

Campus Porto Velho Zona Norte
Coordenação do Curso Superior de Tecnologia em Sistemas para Internet

MÁRCIA AZEVEDO DE SOUZA
RIQUELMI NEVES DO NASCIMENTO

**MONITORAMENTO AMBIENTAL E AUTOMAÇÃO DE IRRIGAÇÃO BASEADOS
EM IOT PARA A AGRICULTURA FAMILIAR EM RONDÔNIA**

PORTO VELHO
2026

**MÁRCIA AZEVEDO DE SOUZA
RIQUELMI NEVES DO NASCIMENTO**

**MONITORAMENTO AMBIENTAL E AUTOMAÇÃO DE IRRIGAÇÃO BASEADOS
EM IOT PARA A AGRICULTURA FAMILIAR EM RONDÔNIA**

Artigo tecnológico entregue como Trabalho de Conclusão de Curso ao Instituto Federal de Educação, Ciência e Tecnologia de Rondônia – (IFRO), *Campus* Porto Velho Zona Norte, como requisito parcial para obtenção do grau de tecnólogo, junto ao Curso de Sistemas Para Internet, sob a orientação do doutor Jhordano Malacarne Bravim e co-orientação do Mestre Renato Almeida de Oliveira.

Ficha catalográfica elaborada pelo Sistema Gerador de Ficha Catalográfica do IFRO.

Souza, Márcia Azevedo de.
Monitoramento ambiental e automação de irrigação baseados em IoT para a agricultura familiar em Rondônia / Márcia Azevedo de Souza, Riquelmi Neves do Nascimento. - Porto Velho, 2026.
37 f. : il.

Orientador(a): Prof. Dr. Jhordano Malacarne Bravim.
Coorientador(a): Prof. Me. Renato Almeida de Oliveira.

Trabalho de Conclusão de Curso (Superior de Tecnologia em Sistemas para Internet) – Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO, Porto Velho, 2026.

1. internet das coisas. 2. home assistant . 3. automação agrícola. 4. agricultura familiar . 5. edge computing. I. Nascimento, Riquelmi Neves do. II. Bravim, Jhordano Malacarne (orient.). III. Oliveira, Renato Almeida de (coorient.). IV. Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO. V. Título.

Bibliotecário(a) Responsável: Gizele de Melo Viana, CRB-11/914

MÁRCIA AZEVEDO DE SOUZA
RIQUELMI NEVES DO NASCIMENTO

**MONITORAMENTO AMBIENTAL E AUTOMAÇÃO DE IRRIGAÇÃO BASEADOS
EM IOT PARA A AGRICULTURA FAMILIAR EM RONDÔNIA**

Artigo tecnológico entregue como Trabalho de Conclusão de Curso ao Instituto Federal de Educação, Ciência e Tecnologia de Rondônia – (IFRO), *Campus* Porto Velho Zona Norte, como requisito parcial para obtenção do grau de tecnólogo, junto ao Curso Superior de Tecnologia em Sistemas para Internet, sob a orientação do professor Jhordano Malacarne Bravim e co-orientação do professor Renato Almeida de Oliveira.

Aprovado em: 29/07/2026 pela banca examinadora.

Documento assinado digitalmente
 JULIANA BRAZ DA COSTA
Data: 05/08/2026 09:40:15-0300
Verifique em <https://validar.iti.gov.br>

Prof Me. Juliana Braz da Costa

Documento assinado digitalmente
 MARIELA MIZOTA TAMADA
Data: 05/08/2026 03:26:29-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dra. Mariela Mizota Tamada

Documento assinado digitalmente
 RENATO ALMEIDA DE OLIVEIRA
Data: 05/08/2026 12:13:46-0300
Verifique em <https://validar.iti.gov.br>

Prof Me. Renato Almeida de Oliveira

Documento assinado digitalmente
 JHORDANO MALACARNE BRAVIM
Data: 05/08/2026 12:22:05-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Jhordano Malacarne Bravim

MONITORAMENTO AMBIENTAL E AUTOMAÇÃO DE IRRIGAÇÃO BASEADOS EM IOT PARA A AGRICULTURA FAMILIAR EM RONDÔNIA

RESUMO: A agricultura familiar em Rondônia representa um pilar estratégico para a segurança alimentar regional, mas enfrenta desafios persistentes relacionados ao manejo agrícola em ambientes de clima quente e úmido, agravados pela baixa adoção de tecnologias digitais acessíveis ao pequeno produtor rural. Este trabalho apresenta o desenvolvimento de um sistema de monitoramento ambiental e automação de irrigação para ambientes agrícolas, fundamentado em tecnologias de Internet das Coisas (IoT) e integrado à plataforma Home Assistant. O sistema utiliza o microcontrolador ESP32 conectado a um sensor DHT22 de temperatura e umidade do ar e a um sensor capacitivo de umidade do solo V1.2, com transmissão de dados em tempo real via protocolo MQTT ao *broker* Mosquitto. O principal diferencial técnico da solução é sua arquitetura de computação de borda (edge computing): o *firmware* embarcado no ESP32 executa a lógica de irrigação de forma autônoma e independente de conectividade, garantindo a continuidade do controle mesmo em caso de queda da rede ou indisponibilidade do Home Assistant, enquanto este atua exclusivamente como camada de supervisão e controle manual. Os testes realizados entre 5 e 9 de junho de 2026 registraram temperaturas entre 27,7°C e 32,3°C, umidade do ar entre 75% e 96% e comunicação MQTT estável por mais de 96 horas contínuas sem interrupções. A solução tem custo de implementação inferior a duzentos reais, utiliza exclusivamente ferramentas de código aberto e opera em rede local sem dependência de infraestrutura de nuvem, demonstrando viabilidade como prova de conceito tecnológico para aplicação na agricultura familiar amazônica.

PALAVRAS-CHAVE: internet das coisas; home assistant; automação agrícola; agricultura familiar; edge computing.

ABSTRACT: Family farming in Rondônia represents a strategic pillar for regional food security, but it faces persistent challenges related to agricultural management in hot and humid climates, aggravated by the low adoption of affordable digital technologies by small-scale farmers. This study presents the development of an environmental monitoring and irrigation automation system for agricultural environments based on Internet of Things (IoT) technologies and integrated with the Home Assistant platform. The system uses an ESP32 microcontroller connected to a DHT22 temperature and air humidity sensor and a V1.2 capacitive soil moisture sensor, with real-time data transmission via the MQTT protocol to the Mosquitto *broker*. The main technical distinction of the proposed solution is its edge computing architecture: the *firmware* embedded in the ESP32 executes the irrigation logic autonomously and independently of network connectivity, ensuring continuous control even in the event of network failures or Home Assistant unavailability, while Home Assistant acts exclusively as the supervision and manual control layer. Tests conducted between June 5 and June 9, 2026, recorded temperatures ranging from 27.7°C to 32.3°C, relative air humidity between 75% and 96%, and stable MQTT communication for more than 96 consecutive hours without interruptions. The solution has an implementation cost of less than two hundred Brazilian reais, relies exclusively on open-source tools, and operates on a local network without dependence on cloud infrastructure, demonstrating its feasibility as a technological proof of concept for application in Amazonian family farming.

KEYWORDS: internet of things; home assistant; agricultural automation; family farming; edge computing.

1 INTRODUÇÃO

O avanço das tecnologias digitais nas últimas décadas inaugurou um novo paradigma de integração entre o mundo físico e o digital, no qual dispositivos embarcados, sensores e sistemas em rede coletam, transmitem e processam dados em tempo real (Gubbi et al., 2013). Esse fenômeno expandiu-se progressivamente para setores como a saúde, a mobilidade urbana, a gestão ambiental e, de forma crescente, a agricultura. Nesse contexto, a Agricultura 4.0 emerge como uma fronteira promissora da inovação tecnológica aplicada, integrando recursos como Internet das Coisas (IoT), automação, sensoriamento e análise de dados para apoiar a tomada de decisão e promover maior eficiência nos processos produtivos rurais (Shahab et al., 2024).

No Brasil, a agricultura familiar representa 77% dos estabelecimentos agropecuários e concentra mais de dois terços do pessoal ocupado no campo, constituindo um segmento essencial para o abastecimento interno e para a segurança alimentar (IBGE, 2022). Em Rondônia, essa participação é ainda mais expressiva. Dos 91.438 estabelecimentos agropecuários existentes no estado, 74.329, aproximadamente 81%, são classificados como pertencentes à agricultura familiar. Essas unidades ocupam cerca de 38% da área total dos estabelecimentos agropecuários rondonienses e apresentam participação relevante na produção de café, mandioca, leite, frutas, feijão e hortaliças (Silva; Silva, 2024).

Além dos dados estatísticos, estudos de campo realizados em pequenas propriedades de diferentes municípios de Rondônia, como Porto Velho, Ariquemes, Cacoal, Vilhena, Ouro Preto d'Oeste e Mirante da Serra, demonstram a diversidade da produção familiar e os desafios enfrentados pelos produtores. Entre as principais dificuldades relatadas está o acesso limitado à Assistência Técnica e Extensão Rural, especialmente para o acompanhamento das atividades produtivas e para a adoção de novas práticas de manejo (Silva; Silva, 2024). Esse cenário reforça a importância de soluções tecnológicas acessíveis, de baixo custo e de fácil replicação, capazes de apoiar o monitoramento das condições ambientais e o uso eficiente de recursos como água e energia.

