

UNESP - Universidade Estadual Paulista "Júlio de Mesquita Filho"

FEIS - Faculdade de Engenharia de Ilha Solteira

DEE - Departamento de Engenharia Elétrica

LARISSA CASSIANO GALAN

**CONTROLE E MONITORAMENTO EM TEMPO REAL DE UMA ESTUFA
PARA CULTIVO DE HORTALIÇAS**

Ilha Solteira

2025

LARISSA CASSIANO GALAN

**CONTROLE E MONITORAMENTO EM TEMPO REAL DE UMA ESTUFA
PARA CULTIVO DE HORTALIÇAS**

Trabalho de Graduação apresentado à
Faculdade de Engenharia de Ilha
Solteira – UNESP como parte dos
requisitos para obtenção do título de
Engenheiro Eletricista.

Nome do orientador

**Profa Dra. Suely Cunha Amaro
Mantovani**

Ilha Solteira
2025

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

- G147c Galan, Larissa Cassiano.
Controle e monitoramento em tempo real de uma estufa para cultivo de hortaliças / Larissa Cassiano Galan. -- Ilha Solteira: [s.n.], 2025
81 f. : il.
- Trabalho de conclusão de curso (Graduação em Engenharia Elétrica) - Universidade Estadual Paulista (UNESP), Faculdade de Engenharia, Ilha Solteira, 2025
- Orientadora: Suely Cunha Amaro Mantovani
- Inclui bibliografia
1. Esp32. 2. Controle. 3. Monitoramento. 4. Sistemas microcontrolados. 5. Cultivo de hortaliças em estufa automatizado. 6. Node-red.

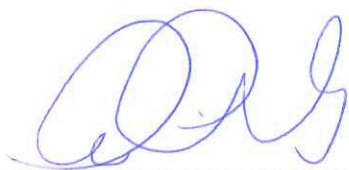
ATA DE DEFESA DE TRABALHO DE GRADUAÇÃO

Aos seis dias do mês de junho do ano de dois mil e vinte e cinco, a discente **Larissa Cassiano Galan**, matriculada sob o nº **172055318**, tendo como banca examinadora a sua orientadora, a Profa. Dra. Suely Cunha Amaro Mantovani, o Prof. Dr. Carlos Antônio Alves e o Doutorando Pablo Maurício Portilla Hernandez, apresentou o Trabalho de Graduação intitulado "**CONTROLE E MONITORAMENTO EM TEMPO REAL DE UMA ESTUFA PARA CULTIVO DE HORTALIÇAS**", obtendo a nota 10 (dez) e conceito Aprovado.

Suely Cunha Amaro Mantovani

Profa. Dra. Suely Cunha Amaro Mantovani

- Orientadora -



Prof. Dr. Prof. Dr. Carlos Antônio Alves

- Membro da Banca -

Larissa C. Galan

Larissa Cassiano Galan

- Discente -



Doutorando Pablo Maurício Portilla
Hernandez

- Membro da Banca -

AGRADECIMENTOS

Agradeço primeiramente aos meus pais, Damaris e Laércio, pelo amor incondicional, pelo apoio em todos os momentos e por sempre acreditarem no meu potencial, mesmo quando eu não acreditei.

À minha orientadora, Profa Suely Cunha Amaro Mantovani pelo acompanhamento, paciência, dedicação e pelos valiosos conselhos ao longo da minha passagem pela faculdade.

E agradeço também aos meus amigos, Vitor, Isabella, Julia, Karen, Brenda e Noah, pelo incentivo constante, pelas conversas que aliviaram os momentos de estresse, por cada noite sem dormir estudando juntos e por tantos outros motivos. Vocês foram a minha família longe de casa.

Agradeço a todos os professores e equipe técnica do curso que deram a sua contribuição para minha formação como Engenheira Eletricista.

“O sucesso é ir de fracasso em fracasso sem perder o entusiasmo.” — *Winston Churchill*

RESUMO

O cultivo abrigado em estufas é importante para o crescimento e desenvolvimento adequado de hortaliças em geral, ao conservar suas propriedades de acordo com os parâmetros de qualidade. Para isso, as estufas necessitam de um monitoramento e controle de alguns dos principais parâmetros como a temperatura, umidade do solo e ar, e a luminosidade para uma adequada climatização do ambiente. O sistema projetado deve automatizar o controle dessas variáveis de forma que a cultura presente na estufa seja beneficiada, podendo ser aplicado a estufas em pequenas propriedades e que exigem um baixo custo. Os parâmetros utilizados foram escolhidos para atender a maioria das hortaliças, mas podem ser adicionados outros sensores, dependendo das necessidades da hortaliça específica. O sistema foi desenvolvido em laboratório utilizando o conceito de Internet of Things (IoT) e a plataforma ESP32 que possui comunicação *wifi*, permitindo o interfaceamento com os sensores e atuadores, via internet. Com isso proporciona vantagens como acesso aos dados da plantação, controle e monitoramento em tempo real dos parâmetros da estufa, remotamente e a sua visualização pela tela de um computador, usando a interface node-red que proporciona uma visualização gráfica das quantidades mensuradas e a visualização do estado do solo, umidade e temperatura, mostrando botões de estado e acionamento. . O sistema projetado obteve bons resultados no que concerne aos seus objetivos, pois tem-se via internet, a visualização dos dados na tela de um computador, possibilitando que os agricultores tenham acesso aos dados com boa precisão devido a qualidade atual dos sensores, e consigam, por intermédio da tecnologia, obter um ganho na produtividade, a custo reduzido.

