

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS  
GERAIS – *CAMPUS BAMBUÍ*  
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

Deivison Oliveira Costa

**DESENVOLVIMENTO DE UM SISTEMA PARA  
MONITORAMENTO DE IRRIGAÇÃO EM TEMPO REAL**

Bambuí - MG  
2024

DEIVISON OLIVEIRA COSTA

**DESENVOLVIMENTO DE UM SISTEMA PARA  
MONITORAMENTO DE IRRIGAÇÃO EM TEMPO REAL**

Trabalho de conclusão de curso apresentado  
ao Curso de Bacharelado em Engenharia de  
Computação do Instituto Federal de Educa-  
ção, Ciência e Tecnologia de Minas Gerais –  
*Campus Bambuí* como requisito parcial para  
obtenção do grau de Bacharel em Engenharia  
de Computação.

Orientador: Prof. Dr. Mateus Clemente de  
Sousa

Bambuí - MG  
2024

*Dedico este trabalho a todos aqueles que acreditaram em mim. À minha família, pelo apoio e incentivo ao longo desta trajetória. Às minhas amizades, pelas palavras de encorajamento e momentos de descontração que me ajudaram a manter o equilíbrio. A todos os que de alguma forma contribuíram para a minha formação acadêmica e pessoal, o meu mais sincero agradecimento.*

## AGRADECIMENTOS

Primeiramente, agradeço a Deus por guiar meus passos e iluminar meu caminho ao longo desta caminhada acadêmica. Sem sua orientação, nada disso seria possível. Agradeço por cada desafio superado, por cada aprendizado e por cada conquista.

Expresso minha gratidão ao meu orientador Prof. Dr. Mateus, pela orientação produtiva, pelos conselhos e pela paciência dedicada na condução deste trabalho. Suas contribuições foram fundamentais para o meu crescimento acadêmico e profissional.

Agradeço também aos professores e colaboradores desta instituição, que compartilharam seus conhecimentos e experiências, enriquecendo meu percurso acadêmico e contribuindo para o desenvolvimento deste trabalho.

Aos meus pais e familiares, expresso minha profunda gratidão pelo apoio e pela compreensão durante todo este período. Eles foram minha âncora nos momentos difíceis e minha maior fonte de inspiração.

Por fim, dedico um agradecimento especial aos meus amigos e colegas de estudo, pela amizade e pela colaboração mútua. O apoio deles foi essencial.

*“Bem-aventurado o homem que acha sabedoria, e o homem que produz conhecimento.”*  
*(Provérbios 3:13 - Bíblia Sagrada)*

## RESUMO

Este trabalho apresenta o desenvolvimento de um sistema de monitoramento em tempo real para a irrigação agrícola. O sistema integra tecnologias modernas, como Sistemas Cyberfísicos (CPS), aplicações web e protocolos de comunicação (HTTP e MQTT), com o método da Evapotranspiração da Cultura (ETc). A crescente demanda por soluções eficientes na gestão de recursos hídricos e a necessidade de práticas sustentáveis na agricultura motivam esta pesquisa. O sistema desenvolvido utiliza sensores de telemetria para coletar dados ambientais em tempo real. Além disso, algoritmos de cálculo da ETc são integrados ao sistema para fornecer métricas sobre as necessidades hídricas das culturas. A agricultura de precisão, aliada ao conceito de Internet das Coisas (IoT), permite um monitoramento contínuo e detalhado das condições do solo e das plantas, promovendo uma gestão mais eficaz da irrigação. Espera-se que o sistema proposto contribua para a redução do desperdício de água, aumento da produtividade agrícola e promoção da sustentabilidade. Testes e validações em campo visam demonstrar a eficácia do sistema em fornecer informações úteis para os agricultores, facilitando a tomada de decisões.

**Palavras-chave:** Monitoramento em tempo real. Agricultura de precisão. Evapotranspiração da Cultura (ETc). Protocolo MQTT. Internet das Coisas (IoT).

## ABSTRACT

This work presents the development of a real-time monitoring system for agricultural irrigation. The system integrates modern technologies such as Cyber-Physical Systems (CPS), web applications, and communication protocols (HTTP and MQTT) with the Crop Evapotranspiration (ET<sub>c</sub>) method. The increasing demand for efficient water resource management solutions and the need for sustainable agricultural practices motivate this research. The developed system uses telemetry sensors to collect real-time environmental data. Additionally, ET<sub>c</sub> calculation algorithms are integrated into the system to provide metrics on crop water requirements. Precision agriculture, combined with the concept of the Internet of Things (IoT), allows for continuous and detailed monitoring of soil and plant conditions, promoting more effective irrigation management. It is expected that the proposed system will contribute to reducing water waste, increasing agricultural productivity, and promoting sustainability. Field tests and validations aim to demonstrate the system's effectiveness in providing useful information to farmers, facilitating decision-making.

**Keywords:** Real-time monitoring. Precision agriculture. Crop Evapotranspiration (ET<sub>c</sub>). MQTT protocol. Internet of Things (IoT).

## LISTA DE FIGURAS

Figura 1 – "Tsunami" da IoT . . . . .	26
Figura 2 – Ciclo Hidrológico . . . . .	33
Figura 3 – Sensor de temperatura e umidade DHT22 . . . . .	41
Figura 4 – Sensor de pressão barométrica BMP280 . . . . .	42
Figura 5 – Sensor de luminosidade BH1750 . . . . .	43
Figura 6 – Anemômetro RS-FSJT-N01 . . . . .	44
Figura 7 – Comparador de umidade do solo LM393 . . . . .	45
Figura 8 – Módulo GPS NEO-6M V2 . . . . .	46
Figura 9 – Módulo Conversor RS485 para TTL . . . . .	47
Figura 10 – Interface da plataforma EMQX . . . . .	48
Figura 11 – Interface da aplicação Figma . . . . .	55
Figura 12 – Interface do Visual Studio Code . . . . .	55
Figura 13 – Interface do Arduino IDE . . . . .	56
Figura 14 – Interface da aplicação Fritzing . . . . .	57
Figura 15 – Interface da aplicação EasyEDA . . . . .	57
Figura 16 – <i>Layout</i> do protótipo . . . . .	63
Figura 17 – Matriz exportada do EasyEDA . . . . .	63
Figura 18 – Matriz impressa em papel filme . . . . .	64
Figura 19 – PCB finalizada . . . . .	64
Figura 20 – PCB pós corrosão . . . . .	65
Figura 21 – <i>IrrigaSync desktop</i> . . . . .	66
Figura 22 – <i>IrrigaSync mobile</i> . . . . .	67
Figura 23 – Gabinete com PCB instalado . . . . .	68
Figura 24 – Evolução da temperatura em 12 horas (06:00 - 18:00) . . . . .	69
Figura 25 – Evolução da umidade em 12 horas (06:00 - 18:00) . . . . .	69
Figura 26 – Evolução da velocidade do vento em 12 horas (06:00 - 18:00) . . . . .	70
Figura 27 – Evolução da luminosidade em 12 horas (06:00 - 18:00) . . . . .	70
Figura 28 – Evolução das variáveis meteorológicas ao longo de 12 horas (06:00 - 18:00)	71
Figura 29 – Microcultura de café arábica do IFMG - <i>Campus Bambuí</i> . . . . .	72

## LISTA DE TABELAS

Tabela 1 – Evolução do sensoriamento remoto.	19
Tabela 2 – Evolução dos microcontroladores.	23
Tabela 3 – Periféricos do microcontrolador ESP32	24
Tabela 4 – Métodos HTTP	27
Tabela 5 – Características do protocolo MQTT	29
Tabela 6 – Comparaçao das tecnologias	38
Tabela 7 – Especificações técnicas do sensor DHT22	41
Tabela 8 – Especificações técnicas do sensor BMP280	42
Tabela 9 – Especificações técnicas do sensor BH1750	43
Tabela 10 – Especificações técnicas do Anemômetro RS-FSJT-N01	44
Tabela 11 – Especificações técnicas do Comparador LM393	45
Tabela 12 – Especificações técnicas do Módulo GPS NEO-6M V2	46
Tabela 13 – Especificações técnicas do Módulo Conversor RS485 para TTL	47
Tabela 14 – Custos dos componentes utilizados no sistema	47
Tabela 15 – Comparaçao dos dados climáticos em Bambuí em 16/10/2024	71
Tabela 16 – Valores de ETo e ETc coletados pelo sistema <i>IrrigaSync</i>	73

## LISTA DE ABREVIATURAS E SIGLAS

**ADC** – Conversor Analógico-Digital (*Analog-to-Digital Converter*)

**API** – Interface de Programação de Aplicações (*Application Programming Interface*)

**BDA** – Análise de grandes dados (*Big Data Analytics*)

**BLE** – Bluetooth de Baixa Energia (*Bluetooth Low Energy*)

**CERN** – Organização Europeia para a Pesquisa Nuclear (*Conseil Européen pour la Recherche Nucléaire*)

**CPU** – Unidade Central de Processamento (*Central Processing Unit*)

**CPS** – Sistemas Ciberfísicos (*Cyber-Physical Systems*)

**DDD** – *Design* Orientado a Domínio (*Domain-Driven Design*)

**DIP** – Princípio de Inversão de Dependência (*Dependency Inversion Principle*)

**ETc** – Evapotranspiração da Cultura

**ETO** – Evapotranspiração de Referência

**GPIO** – Pinos de Entrada/Saída Gerais (*General-Purpose Input/Output*)

**HTML** – Linguagem de Marcação de Hipertexto (*HyperText Markup Language*)

**HTTP** – Protocolo de Transferência de Hipertexto (*Hypertext Transfer Protocol*)

**IA** – Inteligência Artificial

**IBM** – International Business Machines

**I2C** – Circuito Inter-Integrado (*Inter-Integrated Circuit*)

**IDE** – Ambiente de Desenvolvimento Integrado (*Integrated Development Environment*)

**INMET** – Instituto Nacional de Meteorologia

**IoT** – Internet das Coisas (*Internet of Things*)

**IP** – Protocolo de Internet (*Internet Protocol*)

**ISP** – Princípio de Segregação de Interface (*Interface Segregation Principle*)

**JSON** – Notação de Objetos JavaScript (*JavaScript Object Notation*)

**Kc** – Coeficiente de cultura

**LAI** – Índice de Área Foliar (*Leaf Area Index*)

**LSP** – Princípio de Substituição de Liskov (*Liskov Substitution Principle*)

**MQTT** – Protocolo de Telemetria de Fila de Mensagens (*Message Queuing Telemetry Transport*)

**MVC** – Modelo-Visão-Controle (*Model-View-Controller*)

**OGC** – Consórcio Geoespacial Aberto (*Open Geospatial Consortium*)

**OCP** – Princípio de Aberto/Fechado (*Open/Closed Principle*)

**OPeNDAP** – Projeto de código aberto para um protocolo de acesso a dados de rede (*Open-source Project for a Network Data Access Protocol*)

**ORM** – Mapeador Objeto-Relacional (*Object-Relational Mapping*)

**PCB** – Placa de Circuito Impresso (*Printed Circuit Board*)

**QoS** – Qualidade de Serviço (*Quality of Service*)

**RAM** – Memória de Acesso Aleatório (*Random Access Memory*)

**ROM** – Memória Somente de Leitura (*Read-Only Memory*)

**SCPS** – Sistemas Socio-Ciberfísicos (*Social-Cyber-Physical Systems*)

**SI** – Sistema Internacional de Unidades

**SPI** – Interface Periférica Serial (*Serial Peripheral Interface*)

**SRP** – Princípio de Responsabilidade Única (*Single Responsibility Principle*)

**TCP** – Protocolo de Controle de Transmissão (*Transmission Control Protocol*)

**TLS** – Segurança de Transporte de Camada (*Transport Layer Security*)

**UART** – Transmissor/Receptor Assíncrono Universal (*Universal Asynchronous Receiver-Transmitter*)

**URI** – Identificador Uniforme de Recurso (*Uniform Resource Identifier*)

**USGS** – Serviço Geológico dos Estados Unidos (*United States Geological Survey*)

**WWW** – Rede Mundial de Computadores (*World Wide Web*)

**XML** – Linguagem de Marcação Extensível (*eXtensible Markup Language*)

## LISTA DE SÍMBOLOS

$\Delta$  – Déficit de pressão de vapor

$d_r$  – Distância relativa inversa Terra-Sol

$e_a$  – Pressão de vapor real

$e_s$  – Pressão de vapor saturado

$E$  – Irradiância solar

$ET_c$  – Evapotranspiração da Cultura

$ET_o$  – Evapotranspiração de Referência

$G$  – Fluxo de calor no solo

$G_{sc}$  – Constante solar ( $0.0820 \text{ } MJ \text{ } m^{-2} \text{ } min^{-1}$ )

$h$  – Umidade do solo

$J$  – Dia do ano

$K_c$  – Coeficiente de cultura

$lx$  – Lux

$P$  – Pressão atmosférica

$R^2$  – Coeficiente de determinação

$R_a$  – Radiação extraterrestre

$R_n$  – Radiação líquida

$R_s$  – Radiação solar

$R_{ns}$  – Radiação solar líquida

$R_{nl}$  – Radiação líquida de ondas longas

$R_{so}$  – Radiação solar com céu limpo

$RH$  – Umidade relativa do ar

$T$  – Temperatura

$T_{max,K}$  – Temperatura máxima em Kelvin

$T_{min,K}$  – Temperatura mínima em Kelvin

$u_2$  – Velocidade do vento a 2 metros de altura

$z$  – Elevação em metros do nível do mar

$\delta$  – Declinação solar

$\gamma$  – Constante psicrométrica

$\phi$  – Latitude

$\omega_1$  – Ângulo da hora solar no início do período

$\omega_2$  – Ângulo da hora solar no final do período

$\sigma$  – Constante de Stefan-Boltzmann ( $4.903 \cdot 10^{-9} MJ K^{-4} m^{-2} dia^{-1}$ )

## SUMÁRIO

1	INTRODUÇÃO . . . . .	16
1.1	Contextualização do tema . . . . .	16
1.2	Justificativa . . . . .	16
1.3	Objetivos . . . . .	17
1.4	Estrutura do trabalho . . . . .	17
2	FUNDAMENTAÇÃO TEÓRICA . . . . .	19
2.1	Sensoriamento remoto . . . . .	19
2.1.1	<i>Telemetria</i> . . . . .	20
2.2	Sistemas embarcados . . . . .	21
2.3	Microcontroladores . . . . .	21
2.3.1	<i>ESP32</i> . . . . .	22
2.4	Sistemas cyberfísicos . . . . .	23
2.5	Internet das coisas . . . . .	25
2.6	Protocolo HTTP . . . . .	26
2.7	Protocolo MQTT . . . . .	27
2.8	Agricultura inteligente . . . . .	29
2.9	Agricultura de precisão . . . . .	30
2.10	Gestão da irrigação . . . . .	31
2.10.1	<i>Gestão baseada no consumo de água</i> . . . . .	31
2.11	Evapotranspiração . . . . .	32
2.11.1	<i>Evapotranspiração de referência</i> . . . . .	34
2.11.2	<i>Evapotranspiração de cultura</i> . . . . .	35
2.12	Revisão bibliográfica . . . . .	35
2.12.1	<i>Integração de dados e padrões web para modelos de ETo</i> . . . . .	35
2.12.2	<i>Ferramentas web para determinação de ETo</i> . . . . .	36
2.12.3	<i>Desenvolvimento e aplicação de ferramentas para ETo</i> . . . . .	37
3	MATERIAIS E MÉTODOS . . . . .	39
3.1	Descrição da área de estudo . . . . .	39
3.2	Classificação da pesquisa . . . . .	39
3.2.1	<i>Quanto aos objetivos</i> . . . . .	39
3.2.2	<i>Quanto à natureza</i> . . . . .	39
3.2.3	<i>Quanto aos procedimentos técnicos</i> . . . . .	40
3.2.4	<i>Quanto à abordagem do problema</i> . . . . .	40
3.3	Componentes do Sistema de Monitoramento Meteorológico . . . . .	40
3.3.1	<i>Sensor de Temperatura e Umidade DHT22</i> . . . . .	40
3.3.2	<i>Sensor de Pressão BMP280</i> . . . . .	41

<i>3.3.3</i>	<i>Sensor de Luminosidade BH1750</i>	42
<i>3.3.4</i>	<i>Anemômetro RS-FSJT-N01</i>	43
<i>3.3.5</i>	<i>Comparador de Umidade do Solo LM393</i>	44
<i>3.3.6</i>	<i>Módulo GPS NEO-6M V2</i>	45
<i>3.3.7</i>	<i>Módulo Conversor RS485 para TTL</i>	46
<i>3.3.8</i>	<i>Custos dos Componentes</i>	47
<i>3.4</i>	<i>Integração dos sensores ao ESP32</i>	47
<i>3.5</i>	<i>Ferramentas e Tecnologias Utilizadas</i>	49
<i>3.5.1</i>	<i>Linguagem C</i>	49
<i>3.5.2</i>	<i>Node.js</i>	50
<i>3.5.3</i>	<i>Next.js</i>	51
<i>3.5.4</i>	<i>Jest</i>	53
<i>3.5.5</i>	<i>Shadcn/ui</i>	53
<i>3.5.6</i>	<i>TailwindCSS</i>	54
<i>3.5.7</i>	<i>Figma</i>	54
<i>3.5.8</i>	<i>Visual Studio Code</i>	54
<i>3.5.9</i>	<i>Arduino IDE</i>	54
<i>3.5.10</i>	<i>Fritzing</i>	56
<i>3.5.11</i>	<i>EasyEDA</i>	56
<i>3.5.12</i>	<i>Resumo das escolhas</i>	56
<i>3.6</i>	<i>Estimativa da evapotranspiração</i>	58
<i>3.6.1</i>	<i>Variáveis meteorológicas e sua obtenção</i>	58
<i>3.7</i>	<i>Desenvolvimento da aplicação</i>	60
<i>3.7.1</i>	<i>Padrões arquiteturais e práticas utilizadas</i>	60
<i>3.7.2</i>	<i>Ciclo de desenvolvimento do software</i>	61
<i>3.7.3</i>	<i>Prototipagem e distribuição dos componentes</i>	62
<i>4</i>	<b>RESULTADOS</b>	66
<i>4.1</i>	<i>Interfaces do sistema IrrigaSync</i>	66
<i>4.2</i>	<i>Montagem do hardware</i>	67
<i>4.3</i>	<i>Dados coletados</i>	67
<i>4.3.1</i>	<i>Temperatura e umidade relativa do ar</i>	68
<i>4.3.2</i>	<i>Velocidade do vento</i>	68
<i>4.3.3</i>	<i>Luminosidade</i>	68
<i>4.3.4</i>	<i>Pressão atmosférica</i>	70
<i>4.4</i>	<i>Comparações com dados de referência</i>	71
<i>4.5</i>	<i>Estimativa da evapotranspiração</i>	72
<i>4.6</i>	<i>Discussão dos resultados</i>	73
<i>5</i>	<b>CONCLUSÃO</b>	75

6	TRABALHOS FUTUROS . . . . .	76
	REFERÊNCIAS . . . . .	78

## 1 INTRODUÇÃO

Na presente seção será apresentada uma contextualização do tema, posteriormente a justificativa e objetivos. Por fim, a estrutura do trabalho é mostrada.

### 1.1 Contextualização do tema

A agricultura é uma atividade fundamental para a economia global, sendo responsável pela produção de alimentos e matérias-primas essenciais para a sobrevivência e o bem-estar da humanidade. De acordo Carmody e Turrall (2023), o setor agrícola enfrenta desafios significativos. A crescente demanda por alimentos devido ao aumento populacional e as mudanças climáticas afetam a disponibilidade de recursos naturais. Diante desses desafios, a gestão eficiente dos recursos hídricos e a adoção de tecnologias avançadas tornam-se cruciais para garantir a sustentabilidade e a produtividade das culturas (CARMODY; TURRAL, 2023).

No Brasil, a agricultura desempenha um papel vital na economia, sendo uma das principais fontes de crescimento e desenvolvimento do país (RAMOS *et al.*, 2022). O Brasil é um dos maiores produtores e exportadores de alimentos do mundo, destacando-se na produção de soja, milho, café e carne bovina (RAMOS *et al.*, 2022).

A irrigação é uma prática que tem sido objeto de estudo no país, visando aumentar a produtividade agrícola e contribuir para a segurança alimentar, conforme aponta Ramos *et al.* (2022). Métodos tradicionais de irrigação muitas vezes carecem de precisão e eficiência, levando ao desperdício de água e outros recursos (PEREIRA; OWEIS; ZAIRI, 2002).

A evapotranspiração (ET) é uma das principais variáveis a serem consideradas na gestão eficiente da água (YANG *et al.*, 2023), sendo fundamental para determinar as necessidades hídricas das culturas e otimizar o uso da água na agricultura (CARMODY; TURRAL, 2023; YANG *et al.*, 2023). Esta variável (ET) representa a perda de água do solo por evaporação e pela transpiração das plantas.

O desenvolvimento de sistemas de monitoramento em tempo real para a irrigação surge como uma abordagem promissora para otimizar o uso da água e melhorar o rendimento das culturas (VIJH *et al.*, 2024; DUTTA, L. *et al.*, 2024; IDRISI *et al.*, 2024). De acordo com Prathibha, Hongal e Jyothi (2017), tais sistemas fornecem aos agricultores informações precisas e atualizadas sobre as condições do solo e das plantas, permitindo uma gestão mais eficiente da irrigação ao longo do tempo.

### 1.2 Justificativa

Este trabalho propõe o desenvolvimento de um sistema *web* para monitoramento de variáveis ambientais, com o objetivo de otimizar a gestão da irrigação na agricultura.

Integrando tecnologias de *hardware* e *software*, o sistema coleta e analisa dados climáticos em tempo real, oferecendo informações para a tomada de decisões mais sustentáveis e eficientes.

A adoção de ferramentas tecnológicas para o monitoramento ambiental incentiva a agricultura inteligente (GARG; ALAM, 2023). Essas tecnologias permitem o ajuste dinâmico das atividades agrícolas com base nas variáveis climáticas coletadas. Isso possibilita aos produtores maximizar a produtividade enquanto reduzem os impactos ambientais, especialmente no uso de água (SINGH *et al.*, 2022).

Considerando o alto consumo hídrico da agricultura, são necessárias soluções para otimizar o uso desses recursos (RAMOS *et al.*, 2022). Ferramentas de monitoramento mais precisas podem favorecer uma gestão hídrica eficiente, reduzindo o desperdício e os custos operacionais (CARMODY; TURRAL, 2023).

Assim, a relevância deste estudo está na contribuição para uma agricultura mais prática e adaptável, que responda às diferentes condições climáticas e de solo, promovendo um modelo de produção sustentável.

### **1.3 Objetivos**

Este trabalho tem como objetivo geral desenvolver um sistema de monitoramento em tempo real para o cultivo agrícola. A proposta é utilizar tecnologias modernas, como Sistemas Cyberfísicos (CPS), aplicações web e protocolos de comunicação (HTTP e MQTT), com o método da Evapotranspiração da Cultura (ETc). O objetivo é aumentar a eficácia do monitoramento da cultura e de suas necessidades hídricas.

Os objetivos específicos deste trabalho são os seguintes:

- Implementar um sistema de telemetria de sensores para coletar dados ambientais-chave em tempo real;
- Integrar algoritmos de cálculo da ETc ao sistema para fornecer métricas;
- Desenvolver uma interface de usuário intuitiva para facilitar o acesso e a compreensão dos dados pelos agricultores.

### **1.4 Estrutura do trabalho**

Este trabalho está estruturado da seguinte forma:

- O Capítulo 2 apresenta os conceitos e teorias que servem como base para o desenvolvimento deste trabalho, além de fornecer o estado da arte da área de estudo;
- O Capítulo 3 descreve os procedimentos, técnicas e ferramentas utilizadas para a condução do trabalho;

- O Capítulo 4 expõe os principais achados do estudo, analisando os dados obtidos e destacando suas implicações.
- O Capítulo 5 apresenta a conclusão do trabalho, discutindo os resultados obtidos e as contribuições do estudo.
- Por fim, o Capítulo 6 traz possíveis direções futuras para a continuidade e aprimoramento do sistema proposto.

## 2 FUNDAMENTAÇÃO TEÓRICA

A seguir, são apresentadas perspectivas teóricas relevantes para o contexto do presente trabalho.

### 2.1 Sensoriamento remoto

O sensoriamento remoto é uma área fundamental em vários campos científicos e tecnológicos. Ele pode ser definido como:

"A arte ou ciência de dizer algo sobre um objeto sem tocá-lo." (FISCHER; HEMPHILL; KOVER, 1976, p. 34)

Ao longo dos anos, o sensoriamento remoto tem evoluído, proporcionando uma variedade de aplicações em diversas áreas. Uma visão geral da evolução do sensoriamento remoto pode ser observada na Tabela 1

Tabela 1 – Evolução do sensoriamento remoto.

Ano	Evento
1800	Descoberta do infravermelho por Sir William Herschel.
1839	Início da prática da fotografia.
1847	Especro infravermelho mostrado por A. H. L. Fizeau e J. B. L. Foucault para compartilhar propriedades com a luz visível.
1850-1860	Fotografia a partir de balões.
1873	Teoria da energia eletromagnética desenvolvida por James Clerk Maxwell.
1909	Fotografia a partir de aviões.
1914-1918	Primeira Guerra Mundial: reconhecimento aéreo.
1920-1930	Desenvolvimento e aplicações iniciais da fotografia aérea e fotogrametria.
1929-1939	Depressão econômica gera crises ambientais que levam a aplicações governamentais da fotografia aérea.
1930-1940	Desenvolvimento de radares na Alemanha, Estados Unidos e Reino Unido.
1939-1945	Aplicações da Segunda Guerra Mundial de porções não visíveis do espectro eletromagnético.
1950-1960	Pesquisa e desenvolvimento militares.
1956	Pesquisa de Colwell sobre detecção de doenças de plantas com fotografia infravermelha.
1960-1970	Primeiro uso do termo sensoriamento remoto, satélite meteorológico TIROS.
1972	Lançamento do Landsat 1.
1970-1980	Avanços rápidos no processamento digital de imagens.
1980-1990	Landsat 4: nova geração de sensores Landsat.
1980s	Desenvolvimento de sensores hiperespectrais.
1990s	Sistemas globais de sensoriamento remoto, lidares.
2000s	Lançamento dos satélites Terra e Aqua da NASA para monitoramento climático.
2010s	Lançamento dos satélites Sentinel pelo programa Copernicus da ESA.

