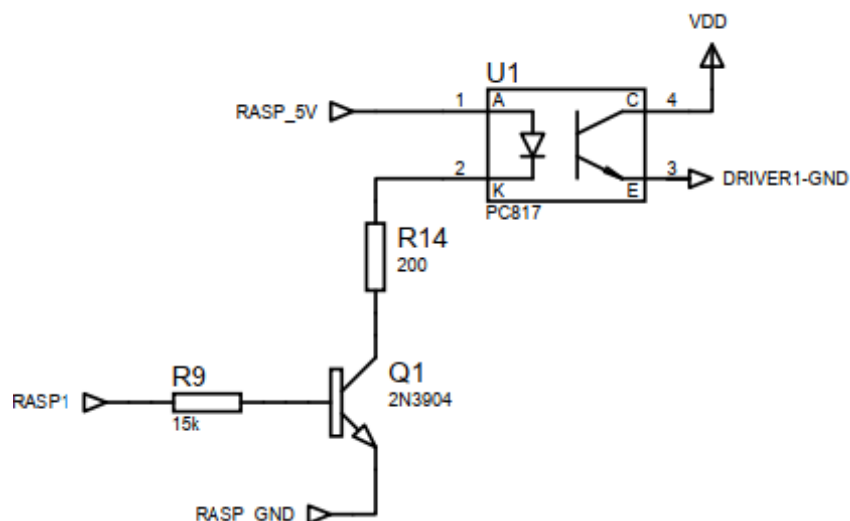
O motor de passo pode ser alimentado com 5 V e 12 V e consome baixa corrente, é unipolar pois possui 4 enrolamentos, sendo seu comum positivo. Já em sua construção física, a distância entre os dois suportes de parafusos é de 35 mm. O diâmetro do motor é de 28 mm e a profundidade é de 19 mm. O diâmetro do eixo é 5 mm, possuindo 2 chanfrados, como facilitador para acoplamento.

Os pinos do GPIO do Raspberry somente disponibilizam 16 mA, sendo necessária a criação do circuito que será apresentado.

Os optoacopladores PC817 estão no encapsulamento DIP-4, sendo os pinos 1,2,3 e 4 respectivamente anodo, catodo, emissor e coletor. Entre os pinos 1 e 2 há uma LED IR e nos pinos 3 e 4 um fototransistor (controlado por luz). Quando a LED é acionada, emite luz saturando a base do transistor, que permite a passagem de energia através de seu coletor e emissor.

Conforme o datasheet, nas características do LED, é requerido, 1,2 V e 20 mA. Vide o circuito para o comando conforme Figura 32.

Figura 32 – Detalhe esquema elétrico.



Fonte: Elaborado pelo Autor.

O resistor R14 está sendo usado para limitar a tensão a 1,2 V, conforme equação (9), sendo 5 V a tensão de alimentação do *Raspberry Pi*:

$$R = \frac{\text{Tensão } 5V \text{ Raspberry} - \text{Tensão LED PC817}}{I} \quad (8)$$

$$R = \frac{5V - 1,2V}{0,020} = 190 \Omega \quad (9)$$

O resistor superior mais comum é o de 200 Ω , sendo este o utilizado.

O resistor R9, está sendo usado para limitar a corrente do GPIO a no máximo 16 mA e aplica-la a base do transistor, a equação (11) apresenta o cálculo da tensão que estará sob o resistor na base.

$$Vr = \text{Tensão } 3,3V \text{ GPIO Raspberry} - \text{Queda Tensão Base Transistor} \quad (10)$$

$$Vr = 3,3 - 0,7 = 2,6 V \quad (11)$$

Após obter a tensão que será aplicada, é necessário localizar no *datasheet* do transistor o ganho (β). Neste caso será de aproximadamente 90. Aplicando a equação (3), será obtido a corrente de base necessário para acionar o transistor.

$$IB = \frac{0,016}{90} = 0,000180 A \quad (12)$$

Portanto o GPIO fornecerá até 16 mA, entretanto, será consumido efetivamente 180 uA. Assim, é possível calcular a resistência que deverá ser conectada a base do transistor, conforme evidenciado na equação (5).

$$R = \frac{2,6}{0,000180} = 14.444,40 \Omega \quad (13)$$

O resistor superior mais comum é o de 15k Ω , sendo este o utilizado.

Quando a LED IR do optoacoplador for acionada, irá permitir a passagem da energia pelo fototransistor, sendo sua corrente máxima de 50 mA e 50 V. Como a tensão de alimentação é de somente 5 V, está dentro do intervalo, já a corrente será limitada por um resistor de *pull-down*, conforme equação (14).

$$I = \frac{V}{R} \quad (14)$$

$$I = \frac{5}{10.000} = 0,000500 \text{ A} \quad (15)$$

O resistor de *pull-down* calculado na equação (15), protegerá também o CI ULN2003A que em seu datasheet informa a tensão máxima de entrada de 30 V e corrente máxima de 25 mA. O CI terá utilizado os pinos 1 ao 4 como entrada, pino 9 como comum e pinos 13 ao 16 como saída. O seu circuito interno possui todas as proteções e suporta até 500 mA por porta de saída, tornando-o uma das opções com melhor custo/benefício para acionar o motor de passo, analisando-se seu custo em comparação a construir o mesmo circuito para acionar o motor.

Conectado as saídas do ULN2003A, estão os LED's de indicação do acionamento e o motor. Essas LED's são de 1,2 V e 20 mA, tornando possível a aplicação da equação (16).

$$R = \frac{\text{Tensão } 5V \text{ Alimentação} - \text{Tensão LED}}{I} \quad (16)$$

$$R = \frac{5V - 1,2V}{0,020A} = 190 \Omega \quad (17)$$

Porém foi optado por se limitar a corrente menor, portanto, o resistor foi definido como 470Ω.

Para o motor, a resistência medida entre comum e as bobinas foi aproximadamente 23 ohms, tornando possível o cálculo da corrente em cada bobina conforme equação (18).

$$I = \frac{5}{23} = 0,217 \text{ A} \quad (18)$$

Portanto os pinos de saída do CI estão com aproximadamente 57% de sua carga nominal, possibilitando assim, que com o mesmo circuito seja possível aumentar a potência do motor.

3.7.3 CI ULN2003A

Esse chip possui um conjunto de sete *drivers* de transistores *Darlington* que permitem o acionamento de cargas indutivas. Todas as saídas tem o coletor aberto e diodos de supressão

(*Clamp*). Os transistores suportam tensões de até 50 V e correntes de até 500 mA. Todas as entradas são compatíveis com sinais TTL e CMOS, com limite de 5 V. O pino comum tem que ser conectado na tensão de alimentação do motor, sendo 5 V. Os quatro leds vermelhos (A,B,C e D) são usados para indicar o acionamento de cada um dos *drivers* (fases do motor). Cada circuito *driver* do chip ULN2003A aciona uma das fases. E a ativação de cada *driver* é realizada pelas portas digitais do *Raspberry Pi*.

3.7.4 Célula de Carga

A célula de carga trata-se de um acessório eletrônico capaz de detectar diferentes cargas que estejam sobre sua meia-ponte. Seu funcionamento é simples, sobre seu centro existe uma área sensível responsável por detectar a carga, já construtivamente é composto por uma ponte resistiva que altera sua resistência conforme o peso aplicado, e ao entrar em operação, envia uma tensão ao microcontrolador, conforme a variação do peso a tensão de saída varia. Porém a tensão é tão baixa que é necessário utilizar um amplificador de sinal para comunicação com o *Raspberry Pi*, comunicação esta que não deve ser feita por meio de pino digital.

