

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
GOIANO - CAMPUS RIO VERDE, PROGRAMA DE
PÓS-GRADUAÇÃO EM CIÊNCIAS AGRÁRIAS – AGRONOMIA

SISTEMA DE IRRIGAÇÃO INTELIGENTE BASEADO EM LÓGICA
FUZZY INTEGRADO COM INTERNET DAS COISAS PARA A
CULTURA DO TOMATE CEREJA

Autor: Sergio Souza Novak
Orientador: Prof. Dr. Adriano Soares de Oliveira Bailão

RIO VERDE – GO
Dezembro - 2025

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
GOIANO - CAMPUS RIO VERDE, PROGRAMA DE
PÓS-GRADUAÇÃO EM CIÊNCIAS AGRÁRIAS – AGRONOMIA

**SISTEMA DE IRRIGAÇÃO INTELIGENTE BASEADO EM
LÓGICA FUZZY INTEGRADO COM INTERNET DAS
COISAS PARA A CULTURA DO TOMATE CEREJA**

Autor: Sergio Souza Novak
Orientador: Prof. Dr. Adriano Soares de Oliveira Bailão

Rio Verde – Go

Dezembro de 2025

**Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema Integrado de Bibliotecas do IF Goiano - SIBi**

N936s Souza Novak, Sergio
 SISTEMA DE IRRIGAÇÃO INTELIGENTE BASEADO EM
 LÓGICA FUZZY INTEGRADO COM INTERNET DAS
 COISAS PARA A CULTURA DO TOMATE CEREJA / Sergio
 Souza Novak. Rio Verde 2026.

80f. il.

Orientador: Prof. Dr. Adriano Soares de Oliveira Bailão.
Dissertação (Mestre) - Instituto Federal Goiano, curso de
0231014 - Mestrado em Ciências Agrárias - Agronomia - Rio
Verde (Campus Rio Verde).

1. Sistema Especialista. 2. Controle de Irrigação. 3. Conservação
de Água. 4. Agricultura de Precisão. 5. Produto IoT. I. Título.

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610, de 19 de fevereiro de 1998, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano a disponibilizar gratuitamente o documento em formato digital no Repositório Institucional do IF Goiano (RIIF Goiano), sem resarcimento de direitos autorais, conforme permissão assinada abaixo, para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

IDENTIFICAÇÃO DA PRODUÇÃO TÉCNICO-CIENTÍFICA

- | | |
|--|--------------------------------|
| Tese (doutorado) | Artigo científico |
| <input checked="" type="checkbox"/> Dissertação (mestrado) | Capítulo de livro |
| Monografia (especialização) | Livro |
| TCC (graduação) | Trabalho apresentado em evento |

Produto técnico e educacional - Tipo:

Nome completo do autor:
Sergio Souza Novak
Matrícula:
2024102310140007
Título do trabalho:
SISTEMA DE IRRIGAÇÃO INTELIGENTE BASEADO EM LÓGICA FUZZY INTEGRADO COM INTERNET DAS COISAS PARA A CULTURA DO TOMATE CEREJA

RESTRIÇÕES DE ACESSO AO DOCUMENTO

Documento confidencial: Não Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: 5 / 1 /2026

O documento está sujeito a registro de patente? Sim Não

O documento pode vir a ser publicado como livro? Sim Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O(a) referido(a) autor(a) declara:

- Que o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- Que obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autoria, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- Que cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Documento assinado digitalmente
gov.br
SERGIO SOUZA NOVAK
Data: 05/01/2026 15:27:16-0300
Verifique em <https://validar.iti.gov.br>

Rio Verde
Local

5 / 1 /2026
Data

Ciente e de acordo:

Assinatura do autor e/ou detentor dos direitos autorais

Documento assinado digitalmente
gov.br ADRIANO SOARES DE OLIVEIRA BAILAO
Data: 05/01/2026 15:30:52-0300
Verifique em <https://validar.iti.gov.br>

Assinatura do(a) orientador(a)



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÉNCIA E TECNOLOGIA GOIANO

Documentos 80/2025 - SREPG/CMPR/CPG-RV/DPGPI-RV/CMPPRV/IFGOIANO

SISTEMA DE IRRIGAÇÃO INTELIGENTE BASEADO EM LÓGICA FUZZY INTEGRADO COM INTERNET DAS COISAS PARA A CULTURA DO TOMATE CEREJA

Autor: Sergio Souza Novak

Orientador: Dr. Adriano Soares de Oliveira Bailão

TITULAÇÃO: Mestre em Ciências Agrárias-Agronomia - Área de Concentração
em Produção Vegetal Sustentável no Cerrado

APROVADA em, 28 de novembro de 2025.

Prof. Dr. Adriano Soares de Oliveira Bailão (Presidente)

Prof. Dr. Marconi Batista Teixeira (Avaliador interno)

Profª. Dra. Fabiana Girotto Ribeiro (Avaliadora externa)

Prof. Dr. Charles Barbosa Santos (Avaliador externo)

Prof. Dr. Ricardo Francischini (Avaliador externo)

FABIANA GIROTTTO / Assinado de forma digital por
RIBEIRO / FABIANA GIROTTTO RIBEIRO
Dados: 2025.12.03 08:49:30 -03'00'

Documento assinado digitalmente

gov.br CHARLES BARBOSA SANTOS
Data: 02/12/2025 21:53:41-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente

gov.br RICARDO FRANCISCHINI
Data: 03/12/2025 07:22:30-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado eletronicamente por:

- Adriano Soares de Oliveira Bailao, PROFESSOR ENS BASICO TECN TECNOLOGICO , em 02/12/2025 17:52:25.
- Marconi Batista Teixeira, PROFESSOR ENS BASICO TECN TECNOLOGICO , em 02/12/2025 18:55:52.

Este documento foi emitido pelo SUAP em 26/11/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 766644

Código de Autenticação: 6539c08cf8



ÍNDICE

LISTA DE SÍMBOLOS, SIGLAS, ABREVIACÕES E UNIDADES	10
RESUMO	14
ABSTRACT	15
1. INTRODUÇÃO	16
2. OBJETIVOS	19
3. MATERIAL E MÉTODOS	20
3.1. Delineamento Experimental	20
3.3. Características do Solo	23
3.4. Fluxo Experimental	25
3.2. Dispositivos de internet das coisas e seu posicionamento	26
3.2.1. Microcontrolador ESP32	28
3.2.2. Sensores	29
3.2.3 Circuito de leitura dos dados	30
3.2.4. Circuito de Acionamento da Bomba	31
3.3. Lógica Fuzzy	33
3.3.1. Conjuntos Fuzzy	35
3.3.2. Regras Fuzzy	39
3.3.3. Regras para a temperatura fria (Regras 1 a 9)	42
3.3.4. Regras para a temperatura moderada (Regras 10 a 18)	43
3.3.5. Regras para a temperatura quente (Regras 19 a 27)	43
3.3.6. Método de Inferência pelo Centroide	44
3.4. Linguagens de programação, códigos, software e protocolos utilizados	45
3.5. Aspectos Arquiteturais do Sistema	47
3.5.1. Módulo Arquitetural Backend API (Flask)	50
3.5.2. Módulo Arquitetural Sistemas Embarcados	52
3.6 Interface Gráfica do Site	55
3.6.1. Tela de Inserção de Classes Fuzzy da Irrigação Fuzzy Inteligente	55
3.6.2. Tela de Irrigação Atual da Irrigação Fuzzy Inteligente	57
3.6.3. Tela de Visão de Irrigação para irrigação agendada	58
4. RESULTADOS	60
4.1 Comparação entre volumes de substrato	61
4.2. Comparação entre manejos de irrigação	62
4.3. Publicações e Registros	63
5. CONCLUSÃO	63
6. REFERÊNCIAS	65
APÊNDICES 73	

ÍNDICE DE FIGURAS

Figura 1 – Foto do SILF	19
Figura 2 – Visualização esquemática do delineamento experimental.	20
Figura 3 – Fotos do sistema de drenagem.	20
Figura 4 – Visualização esquemática dos Tratamentos e Manejos que compõem o Arranjo Fatorial deste trabalho.	21
Figura 5 – Visualização esquemática do experimento.	24
Figura 6 – Modelo 3D da estação de controle do experimento, juntamente com sua comparação foto real.	25
Figura 7 – ESP32, o microcontrolador utilizado neste trabalho.	26
Figura 8 – Sensores utilizados no trabalho.	26
Figura 9 – Circuito de ligação dos cinco sensores de umidade de solo e do sensor DHT11.	28
Figura 10 – Atuadores do SILF.	28
Figura 11 – Contator e Relé.	29
Figura 12 – Circuito de acionamento do motor.	29
Figura 13 – Desenho esquemático do modelo lógica fuzzy.	31
Figura 14 – O processo de inferência Fuzzy.	32
Figura 15 – Os conjuntos fuzzy definidos neste trabalho.	33
Figura 16 – Interface gráfica no site para inserção de funções de pertinência dos conjuntos fuzzy.	34
Figura 17 – Processo de decodificação da saída.	36
Figura 18 – Esquema de comunicação dos dispositivos do sistema.	42
Figura 19 – Foto de um trecho do código fonte implementado na placa ESP32.	44
Figura 20 – Diagrama de componentes.	45
Figura 21 – Diagrama de classes para o Módulo Arquitetural Backend API (Flask).	46
Figura 22 – Diagrama de Sequência para consulta do estado da bomba.	48
Figura 23 – Diagrama de Sequência para atualização da bomba.	49
Figura 24 – Diagrama de componentes para o Módulo Arquitetural Sistemas Embarcados.	50

Figura 25 – Diagrama de classes para o Módulo Arquitetural Sistemas Embarcados.	51
Figura 26 – Tela de Inserção de Classes Fuzzy.	52
Figura 27 – Inserção e alteração de conjuntos fuzzy para umidade do solo.	53
Figura 28 – Inserção e alteração de conjuntos fuzzy para temperatura do ar.	53
Figura 29 – Inserção e alteração de regras.	54
Figura 30 – Tela de irrigação atual com irrigação acontecendo.	55
Figura 31 – Tela de irrigação atual com irrigação inativa.	55
Figura 32 – Tela de Visão de Irrigação para irrigação agendada.	56
Figura 33 – Modal de inserção de irrigação agendada.	56
Figura 34 – Gráfico de contribuições.	58
Figura A – Sistema de Drenagem: construção do sistema e drenagem sendo coletada.	70
Figura B – Teste para verificar a equidade na vazão dos gotejadores.	70
Figura C – Apresentação do SILF em um evento.	71

ÍNDICE DE TABELAS

Tabela 1 – Graus de liberdade do experimento.	20
Tabela 2 – Laudo de Programa de Qualidade de Análise de Solo, incluindo Análise Básica, referente ao ano de 2025 da Fazenda Experimental do Instituto Federal Goiano, em Rio Verde, Goiás.	21
Tabela 3 – Laudo de Programa de Qualidade de Análise de Solo da Análise Granulométrica referente ao ano de 2025 da Fazenda Experimental do Instituto Federal Goiano, em Rio Verde, Goiás.	22
Tabela 4 – Cronograma de estágios do experimento.	23
Tabela 5 – Cronograma de coleta de resultados.	23
Tabela 6 – Conjuntos Fuzzy definidos para o SILF.	33
Tabela 7 – Regras Fuzzy.	37
Tabela 8 – Fontes de variação.	58
Tabela 9 – Comparações múltiplas de médias da drenagem.	59
Tabela 10 – Comparações múltiplas de médias da drenagem (Tukey HSD) – Manejos de irrigação.	61

LISTA DE SÍMBOLOS, SIGLAS, ABREVIAÇÕES E UNIDADES

ANOVA - Análise de Variância

API - Application Programming Interface

ESP32 - Microcontrolador com Wi-Fi integrado (nome comercial)

FWER - Family-Wise Error Rate

INPI - Instituto Nacional de Propriedade Intelectual

HTTP - Hypertext Transfer Protocol

IoT - Internet das Coisas

PPGCA-AGRO - Programa de Pós-Graduação em Ciências Agrárias – Agronomia

REST - Representational State Transfer

SILF - Sistema de Irrigação Inteligente por Lógica Fuzzy

UML - Unified Modeling Language

V - Volts

AGRADECIMENTOS

Minha mais profunda e carinhosa gratidão aos meus pais, Sergio e Rosilei. Sua sabedoria, amor incondicional e fé inabalável foram luzes constantes em minha caminhada. Suas palavras de encorajamento e seus exemplos de resiliência seguem sendo fonte diária de inspiração. Vocês me ensinaram o valor da persistência e a importância de acreditar em grandes sonhos. Por isso, esta conquista também é de vocês, meus pais queridos. Agradeço também ao Hector Alexis, que me ajudou plantando cada tomateiro, preparando cada vaso, a fim de colhermos os frutos desta dissertação. Agradeço a todo o conhecimento compartilhado por você, Hector. Aos professores que marcaram minha trajetória acadêmica, agradeço pela dedicação e pelos ensinamentos transmitidos ao longo dos anos. Em especial, ao professor Dr. Adriano Soares de Oliveira Bailão, meu orientador, que foi mais do que um orientador, foi um verdadeiro amigo. Sua confiança, disponibilidade e incansável empenho foram fundamentais para meu desenvolvimento acadêmico, profissional e pessoal. Agradeço também ao Dr. Marconi Batista Teixeira, que proveu recursos financeiros e espaço para o desenvolvimento deste trabalho. Também faço agradecimentos à Dra. Heyde Francielle do Carmo França, por todas as parcerias e artigos que escrevemos juntos. Por fim, agradeço ao Instituto Federal Goiano, pelo suporte institucional e pela sólida formação desde a graduação, e ao Programa de Pós-Graduação em Ciências Agrárias (PPGCA-AGRO).

Muito Obrigado!

BIOGRAFIA DO AUTOR

Sergio Souza Novak, nascido em 7 de dezembro de 1996 em Corbélia (PR), iniciou o curso de Ciência da Computação na UNIOESTE em 2017, onde também atuou como professor de apoio a pessoas com necessidades especiais, especialmente à comunidade surda, nas áreas de Programação, Física e Matemática. Filho de Sergio Novak, mecânico, e de Rosilei de Souza Novak, doutora pela Universidade Federal do Paraná(UFPR), mudou-se com a família para Rio Verde (GO) em 2018. Nesse mesmo ano transferiu-se para o Instituto Federal Goiano (IF Goiano) para dar continuidade à graduação, período em que também lecionou cursos de Informática na Prefeitura Municipal de Rio Verde. Além de ministrar cursos na prefeitura, ainda ministrou aulas em 2019 no Programa Nacional de Acesso ao Ensino Técnico e Emprego (PRONATEC), do Governo Federal, e para o curso de Redes de Computadores e Programação de Jogos Digitais. Tendo se formado em Bacharelado em Ciência da Computação, em 2022, decidiu atuar como Desenvolvedor de Software em uma empresa de Belo Horizonte, Minas Gerais, chamada Engenharia Assistida por Computador (ENACOM), concomitantemente com a rotina de professor. Atuou como desenvolvedor, construindo sites por mais de dois anos na ENACOM, produtos voltados à otimização de processos industriais, siderurgia e logística. Durante a pandemia, apesar de não estar matriculado em nenhum curso do IF GOIANO, desenvolveu pesquisas e trabalhos sobre a COVID-19 e o funcionalismo público, graças a parcerias com pesquisadores do IF GOIANO, rendendo, então, publicações na área da Gestão Pública. Ingressou em 2024 no Programa de Pós-Graduação em Ciências Agrárias (PPGCA-AGRO), nível de mestrado, do Instituto Federal Goiano. Durante o mestrado, no Laboratório de Irrigação do Instituto Federal Goiano, desenvolveu o trabalho desta dissertação, rendendo dois artigos publicados em periódicos e mais de dez programas de computador registrados no Instituto Nacional da Propriedade Industrial (INPI). Além do tema desta dissertação, no IF GOIANO atuou em projetos em outros programas de pós-graduação. Em parceria com a Pós-Graduação em Biodiversidade e Conservação (PPGBio), publicou trabalhos em Inteligência Artificial para detecção de abelhas nativas do Cerrado e também na fruticultura. Ainda durante o mestrado, em 2025, influenciado pela família e amigos, resolveu se dedicar exclusivamente ao mestrado e à carreira de docente. Atualmente, Sergio Souza Novak é mestrandando no PPGCA-AGRO, professor da PMRV e do PRONATEC, do Governo Federal.

RESUMO

NOVAK, SERGIO SOUZA; Instituto Federal Goiano – Campus Rio Verde – GO, outubro de 2025. **Sistema de irrigação inteligente baseado em lógica fuzzy integrado com internet das coisas para a cultura do tomate cereja.** Orientador: Dr. Adriano Soares de Oliveira Bailão, Coorientador: Dr. Marconi Batista Teixeira.

Este trabalho apresenta o desenvolvimento, a implementação e a validação experimental de um Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), integrado à Internet das Coisas (IoT), projetado para otimizar o uso de água na cultura do tomate cereja (*Solanum lycopersicum* var. *cerasiforme*). O sistema visa a superar as limitações dos métodos tradicionais de irrigação por tempo fixo, que, frequentemente, resultam em sub ou super irrigação, lixiviação de nutrientes e desperdício de água. A arquitetura do SILF é composta por: i) uma rede de sensores (umidade do solo, temperatura e umidade do ar) baseada no microcontrolador ESP32; ii) um backend servidor desenvolvido em Python/Flask, que hospeda o módulo de inferência fuzzy; e iii) uma interface web (frontend em Angular) para monitoramento e configuração remota. O núcleo do sistema é um controlador fuzzy do tipo Mamdani, cujos conjuntos e regras de inferência foram definidos tendo como base consultas a um especialista em fruticultura e da literatura fisiológica da cultura. A cada hora, o sistema processa os dados ambientais em tempo real, aplica 27 regras fuzzy e, por meio do método de defuzzificação do centroide, determina o tempo ótimo de acionamento da bomba hidráulica e da válvula solenoide (0 a 60 minutos). A validação experimental foi conduzida em ambiente protegido, utilizando um delineamento em blocos casualizados em arranjo fatorial 5x3, com cinco volumes de substrato (3, 6, 9, 12 e 15 L) e três métodos de irrigação: i) tradicional (30 minutos contínuos, uma vez ao dia); ii) fracionado (6 minutos a cada 2 horas, totalizando 30 min/dia); e iii) inteligente (via SILF). Foram avaliadas 80 plantas de tomate cereja, com a drenagem (água excedente) como variável-resposta principal. Os resultados da Análise de Variância revelaram efeitos altamente significativos ($p < 0,0001$) tanto do método de irrigação quanto do volume de substrato sobre a drenagem, sendo que a interação entre esses fatores não foi significativa ($p=0,3494$), indicando robustez do SILF em diferentes condições. O teste de comparação múltipla de Tukey ($\alpha=0,05$) demonstrou que o SILF desencadeou redução estatisticamente significativa na drenagem média quando comparado aos métodos tradicionais. Não houve diferença significativa entre os dois métodos de irrigação por tempo fixo. Em termos de volume

de substrato, os vasos de 15 L apresentaram drenagem significativamente menor que os vasos de 3 L e 9 L. Conclui-se que o SILF é uma solução tecnológica viável e eficaz, capaz de economizar, em média, 200 a 300 mL de água por planta por dia, o que representa uma economia potencial de aproximadamente 24 L/dia em uma bancada com 80 vasos. O sistema combina baixo custo, escalabilidade, controle remoto e uma lógica de decisão adaptativa que imita o raciocínio especialista. O trabalho valida o potencial da integração Lógica Fuzzy-IoT para a agricultura de precisão, promovendo um uso mais sustentável dos recursos hídricos.

Palavras-chave: Sistema Especialista. Controle de Irrigação. Conservação de Água. Agricultura de Precisão. Produto IoT.

ABSTRACT

NOVAK, SERGIO SOUZA; Instituto Federal Goiano – Campus Rio Verde – GO, outubro de 2025. **Sistema de irrigação inteligente baseado em lógica fuzzy integrado com internet das coisas para a cultura do tomate cereja.** Orientador: Dr. Adriano Soares de Oliveira Bailão, Coorientador: Dr. Marconi Batista Teixeira.