Fonte: Retirado de Campbell e Wynne (2011).

Uma das áreas de aplicação do sensoriamento remoto na atualidade é a agricultura. Estudos recentes, como o de Weiss, Jacob e Duveiller (2020), destacam sua importância na adaptação e evolução das práticas agrícolas. Eles fornecem informações repetitivas sobre o estado das culturas ao longo da safra, em diferentes escalas e para diferentes partes interessadas.

As pesquisas ressaltam suas vantagens na monitorização das culturas, estimativa de produtividade e tomada de decisão em diferentes estágios da produção agrícola (SHAN-MUGAPRIYA *et al.*, 2019; KHANAL *et al.*, 2020). Entre esses benefícios, destacam-se a capacidade de monitorar vastas extensões de terra de maneira rápida e eficaz, a obtenção de informações em tempo real e a habilidade de integrar dados oriundos de diversas fontes para análises mais abrangentes.

Além disso, estudos evidenciam seu papel na obtenção de estatísticas agrícolas, contribuindo para a definição de unidades amostrais, estratificação e alocação de amostras (CARFAGNA; GALLEGOS, 2006). Essa contribuição é viabilizada pela integração de análise de imagens de satélite e dados de sensores, permitindo estimar com precisão a área plantada e a produtividade das culturas. Isso fornece subsídios essenciais para a formulação de políticas públicas e embasamento na tomada de decisões estratégicas no âmbito agrícola.

### **2.1.1 Telemetria**

A telemetria é um campo complementar ao sensoriamento remoto e é definida como:

"A indicação, registro ou integração de uma quantidade em um ponto remoto por meios eletrônicos e de tradução métrica. Grandezas de estados elétricos ou mecânicos estão normalmente envolvidas, embora existam outras aplicações diversas para telemedição." (AIEE TELEMETERING SUBCOMMITTEE, 1941, p. 3)

De acordo com Carden, Jedlicka e Henry (2002), ao longo das décadas, os sistemas de telemetria evoluíram gradativamente, impulsionada por avanços tecnológicos em eletrônica, comunicações e processamento de dados. Inicialmente, a telemetria era predominantemente utilizada em aplicações aeroespaciais e militares para monitorar o *status* e o desempenho de aeronaves e foguetes.

Com o passar do tempo, sua aplicabilidade se expandiu para uma variedade de setores, incluindo medicina, indústria automotiva, energia e telecomunicações (TAN *et al.*, 2021; HASSANIEN; DARWISH; ABDELGHAFAR, 2020; DING; KULKARNI; YEKHANIN, 2017).

Para Tan *et al.* (2021), os avanços na miniaturização de sensores e no aumento da capacidade de armazenamento de dados possibilitaram a coleta de uma quantidade

crescente de dados de telemetria. Além disso, melhorias nas técnicas de transmissão de dados têm aprimorado a análise dessas informações.

Ding, Kulkarni e Yekhanin (2017) afirmam que o desenvolvimento de algoritmos de análise de dados e inteligência artificial (IA) tem permitido uma compreensão mais profunda dos padrões e tendências nos dados telemétricos. Isso tem levado a aprimoramentos em diversos processos e sistemas.

Neste trabalho, a telemetria é empregada para monitorar variáveis ambientais, tais como umidade do solo, temperatura e umidade relativa do ar, em uma área de cultivo. Os dados são coletados por sensores remotos e transmitidos através de protocolos de comunicação. Os protocolos de comunicação usadas neste projeto foram o Transporte de Telemetria de Enfileiramento de Mensagens - MQTT (sigla do inglês, *Message Queuing Telemetry Transport*) e o Protocolo de Transferência de Hipertexto - HTTP (sigla do inglês, *Hypertext Transfer Protocol*) para um servidor remoto (*broker*).

## 2.2 Sistemas embarcados

Há várias concepções propostas para sistemas embarcados; um conceito geral pode ser definido como:

"Aquele que possui *hardware* de computador com *software* embutido como um de seus componentes mais importantes. É um sistema baseado em computador dedicado para uma ou várias aplicações. Pode ser um sistema independente ou parte de um sistema maior."(DUTTA, P. et al., 2014, p. 39)

Sua utilização abrange desde dispositivos simples, como controles remotos e eletrodomésticos, até sistemas complexos, como veículos autônomos e equipamentos médicos (LEE, 2008; KATO et al., 2018). Eles desempenham um papel fundamental na automação de processos industriais, na monitoração e controle de dispositivos, na coleta e processamento de dados em tempo real, entre outras funções (DUTTA, P. et al., 2014).

Os sistemas embarcados são compostos por diversos componentes, incluindo microcontroladores, sensores, atuadores, interfaces de comunicação e *software* embarcado (LEE, 2008). Eles são projetados para atender a requisitos específicos de desempenho, consumo de energia, tamanho e custo, dependendo da aplicação (MAZIDI; NAIMI; NAIMI, 2011).

Na seção seguinte, serão abordados os microcontroladores e sua importância na implementação de soluções embarcadas.

## 2.3 Microcontroladores

Segundo Lee (2008), os microcontroladores são componentes essenciais dos sistemas embarcados. Estes desempenham um papel central na execução de tarefas

específicas e no controle de dispositivos em uma variedade de aplicações. Ele é composto por:

"Um microprocessador (CPU), além de uma quantidade fixa de memória principal (RAM), memória de programa (ROM/Flash), interfaces de entrada/saída (I/O), temporizadores e conversores analógico-digitais (ADC), todos em um único chip." (MAZIDI; NAIMI; NAIMI, 2011, p. 40)

A Tabela 2 mostra alguns modelos de microcontroladores e suas finalidades, desde modelos antigos até os mais recentes. Neste trabalho, o principal motivo da escolha do microcontrolador ESP32 se deu em razão de sua capacidade de comunicação *wireless* embutida. Esse recurso também é nativo em outros microcontroladores como Raspberry Pi 4, porém, o ESP32 se destaca por um menor custo e consumo de energia (MAIER; SHARP; VAGAPOV, 2017).

A seguir, serão discutidas as características do ESP32 e sua relevância na implementação de sistemas de monitoramento e controle.

### **2.3.1   ESP32**

O ESP32 é um microcontrolador de baixo custo desenvolvido pela Espressif Systems. De acordo com Maier, Sharp e Vagapov (2017), este componente é amplamente utilizado em projetos de Internet das Coisas - IoT (sigla do inglês, *Internet of Things*) devido às suas capacidades de conexão integradas e poder de processamento.

Baseado em um microprocessador *dual-core* Tensilica Xtensa LX6, o ESP32 opera em frequências de até 240 MHz, apresentando uma arquitetura de 32 bits. Além disso, possui uma variedade de periféricos (Tabela 3) o que o torna versátil para uma variedade de aplicações (ESPRESSIF SYSTEMS, 2024).

Uma das principais vantagens do ESP32, como destacado por Maier, Sharp e Vagapov (2017), é sua capacidade de conexão Wi-Fi e Bluetooth embutida, suportando os padrões 802.11 b/g/n Wi-Fi, Bluetooth 4.2 e Bluetooth Low Energy (BLE). Isso possibilita a comunicação sem fio em ambientes de IoT.

O ESP32 pode ser programado usando o Arduino IDE e o ESP-IDF (Espressif IoT Development Framework), ambos baseados em C/C++. Através dessas plataformas (IDEs), os desenvolvedores podem acessar os recursos do ESP32 e criar aplicações. Esses projetos vão desde simples automações residenciais até dispositivos complexos de monitoramento (FERRÁNDEZ-PASTOR *et al.*, 2018; JUNIOR *et al.*, 2022; HSU *et al.*, 2020).

Neste trabalho, o ESP32 atua como um ponto de controle, permitindo a comunicação bidirecional entre o mundo físico (representado pelas variáveis ambientais) e o mundo digital (representado pelo *broker*). Essa capacidade de interação direta entre

Tabela 2 – Evolução dos microcontroladores.

Modelo	Ano de Lançamento	Preço (USD)	Componentes Embutidos
Intel 8051	1977	\$1 - \$2	CPU de 8 bits, memória RAM e ROM integradas, temporizador, portas de I/O (WHARTON, 1980).
PIC16F84A	1993	\$2 - \$4	CPU de 8 bits, memória flash e RAM integradas, temporizador, portas I/O (CUENCA; BALBOA, 1997).
Arduino Uno	2010	\$3 - \$4	CPU de 8 bits, memória flash e RAM integradas, temporizador, portas I/O digitais, USB (WARREN; ADAMS; MOLLE, 2011).
STM32F4	2011	\$20 - \$25	CPU de 32 bits, memória flash e RAM integradas, temporizador, interfaces SPI, UART, I2C (BROWN, 2012).
ESP32	2016	\$3 - \$4	CPU de 32 bits, <i>Wi-Fi</i> e <i>Bluetooth</i> integrados, memória flash e RAM integradas, temporizador, portas GPIO, SPI, UART, I2C (MAIER; SHARP; VAGAPOV, 2017).
Raspberry Pi 4	2019	\$55 - \$60	CPU de 64 bits, memória RAM integrada, Ethernet, Wi-Fi, portas USB, HDMI, GPIO (MARAGATHAM; BALASUBRAMANIE; VIVEKANANDHAN, 2021).

Fonte: Elaborado pelo autor, 2024.

*hardware, software* e o ambiente físico exemplifica a definição de um Sistema Cyberfísico - CPS (sigla do inglês, *Cyber-Physical System*), como será abordado na próxima subseção.

## 2.4 Sistemas cyberfísicos

Os sistemas CPSS representam uma evolução dos sistemas embarcados. Eles podem ser definidos como:

"Interações de computação com processos físicos. Computadores embarcados

Tabela 3 – Periféricos do microcontrolador ESP32

Abreviação	Significado	Função
GPIO	Entrada/saída de uso geral.	Permite a comunicação de propósito geral com dispositivos externos através de sinais de entrada e saída digitais.
UART	Receptor/transmissor assíncrono universal.	Facilita a comunicação serial assíncrona entre o microcontrolador e outros dispositivos, como sensores e módulos de comunicação.
SPI	Interface periférica serial.	Permite a comunicação serial síncrona de alta velocidade entre o microcontrolador e dispositivos periféricos, como sensores, displays e memórias.
I2C	Circuito Interintegrado.	Oferece uma interface serial de dois fios para comunicação entre vários dispositivos.

Fonte: Adaptado de Espressif Systems (2024).

e redes que monitoram e controlam os processos físicos, geralmente com *loops* de *feedback* onde os processos físicos afetam as computações e vice-versa." (LEE, 2008, p. 363)

Recentemente, tem havido uma tendência crescente no uso de CPSs em aplicações práticas na agricultura. Por exemplo, Ahmad e Pothuganti (2020) demonstraram um sistema de monitoramento inteligente que utiliza abordagens para rastrear o comportamento e as atividades de roedores no campo. O sistema auxilia especialistas em controle de pragas e agricultores na aplicação mais eficiente de métodos de controle tradicionais.

Da mesma forma, Rad *et al.* (2015) propuseram um modelo de arquitetura de CPS para o monitoramento de culturas de batata, resultando na melhoria da produtividade. Rijswijk *et al.* (2021) discutiram o papel dos sistemas socio-cyberfísicos (SCPSs) na transformação digital da agricultura e das áreas rurais. Eles destacaram a necessidade de compreender as relações entre os aspectos sociais e tecnológicos para uma aplicação bem-sucedida.

O presente trabalho se enquadra na categoria de CPS, pois integra componentes físicos, computacionais e de comunicação. Seu papel é monitorar um processo físico (a irrigação) de maneira automatizada e em tempo real. A infraestrutura necessária para conectar dispositivos físicos à internet e entre si é definida como IoT, como será abordado na próxima subseção.

## 2.5 Internet das coisas

A IoT é um campo multidisciplinar que se desenvolveu ao longo do tempo, com contribuições de diversas áreas. Datta (2017) representa esse campo como um "*tsunami*", que começa com a exploração marítima, seguida pelo desenvolvimento da engenharia, passando pela automação, eletricidade, sensoriamento, e culminando na IoT (Figura 1).

Cada uma dessas etapas representa um avanço na integração de tecnologia e comunicação, resultando em um sistema interconectado e inteligente. A exploração marítima marcou o início do uso de tecnologias para navegação e comunicação a longas distâncias. Com a engenharia, foram estabelecidas as bases para a construção de infraestruturas complexas e sistemas mecânicos.

A automação trouxe a capacidade de controlar processos e máquinas de forma eficiente, enquanto os avanços na eletricidade permitiram a disseminação de dispositivos eletrônicos. O sensoriamento adicionou a capacidade de monitorar e coletar dados em tempo real, preparando o terreno para a IoT, onde todos esses elementos se unem para criar um ambiente altamente conectado e responsivo.

A IoT pode ser definida como:

"Um paradigma emergente de objetos cotidianos que, através da interação com o ambiente e com outros dispositivos, coletam e compartilham dados, geralmente utilizando protocolos de comunicação padronizados e interoperáveis, e, em alguns casos, interagindo com usuários ou outros dispositivos." (KORTUEM *et al.*, 2009, p. 44)

Segundo Minerva, Biru e Rotondi (2015), a ideia de IoT foi proposta pela primeira vez por Ashton (1999). Ele argumentou que poderia revolucionar a maneira como interage-se com o mundo físico, permitindo que artefatos se comuniquem entre si e com sistemas de computador.

Ao longo dos anos 2000, avanços em tecnologias de comunicação sem fio, sensores e sistemas embarcados possibilitaram o desenvolvimento prático da IoT, como destacado por Kortuem *et al.* (2009). Na década de 2010, a IoT começou a ganhar notoriedade em uma variedade de setores, incluindo saúde, manufatura, transporte e agricultura. Vários autores discutiram como essas tecnologias abriram caminho para a criação de redes de dispositivos interconectados em diversos campos (KORTUEM *et al.*, 2009; TAN; WANG, 2010; MINERVA; BIRU; ROTONDI, 2015).

Atualmente, a IoT continua a evoluir, com o surgimento de novas tecnologias como a computação em névoa, termo conhecido em inglês como *fog computing*, e integrações com inteligências artificiais (IAs) (JUNIOR *et al.*, 2022). Na agricultura, a IoT é empregada para melhorar a eficiência e produtividade.

O estudo conduzido por Ferrández-Pastor *et al.* (2018) exemplifica o uso da IoT para otimizar os processos agrícolas. Através de sensores interconectados, demonstrou-se ser

possível monitorar uma série de variáveis físicas, como a umidade do solo, a temperatura e a umidade relativa do ar. Essa abordagem mostrou-se promissora, capacitando os agricultores a tomarem decisões mais embasadas em relação ao momento ideal para o plantio, à irrigação e à aplicação de fertilizantes.

Combinada à computação em névoa, a IoT possibilita o processamento local dos dados coletados pelos sensores, o que reduz o tempo de resposta e os custos associados ao envio e recebimento de dados (HSU *et al.*, 2020). Além disso, a integração com a IA permite a análise preditiva e a otimização desses sistemas, fornecendo informações que aumentam a produtividade e diminuem o desperdício.

Para facilitar a comunicação entre dispositivos, servidores e outros sistemas, o presente trabalho utilizou os protocolos HTTP e MQTT. Esses protocolos permitem o acesso remoto e o controle dos dispositivos conectados à IoT, como será abordado nas próximas subseções.

Figura 1 – "Tsunami" da IoT



Fonte: Retirado de Datta (2017).

## 2.6 Protocolo HTTP

De acordo com Fielding *et al.* (1999), o HTTP é um protocolo de comunicação utilizado para sistemas de informação distribuídos e colaborativos. Ele foi concebido para possibilitar a comunicação entre clientes e servidores na Rede Mundial de Computadores - WWW (sigla do inglês, *World Wide Web*).

O HTTP foi introduzido em 1989 por Tim Berners-Lee enquanto ele trabalhava na Organização Europeia para a Pesquisa Nuclear - CERN (sigla do francês, *Conseil*

*Européen pour la Recherche Nucléaire*). A necessidade premente de compartilhar e distribuir informações de forma eficiente em uma rede de computadores motivou o desenvolvimento desse protocolo, que se tornou fundamental para a *web* moderna (MCPHERSON, 2010). Ele pode ser definido como:

"Um protocolo de camada de aplicação que define a estrutura de mensagens trocadas entre clientes e servidores. Ele opera sobre o Protocolo de Controle de Transmissão/Protocolo de Internet (TCP/IP), permitindo, nesta instância, a solicitação e transferência de recursos, como documentos HTML, imagens e vídeos, pela web." (FIELDING *et al.*, 1999, p. 7)

No âmbito do desenvolvimento de sistemas, o HTTP desempenha um papel importante na criação de Interfaces de Programação de Aplicativos - APIs (sigla do inglês, *Application Programming Interfaces*), possibilitando a comunicação entre diferentes componentes. Os desenvolvedores têm a capacidade de criar pontos de acesso HTTP que aceitam solicitações e respondem com dados formatados, como JSON ou XML (NEUMANN; LARANJEIRO; BERNARDINO, 2021).

Na Tabela 4 são destacados os principais métodos de solicitação do HTTP, utilizados para a interação entre clientes e servidores. No cotidiano, esses métodos são usados para a obtenção de informações, o envio de dados e a execução de operações em sistemas distribuídos, como aplicações *web* e APIs.

Contudo, por si só, o HTTP não é adequado para aplicações da IoT, devido à sua natureza orientada a texto e ao alto consumo de largura de banda (YASSEIN *et al.*, 2017). Para esses casos, o protocolo MQTT é mais adequado, como será abordado na próxima subseção.

Tabela 4 – Métodos HTTP

Método	Descrição
GET	Solicita a representação de um recurso específico.
POST	Envia dados para serem processados por um recurso identificado no servidor.
PUT	Substitui todas as representações atuais do recurso de destino com os dados enviados na solicitação.
DELETE	Remove o recurso identificado no URI.
PATCH	Aplica modificações parciais a um recurso.

Fonte: Adaptado de Fielding *et al.* (1999).

## 2.7 Protocolo MQTT

O MQTT foi introduzido pela primeira vez em 1999 por Dr. Andy Stanford-Clark, da IBM, e Arlen Nipper, da Arcom (agora Eurotech) (YASSEIN *et al.*, 2017).

Segundo Stanford-Clark (2004), o MQTT foi inicialmente desenvolvido para o monitoramento de oleodutos e gasodutos através de comunicações de telemetria. Ele é definido como:

"Um método de transporte de dados leve para publicação e assinatura sobre TCP/IP com várias garantias de entrega. Otimizado para uso mínimo de largura de banda de rede e facilidade de implementação em sistemas embarcados, ele serve como um mecanismo de transporte onde é possível decidir o que enviar e quais ações tomar ao receber os dados."(STANFORD-CLARK, 2004, p. 4)

Nos sistemas embarcados, o MQTT é implementado nos dispositivos para permitir a troca de dados (STANFORD-CLARK, 2004). Os dispositivos podem publicar dados de sensores e *status* usando tópicos MQTT, enquanto as aplicações podem se inscrever nesses tópicos para receber e processar os dados em tempo real (YASSEIN *et al.*, 2017).

Do lado do servidor, os *brokers* MQTT são implantados para gerenciar a troca de mensagens entre os dispositivos e as aplicações. Segundo Yassein *et al.* (2017), esses *brokers* são responsáveis por rotear as mensagens entre os clientes MQTT, garantindo a entrega dos dados. Além disso, os desenvolvedores podem usar bibliotecas MQTT em suas aplicações para se conectar aos *brokers*, facilitando a implementação de funcionalidades de comunicação em seus sistemas (BANKS; BORGENDALE; GUPTA, 2019).

Em sistemas IoT, o MQTT é utilizado para conectar dispositivos a plataformas de nuvem, permitindo o monitoramento e controle remoto dos dispositivos (JUNIOR *et al.*, 2022). As aplicações podem processar os dados recebidos dos dispositivos IoT, realizar análises em tempo real e realizar ações com base nos eventos detectados (FERRÁNDEZ-PASTOR *et al.*, 2018; HSU *et al.*, 2020). Suas principais características são definidas na Tabela 5.

A implementação de sistemas de IoT na zona rural enfrenta desafios devido à baixa disponibilidade de internet e eletricidade na maioria dos casos (RIJSWIJK *et al.*, 2021). No entanto, o MQTT se destaca como uma solução para esses cenários. Ele oferece uma comunicação assíncrona entre dispositivos, reduzindo a dependência de conexões de alta velocidade e garantindo a entrega de mensagens (STANFORD-CLARK, 2004; YASSEIN *et al.*, 2017). Essa característica o torna uma opção para superar limitações de infraestrutura em áreas remotas.

Sua eficiência o posiciona como uma ferramenta para a agricultura inteligente (SINGH *et al.*, 2022), tema que será discutido na próxima subseção.

Tabela 5 – Características do protocolo MQTT

Característica	Descrição
Leve	É leve e eficiente em termos de largura de banda e recursos de hardware.
Assíncrono	Dispositivos podem enviar e receber mensagens de forma independente, sem bloqueio ou espera ativa.
Padrão aberto	É um protocolo aberto e bem documentado.
Qualidade de Serviço (QoS)	Oferece três níveis de QoS para garantir a entrega confiável das mensagens: QoS 0 (entrega no máximo uma vez), QoS 1 (entrega pelo menos uma vez) e QoS 2 (entrega exatamente uma vez).
Tópicos	Usa um modelo de publicação/assinatura, onde os dispositivos podem publicar mensagens em tópicos específicos e se inscrever para receber mensagens de tópicos de interesse.
Baixa latência	É projetado para minimizar a latência de rede, tornando-o adequado para casos de uso em tempo real, como telemetria, monitoramento e controle remoto.
Segurança	Suporta autenticação e criptografia.
Retenção de mensagens	Permite que o broker MQTT retenha as últimas mensagens publicadas em um tópico.

Fonte: Adaptado de Banks, Borgendale e Gupta (2019).

## 2.8 Agricultura inteligente

A agricultura passou por uma transformação impulsionada pelo advento das tecnologias digitais, marcando a era da agricultura inteligente.

"Também conhecida como agricultura 4.0 ou agricultura digital, é a próxima fase da agricultura industrial, impulsionada pela integração de tecnologias na agricultura."(SINGH *et al.*, 2022, p. 424)

Essa mudança de práticas agrícolas tradicionais para agricultura inteligente tornou-se imperativa. Ela é caracterizada pela integração de tecnologias emergentes, como a IoT, Análise de Dados em Grande Escala - BDA (sigla do inglês, *Big Data Analysis*), computação em nuvem e IA. Seu objetivo é atender às crescentes demandas por segurança alimentar em uma população global em rápido crescimento (SINGH *et al.*, 2022; GARG; ALAM, 2023).

Segundo Garg e Alam (2023), a convergência dessas tecnologias possibilitou o paradigma da agricultura orientada por dados, onde a coleta, análise e tomada de decisões em tempo real são centrais para otimizar práticas agrícolas.

Singh *et al.* (2022) complementa que, ao elucidar o estado atual da adoção de tecnologia digital na agricultura e identificar perspectivas futuras, são abertos caminhos para práticas agrícolas sustentáveis e eficientes.

A agricultura inteligente, ao incorporar IA e análise avançada de dados, busca orientar decisões mais adaptativas e holísticas (GARG; ALAM, 2023). Para isso, é necessário a coleta e aplicação precisa de dados para otimizar a produção (LAMINE *et al.*, 2024), a qual será abordada na próxima subseção.

## 2.9 Agricultura de precisão

A agricultura de precisão pode ser definida como:

"A aplicação de tecnologias e princípios para gerenciar a variabilidade espacial e temporal associada a todos os aspectos da produção agrícola com o propósito de melhorar o desempenho das culturas e a qualidade ambiental." (PIERCE; NOWAK, 1999, p. 1)

Segundo Pierce e Nowak (1999), o sucesso da agricultura de precisão está relacionado à sua capacidade de avaliar e gerenciar o continuum espaço-tempo na produção de culturas. Ela é habilitada pela tecnologia e integrada por tecnologias específicas que permitem avaliar e gerenciar a variabilidade em níveis de detalhe nunca antes alcançados (ZHANG; WANG; WANG, 2002). No entanto, o sucesso agronômico da agricultura de precisão tem sido limitado e inconsistente, embora bastante convincente em alguns casos, complementa Zhang, Wang e Wang (2002).

Atualmente, de acordo com Lamine *et al.* (2024), a agricultura de precisão é uma prática que busca observar, medir e responder às variações que ocorrem dentro de um mesmo campo e entre diferentes campos. Isso significa identificar as diferenças no solo, nas plantas e nas condições ambientais, tanto dentro de uma área de cultivo quanto entre áreas agrícolas distintas. Ao entender essas variações, é possível adotar abordagens específicas para cada local. Exemplo disso é ajustar a quantidade de fertilizantes ou de água, o que torna a produção mais eficiente e contribui para a conservação ambiental (LAMINE *et al.*, 2024).

Apesar dos avanços tecnológicos e inovações, a agricultura de precisão ainda enfrenta desafios em países em desenvolvimento, especialmente em agricultura de pequena escala, conforme apontado por Lamine *et al.* (2024). No entanto, existem amplas oportunidades e perspectivas para a adoção de inovações em agricultura de precisão (LAMINE *et al.*, 2024).

A agricultura de precisão compreende um conjunto de tecnologias que combinam sensores e sistemas de informação. Essas tecnologias são utilizadas para otimizar a produção, levando em consideração a variabilidade e as incertezas dentro dos sistemas agrícolas. Ela fornece meios para monitorar a cadeia de produção de alimentos e gerenciar tanto a quantidade quanto a qualidade dos produtos agrícolas (GEBBERS; ADAMCHUK, 2010).

