

Universidade de Brasília – UnB
Faculdade de Ciências e Tecnologias em Engenharia (FCTE)
Engenharia de Software

IrrigaSync: Otimização do Racionamento Hídrico Através de Monitoramento Inteligente de Irrigação

Autor: Caio Vitor Carneiro de Oliveira, Cícero Barrozo
Fernandes Filho

Orientador: Dr. Renato Coral Sampaio

Brasília, DF
2025



Caio Vitor Carneiro de Oliveira, Cícero Barrozo Fernandes Filho

IrrigaSync: Otimização do Racionamento Hídrico Através de Monitoramento Inteligente de Irrigação

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade de Ciências e Tecnologias em Engenharia (FCTE)

Orientador: Dr. Renato Coral Sampaio

Coorientador: Dr. João José Da Silva Junior

Brasília, DF

2025

Caio Vitor Carneiro de Oliveira, Cícero Barrozo Fernandes Filho
IrrigaSync: Otimização do Racionamento Hídrico Através de Monitoramento
Inteligente de Irrigação/ Caio Vitor Carneiro de Oliveira, Cícero Barrozo Fernan-
des Filho. – Brasília, DF, 2025-
64 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. Renato Coral Sampaio

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade de Ciências e Tecnologias em Engenharia (FCTE) , 2025.

1. Palavra-chave01. 2. Palavra-chave02. I. Dr. Renato Coral Sampaio. II.
Universidade de Brasília. III. Faculdade de Ciências e Tecnologias em Engenharia
(FCTE). IV. IrrigaSync: Otimização do Racionamento Hídrico Através de
Monitoramento Inteligente de Irrigação

CDU 02:141:005.6

Caio Vitor Carneiro de Oliveira, Cícero Barrozo Fernandes Filho

IrrigaSync: Otimização do Racionamento Hídrico Através de Monitoramento Inteligente de Irrigação

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Dr. Renato Coral Sampaio
Orientador

Dr. João José Da Silva Junior
Coorientador

Dr. Daniel Chaves Café
Membro Externo

Dr. Maurício Serrano
Membro Interno

Brasília, DF
2025

Agradecimentos

Gostaríamos de expressar nossa profunda gratidão a todos que contribuíram para a realização deste trabalho. Agradecemos primeiramente à nossa família, pelo apoio incondicional e incentivo constante ao longo desta jornada acadêmica. Aos nossos professores e orientadores, pela orientação, paciência e compartilhamento de conhecimento que foram fundamentais para a conclusão deste trabalho. Em especial, agradecemos ao Professor Renato Coral, por suas valiosas contribuições e sugestões, e ao Professor Auxiliar João, pelo suporte e dedicação ao longo do processo. Aos colegas e amigos, pelo companheirismo e pelas discussões construtivas que enriqueceram nosso aprendizado e crescimento pessoal. Às instituições e empresas que forneceram suporte técnico e financeiro, sem os quais este projeto não seria possível. Nosso agradecimento especial à Universidade de Brasília, pela oportunidade e recursos disponibilizados.

Por fim, agradecemos a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho. Sem o apoio de cada um de vocês, esta conquista não seria possível.

Resumo

Este Trabalho de Conclusão de Curso (TCC) apresenta o desenvolvimento do IrrigaSync, um sistema inteligente de monitoramento de umidade do solo voltado para a otimização do uso de recursos hídricos na agricultura. A pesquisa se insere na área de Engenharia de Software aplicada à Agricultura de Precisão, utilizando tecnologias emergentes para minimizar desperdícios e promover a sustentabilidade no manejo da irrigação. O sistema foi projetado para coletar e transmitir dados de umidade do solo utilizando sensores acoplados a microcontroladores ESP32, que se comunicam via LoRa, um protocolo de transmissão sem fio de longo alcance e baixo consumo energético. As informações captadas são enviadas para um servidor central por meio do protocolo MQTT e disponibilizadas em tempo real na plataforma IoT ThingsBoard, permitindo a visualização e análise dos dados por meio de dashboards interativos. A metodologia aplicada seguiu princípios de desenvolvimento ágil, utilizando Scrum para estruturar a implementação do sistema. Foram conduzidos experimentos em uma horta de chuchu, permitindo validar o funcionamento da comunicação sem fio e a eficiência do monitoramento da umidade do solo. Os testes mostraram que a tecnologia é capaz de fornecer comunicação persistente entre os dispositivos e garantir uma transmissão confiável dos dados, embora tenham sido observados desafios relacionados à calibração dos sensores, estabilidade da comunicação em distâncias superiores a 325 metros e consumo energético. Os resultados alcançados indicam que o IrrigaSync pode contribuir para a redução do desperdício de água na agricultura ao fornecer informações precisas e acessíveis para o agricultor, auxiliando na tomada de decisão sobre a irrigação. Entre as melhorias propostas para otimizar o sistema, destacam-se o aprimoramento da escalabilidade com mais sensores distribuídos no campo, otimização do consumo energético por meio de baterias ou painéis solares e o refinamento da calibração dos sensores para garantir maior precisão nas medições.

Palavras-chave: Irrigação inteligente, ESP32, LoRa, MQTT, IoT, monitoramento de solo, sustentabilidade, agricultura de precisão.

Abstract

This Final Paper presents the development of IrrigaSync, an intelligent soil moisture monitoring system aimed at optimizing the use of water resources in agriculture. The research falls within the area of Software Engineering applied to Precision Agriculture, using emerging technologies to minimize waste and promote sustainability in irrigation management. The system was designed to collect and transmit soil moisture data using sensors coupled to ESP32 microcontrollers, which communicate via LoRa, a long-range wireless transmission protocol with low energy consumption. The information captured is sent to a central server via the MQTT protocol and made available in real time on the IoT ThingsBoard platform, allowing data to be visualized and analyzed using interactive dashboards. The methodology applied followed agile development principles, using Scrum to structure the implementation of the system. Experiments were carried out in a chayote garden, allowing the operation of wireless communication and the efficiency of soil moisture monitoring to be validated. The tests showed that the technology is capable of providing persistent communication between the devices and guaranteeing reliable data transmission, although challenges were observed related to sensor calibration, communication stability over distances of more than 325 meters and energy consumption. The results achieved indicate that IrrigaSync can contribute to reducing water wastage in agriculture by providing accurate and accessible information to the farmer, aiding decision-making on irrigation. Improvements proposed to optimize the system include improving scalability with more sensors distributed in the field, optimizing energy consumption by using batteries or solar panels and refining sensor calibration to ensure greater accuracy in measurements. The conclusion is that the system developed has the potential to become a viable solution for efficient irrigation management, promoting water sustainability and increasing agricultural productivity with less environmental impact.

Key-words: smart irrigation, ESP32, LoRa, MQTT, IoT, soil monitoring, sustainability, precision agriculture soil monitoring, sustainability, precision agriculture.

Lista de ilustrações

Figura 1 – Pinout ESP Wi-Fi LoRa 32	28
Figura 2 – Power Bank Peining 10000mAh WUP-653	30
Figura 3 – Sensor de Umidade SEN0193	33
Figura 4 – Arquitetura do Sistema	36
Figura 5 – Esquemático IrrigaSync	38
Figura 6 – Protótipo Dashboard	40
Figura 7 – Horta de Chuchu	41
Figura 8 – Esp da Sede	42
Figura 9 – Esp da horta	43
Figura 10 – Distância entre a Horta e a Central	44
Figura 11 – Esp no Ar	45
Figura 12 – Esp na Água	46
Figura 13 – Gráfico do 1º Experimento	47
Figura 14 – Gráfico do 2º Experimento	48
Figura 15 – Distância Máxima entre a Horta e a Central	49
Figura 16 – Instalação em casa	50
Figura 17 – Resultados experimento controlado	51

Lista de tabelas

Tabela 1 – Modos de energia e seus respectivos consumos.	31
Tabela 2 – Priorização dos Requisitos.	35
Tabela 3 – Mapeamento do Cronograma e Requisitos Concluídos (RC) no Desenvolvimento do Sistema.	39
Tabela 4 – Custo total do experimento	50

Lista de abreviaturas e siglas

IoT	Internet das Coisas
RFID	Identificação por Radiofrequência
CPU	Unidade Central de Processamento
MQTT	MQ Telemetry Transport
LoRa	Long Range
HTTP	Protocolo de Transferência de Hipertexto
DSDM	Método Dinâmico de Desenvolvimento de Software
RFID	Identificação por Radiofrequência
WLAN	Rede Local Sem Fio
WPAN	Rede Pessoal Sem Fio
WMAN	Rede Metropolitana Sem Fio
WWAN	Rede de Área Ampla Sem Fio
Wi-Fi	Wireless Fidelity
API	Interface de Programação de Aplicações
TCC	Trabalho de Conclusão de Curso

Sumário

1	INTRODUÇÃO	13
1.1	Justificativa	14
1.2	Objetivos	14
1.2.1	Objetivo Geral	14
1.2.2	Objetivos específicos	15
1.3	Proposta de Solução	15
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	Irrigação	16
2.1.1	Irrigação Superficial	17
2.1.2	Irrigação por Aspersão	17
2.1.3	Irrigação Localizada	17
2.2	Solo	17
2.2.1	Estrutura e Diversidade Biológica do Solo	18
2.2.2	Diversidade Microbiana no Solo	18
2.2.3	Fatores que Regem a Diversidade Microbiana dos Solos	18
2.2.4	Umidade no Solo	18
2.3	A Cultura do Chuchu e a Umidade do Solo	19
2.4	Condições Climáticas Ideais	19
2.5	Características do Solo e Adubação	20
2.5.1	Implantação e Tutoramento da Cultura	20
2.5.2	Umidade do Solo e Irrigação	20
2.6	Internet das coisas	20
2.7	Redes Wireless	21
2.8	Padrão IEEE 802.11	22
2.9	Topologia em Estrela	22
2.9.1	Características da Topologia em Estrela	22
2.10	LoRa	22
2.11	Plataforma IoT	23
3	METODOLOGIA, MATERIAIS E FERRAMENTAS	24
3.1	Metodologia	24
3.1.1	Tipos de Metodologias	25
3.2	Metodologia de Desenvolvimento	25
3.2.1	Papéis	26
3.2.2	Objetivos	26

3.2.3	MoSCoW	27
3.3	Ferramentas de Desenvolvimento	27
3.3.1	Arduino IDE	27
3.3.2	ThingsBoard	27
3.4	Componentes de Hardware	28
3.4.1	ESP Wi-Fi LoRa 32	28
3.4.1.1	Especificações	28
3.4.2	Power Bank Peining 10000mAh WUP-653	29
3.4.2.1	Especificações	30
3.4.3	Modos de Energia e Consumo	31
3.4.3.1	Descrição dos Modos de Energia	31
3.4.4	Sensor de Umidade DFROBOT SEN0193	32
3.4.4.1	Especificações	32
4	PROPOSTA DE SOLUÇÃO	34
4.1	Requisitos do Sistema	34
4.2	Arquitetura do Sistema	35
4.2.1	Sensor de Umidade	35
4.2.2	ESP32 (Nó Sensor)	35
4.2.3	LoRa	35
4.2.4	WI-FI	37
4.2.5	MQTT	37
4.2.6	ESP32 (Gateway)	37
4.2.7	ThingsBoard	38
4.3	Esquemático do Sistema	38
4.4	Fluxo de Dados	39
4.5	Cronograma do Projeto	39
5	TESTES E RESULTADOS	40
5.1	Funcionamento do Thingsboard	40
5.1.1	Localização do Experimento	40
5.2	Conexão com LORA	41
5.2.1	Distância entre as ESPs	43
5.3	Procedimento de Calibração	44
5.3.1	Determinação do Valor para Solo Seco	44
5.3.2	Determinação do Valor para Solo Úmido	45
5.3.3	Cálculo do Percentual de Umidade	46
5.4	Experimentos não controlados	47
5.4.1	1º Experimento	47
5.4.2	2º Experimento	47

5.4.3	Distância Máxima entre as ESPs	48
5.5	Experimentos Controlados	49
5.5.1	1º Experimento	49
5.6	Custo Total	50
5.7	Possíveis Melhorias	51
5.7.1	Escalabilidade do Sistema	51
5.7.2	Otimização do Consumo de Energia	51
5.7.3	Mudança de Antena	52
5.7.4	Melhoria de Infraestrutura	52
6	CONCLUSÃO	53
7	ANEXOS	54
7.1	Código	54
	REFERÊNCIAS	63

1 Introdução

A água é um recurso essencial para a agricultura, sendo utilizada tanto na irrigação das plantações quanto na criação de animais. A irrigação, em particular, desempenha um papel crucial na produção agrícola, permitindo que as culturas cresçam em regiões com baixa precipitação e ajudando a aumentar a produtividade das colheitas.

Embora a agricultura irrigada seja, geralmente, associada a um elevado nível tecnológico, é consenso que a irrigação no Brasil é ainda praticada de forma inadequada, com grande desperdício de água (E.C, 2006). Este problema não apenas compromete a eficiência da produção agrícola, mas também exerce uma pressão significativa sobre os recursos hídricos do país, que são finitos e necessitam ser utilizados de maneira sustentável.

Com o aumento contínuo da população mundial, a demanda por alimentos está crescendo rapidamente (ORGANIZATION; AGRICULTURE, 2006). Isso coloca uma pressão adicional sobre os recursos hídricos, pois a agricultura precisa aumentar a produção para alimentar mais pessoas. Para enfrentar esse desafio, é essencial adotar práticas de irrigação mais eficientes e tecnologias avançadas que minimizem as perdas de água. Isso inclui a utilização de sistemas de irrigação que permitam um controle mais preciso da quantidade de água aplicada às culturas, bem como a implementação de técnicas de manejo da irrigação que otimizem o uso da água disponível.

Além disso, o uso de sensoriamento agrícola e da *IoT* está se tornando cada vez mais relevante como solução para a gestão eficiente da água na agricultura. Sensores instalados no solo podem fornecer dados em tempo real sobre a umidade do solo, necessidades hídricas das culturas e condições climáticas. Esses dados podem ser analisados e utilizados para ajustar automaticamente os sistemas de irrigação, garantindo que a água seja aplicada de maneira precisa e eficiente.