Apesar do crescimento das tecnologias aplicadas à agricultura, a adoção de ferramentas digitais ainda permanece limitada em parte das propriedades rurais. Barreiras relacionadas aos custos de implantação, à capacitação dos produtores e à

disponibilidade de conectividade dificultam o acesso a sistemas de monitoramento e automação (CETIC.br, 2024; Araújo; Clementino, 2024). Em áreas rurais com infraestrutura digital intermitente, soluções dependentes exclusivamente de serviços externos ou de servidores centralizados podem apresentar baixa confiabilidade operacional.

Embora a literatura apresente diversos estudos sobre monitoramento agrícola baseado em Internet das Coisas (IoT), observa-se que muitas soluções ainda dependem de infraestrutura de comunicação contínua ou de processamento centralizado, o que limita sua aplicação em cenários de conectividade instável. Além disso, são escassos os estudos que integrem computação de borda (edge computing), monitoramento ambiental e automação autônoma em uma arquitetura de baixo custo direcionada às condições da agricultura familiar amazônica. Essa lacuna evidencia a necessidade de soluções capazes de manter suas funcionalidades essenciais mesmo diante de falhas de comunicação com serviços externos.

Diante desse contexto, este trabalho propõe o desenvolvimento de um sistema de monitoramento ambiental e automação de irrigação para ambientes agrícolas, baseado no microcontrolador ESP32, no protocolo MQTT e na plataforma *Home Assistant*. O principal diferencial da solução está na adoção de uma arquitetura de computação de borda (edge computing), na qual o *firmware* embarcado no ESP32 executa localmente a lógica de controle da irrigação. Dessa forma, as funções essenciais do sistema permanecem operacionais mesmo em situações de indisponibilidade da rede Wi-Fi, do *broker* MQTT ou da plataforma Home Assistant, que atua como camada de supervisão, visualização e controle manual.

O objetivo geral deste trabalho é desenvolver um protótipo de sistema integrado de monitoramento ambiental e automação de irrigação para ambientes agrícolas, utilizando o microcontrolador ESP32, sensores *IoT* e a plataforma Home Assistant, visando apoiar a agricultura familiar em Rondônia. Para alcançar esse objetivo, foram definidos os seguintes objetivos específicos: (a) projetar e implementar a arquitetura de hardware integrando ESP32, sensores DHT22, sensor capacitivo de umidade do solo e módulo relé; (b) desenvolver *firmware* com lógica de irrigação autônoma e comunicação MQTT; (c) configurar o *broker* Mosquitto e integrar o sistema ao Home Assistant; (d) construir um dashboard interativo para

visualização e controle em tempo real; e (e) validar o sistema por meio de testes em ambiente controlado.

O presente artigo está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica; a Seção 3 discute os trabalhos relacionados; a Seção 4 descreve a metodologia e o desenvolvimento; a Seção 5 apresenta os resultados e a discussão; e a Seção 6 apresenta as considerações finais.

2. Fundamentação Teórica

Esta seção apresenta os fundamentos teóricos que sustentam o desenvolvimento da solução proposta neste trabalho. São abordados os conceitos de Internet das Coisas (IoT), Agricultura 4.0, computação de borda, protocolo MQTT, microcontrolador ESP32, sensores utilizados e a plataforma *Home Assistant*. A discussão desses elementos busca demonstrar sua relevância para o desenvolvimento de um sistema de monitoramento ambiental e automação de irrigação de baixo custo, resiliente e adequado às necessidades da agricultura familiar em Rondônia.

2.1. Internet das Coisas e Agricultura 4.0

A Internet das Coisas (IoT) é um paradigma tecnológico que propõe a interconexão de objetos físicos à internet, permitindo que dispositivos coletem, processem e compartilhem dados de forma autônoma, sem necessidade de intervenção humana direta (Gubbi et al., 2013). Do ponto de vista arquitetural, os sistemas IoT são organizados em três camadas funcionais: a camada de percepção, responsável pela coleta de dados por meio de sensores e atuadores; a camada de rede, que engloba os protocolos e infraestruturas de transmissão; e a camada de aplicação, onde os dados são processados e apresentados ao usuário (Gubbi et al., 2013). Essa estrutura em camadas confere aos sistemas IoT modularidade, escalabilidade e flexibilidade.

Aplicada à agricultura, a *IoT* viabiliza o monitoramento contínuo de variáveis como temperatura, umidade do ar, umidade do solo e luminosidade, possibilitando uma gestão ambiental mais precisa e a automação de processos como a irrigação (Morchid et al., 2024). Shahab et al. (2024) destacam que sistemas de irrigação

inteligente baseados em sensores de umidade do solo podem reduzir o consumo hídrico em até 30% em comparação com sistemas convencionais por horário fixo, além de contribuir para a sustentabilidade e a resiliência dos sistemas agrícolas frente às mudanças climáticas. Apesar dessas oportunidades, Morchid et al. (2024) ressaltam que a adoção enfrenta barreiras significativas, como conectividade limitada em áreas rurais, baixa familiaridade tecnológica dos produtores e custos iniciais de implantação (CETIC.br, 2024; Araújo e Clementino, 2024).

Diante desse contexto, a *IoT* constitui a base tecnológica da solução proposta neste trabalho, pois possibilita a aquisição contínua de dados ambientais e a automação da irrigação a partir de informações coletadas em tempo real. Sua aplicação mostra-se especialmente adequada à agricultura familiar, uma vez que permite desenvolver soluções de baixo custo, escaláveis e capazes de contribuir para o uso mais eficiente dos recursos naturais.

2.2. Computação de Borda em Sistemas IoT Agrícolas

A computação de borda (edge computing) consiste no processamento de dados próximo à fonte geradora, em vez de enviá-los a um servidor centralizado para processamento. Em sistemas *IoT*, essa abordagem reduz a latência das decisões, diminui a dependência de conectividade contínua e aumenta a resiliência operacional do sistema (Morchid et al., 2024). No contexto da agricultura familiar em zonas rurais com infraestrutura digital intermitente, o edge computing é especialmente relevante: um sistema que depende exclusivamente de um servidor central para tomar decisões de irrigação torna-se inoperante em caso de queda de conexão, com potenciais impactos diretos sobre os cultivos.

No sistema proposto neste trabalho, o ESP32 atua como dispositivo de borda, executando localmente toda a lógica de controle da irrigação, enquanto o Home Assistant é utilizado para supervisão, visualização e controle manual. Dessa forma, mesmo diante da indisponibilidade da rede Wi-Fi, do *broker* MQTT ou da plataforma de supervisão, as funções essenciais de irrigação permanecem operacionais. Essa característica torna a computação de borda uma estratégia particularmente adequada para aplicações agrícolas em regiões com infraestrutura de comunicação limitada, justificando sua adoção na arquitetura desenvolvida.

2.3. Protocolo MQTT e Broker Mosquitto

O MQTT (Message Queuing Telemetry Transport) é um protocolo leve de comunicação baseado no modelo publicação/assinatura (publish/subscribe), desenvolvido originalmente para ambientes com restrições de largura de banda e alta latência (Banks et al., 2019; Hunkeler et al., 2008). No modelo publish/subscribe, um *broker* central recebe as mensagens publicadas pelos dispositivos em tópicos específicos e as redistribui a todos os dispositivos inscritos nesses tópicos, desacoplando logicamente produtores e consumidores de dados (Banks et al., 2019). O protocolo define três níveis de Qualidade de Serviço (QoS): QoS 0 (entrega no máximo uma vez), QoS 1 (entrega pelo menos uma vez) e QoS 2 (entrega exatamente uma vez). Para aplicações de monitoramento em tempo real, o QoS 0 é suficiente, pois leituras perdidas são compensadas pelo próximo ciclo de publicação.

O Eclipse Mosquitto, *broker* MQTT adotado neste trabalho, é um projeto de código aberto amplamente utilizado em aplicações IoT, com suporte a autenticação por credenciais e comunicação cifrada via TLS/SSL (Banks et al., 2019). No sistema proposto, os tópicos MQTT seguem uma estrutura hierárquica organizada por dispositivo e tipo de dado. Um exemplo é o tópico `estufa/esp32_01/sensor/temperatura`, utilizado para publicar as leituras de temperatura. Essa organização facilita a manutenção do sistema e a inclusão de novos dispositivos.

A escolha do protocolo MQTT deve-se à sua baixa sobrecarga de comunicação, simplicidade de implementação e ampla utilização em aplicações IoT. Essas características possibilitam a troca eficiente de dados entre o ESP32 e o Home Assistant, reduzindo o consumo de largura de banda e tornando a comunicação mais adequada para ambientes com recursos computacionais e infraestrutura de rede limitados, como aqueles encontrados em parte das propriedades da agricultura familiar.