Medel-Jiménez *et al.* (2024) demonstraram que a avaliação do ciclo de vida de tecnologias de agricultura de precisão possui potencial para melhorar a sustentabilidade agrícola, embora seu impacto ambiental exato permaneça incerto. Estudos comparativos mostraram que a agricultura de precisão pode reduzir os impactos ambientais da produção agrícola, especialmente no que diz respeito à gestão da irrigação (MEDEL-JIMÉNEZ *et al.*, 2024). No entanto, é importante considerar variáveis locais para uma avaliação ambiental abrangente, como será abordado na próxima subseção.

## **2.10 Gestão da irrigação**

Burton (2010) apresenta a gestão eficaz da irrigação como um desafio que demanda pesquisas e práticas sustentáveis, juntamente com o uso de tecnologias apropriadas. Considerando a complexidade das demandas e ofertas de recursos hídricos em perímetros irrigados, é essencial a utilização de técnicas e instrumentos para auxiliar profissionais nesse processo (RAMOS *et al.*, 2022). Neste contexto, é fundamental o conhecimento da variabilidade espacial dos atributos do solo. Segundo Ramos *et al.* (2022), essa variabilidade está relacionada ao fluxo e armazenamento de água, usados como parâmetros para a otimização da produção agrícola.

Pereira, Oweis e Zairi (2002) discutem diversos aspectos relacionados à gestão da irrigação em propriedades agrícolas, incluindo o uso de águas residuais tratadas e águas salinas. Os autores destacam a importância de estratégias como a uniformidade de distribuição e práticas de irrigação suplementar e deficitária para reduzir a necessidade de água na agricultura.

O estudo conduzido por Burton (2010) complementa essas discussões. Ele destaca a importância das tecnologias, tais como o sensoriamento e a telemetria das condições ambientais, na agricultura irrigada. Além disso, ressalta a necessidade de desenvolver metodologias capazes de avaliar os benefícios sociais, econômicos e ambientais resultantes da otimização desse processo.

A integração dessas estratégias e tecnologias possibilita uma gestão mais eficiente da irrigação em áreas específicas, onde o consumo hídrico é variável (CARMODY; TURRAL, 2023). Essa integração, aliada a espacialização dos dados, apresenta melhorias efetivas do uso da água em escalas, como a de um perímetro irrigado (BURTON, 2010; RAMOS *et al.*, 2022). Na próxima subseção, será discutida uma das perspectivas da gestão hídrica na agricultura relevante para este trabalho.

### **2.10.1 Gestão baseada no consumo de água**

Segundo Carmody e Turrel (2023), a gestão baseada no consumo de água traz uma perspectiva inovadora. Ela considera não apenas a entrega de água, mas também

o uso real da mesma na agricultura. Isso contribui para a eficiência do uso dos recursos hídricos e para a sustentabilidade dos sistemas agrícolas.

Na Figura 2, é apresentado o ciclo da água. Ele é definido como um processo contínuo que envolve a precipitação, infiltração, evaporação, evapotranspiração (ET) e condensação (CARMODY; TURRAL, 2023). A água precipitada em forma de chuva ou neve alimenta os reservatórios superficiais e subterrâneos. Parte dela infiltra-se no solo, recarregando os lençóis freáticos, enquanto outra parte escoa superficialmente, mantendo o fluxo de rios e riachos. A evaporação dos corpos d'água e a ET das plantas retornam a umidade à atmosfera, onde se condensa formando nuvens, completando o ciclo.

Neste modelo, a água pode ser utilizada de duas formas: benéfica, como na transpiração das plantas para a produção de alimentos, e não benéfica, como na evaporação do solo nu ou de superfícies de água nuas. Também existem os fluxos de retorno não consumidos, que podem ser recuperados para consumo benéfico ou para manter os ecossistemas aquáticos (Figura 2). Eles contribuem para o abastecimento de outros usuários a jusante e para a saúde dos rios e lençóis freáticos (CARMODY; TURRAL, 2023).

De acordo com Carmody e Turrel (2023), para alcançar economias reais de água, é necessário atender a quatro requisitos no contexto de um quadro mais amplo. Nesse quadro, os fluxos ambientais são considerados para sustentar os sistemas agrícolas e as pessoas que dependem deles. Esses requisitos incluem:

1. A determinação dos requisitos específicos de ET para a produção de culturas;
2. A definição dos direitos de água em termos de extração, consumo e fluxos de retorno;
3. O uso do monitoramento de ET e cobertura do solo para ajustar as alocações de água;
4. A comparação da ET alvo com a ET real para garantir um balanço hídrico adequado.

Para compreender a ET e sua importância na gestão da água, é necessário analisar o cenário em que a mesma surge, como será abordado na próxima subseção.

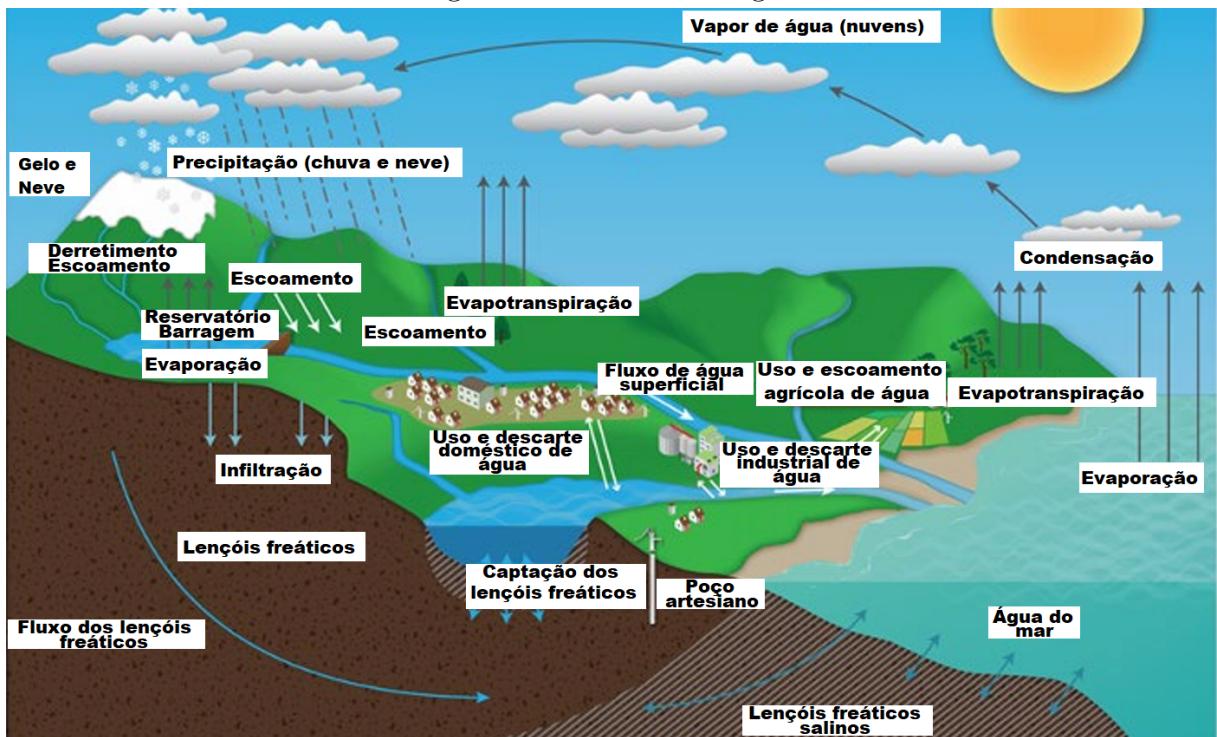
## **2.11 Evapotranspiração**

A ET descreve o processo fundamental de transferência de água da superfície da Terra para a atmosfera (CARMODY; TURRAL, 2023). A ET desempenha um papel importante na dinâmica hídrica. Foi observado que:

"Três tipos de superfície são importantes no retorno da chuva à atmosfera.

Para extensas áreas de terra, são eles, em ordem de importância: vegetação, na qual as folhas das plantas atuam como superfícies transpirantes; solo nu ou

Figura 2 – Ciclo Hidrológico



Fonte: Retirado de Carmody e Turrel (2023).

pousio, de onde a água evapora na interface solo-ar ou logo abaixo dela e em águas abertas, a partir das quais a evaporação ocorre diretamente." (PENMAN, 1948, p. 121)

Penman (1948) explica que a irrigação exerce uma influência direta na evaporação ao fornecer água ao solo. Esse processo não apenas afeta a disponibilidade de água para a evaporação, mas também influencia a temperatura e a umidade do solo, fatores determinantes para a taxa de ET. As interações entre a atividade humana e os ciclos naturais da água são particularmente relevantes no contexto das mudanças climáticas globais.

Conforme destacado por Yang *et al.* (2023), tem-se observado uma tendência de aceleração no aumento da ET desde a década de 1980. Essa tendência está intimamente ligada à expansão da vegetação e ao aumento da área plantada, o que é evidenciado pelo crescimento do índice de área foliar (LAI).

O aumento da ET pode ter consequências tanto positivas quanto negativas. Por um lado, pode indicar um crescimento saudável da vegetação, o que pode ser benéfico para a agricultura e para a absorção de carbono. Por outro lado, um aumento excessivo da ET pode resultar em uma maior demanda por água, potencialmente levando ao estresse hídrico, especialmente em regiões onde a disponibilidade de água é limitada. Yang *et al.* (2023) indicam a mensuração da ET para avaliar os impactos das atividades antropogênicas no ciclo hidrológico e no clima.

Métodos de telemetria da ET têm mostrado resultados promissores na compreensão desse fenômeno em escalas regional e global. De acordo com as revisões feitas por Li *et al.* (2009) e Zhang, Kimball e Running (2016), esses métodos variam desde modelos de equações simplificadas até modelos de balanço energético de duas fontes mais complexos.

Segundo Li *et al.* (2009), os modelos mais eficazes utilizam sensores telemétricos que capturam a radiação emitida pelo solo, cobrindo desde o espectro visível até o infravermelho térmico. Essa abrangência espectral permite uma estimativa precisa da radiação líquida, um dos principais componentes do balanço de energia da superfície (LI *et al.*, 2009). No entanto, há uma dependência, em certa medida, de medições auxiliares baseadas em solo para derivar os fluxos de calor turbulentos em uma escala regional (ZHANG; KIMBALL; RUNNING, 2016).

O coeficiente de determinação ( $R^2$ ) é uma medida estatística que indica a proporção da variabilidade dos dados explicada pelo modelo (MOOD; GRAYBILL; BOES, 1974). Ele é calculado pela fórmula:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

em que:  $y_i$  são os valores observados,  $\hat{y}_i$  são os valores previstos pelo modelo e  $\bar{y}$  é a média dos valores observados. O valor de  $R^2$  varia de 0 a 1, onde um valor próximo de 1 indica que o modelo explica a maior parte da variação nos dados. Anderson *et al.* (2022) apresenta um método de telemedição de ET com coeficiente de determinação de 0,88. Neste estudo, a fusão de dados de diferentes satélites (Landsat 8 e Sentinel-2) com resoluções espaciais e temporais variáveis proporciona uma estimativa contínua e precisa da ET diária em escala de campo.

Plataformas como o *USGS EarthExplorer* e o *Copernicus Open Access Hub* possuem acesso aberto as imagens dos satélites citados com alta resolução. Essas imagens serão utilizadas para a obtenção de dados no presente trabalho.

Nas duas próximas subseções, serão abordados conceitos e técnicas complementares as definições apresentadas acima, que serão de utilidade para o desenvolvimento do sistema proposto.

### **2.11.1 Evapotranspiração de referência**

A ET de referência (ETo) é definida como:

"A taxa de evapotranspiração de uma superfície de referência, sem falta de água."(ALLEN *et al.*, 1998, p. 7)

Para calcular a ETo, pode-se empregar o método da FAO-56 apresentado por Allen *et al.* (1998), uma abordagem recomendada pela Organização das Nações Unidas para Alimentação e Agricultura, que é definida pela equação:

$$ETo = \frac{0.408 \cdot \Delta(R_n - G) + \gamma \cdot \frac{900}{T+273} \cdot u2 \cdot (e_s - e_a)}{\Delta + \gamma \cdot (1 + 0.34 \cdot u2)} \quad (1)$$

em que: ETo é a evapotranspiração de referência;  $R_n$  é a radiação líquida;  $G$  é o calor do solo;  $\Delta$  é o déficit de pressão de vapor;  $\gamma$  é a constante psicrométrica;  $T$  é a temperatura média do ar;  $u2$  é a velocidade do vento a 2 metros de altura;  $e_s$  é a pressão de vapor do ar saturado;  $e_a$  é a pressão de vapor do ar.

### **2.11.2 Evapotranspiração de cultura**

A ET de cultura (ETc), por sua vez, é definida como:

"A evapotranspiração de culturas livres de doenças, bem fertilizadas, cultivadas em grandes campos, sob condições ótimas de água no solo e atingindo plena produção nas condições climáticas dadas." (ALLEN *et al.*, 1998, p. 7)

A ETc pode ser calculada como o produto da ETo e um determinado coeficiente de cultura (Kc), como proposto por Allen *et al.* (1998):

$$ETc = Kc \cdot ETo \quad (2)$$

O Kc é um fator que leva em consideração as características específicas da cultura, como a fase de desenvolvimento, a densidade de plantio e a cobertura do solo. Ele é determinado experimentalmente e varia ao longo do ciclo de cultivo (BERNARDO *et al.*, 2008).

A determinação da ETc em tempo real é um dos principais objetos de estudo dessa pesquisa. Na próxima seção, serão apresentados trabalhos similares e como a proposta deste trabalho se diferencia deles.

## **2.12 Revisão bibliográfica**

O monitoramento da ETc é uma prática que possui diversas abordagens e métodos, dependendo do ambiente e do nível de precisão desejado. A ETc pode ser estimada por meio de métodos diretos, como a pesagem de lisímetros, ou indiretos, como a equação de Allen *et al.* (1998) apresentada na seção anterior. Entre essas abordagens, esta pesquisa propõe uma aplicação *web* para o monitoramento da ETc em tempo real, combinando métodos indiretos para esse fim.

A seguir serão apresentados outros trabalhos com abordagens semelhantes.

### **2.12.1 Integração de dados e padrões web para modelos de ETo**

Zhang *et al.* (2009) abordam o desenvolvimento de sistemas de disseminação de dados em tempo real baseados na aquisição de dados. Eles fazem uso de dados geoespaciais

para estimar a ETo diária, apontando a necessidade de métodos de acesso públicos a esses dados. Isso contrasta com os métodos utilizados nos serviços existentes com o mesmo propósito.

Os autores desenvolveram um sistema que utiliza padrões da *Open Geospatial Consortium* (OGC) e o protocolo *Network Data Access Protocol* (OPeNDAP) para publicar dados de estações meteorológicas. A arquitetura desse sistema incorpora diversas fontes de dados distribuídas e heterogêneas, como registros diários e horários de estações geoestacionárias. Além disso, imagens de satélite e resultados de modelos meteorológicos são utilizados para uma análise espacial.

Através de serviços construídos sobre essas fontes, o sistema prova a eficácia de uma abordagem distribuída. Isso permite tanto a utilização quanto o desenvolvimento de novos serviços e aplicações sem interferir nos já existentes.

O trabalho de Zhang *et al.* (2009) promove várias vantagens em relação à interoperabilidade e padronização de dados. No entanto, esse método não considera medições realizadas diretamente no local de interesse (*in situ*). Isso, em cenários menores, como o de uma propriedade rural, pode ser uma limitação em questão de precisão e custo. Além disso, o sistema proposto não oferece uma interface para a visualização e análise dos dados, o que pode dificultar a interpretação dos resultados por usuários finais.

### **2.12.2 Ferramentas web para determinação de ETo**

O estudo de Wu *et al.* (2021) foca no desenvolvimento e implementação de APIs em nuvem, chamadas *ETWatch*, para a geração de dados regionais da ETo. Essas APIs são desenvolvidas para proporcionar acesso fácil e em tempo real aos dados de ETo.

A metodologia combina dados de sensoriamento remoto, medições *in situ* e modelagem matemática para calcular a evapotranspiração. As APIs facilitam a disseminação desses dados para diversos usuários e aplicações. Entre as vantagens do sistema *ETWatch*, destacam-se o acesso em tempo real aos dados de ETo, permitindo uma resposta rápida para a gestão de recursos hídricos. Além disso, o sistema fornece uma interface para monitorar os dados, facilitando o acesso e a leitura das informações obtidas.

Apesar das vantagens, Wu *et al.* (2021) identificam alguns pontos que necessitam de maior preocupação. A precisão das estimativas pode ser limitada pela qualidade e disponibilidade dos dados de entrada, especialmente em regiões com poucas medições *in situ*. A complexidade de integrar modelos climáticos e hidrológicos pode apresentar desafios computacionais e de implementação. Além disso, há a necessidade de ajustar os modelos para responder a variações e mudanças climáticas contínuas.

Uma abordagem semelhante foi realizada por Payero *et al.* (2011) com o desenvolvimento de uma aplicação *desktop* chamada *Crop Water Use*. Essa aplicação foi desenvolvida para estimar os tempos de irrigação baseados na estimativa de ETc. Assim como Wu *et al.* (2021), Payero *et al.* (2011) utilizaram medições e modelos matemáticos

para estimar a ETo. Contudo, eles realizaram o cálculo da ETc com base na ETo e em valores de Kc específicos para cada cultura, sendo estes inseridos manualmente pelo usuário na aplicação.

Porém, a aplicação *CropWaterUse* não oferece uma interface de acesso distribuído para a visualização dos dados, seja por meio de APIs ou nuvens de dados. Isso limita sua utilização em cenários remotos ou até mesmo restritos, como propriedades rurais.

### **2.12.3 Desenvolvimento e aplicação de ferramentas para ETo**

No estudo de Bortolin *et al.* (2019), foi desenvolvido um sistema que automatiza o cálculo da ETo, integrando dados de estações meteorológicas e métodos matemáticos. Diferente das pesquisas apresentadas, esse sistema adota o padrão de arquitetura *Model-View-Controller (MVC)* para o projeto. Esse *design* separa a lógica de negócio, definida como *Model*, da interface gráfica, chamada de *View*. O *Controller* é responsável por intermediar a comunicação entre o *Model* e a *View*. Dessa forma, é possível desacoplar as camadas do sistema, facilitando a manutenção e a evolução do mesmo.

A aplicação de Bortolin *et al.* (2019) é eficiente quanto a interface e a arquitetura. No entanto, o sistema não considera medições *in situ* para o cálculo da ETo, o que pode limitar a precisão dos resultados para pequenas áreas como já mencionado.

Uma abordagem que contorna esse problema é apresentada por Danilescu (2022). Ele destaca a criação e funcionalidade do *ETCalc*, uma ferramenta gratuita para estimativa de ETo. Seu diferencial está na possibilidade de fornecer dados meteorológicos de múltiplas fontes. Ele permite que os usuários calculem a ETo em suas propriedades, com base em dados fornecidos de estações e medições locais. O *ETCalc*, dentre as pesquisas encontradas, é a aplicação que possui o processamento matemático mais robusto, considerando diferentes métodos e parâmetros. Também fornece uma interface para a inserção e visualização dos dados, facilitando o acesso e o uso da ferramenta.

Apesar das vantagens, o *ETCalc* falha como processo automatizado e em tempo real. O usuário precisa inserir manualmente os dados de entrada, o que pode ser um processo demorado e propenso a erros. Além disso, a ferramenta não oferece APIs para a integração com outros sistemas, o que limita sua utilização em cenários mais complexos.

Com base nas pesquisas apresentadas, na tabela 6, é possível observar as tecnologias e métodos utilizados em cada uma delas. Também é possível identificar as limitações de cada abordagem, que servirão de base para o desenvolvimento da aplicação proposta nesta pesquisa.

Tabela 6 – Comparaçāo das tecnologias

Referência	Tecnologias	Limitações
Bortolin <i>et al.</i> (2019)	<i>PHP, Smarty Template Engine, HTML5, CSS3, JavaScript, Bootstrap, Open-Layers e PostgreSQL.</i>	Escalabilidade limitada e performance inferior em aplicações complexas.
Danielescu (2022)	<i>Python, Django, HTML5, CSS3, JavaScript, jQuery, Bootstrap e PostgreSQL.</i>	Menor performance em aplicações de alta carga e necessidade de alto conhecimento técnico em <i>Python</i> .
Payero <i>et al.</i> (2011)	<i>C#, ASP.Net, Visual Studio.NET, JavaScript, Tee-Chart e SQL Server.</i>	Dependência de plataforma <i>Windows</i> e custos elevados de licenciamento.
Wu <i>et al.</i> (2021)	<i>Python, Bootstrap, jQuery, Alibaba Cloud, IDL, Docker, JSON e PostgreSQL.</i>	Complexidade na integração e maior curva de aprendizado para configuração.
Zhang <i>et al.</i> (2009)	<i>OGC standards, OPeNDAP standards, WMS, WCS, WFS, GIS, DODS, DAS, DDS e Middleware components.</i>	Alta complexidade de implementação e necessidade de especialização em padrões <i>OGC</i> .
Presente pesquisa	<i>C, Node.js, Next.js, Shadcn/ui, Tailwind CSS e MQTT.</i>	Alto consumo de memória em aplicações de alta carga e maior curva de aprendizado inicial para integração do MQTT.

Fonte: Elaborado pelo autor, 2024.

### 3 MATERIAIS E MÉTODOS

Nesta seção, são apresentados os materiais e métodos utilizados no desenvolvimento do sistema. Isso inclui as ferramentas e tecnologias empregadas, a arquitetura do sistema e as integrações realizadas.

#### 3.1 Descrição da área de estudo

O sistema em questão foi implementado no Instituto Federal de Minas Gerais (IFMG), *Campus Bambuí*, especificamente no laboratório de sistemas embarcados. Esse local foi escolhido por possuir uma infraestrutura adequada para o teste de sensores e equipamentos necessários para a captura e transmissão de dados. A escolha do laboratório também se deve à disponibilidade de supervisão técnica e apoio logístico para a instalação e validação dos equipamentos.

#### 3.2 Classificação da pesquisa

A pesquisa realizada no desenvolvimento do sistema de monitoramento meteorológico pode ser classificada segundo os seguintes critérios: quanto aos objetivos, à natureza, aos procedimentos técnicos, e à abordagem do problema.

##### 3.2.1 *Quanto aos objetivos*

De acordo com os objetivos, esta pesquisa é classificada como descritiva e exploratória.

A pesquisa é descritiva porque visa descrever as características dos fenômenos meteorológicos monitorados pelos sensores, bem como o comportamento do sistema de monitoramento implementado. O foco está em detalhar como as variáveis meteorológicas, como temperatura, umidade, pressão, velocidade do vento e luminosidade, são capturadas e processadas pelo sistema, além de documentar o desempenho dos componentes de *hardware* e *software* utilizados.

A pesquisa também é exploratória porque busca investigar a aplicabilidade e eficiência de diferentes tecnologias e métodos no monitoramento meteorológico. Isso inclui a experimentação com diversos sensores e técnicas de integração de dados em um contexto específico. O caráter exploratório é acentuado pela busca de soluções tecnológicas que possam ser escaláveis e aplicáveis a diferentes contextos de monitoramento ambiental.

##### 3.2.2 *Quanto à natureza*

Quanto à natureza, a pesquisa é aplicada. O objetivo principal é o desenvolvimento de uma solução prática para um problema real, neste caso, a necessidade de

monitoramento meteorológico em tempo real. O conhecimento gerado é voltado para a aplicação direta no sistema desenvolvido, com o propósito de melhorar a coleta, processamento e análise de dados meteorológicos. A pesquisa aplicada busca, portanto, transformar o conhecimento teórico em soluções práticas e funcionais.

### ***3.2.3 Quanto aos procedimentos técnicos***

Em relação aos procedimentos técnicos, a pesquisa é classificada como experimental. Durante o desenvolvimento do sistema, foram realizados testes e experimentos com diferentes componentes *hardware* e *software*, a fim de avaliar seu desempenho e adequação ao monitoramento meteorológico. Esses testes incluíram a validação dos sensores, a integração com o microcontrolador ESP32, e a transmissão de dados através do protocolo MQTT. A pesquisa experimental permitiu ajustes e refinamentos no sistema até alcançar um protótipo funcional e validado.

### ***3.2.4 Quanto à abordagem do problema***

Quanto à abordagem do problema, esta pesquisa é qualitativa. O sistema desenvolvido coleta, processa e fornece dados meteorológicos numéricos, que são quantificados e apresentados em *cards* interativos. Isso é feito através de uma interface de visualização com os dados capturados pelos sensores. Assim, foi possível a observação de variáveis como temperatura, umidade, pressão, e velocidade do vento, além de possibilitar o cálculo de índices climáticos, como a evapotranspiração. A abordagem qualitativa foi escolhida como forma de interpretar os dados meteorológicos de maneira específica para cada área de coleta, permitindo uma análise detalhada das variáveis ambientais em seus contextos locais. Essa especificidade garante que os dados refletem as condições e particularidades de cada região, favorecendo uma compreensão fundamentada dos fenômenos climáticos estudados.

## **3.3 Componentes do Sistema de Monitoramento Meteorológico**

O sistema desenvolvido utiliza uma série de sensores para monitorar em tempo real diferentes parâmetros meteorológicos. Nesta seção, são apresentados os sensores e módulos utilizados no sistema, suas funcionalidades, características principais e como foram integrados ao microcontrolador ESP32 para a coleta e transmissão de dados meteorológicos.

### ***3.3.1 Sensor de Temperatura e Umidade DHT22***

O sensor DHT22 (Figura 3) mede a temperatura e a umidade do ar, com precisão dentro do intervalo de operação (AOSONG ELECTRONICS, 2024). O pino de saída do sensor foi conectado ao pino GPIO 4 do ESP32, configurado no código como

DHTPIN. Para alimentação, o sensor foi conectado ao pino 3V3 e ao GND do ESP32. Na Tabela 7 são apresentadas as especificações técnicas do sensor.