This work presents the development, implementation, and experimental validation of an Intelligent Irrigation System using Fuzzy Logic (SILF), integrated with the Internet of Things (IoT), designed to optimize water use in cherry tomato cultivation (*Solanum lycopersicum* var. *cerasiforme*). The system aims to overcome the limitations of traditional fixed-time irrigation methods, which often result in under- or over-irrigation, nutrient leaching, and water waste. The SILF architecture consists of: i) a sensor network (soil moisture, air temperature, and air humidity) based on the ESP32 microcontroller; ii) a Python/Flask backend server hosting the fuzzy inference module; and iii) a web interface (Angular frontend) for remote monitoring and configuration. The core of the system is a Mamdani-type fuzzy controller whose membership functions and inference rules were defined based on consultations with a fruit-growing specialist and the physiological literature of the crop. Every hour, the system processes real-time environmental data, applies 27 fuzzy rules, and—using the centroid defuzzification method—determines the optimal activation time for the hydraulic pump and solenoid valve (0 to 60 minutes). The experimental validation was conducted in a protected environment using a randomized block design in a 5×3 factorial arrangement, with five substrate volumes (3, 6, 9, 12, and 15 L) and three irrigation methods: i) traditional (30 continuous minutes once per day); ii) fractionated (6 minutes every 2 hours, totaling 30 min/day); and iii) intelligent (via SILF). A total of 80 cherry tomato plants were evaluated, with drainage (excess water) as the main response variable. ANOVA results revealed highly significant effects ($p < 0.0001$) for both irrigation method and substrate volume on drainage, and the interaction between these factors was

not significant ($p = 0.3494$), indicating the robustness of SILF under different conditions. Tukey's multiple comparison test ($\alpha = 0.05$) showed that SILF triggered a statistically significant reduction in average drainage compared to traditional methods. No significant difference was found between the two fixed-time irrigation methods. Regarding substrate volume, 15 L pots exhibited significantly lower drainage than the 3 L and 9 L pots. It is concluded that SILF is a viable and effective technological solution, capable of saving an average of 200 to 300 mL of water per plant per day, representing a potential savings of approximately 24 L/day for a bench with 80 pots. The system combines low cost, scalability, remote control, and an adaptive decision-making logic that mimics expert reasoning. This work validates the potential of integrating Fuzzy Logic and IoT for precision agriculture, promoting more sustainable use of water resources.

Keywords: Expert System. Irrigation Control. Water Conservation. Precision Agriculture. IoT Product.

1. INTRODUÇÃO

O início dos métodos de irrigação no Brasil remonta aos processos históricos de adaptação agrícola em resposta à variabilidade hídrica, especialmente nas regiões semiáridas do Nordeste. O uso da irrigação se intensificou a partir do século XX, com políticas públicas voltadas ao combate às secas e à promoção do desenvolvimento agrícola em áreas mais vulneráveis (Levien; Figueirêdo; Arruda, 2021). Sistemas mais organizados de irrigação foram se desenvolvendo a partir das décadas de 1970, com a implantação de perímetros irrigados, principalmente pelo Departamento Nacional de Obras Contra as Secas.

Esses projetos fomentaram novas tecnologias de irrigação, como a irrigação por superfície, aspersão e, posteriormente, o gotejamento, atingindo inicialmente grandes propriedades, posteriormente se espalhando para pequenos e médios produtores. Tradicionalmente, em pequenas propriedades e em cultivos extensivos como cana-de-açúcar, milho e pastagens, tem sido utilizada a irrigação por períodos fixos, sem consideração de variáveis climáticas (de Oliveira *et al.*, 2025).

Entre as desvantagens da irrigação por tempo fixo figura a possível sub ou super irrigação, o risco de lixiviação de nutrientes, maior consumo de energia e, consequentemente, menor eficiência no uso da água. Devido a esses problemas, nos últimos anos observou-se uma transição para sistemas mais eficientes no uso da água, com maior presença de tecnologias automatizadas e agricultura de precisão, visando a aumentar a produtividade e a sustentabilidade da produção agrícola irrigada (García *et al.*, 2020).

Novas inovações, incluindo a inteligência artificial (IA), têm revolucionado a agricultura no século 21, trazendo avanços expressivos em automação, precisão e sustentabilidade para o setor agroalimentar em escala global (Nascimento *et al.*, 2025). Com destaque para a detecção de frutos (Araújo *et al.*, 2025), de pragas (Almeida *et al.*, 2024) e até de agentes polinizadores de certas culturas (França *et al.*, 2025). Considerando a irrigação, surgiram novos tipos de manejos, agregando o uso de Algoritmos de aprendizado de máquina e modelos preditivos, estimando necessidades específicas de água para cada cultura, integrando informações locais do solo e previsão

do clima para ajustes automáticos e sistemas de irrigação autônomos, baseados em Lógica Fuzzy (Liu; Zhao; Rezaeipanah, 2025). De forma geral, essas abordagens visam a uma maior produtividade agrícola, pelo uso de informações personalizadas e automatizadas para cada lote e cultura (Silva, 2023).

O trabalho de García et al. (2020) denota tendências em métodos de irrigação considerando dispositivos de Internet das coisas (IoT) e expõe que a maioria dos artigos na literatura propõe o uso de bombas e válvulas para distribuir a água, em conjunto com sensores, para medir parâmetros ambientais, a fim de calcular as necessidades hídricas. Com relação à automação, há uma tendência no uso de placas e microcontroladores como ESP32 e Arduino, pois essas placas desempenham papel fundamental na democratização de sistemas de irrigação inteligente baseados em IoT, em razão do seu baixo custo e versatilidade. Esses dispositivos permitem que sensores de umidade, temperatura e outros parâmetros ambientais sejam facilmente conectados, coletando dados em tempo real e viabilizando automação no gerenciamento hídrico, especialmente para pequenos produtores que antes não podiam arcar com sistemas comerciais mais caros (Rissino; Rocha; Correia, 2016).

Além do baixo custo, outra vantagem do uso dessas placas é a coleta de dados ambientais, permitindo a análise de grandes volumes de dados ambientais, através de sistemas na web, como, por exemplo, no trabalho de Rissino, Rocha e Correia (2016), que utilizaram uma coleção de algoritmos para otimizar a irrigação de culturas em tempo real.

Com relação ao método computacional, que calcula o tempo de irrigação necessário, o método de decisão via Lógica Fuzzy é muito popularizado e não necessita de grandes bancos de dados experimentais para sua implementação. Tendo em vista essa vantagem, este trabalho, por meio de consultas a um especialista de fruticultura do Instituto Federal Goiano, foi modelado em parâmetros ambientais segundo o modelo computacional - Lógica Fuzzy. Além disso, foi estabelecido um conjunto de regras tendo como base aspectos fisiológicos do tomate cereja. A cultura irrigada escolhida foi o tomate cereja, nessa cultura, a água é parte central do desenvolvimento, e a irrigação adequada ao longo do ciclo sustenta o crescimento, a produção e a qualidade dos frutos.

A construção das regras foi possível graças aos avanços de dados experimentais e de estresse para a cultura do tomate, que delimitaram as condições de temperatura e umidade do ar e de umidade de solo (Silva et al., 2013). Graças a esses trabalhos, foi possível introduzir nas regras do sistema, por exemplo, a temperatura ideal (21°C a 24°C), elevada (32°C até 35°C), entre outras condicionantes. Após a construção do modelo de inferência, foi feita a validação em tempo real, por meio da variável alvo drenagem do sistema. Buscou-se reduzir a drenagem, pois esse fenômeno ocorre quando a irrigação não é adequada ao tipo de solo ou às exigências da cultura, resultando em infiltração excessiva além da zona radicular das plantas (Baêta dos S. et al., 2021).

O Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), software registrado no Instituto Nacional de Propriedade Intelectual (INPI), sob o número BR 51 2025 001780-4 (Bailão et al., 2025), foi comparado com técnicas tradicionais de irrigação, como a irrigação uma vez ao dia e a irrigação em intervalos regulares (durante o dia). Para essas análises, foram comparados diferentes tratamentos, segundo o volume de substrato. As análises produzidas foram capazes de definir o melhor volume de substrato a ser utilizado, além de representar estatisticamente a perda de água por drenagem média de cada vaso do experimento.

2. OBJETIVOS

2.1. Objetivo Geral

O objetivo deste estudo foi construir e validar um Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), baseado em Internet das Coisas, visando à redução do desperdício de recursos hídricos via drenagem.

2.2. Objetivos Específicos.

- Modelar a primeira parte do SILF, denominada definição dos Conjuntos Fuzzy, através de consultas com especialistas;
- Definir em aspectos computacionais, a segunda parte do SILF, definição de Regras, que são características fisiológicas da cultura do tomate cereja, utilizando a literatura experimental da cultura;
- Construir uma interface gráfica de um site na internet, para monitoramento e ajuste dos parâmetros do SILF;
- Estabelecer uma prova de conceito do uso dos protocolos de comunicação utilizados;
- Aplicar metodologias experimentais para avaliar qual o melhor volume de substrato para este manejo; e
- Aplicar metodologias experimentais para avaliar qual o melhor método de irrigação.

3. MATERIAL E MÉTODOS

3.1. Delineamento Experimental

Para cultivo de tomate cereja e validação do Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), foi construída uma área experimental, ilustrada na Figura 1, com plantas de tomate cereja. A foto foi tirada no dia 28 de maio de 2025 no Laboratório de Irrigação do Instituto Federal Goiano, campus Rio Verde. No experimento, há 4 fileiras de 20 plantas de tomate cereja cada uma, totalizando 80 plantas. O ambiente do experimento é coberto, e a irrigação ocorre por gotejamento.



Figura 1. Foto do SILF. Fonte: autores.

A escolha do gotejamento foi feita em razão da economia de água, pois esse método melhora a qualidade e o teor de sólidos solúveis dos frutos e promove maior uniformidade na irrigação, o que contribui para o crescimento equilibrado das plantas

(ALMEIDA ROQUE *et al.*, 2022). A Figura 2 ilustra o delineamento experimental adotado, o Delineamento em Blocos Casualizados (DBC), sendo cada parcela composta por 4 vasos, e o volume de substrato indicado nas cores. As amostras coletadas estão circuladas, indicando seu posicionamento no experimento. O delineamento seguiu um arranjo fatorial 5x3, tendo sido considerados cinco volumes de substrato, em litros: 3L, 6L, 9L, 12L e 15L. Ressalta-se que o número de unidades experimentais não corresponde a 80, mas a 20, uma vez que cada unidade é composta por quatro vasos submetidos ao mesmo tratamento, o que assegura as repetições necessárias ao ensaio. Dessa forma, foram estabelecidas quatro repetições, correspondentes às quatro linhas de vasos.

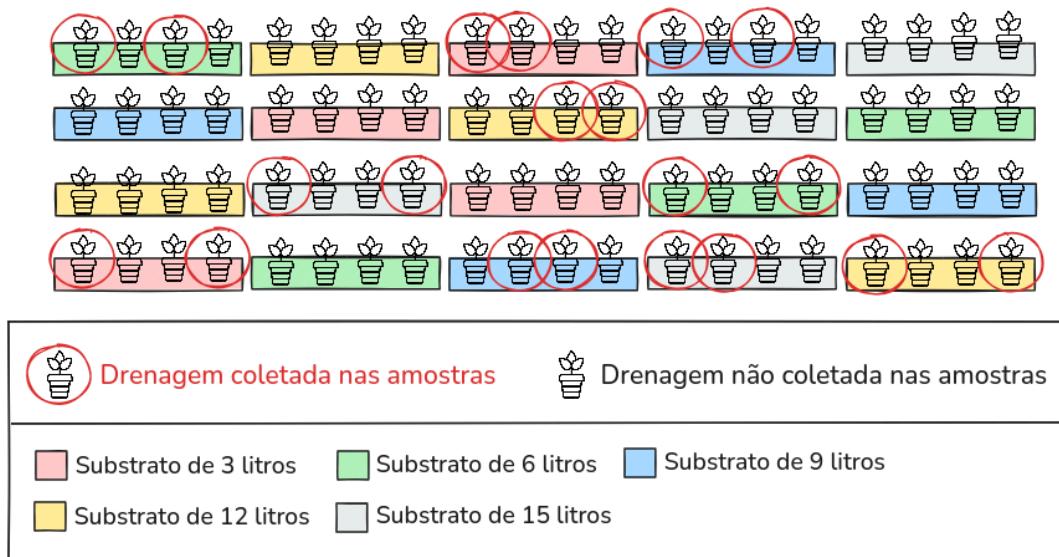


Figura 2. Visualização esquemática do delineamento experimental. Fonte: autores.

Tabela 1. Graus de liberdade do experimento. Fonte: autores.

Item	Valor
Número de parcelas	20
Graus de liberdade totais	19
Graus de liberdade dos tratamentos	4
Graus de liberdade dos blocos	3
Graus de liberdade do erro	12

Considerando o sistema envolvido nesse trabalho, é necessário avaliar os Graus de Liberdade (Tabela 1) que estão relacionados ao número total de observações feitas para cada combinação desses tratamentos, subtraindo o número de parâmetros estimados no modelo. A correta identificação e o uso dos graus de liberdade em delineamentos experimentais asseguram a precisão dos testes estatísticos e a confiabilidade da análise, em que a variabilidade de fatores pode ser elevada (Banzatto; Kronka, 2013). No caso deste trabalho, o Grau de Liberdade do Erro está adequado, tem valor 12. Quando há poucos graus de liberdade para o erro, a estimativa da variância residual torna-se imprecisa, o que dificulta a detecção estatística de diferenças reais entre os tratamentos e aumenta o risco de erro tipo II. Esses valores seguem as recomendações para delineamentos fatorial e de blocos casualizados. A variável alvo do experimento é a drenagem, sua medição foi feita por recipientes do tipo garrafa pet em cada um dos vasos do conjunto de amostra. Após a coleta, foi utilizada uma proveta graduada (ou cilindro graduado), que é um instrumento de laboratório usado para medir o volume de líquidos.



Figura 3. Fotos do sistema de drenagem. Fonte: autores.

Através do sistema de drenagem (Figura 3), foram coletados os resultados de 7 dias para cada método de irrigação: a irrigação uma vez ao dia, a irrigação a cada duas horas e a irrigação utilizando o Silf. Outras fotos do sistema de drenagem podem ser conferidas na seção de Apêndices, Figura A.

A drenagem é consequência direta da irrigação feita sobre a cultura, e segundo a literatura, a irrigação do tomate cereja diária deve ser de 1,5 a 2,5 litros por planta (Santos et al., 2023). Dessa forma, em média, uma planta consome 2 litros de água de irrigação por dia. Foi medida, experimentalmente, a vazão do experimento pelos gotejadores, um gotejador irriga em média 70 mililitros em 1 minuto com a bomba

ligada, isso totaliza 4,2 litros hora. Assim, cerca de 30 minutos diários são necessários para uma irrigação ininterrupta.

Portanto, para comparação foram analisados três manejos de irrigação diferentes: o primeiro, um manejo de irrigação de uma vez ao dia, durante trinta minutos; o segundo, um manejo de irrigação distribuindo esse consumo em intervalos de seis minutos a cada duas horas; e o terceiro e último, que é o método de irrigação desse trabalho, a irrigação pelo Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), que calcula o tempo de irrigação a cada hora do dia, segundo variáveis ambientais. A Figura 4 ilustra uma visualização esquemática do arranjo fatorial, com as variáveis que irão conferir variação ao experimento.

Arranjo fatorial 5x3

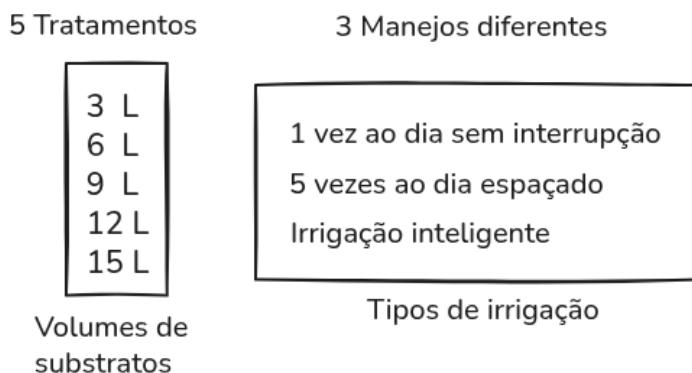


Figura 4. Visualização esquemática dos Tratamentos e Manejos que compõem o Arranjo Fatorial deste trabalho. Fonte: autores.

3.3. Características do Solo

A Tabela 2 mostra o resultado da análise de solo para o ambiente do experimento, na Fazenda Experimental do Instituto Federal Goiano, campus Rio Verde. A coleta dos dados do solo para amostra foi feita no dia 10 de março de 2025, data em que as mudas foram transplantadas para os vasos.

Tabela 2. Laudo de Programa de Qualidade de Análise de Solo, incluindo Análise Básica, referente ao ano de 2025 da Fazenda Experimental do Instituto Federal Goiano, em Rio Verde, Goiás. Fonte: autores.

Propriedade (Unidade)	Resultado	Interpretação
P-resina (mg/dm ³)	5	Baixo
MO (g/dm ³)	38	Alto
pH (CaCl ₂) (pH)	5,6	Alto
K (Potássio) (mmolc/dm ³)	1,6	Médio

Ca (Cálcio) (mmolc/dm ³)	26	Alto
Mg (Magnésio) (mmolc/dm ³)	15	Alto
H+Al (Ac. Potencial) (mmolc/dm ³)	25	Alto
Al (Alumínio) (mmolc/dm ³)	0	Médio
SB (Soma de Bases) (mmolc/dm ³)	42,6	Alto
S-SO ₄ (Enxofre) (mg/dm ³)	12	Alto
CTC (Cap. Troca Cat.) (mmolc/dm ³)	67,6	Alto
V (Sat. Bases) (%)	63	Médio
m (Sat. Alumínio) (%)	0	Médio
Ca/CTC (%)	38	Alto
Mg/CTC (%)	22	Alto
B (Boro mg/dm ³)	0,29	Médio
Cu (Cobre mg/dm ³)	2,4	Alto
Fe (Ferro mg/dm ³)	19	Alto
Mn (Manganês mg/dm ³)	10,7	Alto
Zn (Zinco mg/dm ³)	0,3	Baixo

Com essas características, o solo é apto para o cultivo de tomate, contudo é necessário avaliar sua granularidade para conhecer suas propriedades físicas, como retenção de água, drenagem, fertilidade e compactação (ABNT, 2016). A Tabela 3 apresenta a Análise Granulométrica do Solo, coletado no mesmo dia e local da Análise de Solo Básica (tabela anterior).

Tabela 3. Laudo de Programa de Qualidade de Análise de Solo da Análise Granulométrica referente ao ano de 2025 da Fazenda Experimental do Instituto Federal Goiano, em Rio Verde, Goiás. Fonte: autores.

Amostra	Identificação	Profundidade (cm)	Argila (g/kg)	Areia Total (g/kg)	Silte (g/kg)
1	Latossolo vermelho	0-20	398	443	159
2	Solo experimental vaso	0-25	375	446	178

Foram coletadas amostras do solo presente no Laboratório de Irrigação (amostra 1), bem como dos vasos que compõem o delineamento experimental (amostra 2). Com essa proporção, o solo apresenta textura franco-argilosa (franco argiloso), sendo um solo equilibrado, com boa capacidade de retenção de água. Ele apresenta também drenagem intermediária, não encharca facilmente, nem seca rapidamente. Tem ainda baixa plasticidade e pegajosidade. Já o solo da amostra 2 é aquele utilizado nos vasos.

Esse solo é um pouco mais leve, mais drenado e adequado ao cultivo em recipiente. Considerando o Silte, a amostra 2 é um pouco mais fértil que a amostra 1. Contudo, em outras características, as amostras são parecidas em sua granularidade.

3.4. Fluxo Experimental

Além de ter delineamento experimental em DBC, o experimento seguiu o seguinte cronograma de desenvolvimento: na Tabela 4 o cronograma cumprido de montagem do experimento e na Tabela 5 o cronograma da coleta de resultados.

Tabela 4. Cronograma de estágios do experimento.

Etapa	Início	Fim
Preparação das mudas	10/03/2025	03/04/2025
Desenvolvimento Vegetativo e Reprodutivo	03/04/2025	01/07/2025
Formação de frutos	01/07/2025	20/07/2025
Colheita dos frutos e Coleta de resultados	20/07/2025	11/08/2025

Tabela 5. Cronograma de coleta de resultados.

Etapa	Início	Fim
Coleta de resultados de irrigação 1 vez ao dia	20/07/2025	27/07/2025
Coleta de resultados de irrigação 5 vezes ao dia	27/07/2025	03/08/2025
Coleta de resultados do sistema de irrigação inteligente	03/08/2025	10/08/2025

Na literatura, os estágios de formação vegetal seguem Dias Após a Semeadura (DAS). Segundo f das f ds, a germinação das sementes ocorre entre 5 e 10 DAS, e a muda estará pronta para plantio entre 20 e 30 DAS. O experimento conduziu o plantio em 24 DAS, faixa correta para esse plantio. Durante o crescimento vegetativo e início da formação de flores (estágio reprodutivo), estimado até no máximo com término entre 70 e 90 DAS, foi utilizado manejo de irrigação de 5 vezes ao dia, perfazendo 2 litros de irrigação, o que é indicado pela literatura. Em 89 DAS, teve início o período de frutificação, tendo sido necessários 22 dias para garantir que os frutos fossem

adequadamente formados, garantindo que diferentes estágios da frutificação não causassem problemas na coleta dos resultados.

3.2. Dispositivos de internet das coisas e seu posicionamento

Segundo Carr et al. (2023), a Internet das Coisas (IoT) é um conceito que se refere à conexão de objetos físicos do dia a dia à Internet, permitindo que esses objetos se comuniquem entre si e possam ser monitorados e controlados remotamente. Essa conexão possibilita a automação e a simplificação de atividades rotineiras, trazendo eficiência e visualização em tempo real de dados coletados por sensores, como temperatura, umidade, entre outros. A IoT integra o mundo físico ao virtual, gerando e transmitindo grandes volumes de dados para diversas aplicações, desde ambientes residenciais até industriais e agrícolas. **Neste trabalho, a internet das coisas é o grupo de equipamentos, composto por placas e circuitos digitais, sensores e outros equipamentos periféricos associados.**

A Figura 5 ilustra uma visão geral do posicionamento dos equipamentos de internet das coisas utilizados neste trabalho. Esses equipamentos serão divididos em dois grupos: Circuito de Leitura e Circuito de Açãoamento. Ambos os circuitos são protegidos por uma estrutura fechada, que promove o isolamento do SILF a intempéries do tempo, que poderiam oxidar sensores, placas e módulos.

