

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES
ENGENHARIA DE COMPUTAÇÃO**



**A UTILIZAÇÃO DO RFID PARA AUTOMATIZAR FAZENDAS COM BOVINOS DE
CORTE**

ISMAEL CARLOS DO NASCIMENTO GALVÃO

**GOIÂNIA - GO
2024**

ISMAEL CARLOS DO NASCIMENTO GALVÃO

**A UTILIZAÇÃO DO RFID PARA AUTOMATIZAR FAZENDAS COM BOVINOS DE
CORTE**

Trabalho de Conclusão de Curso apresentado à Escola Politécnica e de Artes, da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador:

Prof. MEE. Marcelo Antonio de Araújo

Banca examinadora:

Prof. MEE. Carlos Alexandre Ferreira de Lima

Prof. Me. Miriam Sandra Rosa Gusmão

GOIÂNIA - GO

2024

ISMAEL CARLOS DO NASCIMENTO GALVÃO

**A UTILIZAÇÃO DO RFID PARA AUTOMATIZAR FAZENDAS COM BOVINOS DE
CORTE**

Trabalho de Conclusão de Curso aprovado em sua forma final pela Escola Politécnica e de Artes, da Pontifícia Universidade Católica de Goiás, para obtenção do título de Bacharel em Engenharia de Computação, em: ____/ ____/ _____

Orientador: Prof. MEE. Marcelo Antonio de Araújo

Prof. MEE. Carlos Alexandre Ferreira de Lima

Prof. Me. Miriam Sandra Rosa Gusmão

GOIÂNIA - GO

2024

AGRADECIMENTOS

Primeiramente, agradeço a Deus por tudo que Ele tem me proporcionado. Agradeço à minha família, especialmente aos meus pais e ao meu irmão, juntamente com sua esposa, que me deram ânimo para continuar e não desistir dos meus sonhos. Agradeço à minha querida mãe, Neuzamar do Nascimento Galvão, e ao meu querido pai, Geraldo Cesário Galvão Filho, que trabalharam dia e noite para me dar a oportunidade, assim como ao meu irmão, de correr atrás dos nossos sonhos.

Quero agradecer ao meu tio, Cesário Geraldo Galvão, que abriu as portas da sua casa para que eu e meu irmão pudéssemos morar por um tempo. Agradeço também aos pais da minha cunhada, Talytha Souza Nogueira Rodrigues, o Pastor Martins e a Pastora Elzi, por terem aberto as portas de sua casa para mim durante um período.

Confesso que nesta caminhada houve muitos momentos de altos e baixos. Um simples adolescente de apenas 14 anos, morando a 1.200 km de distância de meus pais e amigos, mas pude demonstrar todos os ensinamentos que meus pais me deram e que continuam a me ensinar a cada conversa e momento de presença. Não posso deixar de agradecer ao meu irmão, Samuel do Nascimento Galvão, que em todos os momentos dessa caminhada me deu apoio, aconselhando-me e ajudando-me.

Essa distância me proporcionou muitas aventuras e várias amizades, que guardarei para sempre. Sou grato a todos os meus amigos que fiz na cidade de Ceres, Goiás.

Agradeço a todos os professores da PUC, que me proporcionaram muitos conhecimentos e aprendizados, formando um profissional que honrará todos os seus ensinamentos. Obrigado a todos os professores, em especial aos professores orientadores M.E.E. Marcelo Antonio Adad de Araújo e aos coorientadores M.E.E. Mírian Sandra Rosa Gusmão e M.E.E. Carlos Alexandre Ferreira de Lima, por terem confiado e me orientado nesta jornada na PUC.

Quero agradecer a todos os meus amigos que a PUC me proporcionou nesta caminhada e que sempre estiveram comigo nos momentos difíceis. Não posso deixar de agradecer especialmente aos meus amigos Divina Jéssica Souza de Freitas, Wemerson Rodrigues da Silva, Higor Ferreira Alves Santos, João Victor Banczeck Rodrigues e Hotny Dias Motta. Sou muito grato a vocês por tudo que fizeram por mim. Noites em claro me ajudando com cálculos, programações e eletrônica. Quantas noites vocês sacrificaram para me ajudar, ouvindo minhas frustrações, desânimos e tristezas. Realmente, não sei como agradecer. Quantas vezes vocês me incentivaram a não desistir, juntamente com minha família. Agradeço muito a Deus por ter colocado vocês

na minha trajetória.

Muito obrigado a todos de coração, amo cada um de vocês.

RESUMO

Este trabalho apresenta uma pesquisa e desenvolvimento de um sistema de automação para fazendas de bovinos de corte utilizando a tecnologia RFID. A implementação envolve o uso de um microcontrolador ESP32 e o módulo RFID-RC522. O objetivo é demonstrar o funcionamento dos circuitos que utilizam o módulo RFID em comunicação com o microcontrolador ESP32, integrados a um sistema Web que gerencia os estoques de medicamentos e rações, além de fornecer relatórios detalhados sobre os bovinos, incluindo os medicamentos e rações administrados. O sistema permite a identificação segura e eficiente dos bovinos através da leitura dos brincos RFID do tipo Mifare (Tag), possibilitando ao usuário acessar todos os dados do animal por meio da leitura do brinco, automatizando assim a gestão da fazenda.

ABSTRACT

This work presents the research and development of an automation system for beef cattle farms using RFID technology. The implementation involves the use of an ESP32 microcontroller and the RFID-RC522 module. The objective is to demonstrate the operation of circuits that use the RFID module in communication with the ESP32 microcontroller, integrated into a web system that manages the inventory of medications and feed, as well as provides detailed reports on the cattle, including administered medications and feed. The system allows for the secure and efficient identification of the cattle through the reading of Mifare-type RFID tags (Tag), enabling the user to access all the animal's data by reading the tag, thereby automating the farm management.

Lista de ilustrações

Figura 1 – Detalhes da utilização do RFID	5
Figura 2 – Acoplamento em Função da Frequência.	7
Figura 3 – Esquema do Chip Mifare.	7
Figura 4 – Chip Mifare	7
Figura 5 – Diagrama de Bloco dos Chips Mifare	8
Figura 6 – Estrutura de Memória do Chip Mifare	9
Figura 7 – Estrutura do Manufacturer Block	10
Figura 8 – Sector Trailer	11
Figura 9 – Fluxograma da Estrutura de Comunicação	12
Figura 10 – Pinout do Módulo RFID RC522	13
Figura 11 – Internet das Coisas	14
Figura 12 – IoT e Suas Aplicações	15
Figura 13 – Pinout do ESP32	16
Figura 14 – Consumo de Energia por Rádio Frequência	18
Figura 15 – Fluxograma Principal	31
Figura 16 – Fluxograma das Opções de Cadastro.	32
Figura 17 – Fluxograma de Relatório	33
Figura 18 – Tela de Login	33
Figura 19 – Cadastrar Usuário	34
Figura 20 – Inserindo Usuário	34
Figura 21 – Cadastro da Fazenda	35
Figura 22 – Inserindo Fazenda	35
Figura 23 – Tela de Perfil	36
Figura 24 – Registrar Medicamento	36
Figura 25 – Registrando Medicamento	37
Figura 26 – Relatório Parcial dos Medicamentos	37
Figura 27 – Relatório Detalhado do Medicamento	38
Figura 28 – Registrar Ração	38
Figura 29 – Registrando Ração	39
Figura 30 – Relatório Parcial das Rações	39
Figura 31 – Relatório detalhado da Ração	40
Figura 32 – Registrar Bovino	40
Figura 33 – Fluxograma do Funcionamento do Leitor	41
Figura 34 – Leitor do Brinco Aguardando Conexão com o WiFi	42
Figura 35 – Leitor do Brinco Configuração Concluída	43
Figura 36 – Leitor de Brinco Aguardando Operações	44

Figura 37 – Registrando o Bovino	45
Figura 38 – Registrando Brinco no Bovino Cadastrado	45
Figura 39 – Fazendo a Leitura do Chip Mifare do Brinco	46
Figura 40 – Bovino Inserido com Sucesso	47
Figura 41 – Erro ao Inserir o Bovino	47
Figura 42 – Relatório Parcial do Bovino	48
Figura 43 – Relatório detalhado do Bovino	48
Figura 44 – Definir Processo Para o Bovino	49
Figura 45 – Vacinar Bovino	49
Figura 46 – Erro ao ler Bovino	50
Figura 47 – Selecionando o Bovino Para Aplicar Medicamento	50
Figura 48 – Inserindo as Informações do Medicamento	51
Figura 49 – Relatório Detalhado do Bovino com Inserção de Medicament	51

LISTA DE SIGLAS

API Application Programming Interface. 27, 28

CERN European Organization for Nuclear Research. 19

CSS Cascading Style Sheet. 21–23, 52

ECMA European Computer Manufacturers Association. 25

EEPROM Electrically-Erasable Programmable Read-Only Memory. 9

EMBRAPA Empresa Brasileira de Pesquisa Agropecuária. 1

IC Integrated Circuit. 12

IDE Integrated Development Environment. 3

IoT Internet of Things. 13

ISM Industrial Scientific and Medical. 6

LCD Liquid Crystal Display. 41

LTS Long Term Support. 28

NPM Node Package Manager. 28

POR Power-On Reset. 11

RFID Radio Frequency Identification. 1, 3–7, 12, 13, 40–42, 54

SVG Scalable Vector Graphics. 19, 21

XHTML eXtensible Hypertext Markup Language. 21

XML eXtensible Markup Language. 19, 21

LISTA DE ABREVIATURAS

ESP Espressif Systems Platform. 3, 5, 13, 15, 17

HTML HyperText Markup Language. 18, 19, 21, 22

HTTP Hypertext Transfer Protocol. 28, 41

JS JavaScript. 23

MAPA Ministério da Agricultura e Pecuária. 1

MathML Mathematical Markup Language. 19, 21

MCU Microcontroller Unit. 17

MHz Megahertz. 6

UID Unique Identifier. 41, 46

UNIX Unified X. 28

W3C World Wide Web Consortium. 19, 21

Web World Wide Web. 2, 3, 18, 21–23, 28, 42, 52

WiFi Wireless Fidelity. 13, 17, 40, 41

Sumário

Lista de ilustrações	i
Sumário	v
1 INTRODUÇÃO	1
1.1 Objetivos	2
1.1.1 Objetivo Geral	2
1.1.2 Objetivos Específicos	2
1.2 Justificativa	2
1.3 Métodos	3
1.4 Organização do Trabalho	4
2 FUNDAMENTAÇÃO TEÓRICA	5
2.1 Identificação por Radio Frequência (RFID)	5
2.2 Frequência de Operação	6
2.2.1 Tipos de Alcance	6
2.3 Chip Mifare	7
2.3.1 Blocos do Chip Mifare	8
2.3.2 Estrutura da memória	9
2.3.3 Bloco do Fabricante	9
2.3.4 Trailer de Setor	10
2.3.5 Estrutura de Comunicação	11
2.4 Módulo RFID RC522	12
2.5 ESP32	13
2.5.1 A Internet das Coisas (IoT)	13
2.5.2 ESP32 Funcionalidade	15
2.6 Linguagem de Marcação de Hipertexto (HTML)	18
2.6.1 Linguagem de Marcação de Hipertexto Versão 5 (HTML5)	19
2.6.2 Vantagens e Desvantagens do HTML e HTML5	20
2.6.2.1 <u>Vantagens do HTML</u>	20
2.6.2.2 <u>Desvantagens do HTML</u>	20
2.6.2.3 <u>Vantagens do HTML5</u>	20
2.6.2.4 <u>Desvantagens do HTML5</u>	20
2.7 Folha de Estilo Cascata (CSS)	21
2.7.1 Vantagens e Desvantagens de utilizar o CSS	21
2.7.1.1 <u>Vantagens do CSS:</u>	21

2.7.1.2	<u>Desvantagens do CSS:</u>	22
2.8	Bootstrap	22
2.9	JavaScript	23
2.9.1	Vantagens e Desvantagens do JavaScript	24
2.9.1.1	<u>Vantagens do JavaScript:</u>	24
2.9.1.2	<u>Desvantagens do JavaScript:</u>	24
2.10	TypeScript	25
2.11	PostgreSQL	25
2.12	Docker	26
2.13	Express Node.js	27
2.13.1	Node.js	28
2.14	Linguagem de Programação C	28
	3 PROPOSTA DE SOLUÇÃO	30
	4 COMO UTILIZAR O SISTEMA DESENVOLVIDO NO PRESENTE	
	TRABALHO	52
4.1	Frontend	52
4.2	Backend	52
4.3	Banco de Dados	53
	5 CONSIDERAÇÕES FINAIS	54
5.1	O Que Foi Desenvolvido	54
5.2	Trabalhos Futuros	55
	REFERÊNCIAS	57

1 INTRODUÇÃO

A Identificação por radiofrequência, do inglês: *Radio-Frequency Identification* (RFID), utiliza ondas eletromagnéticas para comunicar os dados de identificação de alguns componentes, cartões e veículos (PEDROSO; ZWICKER; SOUZA, 2009).

O Brasil, um dos maiores produtores de carnes bovinas no mundo, tem investido em tecnologias, aumentando a produtividade e tendo grandes melhorias na qualidade, tornando-se exportador para mais de 150 países, sendo o segundo maior exportador com 1,9 milhões de toneladas de carnes bovinas do mundo (EMBRAPA, 2023).

“Parte importante da evolução da pecuária brasileira ocorreu dentro das propriedades pecuárias brasileiras, com forte participação de diversos segmentos da sociedade. Engajados na busca por produtividade, qualidade e sustentabilidade, instituições de ciência e tecnologia, ensino, indústria, associações de produtores, organizações não governamentais, entre outros atores, compõem um grupo extremamente atuante e muitas vezes coordenado, com iniciativas que muito contribuem com incrementos na qualidade dentro e fora da porteira (EMBRAPA, b).”

Em 1975, a EMBRAPA estabeleceu o Centro Nacional de Pesquisa de Gado de Corte em Campo Grande, Mato Grosso do Sul, sob a liderança de José Mendes Barcellos. A pesquisa visou melhorar a produtividade da pecuária de corte brasileira, resultando em avanços tecnológicos e transformando o Brasil no principal produtor e exportador de carne bovina. A unidade mantém cooperação com diversas instituições, e seu complexo de pesquisa abriga uma equipe interdisciplinar focada em soluções sustentáveis para a pecuária de corte. A missão da EMBRAPA Gado de Corte é beneficiar a sociedade brasileira por meio de soluções tecnológicas na pecuária de corte. (EMBRAPA, a).

A Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA) é uma empresa pública, tendo a vinculação com Ministério da Agricultura e Pecuária (MAPA). Desde a virada do século a EMBRAPA vem trabalhando com o conceito de rastreabilidade na bovinocultura brasileira, tendo obtido uma série de progressos técnicos-científicos para as fazendas, com sistemas eletrônicos para identificação do bovino bolus, brin-

cos e bottons, registrando todas as informações em um programa (EMBRAPA, 2023).

1.1 Objetivos

Neste Tópico serão apresentados os objetivos de propósito geral e específico do projeto.

1.1.1 *Objetivo Geral*

Este trabalho tem como objetivo integrar diversas tecnologias, tais como: Módulo de Rádio Frequência; Microcontroladores; Tecnologias Web; entre outras, a fim de unir as tecnologias de baixo nível de hardware¹ às tecnologias de alto nível da Web moderna, de modo a criar um sistema coeso que automatiza fazendas com bovinos de corte.

1.1.2 *Objetivos Específicos*

- Gerenciar fazendas com bovinos de corte;
- Interagir com os processos de engorda do animal;
- Definir as rações e medicamentos para melhor detalhamento nos relatórios;
- Sistematizar a vacinação e medicação do bovino;

1.2 Justificativa

Justifica-se o estudo do tema, pois devido aos avanços tecnológicos, o cotidiano e a mão de obra dos serviços braçais vêm sendo facilitados. A automação é o uso da tecnologia para executar tarefas com o mínimo de assistência humana possível, o que revolucionou as áreas nas quais foi implementada. Tais evoluções tem cada vez mais protagonismo no cotidiano. A automação amadureceu de tal forma que várias outras tecnologias se desenvolveram a partir dela (GROOVER, 2023).

Diante deste contexto, no meio agrário, o trabalho visa responder à seguinte questão de pesquisa: - é possível construir um sistema automatizado, usando RFID, que auxilie fazendas de bovinos de corte em suas tarefas elementares?

O presente trabalho apresenta uma proposta para o desenvolvimento de um sistema na execução de tarefas em fazendas, de formas automatizadas. Possibilita o proprietário da fazenda puxar todas as informações de um bovino através de um sistema, para um melhor desempenho e otimização nas execuções de tarefas.

¹ Hardware é a parte física de um sistema computacional. Com hardware, compreende-se as Unidade Lógica de processamento, memórias, discos de armazenamento etc.

Utiliza-se um brinco com um Chip Mifare utilizando o RFID, como identificador do bovino, tendo um sistema intuitivo para requerer as informações necessárias para gerar relatórios, assim gerenciando a vida do animal individualmente apesar do mesmo participar do rebanho. Para tal, serão utilizadas tecnologias Web, tais quais:

- HTML;
- CSS;
- BootStrap;
- TypeScript;
- PostgreSQL;
- Docker;
- Express Node.js;
- C.

1.3 Métodos

O objetivo principal do presente instrumento é desenvolver um sistema RFID para aprimorar as competências na criação e montagem de circuitos destinados a identificação individual de bovinos. A implementação desse sistema é realizado no microcontrolador ESP32 utilizando a linguagem de programação C e o Ambiente de Desenvolvimento Integrado, do inglês: *Integrated Development Environment (IDE) Visual Studio Code* por ser de livre acesso. Para fundamentar o projeto, utilizar-se-á artigos científicos, trabalhos de conclusão de curso, publicações online e manuais do fabricante.

- Pesquisa Preliminares: São realizadas pesquisas aprofundadas para adquirir o conhecimento e as competências necessárias para a elaboração do sistema RFID. Isso incluirá a revisão de literatura acadêmica sobre a tecnologia RFID, o estudo de casos de uso em pecuária, e a análise de soluções tecnológicas existentes.
- Desenvolvimento do Circuito: O circuito de identificação RFID é projetado e montado com base nos requisitos do projeto. O microcontrolador ESP32 é programado em linguagem C, proporcionando uma plataforma robusta e flexível para o desenvolvimento do sistema. A plataforma Visual Studio Code é utilizada para facilitar o desenvolvimento, fornecendo ferramentas de depuração e integração contínua.

- **Identificação por Brinco:** Os brincos RFID são confeccionados com um Chip Mifare, permitindo a identificação única de cada bovino. Cada brinco é codificado com um identificador único lido pelo sistema RFID. A capacidade de codificação e decodificação é integrada ao sistema, assegurando a precisão e a confiabilidade dos dados coletados.
- **Sistema de Coleta de Dados:** É implementado um sistema intuitivo para coleta de informações, permitindo a geração de relatórios personalizados e detalhados. Isso possibilita o gerenciamento individual dos bovinos, monitorando aspectos como crescimento, saúde, alimentação e produtividade. O sistema de coleta de dados é projetado para ser fácil de usar, com uma interface amigável para os operadores da fazenda.

Este projeto tem como propósito ampliar as competências na criação de sistemas RFID para a identificação individual de bovinos, contribuindo para uma gestão mais eficiente e personalizada da vida dos animais. A capacidade de codificação e decodificação dos brincos, juntamente com o sistema de coleta de informações, torna o sistema versátil e valioso na pecuária moderna. Com o uso dessas tecnologias, será possível garantir a identificação única de cada animal e monitorar seu desempenho, saúde e bem-estar de maneira mais eficaz. Isso resultará em uma melhor gestão do rebanho, redução de custos operacionais e aumento da produtividade.

1.4 Organização do Trabalho

O capítulo 1 apresenta a introdução do trabalho de forma em melhor atendimento desse trabalho de final de curso.

O capítulo 2 trás a fundamentação teórica embasando o presente trabalho conforme a literatura, encontrada para respaldar as necessidades encontradas no decorrer do presente trabalho.

O capítulo 3 apresenta a proposta de solução encontrada de forma a racionalizar a aplicação desenvolvida.

O capítulo 4 trás toda a parte pratica necessárias para reproduzir o presente trabalho informando o passo a passo.

O capítulo 5 trás as considerações finais mostrando o que foi desenvolvido e os trabalhos futuros para este projeto.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, é abordada a fundamentação teórica necessária para a compreensão e desenvolvimento do sistema RFID que é baseada em radiofrequência para a identificação individual de bovinos. A fundamentação teórica cobre os conceitos de RFID, a arquitetura, funcionalidade do microcontrolador ESP32 e as tecnologias utilizadas para fazer o Sistema para gerenciar a fazenda.

2.1 Identificação por Radio Frequência (RFID)

O RFID é uma tecnologia que utiliza ondas de rádio para identificação. Consistindo em três elementos: Chip Mifare que armazena informações, um leitor de antena com antena para transmissão e recepção de dados e uma unidade de controle para processamento (COUTO; MALAFAIA, 2019)

O RFID é amplamente utilizado em indústrias, logística, bibliotecas e supermercados devido à sua versatilidade, como é apresentada na Figura 1. Sua principal vantagem é a identificação sem contato direto, o que o diferencia de códigos de barras. A tecnologia RFID continua a evoluir e tem aplicações em constante crescimento (COUTO; MALAFAIA, 2019)

Figura 1 – Detalhes da utilização do RFID



Fonte: (COUTO, MALAFAIA, 2019).