Palavras-chave: ESP32; Controle; Monitoramento; Sistemas microcontrolados; Cultivo de hortaliças em estufa automatizado; Node-red.

ABSTRACT

Protected cultivation in greenhouses is important for the proper growth and development of vegetables in general, as it helps preserve their properties according to quality parameters. For this purpose, greenhouses require monitoring and control of key parameters such as temperature, soil and air humidity, and light intensity to ensure proper climate regulation. The designed system aims to automate the control of these variables so that the crop inside the greenhouse is benefited. It is suitable for use in small-scale greenhouses that require low-cost solutions. The parameters used were selected to meet the needs of most vegetables, but additional sensors can be included depending on the specific crop requirements. The system was developed in a laboratory setting using the Internet of Things (IoT) concept and the ESP32 platform, which has Wi-Fi communication capabilities, allowing it to interface with sensors and actuators via the internet. This provides advantages such as access to crop data, real-time remote monitoring and control of greenhouse parameters, and visualization through a computer screen using the Node-RED interface, which offers a graphical display of the measured quantities and shows the state of the soil, humidity, and temperature with control buttons. The system achieved good results in relation to its objectives, as it enables the visualization of data through the internet on a computer screen, allowing farmers to access precise data due to the current quality of sensors and through technology, gain productivity at a reduced cost.

Keywords: ESP32; Control; Monitoring; Microcontroller systems; Automated greenhouse vegetable cultivation; Node-RED.

LISTA DE FIGURAS

Figura 1– Modelo de estufa tipo túnel	19
Figura 2 – Arquitetura IoT para uma residência	21
Figura 3 – Evolução da Agricultura	22
Figura 4 – Arquitetura de um sistema embarcado	24
Figura 5 – Microprocessador x Microcontrolador.	25
Figura 6 – Pinagem do ESP32.	27
Figura 7 – Exemplo de utilização do protocolo MQTT.	28
Figura 8 – Visualização de dados na plataforma do Thingspeak	29
Figura 9 – Interface do Node-red.	30
Figura 10 – Nós da paleta dashboard 2.0.	31
Figura 11 – (a) Sensor LDR. (b) Simbologia	32
Figura 12 – Foto resistência do sensor LDR x iluminância	33
Figura 13 – (a) Pinagem (b) Sensores internos	34
Figura 14–Sensor de umidade do solo	35
Figura 15 – Identificação dos pinos do sensor de umidade do solo, módulo HL-69.	36
Figura 16 – Módulo relé 5V , 10A de um canal	38
Figura 17 - Circuito interno de um relé básico	39
Figura 18 – Módulo relé de dois canais 5V com optoacoplador	40
Figura 19 – Circuito esquemático para o módulo relé de 2 canais com optoacoplador	41
Figura 20 – Mini bomba d'água Jt100.	42
Figura 21 – Mini aspersor	44
Figura 22 - Módulo Fonte 12V ajustável para protoboard	45
Figura 23 – Circuito esquemático do Módulo Fonte 12V	46
Figura 24 – Fonte chaveada Colmeia 12V,5A, 60W .	47
Figura 25 – Diagrama esquemático ilustrativo do projeto	50
Figura 26 – Montagem do projeto	52
Figura 27 – Circuito elétrico do projeto	53

Figura 28 – Fluxograma principal com a programação do sistema da estufa	54
Figura 29 – Trecho “void setup”	55
Figura 30 – Fluxograma da subrotina setupWifi()	56
Figura 31 – Fluxogramas: a) Subrotina setupMQTT.b) Subrotina Leiturasensores	57
Figura 32 – Fluxograma da subrotina ControleAtuadores	58
Figura 33 – Fluxo de configuração do Node-red.	59
Figura 34 – Programa criado para o nó de conversão.	60
Figura 35 – Programa criado para o nó de timer	61
Figura 36 – a)Nó de Atuador. b) Nó de dashboard.	62
Figura 37 – Esquemático para os testes com o LDR e o ESP32.	63
Figura 38 – Gráfico tensão de saída x Resistência do LDR	64
Figura 39 – Circuito de teste para o DHT22.	65
Figura 40- Resultados dos testes com o DHT22.	65
Figura 41: Circuito para o teste do HL69.	66
Figura 42 – Programa para o sensor de umidade de solo	67
Figura 43 – Teste de vazão da bomba d’agua.	67
Figura 44 – Sistema de monitoramento e controle de uma estufa.	69
Figura 45 - Interface da <i>dashboard</i> : a) Página inteira. b) Gráfico de Umidade do ar. c) Gráfico de luminosidade. d) Gráfico de Temperatura. e) Gráfico de Umidade do solo.	70
Figura 46 - Interface da <i>dashboard</i> : a) Gráfico da Umidade do ar; b) Gráfico de luminosidade. c) Gráfico da Temperatura. d) Gráfico de Umidade do solo.	71