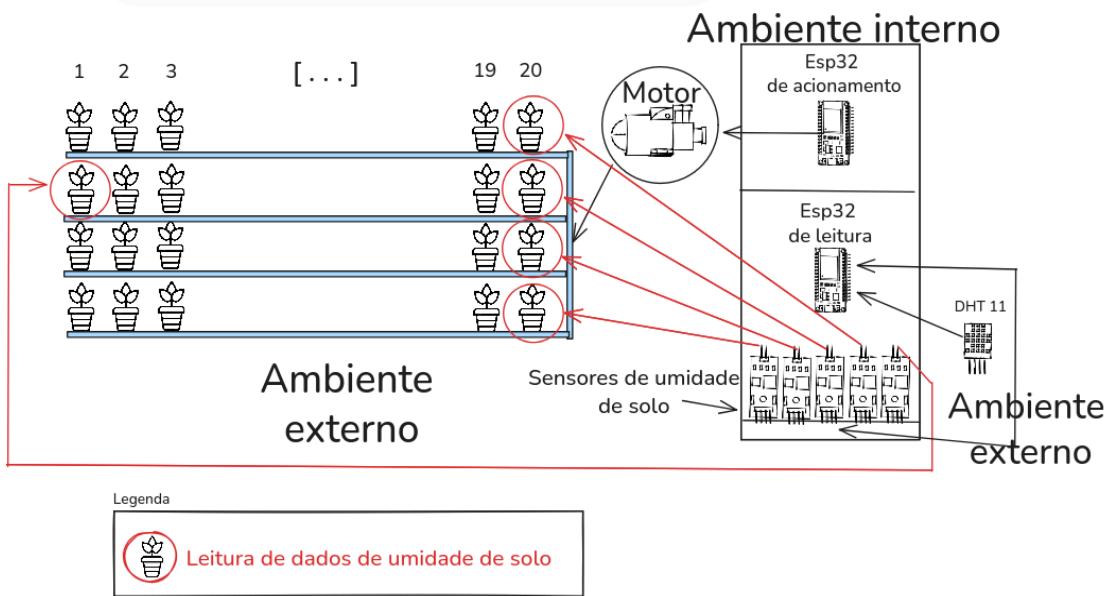


Figura 5. Visualização esquemática do experimento. Fonte: autores.

A Figura 6 é a ilustração do modelo 3D para melhor visualização. Observe que há uma separação espacial entre o circuito de açãoamento (caixa com tamanho maior) e circuito de leitura (caixa com tamanho menor). Esses dois circuitos são conectados à internet, por isso conseguem trabalhar em conjunto, apesar de estarem em caixas separadas. Há também na caixa menor, um roteador de internet, possibilitando uma automação controlada remotamente via internet. Detalhes acerca dos dois circuitos, sensores e atuadores serão abordados a seguir.

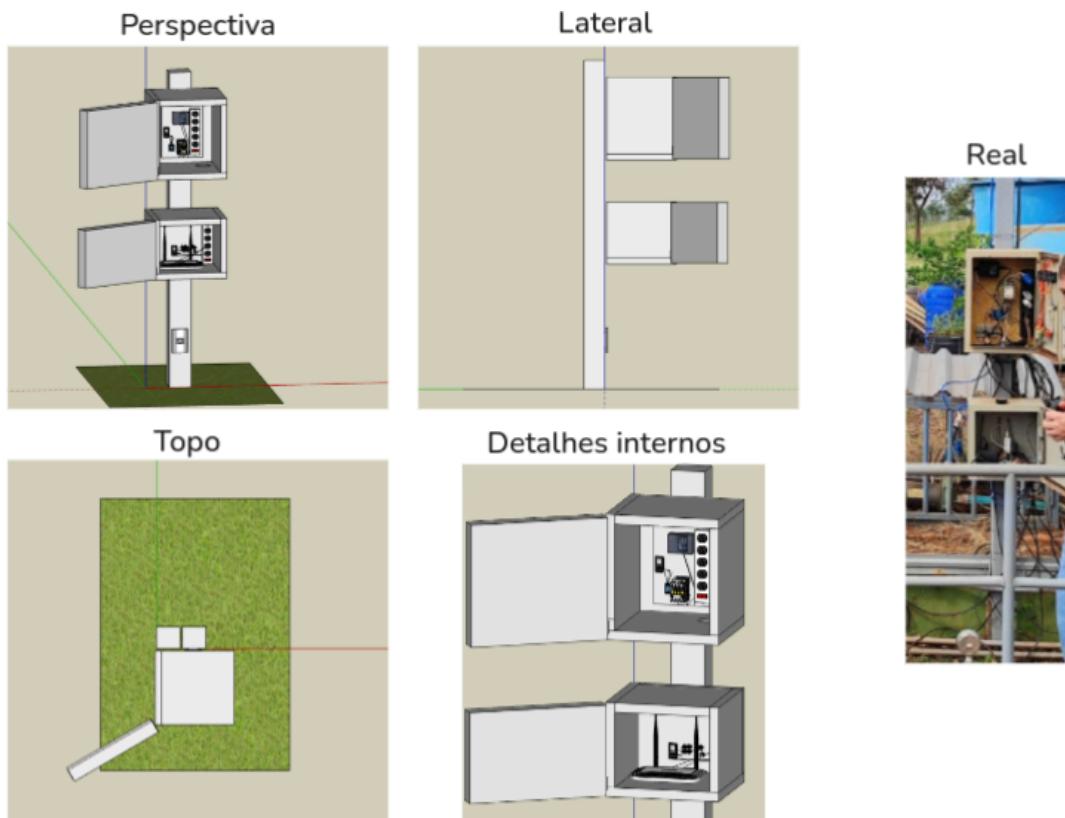


Figura 6. Modelo 3D da estação de controle do experimento, juntamente com sua comparação, foto real. Fonte: autores.

3.2.1. Microcontrolador ESP32

Segundo Oliveira e Dusse (2024), o ESP32 é um microcontrolador de baixo custo com conectividade Wi-Fi e Bluetooth integrada, amplamente utilizado em aplicações de IoT. Ele está ilustrado na Figura 7.



Figura 7. ESP32, o microcontrolador utilizado neste trabalho. Fonte: autores.

Essa placa permite conectar dispositivos à internet para coletar, processar e transmitir dados, sendo usado em projetos como automação residencial, monitoramento ambiental, sistemas agrícolas inteligentes e muitos mais. O ESP32 é popular por seu

desempenho, eficiência energética e versatilidade, suportando linguagens de programação como Python e Linguagem C, oferecendo também uma rica biblioteca de recursos para desenvolvedores (Ionescu; Enescu, 2020). Neste trabalho, a escolha do ESP32, em detrimento de outras placas, como Arduíno, por exemplo, foi por conta de seu suporte nativo a Wi-Fi, sendo-lhe possível então se conectar com a internet sem precisar de adaptadores.

3.2.2. Sensores

Neste trabalho, conforme ilustrado no esquema da Figura 2, foram utilizados cinco sensores de umidade de solo e um sensor de umidade e temperatura do ar do tipo DHT11. A escolha do sensor DHT11 (Figura 8) deve-se ao fato de ser amplamente empregado na medição de temperatura e umidade, destacando-se pela simplicidade de operação e pelo baixo custo.

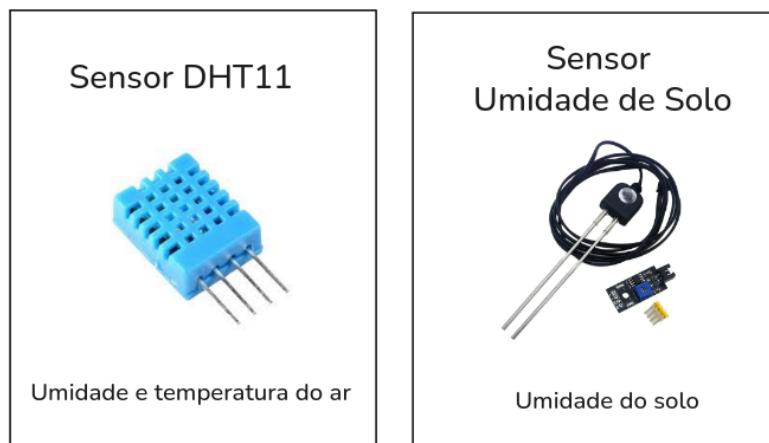


Figura 8. Sensores utilizados no trabalho. Fonte: autores.

Este sensor de temperatura e umidade do ar tem faixa de medição de 0 a 50 °C para temperatura e de 20% a 80% para umidade relativa, com precisão de ± 2 °C e $\pm 5\%$, respectivamente (Oliveira Júnior; Vicentin; Cunha, 2014). Foi utilizado também um resistor *pull-up* para a leitura dos dados do sensor DHT11, conforme indicado na literatura por Alfith *et al.*(2022).

Também foi empregado um sensor de umidade do solo do tipo anticorrosivo — característica que contribui para a maior durabilidade do equipamento — destinado à obtenção de leituras diretamente no solo. A fabricante do sensor é a empresa WJ Componentes Eletrônicos. Ao todo, foram utilizados cinco sensores desse modelo para as medições, sendo que esse sensor tem um módulo que converte a leitura analógica

que o sensor recebe para dois tipos de saída, digital e analógica, adaptando essas leituras às entradas que o ESP32 tem.

3.2.3 Circuito de leitura dos dados

No SILF, como ambas as placas ESP32 estão conectadas à internet, a comunicação entre as placas de leitura e o acionamento de motores ocorre a cada instante. A visualização do circuito de leitura está presente na Figura 9. Nesse fluxo, os dados de umidade de solo são lidos em 5 sensores de umidade de solo, contudo, para fins didáticos está presente na Figura 9 um esquema de circuito de um sensor DHT11 em conjunto com o sensor de umidade de solo.

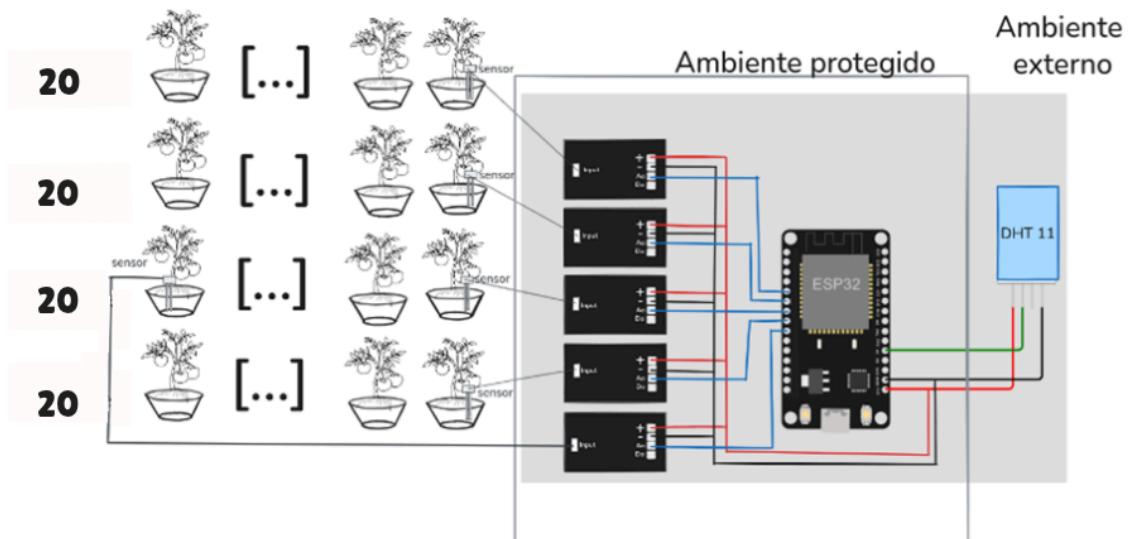


Figura 9. Circuito de ligação dos cinco sensores de umidade de solo e do sensor DHT11. Fonte: autores.

Após a leitura pela placa ESP32, é feito um tratamento para verificar se o sensor está realmente ligado e coletando dados corretamente, em seguida, é feita a média considerando apenas os sensores que estiverem ligados. Depois, a média e o valor de umidade coletado em cada sensor são enviados a uma API Python Flask na internet, a fim de persistir esses dados em um banco de dados relacional MySQL. Os dois sensores não precisam de quaisquer equipamentos transformadores para funcionar, isso ocorrendo por trabalharem na voltagem em que o microcontrolador ESP32 opera: 3,3V.

Para obtenção de um valor de umidade de solo confiável, neste trabalho foram desenvolvidos códigos de programação para detectar se os sensores de umidade de solo estão funcionando, dessa forma, após verificar quais sensores estão realmente ligados, é feita a média dos valores dos que estão ligados.

3.2.4. Circuito de Acionamento da Bomba

Para a irrigação do sistema, foram utilizadas uma bomba hidráulica e uma válvula solenoide, equipamentos ilustrados na Figura 10.



Figura 10. Atuadores do SILF. Fonte: autores.

No SILF, a bomba escolhida foi a Bomba Periférica BB 500P da fabricante BRANCO-9036, de 220 V, em conjunto com a válvula solenoide Hunter de 24 V. O uso da válvula solenoide é requerido em razão da ação da força gravitacional da coluna d'água do reservatório. A pressão hidrostática é diretamente proporcional à altura do reservatório, por consequência, a válvula solenoide oferece resistência à passagem de água na tubulação (Halliday; Resnick; Walker, 2023).

A válvula solenoide escolhida opera em uma voltagem de 24 V de corrente alternada, dessa forma não suporta ser ligada em paralelo com o motor, sendo necessário um transformador para essa válvula. O microcontrolador, através do relé, consegue controlar o acionamento do motor, e como esta placa é ligada à internet, faz consultas a cada segundo a um software (API Flask, que será descrito nas próximas sessões), a fim de buscar o estado lógico da bomba no SILF (ligada ou desligada). Os equipamentos Contator e Relé estão ilustrados na Figura 11.

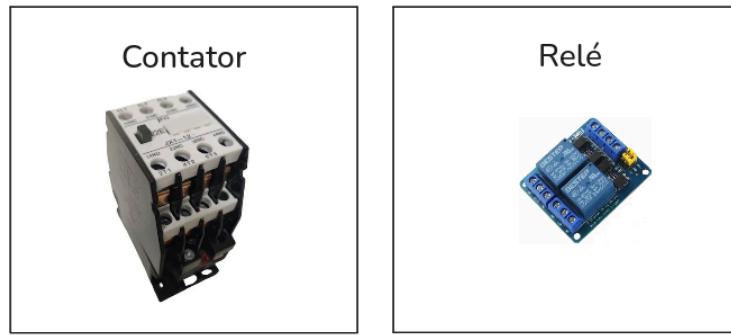


Figura 11. Contator e Relé. Fonte: autores.

De acordo com Braga (2019), a função de um contator é controlar o acionamento e o desligamento de circuitos elétricos de alta potência, como motores elétricos, aquecedores, sistemas de iluminação e outros dispositivos industriais e comerciais. Ela funciona como um interruptor eletromecânico que permite ligar ou desligar cargas elétricas a distância de forma segura e eficiente, por meio do efeito eletromagnético gerado pela bobina interna do contator. Isso possibilita o controle remoto ou automático do fluxo de energia, além de proporcionar segurança, economia de energia e maior vida útil aos equipamentos conectados.

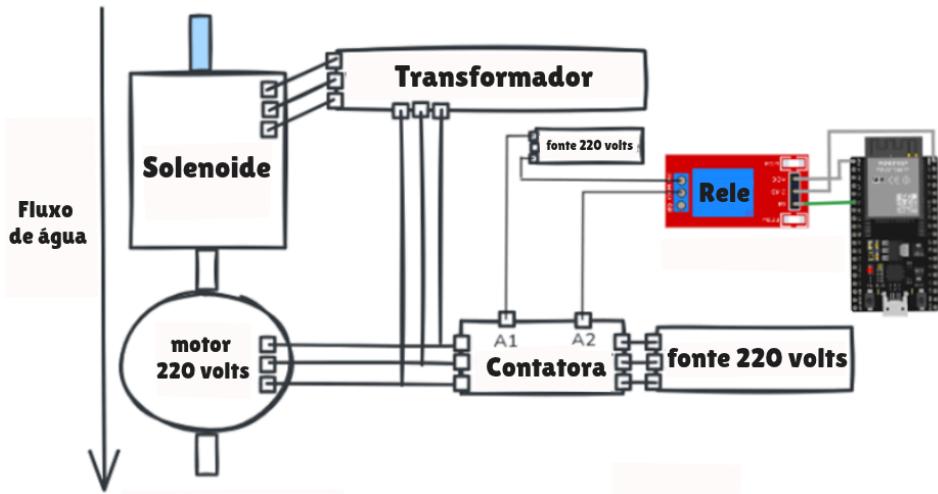


Figura 12. Circuito de acionamento do motor. Fonte: autores.

Ainda segundo Braga (2019), a função do relé é permitir que o microcontrolador controle cargas elétricas de maior potência, como motores, lâmpadas ou válvulas, isolando eletricamente o circuito de controle (3,3 V) do circuito de potência (5 V, 12 V ou 220 V). É importante utilizar uma contadora para proteger o relé, uma vez que a

potência da bomba proporciona aumento na corrente elétrica, o que poderia danificar o relé e até mesmo o microcontrolador ESP32. O circuito completo está denotado na Figura 12. Observe que o acionamento do Relé está sujeito a um sinal (fio verde) enviado pelo ESP32. Já o Relé controla diretamente as entradas A1 e A2 da contadora, dessa forma a alimentação dos dois atuadores, Bomba e Solenoide (este último via transformador), é comutada. Solenoide e motor são ligados em paralelo ao contator, dessa forma, nunca ocorrerá de a bomba ser ligada concomitantemente ao solenoide fechado, condição que poderia reduzir a vida útil do motor.

3.3. Lógica Fuzzy

A lógica fuzzy é o método utilizado neste trabalho para tomada de decisão no manejo da irrigação automatizada. Ela também é conhecida como lógica difusa, trata-se de um sistema lógico desenvolvido para lidar com a incerteza e a imprecisão presentes em diversos fenômenos do mundo real. Diferentemente da lógica clássica ou booleana, que trabalha com valores binários — verdadeiro (1) e falso (0) —, a lógica fuzzy permite que variáveis tenham valores contínuos dentro de um intervalo entre 0 e 1, representando diferentes graus de verdade (Zadeh, 1965). Essa característica a torna especialmente útil em situações nas quais as fronteiras entre os conceitos não são claramente definidas, como "quente", "morno" ou "frio", que não têm limites exatos.

O conceito foi introduzido por Lotfi A. Zadeh, professor da Universidade da Califórnia, em 1965, por meio de seu artigo “Fuzzy Sets”. A proposta de Zadeh buscava formalizar a maneira como o raciocínio humano lida com informações vagas ou subjetivas, criando um modelo matemático capaz de descrever fenômenos complexos de forma aproximada (Zadeh, 1965).



Figura 13. Desenho esquemático do modelo lógica fuzzy. Fonte: autores.

Assim, a lógica fuzzy é amplamente aplicada em áreas como automação industrial, controle de processos, inteligência artificial e sistemas especialistas (ROSS, 2010). O funcionamento da lógica fuzzy baseia-se em três etapas principais: fuzzificação, inferência fuzzy e defuzzificação (Ross, 2010; Simões; Shaw, 2007). Esses processos estão diagramados na Figura 13.

Fuzzificação: nesta etapa, as variáveis de entrada — geralmente valores numéricos obtidos de sensores ou de medições — são convertidas em valores fuzzy, ou seja, em graus de pertinência a determinados conjuntos fuzzy. Por exemplo, a temperatura de 28°C pode ter grau de pertinência 0,8 ao conjunto “quente” e 0,2 ao conjunto “morno”. Essa transformação é feita por meio de funções de pertinência, que podem assumir formas triangulares, trapezoidais, gaussianas, entre outras (Silva; Nogueira; Santos, 2019).

Inferência Fuzzy: Após a fuzzificação, o sistema aplica um conjunto de regras linguísticas do tipo “Se... então...”. Por exemplo: “Se a temperatura é alta e a umidade é baixa, então o ventilador deve girar rapidamente”. Essas regras são processadas por um mecanismo de inferência, que combina os graus de pertinência e calcula o resultado fuzzy correspondente (Simões; Shaw, 2007). Esse processo busca imitar o raciocínio humano ao tomar decisões baseadas em expressões linguísticas, não apenas em equações matemáticas exatas.