A tecnologia RFID envolve a comunicação entre um transceptor e um transponder ou *tag* por meio de ondas de rádio. O transponder responde com informações que são gerenciadas por um sistema computacional. A transferência de dados pode ocorrer em dois modos: *full duplex* e *half duplex* (COUTO; MALAFAIA, 2019)

- Full Duplex: Os dispositivos podem transmitir e receber dados simultaneamente, como em uma conversa bidirecional, onde ambas as partes podem falar e ouvir ao mesmo tempo.
- Half Duplex: Os dispositivos podem alternar entre a transmissão e a recepção, mas não podem realizar as duas coisas ao mesmo tempo. É semelhante a uma conversa em que apenas uma parte pode falar de cada vez, enquanto a outra escuta.

2.2 Frequência de Operação

A frequência de operação do dispositivo utiliza-se de ondas eletromagnéticas, que são classificadas como sistema de rádio e devem operar sem interferir em outros sistemas ou frequência. As frequências disponíveis para o RFID são chamadas de Industrial Científica e Médica, do inglês: *Industrial Scientific Medical* (ISM). Tendo três frequência mais comuns para o RFID passivos são (COUTO; MALAFAIA, 2019)

- Baixa Frequência: Opera entre 125 kHz a 134,2 kHz, adequada para leituras de curto a médio alcance, com custos mais baixos e velocidades de leitura mais lentas. Usos comuns incluem controle de acesso, identificação de animais e controle de inventário.
- Alta Frequência: Opera a 13,56 MHz, adequada para curto e médio alcance, com custos potencialmente baixos e velocidades médias de leitura. Usos típicos incluem controle de acesso e smart cards.
- Ultra Alta Frequência: Opera em frequências entre 300 MHz e 3 GHz, com uso em diversas áreas, incluindo RFID na faixa de 902 a 928 MHz. Oferece maior alcance, alta velocidade e custos mais elevados. É utilizado em monitoramento de veículos em estradas e coleta automatizada de dados.

Para os fins do presente trabalho foi escolhida a alta frequência de 13,56 MHz para curto e médio alcance.

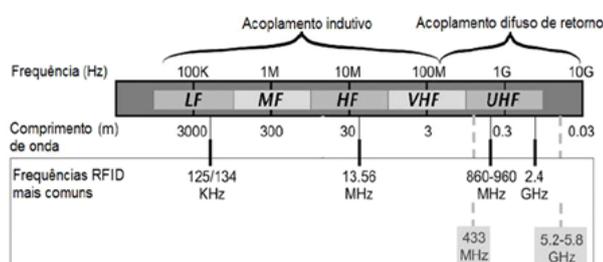
A frequência não apenas afeta a velocidade de transferência de dados, mas também a distância máxima de leitura, a capacidade de penetração em objetos, além de atender a normas e regulamentações. Cada frequência tem características específicas que a tornam adequada para diferentes aplicações (COUTO; MALAFAIA, 2019)

2.2.1 Tipos de Alcance

Para detalhar os componentes de sistema RFID, deve descrever o método de conexão entre o leitor e Chip Mifare no sistema, que pode ser baseado na proximidade com acoplamento indutivo ou na propagação de ondas eletromagnéticas com acoplamento

eletromagnético *backscatter*¹ que envolve a propagação de ondas eletromagnéticas. A Figura 2 ilustra o espectro de frequência no sistema RFID, destacando os distintos tipos de conexão entre a *tag* e o leitor (COUTO; MALAFAIA, 2019)

Figura 2 – Acoplamento em Função da Frequência.

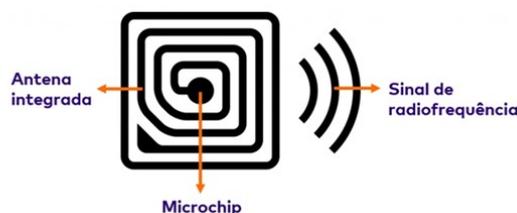


Fonte: (Malafaia, 2023).

2.3 Chip Mifare

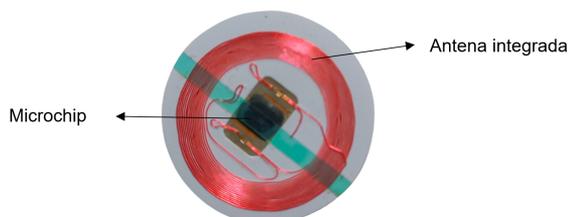
Os Chips Mifare são empregados na faixa de alta frequência da tecnologia RFID, operando especificamente na frequência de 13,56 MHz. São precisos até 10 cm, requerendo apenas uma antena de atenuação, como é representada na Figura 3. Esses Chips poder ser incorporados em cartões, etiquetas, brincos, e uma variedade de outros dispositivos, incluindo implantes em animais (SEMICONDUCTORS, 2017)

Figura 3 – Esquema do Chip Mifare.



Fonte: (COUTO, MALAFAIA, 2019).

Figura 4 – Chip Mifare



Fonte: (Elaborado pelo autor, 2024).

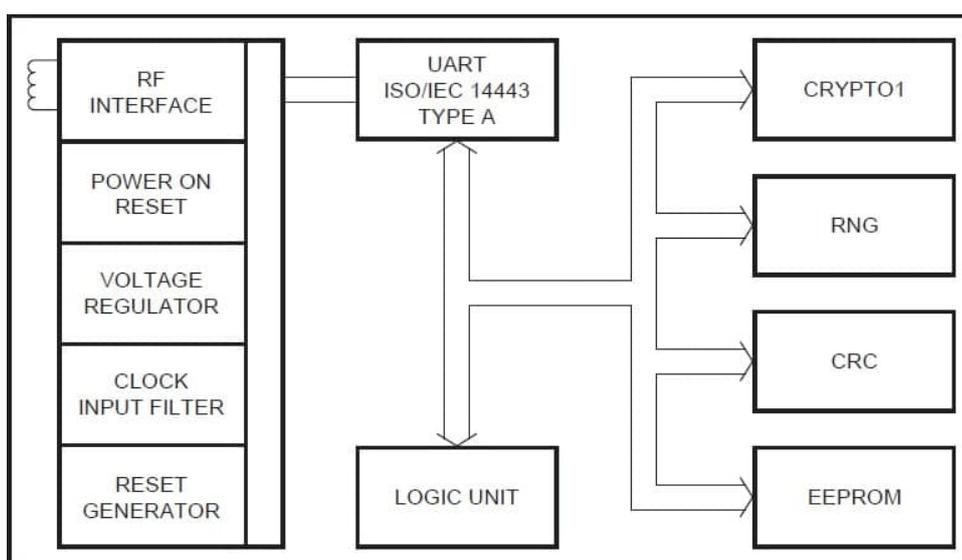
¹ O Backscatter Coefficient é uma medida da quantidade de energia eletromagnética refletida por um alvo quando exposto a um feixe de radiação eletromagnética (FELIPPE, 2023).

2.3.1 Blocos do Chip Mifare

Os chips, em seu interior, são constituídos por diversos blocos, cada qual desempenhando funções específicas essenciais para o seu funcionamento adequado. Os principais blocos incluem (NXP, 2017).

- Bloco de RF: Inclui sistemas de modulação e demodulação do sinal, regulador de tensão, sistema de reinicialização ao ligar Power On Reset (POR) e gerador de clock. Este bloco também incorpora um sistema de anticolisão, conforme a norma ISO-14443, que permite a seleção de um único cartão, mesmo quando vários estão presentes no campo do leitor. Modulador e demodulador; Retificador; Regenerador de relógio; Redefinição de inicialização; Regulador de voltagem.
- Unidade Lógica: Encabeça o controle do lógico dentro do chip, como solicitação de autenticação, operações de leitura e escrita na memória, controle de acesso aos dados e gestão de comandos.
- Unidade de Criptografia: Garante a autenticação da comunicação através do sistema CRYPTO1, criado pela NXP para uso nos chips Mifare, assegurando a segurança na troca de informações.
- EEPROM: Memória interna não volátil onde são armazenados e acessados os dados, permitindo leitura e gravação de informações conforme necessário.

Figura 5 – Diagrama de Bloco dos Chips Mifare



Fonte: (NXP, 2017)

2.3.2 Estrutura da memória

Os chips possuem uma memória interna EEPROM, com capacidades comuns de 1KB, 2KB e 4KB. Internamente, estas memórias são organizadas em setores, que por sua vez são divididos em blocos, cada um contendo 16 bytes. As memórias de 4KB são divididas em 40 setores, alguns com 4 blocos e outros com 16 blocos. As memórias de 2KB são compostas por 32 setores, cada setor contendo 4 blocos. As memórias de 1KB contêm 16 setores, com 4 blocos por setor. A seguir é apresentada na figura 6 a estrutura interna de uma memória de 1KB (NXP, 2017).

Figura 6 – Estrutura de Memória do Chip Mifare

Sector	Block	Byte Number within a Block																Description
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
15	3	Key A				Access Bits			GPB	Key B				Sector Trailer 15				
	2	Data																Data
	1	Data																Data
	0	Data																Data
14	3	Key A				Access Bits			GPB	Key B				Sector Trailer 14				
	2	Data																Data
	1	Data																Data
	0	Data																Data
:	:																	
:	:																	
:	:																	
1	3	Key A				Access Bits			GPB	Key B				Sector Trailer 1				
	2	Data																Data
	1	Data																Data
	0	Data																Data
0	3	Key A				Access Bits			GPB	Key B				Sector Trailer 0				
	2	Data																Data
	1	Data																Data
	0	Manufacturer Block																Manufacturer Block

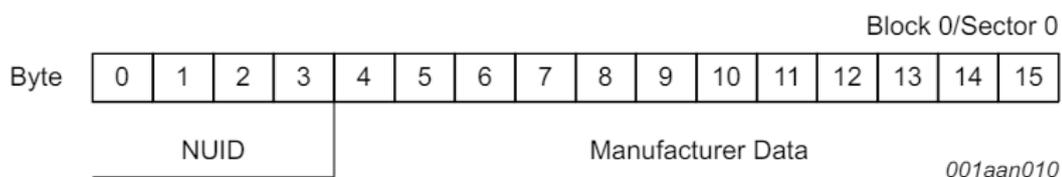
Fonte: (NXP, 2017)

2.3.3 Bloco do Fabricante

O primeiro bloco do primeiro setor é denominado por *manufacturer block* sendo reservado para o armazenamento de informações do fabricante. Cada chip possui um número de série exclusivo, que pode ter 4 ou 7 bytes, indicado pelos primeiros bytes no *manufacturer block*, como é apresentado na Figura 7.

Os três primeiros blocos de cada setor (0 a 2) são destinados ao armazenamento de dados (exceto o bloco 0 do setor 0), podendo ser configurados como blocos de valor (*value blocks*) ou blocos de leitura/escrita (*read/write blocks*). Os blocos de leitura/escrita, conforme o nome indica, são utilizados para a leitura e escrita de dados.

Figura 7 – Estrutura do Manufacturer Block



Fonte: (NXP, 2017)

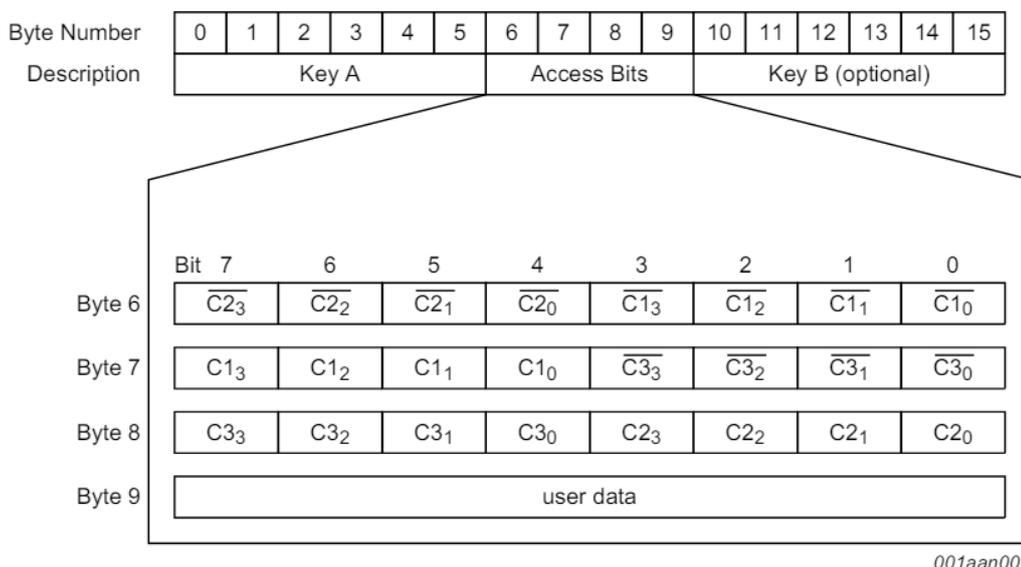
2.3.4 Trailer de Setor

O último bloco de cada setor, denominado *Sector Trailer*, é crucial para o funcionamento do chip. Esse bloco armazena as chaves de cada acesso aos demais blocos do setor, bem como as condições de acesso para cada bloco (NXP, 2017).

Para acessar os blocos dos setores, é necessária a autenticação. O *Sector Trailer* contém duas chaves, *Key A* e *Key B*, sendo a chave B opcional. Uma dessas chaves deve ser utilizada para a autenticação; após a autenticação, os dados dos blocos do setor podem ser acessados conforme definido no campo condição de acesso (NXP, 2017).

O campo condição de acesso é composto por 4 bytes, que armazenam as condições de acesso e o modo de funcionamento de cada bloco do setor. A configuração desse campo determina, por exemplo qual chave deve ser usada para autenticação, o modo de operação dos blocos de dados e se os dados poderão ser acessados. Abaixo a Figura 8 ilustra o *Sector Trailer* (NXP, 2017).

Figura 8 – Sector Trailer



Fonte: (NXP, 2017)

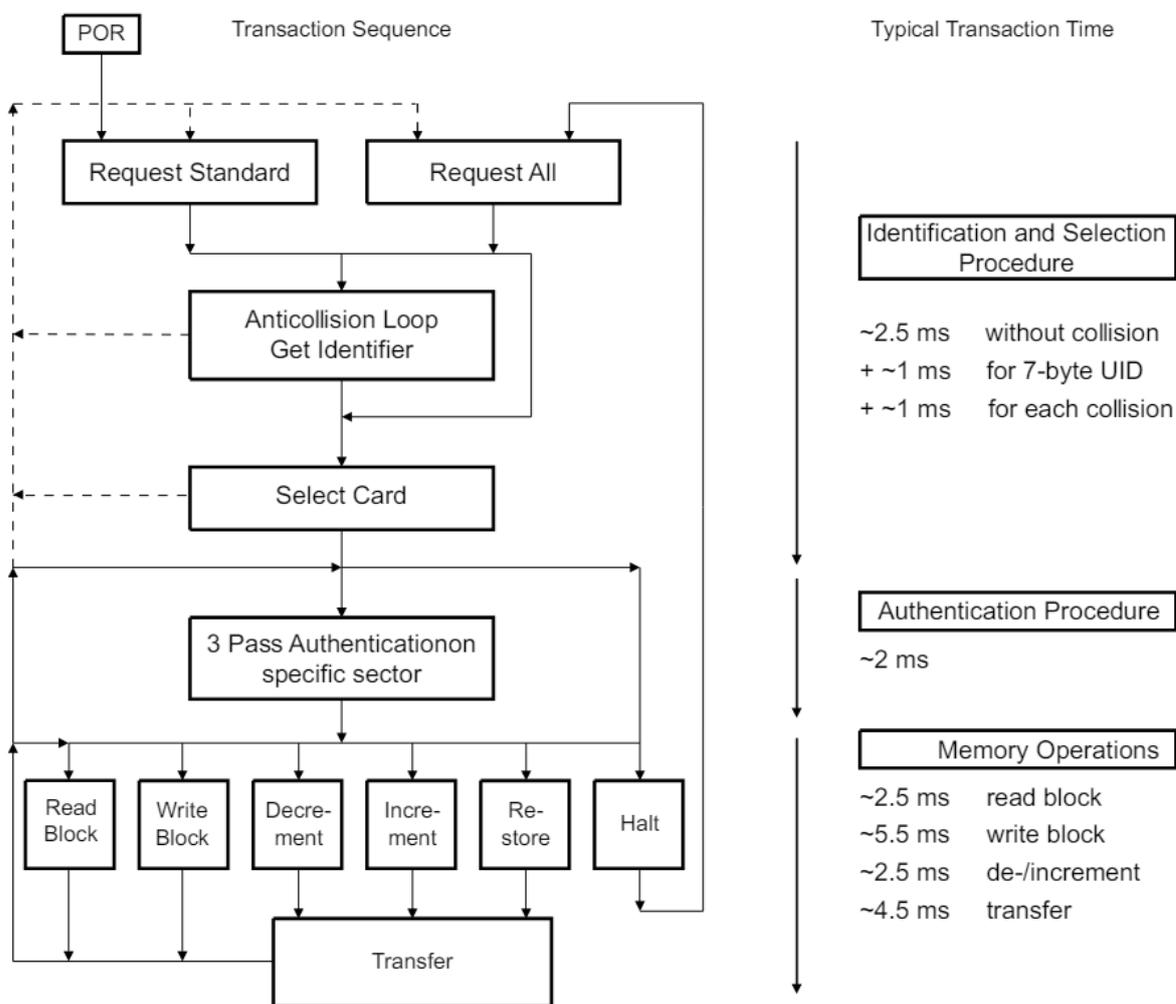
A estrutura da condição de acesso utiliza 3 bits C1, C2 e C3 para definir o funcionamento de cada bloco do setor. Esses bits são armazenados em dois locais por segurança, um de forma normal e outro de forma invertida. Além da identificação do bit 1, 2 ou 3, há um índice adicional que identifica o bloco (NXP, 2017).

2.3.5 Estrutura de Comunicação

A comunicação entre o leitor e os dispositivos Mifare segue uma sequência lógica. Inicialmente, o leitor gera o campo que alimenta os chips POR e envia um *request*. Todos os chips dentro do alcance tentam responder, mas o sistema de anticolisão impede respostas simultâneas, permitindo que apenas um dispositivo responda com seu número identificador. O leitor então seleciona o chip desejado, enquanto os demais entram em modo stand-by, aguardando um novo *request*. Com o chip selecionado, o leitor especifica o local da memória a ser acessado e usa a chave configurada para autenticar o bloco de memória (NXP, 2017).

Após a autenticação, todas as informações trocadas são criptografadas pelo CRYPTO1, uma criptografia desenvolvida pela NXP para dispositivos Mifare, que foi comprometida há alguns anos, tornando recomendável o uso de métodos de segurança adicionais. O leitor então envia o comando para a ação desejada, que pode ser leitura, escrita, incremento, decremento, restauração ou parada. O fluxograma que se segue representa o processo de comunicação.

Figura 9 – Fluxograma da Estrutura de Comunicação



001aan921

Fonte: (NXP, 2017)

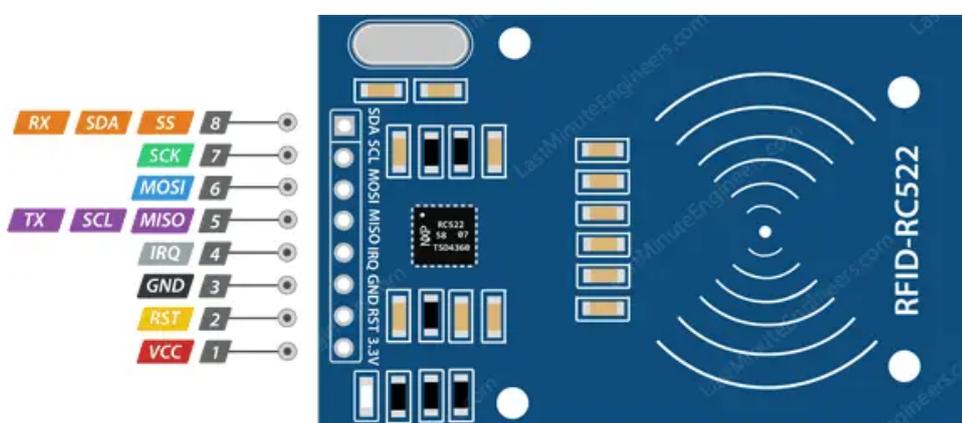
2.4 Módulo RFID RC522

O módulo RFID RC522, baseado no IC MFRC522 da NXP, é uma opção econômica. O RC522 permite a escrita nos Chips Mifare, possibilitando o armazenamento de mensagens personalizadas. O módulo RC522 tem um total de 8 pinos, como mostra a figura 10 apresentando o *Pinout*, tendo as seguintes funcionalidades (LAST-MINUTEENGINEERS, 2024):

1. Campo Eletromagnético: O RC522 gera um campo eletromagnético de 13,56 MHz, compatível com etiquetas RFID padrão ISO 14443A.
2. Comunicação: O módulo comunica-se com microcontroladores através de uma interface SPI de 4 pinos, com taxa de dados máxima de 10 Mbps. Também suporta os protocolos I2C e UART.

3. Interrupção: Pode ser programado para gerar uma interrupção, notificando o microcontrolador sobre a presença de uma tag RFID, eliminando a necessidade de sondagens constantes.
4. Interrupção: Pode ser programado para gerar uma interrupção, notificando o microcontrolador sobre a presença de uma tag RFID, eliminando a necessidade de sondagens constantes.
5. Tensão de Operação: Opera entre 2,5V e 3,3V. Seus pinos lógicos são tolerantes a 5V, permitindo facilmente conexão com microcontroladores de 5V, como o Arduino, sem necessidade de conversores de nível lógico.