LISTA DE QUADROS E TABELAS

Quadro 1- Hortaliças de crescimento rápido - plantio hidropônico	20
Quadro 2 – Arduino Uno R3 e o ESP32.	26
Quadro 3 – Características técnicas do DHT22.	34
Quadro 4 – Características técnicas do sensor de umidade do solo, HL- 69.	36
Quadro 5–Resumo das características do relé	38
Quadro 6 – Módulo Relé 2 Canais 5V com optoacoplador	40
Quadro 7– Dados técnicos da mini bomba d'água	43
Quadro 8 – Características técnicas da fonte de alimentação MB -102	47
Quadro 9 – Características técnicas da fonte de alimentação chaveada.	48
Quadro 10 – Resultados das medidas e os tempos na bomba d'agua.	68
 Tabela 1 — Resumo de erros de calibração do sensor	 37

SUMÁRIO

1. INTRODUÇÃO	14
1.1 Objetivo	15
1.2 Organização do texto	15
2. REVISÃO DE LITERATURA	16
3. AGRICULTURA E A INTERNET DAS COISAS	18
3.1 Cultura em Estufas	18
3.2 Internet das Coisas	20
3.3 IoT na Agricultura	21
4. FUNDAMENTAÇÃO TEÓRICA	24
4.1 Sistemas Embarcados	24
4.2 Placa de desenvolvimento	25
4.3 Protocolo de Comunicação MQTT	27
4.4 Plataforma de Visualização e Análise de Dados	28
4.4.1 Thingspeak	28
4.4.2 Node-Red	29
4.5 Sensores	31
4.5.1 Sensor LDR	32
4.5.2 Sensor DHT22	33
4.5.3 Sensor HL- 69	34
4.6 Atuadores	37
4.6.1 Relé	37
4.6.2 Bomba d'água e aspersor	42
4.7. Fonte de alimentação	44
4.7.1 Módulo MB-102	45

4.7.2 Fonte Chaveada 12V (Tipo Colméia)	47
5. DESENVOLVIMENTO DO PROJETO DE CONTROLE E MONITORAMENTO DA ESTUFA	50
5.1 Diagrama esquemático ilustrativo do projeto	50
5.2 Montagem do projeto	52
5.3 Programação	53
5.4 Plataforma do Node-Red	59
6. RESULTADOS E DISCUSSÕES	63
6.1 Testes dos sensores	63
6.1.1 Teste para o sensor LDR	63
6.1.3 Teste para o sensor de umidade do solo HL- 69	66
6.2 Teste final para o hardware completo	68
7. CONCLUSÃO	74
REFERÊNCIAS BIBLIOGRÁFICAS	76
APÊNDICE A - Valores gastos no projeto em reais	80
APÊNDICE B - Resumo apresentado no XXXIII CIC da UNESP	81
APÊNDICE C - Programas em linguagem C++	81

1. INTRODUÇÃO

Devido ao clima, o cultivo de diversos tipos de hortaliças tem dados bons resultados no Brasil, uma vez que alguns gêneros são típicos de temperatura tropical como pimenta, pepino, coentro e feijão-vagem entre outros, enquanto que a alface, rúcula, ervilha, morango e cebola são as de temperatura amena. Por outro lado, as muitas variações climáticas que acontecem no país dificultam o cultivo de hortaliças a céu aberto, por causa das altas temperaturas e amplitudes térmicas, frio extremo, temporais de granizo, períodos de seca, grande quantidade de pragas e doenças nas lavouras (FERNANDES, 2017).

Outro problema está relacionado com os recursos utilizados no cuidado da plantação e o controle, tais como a água, a energia elétrica e gastos com a mão de obra. Como alternativa a esses fatores, os agricultores tem adotado o cultivo em ambientes protegidos como as estufas que são estruturas fechadas, permitem o controle de variáveis como temperatura, umidade e luminosidade, controle eficiente de pragas e doenças, podendo se adaptar a diversos tipos de cultivo. Ainda permitem o controle adequado durante todas as estações do ano, reduzindo a mão de obra, os riscos e, portanto a perda de recursos (BRITO; GOTTSCHALL; CUNHA; SILVA, 2018)

A utilização de estufas tornou-se fundamental para o abastecimento seguro dos produtos alimentares no mundo. Atualmente, encontram-se nas propriedades rurais estufas destinadas à produção de diversos tipos de hortaliças, onde são utilizados procedimentos de irrigação e umidificação, mas que exigem uma grande atenção e mão de obra para manter os níveis adequados, pois são difíceis o acesso há um registro e controle dos dados adequados. Conforme a literatura, o uso de sistemas automatizados em estufas vem crescendo ano após ano, pois os produtores buscam cada vez mais tecnologias que possam auxiliar no dia a dia (BRITO; GOTTSCHALL; CUNHA; SILVA, 2018; FERNANDES, 2017; SANTOS, 2014).

Processos de automação garantem um controle total do ambiente, proporcionando maior desenvolvimento da produtividade e qualidade dos produtos com maior eficiência e uso coerente dos recursos. O cultivo abrigado em estufas é importante para o crescimento e desenvolvimento adequado de hortaliças em geral, ao conservar suas propriedades de acordo com os parâmetros de qualidade, garantir a padronização e a produtividade, além disso, viabiliza a produção fora de época. Para isso as estufas

podem ter sua eficiência otimizada por meio de um monitoramento e controle de temperatura, umidade do solo e ar, e a luminosidade para uma adequada climatização do ambiente.