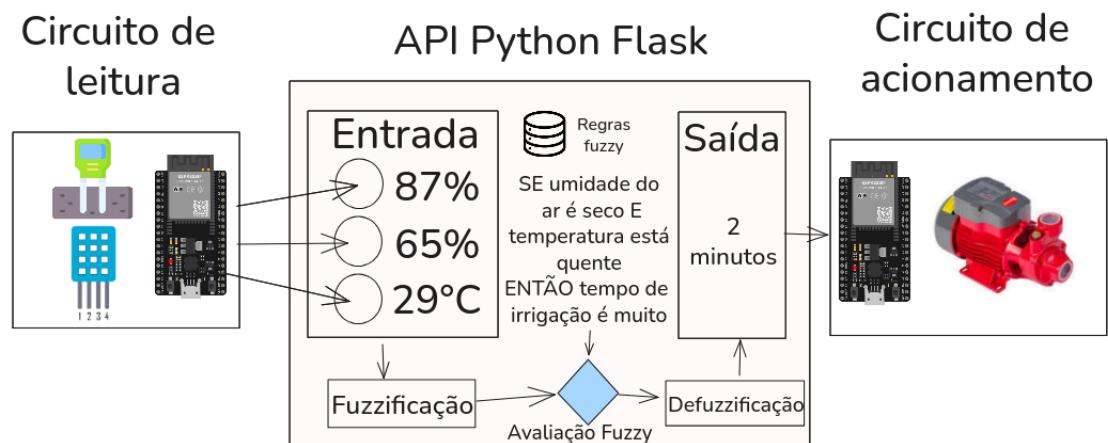


Figura 14. O processo de inferência Fuzzy. Fonte: autores.

Por fim, o resultado fuzzy é convertido em um valor numérico preciso, capaz de ser interpretado ou utilizado por sistemas eletrônicos e controladores. Os métodos mais comuns de defuzzificação incluem o centro de gravidade (centroid), média dos máximos e máximo do máximo (Ross, 2010). O resultado dessa etapa é uma saída que traduz a decisão do sistema fuzzy em termos quantitativos, como, por exemplo, 2 minutos de irrigação.

Neste trabalho, o sistema de irrigação toma decisões com base em valores de umidade do solo, umidade do ar e temperatura do ar. Com base nessas três variáveis, o

SILF determina, em intervalos fixos de uma hora, o tempo de acionamento da bomba, expresso em minutos. Uma representação esquemática desse processo decisório é apresentada na Figura 14.

O modelo recebe como entrada três valores reais, por exemplo, (87%, 65%, 29 °C). Tendo esses dados como referência, o SILF infere um tempo de irrigação na faixa de 0 a 60 minutos. Inicialmente, é feito o processo de fuzzificação das variáveis de entrada, conforme os conjuntos fuzzy definidos. Em seguida, ocorre a avaliação fuzzy, conduzida por meio de regras do tipo SE... ENTÃO. Por fim, aplica-se o processo de defuzzificação, que gera como saída o tempo de irrigação, em minutos, a ser enviado ao circuito de acionamento. Detalhes sobre os processos de fuzzificação e de defuzzificação podem ser encontrados no trabalho de Ojha, Abraham e Snasel (2019).

A construção do sistema de decisão para irrigação foi feita segundo o método de Mamdani (Ojha; Abraham; Snasel, 2019). Inicialmente, foram conduzidas entrevistas com especialistas na cultura do tomate cereja, a partir das quais foram definidos os conjuntos fuzzy referentes ao tempo de irrigação, à umidade do solo, à umidade do ar e à temperatura do ar. Para esse processo, foi entrevistado um especialista em fruticultura do Instituto Federal Goiano – Campus Rio Verde.

3.3.1. Conjuntos Fuzzy

Neste trabalho, os Conjuntos Fuzzy foram produtos diretos da entrevista com um doutor em Agronomia e especialista em fruticultura do Instituto Federal Goiano, campus Rio Verde. Com respaldo na resposta das perguntas feitas ao agrônomo, foi possível modelar os conjuntos do Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF). As questões foram questões práticas como “Qual faixa de temperatura é muito quente para a cultura do tomate cereja?” ou perguntas como “Qual seria a faixa de umidade considerada normal para a cultura?”. Depois de respondidas as perguntas, foram definidas as funções presentes na Tabela 6 e, graficamente, na Figura 15. Esses conjuntos de funções são chamados de Conjuntos Fuzzy.

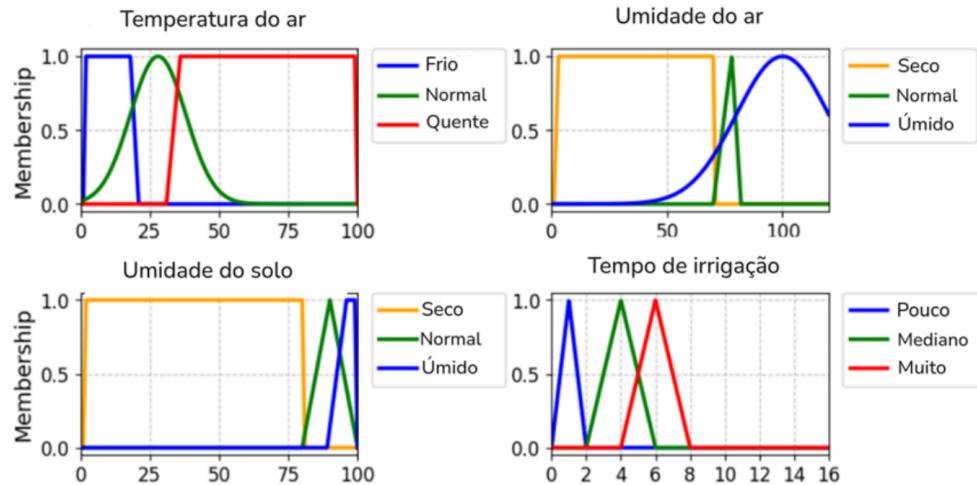


Figura 15. Os conjuntos fuzzy definidos neste trabalho. Fonte: autores.

Tabela 6. Conjuntos Fuzzy definidos para o SILF. Fonte: autores.

Variável	Classe	Função	Parâmetros	Intervalo
Temperatura do Ar	Frio	Trapezoidal	$a = 1, b = 2, c = 18, d = 21$	[1, 21]
Temperatura do Ar	Normal	Gaussiana	$\mu = 28, \sigma = 10$	$(-\infty, +\infty)$
Temperatura do Ar	Quente	Trapezoidal	$a = 31, b = 36, c = 99, d = 100$	[31, 100]
Umidade do Ar	Seco	Trapezoidal	$a = 1, b = 3, c = 70, d = 71$	[1, 71]
Umidade do Ar	Normal	Triangular	$a = 70, b = 78, c = 82$	[70, 82]
Umidade do Ar	Úmido	Gaussiana	$\mu = 100, \sigma = 20$	$(-\infty, +\infty)$
Umidade do Solo	Seco	Trapezoidal	$a = 1, b = 2, c = 80, d = 81$	[1, 81]
Umidade do Solo	Normal	Triangular	$a = 80, b = 90, c = 100$	[80, 100]
Umidade do Solo	Úmido	Trapezoidal	$a = 89, b = 96, c = 99, d = 100$	[89, 100]
Tempo de Irrigação	Pouco	Triangular	$a = 0, b = 1, c = 2$	[1, 2]
Tempo de Irrigação	Mediano	Triangular	$a = 2, b = 4, c = 6$	[2, 6]
Tempo de Irrigação	Muito	Triangular	$a = 4, b = 6, c = 8$	[4, 8]

Conjuntos fuzzy, ou conjuntos difusos, são uma generalização dos conjuntos clássicos que permitem tratar a incerteza e a imprecisão na classificação de elementos. Diferentemente dos conjuntos tradicionais, em que um elemento pertence ou não pertence ao conjunto (valores 0 ou 1), nos conjuntos fuzzy a pertença é gradual,

representada por um valor numérico entre 0 e 1, chamado grau de pertinência. Isso significa que um elemento pode pertencer a um conjunto fuzzy em diferentes níveis, refletindo a ambiguidade ou a incerteza inerente a muitas situações do mundo real (Zadeh, 1965).

Estes conjuntos são utilizados para modelar fenômenos em que os limites são imprecisos, como em lógica fuzzy aplicada a sistemas de controle, tomada de decisão, inteligência artificial e outras áreas em que dados incertos ou subjetivos precisam ser interpretados. A teoria dos conjuntos fuzzy foi proposta por Lotfi Zadeh em 1965, sendo fundamentada em funções de pertinência que quantificam o grau de pertencimento de elementos a um conjunto específico.

Nesse contexto, os conjuntos fuzzy representam categorias linguísticas que traduzem variáveis contínuas em faixas interpretáveis. A Figura 14 ilustra essa definição para o SILF. Por exemplo, no universo de temperatura do ar, definido no intervalo de 0 a 100 °C, é possível estabelecer os subconjuntos “frio”, “normal” e “quente”. Com base nas entrevistas técnicas feitas com especialistas em fruticultura, foram construídos os conjuntos fuzzy listados na Tabela 2. Essas funções referentes aos conjuntos fuzzy foram inseridas utilizando a interface gráfica do site, conforme ilustrado na Figura 16.

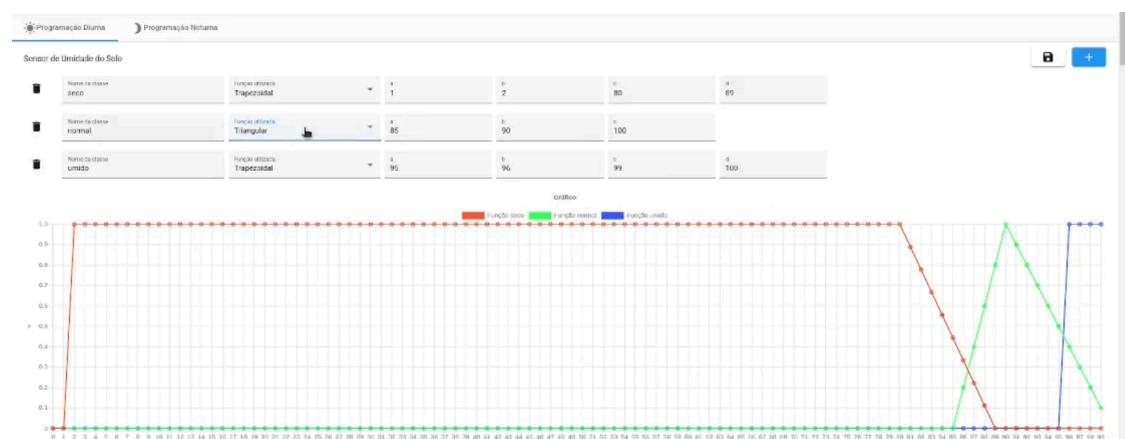


Figura 16. Interface gráfica no site para inserção de funções de pertinência dos conjuntos fuzzy.
Fonte: autores.

No site, o usuário pode adicionar uma lista de várias funções para uma mesma variável, podendo ser Gaussiana, Triangular e Trapezoidal. Funções Degrau podem ser inseridas a partir de variações nas funções Trapezoidais. Dessa forma, o sistema poderá ser utilizado para outras culturas além do tomate cereja, bastando o usuário manipular

essas funções. O profissional pode criar, excluir e alterar as funções, desde que essas funções estejam de acordo com a fisiologia da cultura.

A escolha dessas três funções possíveis ocorreu porque, além de estarem implementadas na biblioteca de programação scikit-fuzzy (ferramenta de desenvolvimento de software), as funções Triangular e Trapezoidal, por exemplo, são conhecidas por serem simples de parametrizar e rápidas de computar, conforme já relatado em outros sistemas fuzzy (Maranduba et al., 2017). Essas funções, especialmente as Triangulares e Trapezoidais, são intuitivas para especialistas, tornando mais fácil traduzir conhecimento empírico ou descrição verbal para modelos matemáticos. Por exemplo, é simples comunicar regras do tipo "umidade próxima do ideal", associando a uma função triangular ou trapezoidal.

Adicionalmente, foram incluídas no site as funções Gaussianas, pois, segundo Maranduba et al. (2017), essas funções são escolhidas quando são desejadas suavidade nos limites do conjunto fuzzy e maior robustez para modelar situações em que a transição entre categorias é progressiva, como temperatura, por exemplo. E a modelagem de normalidade de temperatura foi justamente feita através de uma função desse tipo (Figura 13).

Dos conjuntos de funções definidos, a variável de saída do SILF é o tempo de irrigação, enquanto Temperatura do Ar, umidade do ar e umidade do solo são entradas para o SILF. Por decisão de projeto, o software armazena o tempo como uma porcentagem, por exemplo, na Tabela 2, o tempo de irrigação “pouco” seria uma função triangular modelada de (0% ; 1% ; 2%). Considerando que, se o usuário definiu a escala de tempo no site de 0 até 60 minutos, esses valores são porcentagem do universo para esta variável, sendo, portanto, 0 ; 0,6 ; 1,2, assim sendo, de 0 a 0,6 minutos a função pouco subiria sua pertinência, assumindo pertencimento máximo em 0,6 minutos. Já de 0,6 a 1,2 minutos, sua pertinência seria diminuída.

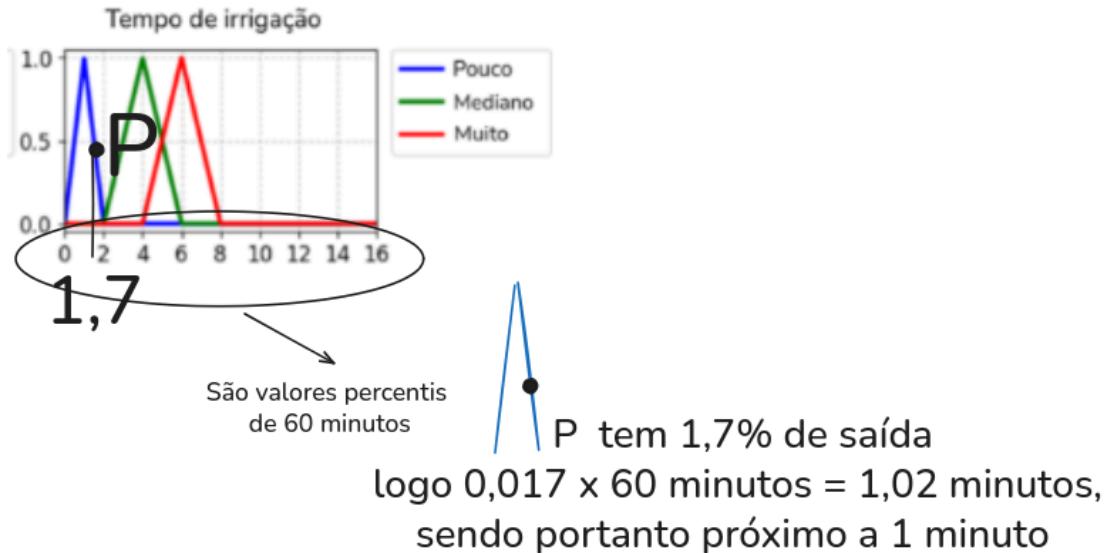


Figura 17. Processo de decodificação da saída. Fonte: autores.

O processo de inferência produz uma saída em percentis, conforme ilustrado na Figura 17. Para uma saída de, por exemplo 1,7%, a inferência produziu uma saída de um minuto de irrigação, pertencendo, portanto, ao conjunto “pouco” para tempo de irrigação. Essa inferência levou em consideração os conjuntos de entrada, temperatura do ar, umidade do ar e umidade do solo. Para fazer essa inferência, o SILF precisou, além dos Conjuntos Fuzzy, de uma outra definição por parte do usuário às Regras Fuzzy, que serão abordadas a seguir.

3.3.2. Regras Fuzzy

De forma introdutória e simples, as Regras Fuzzy são regras, como, por exemplo, “SE solo seco E ar quente E ar seco ENTÃO tempo de irrigação muito”. Apesar de ser simples pensar que o tempo de irrigação seja muito, em uma situação hipotética de um dia seco e quente de verão com solo seco, pode ser que seja necessário variar o tempo de irrigação. Pense em duas situações diferentes: às 9:30 da manhã e ao meio dia. Pode ocorrer de ao meio dia a temperatura do ar estar mais quente que o observado às 9:30 da manhã, apesar de ambas serem muito quentes. O sistema é capaz de, nesse caso, produzir um tempo de saída de 5 minutos para as 9:30 da manhã e 7 minutos para o meio dia, isso porque ele precisa calcular a proporcionalidade entre os conjuntos fuzzy, e as Regras Fuzzy são parte fundamental desse processo.

As Regras Fuzzy representam o núcleo do processo de tomada de decisão em sistemas baseados em lógica fuzzy (Almeida; Silva, 2021). Elas são fundamentais para que o modelo consiga transformar informações incertas ou subjetivas, típicas do ambiente agrícola, em ações automáticas. Em outras palavras, essas regras traduzem o conhecimento agronômico e a experiência de especialistas em proposições linguísticas, como, por exemplo, “Se o solo está seco e a temperatura está alta, então irrigar mais”. Dessa forma, o SILF é capaz de simular o raciocínio humano no processo de tomada de decisão, e o motor de inferência construído consegue calcular, de maneira proporcional, o tempo de irrigação adequado.

A Tabela 7 apresenta o conjunto de regras fuzzy definidas para o período diurno no sistema de irrigação desenvolvido. Todas as regras seguem o formato de implicação com antecedente conjuntivo. Em lógica fuzzy, o antecedente conjuntivo consiste na combinação de duas ou mais condições antecedentes por meio de operadores lógicos — geralmente o operador E (AND) — no interior de uma regra fuzzy (Mamdani; Assilian, 1975).

Um exemplo de regra com antecedente conjuntivo é “SE solo seco E ar quente E ar seco ENTÃO tempo de irrigação muito alto”. Essa estrutura permite representar de forma mais realista a interação entre múltiplas variáveis ambientais. Em situações caracterizadas por altas temperaturas e baixos níveis de umidade do solo e do ar, o SILF determina a necessidade de períodos de irrigação mais prolongados.

Nessa fase, o processo de defuzzificação da variável de saída — tempo de irrigação — converte o resultado fuzzy em um valor numérico em minutos, proporcional ao grau de pertinência associado às variáveis de entrada obtidas pelos sensores de umidade do solo e pelo sensor DHT11.

Tabela 7. Regras Fuzzy. Fonte: autores.

Regra	Temperatura	Umidade do Ar	Umidade do Solo	Tempo
1	Frio	Seco	Seco	Mediano
2	Frio	Seco	Normal	Pouco
3	Frio	Seco	Úmido	Pouco
4	Frio	Normal	Seco	Pouco

5	Frio	Normal	Normal	Pouco
6	Frio	Normal	Úmido	Pouco
7	Frio	Úmido	Seco	Mediano
8	Frio	Úmido	Normal	Pouco
9	Frio	Úmido	Úmido	Pouco
10	Normal	Seco	Seco	Muito
11	Normal	Seco	Normal	Mediano
12	Normal	Seco	Úmido	Pouco
13	Normal	Normal	Seco	Muito
14	Normal	Normal	Normal	Pouco
15	Normal	Normal	Úmido	Pouco
16	Normal	Úmido	Seco	Mediano
17	Normal	Úmido	Normal	Pouco
18	Normal	Úmido	Úmido	Pouco
19	Quente	Seco	Seco	Muito
20	Quente	Seco	Normal	Muito
21	Quente	Seco	Úmido	Muito
22	Quente	Normal	Seco	Mediano
23	Quente	Normal	Normal	Mediano
24	Quente	Normal	Úmido	Mediano
25	Quente	Úmido	Seco	Mediano
26	Quente	Úmido	Normal	Pouco
27	Quente	Úmido	Úmido	Pouco

Para proceder à inferência do tempo de irrigação por meio do método fuzzy, foi utilizada a biblioteca Scikit-Fuzzy em Python (Warner; Roberts; Contributors, 2024). Essa biblioteca implementa os processos de fuzzificação e defuzzificação necessários para gerar a saída em tempo, adotando, como metodologia de defuzzificação, o método

do Centro de Gravidade. A definição dos Conjuntos Fuzzy foi baseada em entrevistas com um especialista em fruticultura, enquanto a formulação das regras do SILF considerou aspectos fisiológicos específicos do tomateiro, presentes na literatura.

3.3.3. Regras para a temperatura fria (Regras 1 a 9)

Em condições de temperatura fria ($<15^{\circ}\text{C}$), a atividade metabólica da planta é reduzida (Taiz et al., 2015). A transpiração diminui significativamente, pois os estômatos tendem a ficar fechados para conservar água e energia (Taiz et al., 2015). No entanto, a umidade do solo ainda é crítica para evitar déficit hídrico. Considerando a primeira regra, apesar da baixa transpiração, causada pelo frio, o solo seco exige irrigação moderada para evitar estresse hídrico (Taiz et al., 2015). Plantas em solo seco podem sofrer danos radiculares mesmo em temperaturas baixas.

Já na segunda regra, estando o solo em condições normais (umidade adequada), ele supre a demanda reduzida de água em temperaturas frias (Taiz et al., 2015). A irrigação mínima é suficiente para manter o equilíbrio. Considerando a terceira regra, o solo úmido compensa a baixa umidade do ar, eliminando a necessidade de irrigação significativa. Excesso de água poderia levar à hipóxia radicular. Hipóxia é a deficiência de oxigênio nos tecidos. No caso das raízes, ocorre quando o solo fica saturado de água, reduzindo a difusão de oxigênio até as células radiculares, o que compromete a respiração celular e o funcionamento da planta (Taiz et al., 2015).