Figura 10 – Pinout do Módulo RFID RC522



Fonte: (LASTMINUTEENGINEERS, 2024)

2.5 ESP32

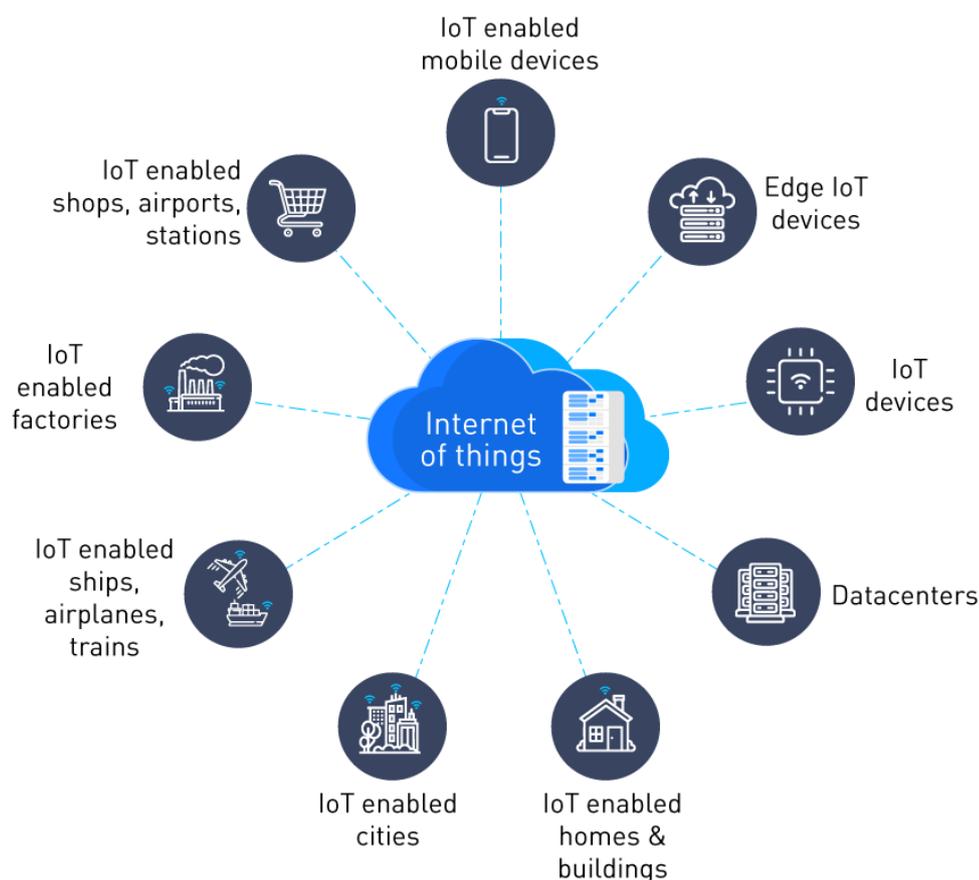
A Plataforma de Sistema Espressif do inglês: Espressif System Platform (ESP), o ESP8266 revolucionou o mundo da Internet das Coisas (IoT) ao oferecer um microcontrolador programável com WiFi com baixo custo, permitindo o monitoramento e controle de dispositivo de qualquer lugar. Após o sucesso do ESP8266, a Espressif System lançou o ESP32, não apenas inclui o WiFi, mas como o *Bluetooth* 4.0, tornando-o ideal para aplicações IoT (LASTMINUTEENGINEERS, 2024).

2.5.1 A Internet das Coisas (IoT)

A Internet das Coisas (IoT) conecta dispositivos físicos à internet, permitindo comunicação e automação. A IoT surgiu com a expansão da internet e avanços tecnológicos como RFID e sensores, e possibilita aplicações em diversas áreas, como saúde e automação industrial, como apresentado na figura 10. Apesar dos benefícios, a IoT traz preocupações com privacidade, devido à coleta e compartilhamento

de dados. Questões legais e éticas são essenciais para a proteção da privacidade e a segurança dos dados, exigindo regulamentações adequadas para equilibrar inovação e proteção dos usuários (SANTOS; SALES, 2016).

Figura 11 – Internet das Coisas



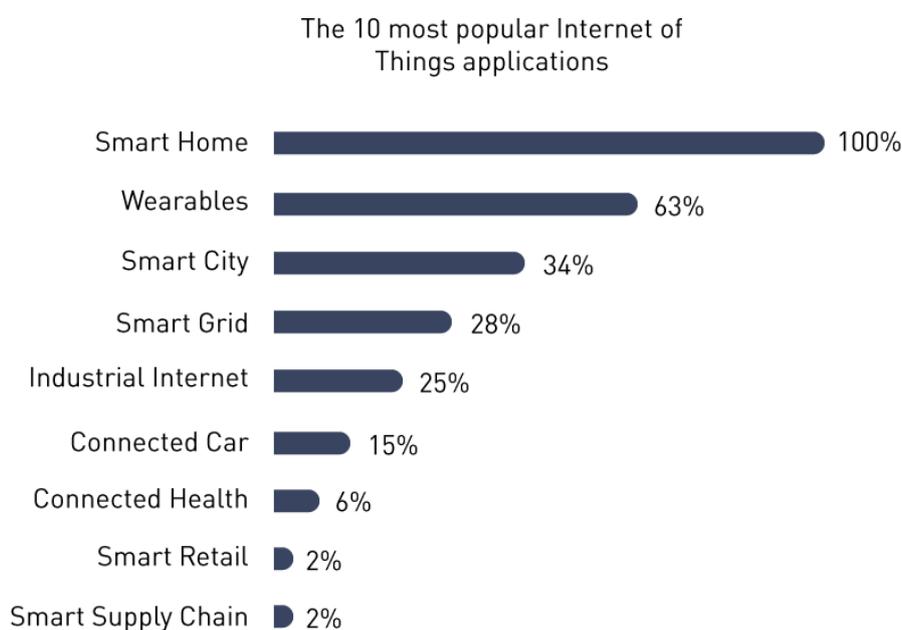
Fonte: (MPS, 2024)

A IoT promove automação e regulação em diversas aplicações, resultando em maior eficácia, melhor tomada de decisões e surgimento de novos serviços. Em uma cidade inteligente, por exemplo, melhorando a administração do tráfego, a gestão de resíduos e consumo de energia, elevando a eficiência e sustentabilidade urbana conforme pode-se verificar na figura anterior (MPS, 2024):

- Casa Inteligente: Introduz dispositivos como termostato, luzes e fechaduras inteligentes, aprimorando a automação e gestão doméstica, podendo ser controlados remotamente.
- Dispositivos Vestíveis: Gadgets como *smartwatches fitness* coletam dados sobre saúde e condicionamento físico, monitorando frequência cardíacas, passos, ciclos de sono dentre outros.

- IoT Industrial: Monitoram e coletam dados de diversos parâmetros como temperatura, pressão e vibrações. Esses dados são utilizados para manutenção preventiva, permitindo o reparo de máquina antes de falhas.
- Cidades Inteligentes: Ambiente urbanos incluem semáforos inteligentes que se ajustam as condições do tráfego, lixeiras que sinalizam quando precisam ser esvaziadas e sensores que monitoram em tempo real a qualidade do ar e da água.
- Agricultura: Monitoram níveis de umidade do solo, crescimento das plantações e comportamento do gado, otimizando a produção e a gestão das fazendas.
- Saúde: Monitora remotamente pacientes, gestão de suprimentos de medicamentos e integrações de equipamentos avançados em procedimentos cirúrgicos.

Figura 12 – IoT e Suas Aplicações



Fonte: (MPS, 2024)

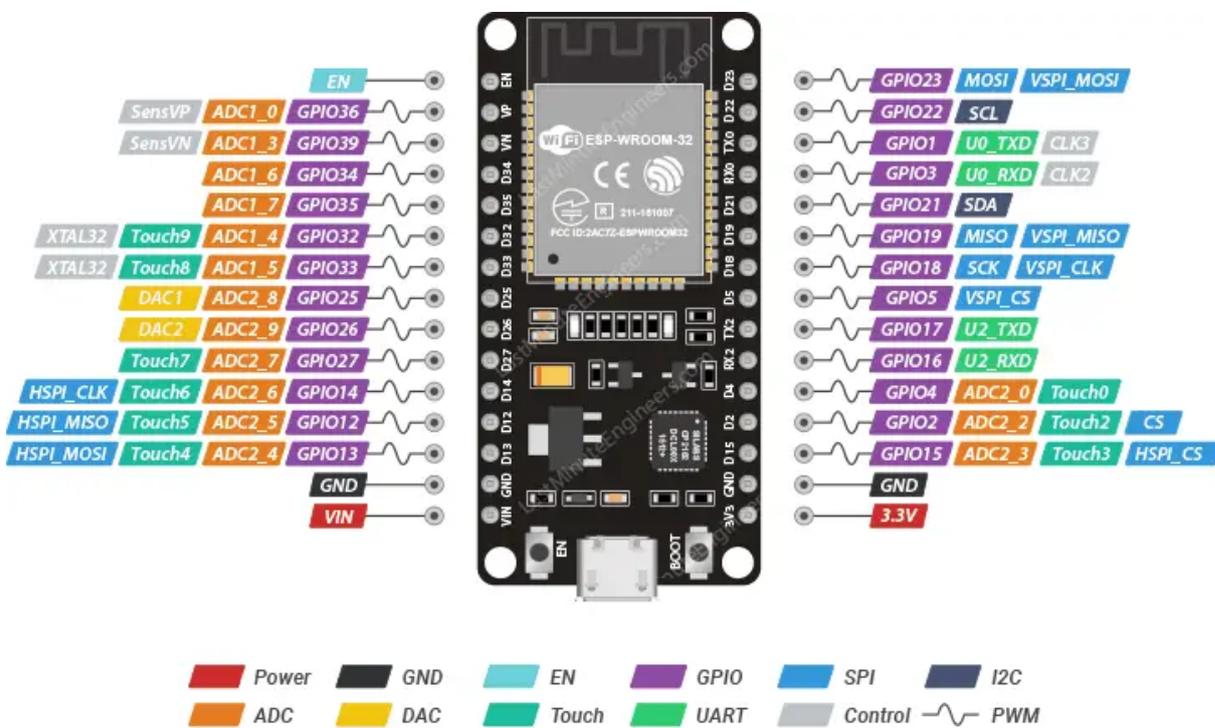
2.5.2 ESP32 Funcionalidade

O ESP32 é uma série de microcontroladores de 36 pinos, (como demonstra a figura acima, Figura 12), desenvolvidos pela Espressif Systems, uma empresa chinesa com sede em Xangai, conhecida por seus dispositivos de baixo custo e alto desempenho tendo as seguintes características:

1. **Processador Dual-Core:** O ESP32 possui um microprocessador Tensilica Xtensa® Dual-Core LX6 de 32 bits, operando a frequências de 80 a 240 MHz, com desempenho de até 600 Dhrystone Million Instructions Per Second (DMIPS)
2. **Consumo de Energia Ultrabaixo:** Com modos de energia variados e escalonamento dinâmico de energia, o ESP32 é ideal para dispositivos móveis e vestíveis, além de outras aplicações IoT.
3. **Capacidades Bluetooth:** Suporta Bluetooth 4.0 e Bluetooth Classic, ampliando suas possibilidades de aplicação.
4. **Conjunto de Periféricos:** Inclui periféricos como toque capacitivo, ADCs, DACs, UART, SPI, I2C e PWM, entre outros.
5. **Plataformas de Desenvolvimento:** Existem diversas plataformas para programação do ESP32, como Espruino, Mongoose OS, MicroPython, SDKs da Espressif e Arduino. A comunidade criou um complemento para Arduino, recomendando-se começar com este ambiente para iniciantes.

(LASTMINUTEENGINEERS, 2024)

Figura 13 – Pinout do ESP32



ESP32 Dev. Board Pinout



Fonte: (LASTMINUTEENGINEERS, 2024)

O ESP32 se destaca em comparação com muitos outros dispositivos WiFi/MCU, tanto em desempenho quanto em preço. No entanto, dependendo do modo de operação, pode ser um dispositivo consumidor de energia. Quando alimentado por uma tomada, o consumo de energia não é uma preocupação significativa. Em projetos alimentados por bateria, gestão de consumo de energia torna-se crucial (LASTMINUTEENGINEERS, 2024).

O ESP32 possui cinco modos de energia configuráveis, que permitem alterar conforme a necessidade de consumo:

- **Modo Ativo:** Todos os periféricos do chip permanecem operacionais. Este é o modo de maior consumo de energia, com o chip consumindo cerca de 240 mA, podendo exceder 790 mA quando WiFi e Bluetooth são usados simultaneamente.
- **Modo Modem Sleep:** Tudo permanece ativo, exceto os módulos WiFi, Bluetooth e rádio. O chip consome aproximadamente 3 mA em baixa velocidade e 20 mA em alta velocidade. Este modo permite que o ESP32 mantenha a conexão ativa, despertando WiFi, Bluetooth e rádio em intervalos pré-definidos.
- **Modo Light Sleep:** Este modo é semelhante ao modem sleep, mas com o CPU, a maior parte da RAM e os periféricos digitais sendo temporariamente desativados. O consumo de energia reduz para cerca de 0,8 mA.
- **Modo Deep Sleep:** Todos os periféricos digitais são desativados, com consumo variando entre 0,15 mA e 10 μ A.
- **Modo Hibernação:** Desativa o oscilador interno de 8 MHz e o ULP-coprocessador, deixando apenas um temporizador RTC e alguns GPIOs RTC para despertar o chip. O consumo de energia é reduzido para cerca de 2,5 μ A, sendo útil para projetos que não precisam estar ativos constantemente.

(ENGINEERS, 2023).

O consumo de energia do ESP32 durante as operações por rádio frequência é conforme verifica-se na Figura 14:

Figura 14 – Consumo de Energia por Rádio Frequência

Modo	Consumo de energia
Pacote Tx Wi-Fi 13dBm~21dBm	160~260mA
Pacote Wi-Fi/BT Tx 0dBm	120mA
Wi-Fi/BT Rx e audição	80~90mA

Fonte: (ENGINEERS, 2023).

2.6 Linguagem de Marcação de Hipertexto (HTML)

A Linguagem de Marcação de Hipertexto (HTML) é um sistema computacional fundamental que compõe a grande maioria das páginas da internet e dos aplicativos online. O termo "hipertexto" se refere a um tipo de texto que inclui referências a outros textos, permitindo uma interconexão de informações. Uma "linguagem de marcação" é composta por uma série de marcações ou etiquetas que servem para instruir os servidores da Web sobre como formatar e estruturar um documento (ANDREI, 2023).

É importante observar que o HTML não se enquadra na categoria de linguagem de programação, uma vez que não possui a capacidade de criar funcionalidades dinâmicas. Em vez disso, ele possibilita aos usuários criarem e organizar seções, parágrafos e links por meio de elementos, *tags* e atributos, tornando a informação mais legível e navegável:

1. Desenvolvimento Web: Os desenvolvedores utilizam códigos HTML para projetar como um navegador exibirá os elementos das páginas, incluindo textos, hiperlinks e conteúdo multimídia. Isso engloba a formatação estética e a estruturação lógica do conteúdo, permitindo que as páginas sejam visualmente atraentes e bem-organizadas.
2. Navegação na Internet: Os usuários podem explorar a Web de forma intuitiva e estabelecer conexões entre páginas e sites relacionados, uma vez que o HTML é amplamente empregado para criar hiperlinks. Esses hiperlinks funcionam como portas de entrada para informações adicionais, possibilitando uma experiência de navegação mais rica e interativa.
3. Documentação e Formatação de Documento: O HTML facilita a organização e a formatação de documentos, de maneira semelhante ao que é feito em aplicativos de edição de texto, como o Microsoft Word. Isso inclui recursos como a criação

de cabeçalhos, listas, tabelas e outros elementos de formatação, o que aprimora a apresentação e a legibilidade de documentos publicados online.

(ANDREI, 2023)(??)

Adicionalmente, é importante destacar que o HTML desempenha um papel central na experiência online, servindo como base para o desenvolvimento de páginas Web interativas e bem estruturadas. Ele atua em conjunto com outros idiomas, como o Folha de Estilo em Cascata, do inglês: *Cascading Style Sheets* (CSS) o JavaScript, para criar uma experiência Web dinâmica e envolvente, repleta de funcionalidades interativas (ANDREI, 2023).

O HTML foi concebido por Tim Berners-Lee, um físico do CERN na Suíça, que propôs um sistema de hipertexto online. Hipertexto denota texto com links para outros conteúdos acessíveis instantaneamente. A primeira versão do HTML, contendo 18 *tags*, foi lançada por ele em 1991. O HTML evoluiu com o tempo, adicionando novas *tags* e atributos. Atualmente, é um padrão web oficial, com o marco sendo o lançamento do HTML5 em 2014, que introduziu *tags* semânticas como `<article>`, `<header>` e `<footer>` (ANDREI, 2023).

2.6.1 Linguagem de Marcação de Hipertexto Versão 5 (HTML5)

O HTML tem experimentado diversas evoluções desde o seu lançamento. O W3C regularmente publica novas versões e atualizações, enquanto marcos históricos também recebem denominações específicas (ANDREI, 2023).

Em 1999, foi publicado o HTML4, frequentemente referido como "HTML". No entanto, a atualização mais notável e recente ocorreu em 2014 com o lançamento do HTML5. Essa versão trouxe inúmeras melhorias, incluindo a capacidade de incorporar vídeos e áudios nativamente, eliminando a necessidade do Flash Player. Além disso, o HTML5 oferece suporte para gráficos vetoriais escaláveis (SVG) e outro exemplo o qual não será utilizado no decorrer do projeto: O MathML para representação de fórmulas científicas e matemáticas (ANDREI, 2023).

Uma outra adição relevante do HTML5 foi a introdução de *tags* semânticas, tais como `<article>`, `<section>`, `<aside>`, `<header>` e `<footer>`, que auxiliam na comunicação do significado do conteúdo tanto para os navegadores quanto para os mecanismos de busca (ANDREI, 2023).

SVG é uma linguagem baseada em XML² usada para representar gráficos bidimensionais. Sua estrutura facilita a associação com manipuladores de eventos JavaScript, permitindo interatividade. Cada figura em SVG é tratada como um objeto separado, e qualquer alteração nos atributos desses objetos provoca uma renderização automática pelo navegador. A renderização da tela ocorre de forma incremental, pixel por

² XML é uma ferramenta independente de software e de hardware para armazenar e transportar dados (W3SCHOOLS, 2024).

pixel. Em contraste, no contexto do elemento canvas, uma vez que o gráfico é desenhado, o navegador não retém as informações. Isso exige um redesenho completo da cena em caso de alterações na posição dos objetos, incluindo aqueles ocultos pelo gráfico (W3SCHOOLS, 2024).

2.6.2 Vantagens e Desvantagens do HTML e HTML5

2.6.2.1 Vantagens do HTML

- **Compatibilidade:** O HTML é amplamente suportado por todos os navegadores, tornando-o uma escolha segura e confiável para criação de páginas da web.
- **Simplicidade:** O HTML é mais simples e fácil de aprender em comparação com versões mais recentes. É uma ótima opção para páginas web básicas.

2.6.2.2 Desvantagens do HTML

- **Limitações de Recursos:** O HTML mais antigo (como o HTML 4) possui limitações significativas em termos de recursos, como multimídia incorporada, gráficos vetoriais e recursos avançados de formulários.
- **Falta de Semântica:** O HTML mais antigo carece de muitas das tags semânticas introduzidas no HTML5, o que pode prejudicar a compreensão do significado do conteúdo por navegadores e mecanismos de busca.

2.6.2.3 Vantagens do HTML5

- **Multimídia Integrada:** O HTML5 introduziu tags de vídeo e áudio nativas, eliminando a necessidade de plug-ins como o Flash Player. Isso torna a incorporação de mídia mais simples e eficiente.
- **Semântica Aprimorada:** O HTML5 inclui tags semânticas, como <article>, <section> e <header>, que ajudam a definir melhor a estrutura do conteúdo, tornando-o mais compreensível para navegadores e mecanismos de busca.
- **Recursos Gráficos Avançados:** O HTML5 oferece suporte a gráficos vetoriais (SVG) e elementos <canvas> para criar gráficos e animações interativas.
- **Suporte a Aplicativos Web:** O HTML5 possibilita o desenvolvimento de aplicativos web mais avançados, aproveitando recursos como armazenamento local, geolocalização e notificações.

2.6.2.4 Desvantagens do HTML5

- **Compatibilidade de Navegadores:** Embora a compatibilidade do HTML5 seja geralmente boa, ainda pode haver diferenças na implementação entre navegado-

res, o que pode exigir ajustes para garantir uma experiência consistente para os usuários.

- Curva de Aprendizado: O HTML5 é mais complexo do que versões anteriores do HTML, o que pode tornar sua aprendizagem um desafio para iniciantes.

2.7 Folha de Estilo Cascata (CSS)

A linguagem de Folha de Estilo em Cascata, do inglês: *Cascading Style Sheets* (CSS). é uma linguagem de formatação usada para definir a aparência de documentosHTML e XML, incluindo linguagens como SVG, MathML e XHTML. Ela controla como os elementos que são exibidos na tela (ARIANE, 2022).

O CSS é uma parte essencial da Web e segue os padrões estabelecidos pelo W3C. Ele é dividido em níveis, onde o CSS1 é considerado obsoleto, o CSS2 é amplamente adotado, e o CSS3, agora organizado em módulos menores, está em contínua evolução em direção à sua padronização (ARIANE, 2022).

Para aqueles que estão começando no desenvolvimento Web, existem tutoriais CSS disponíveis, abrangendo desde noções básicas até conceitos avançados. Além disso, uma referência completa do CSS é fornecida para desenvolvedores experientes, explicando todas as propriedades e conceitos da linguagem. É importante notar que o CSS é fundamental para criar designs atraentes e funcionais em páginas da Web, melhorando a experiência do usuário e a acessibilidade (ARIANE, 2022).