2.4. Microcontrolador ESP32

O ESP32 é um microcontrolador desenvolvido pela *Espressif Systems* que integra, em um único chip, processador *dual-core* Xtensa LX6 com frequência de até 240 MHz, memória RAM de 520 KB, suporte nativo a Wi-Fi (IEEE 802.11 b/g/n) e

Bluetooth, além de conversores analógico-digitais (ADC) de 12 bits e múltiplas interfaces de comunicação (Espressif Systems, 2023). Seu custo reduzido, a ampla disponibilidade de bibliotecas para o ambiente Arduino IDE e a comunidade ativa de desenvolvedores tornam o ESP32 uma plataforma adequada para o desenvolvimento de sistemas IoT destinados à agricultura familiar, pois combinam capacidade de processamento, conectividade sem fio e baixo custo em um único dispositivo. Além disso, a possibilidade de executar localmente a lógica de controle permite implementar estratégias de computação de borda, aumentando a autonomia e a confiabilidade do sistema desenvolvido. Por essas razões, o ESP32 foi escolhido como elemento central da arquitetura proposta neste trabalho.

2.5. Sensores Utilizados

O sensor DHT22 (AM2302) realiza medições simultâneas de temperatura e umidade relativa do ar por meio de um elemento capacitivo e um termistor internos, com faixa de temperatura de -40°C a $+80^{\circ}\text{C}$ ($\pm 0,5^{\circ}\text{C}$) e faixa de umidade de 0% a 100% ($\pm 2\%$) (Aosong Electronics, 2023). Em comparação com o modelo anterior DHT11, o DHT22 apresenta maior faixa de medição, menor incerteza e melhor desempenho em ambientes com variações bruscas de umidade. Essas características são adequadas ao clima equatorial de Rondônia, marcado por temperaturas elevadas e alta umidade durante o ano.

Características estas que justificam a escolha do DHT22 neste trabalho, uma vez que o sensor oferece um equilíbrio entre custo, precisão e facilidade de integração com plataformas embarcadas, sendo adequado para aplicações de monitoramento ambiental em pequenas propriedades rurais. A utilização do sensor capacitivo também se justifica por sua maior durabilidade em comparação aos sensores resistivos, reduzindo a necessidade de manutenção decorrente da oxidação dos eletrodos. Essa característica é especialmente relevante em aplicações agrícolas de uso contínuo, nas quais a confiabilidade e o baixo custo de manutenção são fatores importantes.

O sensor capacitivo de umidade do solo V1.2 opera por variação da capacitância elétrica do meio, dispensando o contato elétrico direto com a água e conferindo maior durabilidade em aplicações contínuas em ambientes úmidos em relação aos modelos resistivos, que sofrem degradação acelerada por oxidação dos

eletrodos (Morchid et al., 2024). O sinal analógico de saída é lido pelo ADC do ESP32 no pino GPIO 34 e convertido em percentual de umidade por mapeamento linear entre valores de calibração experimental: 3300 para solo completamente seco e 1300 para solo completamente saturado.

2.6. Home Assistant

O Home Assistant é uma plataforma de automação de código aberto, desenvolvida em Python e mantida pela Open Home Foundation, que permite integrar dispositivos e serviços de diferentes fabricantes em um único ambiente de controle centralizado, com suporte nativo ao protocolo MQTT [Home Assistant, 2024]. A plataforma oferece sistema de entidades para representação abstrata de sensores e atuadores, sistema de automações baseado em gatilhos e ações, e dashboards personalizáveis acessíveis por navegador web e aplicativo móvel. Neste trabalho, o Home Assistant foi instalado em uma máquina virtual com VirtualBox, configurada no modo de rede Bridge. Essa configuração elimina a necessidade de hardware dedicado, como o Raspberry Pi, e permite a operação em rede local, sem dependência de infraestrutura de nuvem pública, reduzindo custos e aumentando a autonomia do sistema. A escolha da plataforma também se fundamenta em sua natureza de código aberto, ampla compatibilidade com dispositivos IoT e integração nativa com o protocolo MQTT, características que permitem construir uma interface de supervisão flexível, escalável e adequada aos objetivos deste trabalho.

3 Trabalhos Relacionados

O Quadro 1 sintetiza os principais trabalhos relacionados ao sistema proposto, comparando hardware, sensores, plataformas e contribuições.

Quadro 1 – Síntese dos trabalhos relacionados.

Autor/Ano	Hardware	Sensores	Plataforma	Contribuição principal
OLIVEIRA et al. (2024)	Microcontrolador	Temp./umidad e	MQTT + visualizador	MQTT + visualizador
MORCHID et al. (2024)	Revisão	Variados	Diversas	Benefícios e

				desafios IoT na agricultura
SHAHAB et al. (2024)	Revisão	Solo/clima	Diversas	Redução de até 30% no consumo hídrico
STROPARO (2024)	Hardware aberto	Variados	Open-source	IoT acessível para agricultura familiar
ARAÚJO; CLEMENTINO (2024)	Revisão	-	-	Inclusão tecnológica na agricultura familiar brasileira
Este trabalho	ESP32	DHT22 + capacitivo	HA + MQTT local	Edge computing + controle híbrido autônomo sem nuvem

Fonte: Elaborado pelos autores.

Oliveira et al. (2024) demonstraram a viabilidade de sistemas IoT para monitoramento agrícola com microcontroladores e MQTT, destacando a superioridade dos sensores capacitivos em ambientes úmidos. O presente trabalho se alinha a essa conclusão e inova pela adoção de arquitetura de edge computing com hierarquia clara de controle e pela integração com Home Assistant local sem dependência de nuvem pública, tornando a solução mais robusta para zonas rurais com conectividade intermitente.

Morchid et al. (2024) identificaram os custos iniciais e a conectividade limitada como barreiras à adoção de IoT na agricultura familiar. Para reduzir essas limitações, o sistema proposto utiliza hardware de código aberto, opera em rede local e possui custo inferior a R\$ 200,00 e operação em rede local. Shahab et al. (2024) destacaram que a automação por sensores de umidade do solo pode reduzir o consumo hídrico em até 30%, abordagem implementada neste trabalho com o diferencial do controle autônomo embarcado que opera independentemente do servidor de supervisão. Araújo e Clementino (2024) evidenciam que a inclusão

tecnológica na agricultura familiar brasileira ainda é incipiente, reforçando a relevância de soluções acessíveis e replicáveis como a proposta neste trabalho

4 METODOLOGIA E DESENVOLVIMENTO

4.1. Caracterização da Pesquisa

Este trabalho caracteriza-se como pesquisa aplicada de natureza tecnológica, pois tem como produto final um sistema funcional desenvolvido para resolver um problema concreto. Do ponto de vista dos objetivos, é exploratória e experimental, envolvendo a investigação de tecnologias emergentes aplicadas a um contexto regional pouco estudado e a construção e testagem de um protótipo funcional em ambiente controlado (Oliveira et al., 2024). Os procedimentos técnicos combinam pesquisa bibliográfica em bases como Google Scholar, Portal CAPES e IEEE Xplore, pesquisa documental por meio de análise de manuais e datasheets dos componentes utilizados, e pesquisa experimental materializada no desenvolvimento, implementação e validação do protótipo.

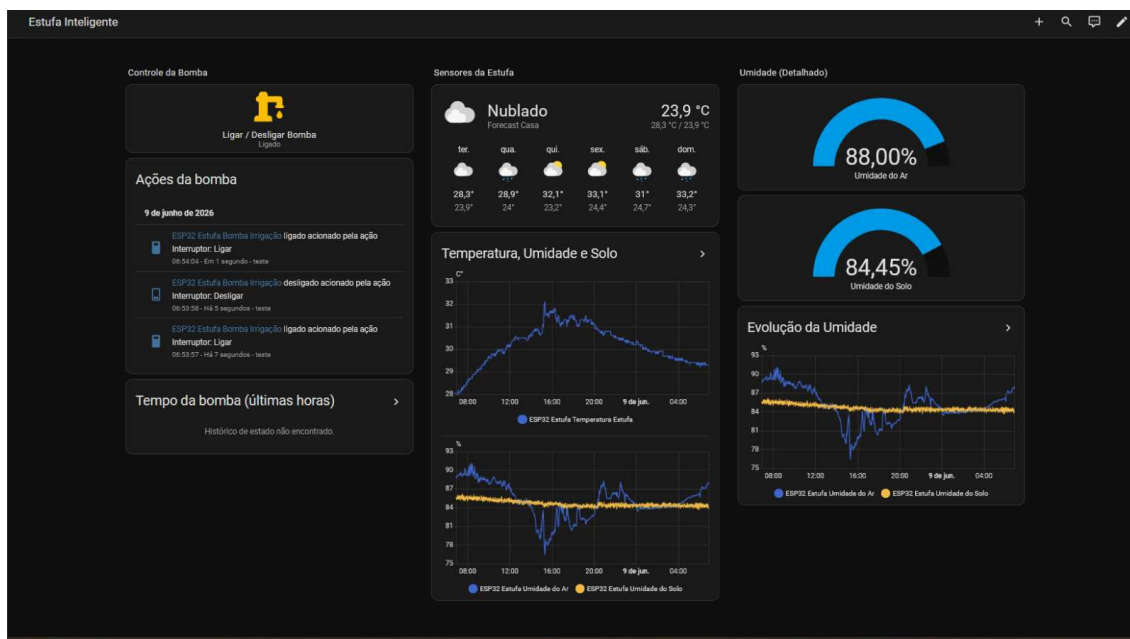
4.2. Arquitetura Geral do Sistema

O sistema segue o modelo de três camadas da IoT (Gubbi et al., 2013): (1) camada de sensoriamento, composta pelo ESP32 e pelos sensores físicos conectados, responsável pela coleta das variáveis ambientais e pelo acionamento da irrigação; (2) camada de comunicação, constituída pelo protocolo MQTT e pelo *broker* Mosquitto, que mediam a troca de mensagens entre o ESP32 e o Home Assistant; e (3) camada de aplicação, representada pelo Home Assistant, responsável pela visualização centralizada dos dados, pelo controle manual e pela geração de alertas.