Na quarta regra, a umidade do ar normal reduz a transpiração, mas o solo seco ainda requer irrigação mínima para evitar ressecamento. A quinta regra denota situação de normalidade ambiental, condições equilibradas em todas as variáveis. A demanda hídrica é mínima. Para a regra 6, é importante considerar que o solo, já úmido, combinado com temperatura fria, torna a irrigação quase desnecessária.

Considerando a regra 7, o ar úmido reduz ainda mais a transpiração, mas o solo seco exige reposição hídrica para evitar déficit hídrico. Na construção da regra 8, há condições ideais com baixa demanda. A planta não sofre estresse hídrico ou térmico. Já na regra 9, em razão do risco alto de encharcamento, a irrigação deve ser mínima para evitar apodrecimento radicular.

3.3.4. Regras para a temperatura moderada (Regras 10 a 18)

Esse conjunto de regras denota temperaturas moderadas (15–25 °C), que representam a faixa ótima para a maioria das plantas (Taiz et al., 2015). A transpiração é equilibrada, e o balanço hídrico depende principalmente da umidade do solo e do ar. Considerando a décima regra, o ar seco aumenta a transpiração, e o solo seco agrava o estresse hídrico, assim a irrigação máxima é necessária.

Considerando a décima primeira, o solo normal atenua parcialmente o efeito do ar seco, mas ainda requer irrigação moderada. Tendo em vista a décima segunda regra, o solo úmido compensa a umidade baixa do ar, reduzindo a necessidade de irrigação. Na décima terceira regra, o solo seco é o fator limitante crítico, o que exige irrigação abundante. Analisando a regra décima quarta, há condições ideais para manutenção do balanço hídrico, demandando irrigação moderada para manter esse equilíbrio. Nas regras 15 e 17, há água suficiente no solo, dispensando irrigação significativa. Já a irrigação mediana da regra 16 ocorre, considerando que o ar úmido reduz a transpiração, mas o solo seco ainda demanda irrigação. A regra 18 infere pouca irrigação por conta do risco de hipóxia radicular.

3.3.5. Regras para a temperatura quente (Regras 19 a 27)

Em altas temperaturas (>25 °C), são aceleradas a transpiração e a evaporação do solo. A demanda por água é crítica, especialmente se combinada com solo seco ou ar seco, como ocorre na regra 19, condição de máximo estresse. Calor e ar seco aumentam a perda de água, e o solo seco não a repõe. Na regra 20, o solo normal não é suficiente para compensar a alta transpiração induzida pelo calor e pelo ar seco, demandando irrigação maior.

Na regra 21, o solo úmido atenua parcialmente o estresse, mas o calor mantém a demanda. Na regra 22, há prioridade absoluta para irrigação, pois o solo está seco e a temperatura quente. Na regra 23, o sistema deve irrigar muito, pois, apesar de as variáveis estarem normais, está muito quente. Tendo em vista a regra 24, o solo úmido reduz a necessidade, mas o calor persiste, demandando uma irrigação média. As demais regras são justificadas pelas altas temperaturas, que demandam mais irrigação para compensar a elevada transpiração da planta.

3.3.6. Método de Inferência pelo Centroide

O método do centroide na lógica fuzzy é uma técnica utilizada para proceder à defuzzificação, ou seja, converter o resultado de um sistema fuzzy (normalmente uma função de pertinência agregada) em um valor numérico único e preciso. Este processo é fundamental para transformar os dados interpretados pela lógica fuzzy em ações ou decisões concretas em sistemas reais (Leite Dias de Mattos; Coelho, 2015).

O método do centroide, também chamado de Centro de Gravidade, ou "center of gravity" em inglês, calcula o centro de massa da função de pertinência resultante da inferência fuzzy. Consiste em identificar o ponto de equilíbrio da área sob a curva da função de saída agregada, fornecendo um valor representativo para essa saída (Maria de Menezes et al., 2024). Em termos matemáticos, é obtido por:

$$x_c = \frac{\int x \cdot \mu(x) dx}{\int \mu(x) dx}$$

De forma que x_c é Centroide, ou valor defuzzificado, é o resultado final da defuzzificação, o valor “crisp” representa o centro de massa da área sob a função de pertinência de saída. Já o x representa o eixo do universo de discurso da variável de saída, que, no caso deste trabalho, é o tempo de irrigação. O μ é a função de pertinência da saída, que é o grau (0–1) que indica quanto cada tempo pertence ao Conjunto Fuzzy Tempo de Irrigação após a agregação das regras. Resultado da combinação das regras que usam temperatura, umidade do ar e umidade do solo. Representa a soma (integral) dos tempos x ponderados pelo grau de pertinência $\mu(x)$.

A integral presente no numerador faz com que valores de x com $\mu(x)$ maior “puxam” o resultado para si, e se as regras favorecerem tempos longos, esse momento aumenta, resultando em um x maior. Essa operação faz com que os valores de x , ou seja, os tempos de irrigação possíveis, com maiores graus de pertinência $\mu(x)$, exerçam maior influência sobre o resultado final. Assim, quando as regras fuzzy favorecem tempos de irrigação longos — por exemplo, em condições de alta temperatura e baixa umidade do solo — o valor do momento aumenta, deslocando o centroide (x_a) para regiões de maior duração de irrigação.

Já a integral presente no denominador corresponde à soma total dos graus de pertinência da função de saída, ou seja, à “massa” total da distribuição fuzzy obtida após a agregação das regras. Essa área tem a função de normalizar o numerador, permitindo o cálculo do ponto médio ponderado, ou seja, o valor defuzzificado que representa o tempo de irrigação final a ser aplicado pelo sistema.

3.4. Linguagens de programação, códigos, software e protocolos utilizados

Esta seção discorre sobre as linguagens de programação, software, códigos e protocolos que foram utilizados na construção do Sistema de Irrigação Inteligente por Lógica Fuzzy (SILF), e sobre a sua arquitetura. O SILF funciona segundo a arquitetura cliente-servidor. A arquitetura cliente-servidor é um modelo de organização de sistemas de computação em que as tarefas são divididas entre dois tipos de entidades: clientes e servidores. Nesse modelo, o cliente geralmente é o componente que faz solicitações de serviços ou recursos, enquanto o servidor é o responsável por processar essas solicitações e fornecer as respostas ou os dados necessários (Stallings, 2017).

A Figura 18 ilustra o esquema de comunicação do SILF. Os dois microcontroladores ESP32, telefones celulares e computadores atuam como clientes, enquanto um programa de computador, o API REST Python Flask, atua como servidor. O site também atua como cliente.

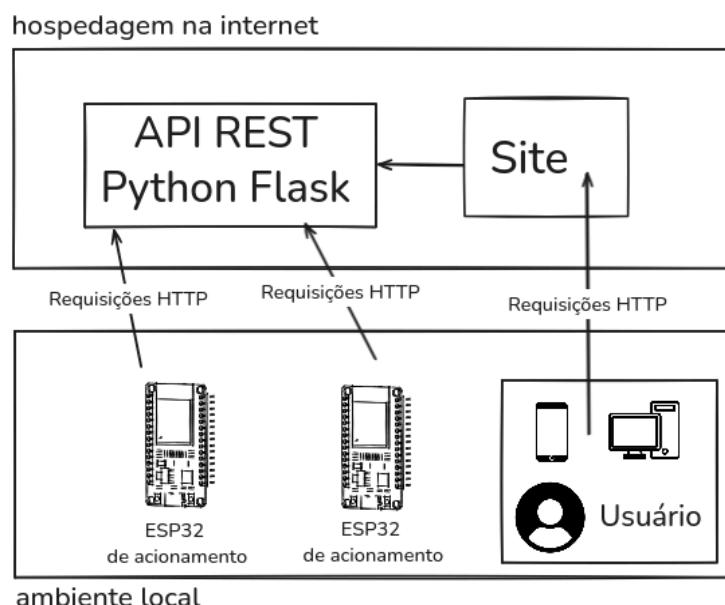


Figura 18. Esquema de comunicação dos dispositivos do sistema. Fonte: autores.

Segundo Valle Román et al. (2023), a comunicação entre microcontroladores pode ocorrer por meio do protocolo HTTP. O Hypertext Transfer Protocol (HTTP) é um protocolo de nível de aplicação fundamental, projetado para sistemas de informação de hipertexto distribuídos e colaborativos. Trata-se de um protocolo sem estado (stateless), o que significa que cada requisição feita pelo cliente ao servidor é processada de forma independente, sem armazenamento de informações de sessão. O HTTP facilita a transferência de documentos de hipertexto e constitui um componente essencial do funcionamento da World Wide Web. Esse protocolo suporta diversos métodos de comunicação, como GET e POST, além de definir um formato estruturado para as mensagens trocadas entre cliente e servidor.

Na arquitetura cliente-servidor, o cliente é responsável pela interface e pelas solicitações do usuário, enquanto o servidor executa o processamento de dados e o gerenciamento de recursos (Orfali, 1999). No presente projeto, foi implementada uma API REST no servidor em nuvem, desenvolvida em linguagem Python, utilizando o framework Flask. Esse programa de computador foi registrado no INPI, sob o número BR 51 2025 002370-7 (Bailão et al., 2024c). As APIs REST (Representational State Transfer Application Programming Interfaces) são amplamente empregadas no desenvolvimento do web moderno por proporcionarem uma comunicação eficiente entre aplicações cliente e servidor (Fielding, 2000).

Essa abordagem se baseia em chamadas HTTP padronizadas, o que simplifica a integração e o intercâmbio de informações. Nesse contexto, a API desenvolvida em Python constitui o back-end do SILF, o qual, segundo Orfali, Harkey e Edwards (1999), comprehende todas as operações do lado do servidor, incluindo armazenamento, processamento e recuperação de dados, elementos fundamentais para o funcionamento do aplicativo.

Além do servidor back-end, foi implementado o serviço front-end utilizando o framework Angular. No escopo deste trabalho, o front-end corresponde à interface pela qual o usuário interage com as funcionalidades do SILF. O Angular, desenvolvido pela Google, é um framework amplamente utilizado na construção de aplicações web modernas e oferece suporte nativo ao protocolo HTTP, atuando como cliente nesse modelo de comunicação (Monk, 2017).

```

#include <DHT.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include <math.h>
#include <esp_system.h> // Necessário para ESP.restart()

#define DHT11PIN 23
const int pinosSolo[5] = {32, 33, 34, 35, 36}; // pinos ADC1 (compatíveis com Wi-Fi)

const char *ssid = "Marvi_2.4GHz";
const char *password = "77m0408v";

DHT dht(DHT11PIN, DHT11);
float map_value(float x, float in_min, float in_max, float out_min, float out_max) {
    return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}
double getUmidadePercentual(double leitura){
    float umidadePercentual = map_value(leitura, 4095, 0, 0, 100); // mapeia leitura para %
    umidadePercentual = constrain(umidadePercentual, 0, 100);
    return umidadePercentual;
}

```

Figura 19. Foto de um trecho do código fonte implementado na placa ESP32. Fonte: autores.

Adicionalmente, foram implementadas soluções de software nas placas ESP32, empregando a linguagem C e o framework Arduino (Freeman; Freeman, 2022). A Figura 19 é uma foto do código embarcado. O código fonte responsável pelo acionamento da bomba hidráulica embarcado no microcontrolador foi registrado no INPI, sob o número BR 51 2025 001771-5 (Bailão et al., 2024a), já o software embarcado de leitura de dados dos sensores foi registrado também no INPI no processo BR 51 2025 001771-5 (Bailão et al., 2024b). Nesse projeto, o código desenvolvido nos microcontroladores desempenhou o papel de cliente HTTP, realizando requisições para controlar o experimento e transmitir os dados obtidos pelos sensores.

3.5. Aspectos Arquiteturais do sistema

Para explicar aspectos arquiteturais do sistema, este trabalho se utilizará de diagramas UML. Segundo Booch, Rumbaugh e Jacobson (2006), os diagramas UML permitem estruturar e visualizar diferentes aspectos de um sistema, incluindo sua arquitetura de componentes. Em um nível mais alto, é utilizado o diagrama UML de componentes para mostrar como os “blocos” de implementação (componentes) se organizam e se conectam para entregar as funcionalidades previstas.

A Figura 20 ilustra o diagrama de componentes geral do sistema. O SILF é dividido em três módulos principais, denominados clientes, que estão presentes no

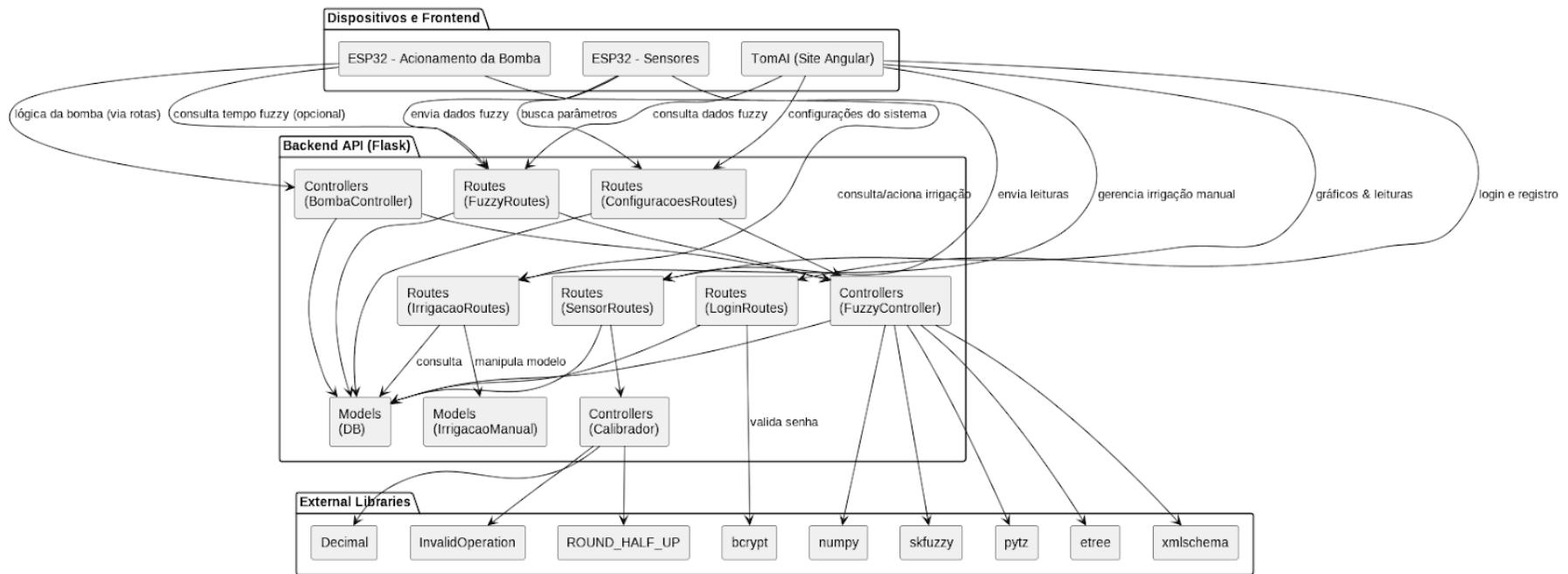


Figura 20. Diagrama de componentes. Fonte: autores.

módulo “Dispositivos e Frontend”, que é responsável por consumir os dados (acionando a bomba hidráulica e exibir a interface gráfica no site), bem como por enviar dados dos sensores (Dispositivos ESP32) e permitir ao usuário fazer suas alterações nos parâmetros de otimização de irrigação (via Site). Já o Módulo Backend API (Flask) é responsável pela tomada de decisão de irrigação ante os parâmetros configurados. É esse módulo que também persiste os dados em um banco de dados na nuvem, e também implementa o módulo de controle de acesso aos dados do sistema. A existência do Backend API (Flask) possibilita que a comunicação seja padronizada para diferentes clientes. Há também bibliotecas externas, pacotes que são importados diretamente no código do servidor, para realização da criptografia, validações e, o mais importante, para o cálculo de fuzzificação e defuzzificação do método pelo centroide.

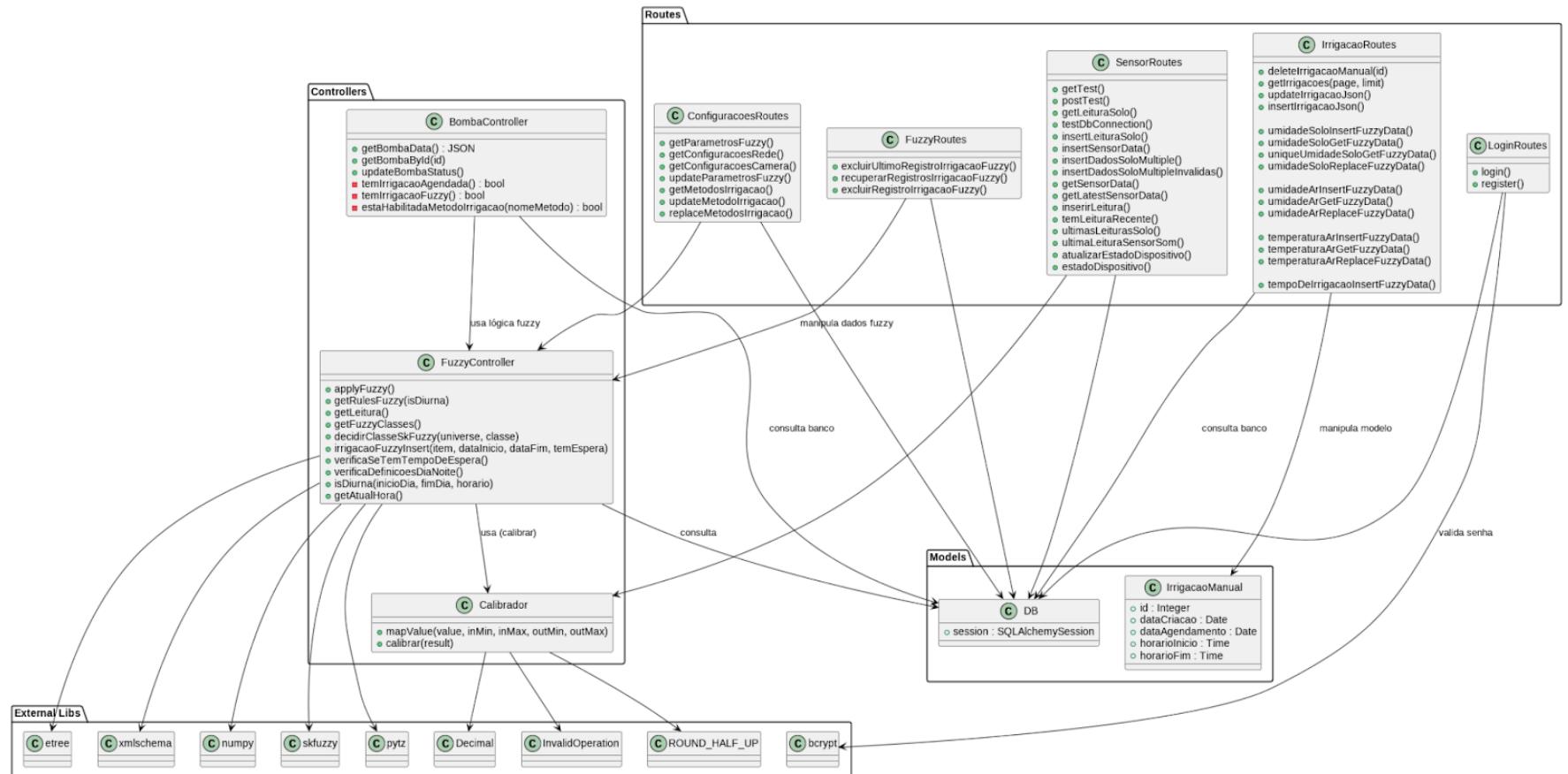


Figura 21. Diagrama de classes para o Módulo Arquitetural Backend API (Flask). Fonte: autores.

3.5.1. Módulo Arquitetural Backend API (Flask)

Segundo Sommerville (2019), nas arquiteturas cliente-servidor, a maior parte do processamento, da lógica da aplicação e do gerenciamento de dados ocorre no servidor, que responde às solicitações feitas pelos clientes. Esse conjunto de funcionalidades corresponde ao que, atualmente, se denomina backend. Este módulo corresponde à parte do sistema que faz o processamento interno, executa regras de negócio, gerencia dados e responde às requisições feitas pelo cliente. É a camada responsável pela lógica do software e pela comunicação com o banco de dados, operando em servidores que não são diretamente acessados pelo usuário final.