O CSS permite criar diversas funcionalidades em páginas Web sem recorrer a linguagens mais complexas, como o JavaScript. Quando usado com parcimônia, o CSS pode aprimorar significativamente a experiência tanto dos desenvolvedores quanto dos usuários. Com o CSS, é possível desenvolver animações complexas, efeitos de parallax para criar profundidade nas imagens de fundo, sites interativos e até mesmo jogos. (ARIANE, 2022).

HTML5 e CSS3. A sua sintaxe envolve seletores que identificam elementos HTML a serem estilizados e blocos de declaração contendo regras que definem o estilo desses elementos. Cada regra compreende uma propriedade CSS e um valor, separados por dois pontos, e é finalizada com um ponto-e-vírgula, sendo agrupadas em blocos delimitados por chaves (ARIANE, 2022).

2.7.1 Vantagens e Desvantagens de utilizar o CSS

2.7.1.1 Vantagens do CSS:

- Separação de Conteúdo e Estilo: O CSS permite separar o conteúdo HTML da formatação visual, facilitando a manutenção e a atualização do design de um site sem afetar o conteúdo.

- **Flexibilidade e Controle:** CSS oferece controle preciso sobre a apresentação de elementos, permitindo personalização de layouts e estilos de forma detalhada.
- **Reutilização do Estilos:** Pode criar estilos reutilizáveis em todo o site, economizando tempo e esforço na criação de páginas consistentes.
- **Eficiência de Carregamento:** CSS é mais leve do que métodos de estilo inline (em linha), tornando o carregamento das páginas mais rápido.
- **Compatibilidade com Dispositivos Múltiplos:** CSS torna possível criar layouts responsivos que se adaptam a diferentes tamanhos de tela e dispositivos.

2.7.1.2 Desvantagens do CSS:

- **Curva de Aprendizado:** Para iniciantes, o CSS pode ser desafiador de aprender devido à sua sintaxe e aos conceitos envolvidos.
- **Compatibilidade entre Navegadores:** Apesar dos padrões, a exibição de páginas pode variar entre diferentes navegadores, exigindo ajustes para garantir a consistência.
- **Complexidade em Projetos Grandes:** Em projetos extensos, o CSS pode se tornar complexo e difícil de gerenciar, requerendo uma organização cuidadosa.
- **Limitações Gráficas:** O CSS pode ter limitações na criação de elementos gráficos complexos, especialmente em comparação com ferramentas de design gráfico.
- **Desafios de Depuração:** Encontrar e corrigir problemas de estilo (bugs) no CSS pode ser uma tarefa complexa, especialmente em layouts complexos.

No geral, o CSS é uma parte fundamental do desenvolvimento Web, oferecendo um equilíbrio entre flexibilidade e controle na estilização de páginas. Suas desvantagens são geralmente superadas por suas vantagens, especialmente com o uso adequado e boas práticas de desenvolvimento (ARIANE, 2022).

2.8 Bootstrap

O Bootstrap é um framework de código aberto amplamente utilizado para o desenvolvimento de interfaces Web responsivas e móveis. Criado pela equipe do Twitter, esse framework oferece um conjunto de ferramentas, estilos e componentes pré-projetados que simplificam significativamente o processo de criação de páginas Web modernas e visualmente atraentes (OTTO; THORNTON, 2023).

Baseado em HTML, CSS e JavaScript, sua popularidade se deve em parte à facilidade de uso e à eficiência que oferece aos desenvolvedores. Além disso, ele é compatível com vários navegadores e possui uma comunidade ativa, o que significa

que há uma ampla gama de recursos e suporte disponíveis para os desenvolvedores que escolhem utilizá-lo em seus projetos (OTTO; THORNTON, 2023).

Oferece um conjunto robusto de recursos para responsividade de layouts em páginas Web, utilizando conceitos como contêiner, grade, colunas, calhas, grade CSS (OTTO; THORNTON, 2023).

- **Contêiner:** é a base estrutural do Bootstrap. Envolve o conteúdo da página e centraliza-o, garantindo margens consistentes em dispositivos diferentes.
- **Grade:** sistema de grade flexível, permitindo criações de layouts da página. As colunas são incrivelmente flexíveis, baseado em doze colunas, e as classes como `col` e `col-md-` são empregadas para definir a largura das colunas em diferentes tamanhos de tela, tendo o tamanho máximo de doze para manter a consistência.
- **Colunas:** as classes `col` são fundamentais para criar o layout da página. Elas são combinadas com identificadores de tamanho de tela, como `col-md-` para determinar o como as colunas se comportará em diferentes breakpoints.
- **Calhas:** referem-se ao espaçamento entre as colunas. O Bootstrap facilita o controle do espaçamento horizontal entre as colunas usando as classes `g-` e `gx-` para definir as calhas.
- **Grade CSS:** oferecendo flexibilidade e simplicidade na personalização. Essas classes intuitivas podem ser combinadas para criar layouts complexos de maneira eficiente, proporcionando uma base sólida para o desenvolvimento de designs responsivos e atrativos. Isso dá aos desenvolvedores a flexibilidade necessária para atender às exigências de diversos dispositivos.

2.9 JavaScript

JavaScript, por vezes abreviado para JS é uma das linguagens de programação mais populares e versátil, amplamente utilizada para desenvolver uma variedade de aplicações e frameworks para diferentes dispositivos. Uma linguagem de alto nível criada em 1996 por Brendan Eich, um dos fundadores do Mozilla Corporation. Desenvolvida inicialmente para o navegador Netscape Navigator, sua finalidade era simplificar processos dentro de páginas web, facilitando a criação de animações e alertas. Embora frequentemente confundida com Java, JavaScript é uma linguagem distinta, sem ligação direta com Java além do nome similar (ESTRELLA, 2023)

Desde seu lançamento, JavaScript evoluiu significativamente, ampliando suas funcionalidades e aplicações. Hoje, além de ser usada para desenvolvimento Web, JavaScript é fundamental na criação de aplicativos desktop e móveis (ESTRELLA, 2023)

- **Aplicação Web:** Permite a implementação de elementos complexos em páginas web, como animações, gráficos interativos e atualizações dinâmicas. É a terceira camada essencial no desenvolvimento web, complementando HTML e CSS.
- **Aplicação Desktop:** Frameworks como Electron JS permitem o desenvolvimento de aplicativos desktop utilizando tecnologias web (HTML, CSS e JavaScript). Exemplos de aplicativos criados com Electron incluem Slack e GitHub Desktop.
- **Aplicação Mobile:** Frameworks como Ionic e VueJS permitem a criação de aplicativos móveis com JavaScript, compatíveis com diversos sistemas operacionais móveis.
- **Desenvolvimento de Jogos:** JavaScript também é utilizado no desenvolvimento de jogos, com frameworks como Phaser, GDevelop e Kiwi.js facilitando a criação de jogos completos.

2.9.1 Vantagens e Desvantagens do JavaScript

2.9.1.1 Vantagens do JavaScript:

- **Interpretação Diretamente pelos Navegadores:** Não necessita de compilador.
- **Facilidade de Aprendizado:** Mais acessível para iniciantes.
- **Depuração Simplificada:** Erros são mais fáceis de localizar e corrigir.
- **Interatividade:** Permite criar sites mais interativos, melhorando a experiência do usuário (UX).
- **Compatibilidade:** Funciona em várias plataformas e navegadores.
- **Validação de Entradas:** Reduz a necessidade de verificações manuais de dados.
- **Desempenho:** É mais rápido e leve comparado a outras linguagens.

2.9.1.2 Desvantagens do JavaScript:

- **Vulnerabilidade de Segurança:** Popularidade atrai hackers.
- **Execução do Código:** Pode ser explorado para fins maliciosos.
- **Inconsistência de Suporte:** Nem todos os navegadores e dispositivos oferecem suporte total.
- **Código Extenso:** Trechos de código em JavaScript podem ser grandes.
- **Renderização Variável:** Pode apresentar inconsistências em diferentes dispositivos.

Após seu lançamento, JavaScript foi rapidamente adotado pela Microsoft em seu navegador, o que ajudou a consolidá-lo como uma linguagem principal. A introdução da Engine V8 pela Google para o navegador Chrome impulsionou ainda mais sua performance e adoção (ESTRELLA, 2023)

2.10 TypeScript

TypeScript é uma extensão de JavaScript que adiciona Funcionalidades não nativas como tipagem de dados e orientação a objetos. Desenvolvido pela Microsoft, ele oferece suporte aprimorado para programação orientada a objeto e permite uma sintaxe mais clara para escrever classes e definir os tipos de dados (EDSON, 2024)

Lançado em uma versão 1.8, foi desenvolvido por Anders Hejlsberg, criador da linguagem C#, e sua equipe na Microsoft. O projeto é *open source* baseado nos padrões ECMAScript 6³, compilando o código TypeScript em JavaScript (EDSON, 2024)

TypeScript introduz tipagem estática, incluindo conceitos como encapsulamento, herança, abstração e polimorfismo permitindo definir tipos de dados para variáveis, funções e propriedades de classe, facilitando a detecção de erros durante o desenvolvimento (EDSON, 2024)

- Encapsulamento: Define pontos de acessos específicos para o ambiente externo, controlando a visibilidade e acessibilidade dos elementos internos da classe.
- Abstração: Destaca as características úteis dos elementos do mundo real para o sistema, reunindo-as em classes.
- Polimorfismo: Objetos podem assumir diferentes formas em determinadas situações, mantendo uma relação com sua definição inicial.

2.11 PostgreSQL

PostgreSQL, anteriormente conhecido como Postgres95, é um sistema de banco de dados objeto relacional de código aberto que utiliza e estende a linguagem SQL. Suas origens no projeto Postgres da Universidade da Califórnia em Berkley em 1986, PostgreSQL evoluiu para incluir um conjunto robusto de recursos que garante o armazenamento seguro e escalável de cargas de dados complexas. A mudança de nome para PostgreSQL ocorreu em 1996 para refletir essas evoluções e incorporar as funcionalidades SQL (POSTGRESQL, 2024).

PostgreSQL é amplamente reconhecido por sua arquitetura confiável, integridade de dados, conjunto de recursos robustos, extensibilidade e suporte da comunidade de código aberto. Sendo compatível com todos os principais sistemas operacionais

³ “ECMAScript é uma especificação de linguagem de script com marca registrada padronizada pela Ecma International nos padrões ECMA-262 e ISO/IEC 16262.” (EDSON, 2024).

e, desde 2001, está em conformidade com ACID⁴. Além disso, extensões poderosas, como o PostGIS⁵ para dados geoespaciais, aumentam sua funcionalidade. A facilidade de uso e a robustez fazem do PostgreSQL a escolha preferida para muitos desenvolvedores e organizações (POSTGRESQL, 2024).

PostgreSQL oferece uma ampla gama de recursos para desenvolvedores na construção de aplicações, administradores na proteção da integridade dos dados e na criação de ambientes tolerantes a falhas, além de permitir a gestão de dados independentemente do tamanho do conjunto de dados. Gratuito e de código aberto, PostgreSQL é altamente extensível, permitindo a definição de tipos de dados personalizados, funções customizadas e a execução de código em várias linguagens de programação sem necessidade de recompilação do banco de dados organizações (POSTGRESQL, 2024)

2.12 Docker

Docker é uma plataforma de software que permite criação, teste e implantação de aplicação de containerizadas de maneira rápida e confiável. Um contêiner é uma unidade leve, autônoma e executável que inclui tudo o que é necessário para rodar uma aplicação, como bibliotecas, ferramentas de sistema, código e tempo de execução (ORACLE, 2024)

O Docker simplifica o desenvolvimento de software, empacotando todo o código e dependências de uma aplicação em um formato padrão garantido a execução a execução consiste em diferentes ambientes computacionais (ORACLE, 2024).

Docker Engine, software de código aberto que constrói e executa os contêineres, atuando como uma aplicação cliente servidor compatível com diversos sistemas operacionais, como Oracle Linux, CentOS, Debian, Fedora dentre outros (ORACLE, 2024).

As Imagens Docker contêm uma coleção de software que instruções necessárias para criar um contêiner que pode ser executado na plataforma Docker. São imutáveis, ou seja, qualquer alteração requer a construção de uma nova imagem (ORACLE, 2024).

Docker beneficia desenvolvedores e equipes de DevOps, permitindo a construção, empacotamento, envio e execução de aplicações como contêineres leves e portáteis. Desenvolvedores podem focar no código da aplicação sem se preocupar com o sistema operacional subjacente ou o sistema de implantação. Além disso, as equipes de DevOps podem integrar Docker em pipelines de integração e desenvolvimento contí-

⁴ As prioridades Atomicidade, Consistência, Isolamento e Durabilidade (ACID) são fundamentais para garantir a validade e confiabilidade das operações de transações em sistemas de Banco de Dados (Appendix M. Glossary, 2024).

⁵ PostGIS amplia os recursos do Banco de Dados relacional PostgreSQL sendo adicionado suporte para armazenamento, indexação e consulta de dados geoespaciais (PostGIS, 2024).

nuos, reduzindo a complexidade e as restrições na implantação e gestão das aplicações (ORACLE, 2024).

2.13 Express Node.js

Express foi lançado em novembro de 2010 e está atualmente na versão 4.16 da API. A popularidade do Node.js e do Express é evidenciada pelo número de empresas que os utilizam, pela quantidade de contribuições ao código e pelo suporte disponível.

Express é a framework mais popular para o Node.js e serve como base para outras frameworks, oferecendo soluções para:

1. Gerenciamento de requisições HTTP: Suporte para diferentes verbos HTTP (GET, POST, DELET) em diferentes URLs.
2. Protocolo de View Engines: Inserção de dados em templates para respostas dinâmicas.
3. Configuração de Aplicação: Definição de configurações comuns, como a porta de conexão e a localização dos modelos.
4. Middleware: Adição de processos de requisição em qualquer ponto da fila de requisições.

(MDN, 2024). Express é minimalista, permitindo que os desenvolvedores criem pacotes de middleware específicos para resolver problemas específicos no desenvolvimento de aplicações. Há bibliotecas para manipulação de cookies, sessões, login de usuários, parâmetros de URL, dados em requisições POST, cabeçalhos de segurança, entre outros:

1. Framework Não Opinativo: Express permite a inserção de qualquer middleware em qualquer ordem, estruturação do aplicativo em um ou múltiplos arquivos, e o uso de qualquer estrutura de pastas.
2. Código do Express: Um aplicativo Express aguarda pedidos HTTP, determina ações necessárias com base no padrão de URL e informações associadas, e pode interagir com bancos de dados ou executar outras tarefas. A resposta é retornada ao cliente, geralmente criando dinamicamente uma página HTML com os dados recuperados.

(MDN, 2024).

2.13.1 Node.js

O Node.js foi lançado em 2009, inicialmente apenas para Linux. O gerenciador de pacotes NPM surgiu em 2010, e o suporte nativo ao Windows em 2012. A versão atual de *Long Term Support (LTS)* é o Node v8.9.3, enquanto a versão mais recente é o Node 9 (MDN, 2024).

Node.js é um ambiente de tempo de execução *open-source* e multiplataforma que permite aos desenvolvedores criarem aplicações e ferramentas do lado servidor em JavaScript, fora do contexto de um navegador. O ambiente omite APIs JavaScript específicas do navegador e adiciona suporte a APIs tradicionais de sistema operacional, incluindo bibliotecas para sistemas HTTP e de arquivos.

1. Performance Excelente: Otimizado para alta taxa de transferência e escalabilidade em aplicações Web, ideal para aplicações em tempo real.
2. Código em JavaScript: Permite o uso de JavaScript tanto no servidor quanto no cliente, facilitando a transição entre eles.
3. Linguagem Moderna: Suporta Linguagens que Compilam para JavaScript, como TypeScript e CoffeeScript.
4. Gerenciador de Pacotes NPM: Acesso a uma vasta coleção de pacotes reutilizáveis, além de automação de ferramenta de compilação.

(MDN, 2024).

No desenvolvimento Web algumas tarefas comuns são suportadas diretamente pelo Node.js. Para gerenciar diferentes verbos HTTP, rotas, arquivos estáticos e templates de respostas, a utilização de uma framework muitas vezes torna o processo mais rápido e produtivo (MDN, 2024).

2.14 Linguagem de Programação C

A linguagem de programação C, criada por Dennis Ritchie nos Bell Laboratories em 1972, destaca-se como uma ferramenta de uso geral com notável popularidade, apesar de sua longa existência. Sua atratividade perdura devido à sua posição fundamental no âmbito da ciência da computação (W3C, 2023).

C encontra-se estreitamente vinculado ao sistema operacional UNIX⁶, sendo concebido originalmente para sua implementação. Esta linguagem figura entre as mais amplamente adotadas globalmente. A familiaridade com C facilita a aprendizagem de outras linguagens de programação proeminentes, tais como Java, Python, C++, C#, entre outras, devido à similaridade em sua sintaxe (W3C, 2023).

⁶ O UNIX é um sistema operacional desenvolvido nos anos 60 no laboratório Bell da AT&T, conhecido pela sua flexibilidade e influência em sistemas operacionais subsequentes.

A eficiência de C é notável, especialmente em comparação com linguagens como Java e Python, proporcionando-lhe uma velocidade de execução superior. Sua versatilidade permite sua aplicação em diversas áreas e tecnologias (W3C, 2023).

Apesar de a maioria dos compiladores atualmente suportarem a linguagem C++, no passado isso não era uma realidade. Devido à escassez de recursos, os desenvolvedores adotavam estratégias de software que não utilizavam os benefícios da orientação a objetos (BERTOLETI, 2017).

Em linguagens de alto nível como C++, um objeto pode ser definido como um espaço de memória alocado para dados de diferentes naturezas. Este espaço é referenciado por uma variável, permitindo que o objeto seja chamado pelo seu nome. Objetos podem conter dados de diferentes tipos e tamanhos, facilitando a manipulação de múltiplos dados em um bloco específico de memória, além de permitir alterações e clonagens (BERTOLETI, 2017).

Nas linguagens de alto nível, um objeto é originado a partir de uma classe, que descreve tudo o que o objeto deve conter. Em C, uma classe pode ser descrita utilizando a seguinte abordagem:

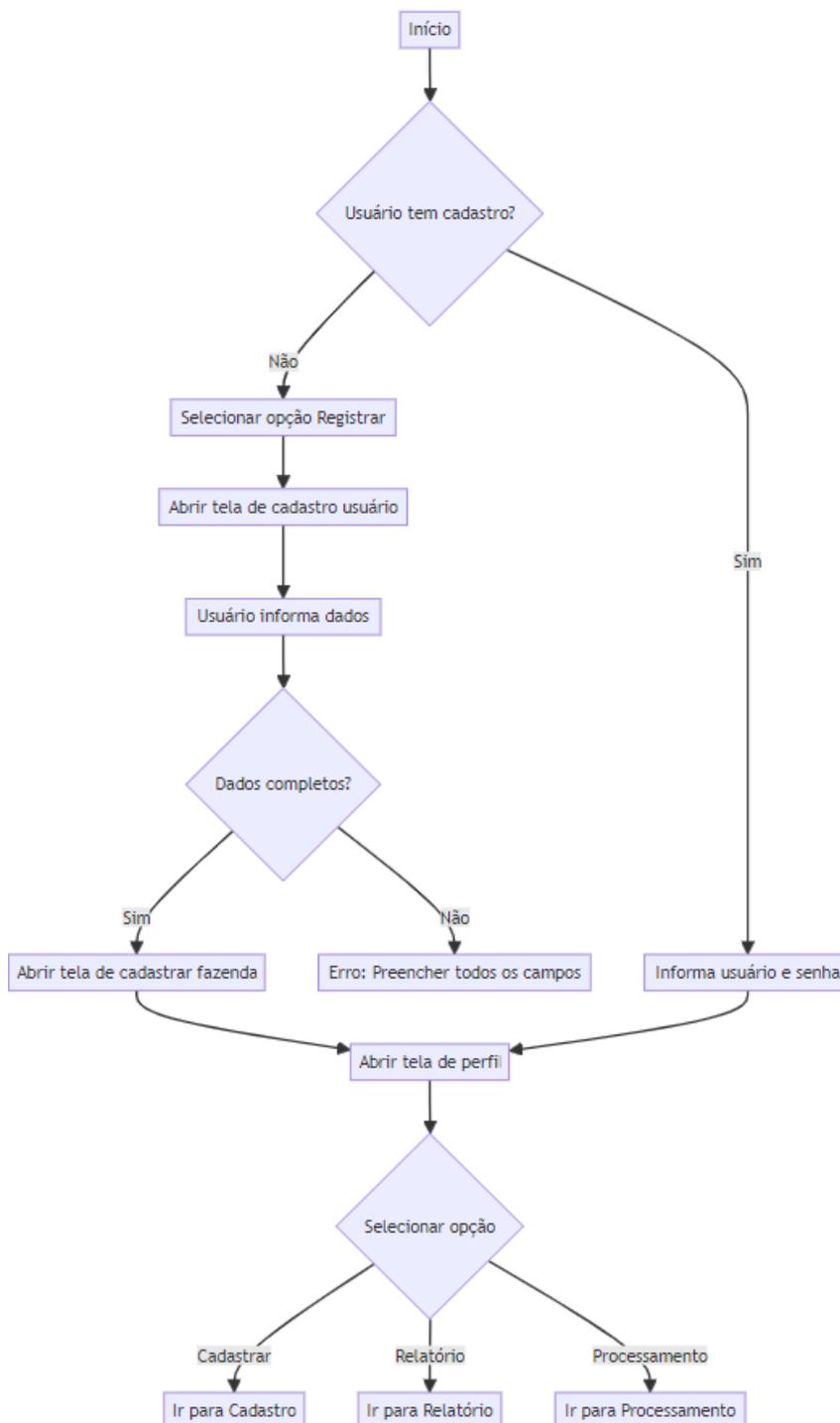
1. Declaração de Objetos: O objeto deve ser declarado como uma variável comum. Para isso, a classe pode ser definida como um 'typedef' de uma 'struct'.
2. Parâmetros: Dentro dessa 'struct', podem ser criadas variáveis de quaisquer tipos, conforme a necessidade, constituindo os parâmetros da classe.
3. Métodos: Objetos podem conter métodos, implementados como ponteiros de função. Esses ponteiros armazenam os endereços das funções que serão executadas.