1.1 Objetivo

O objetivo do trabalho é o desenvolvimento de um sistema de controle e monitoramento de uma estufa para o cultivo de hortaliças. Para isso, são usados sensores de temperatura, luminosidade, umidade do ar e do solo, fatores importantes no manejo de plantas. Como plataforma usa-se o ESP32 e a visualização das medidas na tela de um computador usando o ambiente Node-red. A partir dessas medidas faz-se a análise das variáveis climáticas para a tomada de decisão visando o melhoramento das condições dentro da estufa por meio de atuadores como, ventiladores, lâmpadas e uma bomba d'água com aspersor.

1.2 Organização do texto

Além desta introdução, no capítulo 2 é apresentada a revisão de literatura com a descrição de alguns trabalhos relevantes que embasaram o desenvolvimento desta pesquisa.

Apresenta-se no capítulo 3 uma introdução à agricultura e ao uso da Internet das Coisas no cultivo com estufas.

No Capítulo 4 tem-se a fundamentação teórica, descrevendo os conceitos e componentes utilizados, plataformas de controle, sensores, atuadores e fontes.

O desenvolvimento do sistema de controle e monitoramento da estufa é detalhado no capítulo 5, incluindo a metodologia, o circuito criado, o fluxograma do programa e a interface de visualização na tela de um computador.

No capítulo 6 são apresentados os testes realizados e os resultados obtidos com os sensores e trechos dos programas de teste, com a visualização das medidas de temperatura, umidade do ar e solo e luminosidade.

No capítulo 7 tem-se a conclusão do trabalho com as principais questões abordadas durante o desenvolvimento, seguido das referências bibliográficas e os Apêndices.

2. REVISÃO DE LITERATURA

Neste capítulo são apresentados alguns trabalhos relevantes de interesse deste projeto de pesquisa, tais como o controle dos parâmetros de uma estufa e monitoramento visando obter boa produtividade e com baixo custo, utilizando plataformas de controle amigáveis, modernas, encontradas no mercado de dispositivos eletrônicos, como o ESP32, Arduino e Raspberry Pi.

Fernandes, Preuss e da Silva (2017) descrevem em seu trabalho de conclusão de curso um protótipo de baixo custo para o controle de estufas para hortaliças. Propuseram uma solução automatizada para o controle da temperatura, luminosidade e umidade do solo em estufas visando aplicar em pequenas e médias propriedades de produtores rurais de hortaliças. Com isso, obtiveram um protótipo automatizado para estufas, baseado na plataforma Arduino UNO R3, operado por um aplicativo instalado em um smartphone com sistema operacional Android, proporcionando o acesso aos dados da plantação, controle e monitoramento da estufa em tempo real.

É apresentado em Longo e Abreu (2021) no trabalho de conclusão de curso de suas autorias, a descrição de uma estufa com controle automático para o cultivo de hortaliças. Diante dos crescentes avanços tecnológicos, o uso de estufas tem sido a solução para aumentar a produção e maximizar os lucros e obter um maior controle e produtividade. No mercado de produtos agrícolas encontram-se sistemas para estufas automatizados, mas a um custo elevado e não dispõe de uma plataforma de monitoramento móvel, por isso, a proposta de um projeto que atenda o cultivo de hortaliças por meio de uma estufa automatizada com baixo custo e que disponibilize as informações em tempo real usando um smartphone, criando um aplicativo móvel que seja intuitivo para que os dados possam ser monitorados para aplicação em estufas residenciais e de pequenos produtores. O protótipo para automação da estufa faz a leitura dos sensores, de temperatura do ambiente, umidade do solo e ar, raios UV e vazão, controle de atuadores e monitoramento de dados via aplicativo. Utiliza um microcontrolador PIC18F4520 e comunicação wifi usando o microchip ESP8266 e linguagem de programação C. O sistema se comunica com o servidor que utiliza a NodeJS (*JavaScript Runtime Environment*) desenvolvida em TypeScript que lê e interpreta os dados captados pela placa com os sensores. Os dados são monitorados pelo usuário por meio do aplicativo, que consegue ter o controle total da estufa, tanto de modo automático como manual de uma forma segura e simples. Com base nos

resultados obtidos concluíram que o sistema da estufa obedeceu aos comandos e parâmetros estabelecidos, com precisão e monitoramento dos dados em tempo real.

Brito *et al* (2018) descrevem em seu artigo a proposta de um protótipo de monitoramento e controle dos parâmetros de temperatura, umidade e luminosidade eficientes para a climatização do ambiente de cultivo abrigado de tomates. O cultivo abrigado do tomate em estufas é importante para o crescimento e desenvolvimento adequado da planta, conservando suas propriedades, conforme os parâmetros de qualidade, uma vez que o tomateiro é considerado a segunda hortaliça em importância econômica no Brasil e também uma das mais importantes no mundo. Para isso, a estufa foi monitorada e controlados os seus parâmetros, e a correta climatização do ambiente, com as temperaturas na faixa de 16°C a 29°C e o controle de umidade e luminosidade de acordo com os valores encontrados nas referências consultadas sobre o cultivo de tomateiros. O sistema microcontrolado projetado utiliza o Arduino UNO R3 que permite o interfaceamento com os sensores e atuadores necessários para o desenvolvimento deste protótipo.