O diferencial arquitetural fundamental deste sistema é a separação hierárquica e clara entre as responsabilidades de cada camada. O ESP32 atua como dispositivo de borda, detendo a inteligência principal do laço de automação e operando de forma completamente autônoma. O Home Assistant é responsável pela supervisão e pelo controle manual. A lógica automática de irrigação permanece armazenada e executada no ESP32.

Essa arquitetura garante que o sistema continue irrigando corretamente mesmo em caso de queda da rede Wi-Fi ou indisponibilidade do servidor Home Assistant, característica essencial para zonas rurais com conectividade intermitente (CETIC.br, 2024). A Figura 1 apresenta o dashboard do sistema em operação.

Figura 1 – Dashboard Estufa Inteligente no Home Assistant em operação



Fonte: Home Assistant (2024).

4.3. Hardware e Evolução do Protótipo

O protótipo passou por duas fases evolutivas que resultaram em melhorias significativas de confiabilidade e precisão. Na versão inicial, foram utilizados o sensor DHT11 para medição ambiental, um sensor resistivo de umidade do solo e protoboard para as conexões elétricas. Durante os testes dessa versão, foram identificadas três limitações principais: precisão insuficiente do DHT11 em ambientes com variações bruscas de umidade; degradação acelerada do sensor resistivo de solo por oxidação dos eletrodos em contato com o meio úmido; e instabilidade das conexões em protoboard, que resultava em leituras inconsistentes.

A versão final do protótipo incorporou as seguintes melhorias: substituição do DHT11 pelo DHT22, com maior precisão e faixa de medição mais ampla (Aosong Electronics, 2023); substituição do sensor resistivo pelo capacitivo V1.2, com maior durabilidade e leituras mais estáveis (Morchid et al., 2024); e adoção de Shield

expansor de portas para ESP32 (30 pinos) em substituição à protoboard, proporcionando conexões mais firmes e organizadas.

O Quadro 2 apresenta o mapeamento detalhado das conexões elétricas da versão final do protótipo. Esse registro é relevante para a reprodutibilidade do projeto, pois permite identificar com precisão os pinos utilizados por cada sensor e atuador, facilitando a reconstrução do circuito, a verificação da montagem e a identificação de possíveis erros de conexão por outros pesquisadores ou desenvolvedores.

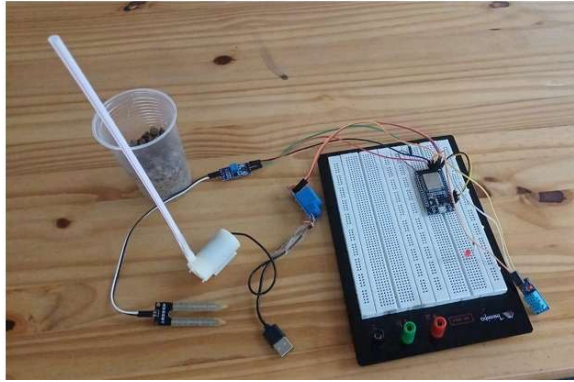
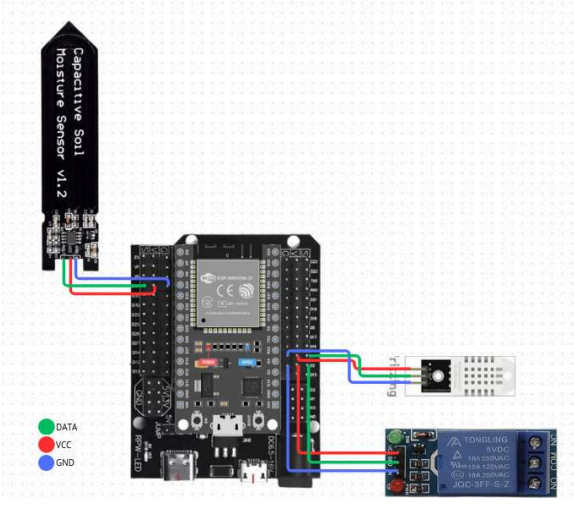
Quadro 2 – Mapa de conexões dos componentes ao ESP32.

Componente	Pino	Pino ESP32
Sensor DHT22	DATA	GPIO 4
Sensor DHT22	VCC	3V3
Sensor DHT22	GND	GND
Sensor de solo	AO	GPIO 34
Sensor de solo	VCC	3V3
Sensor de solo	GND	GND
Módulo Relé	IN	GPIO 2
Módulo Relé	VCC	5V
Módulo Relé	GND	GND

Fonte: Elaborado pelos autores.

O Quadro 3 apresenta um comparativo visual entre as duas fases do protótipo. A versão inicial, montada em protoboard, foi fundamental para a validação rápida do conceito e permitiu ajustes durante os primeiros testes. A versão final, com Shield expensor, apresenta organização superior das conexões e maior estabilidade elétrica, reduzindo significativamente a ocorrência de falhas por mau contato.

Quadro 3 – Evolução do protótipo

a) Versão inicial em protoboard	
b) Versão final com Shield expansor	

Fonte: Elaborado pelos autores.

4.4. Firmware e Lógica de Controle Híbrida

O *firmware* foi desenvolvido em C++ no Arduino IDE e implementa uma arquitetura de controle híbrida com hierarquia clara de prioridades, projetada para eliminar conflitos entre as camadas de automação (Espressif Systems, 2023). O Home Assistant tem prioridade máxima: qualquer comando enviado por meio do tópico MQTT estufa/esp32_01/atuador/bomba/set é registrado com timestamp e executado imediatamente pelo ESP32.

A automação embarcada no *firmware* entra em ação apenas quando duas condições são simultaneamente satisfeitas: nenhum comando do Home Assistant foi recebido nos últimos três minutos e a umidade do solo permanece abaixo de 30% por pelo menos três minutos consecutivos. Esse intervalo é configurável e busca

evitar acionamentos provocados por leituras espúrias, além de garantir que o Home Assistant tenha prioridade de resposta.

Quando acionada autonomamente, a bomba é desligada quando a umidade do solo atinge 70% ou após dez minutos de operação contínua, período definido como limite de segurança do sistema.

Essa arquitetura elimina o conflito de controle entre as camadas. O ESP32 detém a inteligência do laço fechado de automação, enquanto o Home Assistant oferece visibilidade e controle manual ao operador. O sistema continua operando corretamente mesmo sem conectividade, característica que o diferencia de soluções que dependem exclusivamente de um servidor central para tomar decisões de irrigação.

O sensor capacitivo de umidade do solo é lido como média aritmética de 20 amostras ADC consecutivas, coletadas em intervalos de 10 ms. Essa técnica reduz o ruído do sinal e melhora a precisão da medição. Os valores brutos são convertidos em percentual por meio de mapeamento linear entre os valores de calibração experimental: 3300 para solo seco e 1300 para solo saturado. Após cada ciclo de irrigação, o *firmware* publica via MQTT o tempo de acionamento em segundos e o volume estimado de água em litros.

É importante destacar, como limitação de hardware, que o volume de água irrigado é estimado a partir do tempo de acionamento multiplicado pela vazão nominal da bomba, de 1,5 L/min, sem a utilização de um sensor de fluxo. Em caso de obstrução na tubulação, perda de carga ou falta d'água no reservatório, o valor publicado pode não representar o volume real dispensado. A incorporação de um fluxômetro ao sistema é identificada como melhoria prioritária para versões futuras.

4.5. Configuração do Home Assistant

O *Home Assistant* foi instalado em máquina virtual com VirtualBox em modo de rede Bridge, obtendo endereço IP próprio na rede local sem necessidade de hardware dedicado, o que reduz o custo total da solução e facilita sua replicação em qualquer computador disponível. O *broker* Mosquitto foi configurado como complemento (add-on) nativo do Home Assistant com autenticação por usuário e senha, impedindo acesso não autorizado ao sistema.

Foram criadas seis entidades MQTT: sensores de temperatura, umidade do ar, umidade do solo, valor bruto do ADC, tensão do sensor de solo e tempo de acionamento da bomba; além de um interruptor para controle manual da bomba. O dashboard denominado Estufa Inteligente foi construído em YAML com cartões de medidor (gauge), gráfico histórico (history graph), registro de ações (logbook) e previsão meteorológica integrada via API, permitindo ao usuário monitorar as condições ambientais e controlar a irrigação de qualquer dispositivo conectado à rede local (Home Assistant, 2024).