O uso do sensor DHT22 no sistema justifica-se pela sua precisão dentro da faixa de operação desejada, característica necessária para o monitoramento climático (AOSONG ELECTRONICS, 2024). A medição precisa de temperatura e umidade relativa do ar é indispensável em aplicações meteorológicas, pois esses parâmetros influenciam diretamente fenômenos atmosféricos e condições ambientais. Além disso, a facilidade de integração ao ESP32 por meio de um único pino digital reduz a complexidade do circuito, otimizando a implementação do sistema. A escolha do DHT22, em detrimento de outros sensores, também considera o seu equilíbrio entre custo-benefício e desempenho, sendo utilizado em projetos de monitoramento ambiental (FERRÁNDEZ-PASTOR *et al.*, 2018; MEDEL-JIMÉNEZ *et al.*, 2024).

Figura 3 – Sensor de temperatura e umidade DHT22



Fonte: Retirado de Aosong Electronics (2024).

Tabela 7 – Especificações técnicas do sensor DHT22

Parâmetro	Detalhes
Faixa de operação	Umidade: 0-100% RH; Temperatura: -40°C a 80°C
Precisão	Umidade: $\pm 2\%$ RH (Máx $\pm 5\%$ RH); Temperatura: $< \pm 0.5^\circ\text{C}$
Resolução/Sensibilidade	Umidade: 0.1% RH; Temperatura: 0.1°C
Período de amostragem	Média: 2 s
Dimensões	22x28x5 mm

Fonte: Retirado de Aosong Electronics (2024).

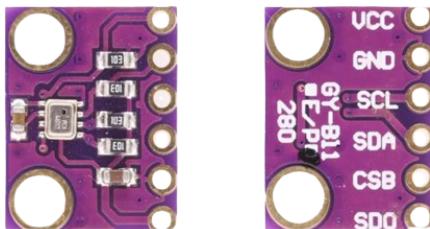
### 3.3.2 Sensor de Pressão BMP280

O sensor BMP280 (Figura 4) mede a pressão atmosférica com precisão, além de oferecer medições de temperatura (BOSCH SENSORTEC, 2015). Ele utiliza o barramento I2C, sendo conectado aos pinos padrão do ESP32 para SDA e SCL. Para alimentação,

o sensor também foi conectado ao pino 3V3 e ao GND do ESP32. Na Tabela 8 são apresentadas as especificações técnicas do sensor.

A inclusão do sensor BMP280 no sistema deve-se à sua capacidade de medir pressão atmosférica com precisão, o que é necessário para a análise de condições meteorológicas e a previsão de mudanças climáticas (BOSCH SENSORTEC, 2015). A pressão atmosférica é um parâmetro utilizado na modelagem de padrões climáticos, e o BMP280 oferece dados confiáveis dentro de uma faixa de operação desejada, adequados tanto para altitudes elevadas quanto para ambientes ao nível do mar. Além disso, a integração por barramento I2C simplifica a comunicação com o microcontrolador ESP32, permitindo o compartilhamento de pinos com outros dispositivos. A resolução do sensor possibilita a detecção de variações sutis na pressão, úteis para aplicações detalhadas, como estudos de microclimas (PRATHIBHA; HONGAL; JYOTHI, 2017; LAMINE *et al.*, 2024).

Figura 4 – Sensor de pressão barométrica BMP280



Fonte: Retirado de Bosch Sensortec (2015).

Tabela 8 – Especificações técnicas do sensor BMP280

Parâmetro	Detalhes
Faixa de operação	300-1100 hPa
Precisão absoluta	$\sim \pm 1$ hPa (950-1050 hPa)
Precisão relativa	$\pm 0.12$ hPa (700-900 hPa)
Resolução/Sensibilidade	0.01 hPa ( $< 10$ cm)
Período de amostragem	Média: 5.5 ms
Dimensões	8x15.62x0.85 mm

Fonte: Retirado de Bosch Sensortec (2015).

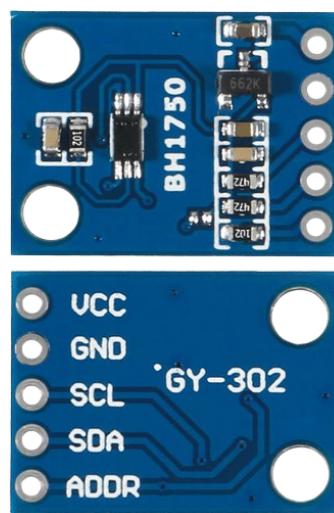
### 3.3.3 Sensor de Luminosidade BH1750

O BH1750 (Figura 5) é um sensor digital de luminosidade que mede a intensidade luminosa em lux. Ele também utiliza o barramento I2C, compartilhando os mesmos pinos do BMP280. Para alimentação, o sensor também foi conectado ao pino 3V3 e ao GND do ESP32. Na Tabela 9 são apresentadas as especificações técnicas do sensor.

A escolha do sensor BH1750 é justificada por sua capacidade de medir a intensidade luminosa em lux, unidade aceita no estudo da iluminação ambiental e solar (PRATHIBHA; HONGAL; JYOTHI, 2017; VIJH *et al.*, 2024; IDRISI *et al.*, 2024). A

intensidade luminosa afeta diversos processos meteorológicos e biológicos, como evaporação e fotossíntese, tornando sua medição uma parte integrante do monitoramento climático (ANDERSON *et al.*, 2022). O BH1750 destaca-se por sua sensibilidade, faixa de operação e comunicação digital por I2C, que asseguram medições rápidas e precisas (ROHM SEMICONDUCTOR, 2011). Além disso, a possibilidade de alternar entre modos de alta e baixa resolução adapta-se às necessidades específicas do sistema, proporcionando flexibilidade na coleta de dados.

Figura 5 – Sensor de luminosidade BH1750



Fonte: Retirado de Rohm Semiconductor (2011).

Tabela 9 – Especificações técnicas do sensor BH1750

Parâmetro	Detalhes
Faixa de operação	0-65535 lx
Precisão	$\pm 0.24$ lx
Resolução/Sensibilidade	Alta Resolução: 1 lx; Baixa Resolução: 4 lx
Período de amostragem	Alta Resolução: 120-180 ms; Baixa Resolução: 16-24 ms
Dimensões	13.9x18.5 mm

Fonte: Retirado de Rohm Semiconductor (2011).

### 3.3.4 Anemômetro RS-FSJT-N01

O anemômetro RS-FSJT-N01 (Figura 6) é utilizado para medir a velocidade do vento. Ele opera através do protocolo Modbus RTU sobre RS485, enviando os dados de velocidade em m/s. Para a integração ao ESP32, foi necessário o uso do módulo conversor RS485 para TTL, que possibilita a comunicação compatível com os níveis lógicos do microcontrolador. Na Tabela 10 são apresentadas as especificações técnicas do anemômetro.

O anemômetro RS-FSJT-N01 foi selecionado devido à sua precisão na medição da velocidade do vento, um dos parâmetros utilizados em estudos meteorológicos e ambientais (PRATHIBHA; HONGAL; JYOTHI, 2017; IDRISI et al., 2024). A velocidade do vento influencia diretamente fenômenos como a dispersão de poluentes, a sensação térmica e o transporte de umidade na atmosfera (ZHANG; KIMBALL; RUNNING, 2016). Este modelo específico utiliza o protocolo Modbus RTU sobre RS485, uma tecnologia que garante comunicação confiável em ambientes sujeitos a ruído eletromagnético (RENKE, 2024).

Figura 6 – Anemômetro RS-FSJT-N01



Fonte: Retirado de Renke (2024).

Tabela 10 – Especificações técnicas do Anemômetro RS-FSJT-N01

Parâmetro	Detalhes
Faixa de operação	0-30 m/s
Precisão	$\pm 0.3$ m/s
Resolução/Sensibilidade	0.1 m/s
Período de amostragem	$\leq 1$ s
Dimensões	182x80x160 mm

Fonte: Retirado de Renke (2024).

### 3.3.5 Comparador de Umidade do Solo LM393

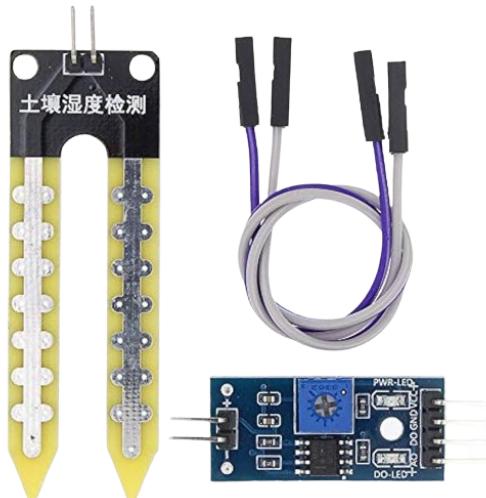
O módulo LM393 (Figura 7) detecta a presença de umidade no solo com base na variação de condutividade elétrica. Ele foi conectado ao pino GPIO 13 do ESP32, configurado como entrada digital. Para alimentação, o módulo foi conectado ao pino 3V3 e ao GND do ESP32.

Para solos úmidos (acima de 50% de umidade), o módulo fornece um sinal digital baixo (0), enquanto para solos secos (abaixo de 50% de umidade), o sinal é alto

(1). O limiar de umidade pode ser ajustado através do potenciômetro presente no módulo usando uma chave *philips*.

O comparador LM393 foi escolhido por ser ajustável e sensível a variações de umidade no solo (ON SEMICONDUCTOR, 2024). A umidade do solo é um fator importante no contexto da irrigação, influenciando diretamente a absorção de nutrientes pelas plantas (ZHANG; KIMBALL; RUNNING, 2016). O LM393 permite a detecção rápida e precisa de mudanças na umidade do solo, tornando a coleta em tempo real mais eficiente. Na Tabela 11 são apresentadas as especificações técnicas do comparador.

Figura 7 – Comparador de umidade do solo LM393



Fonte: Retirado de ON Semiconductor (2024).

Tabela 11 – Especificações técnicas do Comparador LM393

Parâmetro	Detalhes
Faixa de operação	2.0 V a 36 V
Precisão	$\pm 5.0 \text{ mV}$
Resolução/Sensibilidade	5 nA (corrente de offset)
Período de amostragem	$\leq 1.3 \mu\text{s}$
Dimensões	99x16 mm

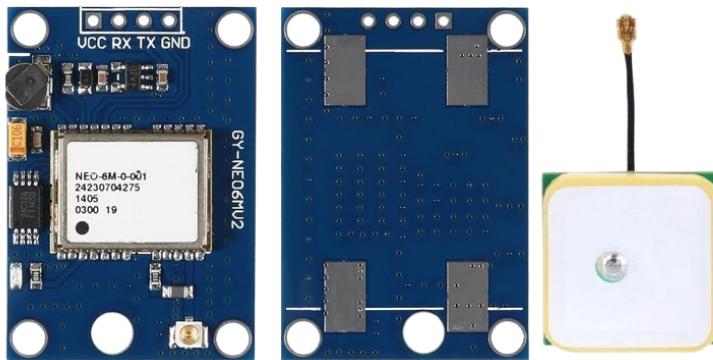
Fonte: Retirado de ON Semiconductor (2024).

### 3.3.6 Módulo GPS NEO-6M V2

O módulo GPS NEO-6M (Figura 8) fornece informações de localização (latitude, longitude e altitude). Ele foi conectado ao ESP32 utilizando os pinos GPIO 18 e 19 para TX e RX, respectivamente. No código os pinos foram configurados como RX0 e TX0 utilizando o protocolo UART para comunicação serial. Para alimentação, o módulo foi também conectado ao pino 3V3 e ao GND do ESP32.

O módulo GPS NEO-6M foi escolhido por sua capacidade de obtenção de coordenadas geográficas (U-BLOX, 2011). Essa funcionalidade permite a correlação de dados meteorológicos com a posição geográfica, facilitando a análise de padrões climáticos (ZHANG; KIMBALL; RUNNING, 2016). O NEO-6M oferece uma precisão de até 2.5 metros, adequada para aplicações que requerem um rastreio preciso da localização (U-BLOX, 2011). Na Tabela 12 são apresentadas as especificações técnicas do módulo.

Figura 8 – Módulo GPS NEO-6M V2



Fonte: Retirado de u-blox (2011).

Tabela 12 – Especificações técnicas do Módulo GPS NEO-6M V2

Parâmetro	Detalhes
Faixa de operação	2.7 V a 3.6 V
Precisão	2.5 m (GPS), 2.0 m (SBAS)
Resolução/Sensibilidade	-161 dBm
Período de amostragem	Até 5 Hz
Dimensões	16x12.2x2.4 mm

Fonte: Retirado de u-blox (2011).

### 3.3.7 Módulo Conversor RS485 para TTL

O módulo conversor RS485 para TTL (Figura 9) é utilizado para conectar o ESP32 ao anemômetro RS-FSJT-N01, convertendo os sinais TTL para níveis compatíveis com RS485 (PCB POWER MARKET, 2024). Ele foi conectado ao ESP32 usando os pinos GPIO 16 e 17 (TX e RX), que eram ligados aos pinos RXD e TXD do módulo, respectivamente. Já o barramento RS485 do anemômetro conectado aos terminais A e B do módulo, e a alimentação do mesmo foi feita por uma fonte de 12V genérica. No código, os pinos foram configurados como `MAX485_RX` e `MAX485_TX` para a comunicação serial. Na Tabela 13 são apresentadas as especificações técnicas do módulo.

Figura 9 – Módulo Conversor RS485 para TTL



Fonte: Retirado de PCB Power Market (2024).

Tabela 13 – Especificações técnicas do Módulo Conversor RS485 para TTL

Parâmetro	Detalhes
Faixa de operação	5 V
Dimensões	44x14 mm

Fonte: Retirado de PCB Power Market (2024).

### 3.3.8 Custos dos Componentes

Os custos dos componentes utilizados no desenvolvimento desse sistema podem variar de acordo com o fornecedor, região de compra e período de aquisição. A Tabela 14 apresenta os custos dos componentes utilizados, já com os impostos e taxas de importação inclusos. Os componentes foram comprados através da loja virtual *AliExpress* no dia 17/06/2024.

Tabela 14 – Custos dos componentes utilizados no sistema

Componente	Custo (R\$)
ESP32	20,19
DHT22	7,66
BMP280	4,65
BH1750	6,04
Anemômetro RS-FSJT-N01	100,43
LM393	4,64
NEO-6M	22,89
Conversor RS485 para TTL	10,86
<b>Total</b>	<b>177,36</b>

Fonte: Elaborado pelo autor, 2024.

## 3.4 Integração dos sensores ao ESP32

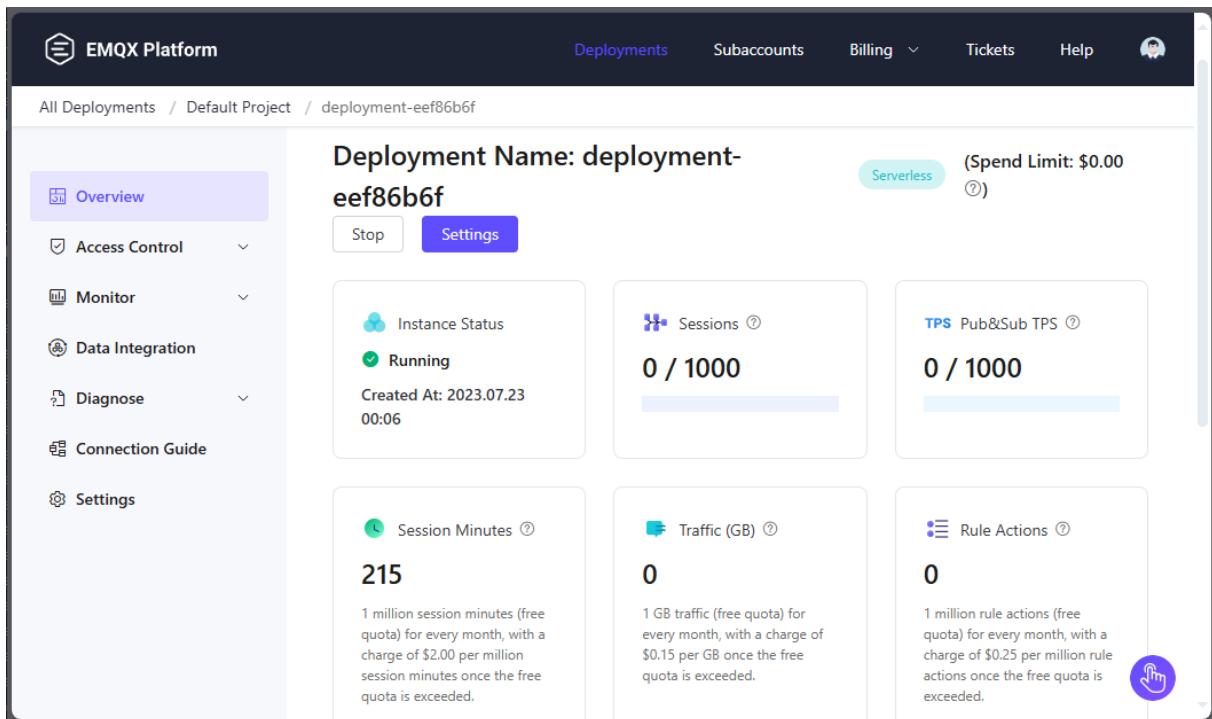
Como foram utilizados sensores para capturar os dados, é importante destacar que todo instrumento de medição possui erros inerentes (VUOLO, 1996), que podem ser classificados em dois tipos:

- Erros sistemáticos, sendo estes previsíveis e reproduzíveis. Normalmente são originados de falhas no próprio instrumento de medição ou de influências ambientais que afetam de maneira constante o resultado. Eles podem ser corrigidos através de calibração e ajustes do sistema;
- Erros estatísticos, que, diferente dos erros sistemáticos, ocorrem de forma aleatória e imprevisível. São geralmente causados por flutuações nas condições de medição, como ruído no sinal ou variações no ambiente. Eles não podem ser eliminados por calibração, mas podem ser reduzidos através da repetição e tratamento estatístico dos dados.

Nesse trabalho, cada sensor passou um teste individual de leitura por um período de 24 horas. Tanto antes quanto depois dos testes, os sensores mantiveram sua operação normal e dentro do esperado de suas especificações técnicas. Isso foi feito a fim de observar e minimizar possíveis erros sistemáticos e garantir o funcionamento ininterrupto dos sensores.

Por fim, os componentes foram integrados em um sistema cyberfísico que publica as leituras individuais de cada sensor em um *broker*. Nesse trabalho optou-se pelo EMQX (Figura 10), uma plataforma de código aberto para MQTT (EMQ TECHNOLOGIES INC., 2024).

Figura 10 – Interface da plataforma EMQX



Fonte: Elaborado pelo autor, 2024.

Nele as mensagens são distribuídas em tópicos usando protocolos de comunicação (MQTT) e criptografia (TLS). Isso garante a integridade e a confidencialidade dos

dados coletados, além do acesso remoto e em tempo real às leituras. Este acesso é feito através da conexão entre o *broker* e o *backend* do sistema, que se inscreve nos tópicos de interesse para receber os dados. Também é feita uma segunda instância de armazenamento em um banco de dados relacional (*MySQL*) para consultas de interesse.

O armazenamento dos dados históricos no *MySQL* é realizado através da funcionalidade de *Rule Engine* do EMQX, que permite criar regras para processar as mensagens recebidas. Por meio dessas regras, as mensagens publicadas nos tópicos são capturadas e inseridas em uma tabela no banco (`mqtt_messages`), previamente configurada com colunas *topic*, *payload*, *qos* e *timestamp*. Esse mecanismo permite a persistência dos dados e possibilita consultas históricas para análises ou monitoramento posterior.

### 3.5 Ferramentas e Tecnologias Utilizadas

Para o desenvolvimento do sistema proposto neste trabalho, foram empregadas ferramentas e tecnologias, em que cada uma delas desempenharam um papel específico.

#### 3.5.1 Linguagem C

A linguagem de programação *C* foi utilizada para o desenvolvimento do *firmware* do ESP32. Essa escolha deve-se à sua proximidade com o *hardware*, permitindo maior controle sobre os recursos embarcados. Além disso, o *C* é suportado nativamente por diversas plataformas de desenvolvimento, o que facilita a portabilidade do código.

No desenvolvimento do *firmware*, diversas bibliotecas foram empregadas para simplificar a implementação e assegurar a confiabilidade das funcionalidades. Abaixo, são listadas as principais bibliotecas utilizadas:

- **Arduino:** Biblioteca base para desenvolvimento com a plataforma Arduino, fornecendo funções para manipulação de pinos, comunicação serial e temporização.
- **ModbusMaster:** Implementação do protocolo Modbus RTU, permitindo comunicação entre dispositivos escravos e mestres em redes industriais.
- **Wire:** Biblioteca utilizada para comunicação I2C, facilitando a integração com sensores e dispositivos externos.
- **WiFi:** Biblioteca nativa para gerenciar conexões Wi-Fi no ESP32, necessária para conectividade com redes locais e a *internet*.
- **Adafruit\_Sensor** e **Adafruit\_BMP280:** Fornecem suporte para leitura de dados ambientais de sensores como o BMP280, usado para medir pressão atmosférica e temperatura.

- **DHT e DHT\_U:** Bibliotecas dedicadas a sensores de temperatura e umidade da linha DHT (por exemplo, DHT11 ou DHT22).
- **BH1750:** Biblioteca para integração com o sensor de luz BH1750, utilizado para medições de luminosidade.
- **WiFiClientSecure:** Extensão da biblioteca Wi-Fi para conexões utilizando TLS/SSL, garantindo comunicações criptografadas.
- **PubSubClient:** Implementação do protocolo MQTT, necessário para a comunicação com o *backend*.
- **TinyGPSPlus:** Biblioteca para interpretação de dados recebidos de módulos GPS, necessário para obter informações de localização geográfica.

A combinação dessas bibliotecas permitiu a criação de um *firmware* capaz de monitorar e controlar diversos sensores e atuadores, além de garantir conectividade e comunicação com sistemas externos.

### **3.5.2 Node.js**

O ambiente de execução *Node.js* foi empregado no desenvolvimento do *backend* devido à sua capacidade de gerenciar múltiplas conexões simultâneas. *Node.js* é particularmente adequado para aplicações que exigem baixa latência, como em sistemas em tempo real.

No desenvolvimento do *backend*, foram usadas diversas bibliotecas para otimizar funcionalidades e assegurar boas práticas de programação. A seguir, são destacadas as principais dependências e suas funções no projeto:

- **Express:** *Framework* para construção de servidores *web*, usado para gerenciar rotas e requisições HTTP.
- **Axios:** Biblioteca utilizada para realizar requisições HTTP, facilitando a comunicação com APIs externas.
- **Cors:** *Middleware* que permite configurar políticas de compartilhamento de recursos entre diferentes origens, necessário para integrar o *frontend* com o *backend*.
- **Dotenv:** Biblioteca usada para gerenciar variáveis de ambiente, permitindo a configuração de credenciais e configurações do sistema.
- **MQTT:** Protocolo para comunicação em tempo real, empregado no gerenciamento de mensagens entre dispositivos conectados.

- **Zod**: Biblioteca para validação e definição de esquemas de dados, assegurando a integridade das entradas e saídas da aplicação.

Além das dependências de produção, foi utilizada uma estrutura de dependências de desenvolvimento para garantir qualidade no código e simplificar o processo:

- **TypeScript**: *Superset* do JavaScript que adiciona tipagem estática ao código, prevendo erros.
- **Jest** e **Supertest**: Ferramentas para testes automatizados, que permitem validar funcionalidades e garantir o funcionamento correto da aplicação.
- **ESLint** e **Prettier**: Configurações para impor boas práticas de codificação e formatação consistente, otimizando a legibilidade e manutenção do código.
- **Nodemon** e **ts-node**: Facilitar o desenvolvimento ao permitir recarga automática do servidor e execução de arquivos TypeScript sem necessidade de transpilação prévia.

Esse conjunto de ferramentas proporcionou um ambiente de desenvolvimento produtivo e um *backend* eficiente, escalável e seguro.

### **3.5.3    *Next.js***

A biblioteca *Next.js* foi utilizada para o desenvolvimento da interface do *frontend*, oferecendo uma arquitetura de renderização híbrida que combina renderização do lado do servidor - SSR (sigla do inglês, *server-side rendering*) e renderização do lado do cliente - CSR (sigla do inglês, *client-side rendering*). Isso permite que as páginas *web* sejam renderizadas no servidor e enviadas para o cliente, proporcionando uma experiência de usuário rápida e responsiva. A seguir, destacam-se as principais dependências empregadas no projeto e suas funcionalidades:

- **@next/font**: Utilizado para integrar fontes personalizadas diretamente no projeto, otimizando o desempenho no carregamento de fontes.
- **@radix-ui/react-dropdown-menu**,        **@radix-ui/react-scroll-area**        e  
**@radix-ui/react-slot**: Componentes acessíveis e estilizados para construção de interfaces reativas e intuitivas.
- **@tailwindcss/typography**: *Plugin* do *TailwindCSS* que adiciona estilos predefinidos para tipografia, simplificando a criação de layouts bem formatados.
- **axios**: Biblioteca utilizada para realizar requisições HTTP, integrando a interface com as APIs do *backend*.

- **class-variance-authority**: Facilita a manipulação de classes CSS com base em variantes condicionais, permitindo maior flexibilidade no estilo.
- **clsx**: Utilizado para combinar classes CSS dinamicamente de maneira concisa e legível.
- **lucide-react**: Biblioteca de ícones leves e modernos, utilizada para enriquecer a interface visual.
- **next-themes**: Fornece suporte à alternância entre temas claros e escuros, aprimorando a experiência do usuário.
- **react-icons**: Conjunto abrangente de ícones para *React*, permitindo a inclusão de gráficos contextuais no *design*.
- **react-responsive-carousel**: Componente de carrossel responsivo para exibição de imagens ou conteúdos de forma dinâmica.
- **sharp**: Biblioteca para manipulação de imagens no lado do servidor, otimizando tamanhos e formatos para diferentes dispositivos.
- **tailwind-merge**: Facilita a mesclagem de classes *TailwindCSS*, ajudando a resolver conflitos de estilo.
- **tailwindcss-animate**: *Plugin* que adiciona animações ao *TailwindCSS*, tornando a experiência do usuário mais fluida e atraente.
- **zod**: Biblioteca para validação e definição de esquemas de dados, assegurando a integridade das informações manipuladas no *frontend*.