Em Santos (2014), preocupada com a agricultura familiar que é uma das principais fontes de produção de alimentos do país, gerando renda e trabalho, mas com pouco investimento em infraestrutura, propôs em suas pesquisas o desenvolvimento de um sistema de irrigação automatizado, aplicado à cultura de morangos, a custos acessíveis, visando os pequenos agricultores. Para isso, utiliza sensores de temperatura, umidade do solo e sensor de luminosidade, prevendo o cultivo de morango, dados que alimentam o microcontrolador Arduino Nano que realiza o controle e o acionamento de uma bomba de irrigação, utilizando a radiofrequência como meio de comunicação. Outra importante contribuição deste trabalho, além da automação e o auxílio aos pequenos agricultores na diminuição da mão de obra e consumo de água, é a preocupação com o meio ambiente.

Concluindo, dos estudos destes trabalhos pôde-se obter os conhecimentos necessários que influenciaram nas escolhas de plataformas de controle e monitoramento, e principalmente, nas escolhas dos sensores que traduzem as condições do meio ambiente e a metodologia para automação e visualização dos dados.

3. AGRICULTURA E A INTERNET DAS COISAS

Neste capítulo faz-se uma breve apresentação dos principais conceitos que envolvem a Agricultura, o cultivo em estufas com Internet of Things (IoT).

3.1 Cultura em Estufas

Com o crescimento da população mundial e uma maior demanda por alimentos, há a necessidade de um cultivo mais eficiente visando aumentar a produção. Outro fator importante é o crescente avanço tecnológico motivando o seu uso na irrigação, em estufas, etc.

A produção agrícola é influenciada pelo clima, uma vez que no verão, chuvas em demasia danificam as plantas e criam condições favoráveis para o aparecimento de doenças, prejudicando a produção, como consequência aumentam os preços dos produtos hortícolas pela falta de oferta para o abastecimento. Enquanto que no inverno o frio e os ventos, prolongam o ciclo dessas culturas, quebrando a produção, causando grandes prejuízos econômicos para os agricultores (BRITO *et al*, 2018). Para solucionar esse problema de forma a manter constante a produção hortícola em todas as épocas do ano, são utilizadas estufas, consideradas importantes para a produção de algumas hortaliças.

O uso de estufas tem sido a solução para aumentar a produção e maximizar os lucros e obter mais controle e produtividade. Soluções de estufas automatizadas são encontradas no mercado de produtos agrícolas, mas por um custo elevado. Desta forma, projetos de pesquisas que envolvem o controle e a automação das variáveis que impactam o ambiente e disponibilizam as informações em tempo real e com custo reduzido, são sempre bem-vindos.

A técnica agrícola que consiste em cultivar plantas em ambientes controlados, como estufas ou viveiros, para proteger os cultivos de fatores climáticos adversos (tempestades de granizo, chuvas fortes e geadas) e melhorar a qualidade da produção é também chamada de cultivo protegido. O cultivo protegido permite controlar elementos como temperatura, umidade, radiação solar e vento, proporcionando condições ideais para o desenvolvimento das plantas. Protege também as plantas contra pássaros e insetos e conserva a umidade do solo. Com isso, a qualidade e a produtividade dos produtos aumentam, e os vegetais tem um melhor valor de mercado. Outra

vantagem é a possibilidade dos produtores poderem oferecer ao mercado, os itens na entressafra, época em que o desenvolvimento dos cultivos ao ar livre é prejudicado por causa das diversas variações do clima em algumas regiões do Brasil (LONGO; ABREU, 2021). Para ilustrar mostra-se na Figura 1, um modelo de estufa tipo túnel, mais comum, usado pelos agricultores.

Figura 1– Modelo de estufa tipo túnel



Fonte: Longo e Abreu (2021).

Uma agricultura mais rentável e com maior produtividade é devida a grande contribuição dos avanços tecnológicos, maquinários mais eficazes, estudos de solos, colaborando com o aumento do volume de produção e reduzindo os custos.

A automação faz parte destes avanços e tem grande importância dentro da agricultura. De acordo com a Embrapa (2017), a automação agropecuária é feita de forma que os processos operacionais de produção sejam monitorados, controlados e executados por intermédio de dispositivos mecânicos, eletrônicos ou computacionais, visando ampliar a produtividade, otimizando o tempo de cultivo, uso racional dos recursos, auxiliando o trabalhador do campo (HEXAGON, 2021).

As estufas são estruturas geralmente metálicas cobertas com plástico filme transparente resistente e flexível, inicialmente muito utilizadas em regiões frias para proteger as plantações das amplitudes climáticas. Atualmente, suas funções não se resumem a isso, sua cobertura protege as culturas não somente das intempéries climáticas, mas proporciona um melhor uso dos recursos de água, nutrientes e fertilizantes e o controle da temperatura, que permitem o fornecimento contínuo de produtos ao mercado, pois é possível cultivar o produto ao longo de todo ano e não

apenas em uma estação específica. A cobertura também ajuda a reduzir a entrada de pragas (ZANNATA, 2020).