(BERTOLETI, 2017).

3 PROPOSTA DE SOLUÇÃO

Neste capítulo, são apresentadas as tecnologias e ferramentas empregadas no projeto com base no referencial teórico do trabalho. O sistema foi desenvolvido com o intuito de facilitar o fluxo do gerenciamento da fazenda. As Figuras 15 a 18 mostram todo o processo que pode ser realizado no sistema.

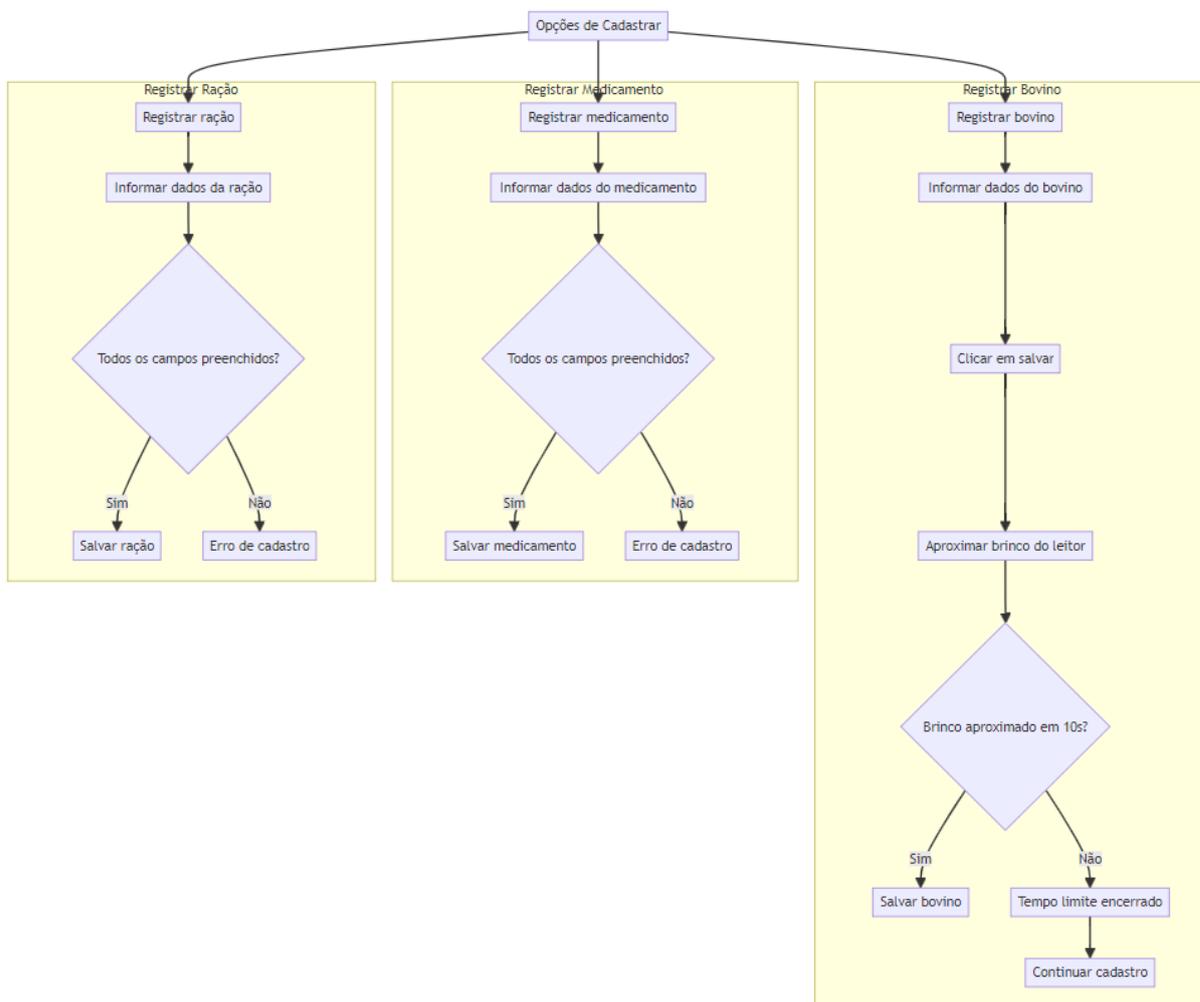
Figura 15 – Fluxograma Principal



Fonte: (Elaborado pelo autor, 2024).

No fluxograma principal, Figura 15, o usuário informará seu *login* de usuário o sistema será redirecionado para o perfil do usuário onde poderá selecionar a opção que desejar sendo elas: Cadastro, Relatórios e Processamento. Caso o usuário não estiver registrado, terá que selecionar a opção de registrar, mostrado nas Figuras 21 e 23, onde poderá colocar seus dados pessoais e dados da fazenda lembrado que todos os campos são obrigatórios, feito isso será redirecionado para o perfil do usuário.

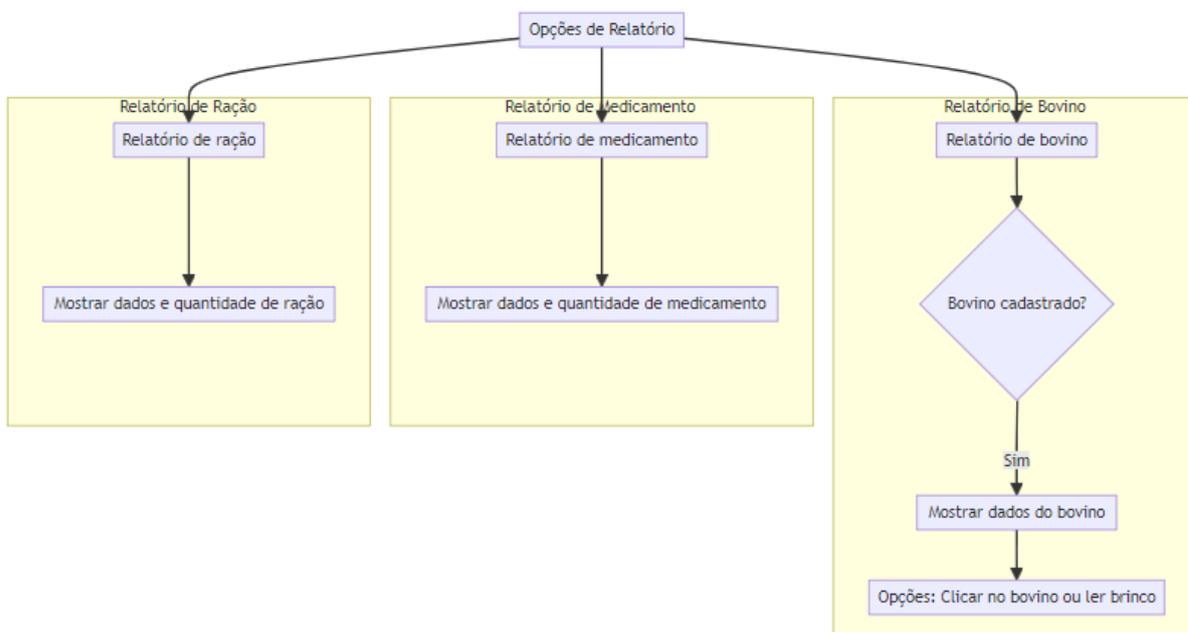
Figura 16 – Fluxograma das Opções de Cadastro.



Fonte: (Elaborado pelo autor, 2024).

No fluxograma da Figura 16 das opções de cadastro são realizados os procedimentos de inserção das informações no banco de dados, são realizados para encher o sistema com informações, nos cadastros de rações e medicamento o usuário coloca os dados iniciais para fazer o controle de estoque, como é mostrado nas Figuras e. No registro do bovino usuário deverá informar os dados iniciais do bovino ao selecionar a opção de salvar será informado ao usuário na tela para aproximar o brinco no leitor desta forma o bovino está identificado com o brinco, permitindo puxar as informações do bovino fazendo a leitura do brinco com o leitor.

Figura 17 – Fluxograma de Relatório



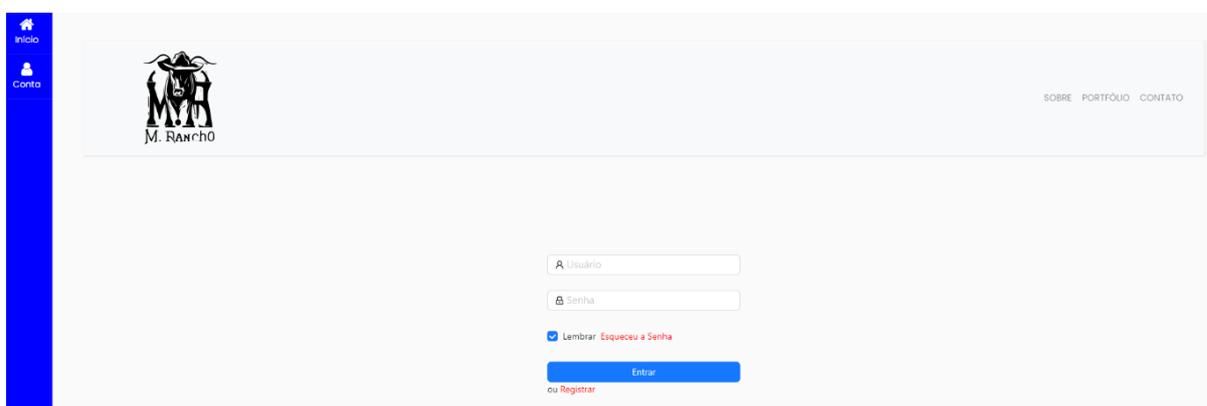
Fonte: (Elaborado pelo autor, 2024).

No fluxograma da Figura 17, o usuário verifica todas as informações cadastradas no relatório de ração e medicamento puxando os dados inseridos no cadastro de ambos. Um dos tópicos principais dos relatórios é informar a quantidade de itens cadastrados para fazer o controle de estoque.

No relatório bovino puxam-se todas as informações dos procedimentos que foram realizados no bovino mostrando o histórico de vacina, relatório do zootecnista e toda a dieta do bovino.

Com o sistema de gerenciamento de bovinos, uma vez cadastrado é possível proceder com gerenciamento da fazenda pelo sistema, conforme exemplificado na figura 18.

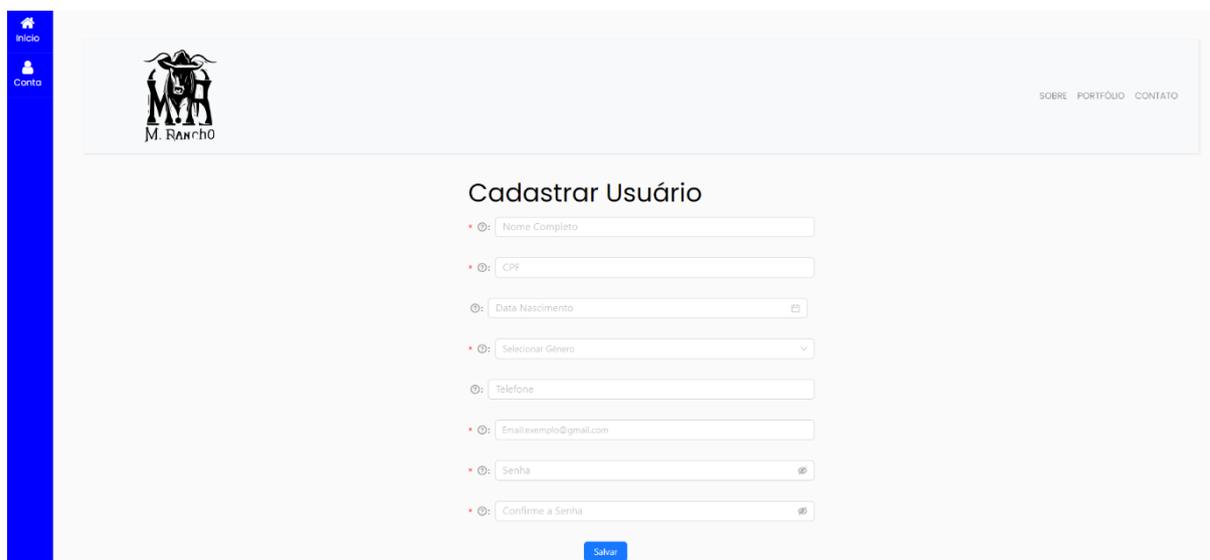
Figura 18 – Tela de Login



Fonte: (Elaborado pelo autor, 2024).

O sistema foi desenvolvido para um único usuário ao criar a conta, conforme mostrado nas Figuras 19 a 22. Após o registro nas seções de cadastro de usuário e cadastro da fazenda, será possível começar a gerenciar a fazenda de maneira eficiente.

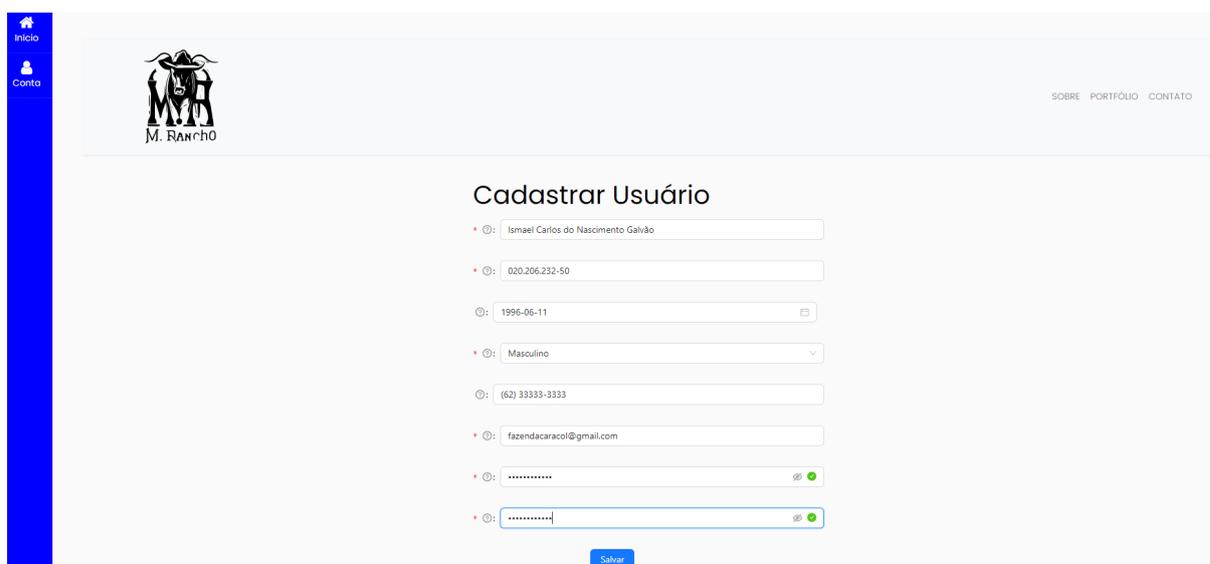
Figura 19 – Cadastrar Usuário



A imagem mostra a interface de usuário para o cadastro de um novo usuário. O cabeçalho contém o logo 'M. RANCHO' e links para 'SOBRE', 'PORTFÓLIO' e 'CONTATO'. Um menu lateral azul à esquerda possui ícones para 'Início' e 'Conta'. O formulário 'Cadastrar Usuário' contém os seguintes campos obrigatórios: Nome Completo, CPF, Data Nascimento (com ícone de calendário), Selecionar Género (menu suspenso), Telefone, Email exemplo@gmail.com, Senha (com ícone de olho) e Confirme a Senha (com ícone de olho). Um botão 'Salvar' está localizado na base do formulário.

Fonte: (Elaborado pelo autor, 2024).

Figura 20 – Inserindo Usuário

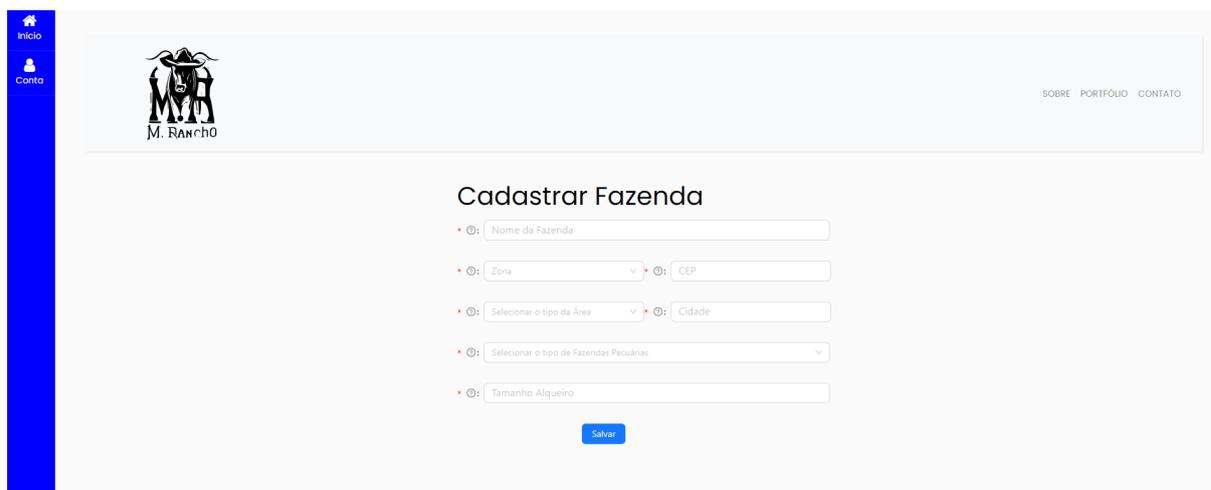


A imagem mostra a mesma interface de usuário, mas com os campos do formulário preenchidos. Os dados inseridos são: Nome Completo: 'Ismael Carlos do Nascimento Galvão', CPF: '020.206.232-50', Data Nascimento: '1996-06-11', Género: 'Masculino', Telefone: '(62) 33333-3333', Email: 'fazendacaracol@gmail.com', Senha: '.....' e Confirme a Senha: '.....'. O botão 'Salvar' permanece visível na base.

Fonte: (Elaborada pelo autor, 2024).

O usuário deverá informar todos os seus dados, pois todos os campos são obrigatórios. O registro só será salvo se não houver nenhum campo em branco ou incompleto.

Figura 21 – Cadastro da Fazenda



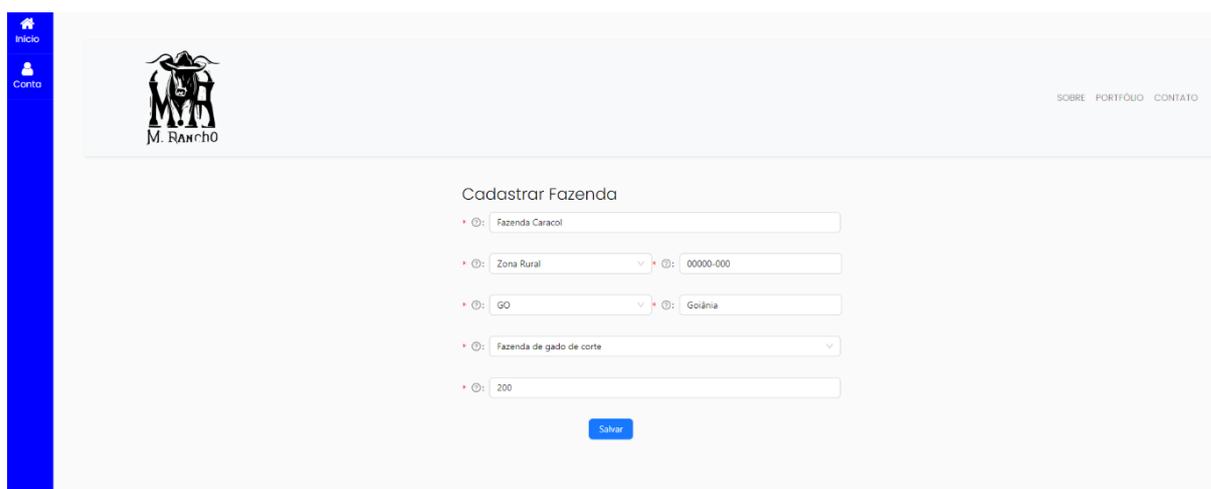
The screenshot shows the 'Cadastrar Fazenda' (Register Farm) form in the M. Rancho system. The form is located on a page with a blue sidebar on the left containing 'Inicio' and 'Conta' links. The top right of the page has links for 'SOBRE', 'PORTFÓLIO', and 'CONTATO'. The M. Rancho logo is in the top left. The form fields are:

- Nome da Fazenda (text input)
- Zona (dropdown menu) and CEP (text input)
- Selecionar o tipo da Área (dropdown menu) and Cidade (text input)
- Selecionar o tipo de Fazendas Pecuárias (dropdown menu)
- Tamanho Alqueiro (text input)

A 'Salvar' button is located below the form fields.

Fonte: (Elaborado pelo autor, 2024).

Figura 22 – Inserindo Fazenda



The screenshot shows the 'Cadastrar Fazenda' form with data entered. The form is located on a page with a blue sidebar on the left containing 'Inicio' and 'Conta' links. The top right of the page has links for 'SOBRE', 'PORTFÓLIO', and 'CONTATO'. The M. Rancho logo is in the top left. The form fields are filled with the following data:

- Nome da Fazenda: Fazenda Caracol
- Zona: Zona Rural and CEP: 00000-000
- Selecionar o tipo da Área: GO and Cidade: Goiânia
- Selecionar o tipo de Fazendas Pecuárias: Fazenda de gado de corte
- Tamanho Alqueiro: 200

A 'Salvar' button is located below the form fields.