Para garantir qualidade e consistência durante o desenvolvimento, foram empregadas diversas dependências de ambiente:

- **@types/node**, **@types/react** e **@types/react-dom**: Adicionam tipagem para as bibliotecas principais, garantindo integração com o **TypeScript**.
- **autoprefixer**: Automatiza a inclusão de prefixos de compatibilidade para navegadores, assegurando suporte abrangente ao CSS.
- **eslint** e **eslint-config-next**: Ferramentas de análise de código estático que impõem boas práticas no desenvolvimento.
- **eslint-config-prettier**: Garante integração entre o **ESLint** e o **Prettier**, evitando conflitos entre as regras.
- **postcss**: Processador CSS que melhora a compatibilidade e otimização dos estilos.

- **prettier**: Ferramenta de formatação automática, assegurando consistência visual no código.
- **tailwindcss**: *Framework* de utilitários CSS que acelera o desenvolvimento de estilos responsivos e customizados.
- **typescript**: Adiciona tipagem estática ao JavaScript, prevenindo erros comuns e proporcionando maior robustez no código.

Esse conjunto de ferramentas e dependências permitiu o desenvolvimento de um *frontend* moderno, dinâmico e escalável, com foco na experiência do usuário e na eficiência do sistema.

#### **3.5.4 Jest**

O *framework* de testes *Jest* foi adotado para a realização de testes automatizados. Essa ferramenta foi escolhida por sua simplicidade de configuração e pela capacidade de executar testes unitários e de integração, garantindo a qualidade e a confiabilidade do código, além de facilitar a manutenção e evolução do sistema.

A estrutura dos testes foi organizada em dois principais níveis, separados por pastas para maior clareza e modularidade:

- Ponta a ponta: Contém testes que verificam o funcionamento completo da aplicação em cenários reais, cobrindo desde as requisições feitas ao servidor até as respostas entregues. No sistema implementado nessa pesquisa, o teste avalia o comportamento do sistema como um todo, validando o fluxo de dados entre os componentes.
- Unitários: Reúne testes focados em unidades específicas do sistema, como funções, serviços ou classes individuais. Cada teste nessa pasta avalia o comportamento isolado de componentes-chave, garantindo que funcionem conforme o esperado. Nessa pesquisa, os testes unitários foram utilizados para validar a lógica de negócios e o transporte dos dados entre as camadas do sistema.

Essa organização entre testes de integração e testes unitários permite isolar falhas e assegurar que tanto os componentes individuais quanto a aplicação como um todo funcionem de forma consistente e modular.

#### **3.5.5 Shadcn/ui**

A biblioteca de componentes *Shadcn/ui* foi utilizada no desenvolvimento do *frontend*, proporcionando componentes reutilizáveis e estilizados que aceleraram o processo

de construção da interface. Essa biblioteca foi selecionada por sua flexibilidade e integração nativa com o *framework TailwindCSS*. Nesse trabalho foram usados os seguintes componentes:

- **Button**: Botão interativo com estilos personalizáveis e animações integradas.
- **Badge**: Indicador visual para destacar informações importantes.
- **Card**: Cartão de conteúdo com layout flexível e responsivo.
- **Dropdown-menu**: Menu suspenso com opções personalizáveis e animações suaves.
- **Scroll-area**: Área de rolagem personalizada com suporte a conteúdo dinâmico.
- **Skeleton**: Efeito de carregamento para indicar que o conteúdo está sendo carregado.

### **3.5.6 TailwindCSS**

O *framework TailwindCSS* foi empregado na estilização do *frontend*. Sua abordagem utilitária permite a criação de interfaces consistentes e bem estilizadas de maneira rápida e eficiente, o que foi importante para garantir um *design* responsivo e atrativo. Nesse trabalho, foram utilizadas paletas de cores voltadas para o uso em ambientes externos, com tons de verde e branco destacados.

### **3.5.7 Figma**

A ferramenta Figma (Figura 11) foi empregada na prototipagem de interfaces gráficas. Sua escolha foi motivada pela variedade de recursos disponíveis para o *design* e validação de interfaces antes de sua implementação.

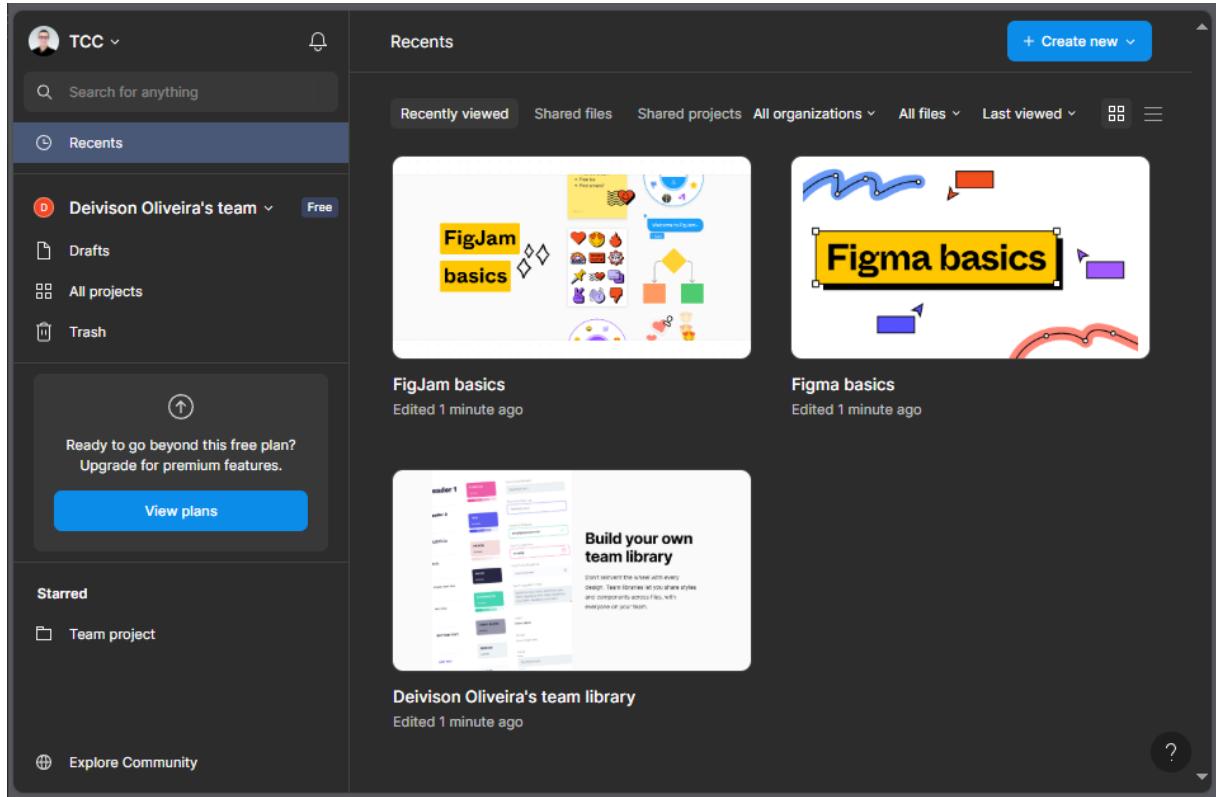
### **3.5.8 Visual Studio Code**

O Visual Studio Code (Figura 12) foi utilizado como ambiente de desenvolvimento integrado - IDE (sigla do inglês, *Integrated Development Environment*) para o desenvolvimento do *frontend* e do *backend*. Essa ferramenta foi escolhida por sua versatilidade, extensibilidade e suporte a diversas linguagens e tecnologias, incluindo as mencionadas neste trabalho.

### **3.5.9 Arduino IDE**

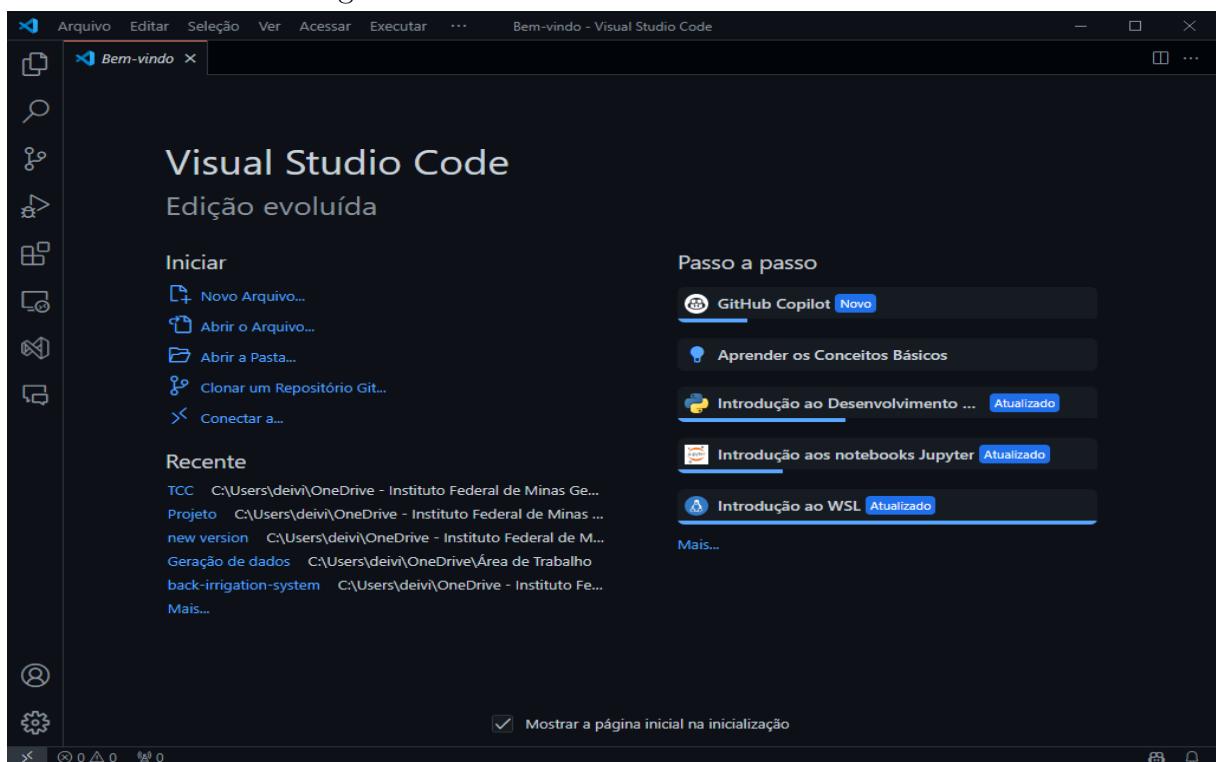
A Arduino IDE (Figura 13) foi utilizada como IDE para o desenvolvimento embarcado no ESP32. Essa escolha se justifica pela usabilidade da ferramenta e pela compatibilidade com a plataforma ESP32, facilitando a criação e a depuração do *firmware*.

Figura 11 – Interface da aplicação Figma



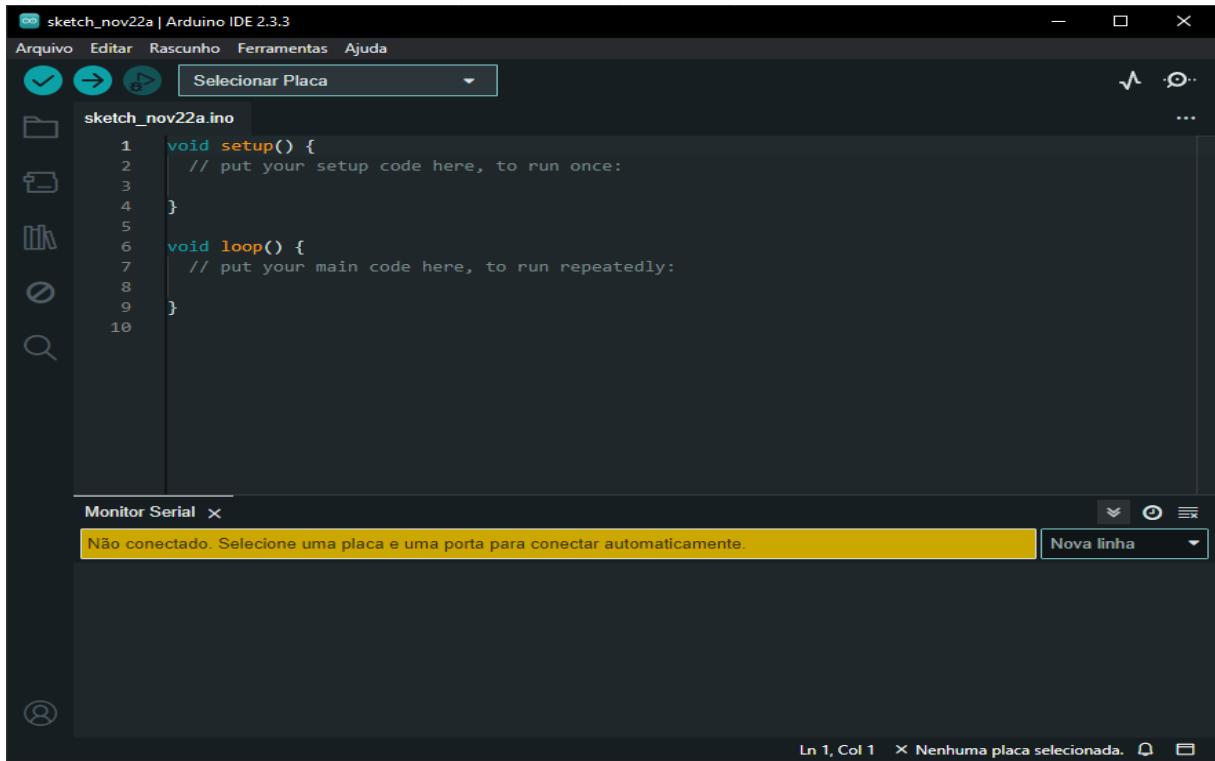
Fonte: Elaborado pelo autor, 2024.

Figura 12 – Interface do Visual Studio Code



Fonte: Elaborado pelo autor, 2024.

Figura 13 – Interface do Arduino IDE



Fonte: Elaborado pelo autor, 2024.

### **3.5.10 Fritzing**

O Fritzing (Figura 14) foi empregado na prototipagem de circuitos eletrônicos. Essa ferramenta permite a criação de esquemas e *layouts* de circuitos de maneira intuitiva, sendo necessária para a validação do *hardware* antes de sua implementação definitiva.

### **3.5.11 EasyEDA**

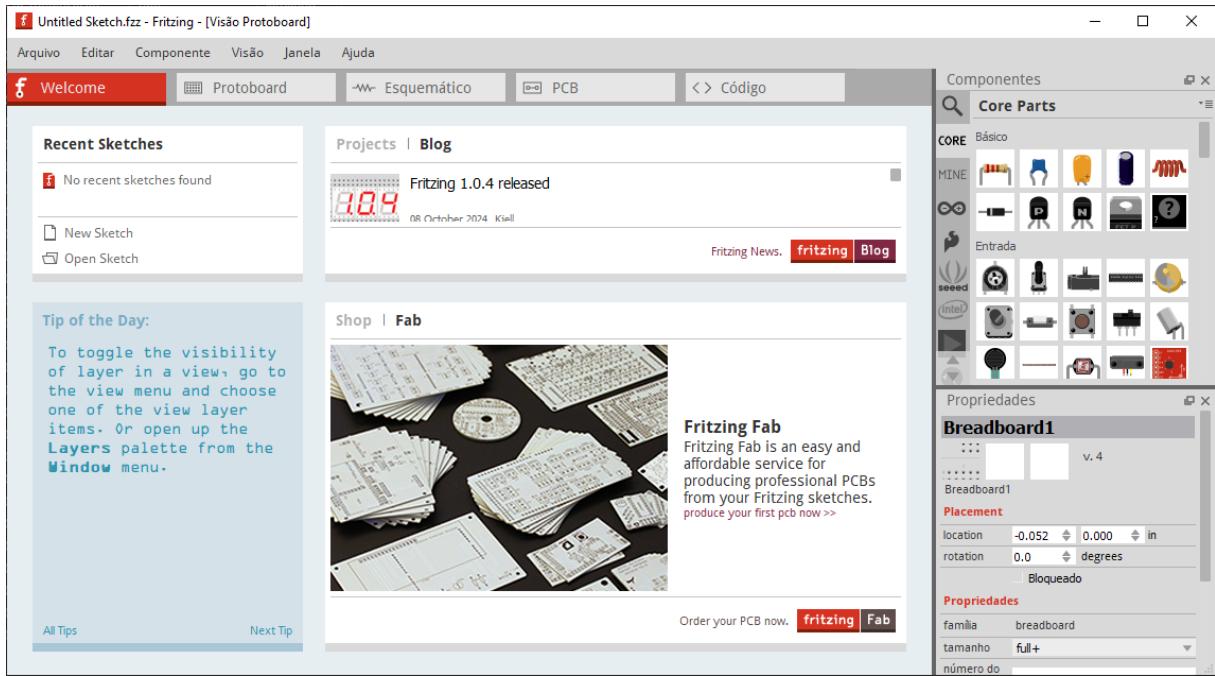
A ferramenta EasyEDA (Figura 15) foi utilizada para o desenho das placas de circuito impresso - PCB (sigla do inglês, *Printed Circuit Board*). Sua escolha foi motivada pela facilidade de uso e pelos recursos para simulação e validação de projetos eletrônicos, garantindo a qualidade e a confiabilidade das placas desenvolvidas.

### **3.5.12 Resumo das escolhas**

As tecnologias e ferramentas mencionadas foram selecionadas para atender aos requisitos específicos do sistema proposto, priorizando desempenho, escalabilidade e experiência prévia de desenvolvimento.

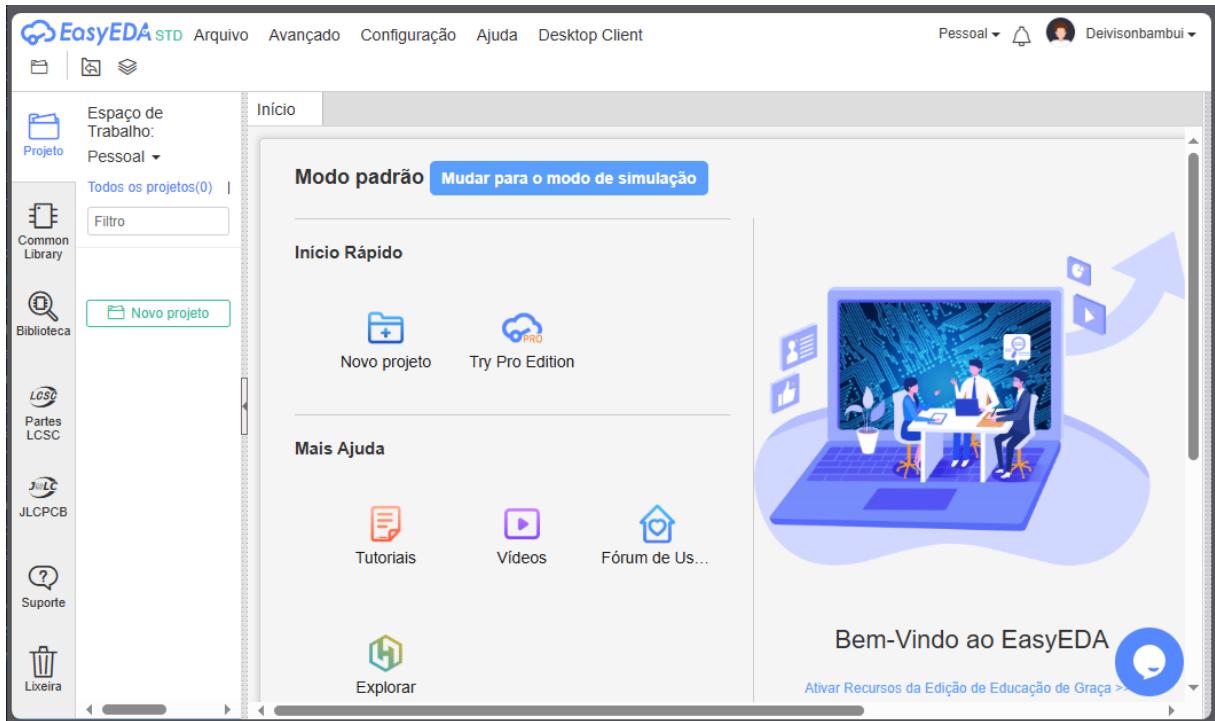
A análise dos dados meteorológicos é realizada de forma contínua para apresentar medições e cálculos estimativos de ET<sub>0</sub> e ET<sub>c</sub> locais. Esse método de análise permite

Figura 14 – Interface da aplicação Fritzing



Fonte: Elaborado pelo autor, 2024.

Figura 15 – Interface da aplicação EasyEDA



Fonte: Elaborado pelo autor, 2024.

a interpretação e avaliação dos dados de forma rápida, além de gerar relatórios periódicos para a tomada de decisões.

### 3.6 Estimativa da evapotranspiração

O cálculo da ETo e da ETc é usada para o monitoramento hídrico em tempo real e é realizado com base nas leituras dos sensores. Foram usadas as equações da FAO-56 apresentadas por Allen *et al.* (1998) como modelagem referencial. Cada leitura dos sensores contribui diretamente para o cálculo dessas variáveis, ou, é convertida de forma a ser utilizada na equação.

#### 3.6.1 Variáveis meteorológicas e sua obtenção

As variáveis meteorológicas utilizadas no cálculo da evapotranspiração são obtidas da seguinte forma:

- A temperatura do ar ( $T$ ) é medida diretamente do sensor DHT22, que fornece a temperatura em graus Celsius;
- A umidade relativa ( $RH$ ) também é medida pelo sensor DHT22, em percentual;
- A umidade do solo ( $h$ ) é medida pelo comparador LM393 com eletrodos externos, em um intervalo analógico;
- A pressão atmosférica ( $P$ ) é obtida do sensor BMP280, em hPa;
- A velocidade do vento ( $u_2$ ) é medida pelo anemômetro, em m/s.

As demais variáveis são estimadas a partir de cálculos intermediários, obtidos em sua maioria da obra de Allen *et al.* (1998):

- A radiação líquida ( $R_n$ ) é estimada a partir das leituras do sensor de luminosidade BH1750, que mede a intensidade luminosa em lux a cada segundo.
  - Michael, Johnston e Moreno (2020) apresentam a conversão de lux ( $lx$ ) para irradiação solar ( $E$ ) em  $\text{W/m}^2$ , dada pela equação:

$$E \approx \frac{lx}{120}$$

- A irradiação solar por sua vez é a própria em radiação solar ( $R_s$ ):

$$R_s = E$$

- Desta forma é possível determinar a radiação líquida solar ( $R_{ns}$ ), dada por:

$$R_{ns} = 0.77 \cdot R_s$$

- A última componente necessária para determinar a  $R_n$  é a radiação líquida de ondas longas ( $R_{nl}$ ), que é resultante das seguintes equações:

- \* A distância relativa inversa Terra-Sol ( $d_r$ ), que é calculada como:

$$d_r = 1 + 0.033 \cos\left(\frac{2\pi J}{365}\right)$$

onde  $J$  é o dia do ano.

- \* A radiação extraterrestre por períodos horários ou mais curtos ( $R_a$ ), dada por:

$$R_a = \frac{12 \cdot 60}{\pi} G_{sc} d_r [(\omega_2 - \omega_1) \sin(\phi) \sin(\delta) + \cos(\phi) \cos(\delta) (\sin(\omega_2) - \sin(\omega_1))]$$

onde  $G_{sc} = 0.0820 \text{ MJ m}^{-2} \text{ min}^{-1}$ ,  $\delta = 0.409 \sin\left(\frac{2\pi J}{365} - 1.39\right)$ ,  $\phi$  é o grau de latitude (obtido pelo módulo GPS NEO-6M V2), e os ângulos  $\omega_1$  e  $\omega_2$  são os ângulos de tempo solar no início e no final do período, respectivamente.

- \* A radiação solar com céu limpo ( $R_{so}$ ), dada por:

$$R_{so} = (0.75 + 2 \cdot 10^{-5} z) R_a$$

onde  $z$  é a elevação em metros do nível do mar.

- \* A radiação líquida de ondas longas ( $R_{nl}$ ), estimada por:

$$R_{nl} = \sigma \left( \frac{T_{max,K}^4 + T_{min,K}^4}{2} \right) [0.34 - 0.14\sqrt{e_a}] \left( 1.35 \frac{R_s}{R_{so}} - 0.35 \right)$$

onde  $T_{max,K}$  e  $T_{min,K}$  são as temperaturas máximas e mínimas em Kelvin,  $\sigma$  é a constante de Stefan-Boltzmann ( $4.903 \cdot 10^{-9} \text{ MJ K}^{-4} \text{ m}^{-2} \text{ dia}^{-1}$ ), e  $e_a$  é a pressão de vapor atual.

- A  $R_n$  resultante é dada por:

$$R_n = R_{ns} - R_{nl}$$

- A variação da pressão de saturação ( $\Delta$ ) é calculada com a equação:

$$\Delta = \frac{4098 \cdot (0.6108 \cdot e^{\frac{17.27 \cdot T}{T+237.3}})}{(T + 237.3)^2}$$

- A pressão de saturação ( $e_s$ ) é calculada usando:

$$e_s = \frac{0.6108 \cdot e^{\frac{17.27 \cdot T_{max}}{T_{max}+237.3}} + 0.6108 \cdot e^{\frac{17.27 \cdot T_{min}}{T_{min}+237.3}}}{2}$$

- A pressão de vapor atual ( $e_a$ ) é obtida com:

$$e_a = 0.6108 \cdot e^{\frac{17.27 \cdot T}{T+237.3}}$$

- O fluxo de calor do solo ( $G$ ) é definido pela hora, ou seja, considera-se:  $G = 0.1 \cdot R_n$  durante o dia e  $G = 0.5 \cdot R_n$  durante a noite;
- A constante psicrométrica ( $\gamma$ ) é dada por:

$$\gamma = 0.665 \cdot 10^{-3} \cdot P$$

onde  $P$  é a pressão atmosférica, fornecida pelo sensor BMP280.