A Figura 21 ilustra em detalhes as entidades de código presentes no backend do SILF. O backend é dividido em Models – classes que representam entidades do banco de dados; em Controllers – classes que tratam regras de negócio; em External Libs – bibliotecas externas usadas pelos controllers; e em Routes – endpoints da API que conversam com controllers e modelos. No backend, as rotas *SensorRoutes*, *IrrigacaoRoutes* e *FuzzyRoutes* recebem requisições do ESP32 e do site TomAI e repassam para os Controllers elaborarem os cálculos de tempo de irrigação segundo o método configurado. As rotas de *ConfiguracoesRoutes* permitem ao site habilitar o método de irrigação que será utilizado: irrigação 1 vez ao dia, 5 vezes ao dia ou irrigação fuzzy. Já as rotas *LoginRoutes* irão gerenciar os usuários do site e configurar o acesso remoto ao sistema de irrigação.

Para realização da persistência das irrigações que não se utilizam de fatores ambientais, o sistema se utiliza de uma entidade chamada de *IrrigacaoManual*, que representa uma irrigação agendada manualmente pelo usuário, ou via automações no sistema. Para efetivamente fazer essa persistência, o sistema se utiliza da classe *DB*, que encapsula a sessão do *SQLAlchemy* (middleware de banco de dados MySQL).

O pacote *Controllers* tem muitas das regras de negócio do sistema, com destaque para a classe *FuzzyController*, que implementa chamadas a *skfuzzy*, que efetivamente faz o cálculo da irrigação pelo método do centroide. Nessa classe, o método *applyFuzzy()* executa a inferência fuzzy com os dados atuais, e o método *getRulesFuzzy()* busca as regras fuzzy, que podem variar entre dia/noite no sistema. Já o método *getFuzzyClasses()* faz a busca dos conjuntos fuzzy definidos pelo usuário,

como, por exemplo, definições de umidade de solo “seco”, “normal” e “úmido”, que foram devidamente definidos através da interface gráfica do site.

O método *decidirClasseSkFuzzy()* é um método que armazena em memória a decisão de irrigação usando skfuzzy, porém não aplica essa decisão, pois a efetivação da ação ocorre pelo método *applyFuzzy()*. Os métodos auxiliares *verificaSeTemTempoDeEspera()* fazem verificações de colisão entre tempos de irrigação, tratando os diferentes casos de irrigação, como, por exemplo, irrigação manual e fuzzy habilitadas colidem horários, dessa forma, o método *verificaSeTemTempoDeEspera()* escalona as irrigações.

O método *verificaDefinicoesDiaNoite()* é um método que checa a hora atual no servidor e verifica se é dia ou noite. Esse método é importante no dimensionamento automático do sistema, pois se for período diurno, o SILF irá se utilizar da base de regras diurnas, se for período noturno, irá se utilizar de regras noturnas. A classe *Calibrador* é uma classe responsável por fazer a calibração dos sensores analógicos de umidade de solo, umidade de ar e temperatura do ar. Essa classe implementa uma calibração linear para esses dados, convertendo valores absolutos para porcentagem, no caso da umidade do ar e de solo, e para graus Celsius, no caso da temperatura do ar.

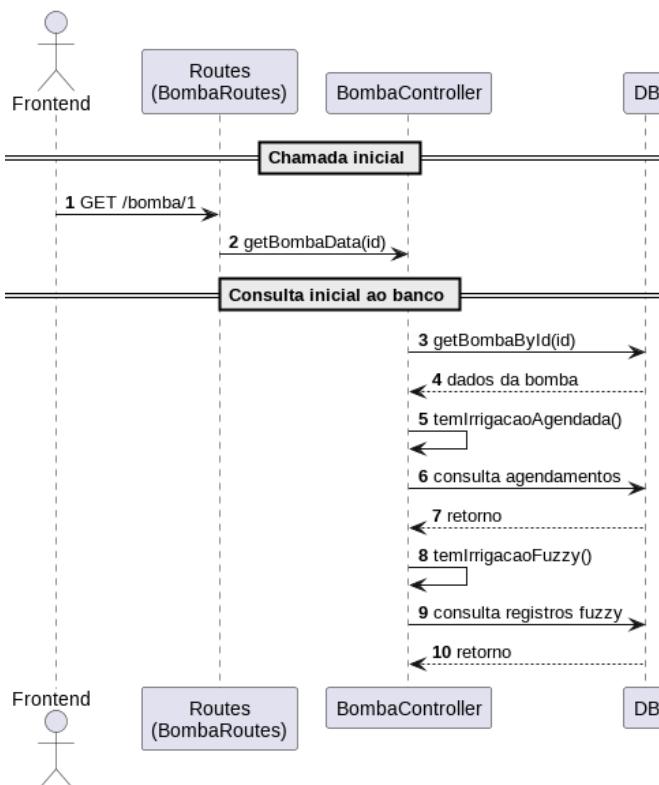


Figura 22. Diagrama de Sequência para consulta do estado da bomba. Fonte: autores.

A classe *BombaController* é responsável pela confiabilidade, integridade e disponibilidade da consulta dos dados do motor hidráulico do sistema. Nessa classe, o método *getBombaData()* é um método para consultar o estado da bomba, e *updateBombaStatus()* é um método para atualizar o estado da bomba no sistema. Há também os métodos *temIrrigacaoAgendada()* e *temIrrigacaoFuzzy()*, que são métodos que tratam colisões entre a possibilidade de o usuário ter habilitado diferentes métodos de irrigação, bem como ter agendado anteriormente irrigações que possam chocar com irrigações automáticas. O diagrama da Figura 22 ilustra o encadeamento dessas consultas com o banco de dados. O método *estaHabilitadaMetodoIrrigacao()* é um método que verifica se existe algum método de irrigação habilitado, se houver algum habilitado ele é retornado. A Figura 23 ilustra o encadeamento de ações de atualização do estado da bomba.

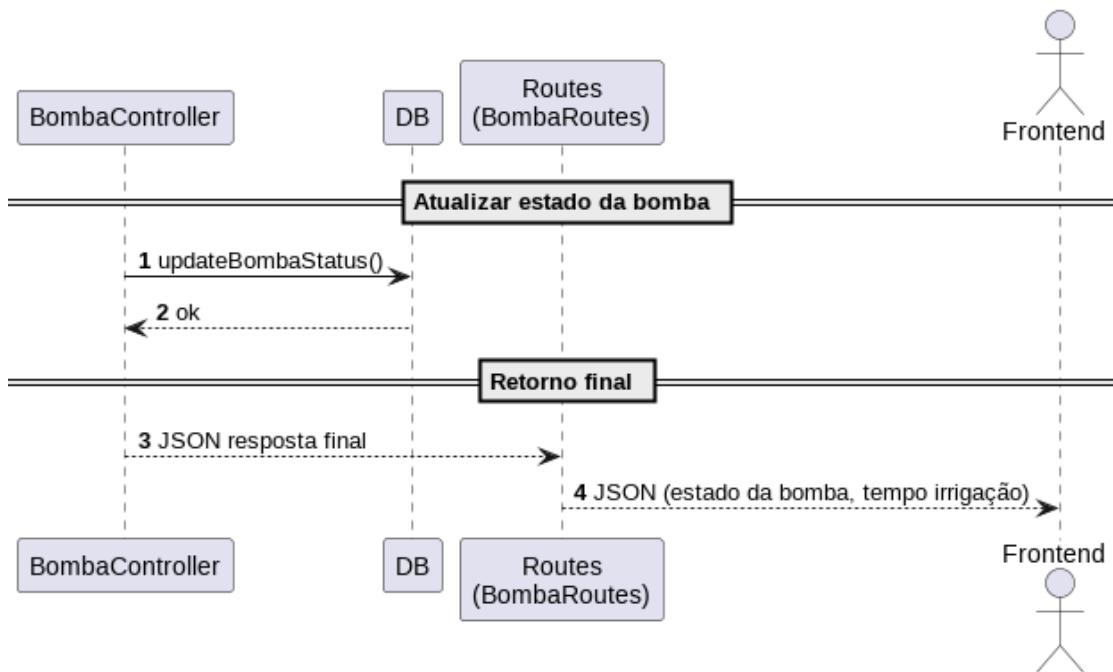


Figura 23. Diagrama de Sequência para atualização da bomba. Fonte: autores.

3.5.2. Módulo Arquitetural Sistemas Embarcados

O sistema apresentado no diagrama da Figura 22 mostra a arquitetura de comunicação entre o ESP32 e o servidor backend responsável por informar se a bomba

deve ser ligada ou desligada. No ESP32, o fluxo principal é conduzido pelo `setup`, que inicializa Wi-Fi, configura pinos e prepara o dispositivo, e pelo `loop`, que executa continuamente a lógica de monitoramento. Dentro desse loop, o microcontrolador verifica a conexão com a rede, acessa os serviços responsáveis por consultar o backend e aplica o estado recebido à bomba, garantindo que o funcionamento seja dinâmico e atualizado em tempo real.

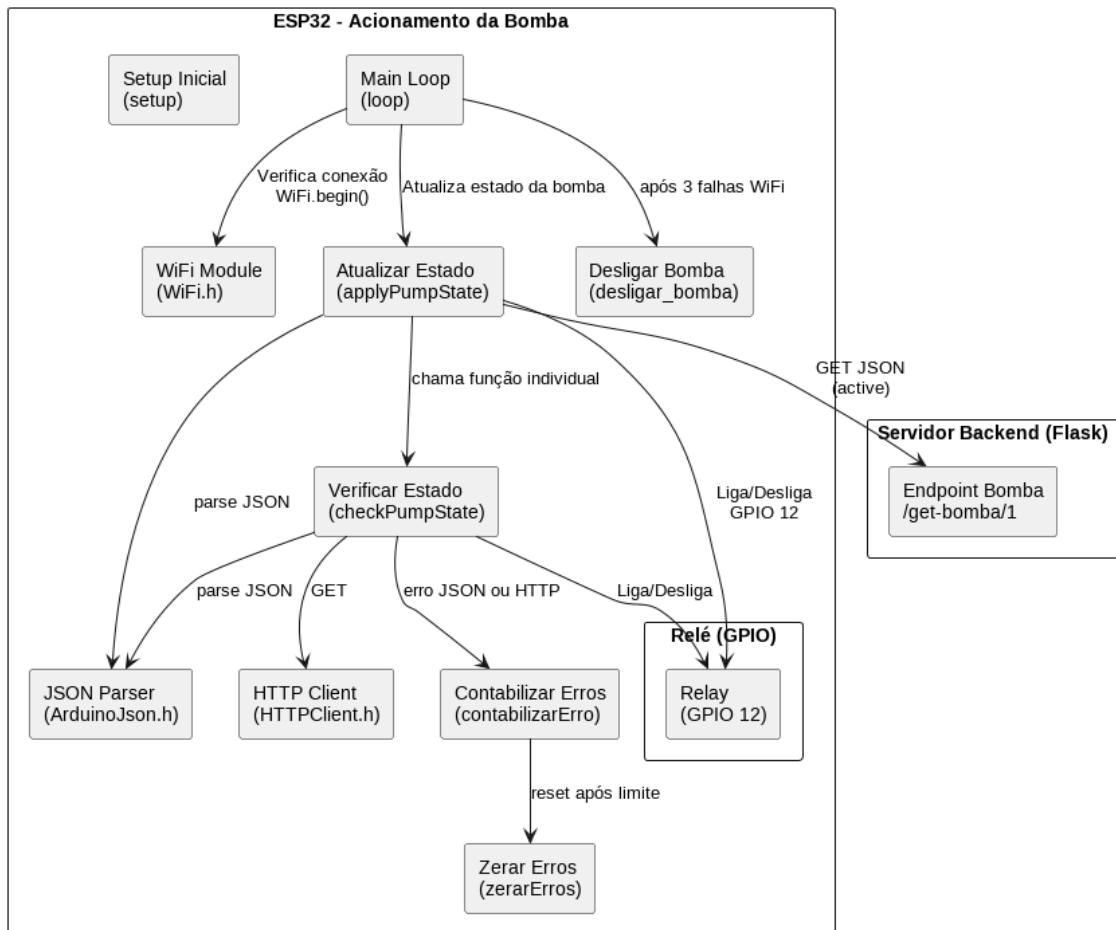


Figura 24. Diagrama de componentes para o Módulo Arquitetural Sistemas Embarcados. Fonte: autores.

Para que isso ocorra, o ESP32 utiliza três componentes essenciais: a biblioteca de WiFi para manter a conectividade, o cliente HTTP para fazer requisições ao servidor Flask e a biblioteca ArduinoJson para interpretar o JSON recebida da API. Com base nessas informações, duas funções de controle entram em atuação: `checkPumpState`, responsável por consultar diretamente o endpoint da bomba e interpretar o retorno, e

applyPumpState, que coordena o processo de atualização, acionando o relé conforme o estado informado pelo backend.

O sistema também incorpora um mecanismo de segurança e confiabilidade. Componentes como *contabilizarErro* e *zerarErros* monitoram falhas de comunicação, seja por erros HTTP, problemas no JSON ou queda do Wi-Fi, e registram ocorrências para impedir que a bomba opere em condições inseguras. Quando erros consecutivos ultrapassam o limite permitido ou a conexão Wi-Fi falha repetidamente, o componente *desligar_bomba* é acionado para garantir o desligamento automático, evitando acionamentos indevidos.

Por fim, o backend em Flask disponibiliza um endpoint, que fornece ao ESP32 o estado da bomba no formato JSON, informando se ela deve permanecer ligada ou desligada. Esse endpoint desempenha papel de “centro de decisão” do sistema, permitindo que o controle da bomba seja remoto e atualizado conforme a lógica definida no servidor. A integração entre os componentes do ESP32, as bibliotecas de comunicação e o backend resulta em um sistema robusto, seguro e modular para o controle automatizado da irrigação.

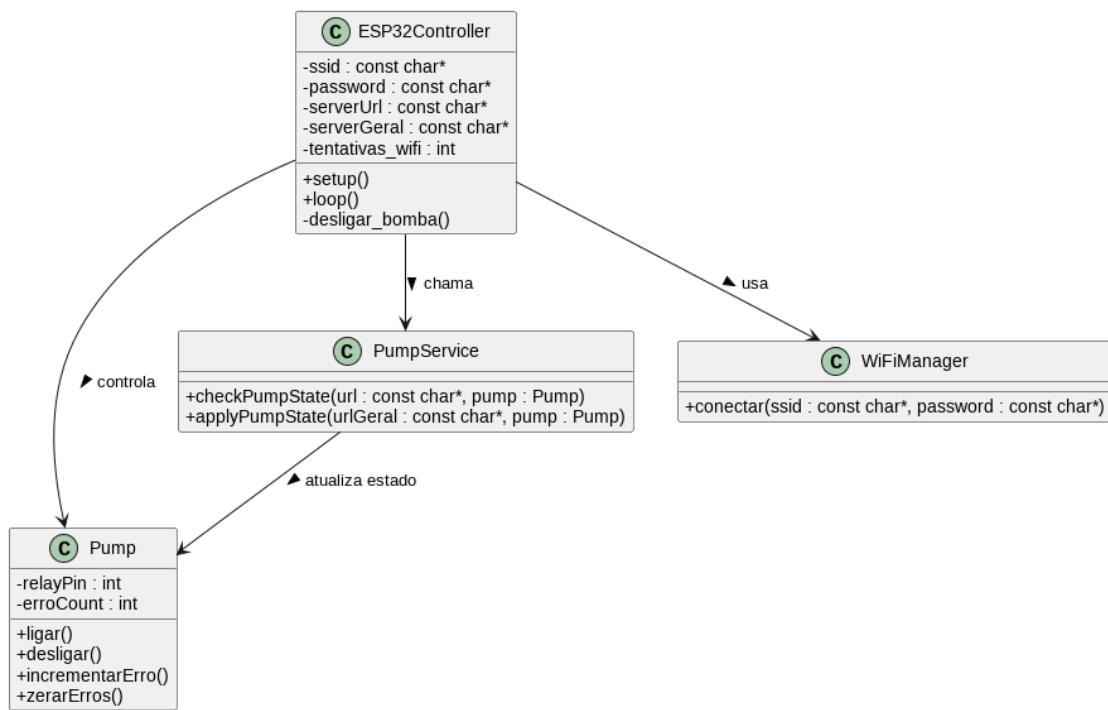


Figura 25. Diagrama de classes para o Módulo Arquitetural Sistemas Embarcados.

Fonte: autores.

O sistema mostrado no diagrama da Figura 25 descreve como o ESP32 controla uma bomba de irrigação através de classes, conectando-se ao Wi-Fi e consultando um servidor HTTP para saber se deve ligar ou desligar a bomba. A arquitetura é dividida em quatro classes principais. A classe ESP32Controller é a classe principal do controlador eletrônico, essa classe inicializa Wi-Fi, configura pinos e objetos, faz a leitura do servidor, a atualização da bomba e refaz a conexão WiFi, se necessário.

3.6 Interface Gráfica do Site

No site do SILF há configurações para dois tipos de irrigações principais: Irrigação Fuzzy Inteligente, que compreende a irrigação considerando fatores ambientais, e a Irrigação Manual, que permite que a irrigação seja feita 1 vez ao dia e 5 vezes ao dia, não considerando essa última fatores ambientais, apenas horário do início e do fim.

3.6.1. Tela de Inserção de Classes Fuzzy da Irrigação Fuzzy Inteligente

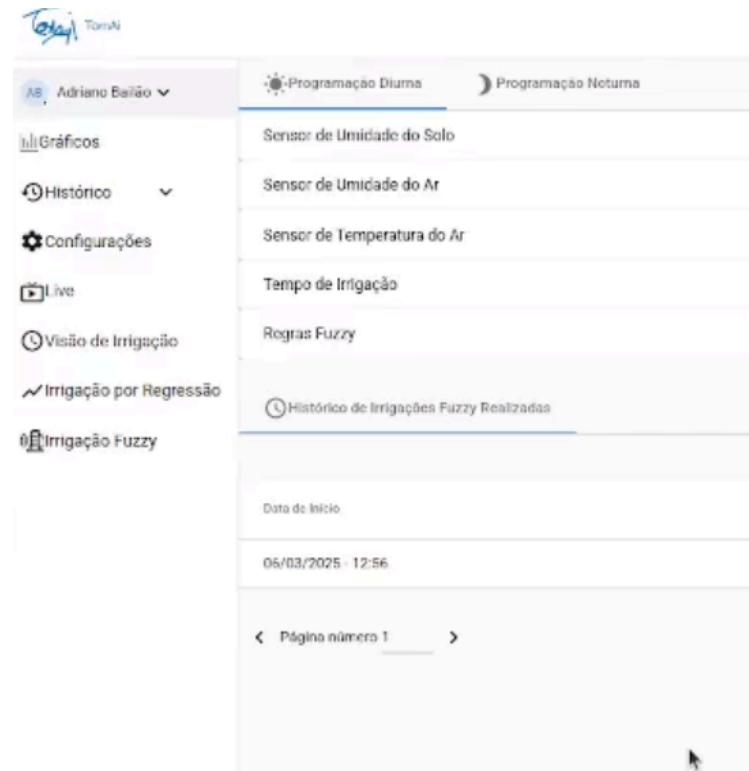


Figura 26. Tela de Inserção de Classes Fuzzy.

Na tela ilustrada na Figura 26, o usuário pode definir os conjuntos fuzzy para cada uma das variáveis ambientais, bem como para a variável de saída, o tempo de irrigação. Há duas abas principais: Programação diurna e Noturna. Dessa forma, o usuário consegue definir conjuntos e regras fuzzy diferentes para o período diurno e noturno. Além disso, nessa tela há uma tabela contendo o histórico de irrigações fuzzy realizadas. As Figuras 27 e 28 representam a interface gráfica do site para definição de variáveis. O usuário pode utilizar funções Trapezoidal, Triangular e Sigmoide para definir cada conjunto de cada variável. Tudo é feito através de cliques do mouse ou via teclado.

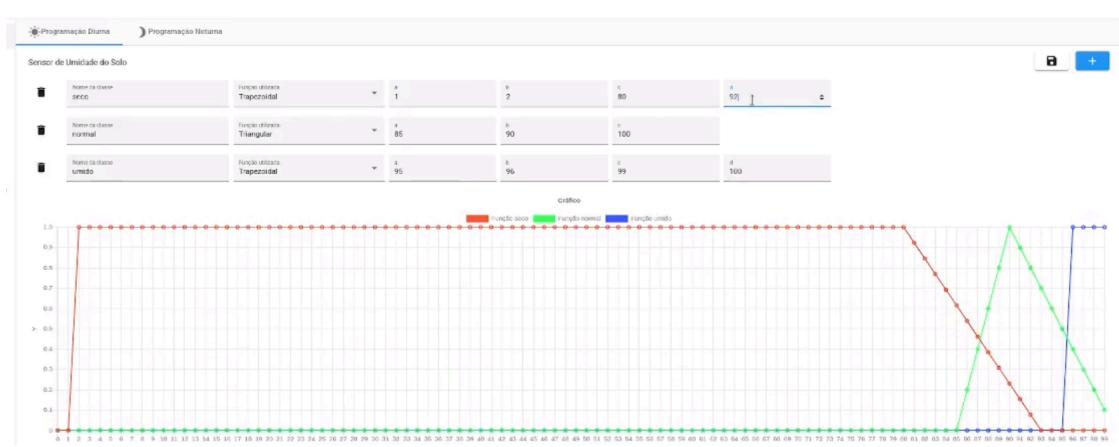


Figura 27. Inserção e alteração de conjuntos fuzzy para umidade do solo.