Os sistemas embarcados, que integram hardware e software específicos para monitoramento e controle, também desempenham um papel importante. Eles podem ser programados para tomar decisões autônomas baseadas nos dados coletados pelos sensores, otimizando o uso da água e reduzindo o desperdício. A implementação dessas tecnologias permite uma abordagem mais inteligente e sustentável na gestão dos recursos hídricos, contribuindo para a segurança alimentar e a preservação ambiental.

Neste contexto, o presente estudo visa analisar as práticas de irrigação atuais e propor soluções que possam melhorar a eficiência do uso da água na agricultura brasileira. Espera-se que, ao adotar práticas mais sustentáveis, seja possível não apenas aumentar a produtividade agrícola, mas também preservar os recursos hídricos para as futuras gerações.

1.1 Justificativa

A agricultura desempenha um papel fundamental na economia brasileira, sendo responsável por uma significativa parcela da produção de alimentos e exportações do país. No entanto, um dos principais desafios enfrentados pelo setor é a gestão eficiente dos recursos hídricos. A água, como recurso limitado e essencial, exige uma administração cuidadosa, especialmente em um cenário onde a demanda por alimentos cresce de forma acelerada, impulsionada pelo aumento da população global.

A prática da irrigação, embora crucial para garantir a produtividade agrícola, muitas vezes é realizada de maneira ineficiente, resultando em desperdício de água e consequente impacto negativo nos recursos hídricos disponíveis. Este problema é particularmente relevante em regiões onde a escassez de água já é uma realidade, tornando a adoção de tecnologias que promovam o uso racional da água uma necessidade urgente. De acordo com a Organização das Nações Unidas para a Alimentação e a Agricultura ([ORGANIZATION; AGRICULTURE, 2006](#)), cerca de 70% da água doce disponível no mundo é utilizada na agricultura, principalmente para a irrigação de culturas agrícolas. Este dado reflete a importância crítica de práticas de irrigação eficientes para garantir o uso sustentável dos recursos hídricos.

Diante desse contexto, a implementação de sistemas de monitoramento e controle que utilizem tecnologias modernas, como a *IoT* e sistemas embarcados, torna-se uma solução viável e eficaz. Estes sistemas permitem um monitoramento contínuo e preciso da umidade do solo, possibilitando ajustes automáticos na irrigação e evitando o desperdício de água.

Este projeto justifica-se pela necessidade de aprimorar a eficiência do uso da água na agricultura, contribuindo para a sustentabilidade dos recursos hídricos e garantindo a segurança alimentar. Ao desenvolver um sistema que monitorea a umidade do solo e fornece dados em tempo real para os agricultores, espera-se não apenas melhorar a gestão da água, mas também aumentar a produtividade agrícola e reduzir custos operacionais. Além disso, a disseminação de tais tecnologias pode representar um avanço significativo para a agricultura brasileira, promovendo práticas mais sustentáveis e adaptadas às necessidades atuais e futuras.

1.2 Objetivos

1.2.1 Objetivo Geral

Por isso, o objetivo deste projeto é coletar e analisar informações que possam auxiliar em melhorias na produção agropecuária e que possam fornecer dados relevantes para os agricultores e engenheiros agrônômicos. Tal concepção possibilitará a criação de

um sistema embarcado que monitora a umidade do solo disponibilizando tais informações por painéis, o que proporcionará uma ação baseada em dados sobre a umidade do solo.

1.2.2 Objetivos específicos

- Realizar o levantamento de requisitos, para melhor atender as necessidades do cliente
- Desenvolver um protótipo para validação
- Implementar uma rede de comunicação sem fio para transmissão de dados
- Disponibilizar os dados por aplicação webinterfaces gráficas
- Investigar a possibilidade da inclusão de acesso remoto ao software que será construído
- Investigar soluções para tolerância a falhas dos sensores

1.3 Proposta de Solução

Diante do problema do desperdício de água na agricultura e da necessidade de um gerenciamento mais eficiente da irrigação, este trabalho propõe o desenvolvimento do IrrigaSync, um sistema de monitoramento inteligente da umidade do solo baseado em tecnologias de Internet das Coisas (IoT) e comunicação sem fio de longo alcance. O objetivo é fornecer uma solução acessível, automatizada e de fácil implementação, permitindo que agricultores monitorem a umidade do solo em tempo real e tomem decisões mais precisas sobre a irrigação, reduzindo desperdícios e otimizando o uso dos recursos hídricos.

O IrrigaSync será composto por sensores de umidade do solo integrados a microcontroladores ESP32-Lora, que se comunicam entre si por meio do protocolo sem fio LoRa (Long Range). Esses dispositivos coletarão dados continuamente e os transmitirão para um servidor central, utilizando o protocolo MQTT. As informações serão processadas e disponibilizadas na plataforma ThingsBoard, permitindo a visualização dos dados por meio de dashboards interativos acessíveis via computador ou dispositivos móveis.

2 Fundamentação Teórica

Este capítulo traz os fundamentos teóricos que sustentam o desenvolvimento deste trabalho, abordando aspectos essenciais para entender o contexto agrícola e tecnológico tratado. Aqui, exploramos temas que vão desde os métodos de irrigação até as inovações tecnológicas que estão transformando a gestão agrícola.

Primeiramente, discutimos os principais métodos de irrigação, destacando suas características, aplicações e adequações a diferentes tipos de solo e condições climáticas. Na sequência, abordamos o papel fundamental do solo na agricultura, com foco em sua estrutura, diversidade biológica e o impacto da umidade no desenvolvimento das plantas.

Também exploramos o avanço das tecnologias de Internet das Coisas (IoT) e redes sem fio, mostrando como essas inovações contribuem para o monitoramento e controle em ambientes agrícolas. Detalhamos soluções específicas, como o padrão IEEE 802.11 para conectividade, a tecnologia LoRa, que permite comunicação de longo alcance com baixo consumo de energia, e plataformas como o ThingsBoard, que auxiliam na visualização e análise de dados de forma inteligente e integrada.

Ao longo deste capítulo, mostramos como a combinação de boas práticas agrícolas e tecnologias modernas pode transformar a gestão dos recursos no campo. Essa integração contribui para práticas mais eficientes de irrigação, o uso sustentável do solo e a superação de desafios enfrentados pelos agricultores. Dessa forma, os conceitos aqui apresentados servem como base para as soluções desenvolvidas ao longo deste trabalho.

2.1 Irrigação

A irrigação é uma prática agrícola essencial para garantir a produtividade e a sustentabilidade das plantações, especialmente em regiões onde a distribuição natural das chuvas é irregular. Sua importância se dá pela capacidade de corrigir a distribuição desigual de precipitações, promovendo um ambiente ideal para o desenvolvimento das culturas. O manejo adequado da irrigação, aliado a outras práticas agrícolas, pode maximizar a produção agrícola, tornando-se uma técnica vital nas regiões Nordeste e Centro-Sul do Brasil, historicamente afetadas por variações climáticas significativas (U.R.III, 2014).

Com o aumento dos custos da terra e a necessidade crescente de capital para a produção agrícola, além da alta demanda por alimentos, a dependência exclusiva do regime natural de chuvas tornou-se inviável. Esses fatores impulsionaram o interesse por técnicas de irrigação em todas as regiões agrícolas do Brasil, proporcionando inúmeros benefícios aos produtores (U.R.III, 2014).

Atualmente, existem diversos métodos de irrigação, cada um com características e custos distintos. A escolha do método mais adequado é crucial para o sucesso da produção, uma vez que uma escolha inadequada pode levar a investimentos elevados e riscos à produção. Os métodos de irrigação são classificados de acordo com a forma de distribuição da água: superficiais, subsuperficiais, por aspersão e localizada.

2.1.1 Irrigação Superficial

A irrigação superficial é um método onde a água é aplicada diretamente ao solo, fluindo por gravidade. Durante seu percurso, a água infiltra-se no solo, necessitando técnicas para controlar seu avanço e infiltração adequadamente. Esse método é indicado para solos profundos, de camada fina a média e com baixa permeabilidade, além de terrenos com declives suaves. No entanto, podem ocorrer perdas de água por escoamento superficial, erosão do solo, percolação e estresse hídrico nas raízes. Apesar disso, a irrigação superficial apresenta baixo custo inicial e manutenção reduzida ([LEVIEN, 2003](#)).

2.1.2 Irrigação por Aspersão

A irrigação por aspersão simula a chuva, distribuindo água em forma de gotas sobre a plantação através de aspersores. Esse método não requer preparação do terreno e pode ser utilizado em diversos tipos de solo e topografias. Além disso, permite a aplicação conjunta de agrotóxicos e fertilizantes e pode ser automatizado. No entanto, apresenta desvantagens como maior gasto energético, interferência do vento, alto custo inicial e perdas de água por evaporação ([BISCARO, 2014](#)).

2.1.3 Irrigação Localizada

A irrigação localizada aplica água diretamente nas proximidades das raízes das plantas, utilizando tubulações de baixa pressão e emissores como gotejadores e microaspersores. Esse método apresenta alta eficiência na aplicação de água, reduzindo perdas por evaporação e percolação, além de proporcionar economia elétrica e baixa necessidade de mão de obra especializada ([BISCARO, 2014](#)).

2.2 Solo

O solo é um dos principais compartimentos da biosfera, atuando como um reservatório biológico crucial e um suporte essencial para o sistema agrícola e diversas atividades humanas. Ele desempenha um papel vital como reservatório de água, suportando a vida microbiana e contribuindo significativamente para a manutenção do equilíbrio ecológico.

Este capítulo discute a importância do solo, sua estrutura, diversidade biológica e os fatores que determinam essa diversidade.

2.2.1 Estrutura e Diversidade Biológica do Solo

O solo é um sistema biológico dinâmico e heterogêneo, considerado o principal reservatório da diversidade biológica na biosfera. Ele é composto por uma malha estreita de inter-relações entre os organismos presentes, essencial para a manutenção da ciclagem de nutrientes (RAAIJMAKERS, 2009). A estrutura do solo é caracterizada por sua composição física e química, que varia ao longo do tempo e do espaço, permitindo a formação de micro-habitats que suportam uma diversidade de microrganismos com habilidades metabólicas distintas (MOREIRA Fátima M. S.;SIQUEIRA, 2006),(DUCHICELA, 2013).

2.2.2 Diversidade Microbiana no Solo

A abundância e diversidade de vida microbiana no solo são notáveis. Estudos estimam que a quantidade média de células vivas no solo varia entre 10^7 a 10^9 por grama de solo, com cerca de 10.000 espécies bacterianas por grama (ALEXANDER, 1977). Essa diversidade é essencial para a funcionalidade do solo, garantindo a ocorrência de diversos processos biogeoquímicos frente às variações ambientais (AL., 2012). A perda dessa diversidade compromete a funcionalidade e a saúde do solo.

2.2.3 Fatores que Regem a Diversidade Microbiana dos Solos

A diversidade microbiana do solo é influenciada por fatores abióticos (como atmosfera, temperatura, água, pH e potencial redox) e bióticos (como genética microbiana e interações entre microrganismos). Esses fatores determinam a ecologia, atividade e dinâmica populacional dos microrganismos no solo (MOREIRA Fátima M. S.;SIQUEIRA, 2006).

2.2.4 Umidade no Solo

A água é um elemento essencial para a vida no solo, atuando como solvente natural onde ocorrem as principais reações bioquímicas intracelulares. Nos solos, a água pode ser encontrada em três formas distintas: água higroscópica (aderida às partículas do solo), água capilar (contida nos microporos) e água livre ou gravitacional (escoa pelos macroporos) (MOREIRA Fátima M. S.;SIQUEIRA, 2006).

A água influencia diretamente a atividade biológica do solo, participando na difusão de nutrientes, motilidade microbiana, determinação dos valores de pH e potencial redox, além de estar relacionada com a temperatura e a aeração. As variações sazonais de temperatura e umidade resultam em ciclos de seca/umidade que liberam substratos das

superfícies das argilas ou de células mortas, estimulando a atividade metabólica do solo (MOREIRA Fátima M. S.;SIQUEIRA, 2006).

2.3 A Cultura do Chuchu e a Umidade do Solo

A escolha da cultura do chuchu (*Sechium edule*) como objeto de estudo neste trabalho se deve a diversos fatores técnicos e agrônômicos. O chuchu é uma hortaliça amplamente cultivada no Brasil, apresentando grande importância econômica e nutricional. Sua produtividade é altamente influenciada pelo manejo adequado da irrigação, tornando-o um excelente modelo para a aplicação de tecnologias voltadas ao monitoramento da umidade do solo.

O chuchuzeiro é uma planta de crescimento vigoroso e alta demanda hídrica, exigindo um suprimento contínuo de água para garantir um desenvolvimento saudável e uma produção satisfatória. No entanto, tanto o excesso quanto a escassez de umidade podem impactar negativamente a cultura. O excesso de água pode levar ao desenvolvimento de doenças fúngicas, como oídio e antracnose, enquanto a deficiência hídrica pode comprometer o enchimento dos frutos e reduzir a produtividade. Dessa forma, o monitoramento preciso da umidade do solo se torna essencial para otimizar a irrigação e garantir a eficiência do uso dos recursos hídricos.

Dessa forma, a cultura do chuchu foi selecionada por sua importância agrícola, pela influência direta da umidade do solo em sua produtividade e pelo potencial de validação e replicação da tecnologia em outras culturas. A implementação do sistema IrrigaSync permitirá testar a eficácia do monitoramento da umidade e a otimização do manejo hídrico, contribuindo para uma irrigação mais eficiente e sustentável.

2.4 Condições Climáticas Ideais

O chuchuzeiro prefere temperaturas amenas, variando entre 15°C e 28°C. O frio é um fator limitante, pois a cultura não tolera geadas, enquanto temperaturas acima de 28°C podem levar à brotação excessiva e queda de flores e frutos. Em termos de altitude, a produção é mais eficiente em locais acima de 700 metros, com ótimos resultados entre 1.000 e 1.200 metros.

O chuchu pode ser cultivado ao longo de todo o ano em regiões de clima quente, desde que haja irrigação adequada. Em locais como o Distrito Federal, recomenda-se o plantio entre outubro e fevereiro, coincidindo com a estação chuvosa, o que minimiza o impacto de temperaturas mais baixas e ventos fortes.