No sistema proposto, cada variável listada acima, seja ela medida diretamente ou calculada, é apresentada no painel da aplicação *web*. Variáveis obtidas diretamente são aquelas cujos sensores tem sua leitura publicada em tópicos MQTT. Já as calculadas são obtidas a partir de manipulações dessas leituras. Isso é feito por um dos *use-cases* do *backend*, que é responsável por calcular e disponibilizar as variáveis em *endpoints* que são consumidos pelo *frontend*. Também há um tópico de MQTT para publicação de erros e alertas, que são gerados a partir de condições específicas. Os erros cobertos incluem valores fora de faixa de leitura esperada, falhas de comunicação (tanto Wi-fi como no *broker* MQTT), ou falhas de *hardware* do ESP32.

### 3.7 Desenvolvimento da aplicação

Nessa subsessão, são apresentados os detalhes do desenvolvimento da aplicação, incluindo a arquitetura do sistema e os métodos de prototipagem utilizados.

#### 3.7.1 Padrões arquiteturais e práticas utilizadas

No desenvolvimento do sistema, foram adotados padrões arquiteturais para garantir a modularidade, escalabilidade e manutenção do código. A arquitetura foi baseada nos princípios de *Domain-Driven Design* (DDD) e nos cinco princípios SOLID (EVANS, 2004).

O DDD foi escolhido como base para a estruturação do sistema devido à sua capacidade de refletir a complexidade do domínio meteorológico no código. A aplicação foi dividida em camadas como *domain*, *application*, *infrastructure*, e *interfaces*, em que cada uma possui responsabilidades bem definidas. Isso facilita a evolução e adaptação do sistema a novos requisitos.

Os princípios SOLID foram aplicados para garantir um código coeso, desacoplado e extensível. Especificamente:

- O princípio da responsabilidade única (SRP) foi adotado para garantir que cada classe ou módulo tenha uma única responsabilidade, facilitando a manutenção e evolução do código;

- O princípio aberto/fechado (OCP) assegurou que os componentes do sistema fossem abertos para extensão, mas fechados para modificação, permitindo a adição de novas funcionalidades sem alterar o código existente;
- O princípio da substituição de Liskov (LSP) foi utilizado para garantir que as classes derivadas pudessem substituir suas classes base sem alterar o comportamento do sistema;
- O princípio da segregação de interfaces (ISP) assegurou que as interfaces fossem específicas para cada tipo de cliente, evitando a implementação de métodos desnecessários;
- O princípio da inversão de dependência (DIP) foi seguido para garantir que módulos de alto nível não dependessem de módulos de baixo nível, mas sim de abstrações, promovendo o desacoplamento do código.

Essas práticas resultaram em um código mais limpo, testável e resiliente, capaz de suportar a evolução contínua do sistema sem comprometer sua integridade.

### **3.7.2 Ciclo de desenvolvimento do software**

O ciclo de desenvolvimento do *software* envolveu tanto a parte *web* quanto a embarcada, seguindo uma abordagem iterativa e incremental. Foi utilizado o *Git Workflow* como estratégia principal de controle de versões (CHACON; STRAUB, 2024).

Para o desenvolvimento *web*, o fluxo começou com a definição dos requisitos e a modelagem do domínio, seguida pela implementação das funcionalidades em *branches* específicas, alinhadas com as funcionalidades do sistema. Cada *feature branch* era integrada à *branch develop* após a aprovação em revisões de código e a passagem pelos testes automatizados, garantindo a qualidade e a integração contínua do código. O *Git Workflow* utilizado incluía a criação de *feature branches*, *hotfix branches*, e *release branches*, de acordo com o estado de desenvolvimento e manutenção do sistema.

Para o desenvolvimento embarcado, foram utilizados ciclos mais curtos devido à necessidade de validação rápida dos sensores e componentes físicos. Cada ciclo incluía a configuração do *hardware*, programação do *firmware* no ESP32 utilizando a Arduino IDE, e testes físicos para validação dos dados coletados. As versões do código embarcado eram gerenciadas em um repositório *Git* separado, seguindo uma estrutura similar à usada para o desenvolvimento *web*. Nesse caso, foram feitas adaptações para as necessidades específicas do desenvolvimento embarcado, incluindo a verificação de integrações entre sensores e o envio dos dados ao *broker* EMQX.

A sinergia entre essas duas frentes de desenvolvimento foi usada para garantir que o sistema operasse de forma coesa e integrada, desde a captura dos dados até a sua visualização na interface *web*.

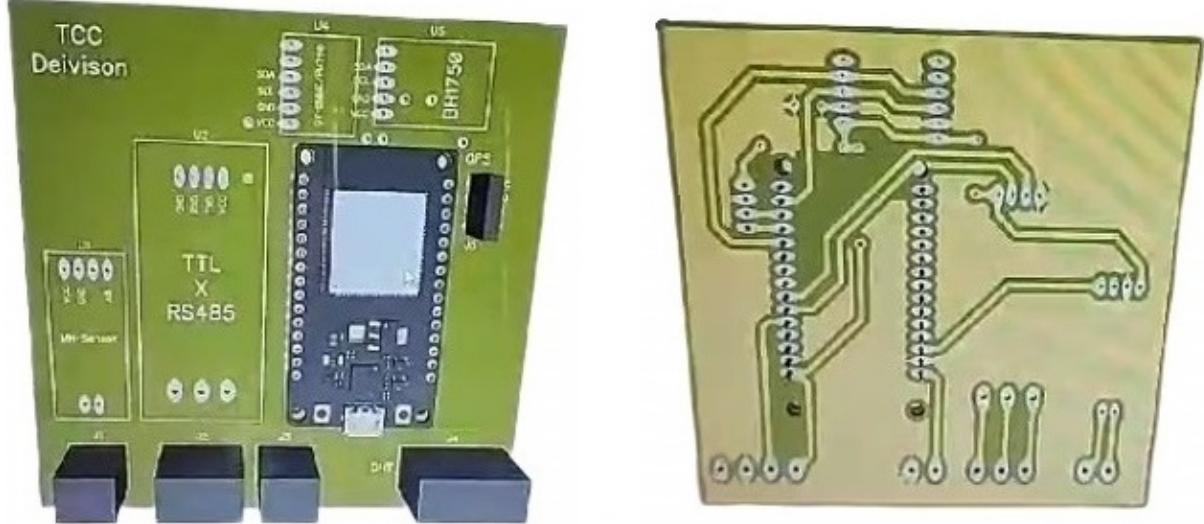
### **3.7.3 Prototipagem e distribuição dos componentes**

A prototipagem do sistema foi uma etapa importante para garantir que os componentes de *hardware* e *software* funcionassem harmoniosamente. O processo começou com a criação de protótipos funcionais simples, utilizando o *Fritzing* para o *design* dos circuitos e a prototipagem das interações entre os sensores e o microcontrolador ESP32.

O desenvolvimento do protótipo envolveu a montagem dos sensores em uma *breadboard*, permitindo ajustes rápidos e trocas de componentes conforme necessário. O *layout* dos componentes foi projetado no EasyEDA de forma a otimizar a coleta de dados, minimizando interferências e garantindo a precisão das medições (Figura 16).

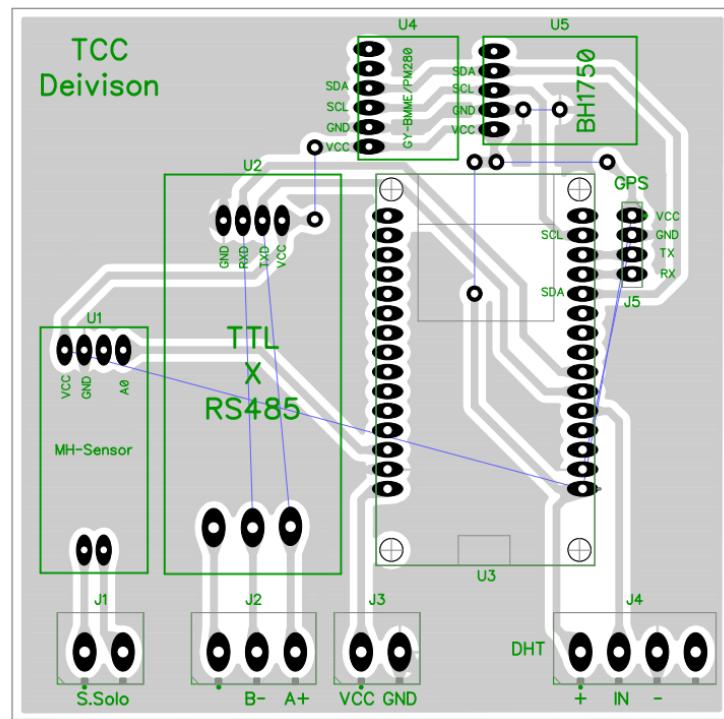
Após a validação do protótipo inicial, os componentes foram instalados em uma montagem permanente. Isso foi feito utilizando uma PCB e cabos para alimentação e comunicação com sensores externos. A PCB foi usinada com base no *layout* exportado do EasyEDA (Figura 17), sendo feita uma matriz para as trilhas e furos necessários (Figura 18). A usinagem foi feita usando percloreto de ferro sobre uma placa de fenolite, tendo a matriz impressa sobre a placa e a corrosão feita com o percloreto (Figura 19). Após a corrosão, a placa foi limpa e os furos foram feitos com uma broca de 1mm (Figura 20).

Figura 16 – Layout do protótipo



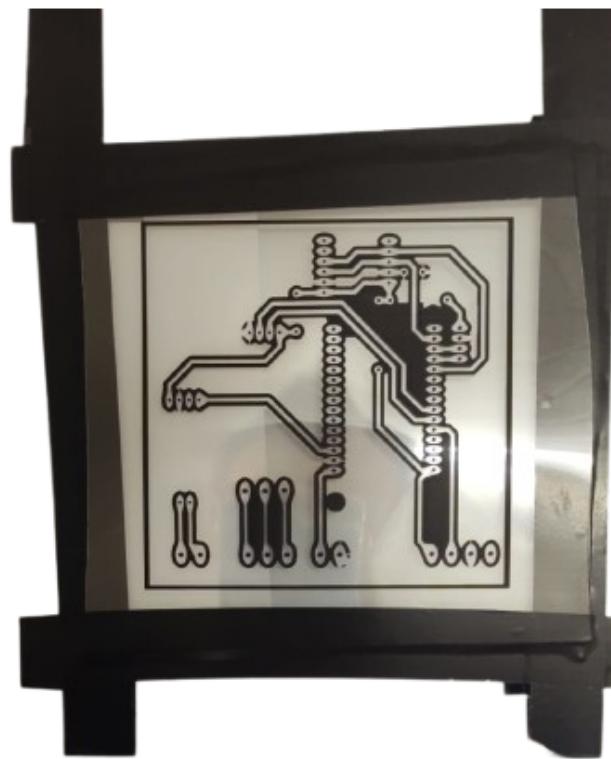
Fonte: Elaborado pelo autor, 2024.

Figura 17 – Matriz exportada do EasyEDA



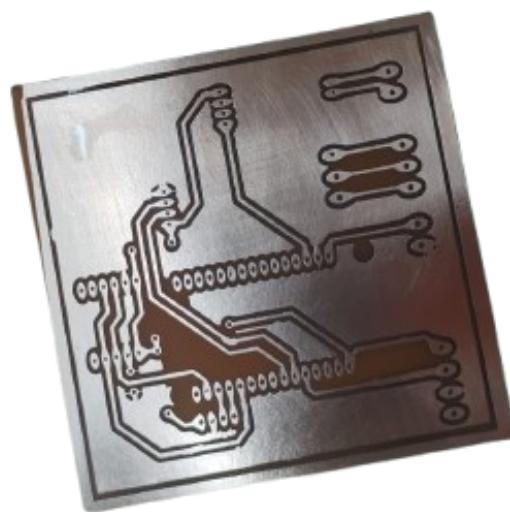
Fonte: Elaborado pelo autor, 2024.

Figura 18 – Matriz impressa em papel filme



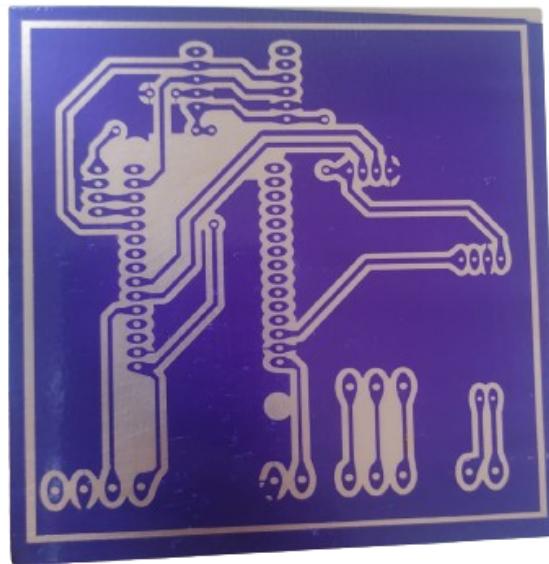
Fonte: Elaborado pelo autor, 2024.

Figura 19 – PCB finalizada



Fonte: Elaborado pelo autor, 2024.

Figura 20 – PCB pós corrosão



Fonte: Elaborado pelo autor, 2024.

Para a prototipagem do *software*, foram utilizados *mockups* e *wireframes* para definir as interfaces e fluxos de usuário. Ferramentas como *Figma* foram empregadas para criar protótipos das telas, que serviram de guia para o desenvolvimento do *frontend* em *Next.js*. A distribuição dos componentes de *software* seguiu o princípio de modularidade, permitindo que cada módulo fosse desenvolvido e testado isoladamente antes de ser integrado ao sistema principal.

A metodologia adotada permitiu uma transição suave do protótipo para a implementação final, garantindo que todos os componentes do sistema fossem adequadamente validados e integrados antes da implantação no laboratório.

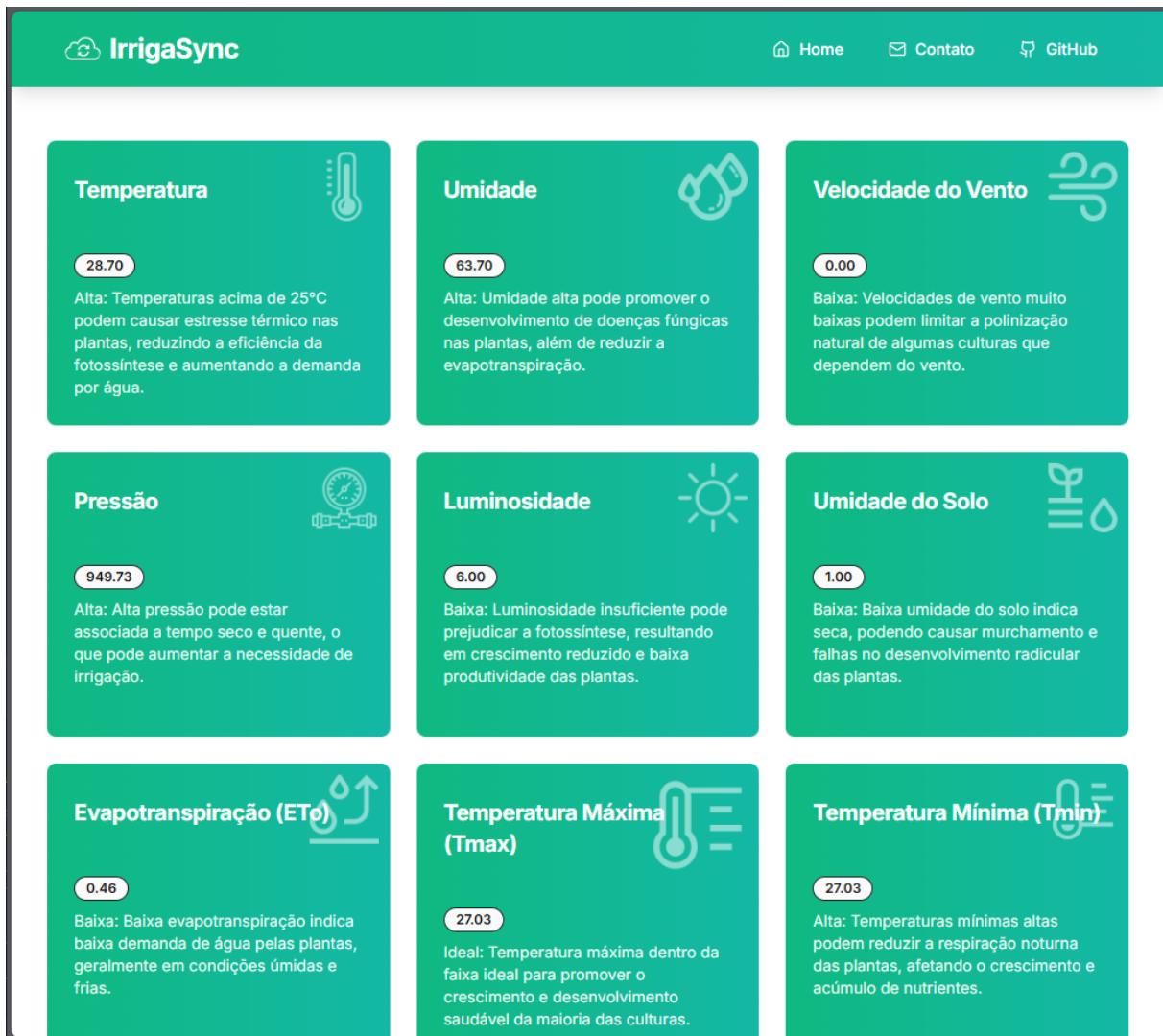
## 4 RESULTADOS

Nesta seção, são apresentados os resultados obtidos com o sistema desenvolvido, incluindo capturas das interfaces e do *hardware* implementado. Também são apresentados gráficos dos dados coletados e armazenados pela aplicação, nomeada de *IrrigaSync*.

### 4.1 Interfaces do sistema *IrrigaSync*

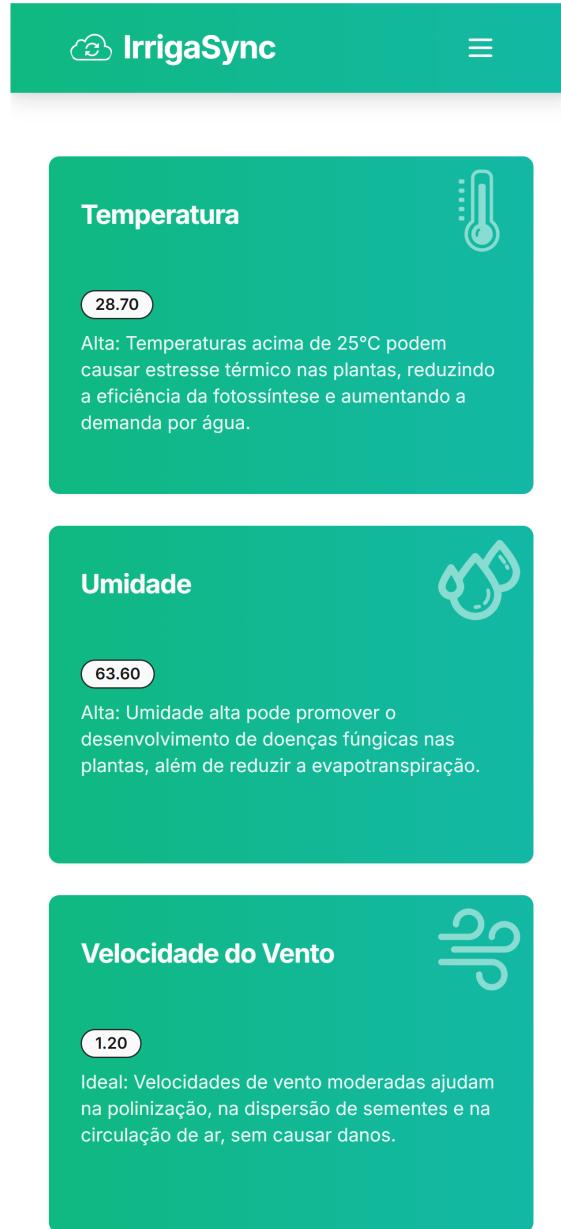
Nas Figuras 21 e 22, são apresentadas imagens da interface do sistema *IrrigaSync* em dispositivos *desktop* e móveis, respectivamente, já na versão final de lançamento (*release branch*).

Figura 21 – *IrrigaSync desktop*



Fonte: Elaborado pelo autor, 2024.

Figura 22 – *IrrigaSync mobile*



Fonte: Elaborado pelo autor, 2024.

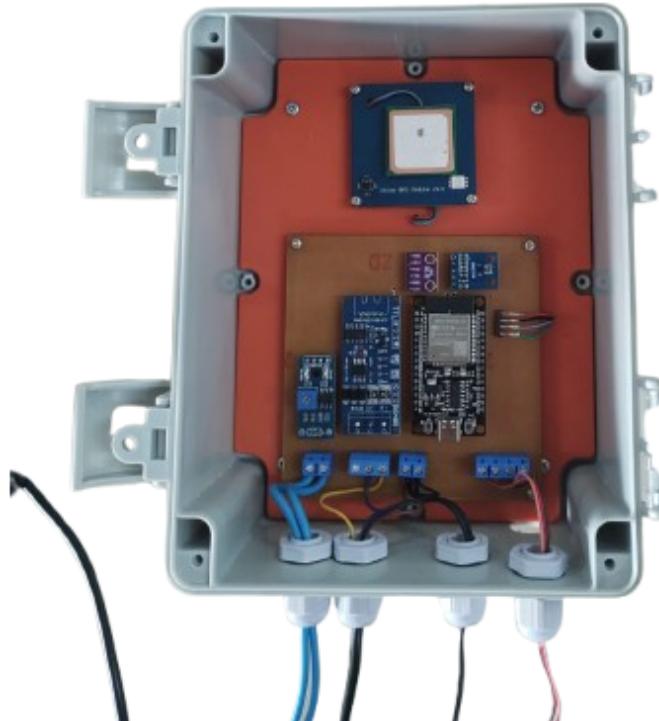
#### 4.2 Montagem do *hardware*

Para ilustrar o processo de montagem do *hardware*, a Figura 23 apresenta o gabinete contendo a PCB instalada e os componentes integrados. Essa montagem final foi projetada para oferecer proteção contra intempéries e garantir a estabilidade do sistema.

#### 4.3 Dados coletados

Os dados meteorológicos capturados diretamente pelo sistema incluem temperatura, umidade relativa do ar, velocidade do vento, luminosidade e pressão atmosférica.

Figura 23 – Gabinete com PCB instalado



Fonte: Elaborado pelo autor, 2024.

A seguir, estão as descrições dos principais resultados obtidos para cada variável durante um dia no local de implementação (16/10/2024).

#### **4.3.1 Temperatura e umidade relativa do ar**

Os valores de temperatura e umidade relativa foram obtidos a partir do sensor DHT22. A temperatura variou entre 19°C e 30°C, enquanto a umidade relativa oscilou entre 40% e 85%. Os gráficos da Figura 24 e Figura 25 apresentam a evolução desses parâmetros ao longo de um período de 12 horas.

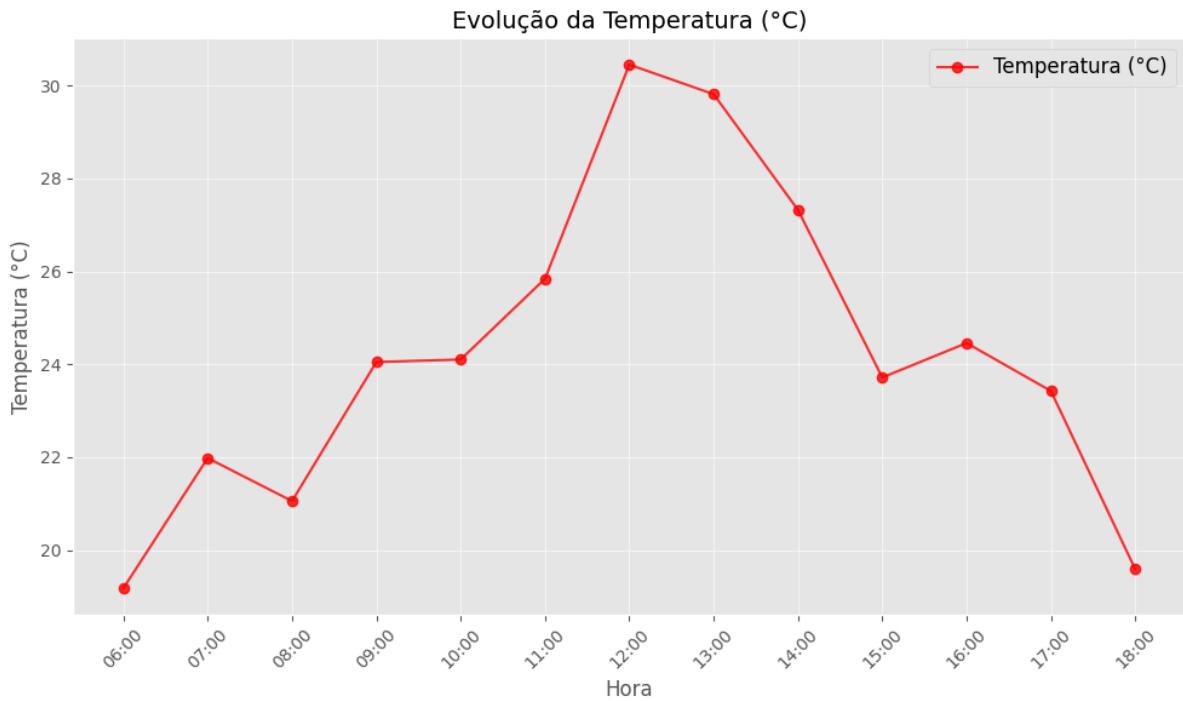
#### **4.3.2 Velocidade do vento**

As medições de velocidade do vento, capturadas pelo anemômetro RS-FSJT-N01, mostraram uma variação entre 0.5 m/s e 12 m/s. A Figura 26 ilustra a distribuição das velocidades ao longo de 12 horas.