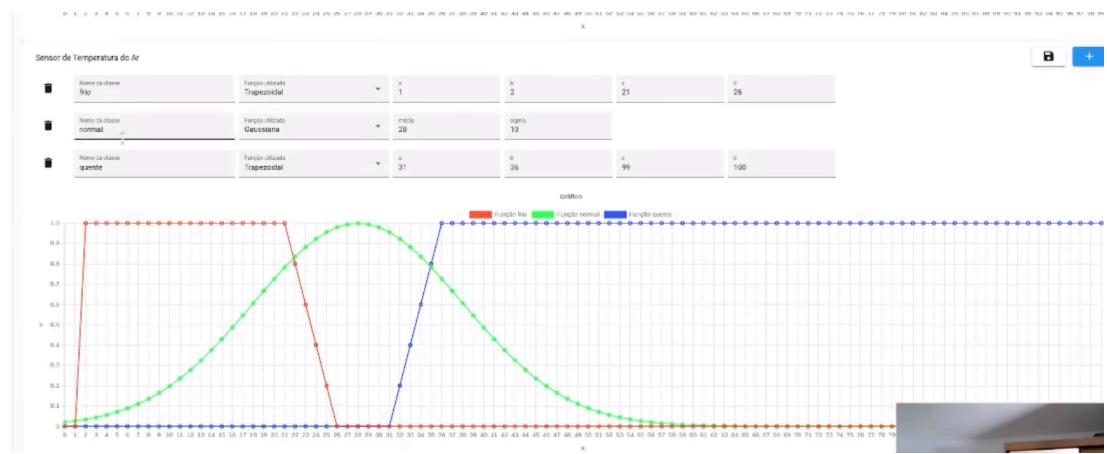


Figura 28 Inserção e alteração de conjuntos fuzzy para temperatura do ar.

O Local onde o usuário constrói a lógica que relaciona as classes fuzzy entre si é o ambiente de regras fuzzy, presente na Figura 26. Características da interface: Tabela com várias linhas, em que cada linha representa uma regra fuzzy. Comboboxes

permitem escolher, para cada variável, a classe fuzzy correspondente (ex.: solo = seco, umidade ar = baixa, temperatura = alta). Há também um campo para definir a classe de saída (tempo de irrigação: pouco, médio, muito). Botões de adicionar, editar ou excluir regras. A interface é semelhante a um editor de regras especialista, facilitando a construção visual da lógica.

Regras Fuzzy
Temperature do ar: não quente e Umidade de ar: seco → Então: muito
Temperature do ar: não frio e Umidade de ar: umido → Então: pouco
Temperature do ar: não normal e Umidade de ar: normal → Então: mediano
Temperature do ar: não quente e Umidade de ar: normal → Então: muito
Temperature do ar: frio e Umidade de ar: seco → Então: pouco
Temperature do ar: não normal e Umidade de ar: umido → Então: mediano
Temperature do ar: não quente e Umidade de ar: umido → Então: muito
Temperature do ar: não frio e Umidade de ar: normal → Então: pouco
Temperature do ar: não normal e Umidade de ar: seco → Então: mediano
Temperature do ar: não normal e Umidade de ar: normal → Então: muito
Temperature do ar: não normal e Umidade de ar: normal → Então: pouco
Temperature do ar: não normal e Umidade de ar: normal → Então: mediano

Historico de Irrigações Fuzzy Realizadas

Figura 29 Inserção e alteração de regras.

3.6.2. Tela de Irrigação Atual da Irrigação Fuzzy Inteligente

A tela de irrigação atual (Figura 27) é dedicada ao monitoramento visual das irrigações no exato momento em que ocorrem. Funcionalidades apresentadas: Indicador visual exibindo se a bomba está ligada ou desligada no minuto atual; Gráfico ou painel mostrando Tempo total disponível (60 minutos); Tempo inferido pelo sistema (ex.: 42 minutos de irrigação); e Tempo restante sem irrigação. Quando não há irrigação acontecendo, um painel informativo comunica a situação. Quando uma irrigação está ativa, a interface demonstra dinamicamente o andamento do ciclo, mostrando a proporção de minutos da bomba ligada e desligada.

A Figura 28 ilustra o estado desta tela quando a bomba está desligada, nesse caso o sistema inferiu apenas 1 minuto de irrigação. Enquanto essa irrigação acontecia, o painel estava com estado semelhante à Figura 27, contudo, nos próximos 59 minutos, permaneceu com o estado semelhante à interface da Figura 28.

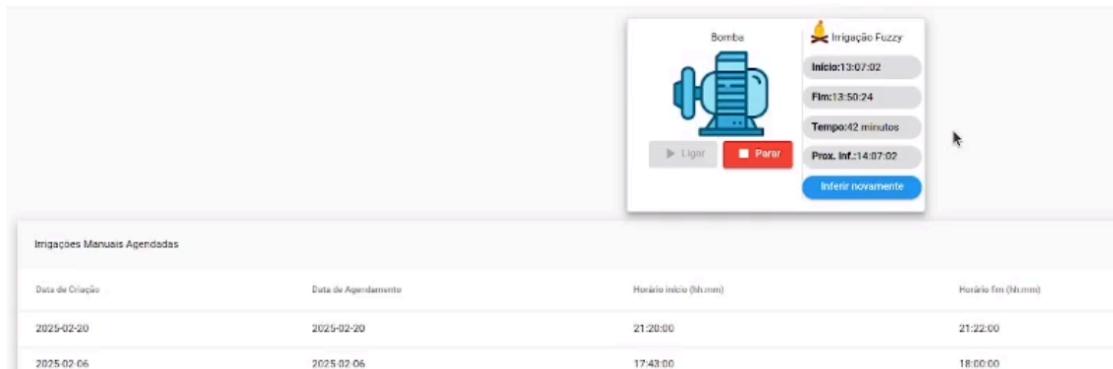


Figura 30 Tela de irrigação atual com irrigação acontecendo.

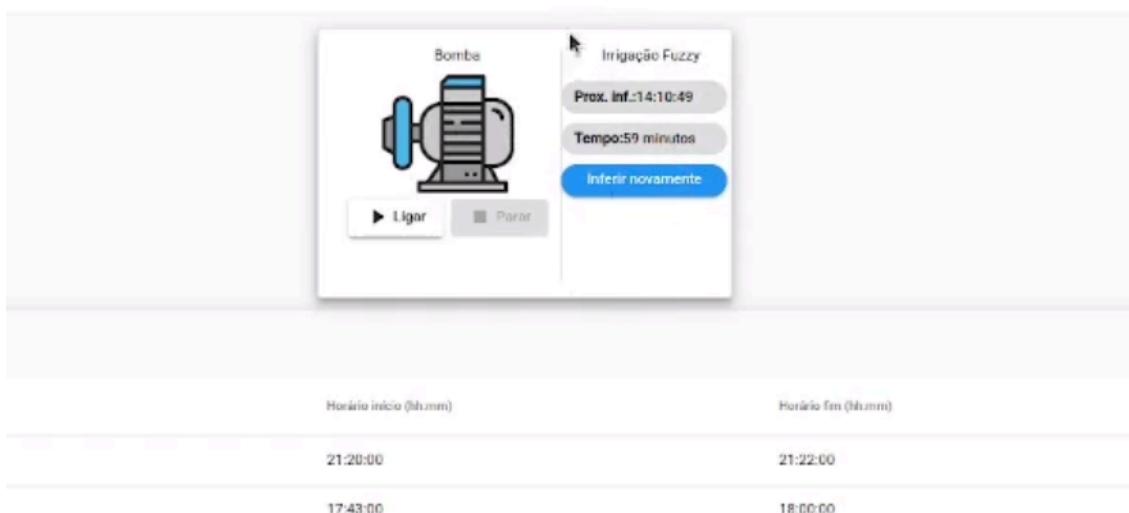


Figura 31 Tela de irrigação atual com irrigação inativa.

3.6.3. Tela de Visão de Irrigação para irrigação agendada

A tela da Figura 29 ilustra a tela de visão de irrigação para a irrigação manual, método que permite configuração de irrigação 1 vez ao dia e 5 vezes ao dia, pois permite cadastro de irrigação com data, horário de início e horário de fim. Na tabela de visão de irrigação (Figura 29) é possível incluir, excluir e editar irrigação manual.

A Figura 30 apresenta o modal de inserção de irrigação agendada com os respectivos campos citados. Esse modal é acessado a partir da tabela anterior, no botão de adicionar.

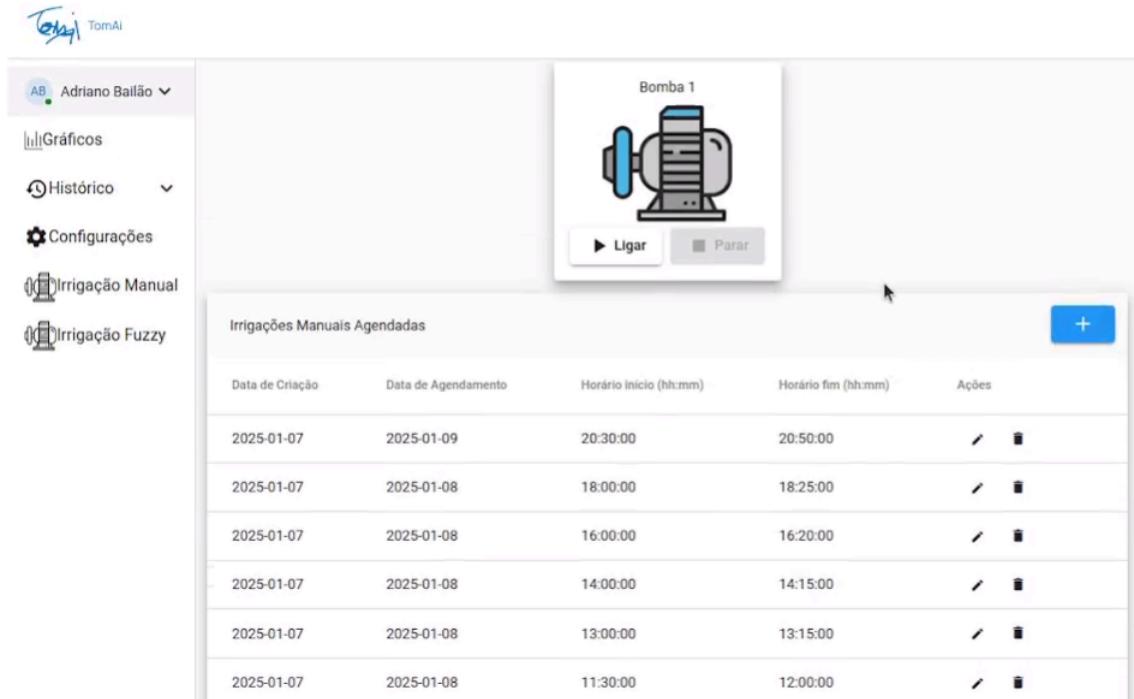


Figura 32 Tela de Visão de Irrigação para irrigação agendada.

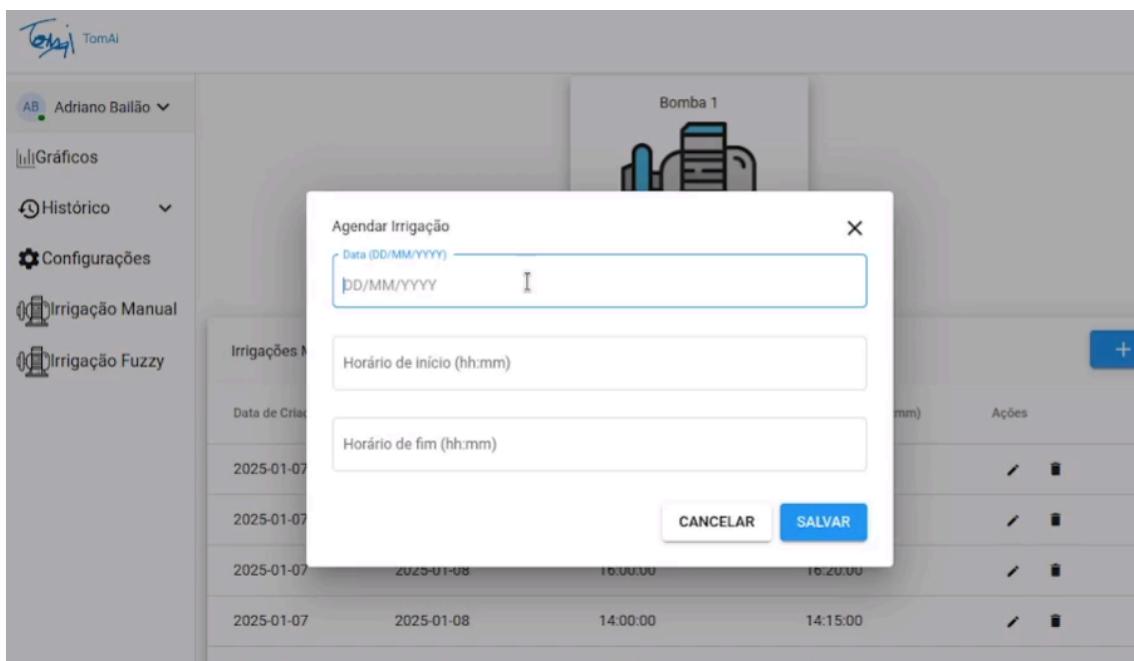


Figura 33 Modal de inserção de irrigação agendada.

4. RESULTADOS

A Análise de Variância, Tabela 8, foi utilizada para verificar se existe diferença estatisticamente significativa entre as médias de diferentes tratamentos (substratos de 3 L, 6L, 9L, 12L e 15 L) e Manejos (irrigação 1 vez ao dia sem interrupção, irrigação 5 vezes ao dia em tempo espaçado e, por fim, irrigação inteligente via SILF).

Tabela 8. Fontes de variação.

Fonte de variação	Soma dos Quadrados	GL	F	p-valor
Bloco	190825,4	1	7,31	0,0097
Tratamento (Volume)	844928,5	4	8,09	0,0001
Manejo	935193,9	2	17,92	0,0000
Interação (Tratamento×Manejo)	240485,6	8	1,15	0,3494
Residual	1148416	44	-	-

No fator Bloco, foi obtido *p-valor* de 0.0097, que é menor que 0.05, até menor que 0.01, tendo sido o efeito do Bloco estatisticamente significativo. Não há uma diferença significativa na drenagem entre os diferentes blocos experimentais. Nesse caso, significa que a estratégia de bloqueio foi eficaz em controlar uma parte da variabilidade experimental. Considerando o Tratamento Volume de Substrato, o *p-valor* de 0.0001 é extremamente baixo, muito menor que 0.05. Isso significa que esse volume tem um efeito altamente significativo na drenagem. Ou seja, a drenagem média é significativamente diferente entre os diferentes volumes de substrato. Vamos agora verificar a variação entre os métodos de irrigação para averiguar se os métodos são estatisticamente diferentes.

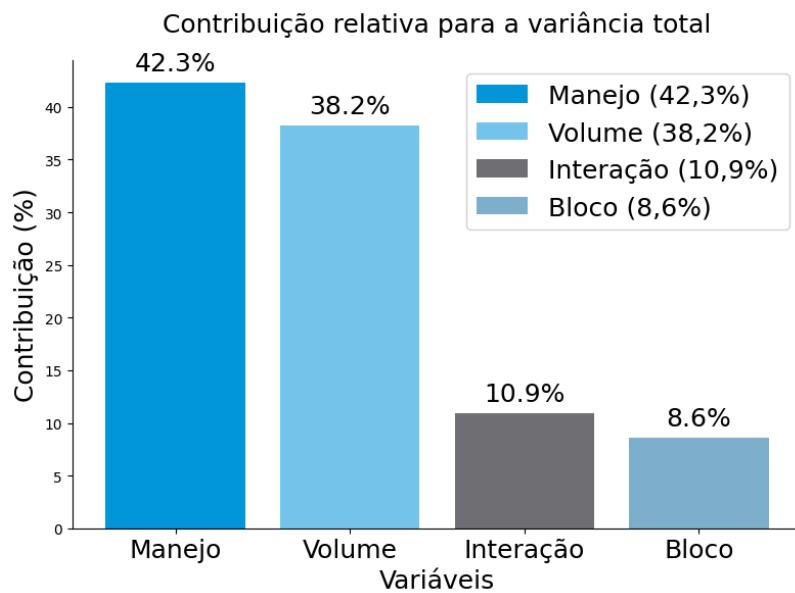


Figura 34. Gráfico de contribuições. Fonte: autores.

No fator Método de Irrigação, o p-valor é virtualmente zero. Isso indica que o Método de irrigação tem um efeito altamente significativo na drenagem. Tratando agora da relação Método de irrigação versus Volume de substrato escolhido, o p-valor de 0.3494 é maior que 0.05. Sendo assim, a interação entre Volume de Substrato e o método não é estatisticamente significativa. Em outras palavras, o efeito de um método de irrigação na drenagem é, de modo geral, consistente em todos os volumes de substrato testados e vice-versa. Não há um efeito combinado ou dependente do complexo entre eles. Por exemplo, se um método é melhor para um volume, ele tende a ser melhor (ou pelo menos manter seu padrão) para outros volumes. Considerando agora o impacto de cada fator nos resultados, o gráfico da Figura 31 mostra a contribuição relativa de diferentes fatores para a variância total dos dados.

4.1 Comparação entre volumes de substrato

Foi feito então o Teste de Tukey, para identificar quais pares de médias entre os grupos são significativamente diferentes, controlando a taxa de erro para comparações múltiplas (FWER = Family-Wise Error Rate). O nível de significância utilizado foi de 0.05. Na Tabela 9, os grupos são os volumes de substrato comparados. Nessa mesma tabela, Diferença Média denota a diferença entre as médias dos dois grupos. Há uma diferença estatisticamente significativa na drenagem média entre o volume de 15 L e de

3 L e 9 L ($p<0,05$), exibindo menor drenagem média. Nos demais pares não houve diferença estatística, sinalizando que, nas condições do ensaio, volumes maiores tendem a reduzir o excedente de água.

Tabela 9. Comparações múltiplas de médias da drenagem. Fonte: autores.

Grupo 1	Grupo 2	Diferença média	p-valor	IC 95% (Inferior ; Superior)	Rejeitar H0
1 (3L)	2 (6L)	-139,3988	0,5058	(-385,6083 ; 106,8107)	Não
1 (3L)	3 (9L)	-46,9167	0,9830	(-293,1262 ; 199,2928)	Não
1 (3L)	4 (12L)	-112,9464	0,6961	(-359,1559 ; 133,2631)	Não
1 (3L)	5 (15L)	-345,0238	0,0020	(-591,2333 ; -98,8143)	Sim
2 (6L)	3 (9L)	92,4821	0,8262	(-153,7273 ; 338,6916)	Não
2 (6L)	4 (12L)	26,4524	0,9981	(-219,7571 ; 272,6619)	Não
2 (6L)	5 (15L)	-205,6250	0,1434	(-451,8345 ; 40,5845)	Não
3 (9L)	4 (12L)	-66,0298	0,9418	(-312,2392 ; 180,1797)	Não
3 (9L)	5 (15L)	-298,1071	0,0102	(-544,3166 ; -51,8977)	Sim
4 (12L)	5 (15 L)	-232,0774	0,0737	(-478,2869 ; 14,1321)	Não

Há também uma diferença estatisticamente significativa entre o volume de substrato de 9L e o de 15L ($p\text{-adj} = 0.0102$). A drenagem em 15L é significativamente menor do que em 9L. Todas as outras comparações entre volumes de substrato (ex: 3L vs. 6L, 6L vs. 9L, 12L vs. 15L) não mostram diferenças estatisticamente significativas na drenagem média. Embora existam diferenças numéricas, elas não são grandes o suficiente para serem consideradas reais e não aleatórias, com base no critério de significância de 0.05.

4.2. Comparação entre manejos de irrigação

A Tabela 10 apresenta a síntese das comparações entre os três esquemas de irrigação avaliados: aplicação contínua de 30 minutos diários, cinco aplicações de 6 minutos cada e ajuste dinâmico por lógica fuzzy em intervalos horários. Verifica-se que o sistema baseado em lógica fuzzy proporcionou redução significativa da drenagem em relação aos manejos convencionais de 30 minutos diários e cinco irrigações diárias

(p<0,05). Entre os dois regimes de tempo fixo não foram constatadas diferenças estatisticamente relevantes, o que evidencia a vantagem prática e adaptativa do método fuzzy.

Tabela 10. Comparações múltiplas de médias da drenagem (Tukey HSD) – Manejos de irrigação.

Grupo 1	Grupo 2	Diferença média	p-valor	IC 95%	Rejeitar H0
Fuzzy	Irrig_1	300,3929	0,0001	143,4438 — 457,3419	Sim
Fuzzy	Irrig_5	199,8214	0,0092	42,8724 — 356,7705	Sim
Irrig_1	Irrig_5	-100,5714	0,2793	-257,5205 — 56,3776	Não

5. CONCLUSÃO

O sistema de irrigação inteligente baseado em lógica fuzzy (SILF) demonstrou redução significativa na drenagem (perda de água) em comparação com métodos tradicionais (irrigação fixa diária ou intervalos regulares). Há, em média, 200 ml de economia de água para cada planta ao adotar SILF em detrimento de um manejo de irrigação em intervalos regulares. E também há uma economia em média de 300 mL por planta ao adotar o SILF em comparação com o manejo de irrigação de 2 litros por planta uma única vez ao dia.