5 RESULTADOS E DISCUSSÃO

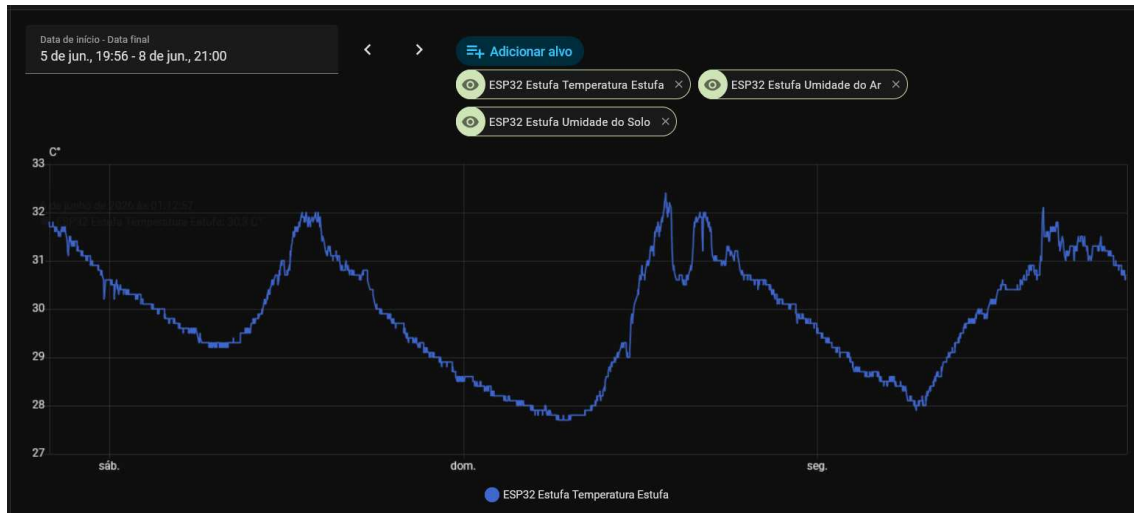
Os testes foram realizados entre 5 e 9 de junho de 2026, com o sistema em operação contínua por mais de 96 horas, coletando e publicando dados a cada 30 segundos via MQTT. Ao longo desse período, o sistema publicou aproximadamente 11.520 mensagens MQTT sem registrar falhas de comunicação ou interrupções de conectividade, demonstrando a estabilidade do protocolo para aplicações de monitoramento contínuo em rede local (Banks et al., 2019).

Os resultados obtidos permitem avaliar o desempenho do protótipo desenvolvido em relação ao objetivo geral desta pesquisa, que consiste em implementar um sistema integrado de monitoramento ambiental e automação de irrigação baseado em tecnologias IoT. As análises apresentadas a seguir evidenciam a capacidade do sistema de realizar monitoramento contínuo, comunicação confiável entre os dispositivos e operação autônoma da lógica de controle em ambiente controlado.

5.1. Monitoramento de Temperatura

A Figura 3 apresenta o gráfico histórico de temperatura registrado pelo sensor DHT22 no período de 5 a 8 de junho de 2026.

Figura 3 – Temperatura registrada pelo DHT22 (5 a 8 de junho de 2026).



Fonte: Home Assistant (2024).

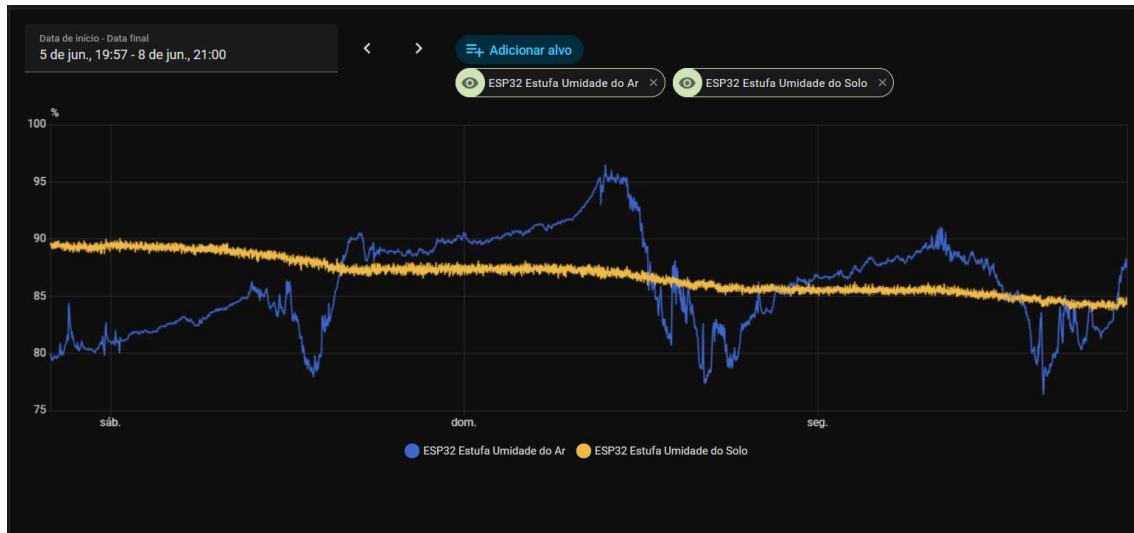
Os dados revelam padrão cíclico bem definido, característico do clima equatorial de Rondônia (Mazer, 2022): temperatura mínima de aproximadamente 27,7°C, observada nas madrugadas, e máxima de cerca de 32,3°C durante os períodos diurnos, com amplitude térmica média de 4,6°C. Três ciclos completos dia/noite são claramente visíveis no gráfico, evidenciando a regularidade das leituras e confirmando a confiabilidade do sensor DHT22 ao longo de 72 horas de operação contínua. A ausência de picos ou quedas abruptas nas leituras indica ausência de falhas de comunicação ou erros de leitura do sensor durante o período avaliado.

Esses resultados demonstram que o sistema foi capaz de realizar o monitoramento contínuo da temperatura ambiente com estabilidade e confiabilidade durante todo o período de testes. Dessa forma, confirma-se que o protótipo atende ao objetivo de adquirir e disponibilizar informações ambientais em tempo real, requisito fundamental para aplicações de monitoramento agrícola baseadas em IoT.

5.2. Monitoramento de Umidade

A Figura 4 apresenta o gráfico comparativo de umidade do ar e do solo no mesmo período.

Figura 4 – Umidade do ar (azul) e do solo (amarelo), 5 a 8 de junho de 2026.



Fonte: Home Assistant (2024).

A umidade do ar variou entre 75% e 96%, com picos pronunciados nos períodos noturnos e quedas durante os horários de maior incidência solar, comportamento coerente com a dinâmica climática amazônica onde a umidade relativa do ar varia inversamente com a temperatura (Mazer, 2022). A umidade do solo manteve-se estável entre 83% e 90% durante todo o período de monitoramento contínuo, permanecendo acima do limiar de 30% configurado para acionamento automático da irrigação.

Essa estabilidade é característica do ambiente laboratorial controlado utilizado nos testes: o substrato foi previamente umedecido e mantido em recipiente fechado, sem as condições de evapotranspiração típicas de um ambiente agrícola real. Em campo real, com maior volume de substrato e exposição às condições climáticas externas, espera-se variação mais expressiva da umidade do solo, acionando a lógica automática com maior frequência. A estabilidade das leituras do sensor capacitivo V1.2 ao longo de 96 horas corrobora sua superioridade em relação ao modelo resistivo em termos de durabilidade e consistência das medições, conforme apontado por Morchid et al. (2024). A análise conjunta dos gráficos de temperatura e umidade do ar revela uma correlação inversa entre as duas variáveis ao longo do período monitorado, comportamento esperado e consistente com a dinâmica climática da região amazônica (Mazer, 2022). Nos períodos diurnos, quando a temperatura atingia valores próximos a 32,3°C, a umidade do ar tendia a cair para seus valores mínimos, em torno de 75%. No período noturno, com a queda

da temperatura para aproximadamente 27,7°C, a umidade do ar subia para valores próximos a 96%. Essa variação cíclica e regular, observada nos três ciclos dia/noite registrados durante o período de testes, demonstra que o sensor DHT22 capturou com fidelidade as condições ambientais reais do local de instalação do protótipo, sem apresentar deriva de leitura ou degradação de desempenho ao longo do tempo.

Essa correlação inversa entre temperatura e umidade do ar tem implicações práticas relevantes para o manejo de ambientes agrícolas na região de Rondônia. Em dias de alta temperatura e baixa umidade do ar, a taxa de evapotranspiração das plantas tende a aumentar. Nessas condições, o sistema de monitoramento contínuo permite ao produtor identificar situações de estresse hídrico e acionar a irrigação preventivamente, antes que os cultivos sejam afetados. Essa capacidade de resposta antecipada, viabilizada pelo monitoramento em tempo real via MQTT e pela visualização no dashboard do Home Assistant, representa uma das principais contribuições práticas da solução para a agricultura familiar (Morchid et al., 2024; Shahab et al., 2024).