O correto funcionamento da célula de carga demanda 2 configurações, a da unidade de referência e o *offset* da tara. Localiza e calcular a unidade de referência nada mais é do que efetuar o "ajuste da balança" que é o ato de referenciar a balança utilizando um peso padrão interno ou externo, dependendo do modelo de cada balança, para que erros de medição sejam minimizados ou corrigidos. Para tanto foi efetuado o seguinte procedimento:

- 1 - No código do programa ou *firmware*, setar a unidade de referência para 1;
- 2 - Utilizar algo que já tenham o peso conhecido, ou com uma balança operacional, pesar algo que será o "peso padrão";
- 3 - Com a célula de carga operacional, depositar este objeto, sob a mesma, assim será mostrado um valor;
- 4 - Com o valor obtido no passo 3, dividir pelo "peso padrão", assim será obtida a unidade de referência;
- 5 - No código do programa ou *firmware*, setar a unidade de referência encontrada no passo 4.

O procedimento acima foi efetuado, obtendo-se as seguintes referências:

Peso Padrão: 7540g

Unidade Medida com a referência setada para 1 adimensional:2.251.887,2732824

Logo a equação (19) evidencia a aplicação do item 4 do procedimento citado acima.

$$\frac{2.251.887,2732824}{7540g} = 298,65878956 \quad (19)$$

Portanto a unidade de referência a ser cadastrada é de 299 adimensional.

Com a parametrização acima a balança já está operacional. Agora pode-se configurar a tara inicial, que normalmente em balanças comerciais, é feito a cada inicialização do sistema. Porém para a aplicação atual, esta definição é indesejada, então configura-se um *offset* ou tara. Assim, foi efetuado várias inicializações da balança, sem carga, e em todas as vezes, a balança recém inicializada trazia um peso sempre muito aproximado de 88g, assim a tara inicial foi setada em 88g.

3.7.5 CI HX711

O CI HX711 é um conversor e amplificador de 24 bits, utilizado para amplificar o sinal de dispositivos como células de carga, fazendo a interligação entre essas células e o *Raspberry Pi*. O CI usa interface SPI para se comunicar com o *Raspberry Pi*. O CI captura a tensão enviada pela célula de carga, amplifica e disponibiliza em sua saída para que seja lido pelo *Raspberry Pi*.

O transistor PNP é usado pelo regulador analógico interno, fornecendo a tensão estável e tensão de excitação para o HX711 e a célula de carga. Os resistores R18 e R19 são usados como um divisor de tensão e fornecendo o *feedback* para o regulador analógico interno. Os resistores R20, R21 e o capacitor C4 formam um filtro passa-baixo diferencial. Os capacitores C1 e C2 são capacitores de desacoplamento para a fonte de alimentação do HX711, assim diminuindo ruídos e variações da fonte. O capacitor C3 está desacoplando para a referência de *bypass*.

3.7.6 RASPBERRY PI

Para a programação, configuração do *firmware* e configuração de comunicação física entre o *Raspberry Pi* e os sensores e atuadores formam utilizados os pinos do GPIO. Assim, cada sensor ou atuador irá comunicar-se por um pino específico, garantindo que não haja interferência entre eles.

Portando foi utilizado o modo de programação de “BCM GPIO”, significando que está sendo referindo aos pinos pelo número "Broadcom SOC *channel*", ou seja, o pino "lógico", e não físico, para tanto, na Tabela 1, é possível verificar os pinos utilizado, sua função, qual é o pino lógico e qual o pino físico.

Tabela 1 – Pinos GPIO

Função	Número Pino Lógico	Número Pino Físico
Bobina 1 Motor de Passo	GPIO24	18
Bobina 2 Motor de Passo	GPIO25	22
Bobina 3 Motor de Passo	GPIO8	24
Bobina 4 Motor de Passo	GPIO7	26
<i>Data CI HX711</i>	GPIO05	29
<i>Serial Clock Input CI HX711</i>	GPIO06	31
<i>Trigger Sensor Ultrassônico - Água</i>	GPIO12	32
<i>Echo Sensor Ultrassônico - Água</i>	GPIO13	33
<i>Trigger Sensor Ultrassônico - Ração</i>	GPIO17	11
<i>Echo Sensor Ultrassônico - Ração</i>	GPIO27	13

Fonte: Elaborado pelo autor

Como detalhado acima, os pinos GPIO24, GPIO25, GPIO8 e GPIO7 estão configurados para o controle do motor, sendo respectivamente para as bobinas 1 a 4, os GPIO05 e GPIO06 recebem os dados provenientes do CI HX711, que interpreta os sinais enviados da célula de carga. Os pinos GPIO12 e GPIO13 recebem os sinais provenientes do *shield* do sensor ultrassônico que controla o nível da água e os pinos GPIO17 e GPIO27 recebem os sinais provenientes do *shield* do sensor ultrassônico que controla o nível da ração.

3.7.7 Parâmetros Gerais

O correto funcionamento demanda de inúmeros pequenos detalhes, tais como:

- Teste de pré e pós abastecimento da tigela, o *firmware* em conjunto ao *software* envia e-mail alertando o nível baixo de ração e água, porém isso não garante que o tutor tenha tempo de efetuar o abastecimento antes do próximo horário agendado de abastecimento, assim para garantir que o motor de passo não fique funcionando a vazio, a cada volta (ou intervalo definido pelo tutor), o sistema irá averiguar o peso anterior (considerar um percentual de erro para mais ou menos), comparar com o peso atual e neste formato verificar se há ração no funil para realizar o abastecimento. Caso não haja ração o sistema irá adicionar uma observação no histórico.

- Definição de tolerância de peso, assim mesmo que a quantidade seja ligeiramente menor que a quantidade definida será interpretado como quantidade correta, pois a célula de carga por mais precisa que seja, ainda sobre efeitos externos, tais como calor e umidade, causando pequenas distorções.

3.7.8 Componentes Eletrônicos Auxiliares

Os resistores conectados ao pino 1 dos CI PC817 são uma proteção contra corrente superior a que pode ser disponibilizada pelo *Raspberry Pi*.

Os resistores conectados aos pinos 1, 2, 3 e 4 do CI ULN2003A são resistores de *pull-down*, garantido que não haja flutuações diminuindo a chance de erros nos pinos.

Os LED's nas saídas 13, 14, 15 e 16 do CI ULN2003A são meramente indicativas de operação, facilitando a visualização de qual fase do motor está sendo alimentado.

3.8 Materiais utilizados

Neste protótipo são utilizados os itens listados na Tabela 2.