2.5 Características do Solo e Adubação

Embora a cultura do chuchu não seja altamente exigente em fertilidade, os melhores resultados são obtidos em solos de textura média, soltos, leves e com altos níveis de fertilidade. Como o chuchu é uma planta perene, sua adubação deve ser planejada para sustentar a cultura por longos períodos([ALVARENGA FAUSTO VEIGA DE; ALVARES, 2021](#)).

2.5.1 Implantação e Tutoramento da Cultura

O plantio do chuchu é feito com frutos maduros e brotados, chamados chuchusmente. O espaçamento pode variar entre 5 x 5 metros, 4 x 5 metros, 4 x 3 metros e 3 x 3 metros, dependendo das condições climáticas e do tempo de permanência da cultura no campo.

A cultura deve ser tutorada, sendo o sistema de latada (caramanchão) o mais utilizado. Essa estrutura melhora a qualidade dos frutos, facilita a colheita e reduz a incidência de pragas. A construção da latada é feita com mourões de eucalipto e arames dispostos a 1,80 metro de altura, permitindo um suporte adequado às ramas da planta.

2.5.2 Umidade do Solo e Irrigação

O chuchuzeiro é extremamente sensível à umidade do solo. O excesso de água pode causar doenças como oídio e antracnose, enquanto a deficiência hídrica compromete o desenvolvimento dos frutos. O manejo adequado da irrigação é essencial para equilibrar a umidade e garantir um crescimento adequado.

O sistema mais utilizado é a irrigação por aspersão sobre a latada, pois promove umidade uniforme e melhora a absorção de nutrientes. Outros sistemas, como microaspersão e sulcos, também são empregados em determinadas regiões, enquanto a irrigação por gotejamento não é comumente utilizada([ALVARENGA FAUSTO VEIGA DE; ALVARES, 2021](#)).

2.6 Internet das coisas

A evolução contínua da tecnologia tem desencadeado revoluções significativas em como interagimos com o mundo ao nosso redor. Uma dessas revoluções, a *IoT*, representa uma expansão profunda das capacidades da internet, estendendo sua conexão além dos computadores tradicionais para incluir uma gama vasta de dispositivos e objetos do dia a dia. Mattern e Floerkemeier, assim como Faccioni Filho, descrevem a *IoT* como uma “nova visão para a conectividade proporcionada pela internet, em que passa a interligar não só computadores, como também, objetos do dia a dia” ([MATTERN; FLOERKEMEIER,](#)

2010). Este conceito não só reconfigura nossa compreensão sobre redes e conectividade mas também abre novas fronteiras em termos de possibilidades tecnológicas e interações digitais.

Originada do poder simbólico do termo “internet”, a *IoT* incorpora a ideia de uma rede globalmente conectada que não se limita a transmissões de dados entre computadores, mas engloba uma interconexão sem precedentes entre objetos físicos. Esses objetos, equipados com sensores e atuadores, são capazes de coletar dados, reagir a estímulos externos e comunicar-se entre si e com os usuários, criando um ecossistema de máquinas inteligentes e automatizadas.

A necessidade de uma infraestrutura que suporte bilhões de dispositivos conectados foi um dos catalisadores para a transição do protocolo de internet IPv4 para o IPv6, que, com seu espaço de endereçamento praticamente ilimitado, é fundamental para a escalabilidade da *IoT*. Esta transformação não é apenas técnica, mas também conceitual, pois redefine o papel da internet em nossa sociedade, destacando seu potencial como uma plataforma para inovação contínua em diversas áreas, desde a automação residencial até soluções complexas para cidades inteligentes.

Historicamente, a *IoT* se enraíza na tecnologia de Identificação por Radiofrequência (RFID), que teve seus primeiros usos durante a Segunda Guerra Mundial e evoluiu significativamente para permitir o desenvolvimento das funcionalidades que hoje caracterizam a Internet das Coisas. Além disso, discute-se como a convergência entre *IoT* e *Big Data* está modelando um novo paradigma em termos de coleta, análise e utilização de dados em escala global, impactando diretamente na economia, na gestão de recursos e na qualidade de vida das populações.

Nas páginas seguintes, será abordado como a *IoT* está moldando o futuro da tecnologia e da interação humana com máquinas, desvendando um universo de possibilidades que poderia parecer ficção científica apenas algumas décadas atrás.

2.7 Redes Wireless

O termo *wireless*, que em tradução livre significa “sem fio”, refere-se a qualquer tipo de conexão para transmissão de informações que não utiliza cabos ou fios. Redes *wireless* operam com equipamentos que utilizam radiofrequência, como comunicação via satélite e comunicação via infravermelho, entre outros. Essas redes são categorizadas em quatro tipos principais: rede local (WLAN), rede de curta distância (WPAN), rede metropolitana (WMAN), e rede de longa distância (WWAN) (PINHEIRO, 2005).

Quase simultaneamente ao surgimento dos notebooks, surgiu o desejo de conectar computadores à internet sem a necessidade de fios. Em resposta a essa demanda, diversos

grupos começaram a trabalhar para alcançar esse objetivo. A solução mais prática encontrada foi equipar notebooks com transmissores e receptores de rádio de ondas curtas, permitindo comunicação entre eles. Isso levou muitas empresas a comercializarem redes sem fio ([TANENBAUM, 2021](#)).

2.8 Padrão IEEE 802.11

Inicialmente, as redes sem fio comercializadas não eram compatíveis entre si. Por exemplo, computadores de diferentes marcas não conseguiam se comunicar. Para resolver esse problema, a indústria decidiu adotar um padrão para as redes sem fio e delegou essa tarefa a um comitê da IEEE. Em 1990, foi criado o padrão 802.11 ([TANENBAUM, 2021](#)).

2.9 Topologia em Estrela

A topologia em estrela é uma das configurações de rede mais comuns e amplamente utilizadas em redes locais (LANs). Nesse arranjo, todos os dispositivos da rede, também conhecidos como nós, são conectados a um ponto central, geralmente um switch ou um hub. Este ponto central, conhecido como nó mestre, é responsável por gerenciar e direcionar a comunicação entre os dispositivos na rede ([TANENBAUM, 2021](#)).

2.9.1 Características da Topologia em Estrela

Na topologia em estrela, cada dispositivo na rede é conectado ao nó central através de um cabo individual. Isso significa que todas as comunicações entre dispositivos devem passar pelo nó central, que atua como um mediador. Essa configuração oferece algumas vantagens, como a facilidade de adição ou remoção de dispositivos sem interromper a rede. Além disso, a falha de um único cabo ou dispositivo não afeta a rede como um todo, exceto se o nó central falhar ([TANENBAUM, 2021](#)).

A topologia em estrela também permite o monitoramento centralizado e a detecção de falhas de forma eficiente. Como o tráfego de dados é direcionado através do nó central, é possível identificar rapidamente problemas de comunicação ou congestionamento. Essa característica faz da topologia em estrela uma escolha popular em ambientes corporativos, onde a confiabilidade e a manutenção são cruciais ([TANENBAUM, 2021](#)).

2.10 LoRa

O LoRa (Long Range) é uma tecnologia de comunicação sem fio que utiliza modulação de rádio de espectro expandido para oferecer transmissões de dados a longas distâncias com baixíssimo consumo de energia. Ele é utilizado para dispositivos de *IoT*

em ambientes distribuídos, sendo capaz de operar em frequências de rádio sem licença, como as bandas ISM (Industrial, Scientific, and Medical), geralmente em 433 MHz, 868 MHz ou 915 MHz. Sua principal característica é permitir a comunicação em áreas amplas, mesmo em locais remotos, sem a necessidade de infraestruturas complexas de rede, o que o torna altamente eficiente para monitoramento e controle remoto de dispositivos.

2.11 Plataforma IoT

ThingsBoard.io é uma plataforma open source para *IoT*, amplamente utilizada para a coleta, processamento e visualização de dados. Ela fornece uma infraestrutura escalável e resiliente, com suporte a protocolos como *MQTT* e HTTP, permitindo o gerenciamento eficiente de dispositivos e dados em ambientes *IoT*.

Entre suas principais funcionalidades estão o gerenciamento de dispositivos, criação e gerenciamento de dashboards, e a administração de usuários e grupos. A plataforma permite que dispositivos sejam provisionados e gerenciados de forma centralizada, oferecendo controle seguro das aplicações por meio de *APIs* do lado do servidor.

Além disso, o ThingsBoard permite definir relações entre dispositivos, clientes e outras entidades, facilitando a integração e a orquestração de diversos componentes do sistema (THINGSBOARD2024, 2024). A plataforma é projetada para ser escalável e tolerante a falhas, garantindo a segurança e a integridade dos dados, mesmo quando há um grande volume de informações a ser processado.

O ThingsBoard depende de uma conexão com a Internet para funcionar, o que possibilita a visualização de dados em painéis personalizados e flexíveis, que podem ser compartilhados com os clientes do sistema. A plataforma também oferece a capacidade de definir regras de processamento de dados, permitindo a criação de alarmes para eventos críticos, como, por exemplo, alertas para temperaturas que ultrapassam limites estabelecidos (THINGSBOARD2024, 2024).

3 Metodologia, Materiais e Ferramentas

Este capítulo apresenta os métodos, materiais e ferramentas utilizados no desenvolvimento do projeto, descrevendo detalhadamente os componentes de hardware, suas especificações técnicas e suas funções no sistema. A seleção dos materiais foi feita com base na eficiência, compatibilidade e adequação às exigências do projeto, considerando a necessidade de comunicação em redes de baixa potência e o monitoramento preciso de variáveis ambientais.

Inicialmente, são discutidas as principais características do microcontrolador ESP Wi-Fi LoRa 32, que desempenha um papel central na comunicação e controle do sistema. Em seguida, abordam-se os componentes essenciais de suporte, como a bateria Li-Ion 18650, seu suporte dedicado e o regulador de tensão SDB628 Step Up, fundamentais para garantir alimentação energética estável e eficiente ao dispositivo.

Os modos de economia de energia do ESP Wi-Fi LoRa 32, como Light Sleep e Deep Sleep, são explorados em detalhe, demonstrando como esses recursos contribuem para prolongar a autonomia do sistema, especialmente em cenários de operação remota. Por fim, destaca-se o uso do Sensor de Umidade SEN0193, um componente chave para monitorar a umidade do solo com precisão, fornecendo dados confiáveis para o funcionamento adequado do sistema de monitoramento IoT.

A integração desses elementos demonstra a escolha criteriosa de tecnologias e componentes voltados para aplicações de IoT, com foco em eficiência energética, durabilidade e desempenho confiável.

3.1 Metodologia

A pesquisa científica tem como objetivo conhecer e explicar os fenômenos que ocorrem no mundo. Ela sempre começa com uma interrogação, uma grande pergunta que a motiva. O processo de investigação científica é consequência dos questionamentos sobre fatos e fenômenos observados na realidade, originando dúvidas que geram questionamentos, os quais desencadeiam atividades operacionais e intelectuais próprias de uma investigação científica. Como explica Trujillo Ferrari ([SILVA, 2015](#)), a pesquisa “destina-se a duas finalidades mais amplas que a simples procura de respostas.” Essas finalidades estão:

- vinculadas ao enriquecimento teórico das ciências;
- relacionadas com o valor prático ou pragmático.

3.1.1 Tipos de Metodologias

A pesquisa científica pura, também conhecida como teórica ou básica, tem como objetivo articular conceitos e sistematizar a produção de conhecimento em uma área específica, segundo (MINAYO, 2001). Essa abordagem visa criar novas questões em um processo contínuo de incorporação e superação do que já foi produzido. De acordo com Trujillo Ferrari (1982), a pesquisa teórica busca aprimorar o próprio conhecimento, contribuindo para a compreensão e explicação de fenômenos. Nessa modalidade de pesquisa, os pesquisadores trabalham para desenvolver novas teorias, enriquecendo o entendimento sobre diversos tópicos. Roesch (2009) fornece exemplos de questões abordadas na pesquisa teórica em ciências sociais, tais como a natureza da cultura, seu surgimento, a maneira como é transmitida, os motivos do comportamento dos indivíduos, e as estruturas e processos da organização social humana. Em resumo, a pesquisa teórica não apenas amplia o conhecimento existente, mas também estabelece as bases para futuras investigações e descobertas científicas.

A pesquisa científica aplicada tem como objetivo principal gerar soluções práticas para problemas humanos específicos, proporcionando um entendimento profundo de como lidar com essas questões. Conforme destacado por Trujillo Ferrari (1982), “não obstante a finalidade prática da pesquisa, ela pode contribuir teoricamente com novos fatos para o planejamento de novas pesquisas ou mesmo para a compreensão teórica de certos setores do conhecimento”.

O presente trabalho pode ser classificado como uma pesquisa aplicada, cujo objetivo é gerar conhecimento para aplicação prática, visando resolver problemas específicos. Dessa forma, trata-se de uma pesquisa de cunho prático, focada em solucionar as questões previamente estabelecidas nos requisitos.

3.2 Metodologia de Desenvolvimento

A metodologia de desenvolvimento de software adotada na criação do sistema foi o Scrum, um dos métodos ágeis amplamente utilizados. De acordo com Sutherland e Schwaber, conforme descrito no *The Scrum Guide*, o Scrum segue uma abordagem iterativa e incremental, com o objetivo de otimizar a previsibilidade e controlar riscos.

Essa metodologia é sustentada pelos pilares empíricos do Scrum: transparência, inspeção e adaptação, os quais são implementados por meio de quatro eventos formais dentro de um evento maior, denominado *Sprint*. A seguir, serão detalhados os papéis do Scrum, conforme aplicados no desenvolvimento deste sistema.

A elicitação de requisitos foi realizada a partir de levantamento bibliográfico e entrevistas com stakeholders, garantindo que o sistema atendesse às necessidades de mo-

nitoramento eficiente da unidade do solo. Para a priorização dos requisitos, utilizou-se a técnica *MoSCoW*, que classifica as funcionalidades do sistema conforme sua importância para o projeto.

3.2.1 Papéis

- **Stakeholders:** São as partes interessadas no projeto, responsáveis por ajudar na definição de requisitos e no planejamento do *Product Backlog*. Entre os principais stakeholders do projeto estão, por exemplo, os Professores Doutores Renato Coral Sampaio e João José.
- **Desenvolvedores:** São os responsáveis por cumprir o *Sprint Backlog*, permitindo a evolução incremental do produto. Além disso, definem os padrões de qualidade e auxiliam na construção e priorização do *Sprint Backlog*. Os autores deste trabalho de conclusão atuaram como desenvolvedores.