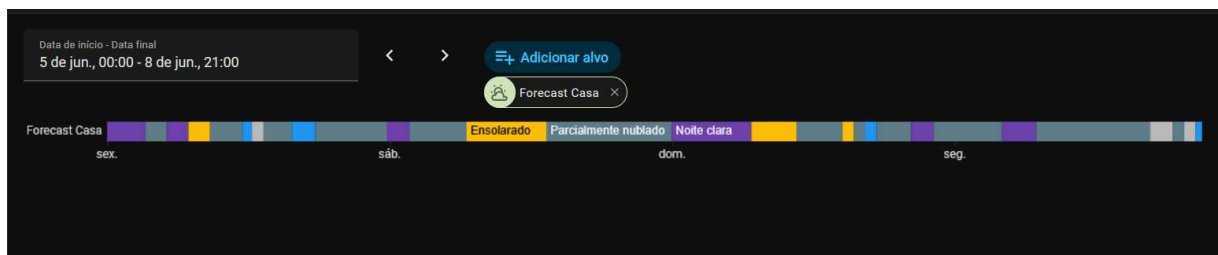
Adicionalmente, a estabilidade da comunicação MQTT ao longo de mais de 96 horas contínuas de operação merece destaque. Com leituras publicadas a cada 30 segundos, o sistema realizou aproximadamente 11.520 publicações MQTT no período sem registrar nenhuma falha de entrega ou interrupção de conectividade. Esse resultado valida a escolha do protocolo MQTT como meio de comunicação para aplicações de monitoramento agrícola contínuo em redes locais, confirmando as características de leveza e confiabilidade descritas por Banks et al. (2019) e Hunkeler et al. (2008). A ausência de falhas de comunicação durante o período de testes também confirma a estabilidade da configuração do *broker* Mosquitto como complemento nativo do Home Assistant, sem necessidade de reinicializações ou intervenções manuais durante todo o período avaliado (Home Assistant, 2024).

Em conjunto, os resultados de temperatura, umidade do ar e umidade do solo demonstram que o sistema foi capaz de monitorar simultaneamente diferentes variáveis ambientais e disponibilizar essas informações de forma contínua e confiável. Esses resultados confirmam que a arquitetura proposta atende ao objetivo de integrar monitoramento ambiental e comunicação em tempo real, fornecendo dados que podem subsidiar decisões relacionadas ao manejo da irrigação na agricultura familiar.

5.3. Previsão Meteorológica Integrada

A Figura 5 apresenta a visualização da previsão meteorológica integrada ao dashboard via API do Home Assistant, registrada no período de 5 a 8 de junho de 2026.

Figura 5 – Previsão meteorológica integrada ao dashboard via API do Home Assistant



Fonte: Home Assistant (2024).

A integração da previsão meteorológica ao dashboard representa uma funcionalidade adicional de relevância prática para o agricultor, pois permite correlacionar as condições ambientais registradas pelos sensores com as condições climáticas externas previstas. Durante o período de testes, foram registradas condições de tempo ensolarado, parcialmente nublado e de noite clara. Essas informações poderão ser utilizadas no desenvolvimento de automações mais avançadas, como a suspensão da irrigação quando houver previsão de chuva, contribuindo para a redução do consumo de água (Home Assistant, 2024; Shahab et al., 2024).

Embora essa funcionalidade não componha diretamente a lógica automática de irrigação implementada neste trabalho, sua integração amplia o potencial de utilização do sistema, demonstrando a flexibilidade da arquitetura desenvolvida e sua capacidade de incorporar novas fontes de dados para apoiar futuras estratégias de automação agrícola.

5.4. Validação da Lógica de Controle

A lógica de irrigação automática embarcada no ESP32 não foi acionada durante o período de monitoramento contínuo, pois a umidade do solo permaneceu acima do limiar de 30% configurado. Contudo, a lógica foi validada por meio de testes específicos nos quais o limiar foi temporariamente ajustado para valores

acima da umidade registrada, confirmando o correto funcionamento da automação embarcada. Da mesma forma, a hierarquia de controle entre o ESP32 e o Home Assistant foi validada por meio de testes de simulação de queda de conectividade: com o Home Assistant desligado, o ESP32 continuou executando a lógica de irrigação autonomamente, confirmando o diferencial de edge computing da solução (Espressif Systems, 2023).

O registro de ações do dashboard confirmou tempo de resposta inferior a um segundo entre o envio de comandos pelo Home Assistant e a execução pelo relé, validando a eficácia do protocolo MQTT para aplicações de controle em tempo real em redes locais (Banks et al., 2019). O controle manual da bomba pelo dashboard também foi testado com sucesso, confirmando a comunicação bidirecional entre Home Assistant e ESP32 e a coexistência harmônica das duas camadas de controle sem conflitos (Home Assistant, 2024).

Os testes realizados demonstram que a arquitetura baseada em computação de borda atingiu o comportamento esperado, mantendo a lógica de irrigação ativa mesmo na ausência da plataforma de supervisão. Esse resultado confirma o principal diferencial da solução proposta e evidencia que o objetivo de desenvolver um sistema com capacidade de operação autônoma foi alcançado em ambiente controlado.

5.5. Análise de Custos do Protótipo

O Quadro 4 apresenta o levantamento dos componentes utilizados no protótipo desenvolvido neste trabalho e seus respectivos custos médios praticados no mercado nacional, obtidos no momento da aquisição dos materiais.

Quadro 4 – Custo dos componentes do protótipo laboratorial.

Componente	Qtd.	Custo aprox. (R\$)
ESP32 ESP-WROOM-32	1	45,00
Sensor DHT22	1	20,00
Sensor capacitivo solo V1.2	1	15,00
Módulo relé 5V 1 canal	1	12,00
Mini bomba submersa	1	18,00
Shield expensor ESP32	1	35,00

Cabos, jumpers e outros	-	30,00
-------------------------	---	-------

Fonte: Elaborado pelos autores.

O custo total do protótipo laboratorial foi de aproximadamente R\$ 175,00 (cento e setenta e cinco reais) em componentes eletrônicos, utilizando exclusivamente software gratuito e de código aberto. É importante destacar que esses valores referem-se ao protótipo em escala reduzida desenvolvido para validação do conceito. Uma implantação em escala real em ambiente agrícola demandaria componentes adicionais, como bomba de irrigação de maior capacidade, tubulação, fonte de alimentação adequada e estrutura física de suporte. Como os custos variam conforme o tamanho do cultivo e as condições locais, cada implantação deve possuir um orçamento específico (Morchid et al., 2024; Shahab et al., 2024).

Neste estudo, o protótipo foi validado em ambiente laboratorial controlado, utilizando um recipiente contendo substrato agrícola umedecido, no qual foi instalado o sensor capacitivo de umidade do solo e o sistema de irrigação em escala experimental. Não houve cultivo de uma cultura agrícola específica nem delimitação de uma área em metros quadrados, uma vez que o objetivo desta etapa foi validar o funcionamento da arquitetura eletrônica, da comunicação entre os dispositivos e da lógica de automação, e não avaliar o desempenho agrônômico do sistema em condições reais de campo.

A análise de custos demonstra que o protótipo pode ser implementado utilizando componentes eletrônicos de baixo custo e softwares livres, corroborando a proposta de desenvolver uma solução economicamente acessível para aplicações na agricultura familiar. Embora a implantação em escala real demande adaptações, o protótipo valida a viabilidade técnica e econômica da arquitetura proposta.

5.6. Discussão e Comparação com Trabalhos Relacionados

Em comparação com os trabalhos relacionados, a solução proposta se diferencia por três aspectos principais. Primeiro, a integração com Home Assistant executado localmente em máquina virtual sem dependência de nuvem pública, que torna a solução mais adequada para zonas rurais com internet intermitente e elimina custos de serviços de nuvem. Segundo, a arquitetura de controle híbrida com

hierarquia clara, que resolve o conflito entre controle embarcado e supervisionado identificado como problema em sistemas similares e garante resiliência operacional independentemente da conectividade. Terceiro, a implementação da automação por umidade do solo com controle autônomo no ESP32, em linha com as recomendações de Shahab et al. (2024), que destacam o potencial de redução do consumo hídrico de até 30% por essa abordagem.

Entre as limitações identificadas: dependência de Wi-Fi para comunicação com o Home Assistant, que pode ser obstáculo em propriedades rurais (CETIC.br, 2024); ausência de sensor de fluxo para medição precisa do volume irrigado, atualmente estimado por tempo e vazão nominal; escala laboratorial do protótipo, com parâmetros a recalibrar para ambientes agrícolas reais; e período de testes de 96 horas em ambiente controlado, insuficiente para validar a robustez do hardware contra as condições reais de campo (Morchid et al., 2024; Shahab et al., 2024).

De maneira geral, os resultados obtidos demonstram que o sistema desenvolvido atingiu o objetivo proposto ao integrar monitoramento ambiental, comunicação baseada em MQTT, supervisão por meio do Home Assistant e controle autônomo executado localmente pelo ESP32. Ainda que a validação tenha sido realizada em ambiente laboratorial controlado, os testes evidenciam a viabilidade técnica da arquitetura adotada e indicam seu potencial de aplicação em sistemas destinados à agricultura familiar, especialmente em cenários com limitações de conectividade.