Tabela 2 – Lista dos itens

Tipo	Item	Quantidade	Unidade
CI	Transistor PNP - BC558	1	un
CI	Circuito Integrado ULN2003A	1	un
Resistor	Resistor 10K 5% (1/2W)	4	un
Resistor	Resistor 160R 5% (1/4W)	10	un
Resistor	Resistor 20K 5% (1/2W)	1	un
Resistor	Resistor 8K2 5% (1/4W)	1	un
Resistor	Resistor 1K 5% (1/2W)	1	un
Capacitor	Capacitor Tântalo 0,1uF / 35V	1	un
Capacitor	Capacitor Tântalo 1uF / 16V	3	un
Soquete	Soquete Torneado 16 Pinos	1	un
Sensor	Sensor de Distância Ultrassônico HC-SR04	2	un
LED	LED de alto brilho 3mm Azul	4	un
Sensor	Sensor de Peso 50Kg Célula de Carga	1	un
Conector	Conector Alojamento Modu 2,54mm 15 Vias	1	un
Terminal	Terminal Fêmea Para Alojamento Modu	30	un
Conector	Conector Alojamento Modu 2,54mm 4 Vias	6	un

Conector	Barra de Pinos 40 vias 11,2mm 180 graus	1	un
Conector	Conector Alojamento Modu 2,54mm 6 Vias	1	un
Cabo	Cabo Flat 10 Vias Cinza 28 AWG	2	m
Cabo	Cabo Flat 5 Vias Colorido 26 AWG	1	m
Cabo	Cabo Flat 4 Vias Colorido 26 AWG	2	m
CI	Optoacoplador PC817	10	un
CI	Conversor Amplificador HX711	1	un
Fonte	Fonte para <i>Raspberry Pi</i> 4 USB 5V 4.2A	1	un
SBC	<i>Raspberry Pi 2</i>	1	un
Atuador	Motor de Passo	1	un
PCB	Placa para PCB	1	un
Impressão	Protótipo em 3D	1	un

Fonte: Elaborado pelo autor

Foi necessária a impressão do protótipo em 3D, e a montagem da PCB.

3.9 Orçamento do Projeto

Na tabela 3 são demonstrados os valores orçados.

Tabela 3 – Descrição dos custos

Tipo	Item	QTD	Unid	Custo Un	Frete	Total
CI	Transistor PNP - BC558	1	un	0,12	0,04	0,16
CI	Circuito Integrado ULN2003A	1	un	1,61	0,52	2,13
Resistor	Resistor 10K 5% (1/2W)	4	un	0,14	0,05	0,76
Resistor	Resistor 160R 5% (1/4W)	10	un	0,06	0,02	0,80
Resistor	Resistor 20K 5% (1/2W)	1	un	0,14	0,05	0,19
Resistor	Resistor 8K2 5% (1/4W)	1	un	0,06	0,02	0,08
Resistor	Resistor 1K 5% (1/2W)	1	un	0,14	0,05	0,19
Capacitor	Capacitor Tântalo 0,1uF / 35V	1	un	0,81	0,26	1,07
Capacitor	Capacitor Tântalo 1uF / 16V	3	un	0,57	0,19	2,28
Soquete	Soquete Torneado 16 Pinos	1	un	2,08	0,68	2,76
Sensor	Sensor de Distância Ultrassônico HC-SR04	2	un	12,90	4,21	34,22
LED	LED de alto brilho 3mm Azul	4	un	0,24	0,08	1,28
Sensor	Sensor de Peso 50Kg Célula de Carga	1	un	12,26	4,00	16,26
Conector	Conector Alojamento Modu 2,54mm 15 Vias	1	un	0,94	0,31	1,25
Terminal	Terminal Fêmea Para Alojamento Modu	30	un	0,10	0,03	3,90

Conector	Conector Alojamento Modu 2,54mm 4 Vias	6	un	0,14	0,05	1,14
Conector	Barra de Pinos 40 vias 11,2mm 180 graus	1	un	0,77	0,25	1,02
Conector	Conector Alojamento Modu 2,54mm 6 Vias	1	un	0,18	0,06	0,24
Cabo	Cabo Flat 10 Vias Cinza 28 AWG	2	m	2,46	0,80	6,52
Cabo	Cabo Flat 5 Vias Colorido 26 AWG	1	m	3,21	1,05	4,26
Cabo	Cabo Flat 4 Vias Colorido 26 AWG	2	m	2,29	0,75	6,08
CI	Optoacoplador PC817	10	un	0,55	0,18	7,30
CI	Conversor Amplificador HX711	1	un	13,76	3,07	16,83
Fonte	Fonte para <i>Raspberry Pi</i> 4 USB 5V 4.2A	1	un	37,90	8,44	46,34
SBC	<i>Raspberry Pi</i> 2	1	un	290,00	10,25	300,25
Atuador	Motor de Passo	1	un	15,00	3,25	18,25
PCB	Placa para PCB	1	un	9,00	1,50	10,50
Impressão	Protótipo em 3D	1	un	180,00	5,24	185,24
Total:						671,30

Fonte: Elaborado pelo autor

Como pode ser constatado, o valor total do projeto/protótipo é expressivo, porém, quando comparado com produtos comercializados, o mesmo está na mesma faixa de preço. Tendo em vista a cotação do dólar (1.00USD \approx R\$6,00) na data de 09/03/2021, o produto "*Smart Feed Automatic Dog and Cat Feeder*", com venda na Amazon, (USD 170.00) e convertemos hoje, sem impostos estaria R\$ 1.020,00. Então a produção em larga escala, tornará o produto competitivo, em relação aos custos e aos concorrentes. Ao considerar que o produto será importado legalmente, pagando-se as taxas, uma pessoa física através do CPF, pagará aproximadamente 50% de imposto de importação, que tem como base o valor do produto mais o frete, então o valor do produto sendo de USD 170.00, e de forma conservadora, considerando o frete em USD 25.00, o montante total está calculado na Tabela 4:

Tabela 4 – Custos do Produto e Importação

Item	Custo Un. USD	Val. R\$
<i>Smart Feed Automatic Dog and Cat Feeder</i>	170.00	1.020,00
Frete	25.00	150,00
Imposto sobre Produto + Frete	50%	585,00
Total:		1.755,00

Fonte: Elaborado pelo autor

Comparando-se os custos, o protótipo está aproximadamente R\$ 1.000,00 abaixo do valor de um produto similar importado.

3.10 Planejamento Empresarial

Para tornar a venda de um eventual produto que se origine deste protótipo, é necessário a criação de um CNPJ, no qual a comercialização poderá ser efetuada. Portanto, deverá ser considerado o custo da abertura de uma empresa e seu custo de operação, neste momento, somente a área fiscal está sendo considerada.

Atualmente o melhor cenário, ou seja, o que possui o menor custo de manutenção, é uma MEI, que é obrigada a pagar mensalmente aproximadamente R\$ 59,00 reais para se manter ativa, considerando-se que o cadastro é gratuito (desde que não houver pendências com a receita federal, senão a abertura do CNPJ pode tornar-se mais complicada).

Na Tabela 5, é exposto um uma sugestão de preço e venda mediante alguns parâmetros baseando-se exclusivamente no custo atualmente calculado.

Tabela 5 – Custos do produto e preço de venda

Item	Custo Un.	Val. R\$
<i>Custo do Produto</i>	671,30	671,30
Custos administrativos	5%	33,56
Frete	5%	33,56
Impostos Federais, Estaduais e Municipais (ICMS, PIS, COFINS, DIFAL, ISS, etc)	40%	268,52
Resultado (Lucro)	10%	67,32
Sugestão do Valor de Venda:		1.074,26

Fonte: Elaborado pelo autor

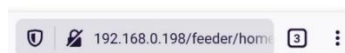
4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados obtidos a partir dos procedimentos metodológicos propostos. Foi realizada uma análise do comportamento operacional dos sistemas, bem como teste de utilização por dois usuários utilizando cinco sistemas operacionais, para verificar a estabilidade em vários sistemas.