3.2.2 Objetivos

A pesquisa exploratória visa ampliar o conhecimento sobre um determinado fenômeno. Segundo os autores, esse tipo de pesquisa, aparentemente simples, explora a realidade em busca de maior compreensão, para posteriormente planejar uma pesquisa descritiva.

A pesquisa descritiva procura conhecer a realidade estudada, suas características e problemas. Seu objetivo é “descrever com exatidão os fatos e fenômenos de determinada realidade” (TRIVIÑOS, 2009).

A pesquisa explicativa se centra na identificação de fatores determinantes ou contribuintes para o desencadeamento de fenômenos. Seu objetivo é explicar a razão pela qual ocorrem determinados eventos sociais ou naturais. No campo social, a complexidade aumenta devido à temporalidade do fenômeno e à necessidade de situar o ambiente social de ocorrência. Portanto, a realidade tempo-espaco é fundamental na identificação de causa e efeito. Os procedimentos básicos incluem registrar, classificar, identificar e aprofundar a análise.

O presente trabalho se caracteriza como uma pesquisa exploratória, com o objetivo de ampliar o conhecimento sobre um determinado fenômeno e gerar insights iniciais que sirvam de base para estudos mais aprofundados. A pesquisa exploratória busca compreender a realidade de forma abrangente, preparando o terreno para investigações subsequentes (GERHARDT TE; SILVEIRA, 2009). Portanto, trata-se de uma pesquisa prática que visa abordar as questões previamente estabelecidas na Questão de Pesquisa, proporcionando uma base sólida para análises futuras e desenvolvimento de soluções.

3.2.3 MoSCoW

Antes de tomar decisões arquiteturais e iniciar o desenvolvimento da ferramenta, foi realizada uma etapa de elicitação de requisitos. O objetivo dessa fase foi detalhar os elementos considerados relevantes ou essenciais para a construção do sistema.

Após a identificação desses requisitos com maior precisão, procedeu-se à sua priorização. Essa priorização foi realizada utilizando a técnica *MoSCoW*, uma abordagem colaborativa derivada do *Dynamic Software Development Method (DSDM)*, que classifica cada requisito em uma das quatro categorias de importância:

- **Must have** ("Essencial"): atribuído a um requisito crítico para o projeto;
- **Should have** ("Importante"): refere-se a uma funcionalidade de alta prioridade, mas não crítica para o projeto;
- **Could have** ("Desejável"): corresponde a um requisito desejável, mas não essencial;
- **Won't have** ("Não necessário"): descreve um requisito que não será implementado na fase atual do projeto, mas que pode ser considerado para futuras implementações.

3.3 Ferramentas de Desenvolvimento

3.3.1 Arduino IDE

Arduino IDE é um ambiente de desenvolvimento integrado (*Integrated Development Environment*) utilizado para programar placas da família Arduino e microcontroladores compatíveis, incluindo o ESP32. Ele fornece uma interface simples e intuitiva para escrita, compilação e upload de código, suportando a linguagem de programação baseada em C/C++. Além disso, o Arduino IDE conta com uma vasta comunidade de usuários, bibliotecas prontas para diversas funcionalidades e suporte a múltiplas plataformas. (Arduino, 2024)

- **Componentes:** FreeRTOS, LWIP, MQTT, Bluetooth, Wi-Fi, drivers de periféricos.
- **Uso:** Ideal para desenvolver aplicações de *IoT* que requerem conectividade de rede e controle de hardware.

3.3.2 ThingsBoard

ThingsBoard é uma plataforma *open-source* para gerenciamento de dispositivos *IoT*, coleta e processamento de dados em tempo real. Ele fornece suporte nativo ao protocolo MQTT, permitindo a comunicação eficiente entre dispositivos e servidores utilizando seu próprio broker MQTT integrado.

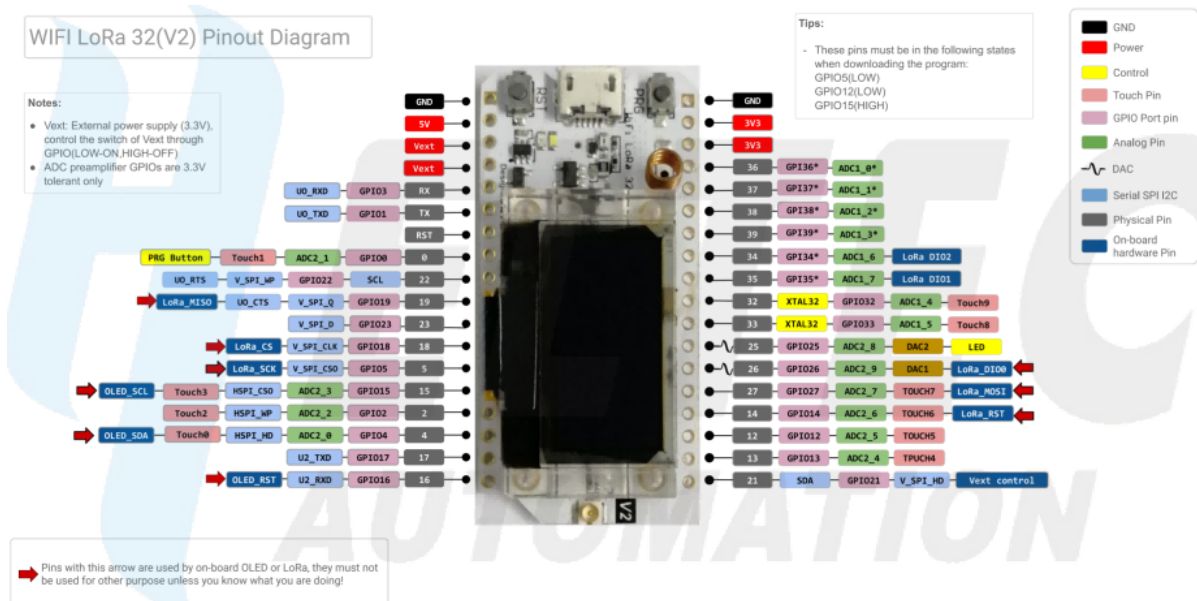
- **Componentes:** Broker MQTT embutido, regras de automação, dashboards interativos, suporte a múltiplos protocolos (*MQTT*, *CoAP*, *HTTP*).
- **Uso:** Ideal para gerenciamento remoto de dispositivos IoT e monitoramento de dados em tempo real.

3.4 Componentes de Hardware

3.4.1 ESP Wi-Fi LoRa 32

O *ESP Wi-Fi LoRa 32* é um microcontrolador de baixo custo e alta performance desenvolvido pela Heltec. Ele combina conectividade *Wi-Fi* e *Bluetooth* integradas com suporte à tecnologia LoRa, tornando-o ideal para aplicações de longo alcance em redes de baixa potência. Além disso, possui dois núcleos de processamento e uma variedade de interfaces de comunicação, como SPI, I2C, UART, dentre outras. O *Wi-Fi LoRa 32* é amplamente utilizado em aplicações de Internet das Coisas (*IoT*), automação residencial, sistemas embarcados, e projetos que exigem conectividade sem fio e comunicação de longo alcance.

Figura 1 – Pinout ESP Wi-Fi LoRa 32



Autor: (AUTOMATION, 2024)

3.4.1.1 Especificações

- **CPU:** Processador dual-core Xtensa® 32-bit LX7 com frequência de até 240 MHz.
- **Memória:** 384 KB de ROM, 512 KB de SRAM e 8 MB de Flash.

- **Conectividade:**
 - Wi-Fi 802.11 b/g/n com velocidade de até 150 Mbps.
 - Bluetooth 5 (LE) com suporte a malha Bluetooth.
 - LoRa com chip SX1262, operando nas frequências de 470–510 MHz e 863–928 MHz, com potência máxima de transmissão de 21 ± 1 dBm e sensibilidade de recepção de até -139 dBm.
- **Interfaces de comunicação:** 7 canais ADC1, 2 canais ADC2, 7 pinos touch, 3 UARTs, 2 I2C, 2 SPI, entre outros.
- **Display:** OLED de 0,96 polegadas com resolução de 128x64 pixels.
- **Alimentação:**
 - Via USB Tipo C: 5V DC (≥ 500 mA).
 - Bateria de lítio: 3,7V a 4,2V (≥ 250 mA), com sistema integrado de gerenciamento de carga e descarga.
- **Consumo de energia:** Baixo consumo em modo *deep sleep*, ideal para dispositivos alimentados por bateria.
- **Dimensões:** 50,2 mm x 25,5 mm x 10,2 mm.
- **Temperatura de operação:** -20°C a 70°C.

3.4.2 Power Bank Peining 10000mAh WUP-653

A *Power Bank Peining WUP-653* é um dispositivo de armazenamento de energia portátil, projetado para fornecer alimentação confiável para diversos dispositivos eletrônicos. Com capacidade de 10.000mAh e potência de saída de 22.5W, é ideal para carregar smartphones, tablets e outros equipamentos portáteis de forma rápida e eficiente.

Figura 2 – Power Bank Peining 10000mAh WUP-653



Fonte: Shopee (2024)

3.4.2.1 Especificações

- **Capacidade:** 10.000mAh.
- **Potência de saída:** 22.5W.
- **Entradas:** Lightning (5V/2A, 9V/2A, 12V/1.5A), Tipo C (5V/2.6A, 9V/2A, 12V/1.5A).
- **Saídas:** USB (4.5V/5A, 5V/3A, 9V/2A, 12V/1.5A), Tipo C (5V/3A, 9V/2.22A, 12V/1.67A).
- **Recursos:** Cabo embutido, compatível com 99

- **Garantia:** 3 meses pelo fornecedor.
- **Compatibilidade:** Compatível com smartphones, tablets e outros dispositivos portáteis.

3.4.3 Modos de Energia e Consumo

A tabela abaixo apresenta os diferentes modos de energia de um dispositivo, juntamente com suas descrições e consumo de energia correspondente.

Tabela 1 – Modos de energia e seus respectivos consumos.

Modo de Energia	Descrição	Consumo de Energia
Ativo (RF funcionando)	Wi-Fi Tx packet 14 dBm \sim 19.5 dBm Wi-Fi / BT Tx packet 0 dBm Wi-Fi / BT Rx e escuta	160mA \sim 260mA
Modem-sleep	CPU ligada Velocidade máxima 240 MHz Velocidade normal 80 MHz Velocidade reduzida 2 MHz	30 mA \sim 50 mA 20 mA \sim 25 mA 2 mA \sim 4 mA
Light-sleep	-	0.8 mA
Deep-sleep	ULP co-processador ligado Padrão monitorado por sensor ULP Timer RTC + Memória RTC	150 μ A 100 μ A @ 1% duty 10 μ A
Hibernação	Apenas Timer RTC ativo	5 μ A
Desligado	CHIP_PU em nível baixo, chip desligado	0.1 μ A

Fonte: ([ENGINEERS, 2025](#))

3.4.3.1 Descrição dos Modos de Energia

- **Ativo (RF funcionando):** O dispositivo está transmitindo ou recebendo pacotes de Wi-Fi/Bluetooth, consumindo mais energia.
- **Modem-sleep:** O processador está ligado, mas o rádio pode estar desativado para economia de energia.
- **Light-sleep:** O processador pode entrar em um estado de baixo consumo com tempo de resposta reduzido.
- **Deep-sleep:** O co-processador ULP pode monitorar sensores enquanto a CPU principal está desligada.

- **Hibernação:** Apenas o temporizador RTC está ativo para possíveis acionamentos futuros.
- **Desligado:** O chip está completamente desligado, consumindo energia mínima.

3.4.4 Sensor de Umidade DFROBOT SEN0193

O **Sensor de Umidade do Solo SEN0193** é um dispositivo eletrônico projetado para medir a umidade do solo utilizando a técnica de detecção capacitiva. Fabricado com materiais que resistem à corrosão, este sensor oferece uma durabilidade superior e uma vida útil prolongada. O SEN0193 possui três pinos: um para conexão ao terminal de ground (terra), outro para o VCC (tensão de alimentação) e um terceiro para a saída analógica. A saída analógica emite uma tensão que varia inversamente com o nível de umidade do solo. Assim, quanto maior a quantidade de água presente no solo, menor será a tensão de saída fornecida pelo sensor.

3.4.4.1 Especificações

- Tensão de operação: 3.3V a 5.5 VDC;
- Tensão de saída: 0 a 3 VDC
- Corrente de trabalho: 5mA;
- Interface: PH2.0-3P;
- Dimensões: 98 x 23 mm (CxL);
- Peso: 15g.

Figura 3 – Sensor de Umidade SEN0193



Autor: (DFROBOT, 2024)

4 Proposta de Solução

Este capítulo tem como objetivo apresentar a proposta de solução que visa resolver o problema identificado na Seção 1.1. Serão demonstrados, passo a passo, os procedimentos adotados ao longo do processo, culminando na conclusão final. Este capítulo está organizado nas seguintes seções:

- **Requisitos:** levantamento e priorização de requisitos para garantir que o software atenda às exigências do usuário;
- **Arquitetura:** Criação do modelo da arquitetura;

4.1 Requisitos do Sistema

Com base nos requisitos identificados e priorizados utilizando a técnica MoSCoW, foi desenvolvido um formulário para coletar a opinião do professor coorientador, João. Esse formulário apresentou os requisitos listados em conjunto com uma escala linear para que ele pudesse selecionar a priorização de cada requisito de acordo com sua importância.

A escala linear utilizada no formulário permitiu ao professor coorientador indicar de forma clara e objetiva a importância de cada requisito, ajudando a refinar ainda mais a priorização. Essa etapa adicional de validação é importante para garantir que as decisões tomadas estejam alinhadas com as expectativas e necessidades acadêmicas e de pesquisa envolvidas no projeto.

Com base nesse método e nos requisitos previamente identificados para a Proposta de Solução, a Tabela 2 foi elaborada para apresentar a priorização dos requisitos.

Autor: Autoria Própria (2024)

Tabela 2 – Priorização dos Requisitos.