Fonte: (Elaborado pelo autor, 2024).

Ao finalizar o cadastro do usuário mostrando na Figura 21, será direcionado para a tela de registro da fazenda, todos os campos devem ser incluídos. Após salvar será direcionado para a tela de perfil do usuário onde pode-se selecionar as opções de acordo com suas necessidades.

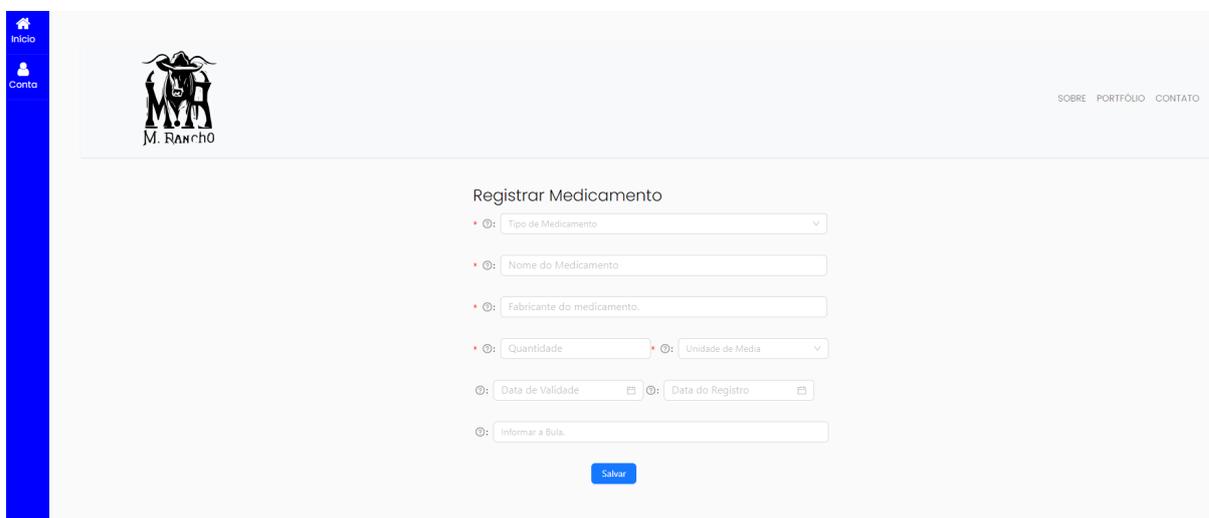
Figura 23 – Tela de Perfil



Fonte: (Elaborado pelo autor, 2024).

Na opção de cadastrar o usuário poderá cadastrar os Bovinos juntamente com o cadastro do Brinco desta forma poderá ser buscado as informações do bovino no sistema fazendo a leitura do Brinco, como mostra nas seguintes Figuras

Figura 24 – Registrar Medicamento



Fonte: (Elaborado pelo autor, 2024).

Para registrar o medicamento, todos os dados devem ser informados, conforme mostrado na Figura 25, pois são campos obrigatórios. Dessa forma, quanto mais informações forem fornecidas, mais detalhados serão os relatórios.

Figura 25 – Registrando Medicamento

Registrar Medicamento

- O: Anti-inflamatórios
- O: Diclofenaco JA
- O: JA Saúde Animal
- O: 100
- O: mL
- O: 2023-06-02
- O: 2025-06-01
- O: Diclofenaco 100ml
Indicação:
DICLOFENACO J.A. é um anti-inflamatório, não narcótico e não esteróide com potente atividade analgésica e antipirética. Indicado para todos os casos de inflamação, dor e febre. Também é indicado no tratamento auxiliar nas infecções agudas, reduzindo o processo inflamatório, a dor e a febre.

< Voltar Salvar

Fonte: (Elaborado pelo autor, 2024).

Assim que forem informados todos os detalhes do medicamento, fazendo o salvamento, o usuário consegue puxar as informações de estoque e os detalhes sobre o medicamento conforme a Figura 26 onde mostra parcialmente as informações do medicamento e na Figurinha 27 onde mostra todos os dados mais detalhados.

Figura 26 – Relatório Parcial dos Medicamentos

Relatórios

Informações iniciais

Buscar

Tipo	Nome	Dt. Entrada	Dt. Registro	Estoque
Antibacteriano	Diclofenaco 100ml	02/06/2023	01/06/2024	20

< Voltar

Fonte: (Elaborada pelo autor, 2024).

Figura 27 – Iatório Detalhado do Medicamento

Logo: M. RANCHO

SOBRE PORTFÓLIO CONTATO

Informações do Antibacteriano de numeração 1 Diclofenaco 100ml
 Fabricante JA Saúde Animal

Dados	DATA DE VALIDADE	DATA DE REGISTRO	QUANTIDADE	TIPO
	02/06/2023	01/06/2024	20	Antibacteriano

Bula do Diclofenaco 100ml
DETALHES
 Indicação: DICLOFENACO J.A. é um anti-inflamatório, não narcótico e não esteróide com potente atividade analgésica e antipirética, indicado para todos os casos de inflamação, dor e febre. Também é indicado no tratamento auxiliar nas infecções agudas, reduzindo o processo inflamatório, a dor e a febre, estimulando o animal a se alimentar. Indicado no tratamento das cólicas, processos dolorosos causados por cirurgia, ferida, distocias, claudicação por pododermatite ou desordens músculoesqueléticas, síndrome Mastite-Metrite-Agalaxia (MMA) e diarreia dos recém-nascidos. Descrição: DICLOFENACO JA é Anti-inflamatório, analgésico e antipirético não esteroidal injetável. Modo de uso: Deve ser administrado na dosagem de 1 mg/kg, ou seja, 1 ml para cada 50 kg de peso corporal. Aplicação: Injeção intravenosa, intramuscular ou subcutânea. Para uso em: Bovinos, caprinos, ovinos e suínos. FÓRMULA: Cada 100 ml contém: Diclofenaco sódico.....5,00 g Veículo q.s.p.....100,00 ml

Vacinhas
 < Voltar < Anterior > Próximo

Fonte: (Elaborado pelo autor, 2024).

Para fazer o estoque de toda a parte de dieta do bovino o usuário deve escolher a opção definir dieta:

Figura 28 – Registrar Ração

Logo: M. RANCHO

SOBRE PORTFÓLIO CONTATO

Registrar Ração

- Tipo de ração: [dropdown]
- Nome da ração: [input]
- Fabricante da ração: [input]
- Quantidade: [input] • Unidade de Medida: [dropdown]
- Data de Validade: [calendar icon] • Data do Registro: [calendar icon]
- Informar Detalhes: [text area]

< Voltar Salvar

Fonte: (Elaborado pelo autor, 2024).

Para registrar a ração informe todos os dados, como mostrado na Figura 29, pois são campos obrigatório, desta forma quanto mais informações aplicadas mais detalhadas são os relatórios.

Figura 29 – Registrando Ração

Registrar Ração

• Tipo: Sal

• Nome: Ração para Bovinos de Corte Fort Semi 18

• Quantidade: 30 kg

• Duração: 2024-06-28 a 2024-04-27

• Indicação de Uso:
Indicada para alimentação de bovinos de corte em fase de terminação, criados em sistema de semiconfinamento. Modo de Usar: Fornecer diariamente aos animais 1 kg do produto para cada 100 kg de peso vivo dos animais ao dia. Fornecer este produto em cochos com comprimento de 30 centímetros lineares por animal, com acesso aos dois (2) lados, e guidos a

< Voltar Salvar

Fonte: (Elaborado pelo autor, 2024).

Ao inserir as informações e salvá-las, o usuário consegue acessar as informações de estoque e os detalhes sobre a ração. Na Figura 30, são mostradas parcialmente as informações de rações:

Figura 30 – Relatório Parcial das Rações

Relatório Parcial das Rações

Relatórios

Informações iniciais

Buscar

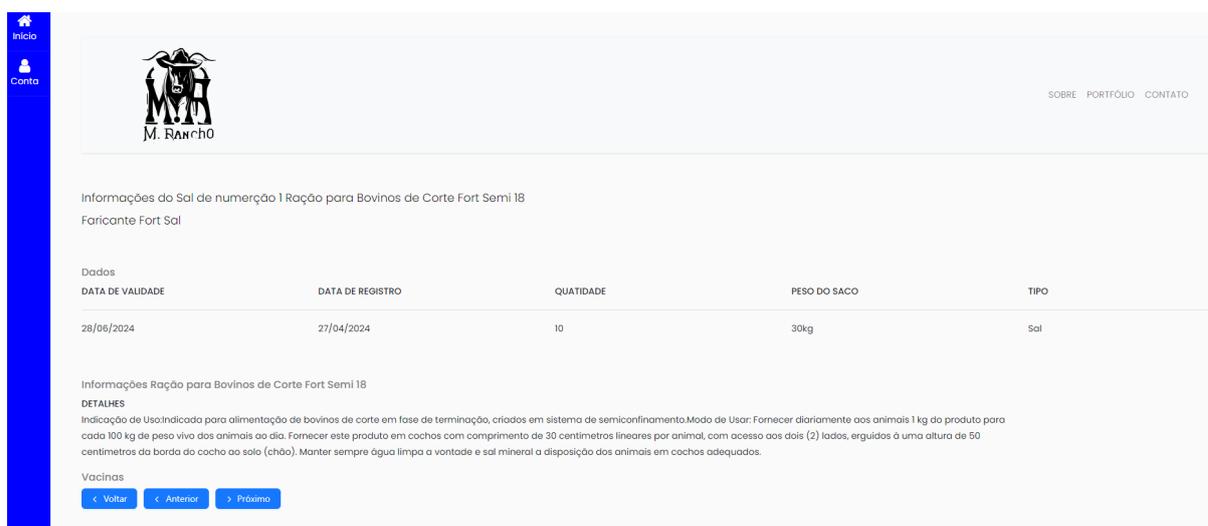
Tipo	Nome	Dt. Entrada	Dt. Registro	Estoque
Sal	Ração para Bovinos de Corte Fort Semi 18	28/06/2024	27/04/2024	12

< Voltar

Fonte: (Elaborado pelo autor, 2024).

Na Figura 31 todos os dados sobre medicamento são exibidos de forma mais detalhada:

Figura 31 – Relatório detalhado da Ração



INICIO
Conta

M. RANCHO

SOBRE PORTFOLIO CONTATO

Informações do Sal de numeração 1 Ração para Bovinos de Corte Fort Semi 18
Fabricante Fort Sal

DATA DE VALIDADE	DATA DE REGISTRO	QUANTIDADE	PESO DO SACO	TIPO
28/08/2024	27/04/2024	10	30kg	Sal

Informações Ração para Bovinos de Corte Fort Semi 18
DETALHES
Indicação de Uso: Indicação para alimentação de bovinos de corte em fase de terminação, criados em sistema de semiconfinamento. Modo de Usar: Fornecer diariamente aos animais 1 kg do produto para cada 100 kg de peso vivo dos animais ao dia. Fornecer este produto em cochos com comprimento de 30 centímetros lineares por animal, com acesso aos dois (2) lados, erguidos a uma altura de 50 centímetros da borda do cocho ao solo (chão). Manter sempre água limpa a vontade e sal mineral a disposição dos animais em cochos adequados.

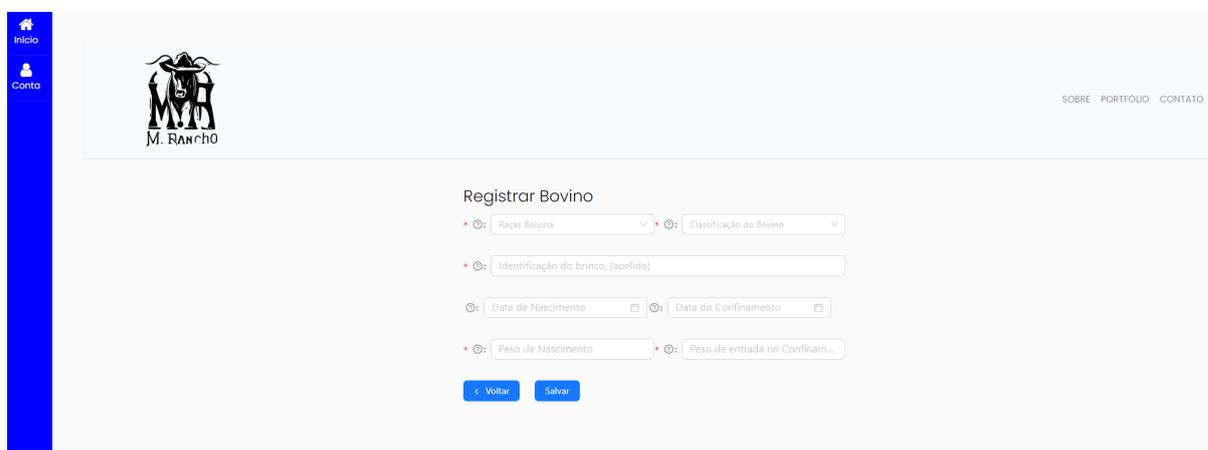
Vacinas

[← Voltar](#) [← Anterior](#) [→ Próximo](#)

Fonte: (Elaborado pelo autor, 2024).

Após inserir os medicamentos e as rações, conforme mostrado nos exemplos nas Figuras 29 a 31, é possível registrar um bovino para então fazer as inserções de vacinas e definir a dieta.

Figura 32 – Registrar Bovino



INICIO
Conta

M. RANCHO

SOBRE PORTFOLIO CONTATO

Registrar Bovino

• Raça Bovina • Classificação do Bovino

• Identificação do brinco, (apelido)

• Data de Nascimento • Data do Confinamento

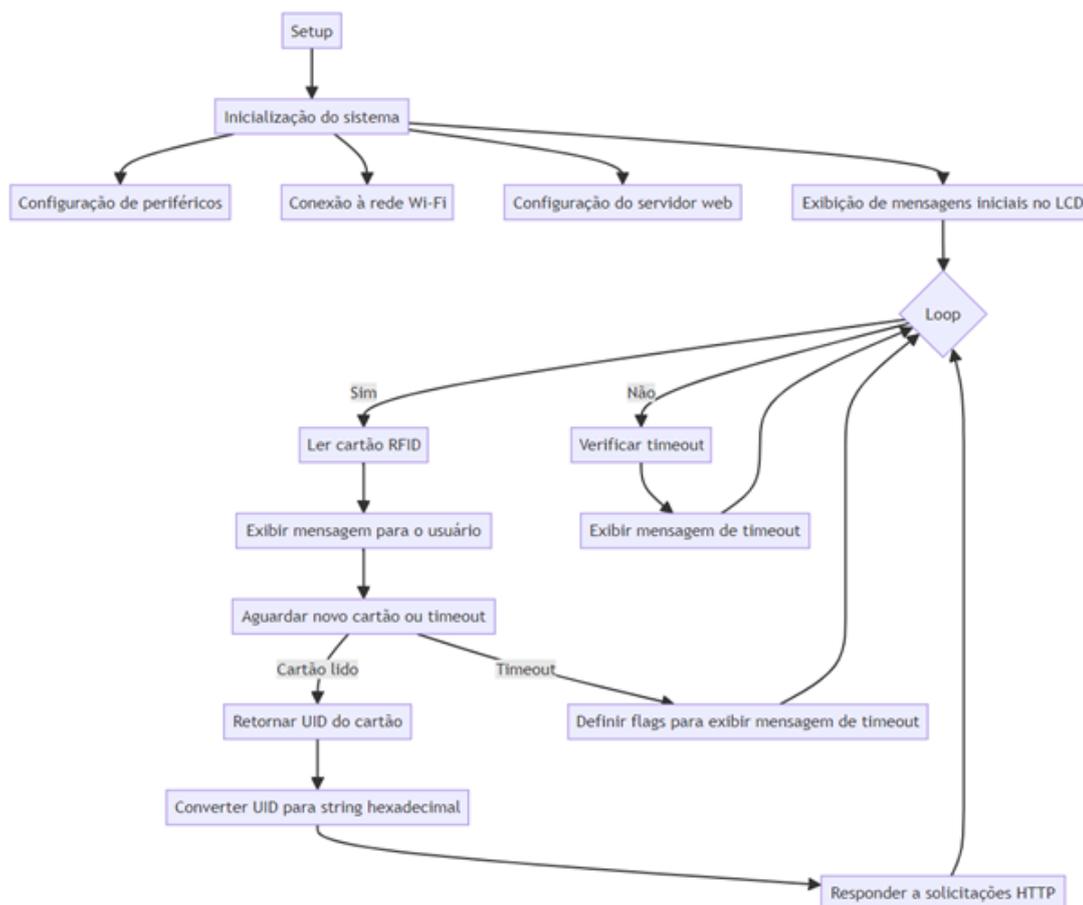
• Peso de Nascimento • Peso de entrada no Confinam...

[← Voltar](#) [Salvar](#)

Fonte: (Elaborado pelo autor, 2024).

Para cadastrar o bovino, o usuário deverá ligar o leitor do brinco. Primeiramente, o leitor fica no modo de espera, durante o qual ele se conecta ao WiFi, conforme demonstrado na Figura 33, que mostra todo o fluxograma da execução do leitor RFID.RC522.

Figura 33 – Fluxograma do Funcionamento do Leitor



Fonte: (Elaborada pelo autor, 2024).

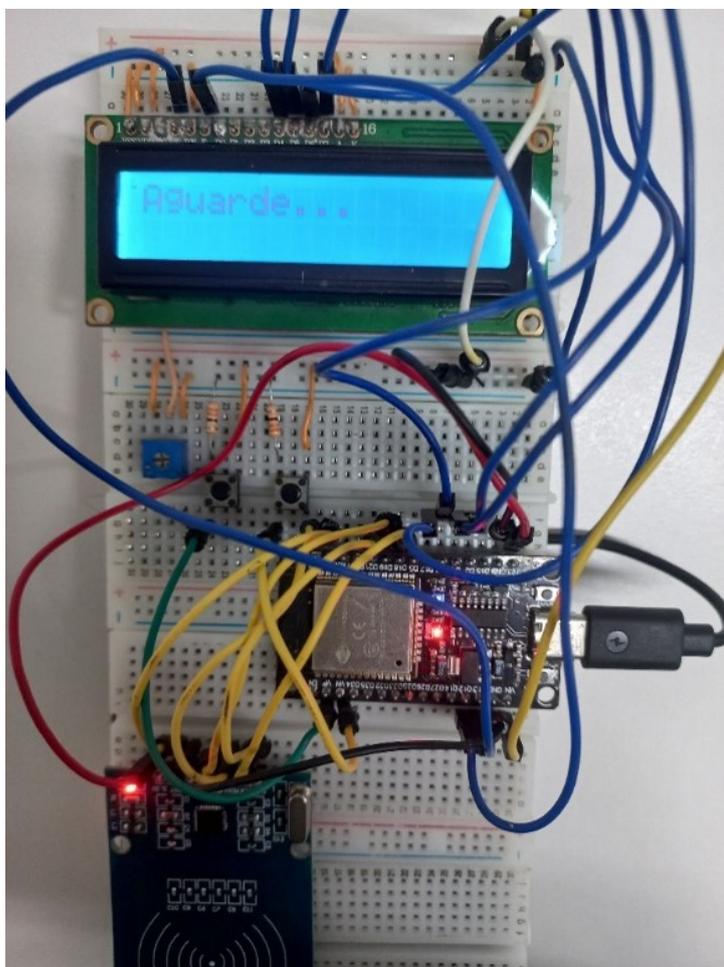
O fluxograma começa com a fase de *Setup*, onde ocorre a inicialização do sistema. Esta fase envolve várias etapas, incluindo a Inicialização do sistema, a Configuração de periféricos (como o leitor RFID e o LCD), a Conexão à rede WiFi, e a Configuração do servidor web para lidar com solicitações HTTP. Uma vez configurado, o sistema exibe mensagens iniciais no LCD para informar o usuário sobre o estado do sistema.

Após a configuração, o programa entra em um *Loop* principal. Dentro deste *loop*, o sistema verifica continuamente se há novos cartões RFID para ler ou se ocorreu um *timeout*. Se um novo cartão RFID estiver presente, o sistema entra na fase de ler cartão RFID, onde uma mensagem é exibida no LCD solicitando ao usuário que aproxime seu cartão do leitor. O sistema então aguarda até que um novo cartão seja apresentado ou até que ocorra um *timeout*.

Se um cartão for lido com sucesso, o UID do cartão é retornado e convertido para uma representação de *string* hexadecimal na função converter UID para *string* hexadecimal. Esse UID convertido é então utilizado para responder a solicitações HTTP na etapa responder a solicitações HTTP. Caso ocorra um *timeout*, as *flags* são definidas para exibir uma mensagem de *timeout* na próxima iteração do *loop*. Nesta situação, uma mensagem informando sobre o tempo limite atingido é exibida no LCD.

Este ciclo de leitura e verificação continua garantindo que o sistema esteja sempre pronto para ler novos cartões RFID e responder às solicitações do servidor Web de maneira adequada.