#### **4.3.3 Luminosidade**

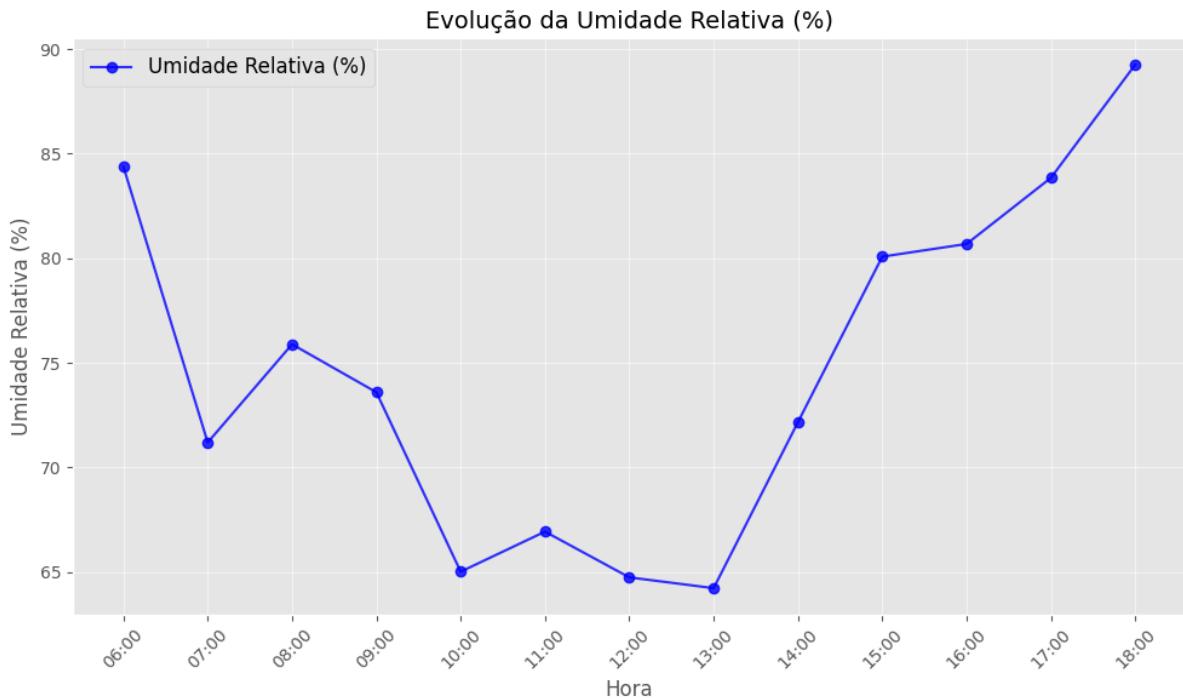
A intensidade luminosa medida pelo sensor BH1750 variou entre 400 lux e 25000 lux, refletindo mudanças na radiação solar ao longo do dia. A Figura 27 mostra a evolução da luminosidade em função do tempo.

Figura 24 – Evolução da temperatura em 12 horas (06:00 - 18:00)



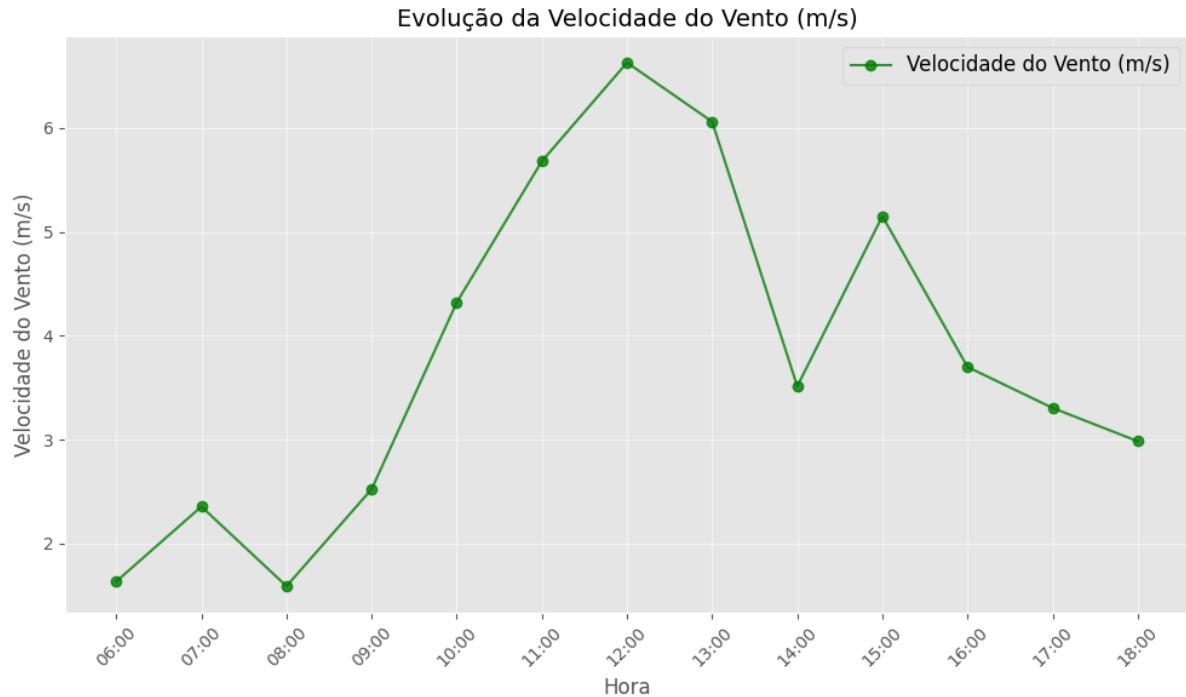
Fonte: Elaborado pelo autor, 2024.

Figura 25 – Evolução da umidade em 12 horas (06:00 - 18:00)



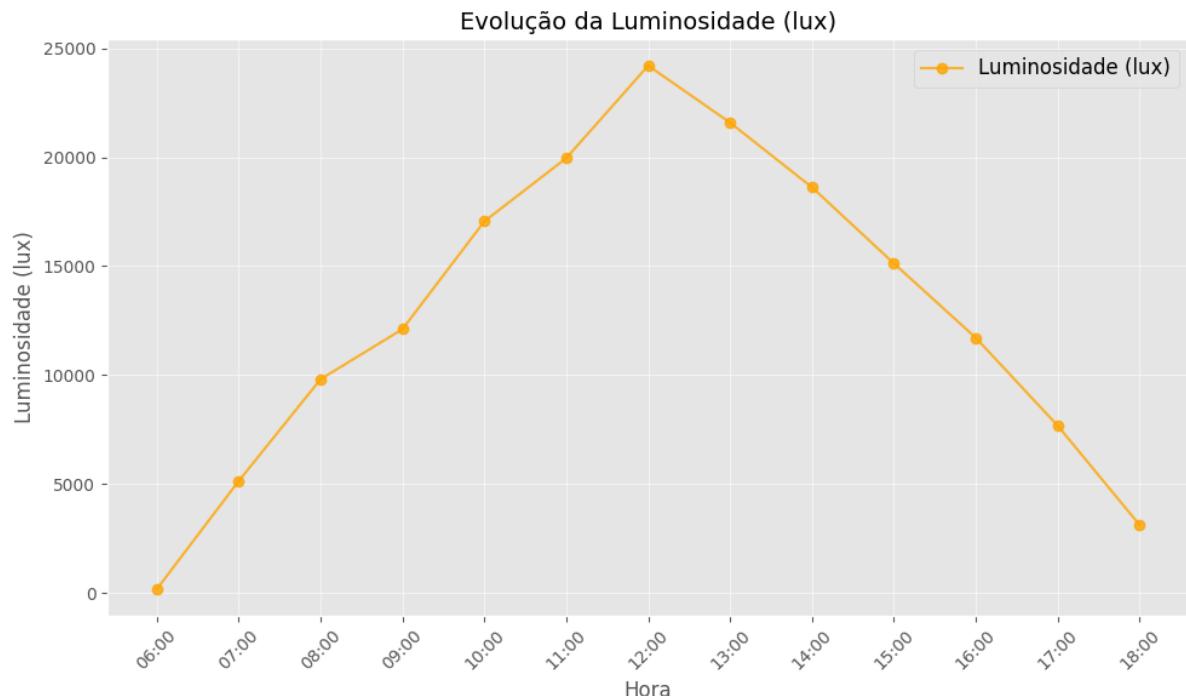
Fonte: Elaborado pelo autor, 2024.

Figura 26 – Evolução da velocidade do vento em 12 horas (06:00 - 18:00)



Fonte: Elaborado pelo autor, 2024.

Figura 27 – Evolução da luminosidade em 12 horas (06:00 - 18:00)



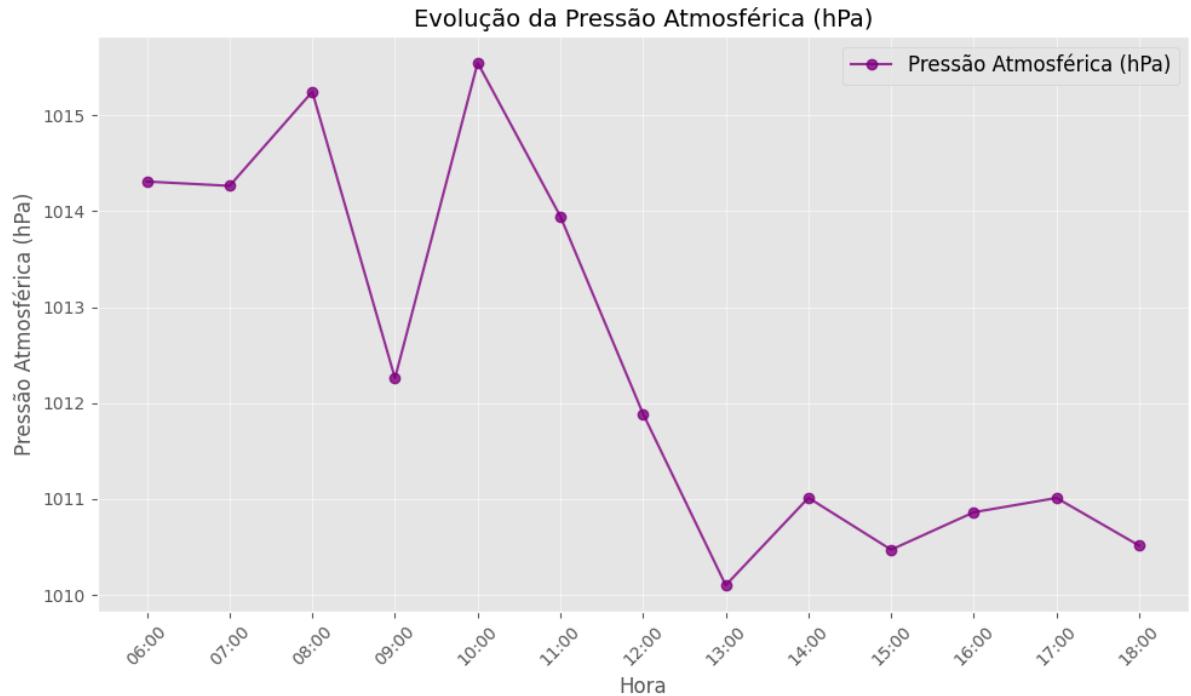
Fonte: Elaborado pelo autor, 2024.

#### 4.3.4 Pressão atmosférica

O sensor BMP280 registrou valores de pressão atmosférica entre 950 hPa e 1020 hPa, refletindo as variações da região de Bambuí. A evolução da pressão ao longo do

tempo é mostrada na Figura 28.

Figura 28 – Evolução das variáveis meteorológicas ao longo de 12 horas (06:00 - 18:00)



Fonte: Elaborado pelo autor, 2024.

#### 4.4 Comparação com dados de referência

Para comparar os dados coletados pelo sistema *IrrigaSync* em 16/10/2024 foram utilizadas informações do Instituto Nacional de Meteorologia (INMET) (2024), cuja estação climática se encontra no IFMG, *Campus Bambuí*. A Tabela 15 apresenta uma comparação entre as variáveis climáticas registradas pelo *IrrigaSync* e os dados disponíveis do INMET para a mesma data e período horário.

Tabela 15 – Comparação dos dados climáticos em Bambuí em 16/10/2024

Variável	<i>IrrigaSync</i>	INMET
Temperatura Mínima (°C)	19	20
Temperatura Máxima (°C)	33	38
Umidade Relativa Mínima (%)	65	40
Umidade Relativa Máxima (%)	85	80

Fonte: Dados do sistema *IrrigaSync* e do Instituto Nacional de Meteorologia (INMET) (2024).

Observa-se que os dados de temperatura mínima e umidade relativa máxima registrados pelo *IrrigaSync* estão próximos aos valores fornecidos pelo INMET. No entanto, há uma discrepância na temperatura máxima e na umidade relativa mínima. Essa diferença

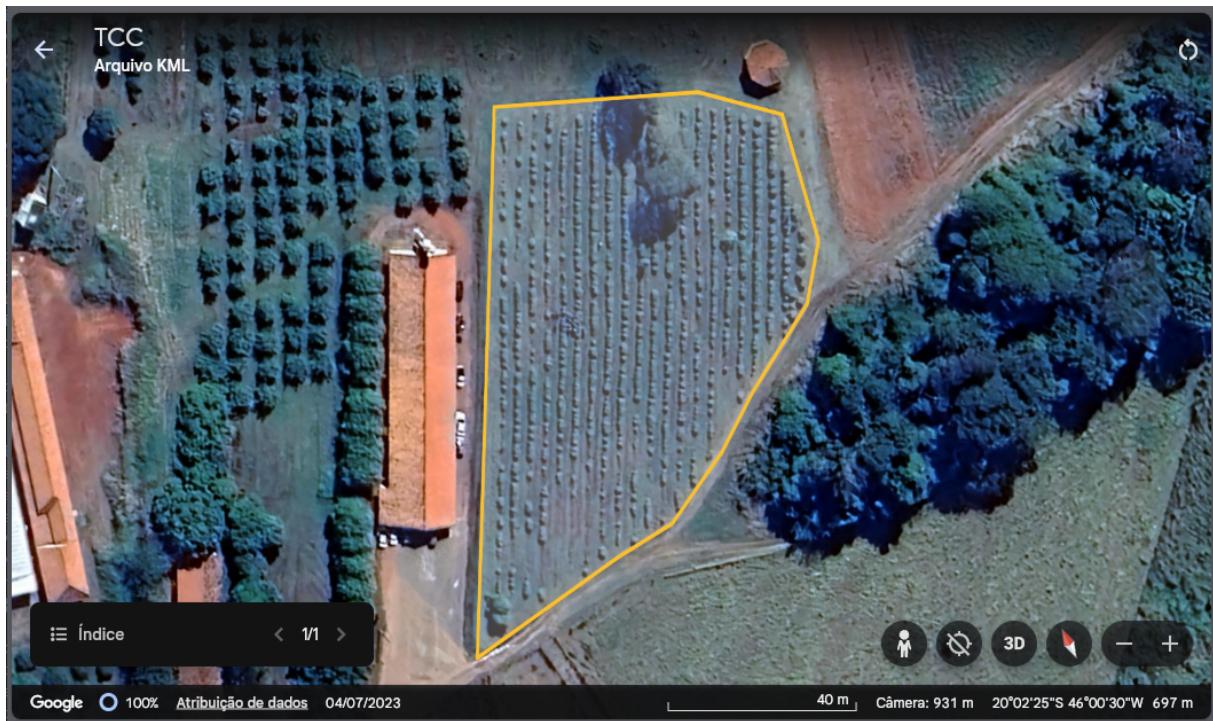
pode ser atribuída a fatores como a localização específica dos sensores do *IrrigaSync*, possíveis microclimas locais ou variações nos horários de medição.

É importante considerar que os dados do INMET são obtidos de estações meteorológicas padronizadas e calibradas regularmente, enquanto o *IrrigaSync* é um sistema em desenvolvimento que pode necessitar de ajustes e calibrações adicionais para melhorar a precisão de suas medições.

#### 4.5 Estimativa da evapotranspiração

Com base nas variáveis coletadas, o sistema calculou a ETo e a ETc para uma pequena plantação de café arábica utilizando o método FAO-56 (ALLEN *et al.*, 1998). A plantação, situada ao lado do laboratório, corresponde a uma microcultura, cobrindo uma área de aproximadamente 6260,5 m<sup>2</sup>, com condições controladas e específicas (Figura 29).

Figura 29 – Microcultura de café arábica do IFMG - *Campus Bambuí*



Fonte: Elaborado pelo autor com *Google Earth*.

De acordo com Rodrigues *et al.* (2013), o Kc do café arábica varia de acordo com o estágio de desenvolvimento da planta e o manejo de irrigação. Em lavouras irrigadas, o valor do Kc para o café arábica pode variar ao longo do ano entre 0,21 e 0,80, com uma média de 0,57, dependendo das condições e do manejo da cultura. No caso específico para essa época do ano, em outubro, geralmente há uma variação dependendo da fase fenológica da planta e das condições climáticas da região.

Estudos mostram que valores de Kc entre 0,72 e 1,50 são comuns em períodos de maior demanda hídrica, que ocorrem tipicamente entre junho e setembro, mas isso

pode se estender, especialmente se houver alta evapotranspiração (SOUZA *et al.*, 2005). Esses valores podem ser ajustados conforme o manejo de irrigação local, como observado em estudos feitos em Minas Gerais, onde o clima e a metodologia de irrigação impactam bastante o valor do Kc aplicado.

Para adotar um valor médio confiável do Kc para o café arábica utilizando uma abordagem científica, um valor de 0,60 é geralmente uma boa referência durante o ano, especialmente em lavouras irrigadas (RODRIGUES *et al.*, 2013). Isso é corroborado por estudos que mostram que o Kc do café arábica varia entre 0,21 e 0,80, com um valor médio ao longo do ciclo produtivo próximo a 0,57. No sistema implementado, o valor de Kc pode ser definido como entrada ao clicar na logo do aplicação na barra de navegação superior, sendo que para a microcultura em questão foi adotado o valor de 0,60.

A Tabela 16 apresenta os valores estimados para ETo e ETc no dia 16/10/2024 para essa microcultura de café arábica capturados pelo sistema *IrrigaSync*.

Tabela 16 – Valores de ETo e ETc coletados pelo sistema *IrrigaSync*

Hora	ETo (mm/dia)	ETc (mm/dia)
06:00	2.88	1.73
09:00	4.32	2.59
12:00	6.00	3.60
15:00	4.80	2.88
18:00	3.60	2.16

Fonte: Resultados obtidos do sistema *IrrigaSync*.

A escolha do café arábica como cultura monitorada reflete a relevância da sua produção na região. Os dados obtidos podem ser úteis para o manejo preciso de água e o controle climático dessa microcultura, exemplificando a aplicabilidade do sistema *IrrigaSync*.

#### 4.6 Discussão dos resultados

Os resultados apresentados demonstram o funcionamento do sistema *IrrigaSync* para monitoramento ambiental e suporte ao manejo de irrigação em uma microcultura de café arábica. A integração entre o *hardware* e o *software* garantiu a coleta e o processamento de dados relevantes para o contexto agrícola, permitindo a estimativa da evapotranspiração.

A análise dos dados coletados revelou que, embora o sistema atenda aos requisitos funcionais propostos, algumas limitações precisam ser abordadas para garantir maior precisão e confiabilidade. Por exemplo, a comparação com os dados do INMET mostrou discrepâncias em algumas variáveis, como a temperatura máxima e a umidade relativa mínima. Tais diferenças podem ser atribuídas a fatores como localização dos sensores, presença de microclimas locais e calibração inicial dos equipamentos.

Além disso, a análise da infraestrutura revelou desafios relacionados à cobertura da rede *Wi-Fi* no laboratório, o que limitou a transmissão de dados em tempo real em algumas áreas.

O sistema também apresentou limitações no sensor de umidade do solo, que registrou variações mais amplas do que o esperado. Isso sugere a necessidade de uma análise mais rigorosa do desempenho desse sensor, incluindo possíveis substituições por modelos mais precisos e adequados ao ambiente em questão.

Outro ponto relevante é a aplicabilidade do *IrrigaSync* para diferentes culturas agrícolas. Embora o protótipo tenha sido validado em uma microcultura de café arábica, sua modularidade e flexibilidade permitem adaptações para outros tipos de cultivo. No entanto, para garantir a eficácia em diferentes cenários, ajustes nos valores de  $K_c$  são necessários, considerando fatores específicos de cada cultura.

## 5 CONCLUSÃO

A realização deste trabalho apresentou um sistema para monitoramento agrícola em tempo real, abordando as necessidades específicas de manejo hídrico e climático para culturas agrícolas. O desenvolvimento da solução baseou-se em boas práticas de engenharia de *software* e *hardware*, garantindo uma integração consistente entre os componentes físicos e computacionais. Além disso, o sistema permitiu a coleta de dados ambientais relevantes, como temperatura, umidade, radiação solar, pressão atmosférica e velocidade do vento, utilizados para calcular a evapotranspiração (ETo e ETc) com base no método FAO-56 (ALLEN *et al.*, 1998).

Os resultados obtidos demonstraram que o sistema atende aos objetivos propostos, especialmente no que tange à usabilidade, flexibilidade e capacidade de adaptação a diferentes cenários agrícolas. A interface intuitiva contribuiu para a acessibilidade e compreensão dos dados pelos usuários finais, enquanto a modularidade da solução oferece espaço para melhorias e expansões futuras.

Contudo, algumas limitações foram identificadas, como a necessidade de calibração mais frequente do sensor de umidade do solo e a dependência de conectividade *Wi-Fi* no raio de acesso. Apesar dessas questões, o sistema mostrou-se promissor e funcional dentro das condições de teste na microcultura observada, com potencial para aplicação em escalas maiores.

Em suma, este trabalho apresenta uma contribuição para o manejo sustentável da agricultura, ao propor uma ferramenta prática e acessível que alia tecnologia e ciência no auxílio à tomada de decisões em campo. As perspectivas de evolução do sistema, descritas na próxima seção, reforçam a viabilidade de ampliação do impacto deste projeto para diferentes cenários e culturas agrícolas.

## 6 TRABALHOS FUTUROS

O sistema *IrrigaSync* oferece uma base para desenvolvimento contínuo e melhorias. Algumas direções futuras incluem:

### 1. Otimização do sistema

- Melhorar os algoritmos de cálculo da evapotranspiração, incorporando novos métodos e equações adaptadas a diferentes condições climáticas e culturas específicas.
- Desenvolver rotinas de autoavaliação para o sistema, permitindo diagnósticos automatizados de falhas em sensores e componentes de rede.

### 2. Calibração e validação

- Realizar calibrações periódicas mais detalhadas para reduzir discrepâncias nos dados coletados, especialmente em relação a fontes oficiais, como o INMET e de outros sistemas de referência.
- Investigar as influências de microclimas locais de forma mais sistemática, para adequar o sistema a diferentes cenários geográficos.

### 3. Expansão funcional

- Integrar sensores adicionais, como medidores de qualidade do ar e sensores de nutrientes para análise do solo.
- Incorporar funcionalidades preditivas, utilizando algoritmos de aprendizado de máquina para prever demandas hídricas e riscos climáticos.

### 4. Interoperabilidade e automação

- Explorar a integração com sistemas de irrigação automatizados, permitindo controle dinâmico baseado em dados em tempo real.
- Desenvolver conectores para plataformas agrícolas existentes e para dispositivos inteligentes, como assistentes virtuais e *smart hubs*.

### 5. Acessibilidade e escalabilidade

- Ampliar a cobertura do sistema em áreas com conectividade limitada, utilizando redes LoRa, Sigfox ou sistemas híbridos.
- Adaptar o sistema para culturas agrícolas de maior escala, analisando seu desempenho em condições de larga produção.

### 6. Validação científica e industrial

- Promover estudos de caso com diferentes tipos de cultivo, em parceria com produtores rurais e instituições de pesquisa, para avaliar o impacto do sistema na produtividade e sustentabilidade agrícola.
- Buscar certificações e validações que permitam a utilização comercial do sistema em larga escala.

Essas melhorias e expansões podem consolidar o *IrrigaSync* como uma solução de referência para monitoramento agrícola, contribuindo para uma agricultura mais sustentável, eficiente e tecnológica.

## REFERÊNCIAS

- PEREIRA, L. S.; OWEIS, T.; ZAIRI, A. Irrigation management under water scarcity. **Agricultural Water Management**, v. 57, n. 3, p. 175–206, dez. 2002. Disponível em: [https://doi.org/10.1016/S0378-3774\(02\)00075-6](https://doi.org/10.1016/S0378-3774(02)00075-6). Acesso em: 20 abr. 2024.
- YANG, Y. *et al.* Evapotranspiration on a greening Earth. **Nature Reviews Earth & Environment**, v. 4, n. 9, p. 626–641, ago. 2023. Disponível em: <https://doi.org/10.1038/s43017-023-00464-3>. Acesso em: 9 abr. 2024.
- DUTTA, L. *et al.* An IoT-Enabled Smart pH Monitoring and Dispensing System for Precision Agriculture Application. **Agricultural Research**, p. 1–10, abr. 2024. Disponível em: <https://doi.org/10.1007/s40003-024-00702-0>. Acesso em: 9 abr. 2024.
- IDRISSI, A. I. D. *et al.* Low-cost IoT-based monitoring system for precision agriculture. **Environment, Energy and Earth Sciences (E3S)**, v. 492, n. 1003, p. 1–9, fev. 2024. Disponível em: <https://doi.org/10.1051/e3sconf/202449201003>. Acesso em: 9 abr. 2024.
- GARG, D.; ALAM, M. Smart agriculture: a literature review. **Journal of Management Analytics**, v. 10, n. 2, p. 359–415, mai. 2023. Disponível em: <https://doi.org/10.1080/23270012.2023.2207184>. Acesso em: 26 abr. 2024.
- SINGH, G. *et al.* Smart Agriculture: A Review. **Siberian Journal of Life Sciences and Agriculture**, v. 14, n. 6, p. 423–454, dez. 2022. Disponível em: <https://doi.org/10.12731/2658-6649-2022-14-6-423-454>. Acesso em: 26 abr. 2024.
- FISCHER, W. A.; HEMPHILL, W. R.; KOVER, A. Progress in Remote Sensing (1972–1976). **Photogrammetria**, v. 32, n. 2, p. 33–72, out. 1976. Disponível em: [https://doi.org/10.1016/0031-8663\(76\)90013-2](https://doi.org/10.1016/0031-8663(76)90013-2). Acesso em: 6 abr. 2024.
- WEISS, M.; JACOB, F.; DUVEILLER, G. Remote Sensing for Agricultural Applications: A Meta-Review. **Remote Sensing of Environment**, v. 236, n. 111402, p. 1–19, jan. 2020. Disponível em: <https://doi.org/10.1016/j.rse.2019.111402>. Acesso em: 6 abr. 2024.
- SHANMUGAPRIYA, P. *et al.* Applications of Remote Sensing in Agriculture-A Review. **International Journal of Current Microbiology and Applied Sciences**, v. 8, n. 1, p. 2270–2283, jan. 2019. Disponível em: <https://doi.org/10.20546/ijcmas.2019.801.238>. Acesso em: 6 abr. 2024.
- KHANAL, S. *et al.* Remote Sensing in Agriculture—Accomplishments, Limitations, and Opportunities. **Remote Sensing**, v. 12, n. 3783, p. 1–29, nov. 2020. Disponível em: <https://doi.org/10.3390/rs12223783>. Acesso em: 6 abr. 2024.
- CARFAGNA, E.; GALLEGOS, F. J. Using Remote Sensing for Agricultural Statistics. **International Statistical Review**, v. 73, n. 3, p. 389–404, dez. 2006. Disponível em: <https://doi.org/10.1111/j.1751-5823.2005.tb00155.x>. Acesso em: 6 abr. 2024.
- TAN, L. *et al.* In-band Network Telemetry: A Survey. **Computer Networks**, v. 186, n. 107763, p. 1–20, fev. 2021. Disponível em: <https://doi.org/10.1016/j.comnet.2020.107763>. Acesso em: 10 abr. 2024.