6. Conclusão

Este trabalho apresentou o desenvolvimento de um sistema de monitoramento ambiental e automação de irrigação para ambientes agrícolas, baseado no microcontrolador ESP32, no protocolo MQTT e na plataforma Home Assistant, como uma prova de conceito tecnológica voltada à agricultura familiar em Rondônia. O protótipo desenvolvido integrou sensores ambientais, comunicação baseada em MQTT, supervisão por meio do Home Assistant e uma arquitetura de computação de borda, na qual o ESP32 executa localmente a lógica de controle da irrigação. Os testes realizados em ambiente controlado, durante 96 horas de operação contínua, demonstraram a estabilidade da comunicação entre os componentes, com aproximadamente 11.520 mensagens MQTT publicadas sem

interrupções, além do funcionamento adequado do monitoramento ambiental, da interface de supervisão e da lógica de controle autônoma implementada.

O principal diferencial técnico da solução é sua arquitetura de edge computing: o *firmware* embarcado no ESP32 executa autonomamente a lógica de irrigação, garantindo a continuidade do controle mesmo sem conectividade, enquanto o Home Assistant opera exclusivamente como camada de supervisão e controle manual. Essa separação hierárquica de responsabilidades elimina o conflito de controle presente em soluções similares, onde a coexistência de duas lógicas de automação sobre o mesmo atuador pode gerar inconsistências de estado. A robustez operacional independente de conectividade é especialmente relevante para o contexto da agricultura familiar em Rondônia, onde a infraestrutura digital é intermitente (CETIC.br, 2024).

As contribuições deste trabalho podem ser sintetizadas em três dimensões. Do ponto de vista técnico, demonstrou-se a viabilidade da integração entre ESP32, MQTT, Mosquitto e Home Assistant em arquitetura local sem dependência de nuvem, com o edge computing como diferencial de robustez. Do ponto de vista econômico, comprovou-se que é possível construir uma infraestrutura funcional de monitoramento e automação agrícola com custo de hardware inferior a R\$ 200,00 (duzentos reais) e uso exclusivo de ferramentas de código aberto, evidenciando potencial de democratização tecnológica na agricultura familiar (Araújo; Clementino, 2024). Do ponto de vista científico, o trabalho contribui para o corpo de literatura sobre aplicações de IoT na agricultura de pequena escala em contextos amazônicos, área carente de estudos que considerem as especificidades socioeconômicas e climáticas da região (Santos, 2021; Morchid et al., 2024).

A validação em escala real constitui a próxima etapa da pesquisa. Essa avaliação deverá utilizar uma bomba de maior porte, um sensor de fluxo para medir o volume irrigado e um período mínimo de três meses de operação em campo. Os resultados dessa etapa permitirão avaliar a aplicabilidade da solução na agricultura familiar amazônica.

Como trabalhos futuros propõem-se: (1) incorporação de sensor de fluxo para medição precisa do volume hídrico dispensado, eliminando a dependência do cálculo estimado por vazão nominal; (2) implementação de comunicação alternativa via LoRaWAN para zonas sem Wi-Fi (CETIC.br, 2024); (3) expansão com sensores de luminosidade, pH do solo e concentração de CO₂; (4) persistência local de dados

via cartão microSD no ESP32; e (5) validação do sistema em ambiente agrícola real por período mínimo de três meses (Shahab et al., 2024; Morchid et al., 2024).

REFERÊNCIAS

AOSONG ELECTRONICS. **DHT22 Digital-output Relative Humidity and Temperature Sensor/Module**. 2023.

ARAÚJO, Tatiane de Moura Fontes; CLEMENTINO, Valdner Daizio Ramos. Análise do uso de inovações tecnológicas pela agricultura familiar: uma revisão sistemática de literatura. **Revista de Estudos Interdisciplinares**, v. 6, n. 3, p. 1–25, 2024. DOI: 10.56579/rei.v6i3.993. Disponível em: <https://revistas.ceeinter.com.br/revistadeestudosinterdisciplinar/article/view/993>. Acesso em: 10 jul. 2026.

BANKS, Andrew; BRIGGS, Ed; BORGENDALE, Ken; GUPTA, Rahul. MQTT Version 5.0. **OASIS Open**, 2019. Disponível: <http://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html>. Acesso em: 8 jun. 2026.

CENTRO REGIONAL DE ESTUDOS PARA O DESENVOLVIMENTO DA SOCIEDADE DA INFORMAÇÃO (CETIC.br). **Pesquisa sobre o uso das tecnologias de informação e comunicação nas empresas brasileiras: TIC Empresas 2023**. São Paulo: NIC.br, 2024. Disponível em: <https://www.cetic.br/pt/publicacao/pesquisa-sobre-o-uso-das-tecnologias-de-informacao-e-comunicacao-nas-empresas-brasileiras-tic-empresas-2023/>. Acesso em: 8 jun. 2026.

ESPRESSIF SYSTEMS. **ESP32 Technical Reference Manual**. 2023. Disponível em: https://documentation.espressif.com/esp32_technical_reference_manual_en.pdf. Acesso em: 25 jun. 2026.

GUBBI, Jayavardhana; BUYYA, Rajkumar; MARUSIC, Slaven; PALANISWAMI, Marimuthu Internet of Things (IoT): a vision, architectural elements, and future directions. **Future Generation Computer Systems**, v. 29, n. 7, p. 1645–1660, 2013. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167739X13000241>. Acesso em: 03 jul. 2026

HOME ASSISTANT. **Home Assistant Documentation**. 2024. Disponível em: <https://www.home-assistant.io/docs/>. Acesso em: 13 mai. 2026.

HUNKELER, Urs; TRUONG, Hong Linh; STANFORD-CLARK, Andy. **MQTT-S: A publish-subscribe protocol for wireless sensor networks**. In: 3rd International Conference on Communication Systems Software and Middleware. [S.l.]: IEEE, 2008. p. 791–798. Disponível em: <http://dx.doi.org/10.1109/COMSWA.2008.4554519>. Acesso em: 03 jul. 2026.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA (IBGE). **Censo Agropecuário 2017**. Rio de Janeiro: IBGE, 2022. Disponível em: <https://www.ibge.gov.br/estatisticas/economicas/agricultura-e-pecuaria/21814-2017-censo-agropecuário.html>. Acesso em: 5 ago. 2026.

MAZER, S. **Agricultura familiar**. 2022.

MORCHID, Abdennabi et al. Applications of Internet of Things (IoT) and sensors technology to increase food security and agricultural sustainability: benefits and challenges. **Ain Shams Engineering Journal**, v. 15, n. 3, art. 102509, 2024. Disponível em: <https://doi.org/10.1016/j.asej.2023.102509>. Acesso em: 28 mai. 2026

OLIVEIRA, Victor Inácio de; SILVA, Isabela Melo da; JUNGER, Alex Paubel; LIMA, Bruno Luis Soares de; OLIVEIRA, Alex Lopes; JANES, Ricardo; YAMAMOTO, Rogerio Issamu. Sistema de controle e monitorização IoT de uma estufa. **FTT Journal of Engineering and Business**, v. 9, n. 1, p. 78–102, 2024. Disponível em: <https://saijournal.cefsa.org.br/index.php/FTT/article/view/534>. Acesso em: 23 jun. 2026.

SANTOS, Tiago Roberto Silva. **Agricultura familiar e agronegócio em Rondônia: uma distinção necessária**. Geographia Opportuno Tempore, v. 7, n. 1, p. 135–148, 2021. Disponível em: 10.5433/got.2021.v7.44817. Acesso em: 13 mai.2026.

SHAHAB, Hammad et al. **IoT-based agriculture management techniques for sustainable farming: a comprehensive review**. Computers and Electronics in Agriculture, v. 220, art. 108851, 2024. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0168169924002424>. Acesso em 15 jul.2026.

SILVA, Wesley Henrique Garcia e; SILVA, Ricardo Gilson da Costa. A agricultura familiar como protagonista na produção de alimentos em Rondônia. **Revista Geonorte**, Manaus, v. 15, n. 47, p. 98–113, 2024. DOI: 10.21170/geonorte.2024.V.15.N.47.98.113. Disponível em: <https://periodicos.ufam.edu.br/index.php/revista-geonorte/article/view/13503>. Acesso em: 05 jun.2026.

STROPARO, Telma Regina. **Digital transformation in agriculture: impacts of the Internet of Things (IoT) on production efficiency and sustainability**. Lumen et Virtus, v. 15, n. 38, p. 1573–1581, 2024. Disponível em: <https://doi.org/10.5281/zenodo.13332140>. Acesso em:30 jun. 2026.