ID	Requisito	Prioridade
RF001	Comunicar os microcontroladores com uso de LoRa	Must have
RF002	Implementar o registro da umidade do solo contínuos	Could have
RF003	Implementar múltiplos registros de intervalados	Could have
RF004	Realizar a remoção de outliers (valores atípicos)	Could have
RF005	Configurar a geração de curvas de registro da umidade do solo	Must have
RF006	Criar histogramas que auxiliem na visualização da umidade ao longo do tempo	Must have
RF007	Criar mecanismo de acionamento temporário	Could have
RF008	Configurar a mecanismo para persistência de dados	Must have
RF009	Construir o Gateway para o servidor em uma das placas utilizando MQTT	Must have

4.2 Arquitetura do Sistema

A Figura 4 apresenta a arquitetura de um sistema de monitoramento de umidade utilizando sensores SEN0193, o protocolo de comunicação LoRa, o protocolo MQTT para envio de mensagem para o Thingsboard e o Thingsboard para visualização dos dados.

4.2.1 Sensor de Umidade

Descrição: Dispositivo responsável por medir a umidade do solo.

Função: Coletar dados de umidade e enviar para o ESP Wi-Fi LoRa 32 (Nó Sensor).

Interface: Conexão direta com o ESP Wi-Fi LoRa 32. Porta ADC 1 com 12 bits de resolução.

4.2.2 ESP32 (Nó Sensor)

Descrição: Microcontrolador que coleta dados do Sensor de Umidade e envia via LORA para a ESP Gateway.

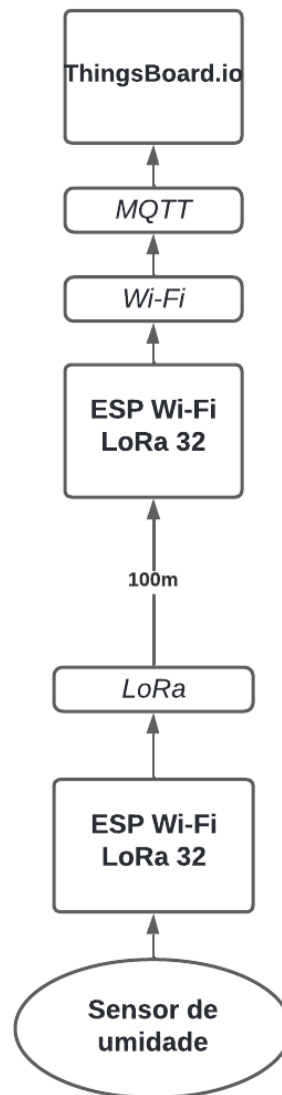
Função:

- Leitura dos dados do Sensor de Umidade.
- Envio dos dados via LORA para o servidor central.

4.2.3 LoRa

Descrição: LoRa (*Long Range*) é uma tecnologia de comunicação sem fio que utiliza modulação de espectro expandido para permitir a transmissão de dados a longas

Figura 4 – Arquitetura do Sistema



Autor: Autoria Própria (2024)

distâncias com baixo consumo de energia. Desenvolvido para aplicações de *Internet das Coisas* (IoT), o LoRa é ideal para ambientes rurais ou locais remotos, onde a infraestrutura de rede tradicional, como Wi-Fi ou redes celulares, pode ser inexistente ou limitada. O protocolo opera em bandas de frequência livre (ISM), permitindo a comunicação eficiente entre sensores distribuídos e um nó central (gateway), que recebe e encaminha os dados para um servidor ou plataforma de monitoramento.

Função: Utilizaremos LoRa para estabelecer uma rede de comunicação eficiente entre os sensores e o gateway, garantindo a transmissão confiável dos dados de umidade do solo mesmo em cenários de difícil conectividade. Esta tecnologia oferece alta resistência a interferências e é particularmente útil para o monitoramento agrícola, pois possibilita a

cobertura de grandes áreas sem necessidade de infraestrutura complexa.

- **Topologia da Rede:** A rede pode ser configurada no modelo estrela ou ponto-a-ponto. No experimento, os nós sensores enviarão os dados diretamente para o gateway LoRa, que será responsável pelo encaminhamento das informações para a plataforma de visualização.
- **Alcance e Consumo de Energia:** Dependendo das condições do ambiente, o LoRa pode alcançar distâncias superiores a 10 km em áreas rurais, mantendo um consumo energético extremamente baixo, ideal para dispositivos alimentados por baterias.

4.2.4 WI-FI

Descrição: Wi-Fi (Wireless Fidelity) é uma tecnologia que permite a conexão sem fio de dispositivos em uma rede de dados. No contexto deste sistema, o Wi-Fi é utilizado pelos dispositivos ESP32 para se conectar à internet e transmitir dados para a plataforma ThingsBoard.io via MQTT. Essa conexão permite que os dados coletados pelos sensores sejam acessados em tempo real, de qualquer lugar com acesso à internet.

Função: Fornecer conectividade sem fio para os dispositivos ESP32, permitindo que eles se comuniquem com servidores remotos, como o ThingsBoard.io, através do protocolo MQTT.

4.2.5 MQTT

Descrição: MQTT (MQ Telemetry Transport) é um protocolo de comunicação leve e publish/subscribe amplamente utilizado em sistemas *IoT*. Ele é projetado para transmitir mensagens entre dispositivos com baixa largura de banda e alta latência, sendo ideal para ambientes com recursos limitados e conectividade intermitente.

Função: Utilizaremos o MQTT para facilitar a comunicação entre os dispositivos ESP32 e um servidor central, permitindo a troca de mensagens de forma eficiente e confiável em uma rede distribuída. O MQTT será responsável por coordenar a transmissão de dados dos sensores e atuadores na rede IoT.

4.2.6 ESP32 (Gateway)

Descrição: Microcontrolador que atuará como central de coleta dos dados enviados pelos nós sensores via LORA e irá enviar esses dados para a Thingsboard.io via MQTT.

Função:

- Receber dados dos nós sensores.
- Enviar dados para o Thingsboard.io.

4.2.7 ThingsBoard

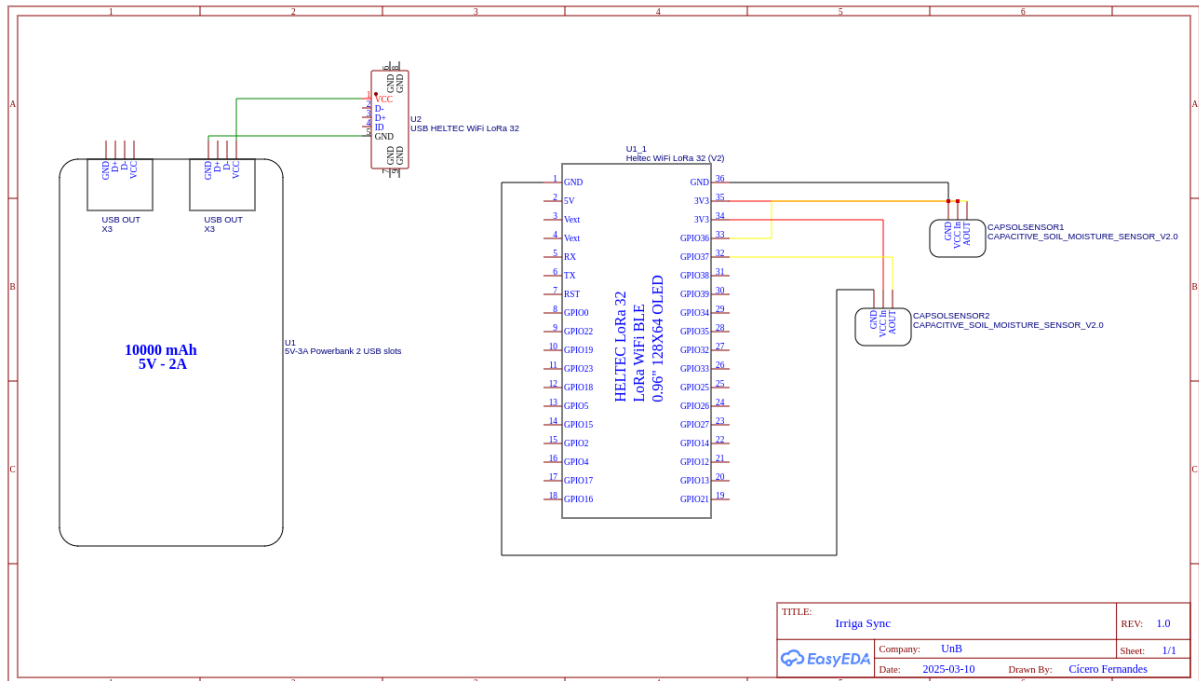
Descrição: Plataforma de código aberto para *IoT*, que permite o gerenciamento de dispositivos, coleta e processamento de dados em tempo real, e visualização através de dashboards customizáveis.

Função: Monitorar e analisar dados de dispositivos conectados, incluindo o processamento de dados de umidade em tempo real e a geração de alertas ou ações automáticas com base em condições predefinidas.

4.3 Esquemático do Sistema

Abaixo, na Figura 5, segue esquemático do sistema representando como as ligações foram realizadas, onde os sensores foram ligados nos pinos 36 e 37 do microcontrolador, que, no WiFi LoRa 32 V2 da Heltec, são pinos analógicos de porta ADC 1.

Figura 5 – Esquemático IrrigaSync



Autor: Autoria Própria (2025)

4.4 Fluxo de Dados

Coleta de Dados:

- O Sensor de Umidade coleta os dados do solo.
- O Sensor de Umidade envia os dados para o ESP32 relacionado a ele.

Transmissão de Dados:

- O ESP32 processa os dados e os transmite via LORA para a ESP32 gateway, assim, a ESP32 que servirá de gateway, transmite os dados via MQTT para o Thingsboard.

4.5 Cronograma do Projeto

A Tabela 3 detalha a distribuição dos requisitos do *Product Backlog* ao longo do desenvolvimento do sistema, bem como as tarefas associadas a cada uma delas.

Tabela 3 – Mapeamento do Cronograma e Requisitos Concluídos (RC) no Desenvolvimento do Sistema.

Período	Requisito(s)	Descrição	RC
05/01/2025 - 12/01/2025	RF001	Implementar comunicação entre microcontroladores usando LORA	Sim
05/01/2025 - 12/01/2025	RF002	Implementar o registro contínuo da umidade do solo	Sim
13/01/2025 - 19/01/2025	RF009	Construir o Gateway para o servidor em uma das placas utilizando MQTT	Sim
13/01/2025 - 19/01/2025	RF003	Implementar múltiplos registros em intervalos programáveis	Sim
20/01/2025 - 26/01/2025	RF008	Configurar o mecanismo para persistência de dados	Sim
20/01/2025 - 26/01/2025	RF006	Desenvolver histogramas para visualização da umidade ao longo do tempo	Sim
27/01/2025 - 02/02/2025	RF004	Configurar o sistema de remoção de outliers (valores atípicos)	Sim
27/01/2025 - 02/02/2025	RF005	Configurar a geração de curvas de registro da umidade do solo	Sim

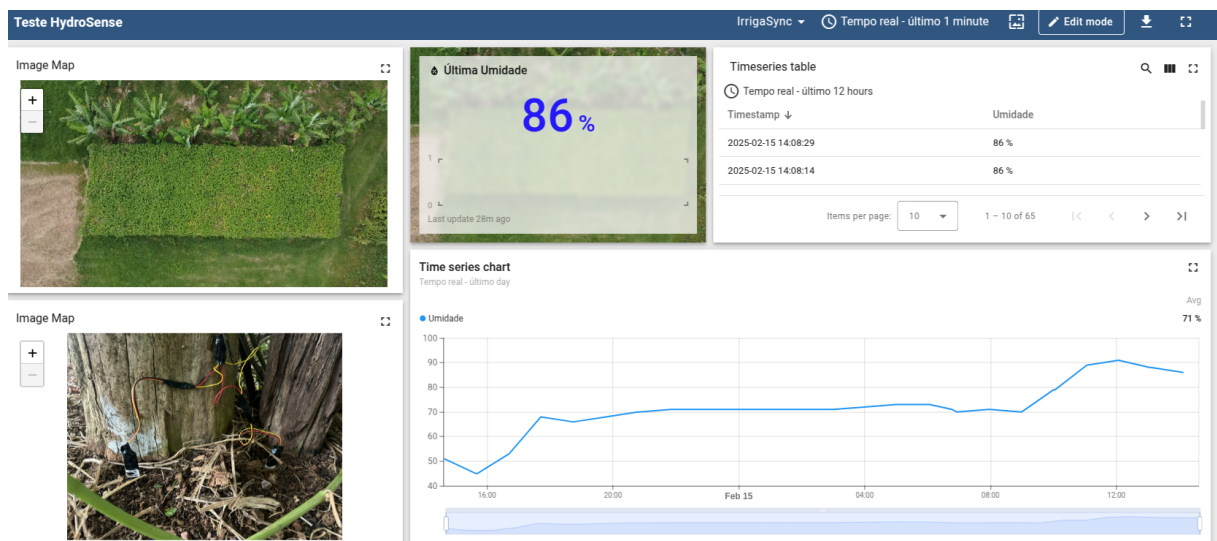
Fonte: Autoria Própria (2025)

5 Testes e Resultados

5.1 Funcionamento do Thingsboard

Para a solução do projeto, foi utilizado a plataforma IoT Thingsboard para controle de dados e visão da solução. Foi construído um protótipo inicial de como os dados ficariam organizados no sistema, com alarmes sobre o estado de cada dispositivo.

Figura 6 – Protótipo Dashboard



Autor: Autoria Própria (2024)

5.1.1 Localização do Experimento

Para a realização do experimento, foi autorizada a coleta de dados de umidade do solo na horta de chuchu da Fazenda Água Limpa. Para auxiliar no mapeamento da área, foi utilizado um drone, que registrou imagens aéreas do local, conforme ilustrado na Figura 7.

Figura 7 – Horta de Chuchu



Autor: Autoria Própria (2025)

5.2 Conexão com LORA

As ESPs de nó sensor e a ESP central irão se comunicar utilizando a tecnologia LoRa por topologia Peer to peer, que opera na camada física da comunicação sem fio. O protocolo LoRa permite a transmissão de dados a longas distâncias por meio de modulação espectral expandida, sendo robusto contra interferências. A comunicação pode ocorrer ponto a ponto ou através de uma rede LoRaWAN, que adiciona camadas superiores para gerenciamento de dispositivos e encaminhamento de pacotes.