Ao comparar com esse manejo tradicional de uma vez ao dia, nesse experimento com 80 vasos, o SILF alcança 24 litros por dia de economia de água, sendo que em um mês consumiria 720 litros de água. Apenas para comparar, pessoas em comunidades urbanas de baixa renda usam de 75L a 117L por dia para todas as atividades pessoais e domésticas (Sultana et al., 2022), então essa economia seria suficiente para abastecer as necessidades domésticas de uma pessoa por cerca de 6 a 9 dias.

A consistência do SILF foi superior em volumes de substrato entre 3L e 12L. Com relação a aspectos arquiteturais do sistema, os dispositivos (sensores DHT11 e de solo anticorrosivo, ESP32, comunicação HTTP e APIs REST em Flask) provaram sua viabilidade, permitindo controle remoto e atualizações centralizadas. O uso de microcontroladores como "clientes magros" simplificou a manutenção e a

escalabilidade, podendo utilizar essa arquitetura de sistema para testar outros métodos de irrigação, principalmente outros agentes inteligentes, como redes neurais.

A ANOVA confirmou que o método de irrigação (42.3% da variância) e o volume do substrato (38.2%) são os fatores mais críticos para a drenagem (p-valor próximo de 0). A interação entre ambos não foi significativa (p-valor=0.3494), indicando que o desempenho do SILF é consistente, independentemente do volume de substrato.

Como trabalhos futuros, pode-se pensar em combinar lógica fuzzy com redes neurais e algoritmos de aprendizado de máquina para ajuste automático das regras com base em dados históricos, para testar o desempenho de outros sistemas baseados em Inteligência Artificial. Outra perspectiva promissora consiste na integração de sensores de condutividade elétrica, permitindo o monitoramento da salinidade do solo e a inclusão desse fator no cálculo do tempo de irrigação. Além disso, sugere-se a adaptação das regras fuzzy para outras culturas, como morango ou pimentão, que apresentam diferentes exigências hídricas.

6. REFERÊNCIAS

ALFITH, Alfit et al. **Pengujian suhu dan kelembapan pada alat pengering gabah menggunakan sensor DHT11**. Ensiklopedia of Journal, v. 4, n. 2, p. 243-247, jan. 2022. DOI: 10.33559/eoj.v4i2.1074.

ALMEIDA, A. V. R. de; SILVA, A. O. da. **Modelagem fuzzy para suporte ao manejo da irrigação e uso de cobertura morta em cultivo de rabanete**. Irriga, Botucatu, v. 26, n. 4, p. 938-955, dez. 2021. DOI: <https://doi.org/10.15809/irriga.2021v26n4p938-955>.

ALMEIDA ROQUE, Iara; ALMEIDA DOS ANJOS SOARES, Lauriane; SOARES DE LIMA, Geovani; PEREIRA LOPES, Iracy Amelia; CHAVES DE ALMEIDA, Alan Keis; GOMES DE OLIVEIRA, Sabrina. **Growth and physicochemical characterization of cherry tomato under irrigation with saline water and nitrogen fertilization**. Comunicata Scientiae, [S. l.], v. 13, p. e3785, 2022. DOI: 10.14295/cs.v13.3785. Disponível em: <https://www.comunicatascientiae.com.br/comunicata/article/view/3785>. Acesso em: 13 out. 2025.

ALMEIDA, G.; NAZÁRIO, L.; SOUZA, L.; GONTIJO, P.; OLIVEIRA, R.; TEIXEIRA, M. C.; OLIVEIRA, M. de; TEIXEIRA, M.; CARMO, H. **Performance Analysis of YOLO and Detectron2 Models for Detecting Corn and Soybean Pests Employing Customized Dataset**. Agronomy, [S. l.], v. 14, n. 10, p. 1-22, 2024. DOI: 10.3390/agronomy14102194.

ARAÚJO, A. B. M.; COSTA, A. F. da; BARBOSA, D. P.; FRANÇA, H. F. do C.; NOVAK, S. S. **Avaliação comparativa de modelos YOLO para classificação e quantificação automatizadas de estágios de amadurecimento de tomates cereja**. Brazilian Journal of Development, [S. l.], v. 11, n. 8, p. e81856, 2025. DOI: 10.34117/bjdv11n8-050. Disponível em: <https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/81856>. Acesso em: 27 oct. 2025.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 7181: Solo – Análise granulométrica.** São Paulo: ABNT, 2025.

BANZATTO, D. A.; KRONKA, S. N. **Experimentação agrícola.** 4. ed. Jaboticabal: FUNEP, 2013.

BAÊTA DOS SANTOS, A. et al. **Manejo da irrigação na cultura do arroz irrigado e na eficiência do uso da água em várzeas tropicais.** Agri-Environmental Sciences, v. 7, n. 2, p. 13, 14 jun. 2021.

BAILÃO, Adriano Soares de Oliveira et al. **Acionador Bomba hidráulica TomAi para microcontroladores ESP32 TomAi.** 2024. Não paginado. Linguagem: C; outros. Campo de Aplicação: AG-06. Registro de programa de computador. BR 51 2025 001771-5, Instituto Federal de Educação, Ciência e Tecnologia Goiano, 26 out. 2024.

BAILÃO, Adriano Soares de Oliveira et al. **Leitor de Sensor de Umidade de Solo e DHT para microcontroladores ESP32 TomAi.** 2024. Não paginado. Linguagem: C. Campo de Aplicação: AG-06. Registro de programa de computador. BR 51 2025 001784-7, Instituto Federal de Educação, Ciência e Tecnologia Goiano, 27 out. 2024.

BAILÃO, Adriano Soares de Oliveira et al. **Servidor python back-end TomAi.** 2024. Não paginado. Linguagem: Flash; MySQL; Python. Campo de Aplicação: CC-09. Registro de programa de computador. BR 51 2025 002370-7, Instituto Federal de Educação, Ciência e Tecnologia Goiano, 23 out. 2024.

BAILÃO, Adriano Soares de Oliveira et al. **Sistema de irrigação fuzzy inteligente.** 2025. Não paginado. Linguagem: Java Script; outros. Campo de Aplicação: AG-02; AG-06; AG-09; MT-06. Registro de programa de computador. BR 51 2025 001780-4, Instituto Federal de Educação, Ciência e Tecnologia Goiano, 06 fev. 2025.

BARROS, G. M. et al. **Dia - Software de dimensionamento de irrigação por aspersão.** 2024. Programa de Computador. Registro INPI n. BR512024004851-0, 04 jun. 2024.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. UML: guia do usuário. 2. ed. Rio de Janeiro: Campus, 2006.

BRAGA, Newton C. **Eletrociadade e Instalações Elétricas Industriais**. São Paulo: Editora Saber, 2019.

CARR, C. N.; ARAÚJO, A. do E. S. T.; DE SEIXAS, L. do S. F.; DE JESUS, P. C. A.; ARAUJO, D. S.; ARAUJO, E. N. **Modelagem de dados para a internet das coisas (IoT): desafios e soluções ao modelar dados gerados por dispositivos conectados**. **Observatório de la Economía Latinoamericana**, [S. l.], v. 21, n. 12, p. 23858–23874, 2023. DOI: 10.55905/oelv21n12-022. Disponível em: <https://ojs.observatoriolatinoamericano.com/ojs/index.php/olel/article/view/1421>. Acesso em: 15 out. 2025.

FRANÇA, E. L. do C.; FRANÇA, H. F. do C.; NOVAK, S. S.; FARACHE, F. H. A.; VASCONCELOS FILHO, S. C.; MENINO, G. C. de O.; RAMOS, F. M.; COSTA, A. F. da. **Detecção de abelhas nativas do Cerrado Brasileiro com Rede Neural Convolucional YOLOv8**. Caderno Pedagógico, [S. l.], v. 22, n. 11, p. e19970, 2025. DOI: 10.54033/cadpedv22n11-207. Disponível em: <https://ojs.studiespublicacoes.com.br/ojs/index.php/cadped/article/view/19970>. Acesso em: 27 out. 2025.

FREEMAN, Adam; FREEMAN, Steven. **Pro Angular**. 4. ed. Nova York: Apress, 2022.

GARCIA, Laura et al. **IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture**. Sensors, Basel, v. 20, n. 4, p. 1042, fev. 2020. Disponível em: <https://doi.org/10.3390/s20041042>. Acesso em: 15 out. 2025.

HALLIDAY, D.; RESNICK, R.; WALKER, J. **Fundamentals of physics**. 11. ed. Hoboken: Wiley, 2023. ISBN 9781119773517. Nota: Capítulo sobre pressão em fluidos e força hidrostática.

LEITE DIAS DE MATTOS, Viviane; COELHO, Leonardo. **Propriedades de controladores fuzzy: um estudo de caso.** Scientia Plena, v. 11, n. 8, p. 7, ago. 2015. DOI: 10.14808/sci.plena.2015.081306.

LEVIEN, Sérgio Luiz Aguilar; FIGUEIRÊDO, Vladimir Batista; ARRUDA, Luiz Eduardo Vieira de. **Panorama da atual área de agricultura irrigada no Brasil.** Nova Xavantina, MT: Pantanal Editora, 2021. Disponível em: <https://doi.org/10.46420/9786588319963>.

LIU, Xinying; ZHAO, Zhihuan; REZAEIPANAH, Amin. **Intelligent and automatic irrigation system based on internet of things, using fuzzy control technology.** Scientific Reports, v. 15, abr. 2025. Disponível em: <https://doi.org/10.1038/s41598-025-98137-2>.

SILVA, M. D.; BAILAO, A. S. O.; NOVAK, S. S. Analisador e agrupador não supervisionado de strings longas. 2024. Programa de Computador. Registro INPI n. BR5120240048-5, 17 dez. 2024.

NOVAK, S. S.; BAILAO, A. S. O.; SILVA, M. D. **GraphSweetGrapes- Sistema Educacional de Algoritmos de Grafos.** 2024. Programa de Computador. Registro INPI n. BR512025000991-7, 24 out. 2024.

NOVAK, S. S.; DAMKE, C. R. **Cbased – compilador de linguagem baseada em C para Assembly AT&T..** 2025. Programa de Computador. Registro INPI n. BR5120250009-8, 25 mar. 2025.

NOVAK, S. S.; BAILAO, A. S. O.; DAMKE, C. R. **GraphicalAnt - Sistema Educacional de Algoritmos de Colônia de Formigas.** 2025. Programa de Computador. Registro INPI n. BR5120250009-9, 25 mar. 2025.

NOVAK, S. S. et al. **CellularAutomataGraph 3 Sistema Educacional de Autômatos Celulares Elementares e Redes de Autômato Celular do tipo Pequeno Mundo.** 2025. Programa de Computador. Registro INPI n. BR5120250009-9, 25 mar. 2025.

NOVAK, S. S. et al. **Leitor de Sensor de Umidade de Solo e DHT para microcontroladores ESP32 TomAi.** 2025. Programa de Computador. Registro INPI n. BR512025001784-7, 13 maio 2025.

NOVAK, S. S. et al. **Sistema de Agendamentos de Irrigação TomAi.** 2025. Programa de Computador. Registro INPI n. BR512025001773-1, 13 maio 2025.

NOVAK, S. S. et al. **Acionador Bomba hidráulica TomAi para microcontroladores ESP32 TomAi.** 2025. Programa de Computador. Registro INPI n. BR512025001771-5, 13 maio 2025.

NOVAK, S. S. et al. **Sistema de irrigação fuzzy inteligente.** 2025. **Programa de Computador.** Registro INPI n. BR512025001780-4, 13 maio 2025.

NOVAK, S. S. et al. **Servidor python back-end TomAi.** 2025. Programa de Computador. Registro INPI n. BR512025002370-7, 10 jun. 2025.

NOVAK, S. S. et al. **Workshop Tópicos em Internet das Coisas e Visão Computacional com estudo de caso em um sistema de irrigação de tomate cereja.** 2025. Apresentação de Trabalho/Conferência ou palestra.

NOVAK, S. S. **Uso da Lógica Fuzzy na solução de problemas da Agroindústria.** 2025. Apresentação de Trabalho/Conferência ou palestra.

NOVAK, S. S. et al. **Sistema de Agendamentos de Irrigação Tomai.** 2025. Apresentação de Trabalho/Congresso.

NOVAK, S. S. et al. **Acionador bomba hidráulica para microcontroladores ESP32 TOMAI.** 2025. Apresentação de Trabalho/Congresso.

NOVAK, S. S. et al. **Sistema Fuzzy de Irrigação Inteligente.** 2025. Apresentação de Trabalho/Congresso.

NOVAK, S. S. et al. **Implementação prática de um sistema automatizado de irrigação com lógica fuzzy e IoT para horticultura protegida.** ITEM. IRRIGAÇÃO E TECNOLOGIA MODERNA, v. 129, p. 5, 2025.

NOVAK, S. S. et al. **Sistema de irrigação inteligente baseado em lógica fuzzy integrado com internet das coisas para a cultura do tomate cereja.** CADERNO PEDAGÓGICO (LAJEADO. ON-LINE), v. 22, p. e20529, 2025.

MAMDANI, E. H.; ASSILIAN, S. **An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller.** International Journal of Man-Machine Studies, v. 51, n. 1, p. 1-13, jan. 1975. DOI: 10.1016/S0020-7373(75)80002-2.

MARANDUBA, Henrique et al. **Uso da Lógica Fuzzy na avaliação da viabilidade do ciclo de vida do biodiesel.** LALCA - Revista Latino-Americana em Avaliação do Ciclo de Vida, v. 1, n. 1, p. 22, jul. 2017. DOI: 10.18225/lalca.v1i1.3070.

MONK, Simon. **Programação com Arduino: Comece a desenvolver com a plataforma open source de prototipagem rápida.** 2. ed. São Paulo: Novatec Editora, 2017.

OJHA, Varun; ABRAHAM, Ajith; SNASEL, Vaclav. **Heuristic design of fuzzy inference systems: A review of three decades of research.** Engineering Applications of Artificial Intelligence, v. 85, p. 845-864, ago. 2019. DOI: 10.1016/j.engappai.2019.08.010.

OLIVEIRA, Filipe Miranda de; DUSSE, Flávio. **Aplicação de IoT, ESP32 e MQTT em um Aparelho de Ar-condicionado de uma Universidade.** In: Escola Regional de Computação, Bahia, Alagoas e Sergipe (ERBASE), 24., 2024, Salvador/BA. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2024. p. 1-8. DOI: <https://doi.org/10.5753/erbase.2024.4516>.

OLIVEIRA JÚNIOR, A. J.; VICENTIN, T. A.; CUNHA, A. R. da. **Avaliação do sensor DHT11 – medidas de temperatura e umidade relativa do ar.** 2014.

Disponível em: <https://api.semanticscholar.org/CorpusID:170523545>. Acesso em: 19 set. 2025.

ONESCU, Valeriu; ENESCU, Florentina. **Investigating the performance of MicroPython and C on ESP32 and STM32 microcontrollers**. In: International Symposium on Design and Technology of Electronic Packaging (SIITME), 26., 2020, Sibiu, Romênia. Anais [...]. Piscataway: IEEE, 2020. p. 234-237. DOI: 10.1109/SIITME50350.2020.9292199.

ORFALI, Robert; HARKEY, Dan; EDWARDS, Jeri. **Client/Server Survival Guide**. 3. ed. Nova York: John Wiley & Sons, 1999.

RISSINO, Silvia; ROCHA, Helder; CORREIA, Gustavo. **Automação de Sistema de Irrigação com Monitoramento Via Aplicativo Web**. Engenharia na agricultura - REVENG, v. 24, n. 4, p. 314-325, jul. 2016. DOI: 10.13083/1414-3984/reveng.v24n4p314-325.

ROSS, Timothy J. **Fuzzy Logic with Engineering Applications**. 3. ed. Chichester: John Wiley & Sons, 2010.

SANTOS, Roberto et al. **Production of irrigated cherry tomatoes in economical planting beds with mulching**. Revista Caatinga, Mossoró, v. 36, n. 4, p. 907-915, out. 2023. DOI: 10.1590/1983-21252023v36n418rc.

SILVA, André L.; NOGUEIRA, Rafael F.; SANTOS, Carla M. **Introdução à Lógica Fuzzy e suas Aplicações**. Rio de Janeiro: LTC, 2019.

SIMÕES, Mário G.; SHAW, Ian D. **Fuzzy Control Engineering: Theory and Applications**. Hoboken: John Wiley & Sons, 2007.

SOMMERVILLE, Ian. **Engenharia de Software**. 10. ed. São Paulo: Pearson, 2019.

STALLINGS, William. **Arquitetura e Organização de Computadores**. 10. ed. São Paulo: Pearson, 2017.

SULTANA, Rebeca et al. **Measuring Water Quantity Used for Personal and Domestic Hygiene and Determinants of Water Use in a Low-Income Urban Community.** International Journal of Environmental Research and Public Health, v. 19, n. 23, p. 15656, nov. 2022. DOI: 10.3390/ijerph192315656.

TAIZ, L.; ZEIGER, E.; MOLLER, I. M.; MURPHY, A. **Plant physiology and development.** 6. ed. Sunderland: Sinauer Associates, 2017.

VALLE ROMÁN, Jairo; TORRES, Rommel; ENCISO, Liliana; LUDEÑA, Patricia. **Comparación del rendimiento en la transferencia de tráfico en servidores HTTP/2 y QUIC.** Revista Tecnológica – ESPOL, v. 35, n. 3, p. 68-82, 2023. DOI: 10.37815/rte.v35n3.1063.

ZADEH, Lotfi A. **Fuzzy Sets.** Information and Control, v. 8, n. 3, p. 338–353, 1965.

APÊNDICES

Figura A. Sistema de Drenagem. Construção do sistema e drenagem sendo coletada para compor os resultados.

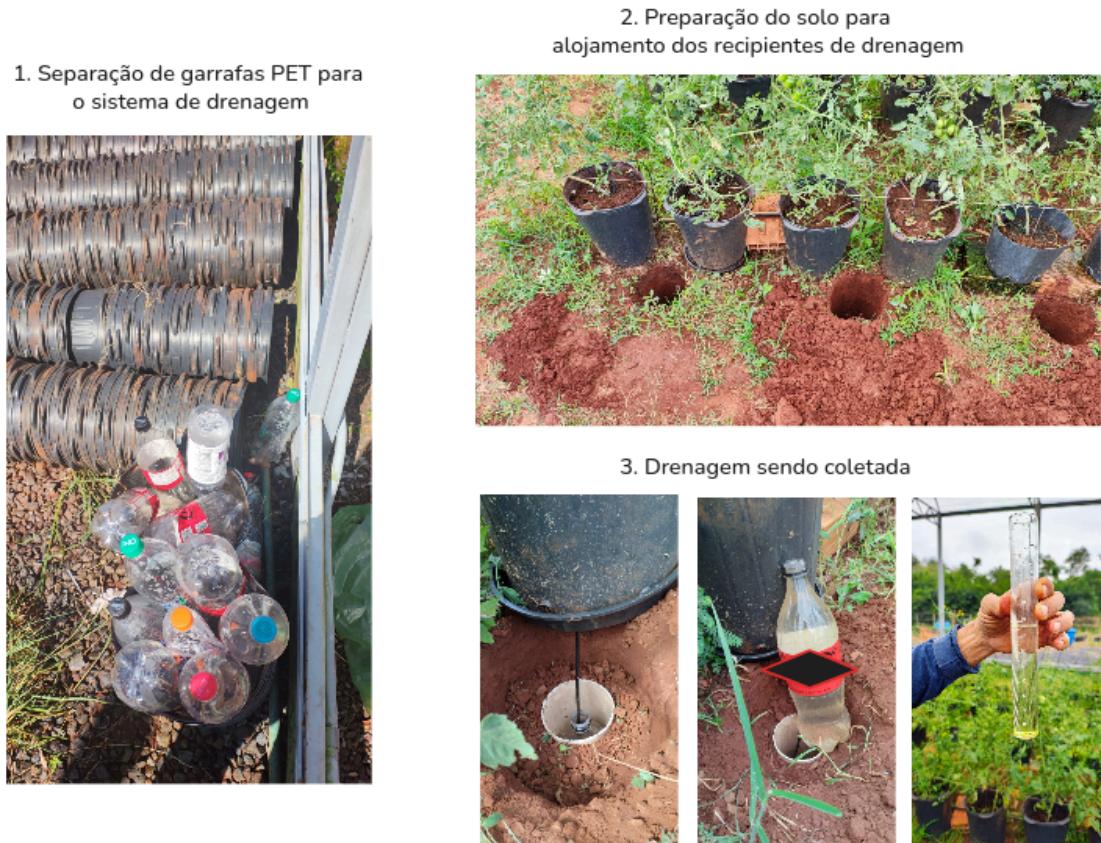


Figura B. Teste para verificar a equidade na vazão dos gotejadores.



Figura C. Apresentação do SILF em um evento.