4.1 Configuração da interface gráfica

A interface gráfica é disponibilizada em HTML e PHP, construída para dimensionamento automático. Desta forma é possível abrir a interface em qualquer dispositivo, desde um *smartphone* a um computador. Esta foi desenvolvida para ser de fácil utilização pelo tutor, para que a configuração inicial não demore mais que cinco minutos. A Figura 33 apresenta a tela inicial do sistema web. Nos apêndices deste trabalho podem ser encontradas mais telas

Figura 33 – Interface gráfica – Tela inicial (Home).



Fonte: Elaborado pelo autor.

4.2 Envio de *E-mail* ao tutor

Conforme os parâmetros cadastrados pelo tutor, o sistema/*firmware* irá executar uma função de envio de e-mail. A cada execução, será verificado se os níveis definidos estão em

conformidade, e caso não estejam, será enviado um e-mail conforme Figura 34, informando o tutor sobre os níveis atuais.

Figura 34 – *E-mail* de alerta.



Fonte: Elaborado pelo Autor.

Com esse *e-mail*, será possível que o tutor saiba com antecedência que os funis estão próximos de estarem vazios.

4.3 Construção Física em 3D do Protótipo

O funil do protótipo na escala 1:4 armazenou aproximadamente 150 g de ração, então o protótipo em escala 1:1 suportará aproximadamente 600 g de ração.

Foram testados 5 modelos de ração, destas somente 2 foram aprovadas para serem usados com o produto, pois são rações para filhotes, portanto fisicamente menores, e tendo em vista o protótipo em escala, esta era uma situação esperada.

As rações reprovadas são de cachorros adultos, e devido ao seu formato e tamanho bloquearam o motor de passo, que em seu torque máximo produz 2,2 kgf/m, essa também era uma situação esperada, devido ao baixo torque.

A junção do formato da construção do funil, rosca helicoidal e torque do motor de passo gera um novo problema, que é o travamento da rosca helicoidal e conseqüentemente o motor, devido ao bloqueio por uma ração. Assim foi elaborada uma lógica, que a cada meia volta lógica do motor de passo, este irá parar, girar no sentido reverso 1/8 de volta e, posteriormente, voltar ao sentido programado de giro. Com esta lógica espera-se, que caso

haja algum travamento por bloqueio de ração, o bloqueio seja desfeito, permitindo a volta do normal funcionamento ao sistema. A Figura 35 apresenta o produto durante um teste em bancada.

Figura 35 – Vista frontal do produto.



Fonte: Elaborado pelo Autor.

4.4 Testes e medições

Para certificar que o projeto apresentado atingiu seus objetivos específicos, testes de validação relacionado à precisão da balança, precisão do sensor do reservatório, assertividade dos comandos variados e teste em bancada da comunicação entre os dispositivos.

4.4.1 Teste em bancada da comunicação entre os dispositivos

O teste experimental da comunicação entre os dispositivos em bancada foi realizado e vários trechos do *firmware* necessitaram de manutenção ou novas validações/bloqueios, sendo assim escritos. Foram realizados os mais variados testes: falha de energização, seccionamento de cabos, inversão das conexões dos *plugs* dos sensores e atuadores.

4.4.1.1 Teste de Falha de energização

O procedimento empregado neste teste, foi o de desligar a fonte de alimentação ou desconectar o pino positivo ou negativo da fonte. Foram repetidas 30 vezes, em todas as situações, o *firmware* identificou 2 situações distintas.

- Situação 1: Sistema desenergizado, com o motor de passo acionado.
Nesta situação, o sistema identificou como se não houvesse mais ração no funil, pois mesmo sendo acionado, a célula de carga não detectava aumento da quantidade de ração na tigela.
- Situação 2: Sistema desenergizado, sem o motor de passo acionado.
Nesta situação, o sistema travava, pois era enviado o sinal para os sensores ultrassônicos, porém não havia resposta, assim, foi inserida uma lógica para que quando não houver resposta dos sensores ultrassônicos, o sistema passa a entender que houve um erro de leitura, não trava seu processamento e considera que a quantidade de ração e água está na máxima.

4.4.1.2 Teste de Seccionamento de cabos

Neste teste, o procedimento empregado, foi o de durante a execução do *firmware*, os cabos serem desconectados da interface de comando/potência. Foram desligados aleatoriamente por 30 vezes, os cabos dos sensores ultrassônicos, da célula de carga, do motor de passo ou do comando do *Raspberry Pi*. Em todos os testes as situações foram idênticas as dos testes de desligamento.

4.4.1.3 Teste de Inversão das conexões dos *plugs* dos sensores e atuadores

Este teste foi somente feito no motor de passo e na célula de carga, pois os sensores ultrassônicos e o *Raspberry Pi* são mais sensíveis, podendo danificar permanentemente. O motor de passo, quando ligado invertido era acionado, mas não conseguia girar seu eixo. Por outro lado, a célula de carga não efetuava a pesagem. A solução para este problema é física,

ou seja, durante a elaboração do projeto, inseriu-se *plugs* que impedem a conexão de forma inversa.

4.4.2 Precisão da Balança

A forma de teste utilizada é a descrita abaixo e a cada processo foi registrado o valor. Primeiramente, foi pesado a tigela, na célula de carga e em balança externa, então com a tigela completamente vazia, e o *firmware* em *standy-by*. Foi alterado o SQL, que recupera da base de dados o valor a ser depositado na tigela. Em seguida, houve a execução do *firmware*, iniciando o processo de abastecimento da ração para a tigela, e com o término da execução, visualizou-se o peso do recipiente no *log* gerado pela execução do *firmware*. Registrou-se o valor, e o mesmo recipiente teve a medida aferida na balança externa. Posteriormente, comparou-se o valor resultante da balança com o que era exibido no *log*.

A estatística amostral dos resultados das 20 medições realizadas individualmente para cada padrão de peso 50 g, 75 g e 100 g, pode ser observado na Tabela 6.

Tabela 6 – Estatística Amostral

Gramas	50g	75g	100g
Média	50,02	74,115	99,91
Erro padrão	0,35980	0,43603	0,69566
Modo	50,6	75,2	100,8
Mediana	50,55	74,3	100,45
Primeiro quartil	48,35	72,6	96,675
Terceiro quartil	51,225	75,2	102,475
Variância	2,58905	3,80239	9,67884
Desvio padrão	1,60905	1,94997	3,11108
Curtose	-1,50447	-0,46038	-1,53207
Inclinação	-0,20040	0,26083	0,08912
Intervalo	4,40000	7,00000	8,70000
Mínimo	47,7	71,3	95,9
Máximo	52,1	78,3	104,6
Soma	1000,4	1482,3	1998,2
Contagem	20	20	20

Fonte: Elaborado pelo autor

Percebe-se que existe uma variação, pelo fato de usarmos uma rosca helicoidal, isto é, o sistema realiza o acionamento da rosca helicoidal, que gira e em seguida é feito a captura do

peso da ração que foi depositada. Desta forma, o programa verifica se o peso está no intervalo do peso solicitado, e no intervalo do percentual de variação cadastrada.