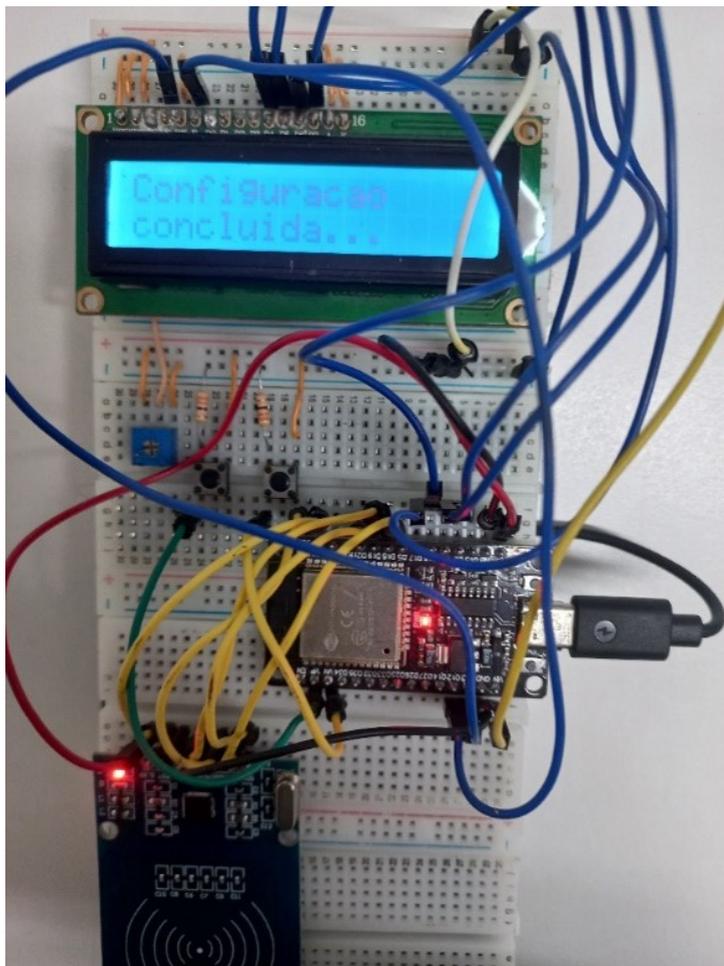
Figura 34 – Leitor do Brinco Aguardando Conexão com o WiFi



Fonte: (Elaborada pelo autor, 2024).

Após conexão, a tela mostra Configuração concluída, como na Figura 35.

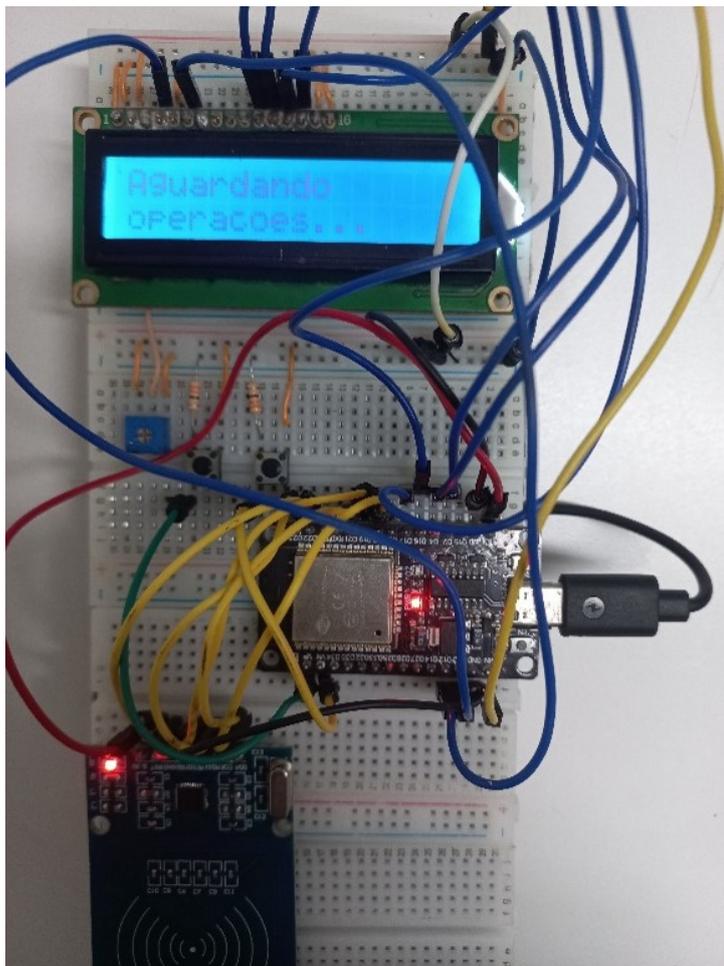
Figura 35 – Leitor do Brinco Configuração Concluída



Fonte: (Elaborada pelo autor, 2024).

Desta forma, o leitor fica no modo Aguardando Operações, conforme demonstrado na Figura 36. Nesse modo, o leitor aguarda o sistema comunicar-se com ele tanto para gravar o brinco quanto para realizar a leitura.

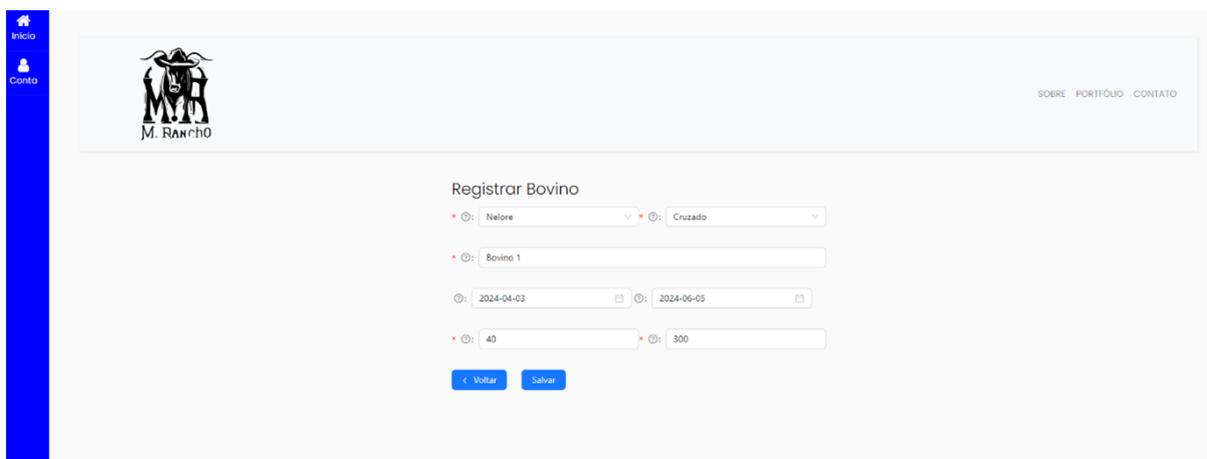
Figura 36 – Leitor de Brinco Aguardando Operações



Fonte: (Elaborada pelo autor, 2024)

Voltando ao sistema pode-se dar continuidade ao processo assim que colocar os dados de cadastro do bovino ao clicar em salvar será informado para o usuário pegar o brinco com o Chi Mifare e aproximar ao leitor o sistema para dar continuidade, como demonstrado na Figura 37 o usuário terá que preencher todas as informações do registro de bovino não podendo deixar nenhum campo em branco pois são campos obrigatórios.

Figura 37 – Registrando o Bovino



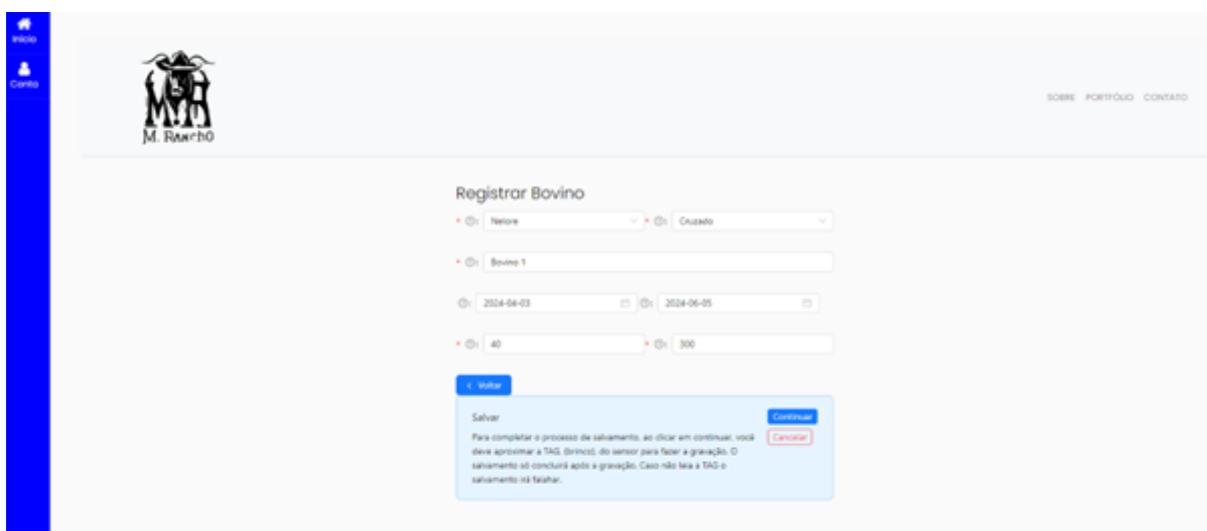
The screenshot shows a web interface for 'M. Rancho' with a blue sidebar containing 'Início' and 'Conta'. The main content area is titled 'Registrar Bovino' and contains the following fields:

- Two dropdown menus: 'Nelore' and 'Cruzado'.
- A text input field labeled 'Bovino 1'.
- Two date pickers: '2024-04-03' and '2024-06-05'.
- Two numeric input fields: '40' and '300'.
- Two buttons at the bottom: 'Voltar' and 'Salvar'.

Fonte: (Elaborado pelo autor, 2024).

Após inserir todos os dados iniciais do bovino, selecionar a opção salvar o sistema mostrará uma mensagem dando uma breve explicação de como prosseguir com registro do bovino, como mostra a Figura 38.

Figura 38 – Registrando Brinco no Bovino Cadastrado



This screenshot shows the same 'Registrar Bovino' form as in Figure 37, but with a modal message box displayed. The message box contains the following text:

Salvar

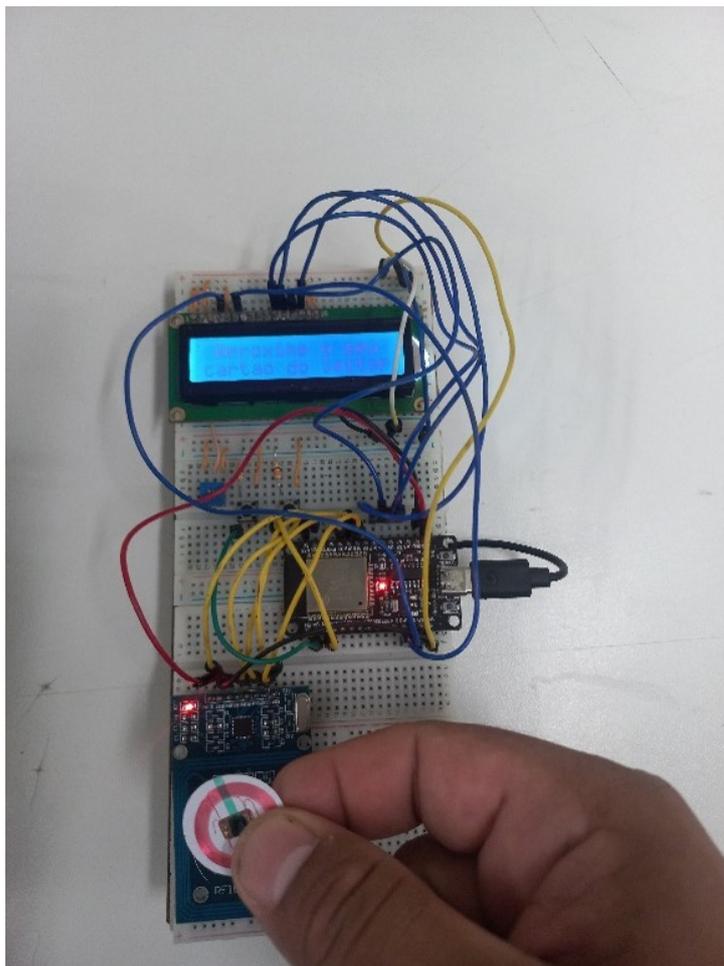
Para completar o processo de salvamento, ao clicar em continuar, você deve aproximar o TAG (brinco) do sensor para fazer a gravação. O salvamento só concluirá após a gravação. Caso não leia o salvamento irá falhar.

Buttons: 'Continuar' (blue) and 'Cancelar' (red).

Fonte: (Elaborado pelo autor, 2024).

Ao selecionar para continua o usuário terá que aproximar o Chip Mifare do brinco no leitor como está exemplificado na Figura 39.

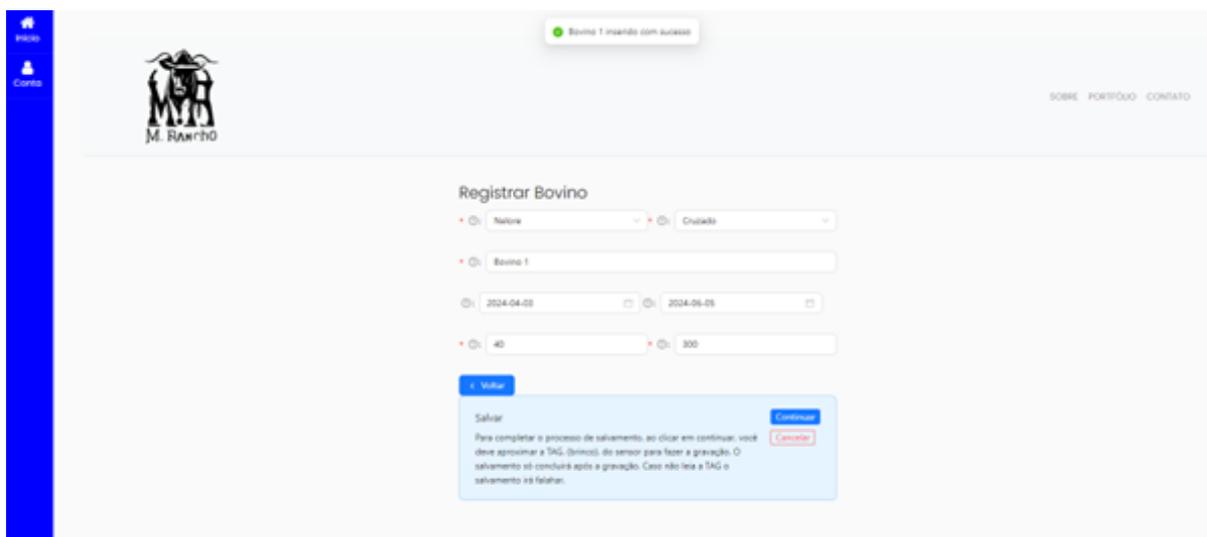
Figura 39 – Fazendo a Leitura do Chip Mifare do Brinco



Fonte: (Elaborado pelo autor, 2024)

Ao aproximar o Chip Mifare do brinco no leitor, o bovino será registrado com a UID deste Chip. Na tela do usuário, será exibida a seguinte mensagem: "Bovino 1 Inserido com sucesso", conforme mostrado na Figura 40.

Figura 40 – Bovino Inserido com Sucesso

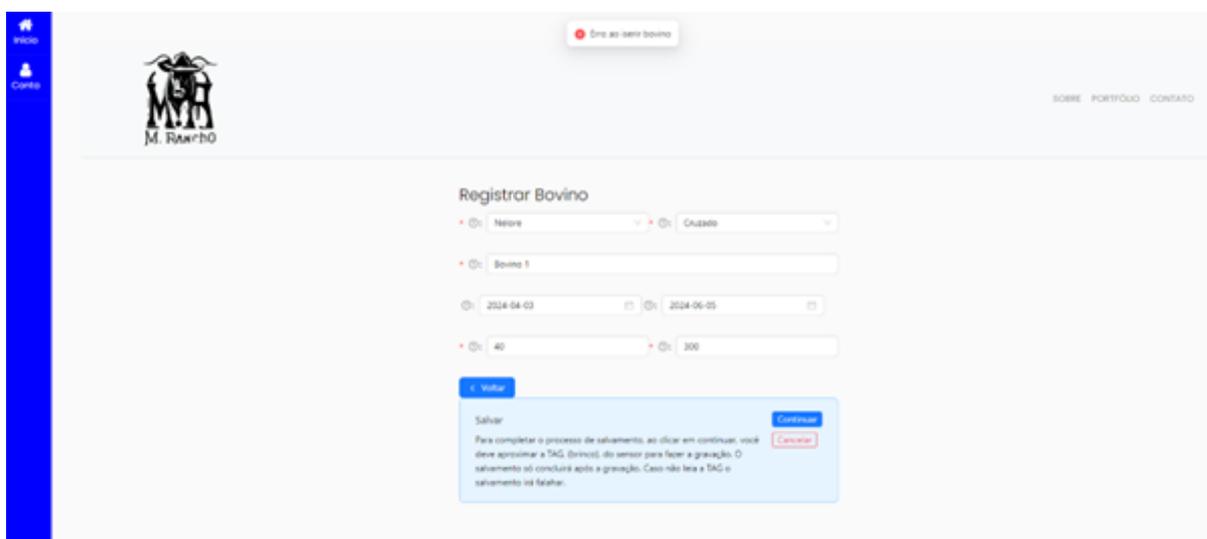


The screenshot shows a web interface for registering a bovine. At the top, a green notification bubble states "Bovino 1 inserido com sucesso". The main form, titled "Registrar Bovino", contains the following fields: "Raça" (set to "Nelora"), "Criança" (set to "Criança"), "Bovino 1" (text input), "Data de Nascimento" (2024-04-03), "Data de Registro" (2024-06-05), "Peso" (40), and "Altura" (300). A "Salvar" button is visible. Below the form, a light blue box contains instructions: "Para completar o processo de salvamento, ao clicar em continuar, você deve aproximar a TAG (brinco) do sensor para fazer a gravação. O salvamento só concluirá após a gravação. Caso não leia a TAG o salvamento irá falhar." It includes "Continuar" and "Cancelar" buttons.

Fonte: (Elaborado pelo autor, 2024).

Caso o usuário não aproxime o Chip Mifare do brinco, como mostrado na Figura 39, o sistema aguardará por 10 segundos a aproximação. Se esse tempo for completado sem sucesso, o sistema emitirá um erro de *Timeout*. O usuário será informado desse erro, conforme mostrado na Figura 41.

Figura 41 – Erro ao Inserir o Bovino



The screenshot shows the same "Registrar Bovino" form as in Figure 40, but with a red notification bubble at the top stating "Erro ao inserir bovino". The form fields and the instruction box at the bottom are identical to the previous figure.

Fonte: (Elaborado pelo autor, 2024).

Quando ocorrer este erro o usuário pode apertar em continuar novamente que o tempo será recontado para finalizar o registro do Bovino.

Após fazer o Registro do bovino, automaticamente seus dados estão nos relatórios mostrando os dados iniciais dos bovinos, como mostra as Figuras 42 e 43, pois até

o momento não foi registrado nenhuma informações de vacinação ou de dieta do bovino.

Figura 42 – Relatório Parcial do Bovino

Relatórios

Informações iniciais

Ler bovino

Identificador	Raça	Data de entrada	Peso Inicial	Peso Final
Bovino 1	Nelore	05/06/2024, 00:00:00	40kg	kg

Voltar

Fonte: (Elaborado pelo autor, 2024).

Figura 43 – Relatório detalhado do Bovino

Bovino

1

Relatórios do Bovino do brinco 99497CID

Dados

DATA DE NASCIMENTO	PESO DE NASCIMENTO KG	PESO ATUAL KG	RAÇA
03/04/2024, 00:00:00	40		Nelore

Alimentação e Suplementação

RAÇÃO	CAPM	SUPLENTO	SILO

Vacinas

NOME	TIPO	DOSAGEM/QUANTIDADE	DATA
*	*	*	*

Relatório prescrito pelo Zootecnista:

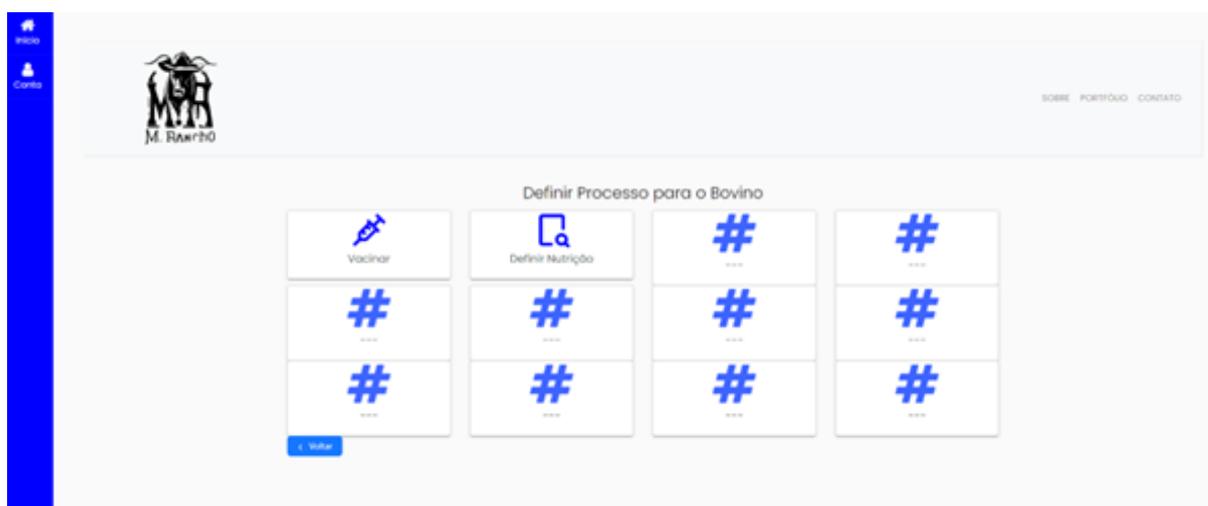
*

Voltar

Fonte: (Elaborado pelo autor, 2024).