Na Figura 8 ESP central fica localizada dentro de uma residência, funcionando como um concentrador de dados e recebendo as informações enviadas pela ESP remota, que está instalada na horta de chuchu.

Figura 8 – Esp da Sede



Autor: Autoria Própria (2024)

Na figura 9 ESP da horta é responsável pela coleta de dados da umidade do solo por meio de sensores, transmitindo periodicamente esses valores para a ESP central via LoRa. Dessa forma, é possível monitorar as condições do solo em tempo real, permitindo uma análise contínua e a tomada de decisões para otimizar a irrigação da plantação.

Figura 9 – Esp da horta



Autor: Autoria Própria (2025)

5.2.1 Distância entre as ESPs

Conforme ilustrado na Figura 10, a distância entre as ESPs foi de aproximadamente 90 metros. Mesmo nessa faixa de alcance, a comunicação entre os dispositivos ocorreu de forma eficiente, garantindo a transmissão precisa e confiável dos dados coletados na horta.

Figura 10 – Distância entre a Horta e a Central



Fonte: Google Maps (2025)

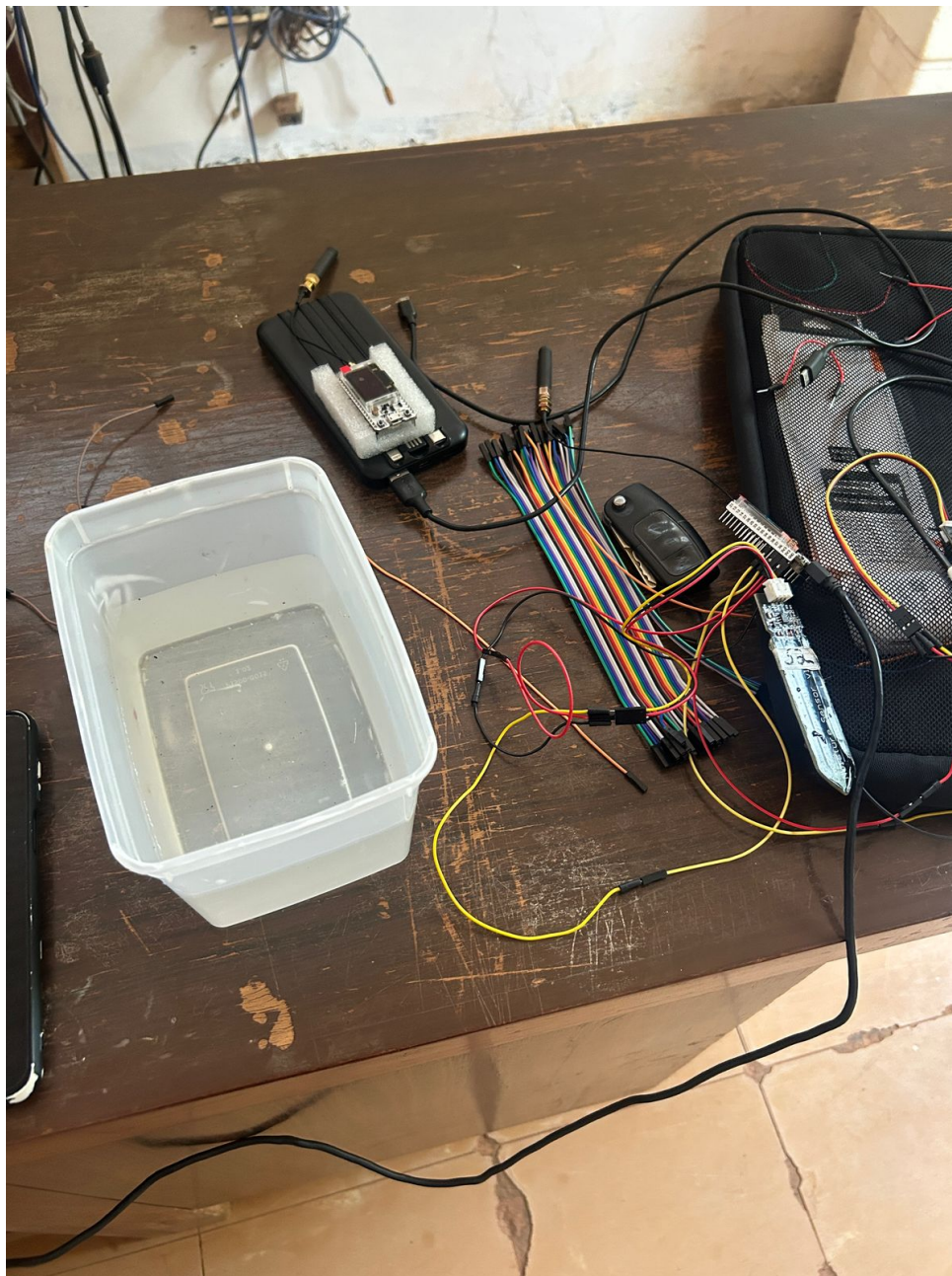
5.3 Procedimento de Calibração

A calibração do sensor SEN0193 é fundamental para garantir medições precisas. O procedimento consiste na obtenção de valores de referência para as condições de solo seco e solo úmido, permitindo o mapeamento da umidade em uma escala percentual.

5.3.1 Determinação do Valor para Solo Seco

Para determinar o valor correspondente ao solo seco, o sensor deve ser posicionado em ambiente livre de umidade, sem contato com o solo ou água. Como pode ser observado na Figura 11.

Figura 11 – Esp no Ar

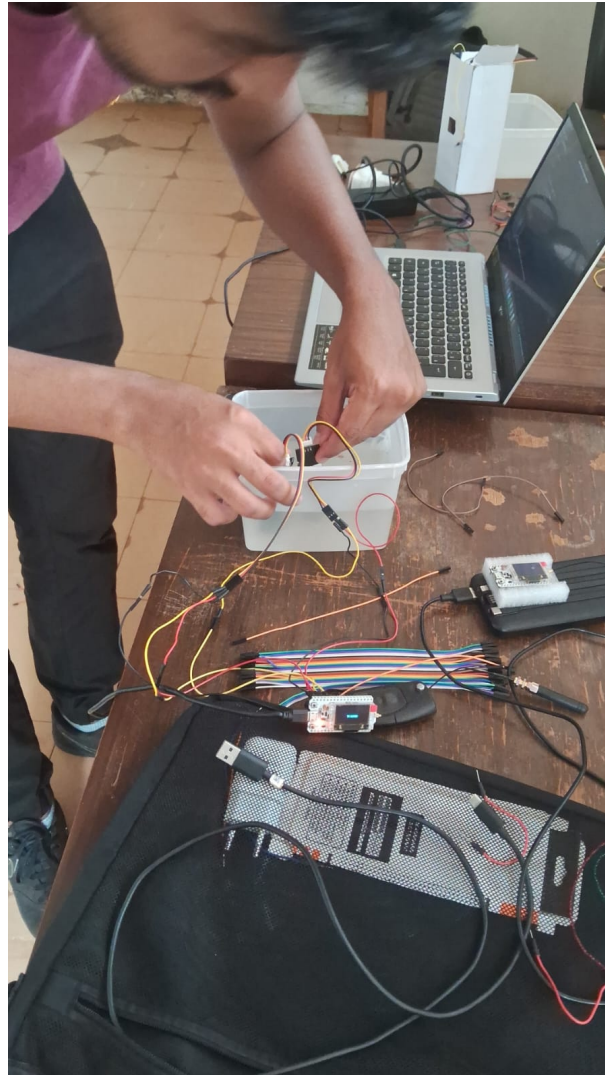


Autor: Autoria Própria (2025)

5.3.2 Determinação do Valor para Solo Úmido

Para a calibração do solo úmido, o sensor deve ser inserido em um solo totalmente saturado de água, como pode ser observado na Figura 12.

Figura 12 – Esp na Água



Autor: Autoria Própria (2025)

5.3.3 Cálculo do Percentual de Umidade

Após a obtenção dos valores de referência, a umidade do solo pode ser calculada pela seguinte equação:

$$U(\%) = \frac{(L - S)}{(U - S)} \times 100 \quad (5.1)$$

onde:

- $U(\%)$ representa a umidade do solo em percentual;
- L é a leitura do sensor em tempo real;
- S é o valor de referência para o solo seco;

- U é o valor de referência para o solo úmido.

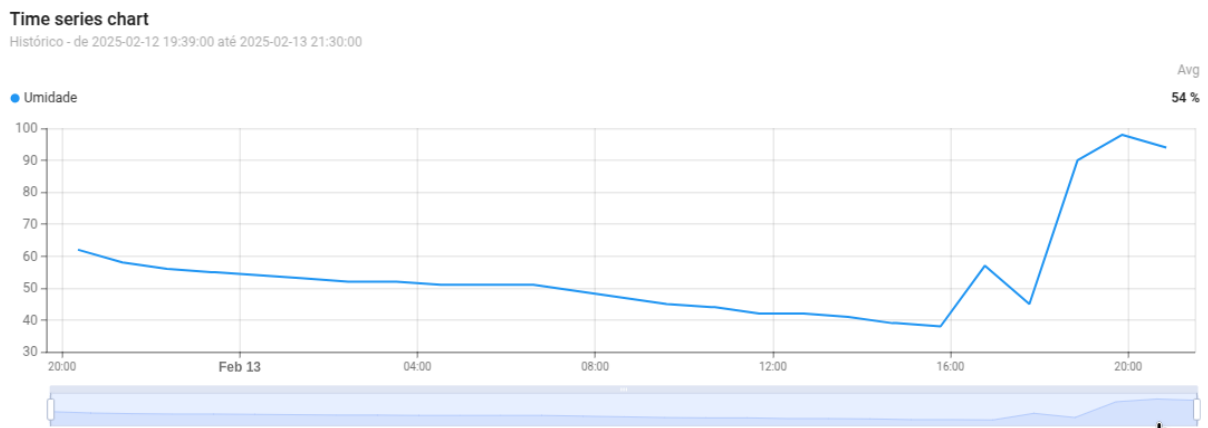
5.4 Experimentos não controlados

Nesta seção, são apresentados os três experimentos realizados em diferentes dias e horários. Cada experimento teve como objetivo corrigir problemas identificados nas etapas anteriores e aprimorar a metodologia aplicada.

5.4.1 1º Experimento

No primeiro experimento, a cada intervalo de 1 hora, foram coletados dados de umidade do solo por meio de dois sensores. A média dessas medições foi calculada para garantir maior precisão na análise e no monitoramento contínuo das condições do solo.

Figura 13 – Gráfico do 1º Experimento



Fonte: Autoria Própria (2025)

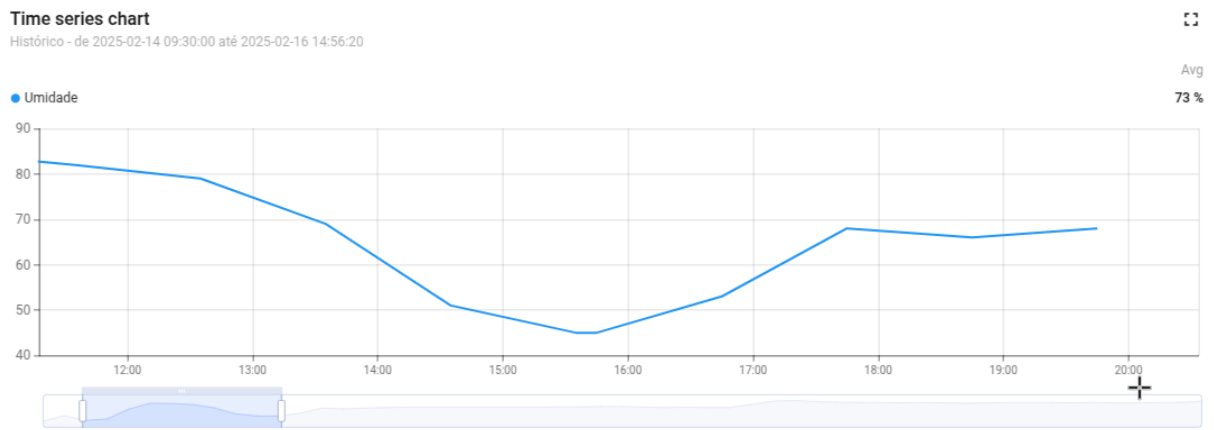
Resultados: Conforme ilustrado na Figura 13, após 20 horas de operação, o sistema apresentou um comportamento inesperado, no qual os valores de umidade do solo aumentaram significativamente sem qualquer interferência externa. Esse fenômeno indicou a necessidade de ajustes no experimento para investigar possíveis causas, como falhas nos sensores ou interferências ambientais. Então, o ambiente construído na horta de chuchu da fazenda foi visitado e os sensores foram testados novamente, evidenciando uma diferença muito grande entre a calibração dos dois sensores.

5.4.2 2º Experimento

Para o segundo experimento, o sensor que estava com a maior discrepância de valores foi substituído por um terceiro sensor e novos testes foram realizados para a

coleta de dados deles, fazendo a mesma abordagem de testar os valores retornados quando o sensor está seco e quando o sensor está molhado. Foi implementada uma rotina para que a ESP fosse reiniciada automaticamente a cada intervalo de 1 hora. Durante esse período, os dados de umidade do solo foram coletados por dois sensores, e a média dessas medições foi calculada para proporcionar maior precisão na análise e no monitoramento das condições do solo.