4.4.3 Precisão do Sensor dos Reservatórios

Foram efetuados testes de precisão com os sensores dos reservatórios verificando a efetividade da captura dos sensores e se estavam realmente calibrados com os níveis depositados. Nesta metodologia, o reservatório foi completamente esvaziado, e o *firmware* executado. Em seguida foi observado o nível, e posteriormente adicionou-se ração/água ao reservatório e verificou-se se o valor correspondia realmente à quantidade depositada no reservatório, pois com o auxílio de uma regra foi encontrada a mesma altura. Foram 10 medições realizadas individualmente para os 2 sensores. Constatou-se que a calibragem dos sensores, com o protótipo, estava ideal para o cenário, pois o resultado foi excelente, 100% de acerto.

4.4.4 Teste de Desempenho de Comandos

O teste de desempenho de comandos variados foi realizado para verificar o tempo de resposta para executar as diversas funções do aplicativo/*firmware*. Este teste foi praticado através da seguinte metodologia: inicialmente abriu-se o aplicativo e de forma aleatória selecionou-se a funcionalidade desejada. Neste meio tempo contabilizou-se através de um cronômetro o tempo para executar tal tarefa. Os resultados das 50 medições realizadas referentes aos comandos das funções do aplicativo/*firmware*, desde a verificação do nível de ração, nível do recipiente, visualização do relatório até o acionamento de ração agendada. Majoritariamente o tempo de execução de um ciclo é de 6 segundos, compreendendo, a leitura dos sensores de ração, água e célula de carga, recuperação e gravação dos dados no banco de dados e envio de e-mail, quando necessário. Já o tempo de execução quando há agendamento para disponibilizar a ração é muito variado, pois o tempo está diretamente proporcional a quantidade de ração e/ou se houve algum travamento da rosca helicoidal, assim aumentando o tempo de execução. A maior quantidade testada foi de 300 g de ração, aonde o tempo de execução foi de aproximadamente 3 minutos.

4.4.5 Teste de Assertividade de Comandos

O teste de assertividade de comandos variados foi realizado para verificar a quantidade de comandos que foram reconhecidos corretamente nas diversas funções do aplicativo/*firmware*. Este teste foi praticado através da seguinte metodologia: inicialmente abriu-se o aplicativo e de forma aleatória selecionou-se a funcionalidade desejada, verificando-se então se o comando solicitado era realizado com sucesso. Os resultados das 50 medições realizadas referentes aos comandos das funções do aplicativo/*firmware*, desde verificar o nível de ração, nível do recipiente, visualização do relatório e acionamento de ração imediato ou agendado. Conclui-se que existe uma eficácia na execução dos comandos, sendo unânime o sucesso em todas execuções, talvez pelo fato de ser em rede local. Isto favoreceu em 100% os resultados em todos os testes de assertividade.

4.4.6 Medições

Durante a operação do sistema foram efetuadas várias medidas de corrente e tensão em diversos pontos do sistema, sendo apresentados na Tabela 7, em todas as medições a variação da tensão foi de no máximo 0,2 V, ou seja, tensão entre 4,8 V e 5,2 V.

Tabela 7 – Medições de corrente.

Situação	Corrente Mínima (mA)	Corrente Máxima (mA)
Sensores e atuadores acionados	248	253
Motor de passo acionado	215	217
Célula de carga acionada	10	10
Sensor ultrassônico da ração acionado	14	15
Sensor ultrassônico da ração acionado	14	15

Fonte: Elaborado pelo autor

4.5 Simulação/cadastrros

Todas as telas apresentadas de cadastro, edição e deleção foram testadas em terminais *Desktop* (*Windows*, *Ubuntu* e *RaspberryOS*), celulares com sistema operacional *Android* (*Google*) e *IOS* (*Apple*). Em todas as simulações o cadastro, edição e deleção foram efetuados com sucesso, demonstrando estabilidade por parte do servidor (*Raspberry Pi*), sistema web (mescla entre *HTML* e *PHP*) e da base de dados (*MySQL*). Em todos os testes o tempo

máximo da execução das tarefas, não demorou mais de 1ms, o que indica que a plataforma escolhida atende à necessidade, tendo ainda possibilidade de crescimento.

Pela natureza do sistema sempre será necessário que o tutor possua um navegador. Neste sentido foram testados os principais navegadores disponíveis no mercado, sendo eles Mozilla Firefox, Internet Explorer, Microsoft Edge, Opera, Chrome (e algumas distros) e Safari.

4.5.1 Registro das informações

Todos os dados parametrizados nas telas da interface gráfica, ou dados obtido/capturados pelos sensores serão registrados no banco de dados deste formato, será possível efetuar consultas posteriores. O tempo de retenção dos dados é indefinida.

4.6 Falhas, Erros e Desistências

Durante a elaboração do projeto e do protótipo, foi pesquisado sobre a qualidade e temperatura que a água deveria ser oferecida ao *pet*. Neste sentido foi idealizado uma pastilha de *Pielter* sendo este o meio de refrigeração da água na tigela. Contudo, conforme o avanço do projeto, foi verificado que para todas as funções ativas simultaneamente, a corrente total passaria de 10A, sendo a pastilha de *Pielter* um consumo máximo de 6A. Esse consumo impactaria em alto incremento de custos e consumo de energia, pois caso a temperatura configurada pelo tutor, fosse muito diferente do ambiente, a pastilha ficaria por longos períodos ativa aumentando consideravelmente o consumo. Então nos tempos de consciência ecológica, não faria muito sentido, disponibilizar esta função. Apesar disso, será avaliada uma alternativa para o seguimento do trabalho.

A definição de qual sistema operacional deveria ser usado foi outro desafio, pois cada um deles traria algum benefício/malefício ao projeto, além de vários obstáculos, devido ao tipo de sistema, consumo de memória ou processamento, por fim foi definido que seria o *RaspberryOS*, o qual se mostrou mais eficiente na SBC *Raspberry Pi*, mesmo assim, foram utilizadas várias horas do projeto para conseguir instalar e configurar todos os aplicativos necessários ao correto funcionamento do projeto.

Até o momento, há uma falha no *RaspberryOS*, que causa a interrupção aleatória da comunicação via *wi-fi* do dispositivo a rede interna ao qual ele está conectado, porém o acesso à internet permanece operacional. Assim, enquanto o desenvolvedor não disponibilizar uma

correção, a comunicação *wi-fi* poderá ficar intermitente, causando alguns problemas. A solução mais simples foi conectar o dispositivo de forma cabeada.

Erros não documentados na base de dados MySQL foram outra fonte de problemas, pois em outras bases de dados, a função idêntica funcionava perfeitamente, enquanto que na execução do SQL no banco mencionado não eram localizadas informações, causando grande transtorno até a criação de uma solução alternativa que efetuasse a operação muito próxima.

4.7 Possibilidades de melhorias

Verificou-se que a saída do funil da ração ficará aberto, o que ocasionara exposição da ração com o ar, podendo o *pet* se recusar a comê-la. Desta forma uma melhoria seria fechar a parte inferior para impedir o contato da ração com o meio.

Outra melhoria seria criar uma rotina para enviar um e-mail ao tutor, alertando que o sistema identificou durante o dia, que o pet não ingeriu a quantidade de água/ração recomendada (esta informações já está sendo gravada no banco de dados), pois em muitos casos, a falta de ingestão de água indica algum problema de saúde do animal.

5 CONSIDERAÇÕES FINAIS

O mercado voltado ao *pet* aumenta cada vez mais e com isso as possibilidades de criação de máquinas e dispositivos que se relacionem aos animais de estimação. Neste contexto, este trabalho tem proposto o desenvolvimento de um alimentador automatizado. Este dispositivo proporciona uma autonomia de ilimitadas refeições programáveis em um período de sete dias, atendendo assim a maioria das rotinas diárias de alimentações recomendadas. O maior benefício frente aos alimentadores não automatizados é a possibilidade de definir horários para alimentações, dando assim total comodidade ao tutor, bem como o sistema de reposição de água, incomum até em equipamentos comercializados.