As hortaliças são muito beneficiadas pelo plantio em ambientes controlados, devido à estufa que proporciona um ambiente próximo do ideal durante todo o ano, conservando-as com um bom aspecto, mesmo em climas mais frios como o sul do Brasil, protegendo-as das geadas.

A hidroponia também é uma forma de cultivo protegido feito em água. Nesta, o solo ou substrato é inteiramente substituído por uma solução nutritiva contendo todos os elementos minerais essenciais para as plantas (FURLANI *et al.*, 2009). São mostradas no Quadro 1, as espécies de hortaliças de pequeno porte e crescimento rápido, mais adequado para o plantio hidropônico.

Quadro 1- Hortaliças de crescimento rápido - plantio hidropônico

Hortaliças folhosas	alface, agrião, rúcula e couve
Hortaliças frutos	tomate, pimentão, abobrinha e pepino
Hortaliças condimentares	cebolinha, coentro, manjeriço, mostarda e salsa
Medicinais	alecrim, alfavaca, bálsamo, hortelã, melissa
Ornamentais	antúrio, crisântemo, Lisianthus, hortênsia
Frutífera	melão, morango e melancia-baby
Forrageiras	milho, sorgo, cevada e alfafa

Fonte: SaberHortifruti (2022).

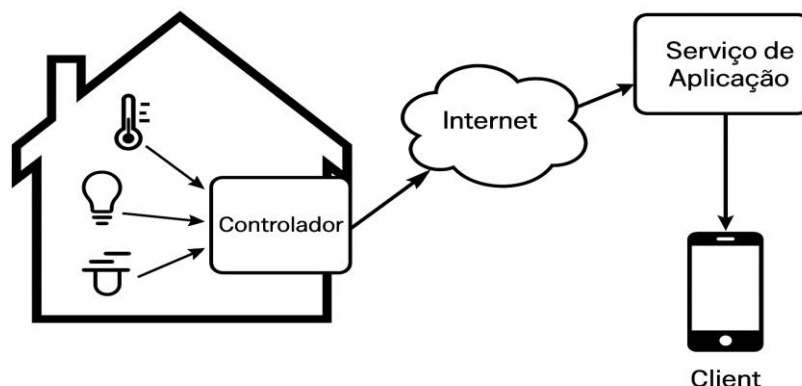
3.2 Internet das Coisas

O termo “Internet das Coisas” (IoT – *Internet of Things*) foi criado por Kevin Ashton em uma apresentação na *Proctor & Gamble* (P&G), empresa em que trabalhava em 1999, quando relatou que objetos físicos poderiam se conectar a objetos virtuais por meio da internet. Desde então, a IoT vem transformando como as pessoas vivem e trabalham, com objetos do cotidiano sendo conectados à internet proporcionando o controle e a análise de dados (MOUHA, 2021).

A Internet das coisas é aplicada nas mais diversas áreas, seja residência tornando as casas inteligentes, saúde, cidades inteligentes, indústria e agricultura. Os principais elementos que a caracterizam são o *hardware* (com componentes

embarcados), um *middleware* com capacidade de manipular os dados enviados pelos objetos e uma camada de apresentação em formato de software ou sistema, conforme exemplo de arquitetura para uma residência, apresentada na Figura 2.

Figura 2 – Arquitetura IoT para uma residência



Fonte: Adaptado de Albuquerque e Bastos (2017).

Na Figura os sensores coletam os dados do ambiente e enviam os dados a um dispositivo controlador IoT conectado à internet, também chamado *gateway*, que por sua vez envia os dados para a rede. Os dados são visualizados por um *dashboard* (tela de um monitor ou celular) para serem apresentados aos clientes (*client*) ou são processados para enviar um sinal de controle para os atuadores (acionando um ar condicionado, ligando uma lâmpada em uma residência, etc.), automatizando o sistema.

3.3 IoT na Agricultura

A Internet das Coisas (IoT) aplicada na agricultura tem utilizado vários dispositivos conectados à internet, como sensores, drones e máquinas automatizadas, para monitorar, otimizar e automatizar as atividades agrícolas. Para isso, coleta e analisa dados em tempo real, facilitando a tomada de decisões de forma mais precisa e eficiente.

Na era da tecnologia, a agricultura vem passando por constantes transformações, oriundas da internet das coisas, chegando na Agricultura 4.0 que representa a convergência entre a revolução digital e a agropecuária. Na Figura 3 pode-se acompanhar as evoluções que ocorreram na Agricultura.

Figura 3 – Evolução da Agricultura



Fonte: Colli (2023).

A IoT possibilitou a conexão inteligente entre objetos físicos e dispositivos, criando uma rede interconectada que proporciona a coleta e o compartilhamento de dados em tempo real, permitindo que os agricultores monitorem e gerenciem suas operações de maneira mais eficiente e com precisão (COLLI, 2023).

A quarta revolução na agricultura, que veio com a revolução digital, trouxe uma inovação necessária para atender de modo incisivo, o aumento da demanda por alimentos, as mudanças climáticas e a necessidade da adoção de práticas mais sustentáveis.

A tecnologia atual vai além da automação convencional a qual estávamos acostumados, pois cria uma rede interconectada, na qual esses dados podem ser coletados e analisados em tempo real, de maneira facilitada.