Figura 14 – Gráfico do 2º Experimento



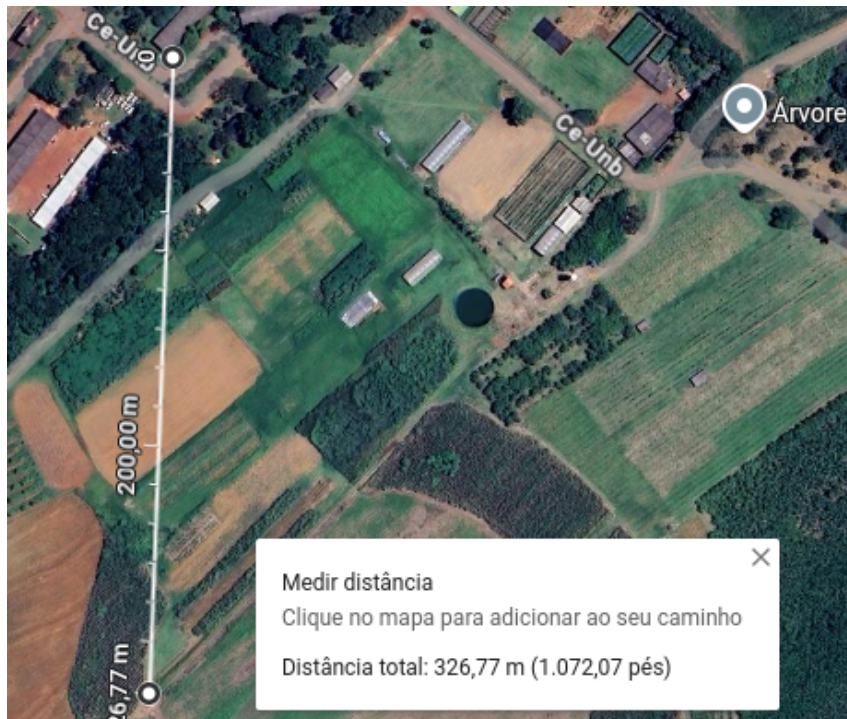
Fonte: Autoria Própria (2025)

Resultados: Como mostrado na Figura 14, após 5 horas de operação, o sistema voltou a apresentar um comportamento inesperado, no qual os valores de umidade aumentaram. Portanto, a fazenda foi visitada mais uma vez para averiguar como estava o estado dos sensores e da solução, ao chegar lá, tudo parecia normal, fizemos testes específicos nos sensores e também na transmissão de dados, provavelmente por conta da chuva ou da irrigação local utilizando as aspersores a umidade subiu.

5.4.3 Distância Máxima entre as ESPs

Conforme ilustrado na Figura 15, a distância entre as ESPs foi de aproximadamente 325 metros. Nessa faixa, a comunicação apresentou instabilidades ocasionais, resultando em falhas na transmissão de dados e redução da persistência da conexão.

Figura 15 – Distância Máxima entre a Horta e a Central



Fonte: Google Maps (2025)

5.5 Experimentos Controlados

Como experimentos em locais não controlados trouxeram mudanças repentinas na umidade que não conseguíamos definir ao certo o motivo, decidimos fazer testes em ambientes controlados para que fosse possível ter mais embasamento no que está acontecendo de fato.

5.5.1 1º Experimento

A solução foi instalada em casa, seguindo a mesma ideia de antes, foram instalados dois sensores e a média entre eles foi calculada. Para este experimento, também, foi implementada uma forma de calibragem dinâmica, onde a ESP reconhece caso algum valor esteja abaixo ou acima dos primeiros limites definidos para os valores se “seco” e “molhado”. Então, caso a média seja menor do que o valor molhado, nós fazemos uma troca no valor referência para molhado e subtraímos a diferença no valor referência de seco, para que os limites acompanhem as mudanças que o sensor vai sofrendo com o passar do tempo. A Figura 16 mostra como ficou a instalação no pote.

Figura 16 – Instalação em casa



Fonte: Autoria Própria (2025)

Resultados parciais: No primeiro dia de coleta, a umidade foi descendo gradativamente, seguindo o esperado, pois começou com uma quantidade alta de água colocada no pote e também foi pouco exposta ao sol, em próximos testes deixaremos o pote com a planta o dia inteiro no sol para ver a reação do sensor.

5.6 Custo Total

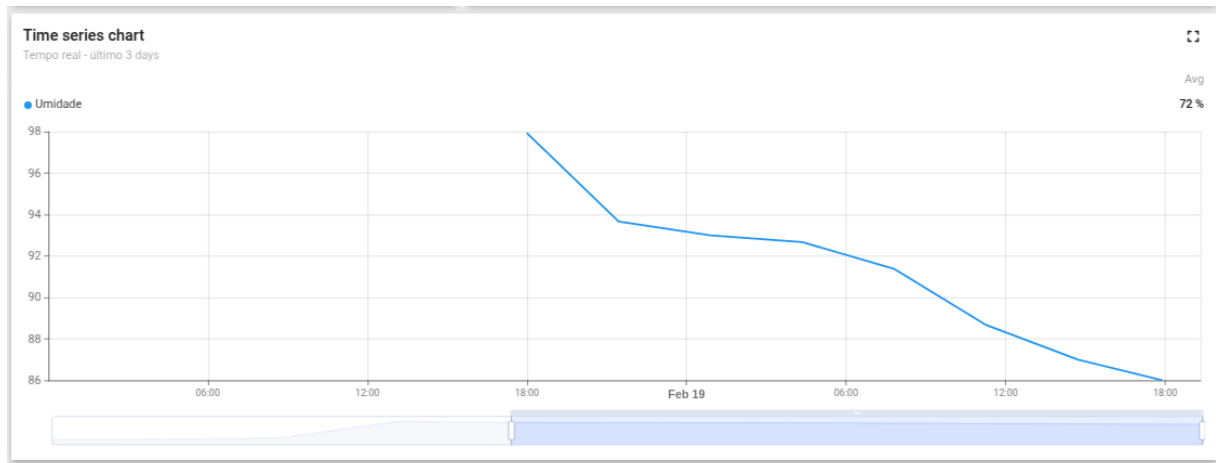
A Tabela 4 apresenta os custos estimados para a realização do experimento, incluindo os componentes utilizados e seus respectivos valores.

Tabela 4 – Custo total do experimento

Item	Quantidade	Custo Por Unidade(R\$)
ESP32 Wi-Fi LoRa 32	2	120,00
Sensor de Umidade SEN0193	3	50,00
Power Bank 10.000mAh	1	100,00
Cabos e Conectores	1	30,00
Materiais diversos (suporte, fixadores)	-	15,00
Total	-	2135,00

Fonte: Autoria Própria (2025)

Figura 17 – Resultados experimento controlado



Fonte: Autoria Própria (2025)

5.7 Possíveis Melhorias

Apesar do funcionamento satisfatório do sistema atual, existem diversas oportunidades de melhoria que podem aprimorar sua escalabilidade, eficiência energética e precisão dos dados coletados.

5.7.1 Escalabilidade do Sistema

Atualmente, o sistema conta com apenas uma ESP LoRa Wi-Fi no campo enviando dados para a ESP na sede. No entanto, para ampliar a cobertura da coleta de dados e aumentar a confiabilidade do monitoramento, pretendemos escalar a solução adicionando mais módulos ESP distribuídos estrategicamente pelo campo. Essa abordagem permitirá uma coleta mais abrangente e resiliente, reduzindo a dependência de um único ponto de comunicação e melhorando a tolerância a falhas.

5.7.2 Otimização do Consumo de Energia

Durante o desenvolvimento, identificamos desafios relacionados ao consumo de energia, especialmente devido à impossibilidade de implementar o modo *Deep Sleep*. O principal obstáculo foi o uso de um *power bank*, que desligava a alimentação ao não detectar consumo constante de corrente. Para resolver essa limitação, futuras melhorias podem incluir a adoção de fontes de energia mais apropriadas, como painéis solares com gerenciamento eficiente de carga ou baterias de lítio personalizadas, garantindo uma operação mais eficiente e prolongada no campo.

5.7.3 Mudança de Antena

Durante os testes, atingimos uma distância de 325 metros utilizando uma antena de 5dBi. Com a substituição por uma antena de maior ganho, como a Yagi 868/915MHz 9 12dBi, espera-se um aumento significativo no alcance da comunicação. Esse modelo é amplamente empregado para transmissão de longo alcance em redes LoRa e IoT, proporcionando maior estabilidade e eficiência na recepção do sinal. Além disso, melhorias estruturais e no material da antena podem otimizar ainda mais seu desempenho, reduzindo perdas e ampliando a confiabilidade da comunicação.

5.7.4 Melhoria de Infraestrutura

Atualmente, a ESP utilizada no campo está acondicionada em um recipiente improvisado, como um pote de sorvete, o que pode comprometer sua proteção contra intempéries e danos mecânicos. Para garantir maior durabilidade e confiabilidade do sistema, recomenda-se a substituição por uma caixa de proteção adequada, como:

- Caixa estanque à prova d'água e poeira, garantindo proteção contra condições adversas.
- Material resistente a raios UV para evitar degradação por exposição ao sol.
- Boa ventilação para evitar superaquecimento da ESP e dos demais componentes eletrônicos.
- Conectores vedados para evitar a entrada de umidade e sujeira.

6 Conclusão

Este trabalho teve como objetivo o desenvolvimento e a avaliação do sistema IrrigaSync, uma solução de monitoramento de umidade do solo voltada para a otimização do uso da água na agricultura. Combinando tecnologias de comunicação sem fio, como *LoRa* e *MQTT*, com a plataforma *ThingsBoard*, o sistema foi projetado para fornecer dados em tempo real sobre a umidade do solo, permitindo um gerenciamento mais eficiente da irrigação.

Durante o desenvolvimento e os testes, foi possível validar a funcionalidade do sistema, mas também identificar desafios técnicos e melhorias que podem ser implementadas. Os experimentos demonstraram a viabilidade do uso do *ESP32 LoRa* para a transmissão de dados a longas distâncias, porém evidenciaram limitações na estabilidade da comunicação, especialmente quando os dispositivos estavam a mais de 325 metros de distância. Além disso, foram observados problemas na calibração dos sensores e no consumo de energia, que impactaram a precisão das medições e a autonomia do sistema em campo.

Diante dessas dificuldades, algumas soluções foram propostas. A escalabilidade do sistema pode ser aprimorada com a adição de mais sensores distribuídos estrategicamente pelo campo, aumentando a confiabilidade da coleta de dados e reduzindo a dependência de um único nó de comunicação. O consumo de energia também pode ser otimizado com a adoção de baterias de lítio com gerenciamento de carga ou painéis solares, garantindo maior autonomia para os dispositivos. Além disso, a calibração dinâmica dos sensores surge como uma alternativa promissora para minimizar desvios nas medições ao longo do tempo.

Dessa forma, conclui-se que o IrrigaSync demonstra viabilidade como uma solução para o monitoramento da umidade do solo, possibilitando um uso mais eficiente da água na irrigação. Embora o sistema tenha apresentado limitações durante os testes, as propostas de aprimoramento discutidas ao longo deste trabalho indicam caminhos para aumentar sua robustez e escalabilidade. A implementação dessas melhorias poderá contribuir para a otimização da gestão hídrica na agricultura, tornando a solução mais confiável e adaptável a diferentes cenários de aplicação.

7 Anexos

7.1 Código

O Código foi implementado em ArduinoIDE em dois arquivos, o sender e o receiver. Basicamente o sender colhia os dados dos dois sensores fazendo a média entre eles e enviava por LoRa, e o receiver os recebia e fazia a conexão com o WiFi e o MQTT para poder enviar os dados para a Thingsboard. Link para o repositório com o código fonte: <https://github.com/ciceroff/hydro-sense/>

No arquivo sender.ino:

```

1  #include "HT_SSD1306Wire.h"
2  #include "LoRaWan_APP.h"
3  #include "Arduino.h"
4  #include "esp_sleep.h"
5
6  #define RF_FREQUENCY 915000000 // Hz
7  #define TX_OUTPUT_POWER 5 // dBm
8  #define LORA_BANDWIDTH 0 // 125 kHz
9  #define LORA_SPREADING_FACTOR 12
10 #define LORA_CODINGRATE 1
11 #define LORA_PREAMBLE_LENGTH 8
12 #define LORA_SYMBOL_TIMEOUT 0
13 #define LORA_FIX_LENGTH_PAYLOAD_ON false
14 #define LORA_IQ_INVERSION_ON false
15 #define AOUT_PIN 36
16 #define AOUT_PIN_2 37
17 #define RX_TIMEOUT_VALUE 1000
18 #define BUFFER_SIZE 30
19
20 char txpacket[BUFFER_SIZE];
21 bool lora_idle = true;
22 int dry = 3100;
23 int wet = 1400;
24 static SSD1306Wire display(0x3c, 500000, SDA_OLED, SCL_OLED,
    GEOMETRY_128_64, RST_OLED);
25 static RadioEvents_t RadioEvents;
26
27 #define SLEEP_DURATION (60 * 1000000ULL)
28
29 void OnTxDone(void);
30 void OnTxTimeout(void);
31