Para tornar o protótipo economicamente viável o alimentador automático foi concebido utilizando um SBC dedicado para realizar o controle. O protótipo desenvolvido é alimentado por energia e dispõe de local de armazenagem para alimentos sólidos e água. Para automatizar o protótipo foi escolhida a linguagem Python, como *firmware*, HTML e PHP como interface gráfica e MySQL como banco de dados e utiliza uma interface para controlar os atuadores e sensores existentes. Todos os *softwares* utilizados são *freewares*, tornando o investimento em licenças desnecessário.

Com a proposição do desenvolvimento de um protótipo de alimentador automático para *pets* visando à obtenção de dados para facilitar a vida do tutor, mediante monitoramento do alimento e água disponibilizada ao *pet*. Baseado em vários produtos comerciais, projetos *do it yourself*, necessidades identificadas pelo autor (tutor de um *pet*), onde foi possível levantar as informações referentes aos objetivos. Esses dados foram relevantes para o desenvolvimento do projeto.

Conforme apresentado nos resultados, as informações armazenadas oriundas dos requisitos da automação são armazenadas no banco de dados. Esses dados podem ser utilizados, tanto na identificação de problemas de saúde, como na necessidade de aquisição de ração. A implantação do sistema WEB como interface gráfica, faz com que as variáveis adquiridas pelo sistema sejam dispostas em tempo real (na realidade com atraso de 1 minuto) ao tutor, assim facilitando o monitoramento.

A automação é capaz de trabalhar por longos períodos de tempo, sem a intervenção do tutor, e atualmente, já estão registrados mais de 325000 tuplas no bando de dados, referente somente ao histórico do dispositivo.

5.1 Sugestão para Continuação do Trabalho

A próxima etapa que poderá ser realizada, é a remodelagem e reimpressão do protótipo, considerando várias melhorias verificadas durante sua construção. No quesito eletrônico, poderá ser reelaborada a placa de comando considerando-se *plugs* que não permitam sua conexão inversa evitando a queima e a placa ser completamente construída com componentes *Surface Mounted Device* (SMD), visando diminuir ao máximo o tamanho físico da placa, com posterior impressão e montagem.

Uma melhoria seria alterar a posição da célula de carga para baixo da estrutura do funil de abastecimento da ração, assim seria eliminado o problema de pet comer a ração simultaneamente ao abastecimento.

Poderia ser criado também o envio de mensagens via Whatsapp, ideia que foi considerada, porém necessita de APIs externas, que são pagas, criando um custo mensal ao tutor.

REFERÊNCIAS

ABDULLA R.; SELVAPERUMAL S. K.; AL-ADILEE M. A. **IOT based Pet Feeder**. Disponível em: <<https://www.researchgate.net/publication/340261266>> Acesso em 12 de outubro de 2020.

ABINPET **Informações gerais do setor Pet** Disponível em: <http://abinpet.org.br/infos_gerais/> Acesso em 08 de novembro de 2020.

ALBUQUERQUE, M. S. M.; EGITO, A. A.; MARIANTE, A. S. **Programa brasileiro de conservação de recursos genéticos animais**. Archivos de zootecnia, v. 51, n. 193, p. 7, 2002. Disponível em: <<https://dialnet.unirioja.es/servlet/articulo?codigo=279936/>>. Acesso em 31 de agosto de 2020.

ALURA **O que é SQL?** Disponível em: <<https://www.alura.com.br/artigos/o-que-e-sql>> Acesso em 08 de novembro de 2020.

AVNRDF **Bake Mate - Pi Chef Blog #5 - Testing the weighing scale: load cell + HX711**. Disponível em: <<https://www.element14.com/community/community/design-challenges/pichef/blog/2018/02/18/bake-mate-pi-chef-blog-5-testing-the-weighing-scale-load-cell-hx711>> Acesso em 22 de setembro de 2020.

BALAKRISHNAN, S. M.; SANGAIAH, A. K. **Integrated QoUE and QoS approach for optimal service composition selection in internet of services (IoS)**. New York: Ed. Springer, 2016

BUENO, C. **Raspberry PI – GPIO input com Python** Disponível em: <<https://www.embarcados.com.br/raspberry-pi-gpio-modo-input-python/>> Acesso em 03 de agosto de 2020.

BRAH G. **42 Smart Pet Accessories** Disponível em: <<https://www.trendhunter.com/slideshow/smart-pet-accessories>> Acesso em 02 de novembro de 2020.

BREMER, R. **DIY Auto Fish Feeder Feeds Fish Automatically** Disponível em: <<https://hackaday.com/2014/10/14/diy-auto-fish-feeder-feeds-fish-automatically/>> Acesso em 03 de agosto de 2020.

CALLAWAY, E. **Dog's dinner was key to domestication**. Nature News. Disponível em: <<https://www.nature.com/news/dog-s-dinner-was-key-to-domestication-1.12280/>>. Acesso em 11 de setembro de 2020.

CLEVERPET. **Give them something to do while you work from home** Disponível em: <<https://clever.pet/>> Acesso em 08 de outubro de 2020.

CONSTANDINOU, C. T. **Tudo sobre motores de passo** Disponível em: <http://w3.ufsm.br/fuentes/index_arquivos/step.pdf> Acesso em 08 de outubro de 2020.

DANEKSHEA. **Auger-based feeder for animals, fish etc using Nema 17 stepper motor** Disponível em: <<https://br.pinterest.com/pin/372039619197980026/>> Acesso em 03 de agosto

de 2020.

DALY, Matasha. **National Geographic. Domesticated animals, explained.** Disponível em: <<https://www.nationalgeographic.com/animals/reference/domesticated-animals/>>. Acesso em 10 de outubro de 2020.

DESIGN 1ST **The 17 Hottest Pet Tech Products Ideas of 2020** Disponível em: <<https://www.design1st.com/pet-tech-products/>> Acesso em 02 de novembro de 2020.

FEHRENBACH, R. B. **Desenvolvimento de Sistema Automatizado para Controle de Dosagem de Ração Animal.** Trabalho de conclusão de curso - Curso de Engenharia de Controle e Automação, Centro de Ciências Exatas e Tecnológicas do Centro Universitário. Lageado, 2017

FIDO. **Hard Leaving Your Pet Home Alone?** Disponível em: <<https://www.felixfidopets.com/petbot>> Acesso em 08 de outubro de 2020.

FILIFEFLOP. **Sensor de Peso 50Kg Célula de Carga** Disponível em: <<https://www.filife flop.com/produto/sensor-de-peso-50kg-celula-de-carga/>> Acesso em 03 de agosto de 2020.

FILIFEFLOP. **Sensor de Distância Ultrassônico HC-SR04** Disponível em: <<https://www.filife flop.com/produto/sensor-de-distancia-ultrassonico-hc-sr04>> Acesso em 28 de março de 2021.

FRENZEL, L. E. **Contemporary Electronics - Fundamentals, Devices, Circuits, and Systems.** Ed. Mc Graw Hill, 2014.

FRENZEL, L. E. **Electronics Explained - Fundamentals for Engineers, Technicians, and Makers.** Ed. Elsevier, 2018.