Para inserir medicamento e a dieta do bovino na tela do perfil usuário, na Figura 23, deve-se escolher a opção de processamento, com isto quando o usuário for aplicar uma vacina no bovino ou definir a dieta do bovino basta escolher a opção desejado, como mostra na Figura 44.

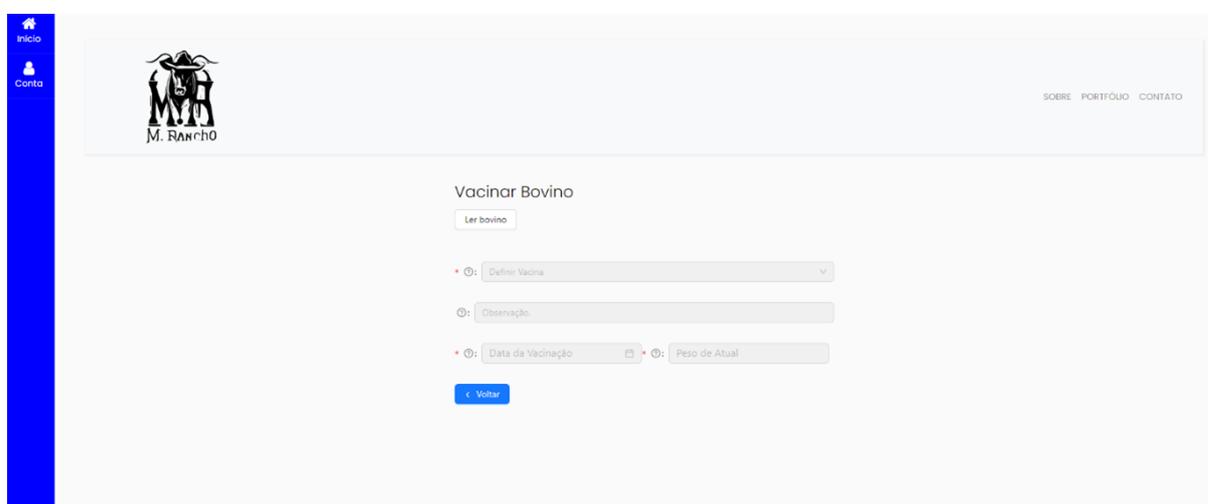
Figura 44 – Definir Processo Para o Bovino



Fonte: (Elaborado pelo autor, 2024).

Para aplicar a vacina no bovino o usuário precisa fazer a leitura do Chip Mifare para ter total precisão de não selecionar um bovino errado, mantendo total segurança nos dados dos relatórios. Para fazer aplicação deve-se selecionar a opção de vacina do bovino, mostrado na Figura 45.

Figura 45 – Vacinar Bovino

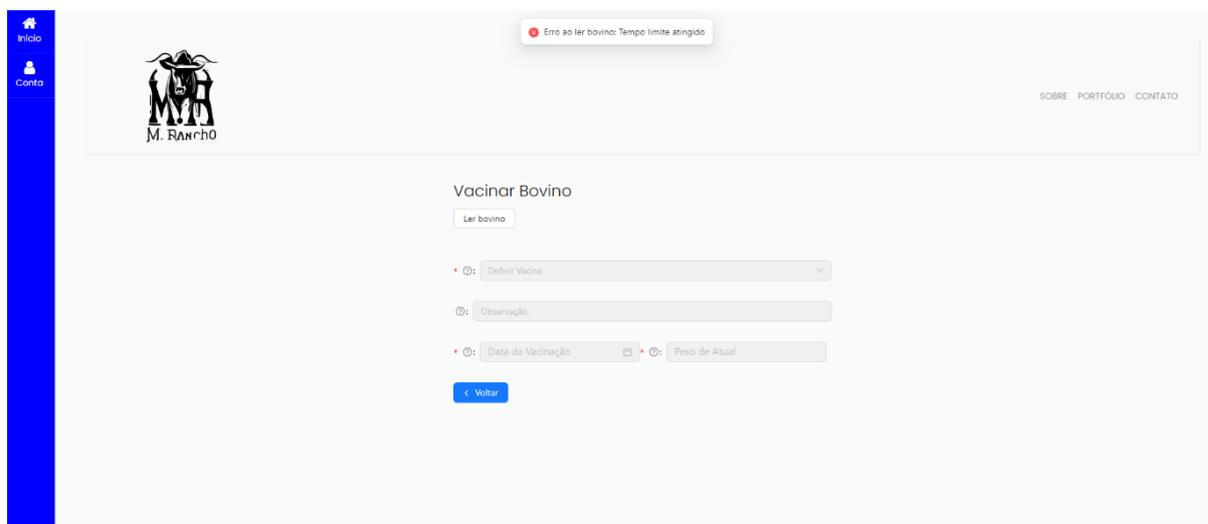


Fonte: (Elaborado pelo autor, 2024).

Os campos estão bloqueados para liberar os campos o usuário seleciona a opção de ler brinco, desta forma tem se o bovino selecionado para fazer a aplicação do medicamento. Ao selecionar a opção ler bovino o leitor fica no modo de espera até o Chip Mifare ser aproximado, como mostra na Figura 39.

Caso o Brinco não seja lido o usuário receberá notificação informando que o brinco não foi lido pois deu erro de *Timeout*, mostrado na Figura 46.

Figura 46 – Erro ao ler Bovino



Fonte: (Elaborado pelo autor, 2024).

Ao aproximar o cartão e o leitor reconhecer o Chip Mifare do brinco, o bovino será mostrado com sua identificação, liberando assim os campos para inserção de dados, conforme mostrado na Figura 46. Os dados são então inseridos como na Figura 47, permitindo que o usuário registre as informações da vacina. Isso possibilitará a criação de um relatório mais detalhado do bovino.

Figura 47 – Selecionando o Bovino Para Aplicar Medicamento



Fonte: (Elaborado pelo autor, 2024).

Inserindo as vacinas e gerando o relatório, o zootecnista pode optar por criar um relatório (observação) para o bovino ao salvar, conforme mostrado na Figura 48.

Figura 48 – Inserindo as Informações do Medicamento

The screenshot shows a web application interface for managing a bovine animal. On the left, there is a blue sidebar with 'Inicio' and 'Contato' buttons. The main header features a logo with a bull and the text 'M. RUIRHO' and navigation links 'SOBRE PORTFOLIO CONTATO'. The central area is titled 'Bovino: Bovino 1' and displays the following information:

- Raça:** Nelore
- Id brinco:** 91497C1D
- Nascimento:** 03/04/2024
- Peso:** (empty field)

Below this information, there is a form for adding medication:

- A dropdown menu showing 'Baytril Antibiótico'.
- A 'Detalhes do tratamento' section with a scrollable list:
 - Medicamento: Baytril (Enrofloxacino)
 - Dosagem: 2,5 mg/kg
 - Administração: Intramuscular
 - Período de tratamento: 5 dias
 - Data de início: 05/06/2024
- Input fields for '2024-06-05' and '300'.
- A green 'Salvar' button.

Fonte: (Elaborado pelo autor, 2024).

Todas essas informações serão armazenadas e poderão ser visualizadas no relatório do bovino, garantindo assim o completo gerenciamento, conforme mostrado na Figura 49:

Figura 49 – Relatório Detalhado do Bovino com Inserção de Medicament

The screenshot shows a detailed report for 'Bovino 1' with brinco ID 91497C1D. The report is organized into several sections:

- Dados:**

DATA DE NASCIMENTO	PESO DE NASCIMENTO KG	PESO ATUAL KG	RAÇA
03/04/2024 00:00:00	40	300	Nelore
- Alimentação e Suplementação:**

RAÇÃO	CAPIM	SUPLENTO	SILO
- Vacinas:**

NOME	TIPO	DOSAGEM/QUANTIDADE	DATA
• Baytril®	• Antibiótico	• 2,5	• 05/06/2024
- Relatório prescrito pelo Zootecnista:**
 - O relatório foi realizado no dia 05/06/2024 O bovino 047 apresentou sinais de infecção respiratória, incluindo tosse, secreção nasal purulenta e febre (39,5°C). Após a avaliação clínica, foi decidido administrar um tratamento com o antibiótico Baytril (Enrofloxacino). Detalhes do Tratamento Medicamento: Baytril (Enrofloxacino) Dosagem: 2,5 mg/kg Administração: Intramuscular Período de tratamento: 5 dias Data de início: 05/06/2024

A 'Voltar' button is located at the bottom left of the report area.

Fonte: (Elaborado pelo autor, 2024).

4 COMO UTILIZAR O SISTEMA DESENVOLVIDO NO PRESENTE TRABALHO

Neste capítulo explica-se como se pode executar o *frontend* e *backend* para ser rodado no *localhost*.

4.1 Frontend

Inicialmente deve-se utilizar do acesso ao seguinte link para acessar o Rancho *Frontend*, para utilizar o software desenvolvido:

- Frontend: <https://github.com/IsmaelCarlos/rancho-2>

O GitHub se apresenta hoje como repositório universal para manter e gerenciar os protótipos desenvolvidos.

Para utilizar o sistema, é necessário instalar as bibliotecas para serem executadas em um servidor local. O usuário deve acessar o arquivo `package.json` dentro da pasta `rancho-2` e executar o seguinte comando:

- `npm install`

Após a instalação dos pacotes do arquivo `package.json`, dentro da pasta `rancho-2`, o usuário deve executar o seguinte comando:

- `npm run dev`

Executando este comando, suas devidas dependências serão instaladas automaticamente para rodar o *frontend*. Todo o *frontend* foi desenvolvido com TypeScript, CSS, BOOTSTRAP.

4.2 Backend

Todo o *Backend* foi desenvolvido com *Express Framework*. Tal framework Web é rápida, flexível e minimalista para Node.js. Para subir o banco de dados foi utilizado o Docker. O *link* que se segue contém a referência a todo o *backend* para ser baixado:

- Backend: <https://github.com/IsmaelCarlos/rancho-backend>

Para executar o *backend* é necessário instalar as bibliotecas de modo que se possa subir o servidor, elas serão instaladas de forma automática. Basta navegar para o diretório raiz que contém o arquivo `package.json` e executar o comando de instalação.

- npm install

Ao realizar as instalações dos pacotes `package.json`, dentro da pasta `rancho-backend` basta executar o seguinte comando:

- npm run dev

4.3 Banco de Dados

Para executar o banco de dados foi escolhido o *PostgresSql*, pois facilita a comunicação com o *Backend*. É necessário instalar o *Docker* pois todo o banco foi configurado para ser criado no numa imagem do *Docker*. Para subir o banco utiliza-se o seguinte código:

- docker compose up -d --force-recreate

5 CONSIDERAÇÕES FINAIS

O presente trabalho teve como objetivo desenvolver um sistema de gerenciamento para bovinos que seja útil e de fácil compreensão para o usuário. Durante todo o desenvolvimento, pode-se criar uma ferramenta que permitisse ao usuário gerenciar aspectos importantes, como vacinas, dieta dos animais e gerenciamento de estoques, de maneira eficiente.

O sistema proposto visa facilitar o gerenciamento, mantendo a segurança e proporcionando um relatório detalhado de forma rápida e precisa. A interface foi projetada para ser intuitiva e não cansativa para o usuário, eliminando a necessidade de digitar identificações de cada brinco manualmente. Com a tecnologia RFID, o sistema permite que o usuário faça a identificação do brinco apenas aproximando o leitor do brinco, agilizando todo o processo.

Essa abordagem não só economiza tempo, como também reduz a possibilidade de erros humanos na entrada de dados, aumentando a precisão do gerenciamento do rebanho. O sistema automatizado melhora a eficiência operacional e proporciona uma maneira mais prática de monitorar a saúde e o bem-estar dos animais, além de otimizar a administração de vacinas e a definição de dietas.

Em resumo, o sistema desenvolvido oferece uma solução moderna e eficaz para a gestão de bovinos, trazendo benefícios significativos tanto para os gestores quanto para os animais, ao garantir um manejo mais organizado e preciso. Este trabalho contribui para a inovação na pecuária, promovendo o uso de tecnologias avançadas para melhorar a produtividade e a qualidade do gerenciamento do rebanho.

5.1 O Que Foi Desenvolvido

Foi desenvolvido um sistema para a automação de fazendas de bovinos de corte utilizando a tecnologia RFID. O sistema consiste em utilizar um brinco com um chip Mifare, que atua como identificador do bovino. O processo de gerenciamento está funcional para os seguintes módulos:

- Registrar Usuário e Fazenda: Permite o cadastro de usuários e a configuração da fazenda no sistema.
- Registrar o Bovino com o Brinco: Facilita o registro dos bovinos utilizando o brinco RFID para identificação individual.

- Registrar Medicamentos: Permite o registro de medicamentos administrados aos bovinos, garantindo um controle preciso e atualizado.
- Registrar Dietas: Permite a definição e o controle das dietas dos bovinos, ajudando na gestão nutricional.
- Visualizar Relatório do Bovino: Proporciona uma visão detalhada sobre cada bovino, incluindo informações sobre saúde, vacinação e dieta.
- Visualizar Relatório de Estoque de Medicamentos: Oferece um relatório atualizado sobre o estoque de medicamentos disponíveis na fazenda.
- Visualizar Relatório de Estoque de Rações: Fornece um relatório detalhado sobre o estoque de rações, facilitando o planejamento e a logística alimentar.

5.2 Trabalhos Futuros

Para melhorar ainda mais o sistema de automação de fazendas de bovinos de corte, algumas áreas foram identificadas como potenciais para futuros desenvolvimentos e aprimoramentos:

- Integração com Sensores de Saúde: Adicionar sensores para monitorar a temperatura corporal, frequência cardíaca e outros indicadores de saúde dos bovinos, permitindo uma vigilância mais proativa da saúde animal.
- Aplicativo Móvel: Desenvolver um aplicativo móvel que permita aos usuários acessarem e gerenciar o sistema de qualquer lugar, aumentando a conveniência e a flexibilidade na administração da fazenda.
- Análise Preditiva: Implementar algoritmos de análise preditiva para prever problemas de saúde, necessidades nutricionais e otimizar a produção com base nos dados coletados, melhorando a tomada de decisões.
- Integração com Sistemas de Gestão Agrícola: Expandir a integração com outros sistemas de gestão agrícola para centralizar todos os dados relevantes e facilitar a coordenação de diversas atividades na fazenda.
- Melhoria na Interface de Usuário: Continuar aprimorando a interface do usuário para torná-la ainda mais intuitiva e fácil de usar, baseado no feedback dos usuários.
- Segurança e Privacidade dos Dados: Implementar medidas adicionais para garantir a segurança e a privacidade dos dados coletados, incluindo a criptografia de dados e autenticação de usuários.

- **Treinamento e Suporte ao Usuário:** Desenvolver materiais de treinamento e oferecer suporte contínuo para ajudar os usuários a maximizarem o potencial do sistema.
- **Expansão do Sistema para Outras Espécies:** Adaptar o sistema para ser utilizado com outras espécies de animais de criação, ampliando o alcance e a aplicabilidade da tecnologia.
- **Implementação de Inteligência Artificial:** Incorporar técnicas de inteligência artificial para automatizar ainda mais processos e fornecer insights avançados sobre o manejo e a produção.

Esses futuros trabalhos têm o potencial de transformar o sistema em uma ferramenta ainda mais poderosa e abrangente, oferecendo uma gestão completa e eficiente das fazendas de bovinos de corte. Com essas melhorias, espera-se não apenas aumentar a produtividade e reduzir custos, mas também melhorar o bem-estar dos animais e a sustentabilidade da produção.

REFERÊNCIAS

- ANDREI. O que é html: O guia definitivo para iniciantes. 2023. [Acesso em: 28 maio. 2024]. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos>>.
- ARIANE. O que é css? guia básico para iniciantes. Hostinger Tutoriais, 2022. [Acesso em: 27 out. 2023]. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css>>.
- BERTOLETI, P. Objetos em linguagem c. Embarcados, 2017. [Acesso em: 1 jun. 2024]. Disponível em: <<https://embarcados.com.br/objetos-em-linguagem-c/>>.
- COUTO, G. D.; MALAFAIA, T. S. Rfid radio frequency identification. *Escola Politécnica*, Universidade Federal do Rio de Janeiro, 2019. Disponível em: <<https://www.gta.ufrj.br/ensino/eel878/redes1-2019-1/vf/rfid/>>.
- EDSON. Introdução ao typescript. 2024. Acesso em: 30 maio. 2024. Disponível em: <<https://www.devmedia.com.br/introducao-ao-typescript/36729>>.
- EMBRAPA. História da embrapa. [Acesso em: 29 nov. 2023]. Disponível em: <<https://www.embrapa.br/en/gado-de-corte/historia>>.
- EMBRAPA. Qualidade da produção da carne bovina. [Acesso em: 29 nov. 2023]. Disponível em: <<https://www.embrapa.br/en/qualidade-da-carne/carne-bovina/producao-de-carne-bovina>>.
- EMBRAPA. Carne bovina. *Portal embrapa*, 2023. [Acesso em: 25 mar. 2023]. Disponível em: <<https://www.embrapa.br/en/qualidade-da-carne/carne-bovina>>.
- ENGINEERS, L. M. Getting started with esp32: A beginner's guide. jan 2023. Accessed: 2024-6-23.
- ESTRELLA, C. O que é javascript e para que serve na programação web. 2023. Acesso em: 29 nov. 2023. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-javascript>>.
- GROOVER, M. P. Automation. *ENCYCLOPAEDIA BRITANNICA*, 2023. [Acesso em: 19 mar. 2023]. Disponível em: <<https://www.britannica.com/technology/automation>>.
- LASTMINUTEENGINEERS. Introducing the esp32. 2024. [Acesso em: 24 maio. 2024]. Disponível em: <https://lastminuteengineers.com/getting-started-with-esp32/#google_vignette>.
- MDN. Express/node introduction. *MDN Web Docs*, Mozilla Developer Network, 2024. [Acesso em: 1 jun. 2024]. Disponível em: <https://developer.mozilla.org/enUS/docs/Learn/Server-side/Express_Nodejs/Introduction>.

MPS. Adcs for iot applications. *MpScholar*, Monolithic Power Systems, 2024. [Acesso em: 28 maio. 2024]. Disponível em: <<https://www.monolithicpower.com/en/learning/mpscholar/analog-to-digital-converters/future-trends-and-challenges-in-adcs/adcs-for-iot-applications>>.

NXP. Mifare classic ev1 4k - mainstream contactless smart card ic for fast and easy solution development. NXP Semiconductors, 2017. [Acesso em: 29 nov. 2023]. Disponível em: <https://www.nxp.com/docs/en/data-sheet/MF1S70YYX_V1.pdf>.

ORACLE. O que é o docker? 2024. [Acesso em: 30 maio. 2024]. Disponível em: <<https://www.oracle.com/br/cloud/cloud-native/container-registry/what-is-docker/>>.

OTTO, M.; THORNTON, J. Get started with bootstrap. Bootstrap Docs, 2023. [Acesso em: 29 nov. 2023]. Disponível em: <<https://getbootstrap.com/docs/5.3/getting-started/introduction/>>.

PEDROSO, M. C.; ZWICKER, R.; SOUZA, C. A. de. Adoção de rfid no brasil: um estudo exploratório. *RAM Revista de Administração Mackenzie*, SciELO, v. 10, n. 1, 2009. [Acesso em: 19 mar. 2023]. Disponível em: <<https://www.scielo.br/j/ram/a/TwbxMBFsjTTsDZTKdJBYRC/?lang=pt#>>.

POSTGRESQL. About. 2024. Acesso em: 30 maio. 2024. Disponível em: <<https://www.postgresql.org/about/>>.

SANTOS, C.; SALES, J. O desafio da privacidade na internet das coisas. 2016. [Acesso em: 9 de maio. 2016]. Disponível em: <<https://periodicos.ufpe.br/revistas/gestaoorg/article/download/22115/18481>>.

SEMICONDUCTORS, N. *MIFARE Classic EV1 4K - Mainstream contactless smart card IC for fast and easy solution development*. 2017. Acesso em: 23 maio. 2024. Disponível em: <https://www.nxp.com/docs/en/data-sheet/MF1S70YYX_V1.pdf>.

W3C. C introduction. W3Schools, 2023. [Acesso em: 29 nov. 2023]. Disponível em: <https://www.w3schools.com/c/c_intro.php>.

W3SCHOOLS. Xml introduction. W3Schools, 2024. [Acesso em: 28 maio. 2024]. Disponível em: <https://www.w3schools.com/xml/xml_what.asp>.

RESOLUÇÃO nº 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

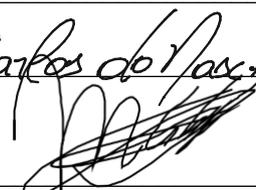
Termo de autorização de publicação de produção acadêmica

O(A) estudante Ismael Carlos do Nascimento Galvão
do Curso de Engenharia de Computação matrícula 20151003302067,
telefone: (62) 9 82244942 e-mail leamsi.carlos100@gmail.com
na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei
dos Direitos do Autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás)
a disponibilizar o Trabalho de Conclusão de Curso intitulado
A utilização do RFID para automatizar fazendas de
café, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos,
conforme permissões do documento, em meio eletrônico, na rede mundial de
computadores, no formato especificado (Texto(PDF); Imagem (GIF ou JPEG); Som
(WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da
área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção
científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 12 de Abril de 2024.

Assinatura do autor: 

Nome completo do autor: Ismael Carlos do Nascimento Galvão

Assinatura do professor-orientador: 

Nome completo do professor-orientador: Marcelo Antônio Adad de Araújo