A utilização da IoT também oferece benefícios e retornos na agropecuária, destacando-se o monitoramento em tempo real e detalhado, com análises precisas; disponibilidade de dados que contribuem diretamente para o uso eficiente de insumos, como água e defensivos e a adoção de práticas mais eficientes que resultam no aumento da produtividade e o aumento da produtividade (AGROADVANCE, 2024).

A aplicação prática da IoT na agricultura é observada em cinco exemplos, seja na agricultura de precisão, monitoramento das culturas, o rastreamento de ativos e equipamentos, a gestão inteligente da irrigação e o controle fitossanitário.

Na agricultura de precisão, os agricultores se tornaram capazes de realizar a coleta de dados detalhados com o auxílio de sensores e dispositivos conectados, sobre

variáveis (como clima, solo, condições das culturas etc.), em tempo real. Com estes dados, algoritmos avançados realizam seus processamentos, gerando aos produtores resultados, que mostram onde deve haver o ajuste de variáveis como irrigação, fertilização e controle de pragas e doenças, por exemplo. Estas práticas agrícolas, baseada em dados concretos e precisos, representam um avanço significativo na abordagem das práticas agrícolas.

O monitoramento de culturas implica na análise do solo, condições ambientais, etc. por meio de sensores integrados a IoT, que realizam a coleta de dados em tempo real e oferecem uma visão ampla das condições nas quais as plantas se encontram, visando otimizar o uso dos recursos e evitar desperdícios. A captação de imagens aéreas por um drone, aliada a sensores em uma fazenda podem auxiliar um agricultor a tomar decisões com antecedência.

O rastreamento de ativos e equipamentos (maquinários) acontece em tempo real por causa da IoT, com precisão a respeito da localização e do *status* operacional (previne falhas no maquinário e otimiza a sua vida útil).

A gestão inteligente da irrigação é proporcionada pela conexão dos sensores que detectam e realizam uma leitura da umidade do solo à um sistema IoT, com isso, há o uso racional da água, reduzindo o desperdício, promovendo o uso sustentável deste recurso, aumento na produtividade, uma vez que as plantas receberão a quantidade adequada e ideal para seu desenvolvimento (COLLI, 2023).

O controle Fitossanitário pode ser obtido também com as soluções avançadas que a IoT oferece, com isso pode-se observar uma redução na dependência de agroquímicos, isso devido ao fato que o monitoramento e controle de pragas se torna mais preciso. A detecção precoce de pragas e plantas invasoras podem ser feitas combinando sensores IoT com algoritmos de aprendizado de máquina, por exemplo, como consequência os produtores obtêm uma maior assertividade nas intervenções, minimizando o uso de defensivos agrícolas. Também combinações entre sistemas IoT e algoritmos, contribuem para a redução dos impactos ambientais, uma vez que agroquímicos serão utilizados de forma mais racional.

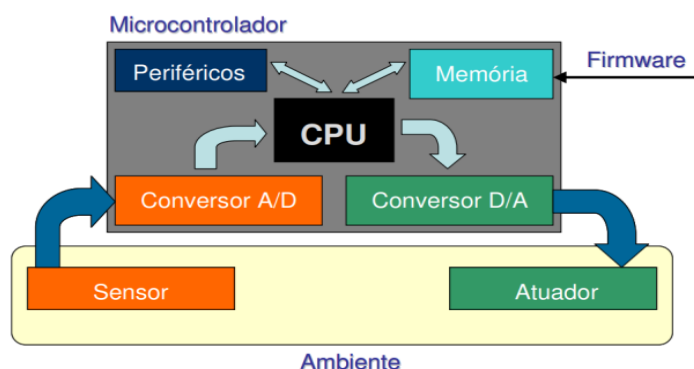
4. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo faz-se uma breve apresentação dos principais conceitos que envolvem o desenvolvimento do sistema de monitoramento e os componentes de hardware e software utilizados no projeto, tais como o ESP32, sensores, atuadores como o módulo relé e bomba d'água e as fontes de alimentação.

4.1 Sistemas Embarcados

Um conceito importante abordado no trabalho é o de sistemas embarcados (Embedded Systems), sistema computacional formado por hardware e software, desenvolvido para desempenhar uma função específica dentro de um sistema maior. Ele é incorporado a dispositivos ou equipamentos com o objetivo de controlar ou supervisionar determinada operação ou processo. Normalmente, esses sistemas são projetados para serem compactos e acessíveis, utilizando um número reduzido de componentes, como microcontroladores, sensores e atuadores. (EMBARCADOS, 2022). Na Figura 4 mostra-se um exemplo de arquitetura de um sistema embarcado utilizando um microcontrolador.