GORNI, A. A. **Introdução À Prototipagem Rápida E Seus Processos.** Disponível em: <<http://www.gorni.eng.br/protrap.html>> Acesso em 05 de setembro de 2020.

HELLO TECH **Best Smart Home Pet Products for 2019** Disponível em: <<https://www.hellotech.com/blog/best-smart-home-pet-products-for-2019>> Acesso em 02 de novembro de 2020.

HI TECH HOME PROS **Pet Automation for your Smart Home** Disponível em: <<http://hitechhomepros.com/pet-automation-smart-home/>> Acesso em 02 de novembro de 2020.

HTML **HTML** Disponível em: <<https://pt.wikipedia.org/wiki/HTML>> Acesso em 08 de novembro de 2020.

IBAMA. **PORTARIA Nº 93 ,DE 07 DE JULHO 1998.** Disponível em: <http://www.institutohorus.org.br/download/marcos_legais/Portaria%20IBAMA%2093%20-%201998%20animais%20domesticos.pdf/>. Acesso em 10 de setembro de 2020.

IDC **As tendências da cibersegurança da internet das coisas para 2019.** Disponível

em:<<https://forbes.com.br/colunas/2019/08/as-tendencias-da-ciberseguranca-da-internet-das-coisas-para-2019/>> Acesso em 22 de outubro de 2020.

IFETCH. The Original iFetch Disponível em:<<https://goifetch.com/products/#ifetch-original>> Acesso em 08 de outubro de 2020.

KAVANAGH J. The essential smart home guide for pet owners Disponível em:<<https://forums.tomsguide.com/faq/the-essential-smart-home-guide-for-pet-owners.122126/>> Acesso em 01 de novembro de 2020.

KPC TEAM. Overclocking for Raspberry Pi 3 Model B Disponível em:<https://xdevs.com/article/rpi3_oc/> Acesso em 03 de agosto de 2020.

KRICHBAUMER, T. Fully automatic cat feeder Disponível em:<<https://www.myminifactory.com/object/3d-print-fully-automatic-cat-feeder-101416>> Acesso em 03 de agosto de 2020.

LASERTOY. PetSafe Zoom - Automatic, Interactive, Rotating Laser Cat Toy with Two Lasers Disponível em:<<https://www.amazon.com/PetSafe-Zoom-Rotating-Laser-Automatic/dp/B07CQL4WXT>> Acesso em 08 de outubro de 2020.

LITTER-ROBOT. Litter-Robot 3 Connect Disponível em:<<https://www.litter-robot.com/litter-robot-iii-open-air-with-connect.html>> Acesso em 08 de outubro de 2020.

MICHAELIS Sensor Disponível em:<<http://michaelis.uol.com.br/busca?r=0&f=0&t=0&palavra=sensor>> Acesso em 08 de novembro de 2020.

MICROCONTROLLERSLAB. 28BYJ-48 5 volt Stepper Motor Guide Disponível em:<<https://microcontrollerslab.com/28byj-48-5-volt-stepper-motor-guide/>> Acesso em 03 de agosto de 2020.

NETO A. M. Circuitos Integrados Disponível em:<<https://www.ime.usp.br/~adao/CIRCUITOSINTEGRADOSF.pdf>> Acesso em 22 de outubro de 2020.

OTTONI, Claudio et al. The palaeogenetics of cat dispersal in the ancient world. Nature Ecology & Evolution, v. 1, n. 7, p. 1-7, 2017. Disponível em:<<https://www.nature.com/articles/s41559-017-0139.epdf>>. Acesso em 03 de setembro de 2020.

P3. CatFi Pro, um “gadget” para melhorar a saúde do teu gato Disponível em:<<https://www.publico.pt/2016/06/08/p3/noticia/catfi-pro-um-gadget-para-melhorar-a-saude-do-teu-gato-1826223>> Acesso em 08 de outubro de 2020.

PEDEN, C. The Hipster Harness Is Here to Help Heal Your Dog’s Hip Dysplasia Disponível em:<<https://www.petguide.com/blog/dog/the-hipster-harness-is-here-to-help-heal-your-dogs-hip-dysplasia/>> Acesso em 08 de outubro de 2020.

PETCHATZ. How PetChatz Comforts Your Home Alone Pet Disponível em:<<https://petchatz.com/explore-petchatz/how-it-works/>> Acesso em 08 de outubro de 2020.

PETSAFE. PetSafe Smart Feed Automatic Dog and Cat Feeder - Smartphone - Wi-Fi Enabled for iPhone and Android Smartphones Disponível em:<<https://www.amazon.com/PetSafe-Automatic-Generation-Smartphone-Replenishment/dp/B07NR47N2Q>> Acesso em 03 de agosto de 2020.

PETSATHOME. SureFlap Product Disponível em:<<https://www.petsathome.com/shop/en/pets/brand-a-to-z/brand-s/sureflap>> Acesso em 08 de outubro de 2020.

PHP O que é o PHP? Disponível em:<https://www.php.net/manual/pt_BR/intro-what-is.php> Acesso em 08 de novembro de 2020.

PLATZER, A. Logical Foundations of Cyber-Physical Systems. Ed. Springer, 2018.

PLAYDATE. The World's Smartest Pet Camera. Interact With Your Pets From Anywhere Disponível em:<<http://www.startplaydate.com/>> Acesso em 08 de outubro de 2020.

PROAVENTURA. Suporte Dog Harness Cachorro Original Para Câmeras De Ação Gopro Hero Sjcam Xiaomi Disponível em:<<https://www.proaventura.com.br/suportes/cachorro/gopro-fetch>> Acesso em 08 de outubro de 2020.

PUPPOD. The most advanced interactive dog game ever created. Disponível em:<<https://puppod.com/>> Acesso em 08 de outubro de 2020.

PYTHON What is Python? Executive Summary Disponível em:<<https://www.python.org/doc/essays/blurb/>> Acesso em 08 de novembro de 2020.

RASPBERRY PI. Raspberry Pi 2 Model B Disponível em:<<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>> Acesso em 03 de agosto de 2020.

RASPBERRY PI FOUNDATION What is a Raspberry Pi? Disponível em:<<https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>> Acesso em 08 de novembro de 2020.

RAYES, A; SALAM, S. Internet of Things From Hype to Reality The Road to Digitization. Ed. Springer, 2019.

SABATINA A. Innovation Smart Pet Products You Didn't Know You Needed Disponível em:<<https://www.inverse.com/article/54686-smart-pet-products-on-amazon>> Acesso em 02 de novembro de 2020.

SOFTWAREBROTHERS. Wearable Dog's Activity Tracker Disponível em:<<https://softwarebrothers.co/our-work/wonder-woof>> Acesso em 08 de outubro de 2020.

STRINGHAM, G. **Hardware/Firmware Interface Design: Best Practices for Improving Embedded Systems Development**. USA: Ed Newnes, 2010.

SUPERFEEDER. **Don't just feed your pets... Super Feed® them!** Disponível em:<<http://www.super-feeder.com/>> Acesso em 08 de outubro de 2020.

TAILIO. **Is Your Cats Health and Happiness Important to You?** Disponível em:<<https://www.tailio.com/>> Acesso em 08 de outubro de 2020.

TATOBARI **HX711 for Raspberry Py**. Disponível em:<<https://github.com/tatobari/hx711py>> Acesso em 01 de setembro de 2020.