```

```
32 #define LORA_RST 9
33
34 void showMessage(String message, int delayMs = 2000) {
35     display.clear();
36     display.setTextAlignment(TEXT_ALIGN_CENTER);
37     display.setFont(ArialMT_Plain_10);
38     display.drawString(64, 32, message);
39     display.display();
40     delay(delayMs);
41 }
42
43 void reset_lora() {
44     showMessage("Resetando LoRa...");
45     pinMode(LORA_RST, OUTPUT);
46     digitalWrite(LORA_RST, LOW);
47     delay(10);
48     digitalWrite(LORA_RST, HIGH);
49     delay(50);
50     showMessage("Reset Conclu do!");
51 }
52
53 void VextON(void) {
54     pinMode(Vext, OUTPUT);
55     digitalWrite(Vext, LOW);
56 }
57
58 void VextOFF(void) {
59     pinMode(Vext, OUTPUT);
60     digitalWrite(Vext, HIGH);
61 }
62
63 void setup() {
64     VextON();
65     delay(100);
66
67     // Inicializa o display OLED
68     display.init();
69     display.clear();
70     display.display();
71     display.setContrast(255);
72     display.setTextAlignment(TEXT_ALIGN_CENTER);
73     display.setFont(ArialMT_Plain_16);
74     showMessage("Inicializando...");
75
76     SPI.begin();
77
78     RadioEvents.TxDone = OnTxDone;
```

```
79     RadioEvents.TxTimeout = OnTxTimeout;
80
81     Radio.Init(&RadioEvents);
82     Radio.SetChannel(RF_FREQUENCY);
83     Radio.SetTxConfig(MODEM_LORA, TX_OUTPUT_POWER, 0, LORA_BANDWIDTH,
84                       LORA_SPREADING_FACTOR, LORA_CODINGRATE,
85                       LORA_PREAMBLE_LENGTH, LORA_FIX_LENGTH_PAYLOAD_ON,
86                       true, 0, 0, LORA_IQ_INVERSION_ON, 3000);
87
88     showMessage("Configura o LoRa OK!");
89     showMessage("LoRa Pronto!");
90     analogSetAttenuation(ADC_11db);
91 }
92
93 void loop() {
94     if (lora_idle) {
95         char sensor1char[10];
96         char sensor2char[10];
97         char mediaChar[10];
98
99         int sensor1 = analogRead(AOUT_PIN);
100        int sensor2 = analogRead(AOUT_PIN_2);
101        sprintf(sensor1char, "%d", sensor1);
102        sprintf(sensor2char, "%d", sensor2);
103        showMessage("Sensor 1");
104
105        showMessage(sensor1char);
106        showMessage("Sensor 2");
107
108        showMessage(sensor2char);
109
110        int media = (sensor1 + sensor2) / 2;
111        if(media > dry){
112            wet = wet + (media - dry);
113            dry = media;
114        }
115        else if(media < wet){
116            dry = dry - (wet - media);
117            wet = media;
118        }
119
120        sprintf(mediaChar, "%d", media);
121        showMessage("Media");
122
123        showMessage(mediaChar);
124        int resultado = map(media, dry, wet, 0, 100);
125        if(resultado < 0)
```

```
126         resultado = 0;
127         if(resultado > 100)
128             resultado = 100;
129         delay(1000);
130         sprintf(txpacket, "%d", resultado);
131
132
133         // Envia o pacote LoRa
134         int i = 0;
135         while(i < 5){
136             showMessage("Enviando...");
137             showMessage(txpacket, 1000);
138
139             Radio.Send((uint8_t *)txpacket, strlen(txpacket));
140             delay(12000);
141             i++;
142         }
143         lora_idle = false;
144     }
145
146     Radio.IrqProcess();
147     showMessage("delay...");
148     // delay(60000); // Para testes apenas, 1min de timing
149     delay(3600000);
150     // showMessage("Reiniciando ESP...");
151     // delay(5000);
152
153     // esp_restart();
154
155 }
156
157 void OnTxDone(void) {
158     lora_idle = true;
159     showMessage("TX DONE!", 2000);
160     // entrarDeepSleep();
161 }
162
163 void OnTxTimeout(void) {
164     lora_idle = true;
165     showMessage("TX ERROR", 2000);
166     // entrarDeepSleep();
167 }
168
169 void entrarDeepSleep() {
170     showMessage("Entrando em Deep Sleep...");
171
172     esp_sleep_enable_timer_wakeup(SLEEP_DURATION);
```

```

173     esp_sleep_enable_touchpad_wakeup();
174     VextOFF();
175
176     esp_light_sleep_start();
177 }

```

No arquivo receiver.ino:

```

1  #include <WiFi.h>
2  #include <WiFiAP.h>
3  #include <WiFiClient.h>
4  #include <WiFiGeneric.h>
5  #include <WiFiMulti.h>
6  #include <WiFiSTA.h>
7  #include <WiFiScan.h>
8  #include <WiFiServer.h>
9  #include <WiFiType.h>
10 #include <WiFiUdp.h>
11
12 #include <PubSubClient.h>
13 #include "HT_SSD1306Wire.h"
14 #include "LoRaWan_APP.h"
15 #include "Arduino.h"
16
17 #define RF_FREQUENCY 915000000 // Hz
18 #define LORA_BANDWIDTH 0 // 125 kHz
19 #define LORA_SPREADING_FACTOR 12
20 #define LORA_CODINGRATE 1
21 #define LORA_PREAMBLE_LENGTH 8
22 #define LORA_SYMBOL_TIMEOUT 0
23 #define LORA_FIX_LENGTH_PAYLOAD_ON false
24 #define LORA_IQ_INVERSION_ON false
25
26 #define RX_TIMEOUT_VALUE 1000
27 #define BUFFER_SIZE 30
28
29 const char* ssid = "ssid";
30 const char* password = "password";
31
32 // Configura o do Broker MQTT
33 const char* mqtt_server = "tb.fse.lappis.rocks";
34 const char* mqtt_user = "4qNqeNBXOPwPEeyratZv";
35 const char* mqtt_password = "";
36 const char* mqtt_topic = "v1/devices/me/telemetry";
37
38 WiFiClient espClient;
39 PubSubClient client(espClient);
40

```

```
41 char rxpacket[BUFFER_SIZE];
42 bool lora_idle = true;
43 int16_t rssi, rxSize;
44
45 static SSD1306Wire display(0x3c, 500000, SDA_OLED, SCL_OLED,
    GEOMETRY_128_64, RST_OLED);
46 static RadioEvents_t RadioEvents;
47
48 void showMessage(String message, int delayMs = 2000) {
49     display.clear();
50     display.setTextAlignment(TEXT_ALIGN_CENTER);
51     display.setFont(ArialMT_Plain_10);
52     display.drawString(64, 32, message);
53     display.display();
54     delay(delayMs);
55 }
56
57 // Ativa energia do display OLED
58 void VextON(void) {
59     pinMode(Vext, OUTPUT);
60     digitalWrite(Vext, LOW);
61 }
62
63 // Desativa energia do display OLED
64 void VextOFF(void) {
65     pinMode(Vext, OUTPUT);
66     digitalWrite(Vext, HIGH);
67 }
68
69 // Inicializa Wi-Fi
70 void setup_wifi() {
71     Serial.print("Conectando ao Wi-Fi...");
72     WiFi.begin(ssid, password);
73     if (WiFi.status() != WL_CONNECTED) {
74         delay(500);
75         showMessage("Erro ao conectar no wifi...");
76
77         Serial.print(".");
78     }
79     showMessage("Conectou Wifi...");
80     Serial.println("\nWiFi conectado!");
81 }
82
83 // Reconectar ao MQTT se necess rio
84 void reconnect_mqtt() {
85
86     while (!client.connected()) {
```



```
87     showMessage("Conectando ao MQTT...");
88     if (client.connect("ESP32_Client", mqtt_user, mqtt_password)) {
89         showMessage("Conectado ao MQTT!");
90     } else {
91         showMessage("Falha. C digo: ");
92         Serial.println(client.state());
93         delay(5000);
94     }
95 }
96 }
97
98 void mqtt_envia_mensagem(const char* topic, const char* mensagem) {
99     if (!client.connected()) {
100         reconnect_mqtt();
101     }
102     client.publish(topic, mensagem);
103 }
104
105 void setup() {
106     Serial.begin(115200);
107     VextON();
108     delay(100);
109
110     client.setServer(mqtt_server, 1883);
111
112     display.init();
113     display.clear();
114     display.display();
115     display.setContrast(255);
116     display.setTextAlignment(TEXT_ALIGN_CENTER);
117     display.setFont(ArialMT_Plain_16);
118     setup_wifi();
119
120     showMessage("Inicializando RX...");
121
122     showMessage("Inicializando LoRa...");
123     SPI.begin();
124
125     showMessage("Config. Eventos LoRa");
126     RadioEvents.RxDone = OnRxDone;
127     showMessage("Eventos Configurados!");
128
129     showMessage("Radio.Init()");
130     Radio.Init(&RadioEvents);
131     showMessage("Setando Canal");
132     Radio.SetChannel(RF_FREQUENCY);
133 }
```

```

134     showMessage("Setando RX Config");
135     Radio.SetRxConfig(MODEM_LORA, LORA_BANDWIDTH, LORA_SPREADING_FACTOR,
136                       LORA_CODINGRATE, 0, LORA_PREAMBLE_LENGTH,
137                       LORA_SYMBOL_TIMEOUT, LORA_FIX_LENGTH_PAYLOAD_ON,
138                       0, true, 0, 0, LORA_IQ_INVERSION_ON, true);
139     showMessage("RX Pronto!");
140 }
141
142 void loop() {
143     if (WiFi.status() != WL_CONNECTED) {
144         Serial.println("Wi-Fi desconectado! Tentando reconectar...");
145         WiFi.disconnect();
146         WiFi.reconnect();
147         int tentativas = 0;
148
149         while (WiFi.status() != WL_CONNECTED && tentativas < 10) {
150             delay(1000);
151             Serial.print(".");
152             tentativas++;
153         }
154
155         if (WiFi.status() == WL_CONNECTED) {
156             Serial.println("\nWi-Fi reconectado!");
157             showMessage("Wi-Fi reconectado!");
158         } else {
159             Serial.println("\nFalha ao reconectar o Wi-Fi.");
160             showMessage("Falha Wi-Fi");
161         }
162     }
163     if (!client.connected()) {
164         reconnect_mqtt();
165     }
166
167     client.loop();
168
169     if (lora_idle) {
170         lora_idle = false;
171         showMessage("Aguardando RX...");
172         Radio.Rx(0); // Modo contínuo de recepção
173     }
174
175     Radio.IrqProcess();
176 }
177
178 void OnRxDone(uint8_t *payload, uint16_t size, int16_t packet_rssi,
179               int8_t snr) {
180     rxSize = size;

```

```
180     rssi = packet_rssi;
181     memcpy(rxpacket, payload, size);
182     rxpacket[size] = '\0'; // Garante que a string termina corretamente
183
184     Radio.Sleep();
185     lora_idle = true;
186
187     display.clear();
188     display.setTextAlignment(TEXT_ALIGN_CENTER);
189     display.setFont(ArialMT_Plain_10);
190     display.drawString(64, 16, "Pacote RX!");
191     display.drawString(64, 32, rxpacket); // Mostra o conte do recebido
192     display.drawString(64, 48, "RSSI: " + String(rssi)); // Mostra a
        potencia do sinal
193     display.display();
194     delay(3000);
195
196     // **Envia a mensagem via MQTT**
197     char mensagem[100];
198     sprintf(mensagem, "{\"umidade\": %s}", rxpacket);
199     showMessage(mensagem);
200     mqtt_envia_mensagem(mqtt_topic, mensagem);
201
202     // showMessage("Mensagem enviada por mqtt!");
203 }
```

Referências

- AL., B. et. *Microbial population and community dynamics on plant roots and their feedbacks on plant communities*. [S.l.], 2012. Citado na página 18.
- ALEXANDER, M. *Introduction to soil microbiology*. New York, EUA, 1977. Citado na página 18.
- ALVARENGA FAUSTO VEIGA DE; ALVARES, P. H. d. M. *A Cultura do Chuchu*. [S.l.], 2021. Citado na página 20.
- Arduino. *Arduino IDE - Open-source Arduino Software*. 2024. Acesso em: 17 fev. 2025. Disponível em: <<https://www.arduino.cc/en/software>>. Citado na página 27.
- AUTOMATION, H. Wifi lora 32 (v2) specification. 2024. Disponível em: <<https://heltec.org/project/wifi-lora-32v2/>>. Citado na página 28.
- BISCARO, G. A. *Sistema de irrigação localizada*. UFGD, Brasil, 2014. Citado na página 17.
- DFROBOT. Suk:sen0193. 2024. Disponível em: <https://wiki.dfrobot.com/Capacitive_Soil_Moisture_Sensor_SKU_SEN0193/>. Citado na página 33.
- DUCHICELA, J. e. a. *Soil aggregate stability increase is strongly related to fungal community succession along an abandoned agricultural field chronosequence in the Bolivian Altiplano*. Oxford, EUA, 2013. Citado na página 18.
- E.C, B. S. S. A. M. *Manual de irrigacao*. Viçosa, Brasil, 2006. Citado na página 13.
- ENGINEERS, L. M. *ESP32 Sleep Modes Power Consumption*. 2025. Acesso em: 19 fev. 2025. Disponível em: <<https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/>>. Citado na página 31.
- FERRARI, A. T. *Metodologia da Pesquisa Científica*. Rio de Janeiro, Brasil, 1982. Citado na página 25.
- GERHARDT TE; SILVEIRA, D. *Métodos de Pesquisa*. Porto Alegre, Brasil, 2009. Citado na página 26.
- LEVIEN, S. L. A. *Irrigação por Superfície: Sistemas de irrigação por superfície*. ESAM, Brasil, 2003. Citado na página 17.
- MATTERN, F.; FLOERKEMEIER, C. *From the Internet of Computers to the Internet of Things*. Springer,Berlin, 2010. Citado na página 21.
- MINAYO, M. C. d. S. *Pesquisa Social*. Petrópolis, Brasil, 2001. Citado na página 25.
- MOREIRA Fátima M. S.;SIQUEIRA, J. O. *BEVER et al*. Lavras, Brasil, 2006. Citado 2 vezes nas páginas 18 e 19.
- ORGANIZATION, F.; AGRICULTURE. *Food and Agriculture Organization of the United Nations*. Rome,Italy, 2006. Citado 2 vezes nas páginas 13 e 14.

PINHEIRO, J. M. S. Aplicações para redes wireless. 2005. Disponível em: <https://www.projetoederedes.com.br/artigos/artigo_aplicacoes_para_redes_wireless.php>. Citado na página 21.

RAAIJMAKERS, J. e. a. *The rhizosphere: a playground and battlefield for soilborne pathogens and beneficial microorganism*. New York, EUA, 2009. Citado na página 18.

ROESCH, S. M. A. *Projetos de estágio e de pesquisa em administração: guia para estágios, trabalhos de conclusão, dissertações e estudos de caso*. São Paulo, Brasil, 2009. Citado na página 25.

SILVA, A. M. D. *Metodologia da Pesquisa*. Fortaleza, Brasil, 2015. 110 p. Citado na página 24.

TANENBAUM, A. S. *Redes de Computadores. 5ª ed. São Paulo: Pearson Prentice Hal*. Universidade Vrije, Amsterdã, 2021. Citado na página 22.

THINGSBOARD2024. Open-source iot platform. 2024. Disponível em: <<https://thingsboard.io/>>. Citado na página 23.

TRIVIÑOS, A. N. S. *Introdução à Pesquisa em Ciências Sociais: a Pesquisa Qualitativa em Educação – O Positivismo, A Fenomenologia, O Marxismo*. São Paulo, Brasil, 2009. Citado na página 26.

U.R.III, L. S. T. P. A. D. S. *Supressão de plantas daninhas na formação de braquiárias sob três métodos de semeadura*. Goiás, Brasil, 2014. Citado na página 16.