APÊNDICE

Apêndice A – Código-Fonte do Firmware ESP32

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include "DHT.h"
4
5 #define DHTPIN 4
6 #define DHTTYPE DHT22
7 #define SOIL_PIN 34
8 #define RELE_PIN 2
9
10 const char* ssid = "";
11 const char* password = "";
12 const char* mqtt_server = "ENDERECO_DO_BROKER";
13 const char* mqtt_user = "USUARIO_MQTT";
14 const char* mqtt_pass = "SENHA_MQTT";
15
16 const char* TOPIC_TEMP = "estufa/esp32_01/sensor/temperatura";
17 const char* TOPIC_UMIDADE_AR = "estufa/esp32_01/sensor/umidade_ar";
18 const char* TOPIC_UMIDADE_SOLO =
19 "estufa/esp32_01/sensor/umidade_solo";
20 const char* TOPIC_SOLO_BRUTO = "estufa/esp32_01/sensor/solo_bruto";
21 const char* TOPIC_SOLO_TENSAO =
22 "estufa/esp32_01/sensor/solo_tensao";
23 const char* TOPIC_BOMBA_SET = "estufa/esp32_01/atuador/bomba/set";
24 const char* TOPIC_BOMBA_STATE =
25 "estufa/esp32_01/atuador/bomba/state";
26 const char* TOPIC_TEMPO_BOMBA =
27 "estufa/esp32_01/sensor/tempo_bomba_hoje";
28 const char* TOPIC_LITROS = "estufa/esp32_01/sensor/litros";
29 const char* TOPIC_STATUS = "estufa/esp32_01/status";
30
31 int soloSeco = 3300, soloMolhado = 1300;
32 float limiteLigaSolo = 30.0, limiteDesligaSolo = 70.0;
33 float vazao = 1.5;
```

```

31 bool releEstado = false;
32 bool bombaAutomatica = false;
33
34 unsigned long inicioLigado = 0;
35 unsigned long ultimaLeitura = 0;
36 unsigned long inicioSoloSeco = 0;
37 unsigned long ultimoComandoHomeAssistant = 0;
38 unsigned long ultimaTentativaWiFi = 0;
39 unsigned long ultimaTentativaMQTT = 0;
40
41 const unsigned long intervaloLeitura = 30UL * 1000UL;
42 const unsigned long intervaloTentativaWiFi = 10UL * 1000UL;
43 const unsigned long intervaloTentativaMQTT = 10UL * 1000UL;
44 const unsigned long tempoSoloSecoParaLigar = 3UL * 60UL * 1000UL;
45 const unsigned long tempoEsperaHomeAssistant = 3UL * 60UL * 1000UL;
46 const unsigned long tempoMaximoBombaLigada = 10UL * 60UL * 1000UL;
47
48 WiFiClient espClient;
49 PubSubClient client(espClient);
50 DHT dht(DHTPIN, DHTTYPE);
51
52 bool mqttConectado() {
53     return WiFi.status() == WL_CONNECTED && client.connected();
54 }
55
56 void publicarMQTT(const char* topico, const char* mensagem) {
57     if (mqttConectado()) client.publish(topico, mensagem, true);
58 }
59
60 void iniciarWiFi() {
61     WiFi.mode(WIFI_STA);
62     WiFi.persistent(false);
63     WiFi.begin(ssid, password);
64     ultimaTentativaWiFi = millis();
65 }
66

```

```

67 void verificarWiFi(unsigned long agora) {
68   if (WiFi.status() == WL_CONNECTED) return;
69   if (agora - ultimaTentativaWiFi < intervaloTentativaWiFi) return;
70
71   ultimaTentativaWiFi = agora;
72   WiFi.disconnect();
73   WiFi.begin(ssid, password);
74 }
75
76 void verificarMQTT(unsigned long agora) {
77   if (WiFi.status() != WL_CONNECTED || client.connected()) return;
78
79   if (ultimaTentativaMQTT != 0 &&
80       agora - ultimaTentativaMQTT < intervaloTentativaMQTT) return;
81
82   ultimaTentativaMQTT = agora;
83
84   if (client.connect("ESP32Client", mqtt_user, mqtt_pass,
85                       TOPIC_STATUS, 0, true, "offline")) {
86     client.subscribe(TOPIC_BOMBA_SET);
87     client.publish(TOPIC_STATUS, "online", true);
88     client.publish(TOPIC_BOMBA_STATE,
89                   releEstado ? "ON" : "OFF", true);
90   }
91 }
92
93 void ligarBomba(bool automatico) {
94   if (releEstado) return;
95
96   digitalWrite(RELE_PIN, LOW);
97   releEstado = true;
98   bombaAutomatica = automatico;
99   inicioLigado = millis();
100  publicarMQTT(TOPIC_BOMBA_STATE, "ON");
101 }
102

```

```

103 void desligarBomba() {
104 if (!releEstado) return;
105
106 digitalWrite(RELE_PIN, HIGH);
107 releEstado = false;
108
109 float tempo = (millis() - inicioLigado) / 1000.0;
110 float litros = (tempo / 60.0) * vazao;
111 char tempoStr[12], litrosStr[12];
112
113 dtostrf(tempo, 1, 2, tempoStr);
114 dtostrf(litros, 1, 2, litrosStr);
115
116 publicarMQTT(TOPIC_BOMBA_STATE, "OFF");
117 publicarMQTT(TOPIC_TEMPO_BOMBA, tempoStr);
118 publicarMQTT(TOPIC_LITROS, litrosStr);
119
120 bombaAutomatica = false;
121 }
122
123 void callback(char* topic, byte* payload, unsigned int length) {
124 String mensagem = "";
125
126 for (unsigned int i = 0; i < length; i++)
127 mensagem += (char)payload[i];
128
129 mensagem.trim();
130 mensagem.toUpperCase();
131
132 if (String(topic) == TOPIC_BOMBA_SET) {
133 ultimoComandoHomeAssistant = millis();
134
135 if (mensagem == "ON") ligarBomba(false);
136 else if (mensagem == "OFF") desligarBomba();
137 }
138 }

```

```

139
140 int lerSoloBruto() {
141     long soma = 0;
142
143     for (int i = 0; i < 20; i++) {
144         soma += analogRead(SOIL_PIN);
145         delay(10);
146     }
147
148     return soma / 20;
149 }
150
151 float calcularUmidadeSolo(int soloBruto) {
152     float percentual = ((float)(soloSeco - soloBruto) /
153     (soloSeco - soloMolhado)) * 100.0;
154
155     return constrain(percentual, 0, 100);
156 }
157
158 void realizarLeituraEControle(unsigned long agora) {
159     float temperatura = dht.readTemperature();
160     float umidadeAr = dht.readHumidity();
161
162     if (!isnan(temperatura) && !isnan(umidadeAr)) {
163         char temperaturaStr[12], umidadeArStr[12];
164
165         dtostrf(temperatura, 1, 2, temperaturaStr);
166         dtostrf(umidadeAr, 1, 2, umidadeArStr);
167
168         publicarMQTT(TOPIC_TEMP, temperaturaStr);
169         publicarMQTT(TOPIC_UMIDADE_AR, umidadeArStr);
170     }
171
172     int soloBruto = lerSoloBruto();
173     float soloPercent = calcularUmidadeSolo(soloBruto);
174     float tensaoSolo = soloBruto * (3.3 / 4095.0);

```

```

175
176 char soloStr[12], soloBrutoStr[12], tensaoStr[12];
177
178 dtostrf(soloPercent, 1, 2, soloStr);
179 dtostrf(soloBruto, 1, 0, soloBrutoStr);
180 dtostrf(tensaoSolo, 1, 2, tensaoStr);
181
182 publicarMQTT(TOPIC_UMIDADE_SOLO, soloStr);
183 publicarMQTT(TOPIC_SOLO_BRUTO, soloBrutoStr);
184 publicarMQTT(TOPIC_SOLO_TENSAO, tensaoStr);
185
186 bool comandoHARecente =
187 ultimoComandoHomeAssistant > 0 &&
188 agora - ultimoComandoHomeAssistant <
189 tempoEsperaHomeAssistant;
190
191 if (soloPercent < limiteLigaSolo) {
192 if (inicioSoloSeco == 0) inicioSoloSeco = agora;
193
194 if (!releEstado &&
195 agora - inicioSoloSeco >= tempoSoloSecoParaLigar &&
196 !comandoHARecente)
197 ligarBomba(true);
198 } else {
199 inicioSoloSeco = 0;
200 }
201
202 if (releEstado && bombaAutomatica &&
203 soloPercent >= limiteDesligaSolo)
204 desligarBomba();
205 }
206
207 void setup() {
208 dht.begin();
209
210 pinMode(RELE_PIN, OUTPUT);

```



```

211 digitalWrite(RELE_PIN, HIGH);
212
213 analogReadResolution(12);
214 analogSetPinAttenuation(SOIL_PIN, ADC_11db);
215
216 client.setServer(mqtt_server, 1883);
217 client.setCallback(callback);
218 client.setSocketTimeout(2);
219 client.setKeepAlive(30);
220
221 iniciarWiFi();
222 }
223
224 void loop() {
225   unsigned long agora = millis();
226
227   if (releEstado &&
228       agora - inicioLigado >= tempoMaximoBombaLigada)
229     desligarBomba();
230
231   if (ultimaLeitura == 0 ||
232       agora - ultimaLeitura >= intervaloLeitura) {
233     ultimaLeitura = agora;
234     realizarLeituraEControle(agora);
235   }
236
237   verificarWiFi(agora);
238   verificarMQTT(agora);
239
240   if (client.connected()) client.loop();
241 }

```

Fonte: Elaborado pelos autores.