HASSANIEN, A. E.; DARWISH, A.; ABDELGHAFAR, S. Machine learning in telemetry data mining of space mission: basics, challenging and future directions. **Artificial Intelligence Review**, v. 53, n. 5, p. 3201–3230, set. 2020. Disponível em: <https://doi.org/10.1007/s10462-019-09760-1>. Acesso em: 10 abr. 2024.

DUTTA, P. *et al.* A comprehensive review of embedded system design aspects for rural application platform. **International Journal of Computer Applications**, v. 106, n. 11, p. 39–44, nov. 2014. Disponível em: <https://doi.org/10.5120/18567-9819>. Acesso em: 6 abr. 2024.

CUENCA, E. M.; BALBOA, J. M. M. Practicas y Aplicaciones con microcontroladores/2: los PIC16F84 y PIC 16C71. **Radiorama**, v. 5, n. 2, p. 38–45, dez. 1997. Disponível em: <https://fisiologia-animal.es/publicaciones/PublicacionesLibro5Articulo02.pdf>. Acesso em: 6 abr. 2024.

FERRÁNDEZ-PASTOR, F. J. *et al.* Precision agriculture design method using a distributed computing architecture on internet of things context. **Sensors**, v. 18, n. 1731, p. 1–21, mai. 2018. Disponível em: <https://doi.org/10.3390/s18061731>. Acesso em: 6 abr. 2024.

JUNIOR, F. M. R. *et al.* Data reduction based on machine learning algorithms for fog computing in IoT smart agriculture. **Biosystems Engineering**, v. 223, n. 2, p. 142–158, nov. 2022. Disponível em: <https://doi.org/10.1016/j.biosystemseng.2021.12.021>. Acesso em: 6 abr. 2024.

HSU, T. C. *et al.* A Creative IoT agriculture platform for cloud fog computing. **Sustainable Computing: Informatics and Systems**, v. 28, n. 100285, p. 1–28, dez. 2020. Disponível em: <https://doi.org/10.1016/j.suscom.2018.10.006>. Acesso em: 6 abr. 2024.

RAD, C. R. *et al.* Smart monitoring of potato crop: a cyber-physical system architecture model in the field of precision agriculture. **Agriculture and Agricultural Science Procedia**, v. 6, p. 73–79, ago. 2015. Disponível em: <https://doi.org/10.1016/j.aaspro.2015.08.041>. Acesso em: 6 abr. 2024.

RIJSWIJK, K. *et al.* Digital transformation of agriculture and rural areas: A socio-cyber-physical system framework to support responsibilisation. **Journal of Rural Studies**, v. 85, p. 79–90, jul. 2021. Disponível em: <https://doi.org/10.1016/j.jrurstud.2021.05.003>. Acesso em: 6 abr. 2024.

DATTA, S. P. A. Brief History of the Internet of Things and the Elusive Quest to Measure Performance. **DSpace@MIT**, p. 1–40, 2017. Disponível em: <https://dspace.mit.edu/bitstream/handle/1721.1/111021/HIP+.pdf?sequence=692>. Acesso em: 7 abr. 2024.

KORTUEM, G. *et al.* Smart objects as building blocks for the Internet of things. **IEEE Internet Computing**, v. 14, n. 1, p. 44–51, fev. 2009. Disponível em: <https://doi.org/10.1109/MIC.2009.143>. Acesso em: 6 abr. 2024.

- MINERVA, R.; BIRU, A.; ROTONDI, D. Towards a definition of the Internet of Things (IoT): Revision 1. **IEEE Internet Iniciative**, p. 1–86, mai. 2015. Disponível em: [https://iot.ieee.org/images/files/pdf/IEEE\\_IoT\\_Towards\\_Definition\\_Internet\\_of\\_Things\\_Revision1\\_27MAY15.pdf](https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf). Acesso em: 6 abr. 2024.
- ASHTON, K. That 'internet of things' thing. **RFID journal**, v. 22, n. 1, p. 1–7, 1999. Disponível em: <https://www.itrc.jp/libraries/RFIDjournal-That%20Internet%20of%20Things%20Thing.pdf>. Acesso em: 6 abr. 2024.
- NEUMANN, A.; LARANJEIRO, N.; BERNARDINO, J. An Analysis of Public REST Web Service APIs. **IEEE Transactions on Services Computing**, v. 14, n. 4, p. 957–970, ago. 2021. Disponível em: <https://doi.org/10.1109/TSC.2018.2847344>. Acesso em: 13 abr. 2024.
- PIERCE, F. J.; NOWAK, P. Aspects of Precision Agriculture. **Advances in Agronomy**, v. 67, n. 1, p. 1–85, mai. 1999. Disponível em: [https://doi.org/10.1016/S0065-2113\(08\)60513-1](https://doi.org/10.1016/S0065-2113(08)60513-1). Acesso em: 1 mai. 2024.
- ZHANG, N.; WANG, M.; WANG, N. Precision agriculture—a worldwide overview. **Computers and Electronics in Agriculture**, v. 36, n. 2, p. 113–132, nov. 2002. Disponível em: [https://doi.org/10.1016/S0168-1699\(02\)00096-0](https://doi.org/10.1016/S0168-1699(02)00096-0). Acesso em: 1 mai. 2024.
- GEBBERS, R.; ADAMCHUK, V. I. Precision Agriculture and Food Security. **Science**, v. 327, n. 5967, p. 828–831, fev. 2010. Disponível em: <https://doi.org/10.1126/science.1183899>. Acesso em: 1 mai. 2024.
- MEDEL-JIMÉNEZ, F. *et al.* Life cycle assessment of four different precision agriculture technologies and comparison with a conventional scheme. **Journal of Cleaner Production**, v. 434, n. 140198, p. 1–11, jan. 2024. Disponível em: <https://doi.org/10.1016/j.jclepro.2023.140198>. Acesso em: 1 mai. 2024.
- PENMAN, H. L. Natural evaporation from open water, bare soil and grass. **Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences**, v. 193, n. 1032, p. 120–145, abr. 1948. Disponível em: <https://doi.org/10.1098/rspa.1948.0037>. Acesso em: 22 abr. 2024.
- LI, Z.-L. *et al.* A Review of Current Methodologies for Regional Evapotranspiration Estimation from Remotely Sensed Data. **Sensors**, v. 9, n. 5, p. 3801–3853, mar. 2009. Disponível em: <https://doi.org/10.3390/s90503801>. Acesso em: 22 abr. 2024.
- ZHANG, K.; KIMBALL, J. S.; RUNNING, S. W. A review of remote sensing based actual evapotranspiration estimation. **Wiley interdisciplinary reviews: Water**, v. 3, n. 6, p. 834–853, jul. 2016. Disponível em: <https://doi.org/10.1002/wat2.1168>. Acesso em: 22 abr. 2024.
- ANDERSON, M. *et al.* Improved Daily Evapotranspiration Estimation Using Remotely Sensed Data in a Data Fusion System. **Remote Sensing**, v. 14, n. 1772, p. 1–16, abr. 2022. Disponível em: <https://doi.org/10.3390/rs14081772>. Acesso em: 22 abr. 2024.

ZHANG, J. *et al.* Sensor data dissemination systems using Web-based standards: a case study of publishing data in support of evapotranspiration models in California. **Civil Engineering and Environmental Systems**, v. 26, n. 1, p. 35–52, fev. 2009. Disponível em: <https://doi.org/10.1080/10286600802003567>. Acesso em: 28 mai. 2024.

WU, F. *et al.* ETWatch cloud: APIs for regional actual evapotranspiration data generation. **Environmental Modelling & Software**, v. 145, n. 105174, p. 1–16, mai. 2021. Disponível em: <https://doi.org/10.1016/j.envsoft.2021.105174>. Acesso em: 28 mai. 2024.

BORTOLIN, T. A. *et al.* Web system for estimation of evapotranspiration potential by different methods. **Ciência e Natura**, v. 41, n. 22, p. 1–14, jul. 2019. Disponível em: <https://doi.org/10.5902/2179460X36937>. Acesso em: 28 mai. 2024.

DANIELESCU, S. Development and Application of ETCalc, a Unique Online Tool for Estimation of Daily Evapotranspiration. **Atmosphere-Ocean**, v. 61, n. 3, p. 135–147, dez. 2022. Disponível em: <https://doi.org/10.1080/07055900.2022.2154191>. Acesso em: 28 mai. 2024.

MICHAEL, P. R.; JOHNSTON, D. E.; MORENO, W. A conversion guide: solar irradiance and lux illuminance. **Journal of Measurements in Engineering**, v. 8, n. 4, p. 153–166, dez. 2020. Disponível em: <https://doi.org/10.21595/jme.2020.21667>. Acesso em: 16 ago. 2024.

VIJH, S. *et al.* IOT Based Real-Time Monitoring System for Precision Agriculture. In: 2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence). Noida, India: IEEE, 2024. p. 53–58. Disponível em: <https://doi.org/10.1109/Confluence60223.2024.10463399>. Acesso em: 9 abr. 2024.

PRATHIBHA, S. R.; HONGAL, A.; JYOTHI, M. P. IOT Based Monitoring System in Smart Agriculture. In: 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT). Bangalore, India: IEEE, 2017. p. 81–84. Disponível em: <https://doi.org/10.1109/ICRAECT.2017.52>. Acesso em: 9 abr. 2024.

DING, B.; KULKARNI, J.; YEKHANIN, S. Collecting Telemetry Data Privately. In: NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA: Curran Associates, 2017. p. 3574–3583. Disponível em: [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/253614bbac999b38b5b60cae531c4969-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/253614bbac999b38b5b60cae531c4969-Paper.pdf). Acesso em: 9 abr. 2024.

LEE, E. A. Cyber physical systems: Design challenges. In: 2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC). Orlando, USA: IEEE, 2008. p. 363–369. Disponível em: <https://doi.org/10.1109/ISORC.2008.25>. Acesso em: 6 abr. 2024.

KATO, S. *et al.* Autoware on board: Enabling autonomous vehicles with embedded systems. In: 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPs). Porto, Portugal: IEEE, 2018. p. 287–296. Disponível em: <https://doi.org/10.1109/ICCPs.2018.00035>. Acesso em: 6 abr. 2024.

- MAIER, A.; SHARP, A.; VAGAPOV, Y. Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. In: 2017 Internet Technologies and Applications (ITA). Wrexham, UK: IEEE, 2017. p. 143–148. Disponível em: <https://doi.org/10.1109/ITECHA.2017.8101926>. Acesso em: 6 abr. 2024.
- MARAGATHAM, T.; BALASUBRAMANIE, P.; VIVEKANANDHAN, M. IoT based home automation system using raspberry Pi 4. In: IOP Conference Series: Materials Science and Engineering. Erode, India: IOP Publishing, 2021. p. 1–7. Disponível em: <https://doi.org/10.1088/1757-899X/1055/1/012081>. Acesso em: 6 abr. 2024.
- AHMAD, I.; POTHUGANTI, K. Smart field monitoring using ToxTrac: A cyber-physical system approach in agriculture. In: 2020 International Conference on Smart Electronics and Communication (ICOSEC). Trichy, India: IEEE, 2020. p. 723–727. Disponível em: <https://doi.org/10.1109/ICOSEC49089.2020.9215282>. Acesso em: 6 abr. 2024.
- TAN, L.; WANG, N. Future internet: The Internet of Things. In: 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE). Chengdu, China: IEEE, 2010. p. 376–380. Disponível em: <https://doi.org/10.1109/ICACTE.2010.5579543>. Acesso em: 6 abr. 2024.
- YASSEIN, M. B. *et al.* Internet of Things: Survey and open issues of MQTT protocol. In: 2017 International Conference on Engineering & MIS (ICEMIS). Monastir, Tunisia: IEEE, 2017. p. 1–6. Disponível em: <https://doi.org/10.1109/ICEMIS.2017.8273112>. Acesso em: 15 abr. 2024.
- STANFORD-CLARK, A. The importance of Internet standards and other topics. In: 2004 Open Control Systems - The Importance of Industrial Standards. Birmingham, UK: IET, 2004. p. 1–12. Disponível em: <https://doi.org/10.1049/ic:20040131>. Acesso em: 15 abr. 2024.
- RAMOS, C. M. C. *et al.* **Solos: Estudos, Potencialidades e Uso**. 1. ed.: Embrapa Instrumentação, 2022. Disponível em: <https://ainfo.cnptia.embrapa.br/digital/bitstream/item/232593/1/P-Solos-e-irrigacao.pdf>. Acesso em: 20 abr. 2024.
- CAMPBELL, J. B.; WYNNE, R. H. **Introduction to Remote Sensing**. 5. ed.: The Guilford Press, 2011. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=NkLmDjSS8TsC&oi=fnd&pg=PP1&dq=Introduction+to+Remote+Sensing&ots=s3Fmx8o\\_0v&sig=sLnEcKNrNmLWjhErgUUUi-2t7FYc&redir\\_esc=y#v=onepage&q=Introduction%20to%20Remote%20Sensing&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=NkLmDjSS8TsC&oi=fnd&pg=PP1&dq=Introduction+to+Remote+Sensing&ots=s3Fmx8o_0v&sig=sLnEcKNrNmLWjhErgUUUi-2t7FYc&redir_esc=y#v=onepage&q=Introduction%20to%20Remote%20Sensing&f=false). Acesso em: 6 abr. 2024.
- CARDEN, F.; JEDLICKA, R.; HENRY, R. **Telemetry Systems Engineering**. 1. ed.: Artech House, 2002. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=3LK44UI0YCEC&oi=fnd&pg=PR25&dq=telemetry&ots=PhIcD\\_akyq&sig=F226i8zRr4R9LrIs2vuhk4KinZU&redir\\_esc=y#v=onepage&q=telemetry&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=3LK44UI0YCEC&oi=fnd&pg=PR25&dq=telemetry&ots=PhIcD_akyq&sig=F226i8zRr4R9LrIs2vuhk4KinZU&redir_esc=y#v=onepage&q=telemetry&f=false). Acesso em: 10 abr. 2024.
- MAZIDI, M. A.; NAIMI, S.; NAIMI, S. **The AVR Microcontroller and Embedded Systems: Using Assembly and C**. 1. ed.: Pearson Prentice Hall, 2011. Disponível em: <https://dl.acm.org/doi/abs/10.5555/3169537>. Acesso em: 25 abr. 2024.

- WARREN, J. D.; ADAMS, J.; MOLLE, H. **Arduino Robotics (Technology in Action)**. 1. ed.: Apress, 2011. Disponível em: <https://doi.org/10.1007/978-1-4302-3184-4> \_ 2. Acesso em: 10 abr. 2024.
- BROWN, G. **Discovering the STM32 Microcontroller**. 1. ed.: Indiana University, 2012. Disponível em: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=d1fbdf8a3a1b32fd9c951f44f0f5403232412795>. Acesso em: 6 abr. 2024.
- MCPHERSON, S. S. **Tim Berners-Lee: Inventor of the World Wide Web**. 1. ed.: USA TODAY lifeline biographies, 2010. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=wK0xBWfL9GkC&oi=fnd&pg=PP1&dq=Tim+Berners-Lee+http&ots=8hQS8517jw&sig=yZRbgIMcN0xFMVPNTQ33b\\_5Rt1E&redir\\_esc=y#v=onepage&q=Tim%20Berners-Lee%20http&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=wK0xBWfL9GkC&oi=fnd&pg=PP1&dq=Tim+Berners-Lee+http&ots=8hQS8517jw&sig=yZRbgIMcN0xFMVPNTQ33b_5Rt1E&redir_esc=y#v=onepage&q=Tim%20Berners-Lee%20http&f=false). Acesso em: 13 abr. 2024.
- LAMINE, S. *et al.* **Remote Sensing in Precision Agriculture**. 1. ed.: Academic Press, 2024. Disponível em: <https://doi.org/10.1016/B978-0-323-91068-2.00014-X>. Acesso em: 1 mai. 2024.
- BURTON, M. **Irrigation Management: Principles and Practices**. 1. ed.: CAB International, 2010. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=n3Il28LP2usC&oi=fnd&pg=P\\_R5&dq=irrigation+management&ots=JWhCdIR0XG&sig=8zuy2jlLkX2zsVoMBJK283jmBLE&redir\\_esc=y#v=onepage&q=irrigation%20management&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=n3Il28LP2usC&oi=fnd&pg=P_R5&dq=irrigation+management&ots=JWhCdIR0XG&sig=8zuy2jlLkX2zsVoMBJK283jmBLE&redir_esc=y#v=onepage&q=irrigation%20management&f=false). Acesso em: 20 abr. 2024.
- MOOD, A. M.; GRAYBILL, F. A.; BOES, D. C. **Introduction to the Theory of Statistics**. 3. ed.: McGraw Hill, 1974. Disponível em: <https://sistemas.fciencias.unam.mx/~misraim/Mood.pdf>. Acesso em: 6 dez. 2024.
- ALLEN, R. G. *et al.* **Crop evapotranspiration - Guidelines for computing crop water requirements**. 1. ed.: FAO Irrigation e drainage Paper, 1998. Disponível em: <https://www.fao.org/3/X0490E/x0490e00.htm>. Acesso em: 22 abr. 2024.
- BERNARDO, S. *et al.* **Manual de irrigação**. 8. ed.: UFV, 2008. Disponível em: [https://www.academia.edu/45657937/Manual\\_de\\_Irriga%C3%A7%C3%A3o\\_8\\_ed\\_S\\_lassier\\_Bernardo\\_Editora\\_UFV](https://www.academia.edu/45657937/Manual_de_Irriga%C3%A7%C3%A3o_8_ed_S_lassier_Bernardo_Editora_UFV). Acesso em: 23 abr. 2024.
- PAYERO, J. O. *et al.* **Application of a New Web-Based Tool (Crop WaterUse) for Determining Evapotranspiration and Irrigation Requirements of Major Crops at Three Locations in Queensland**. 1. ed.: IntechOpen, 2011. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=YheaDwAAQBAJ&oi=fnd&pg=PA295&dq=web+application+evapotranspiration&ots=ZbDFqht\\_7s&sig=tjNS6zBCLOebLDX8Z0SzQ\\_E9SAk#v=onepage&q=web%20application%20evapotranspiration&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=YheaDwAAQBAJ&oi=fnd&pg=PA295&dq=web+application+evapotranspiration&ots=ZbDFqht_7s&sig=tjNS6zBCLOebLDX8Z0SzQ_E9SAk#v=onepage&q=web%20application%20evapotranspiration&f=false). Acesso em: 28 mai. 2024.
- VUOLO, J. H. **Fundamentos da Teoria de Erros**. 2. ed.: Blucher, 1996. Disponível em: [https://books.google.com.br/books?hl=pt-BR&lr=&id=q-uyDwAAQBAJ&oi=fnd&pg=PA1&dq=Fundamentos+da+Teoria+de+Erro&ots=ekWOIUYVLx&sig=MIY-uL3XrAGMCb\\_gb93fBQBsezw#v=onepage&q=Fundamentos%20da%20Teoria%20de%20Erro&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=q-uyDwAAQBAJ&oi=fnd&pg=PA1&dq=Fundamentos+da+Teoria+de+Erro&ots=ekWOIUYVLx&sig=MIY-uL3XrAGMCb_gb93fBQBsezw#v=onepage&q=Fundamentos%20da%20Teoria%20de%20Erro&f=false). Acesso em: 21 set. 2024.

EVANS, E. **Domain-Driven Design**. 1. ed.: Pearson Education, 2004. Disponível em: <https://books.google.com.br/books?hl=pt-BR&lr=&id=xColAAPGubgC&oi=fnd&pg=PR9&dq=domain+driven+design&ots=qdUCHfWI4w&sig=jfdHj78OvDmrMTWRRm5sMwUr6gc#v=onepage&q=domain%20driven%20design&f=false>. Acesso em: 16 ago. 2024.

CHACON, S.; STRAUB, B. **Pro Git**. 2. ed.: Apress, 2024. Disponível em: <https://git-scm.com/book/en/v2>. Acesso em: 16 ago. 2024.

CARMODY, E.; TURRAL, H. **Consumption-based water management**. Set. 2023. Food and Agriculture Organization (FAO). Disponível em: <https://doi.org/10.4060/cc6752en>. Acesso em: 9 abr. 2024.

WHARTON, J. **Using the Intel MCS-51 Boolean Processing Capabilities**. Abr. 1980. Intel Corporation. Disponível em: [http://bear.ces.case.edu/eecs\\_281/8051\\_boolean.pdf](http://bear.ces.case.edu/eecs_281/8051_boolean.pdf). Acesso em: 6 abr. 2024.

FIELDING, R. *et al.* **RFC 2616: Hypertext Transfer Protocol – HTTP/1.1**. Jun. 1999. University of California, Irvine. Disponível em: <https://doi.org/10.17487/RFC2616>. Acesso em: 13 abr. 2024.

BANKS, A.; BORGENDALE, K.; GUPTA, R. **MQTT Version 5**. Mar. 2019. OASIS. Disponível em: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>. Acesso em: 15 abr. 2024.

INSTITUTO NACIONAL DE METEOROLOGIA (INMET). **Dados meteorológicos de Bambuí, MG – 16/10/2024**. Out. 2024. INMET. Disponível em: <https://www.inmet.gov.br>. Acesso em: 16 out. 2024.

ESPRESSIF SYSTEMS. **ESP32 Series Datasheet: 2.4 GHz Wi-Fi + Bluetooth® + Bluetooth LE SoC**. 2024. Disponível em: [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf). Acesso em: 8 abr. 2024.

AOSONG ELECTRONICS. **Digital-output relative humidity & temperature sensor/module DHT22 (DHT22 also named as AM2302)**. 2024. Disponível em: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>. Acesso em: 21 set. 2024.

BOSCH SENSORTEC. **BMP280: Digital Pressure Sensor**. 2015. Disponível em: <https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf>. Acesso em: 21 set. 2024.

ROHM SEMICONDUCTOR. **BH1750: Digital 16bit Serial Output Type Ambient Light Sensor IC**. 2011. Disponível em: <https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf>. Acesso em: 21 set. 2024.

RENKE. **RS-FSJT-N01 Wind speed transmitter user's Guide (Type 485)**. 2024. Disponível em: <https://instrucenter.com/wp-content/uploads/2022/03/RS-FSJT-No1.pdf>. Acesso em: 21 set. 2024.

ON SEMICONDUCTOR. **LM393, LM393E, LM293, LM2903, LM2903E, LM2903V, NCV2903 - LoW Offset Voltage Dual Comparators.** 2024. Disponível em: <https://docs.rs-online.com/943a/0900766b8170d70c.pdf>. Acesso em: 20 nov. 2024.

U-BLOX. **NEO-6 u-blox 6 GPS Modules.** 2011. Disponível em: [https://content.u-blox.com/sites/default/files/products/documents/NEO-6\\_DataSheet\\_%28GPS.G6-HW-09005%29.pdf](https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf). Acesso em: 20 nov. 2024.

PCB POWER MARKET. **RS485-TTL Module.** 2024. Disponível em: <https://loginpcbpower.com/Documents/0252d3ac-d61f-43a7-9b96-010299e5181c/PM14058.pdf>. Acesso em: 20 nov. 2024.

EMQ TECHNOLOGIES INC. **EMQX Documentation.** 2024. Disponível em: <https://docs.emqx.com/en/>. Acesso em: 16 ago. 2024.

AIEE TELEMETERING SUBCOMMITTEE. **Telemetering, Supervisory Control, and Associated Circuits.** New York, USA, 1941. p. 1–29. Disponível em: <https://babel.hathitrust.org/cgi/pt?id=mdp.39015002957788&seq=4>. Acesso em: 10 abr. 2024.

RODRIGUES, W. L. *et al.* Índice de área foliar, coeficiente de cultura e produtividade de cafeeiro fertirrigado. **Engenharia Agrícola**, v. 33, n. 4, p. 794–804, 2013. DOI: 10.1590/S0100-69162013000400014. Disponível em: <https://doi.org/10.1590/S0100-69162013000400014>. Acesso em: 24 out. 2024.

SOUZA, C. F. *et al.* Evapotranspiração e coeficiente de cultura do cafeeiro irrigado por pivô central. **Revista Brasileira de Engenharia Agrícola e Ambiental**, v. 9, n. 4, p. 533–539, 2005. DOI: 10.1590/S1415-43662005000400018. Disponível em: <https://doi.org/10.1590/S1415-43662005000400018>. Acesso em: 24 out. 2024.