THOMAZINI, D.; ALBUQUERQUE P. U. B. **Sensores Industriais: Fundamentos e Aplicações**. 8ª ed. São Paulo: Ed. Érica, 2011.

TI. **ULN200x, ULQ200x High-Voltage, High-Current Darlington Transistor Arrays** Disponível em:<<https://www.ti.com/lit/ds/symlink/uln2003a.pdf>> Acesso em 15 de outubro de 2020.

TRICKDROID. **Whistle Go é um animal de estimação portátil que rastreia o sono, arranhões e lambidas** Disponível em:<<https://trickdroid.org/whistle-go-e-um-animal-de-estimacao-portatil-que-rastreia-o-sono-arranhoes-e-lambidas/>> Acesso em 08 de outubro de 2020.

UBUNTU **Terminal** Disponível em:<https://help.ubuntu.com/kubuntu/desktopguide/pt_BR/terminals.html> Acesso em 08 de novembro de 2020.

USINAINFO. **Isolador Óptico PC817 4 Canais 3.6-24V** Disponível em:<<https://www.usinainfo.com.br/outros-modulos-arduino/isolador-optico-pc817-4-canais-36-24v-6207.html>> Acesso em 15 de outubro de 2020.

USTUNDAG, A.; CEVIKCAN, E. **Industry 4.0: Managing The Digital Transformation**. Istanbul: Ed Springer, 2018.

VOGEL-HEUSER, B. **Industrie 4.0 in Produktion, Automatisierung und Logistik: Herausforderungen und Anforderungen aus Sicht der IT und der Automatisierungstechnik**. Ed. Springer, 2014.

VSSSTUDIO. **Sistema del ejemplo del vector de la tecnología de los iconos de los chips de ordenador del circuito** Disponível em:<https://www.freepik.es/vector-premium/sistema-ejemplo-vector-tecnologia-iconos-chips-ordenador-circuito_4832936.htm> Acesso em 03 de agosto de 2020.

WATELECTRONICS. **Know all about Raspberry Pi Board Technology** Disponível em:<<https://www.watelectronics.com/know-all-about-raspberry-pi-board-technology/>> Acesso em 03 de agosto de 2020.

WILCOX, E. **What is Project Automation?** Disponível em:<<https://rindle.com/resources/ebooks/the-ultimate-guide-to-project-automation/chapter->

one/> Acesso em 03 de outubro de 2020.

WOPET. WOPET SmartFeeder, Automatic Pet Dog and Cat Feeder, 6-Meal Auto Pet Feeder with Timer Programmable, HD Camera for Voice and Video Recording, Wi-Fi Enabled App for iPhone and Android Disponível em: <<https://www.amazon.com/WOpet-SmartFeeder-Automatic-Programmable-Recording/dp/B07N5BBFZY>> Acesso em 03 de agosto de 2020.

APÊNDICE A – Telas.

Tela Inicial

A tela inicial “*Home*”, Figura 36, é composta somente do indexador para as demais telas de gerenciamento e consulta, ainda informando de forma sucinta os níveis de ração, água e ração na tigela, e em qual hora foram obtidos.

Figura 36 – Interface gráfica – Tela inicial (Home).



Fonte: Elaborado pelo autor.

Telas de Gerenciamento

As telas de gerenciamento, aqui definidas como “parâmetros” e “agendamento”, estão voltadas unicamente para gerenciamento dos funis (ração e água) pelo tutor, nela será possível efetuar:

Tela de Parâmetros, disponível na Figura 37:

Figura 37 – Interface gráfica – Tela parâmetros

Fonte: Elaborado pelo autor.

E-mail - E-mail para o qual deverá ser enviado os alertas.

Hab. envio e-mail ração - Habilita o envio de *e-mail* para o nível baixo de ração.

Hab. envio e-mail água - Habilita o envio de *e-mail* para o nível baixo de água.

Nível Ração Baixo (cm) - Nível que o tutor define como baixo, para poder planejar a aquisição de ração.

Nível Água Baixo (cm) - Nível que o tutor define como baixo, para reabastecer o reservatório de água.

Hab. refrig. água - Habilitar a opção de refrigerar a água.

Temperatura da água - Temperatura que o tutor define como adequada.

Ração (h) - Após quantas horas o sistema deverá reenviar o *e-mail* de nível baixo de ração.

Água (h) - Após quantas horas o sistema deverá reenviar o *e-mail* de nível baixo de água.

O motivo de haver 2 opções de envio de *e-mail*, é para ser explícito qual *e-mail* o tutor quer receber, podendo a combinação efetuar o envio para nível de ração e água baixos, somente nível de ração baixo e somente nível de água baixo.

Tela de Agendamento, Figura 38:

Figura 38 – Interface gráfica – Tela agendador.

Fonte: Elaborado pelo autor.

Dia da Semana - São listados todos os dias da semana, para que o tutor possa selecionar em quais dias deseja que o agendamento seja executado.

Hora Alimentação - Hora em que o tutor deseja que o agendamento seja executado.

Minuto Alimentação - Minuto em que o tutor deseja que o agendamento seja executado.

Gramas - Quantidade de gramas que o sistema deverá disponibilizar ao *pet*.

Telas de Cadastro – Usuário e Pet

Estas telas são utilizadas para cadastrar usuários e *pets* no sistema:

Tela de Cadastro de Usuário, Figura 39:

Figura 39 – Interface gráfica – Tela Cadastro Usuário

Fonte: Elaborado pelo autor.

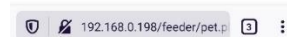
Login – Usuário a ser utilizado

Senha – Senha desejada.

Redigite a senha – Garantia que o usuário está digitando a senha correta.

Tela de Cadastro do *Pet*, disponível na Figura 40:

Figura 40 – Interface gráfica – Tela Cadastro *Pet*



Fonte: Elaborado pelo autor.

Nome do *Pet* – Nome do animal de estimação.

Tipo do *Pet* – Escolher qual é o animal de estimação.

Telas de Consulta e Edição

As telas apresentadas abaixo tem 2 funções, a de consulta e edição. Ao acessá-las é efetuada a consulta ao bando de dados, trazendo a informação ao usuário, e nela há a coluna “Ação”, aonde existem as opções “Hab | Desab | Del”, sendo as funções Habilitar o registro, Desabilitar o registro e Deletar o registro.

A única exceção é a tela de consulta histórico, pois estes dados são gerados de forma automática através da execução do *firmware*, não sendo adequado ao usuário editar estas informações. Na Figura 38, a coluna verde, demonstra o dia da semana no momento da consulta para facilitar a identificação dos agendamentos a serem executados.

Figura 41 apresenta a tela de Consulta Agenda, a Figura 42, evidencia a tela de Consulta Usuário e a Figura 43 mostra a tela de Consulta *Pet*.

Figura 41 – Interface gráfica – Tela consulta e edição agenda



Fonte: Elaborado pelo autor.

Figura 42 – Interface gráfica – Tela consulta e edição usuário



Fonte: Elaborado pelo autor.

Figura 43 – Interface gráfica – Tela consulta e edição Pet



Fonte: Elaborado pelo autor.

Tela de Consulta Parâmetros, Figura 44:

Figura 44 – Interface gráfica – Tela consulta e edição parâmetros



Fonte: Elaborado pelo autor.

Tela de Consulta Histórico, Figura 45:

Figura 45 – Interface gráfica – Tela consulta histórico



Fonte: Elaborado pelo autor.