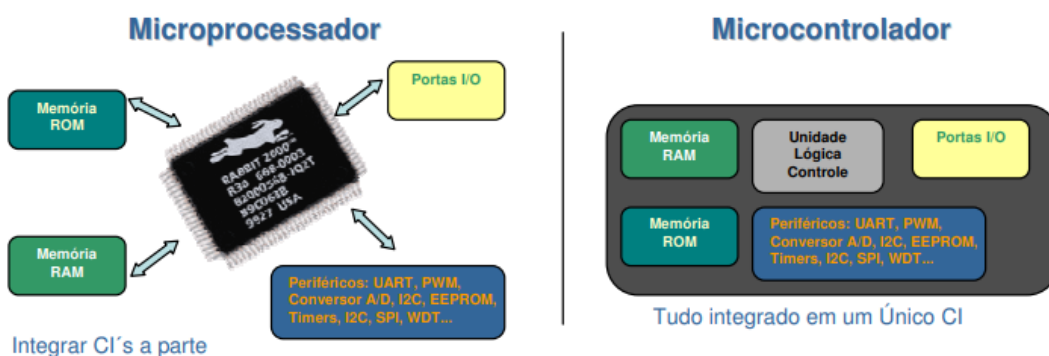
Figura 4 – Arquitetura de um sistema embarcado.



Fonte: SBAJovem (2007).

Um microcontrolador é um circuito integrado (CI) que engloba interfaces de entrada/saída digitais e analógicas, memórias RAM e FLASH, interface de comunicação serial, conversores analógicos/digitais e temporizadores/contadores, diferente de um microprocessador que precisa integrar vários componentes desses como memórias em placas, separadamente, na Figura 5 ilustra-se essas diferenças.

Figura 5 – Microprocessador x Microcontrolador.



Fonte: SBAJovem (2007).

Atualmente, os sistemas embarcados estão presentes em uma ampla variedade de aplicações, como em telas de fornos de micro-ondas e no controle de temperatura de aparelhos de ar-condicionado, além de diversos outros eletrodomésticos, por sua versatilidade e capacidade de programação tem sido utilizados em diferentes setores industriais.

4.2 Placa de desenvolvimento

No mercado de placas de desenvolvimento ou também referidas como plataformas de controle ou controladores de sistemas são encontrados diversos produtos com bons recursos de CPU, memórias, comunicação Wifi, alto clock e etc., a maioria sendo acessível em termos de preço.

O módulo ESP32 é uma plataforma poderosa baseada no chip ESP32-D0WDQ6, evolução do módulo ESP8266 da Espressif Systems, incorporando conectividade WiFi e conexão Bluetooth. Entre as características destacam-se a ampla aplicação em projetos eletrônicos, dos mais simples aos mais complexos, dispensando o uso de outros microcontroladores, usado principalmente em projetos da Internet das Coisas – IOT e Robótica. O ESP 32 Módulo Wifi e Bluetooth possui o microprocessador Xtensa 32-Bits LX6 que apresenta baixo consumo de energia e se destaca também pela presença de dois núcleos de CPU (processador dual core), permitindo aplicação em conjunto com sensores, módulos e diferentes interfaces. Possibilita a atualização do firmware via WiFi e tem ampla capacidade de processamento. No Quadro 2 apresenta-se uma comparação dos recursos do Arduino Uno R3 (um dos mais populares) e o ESP32.

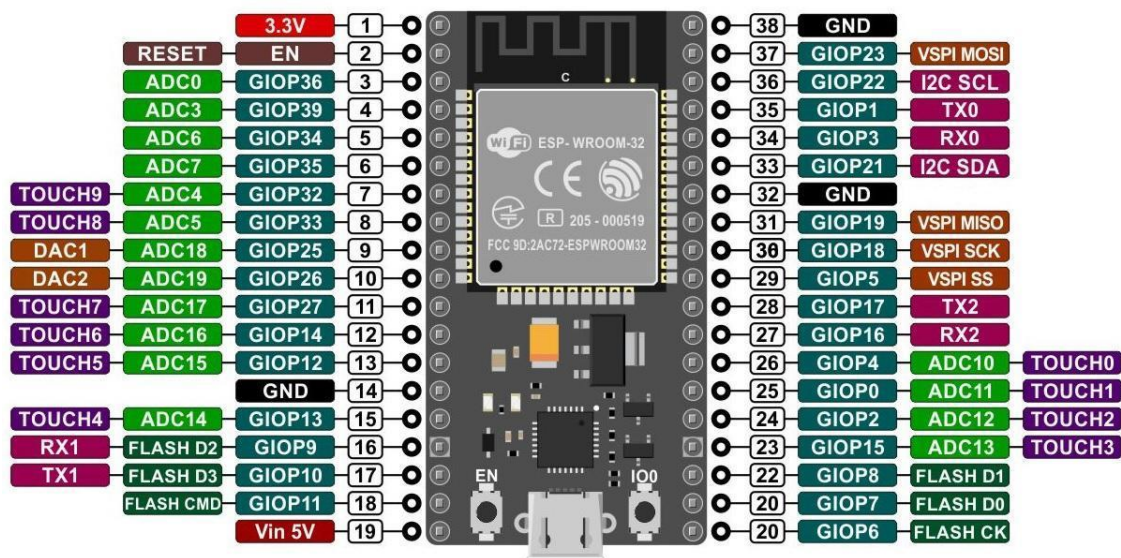
Quadro 2 – Arduino Uno R3 e o ESP32.

	ESP32	Arduino
Processamento	CPU Tensilica Xtensa LX6	Microcontrolador 8-bit ATmega328P
Tensão de operação	3.3V	5V
Memória RAM	520KB	2KB
WiFi	Sim	Não (apenas com Shield)
Bluetooth	Sim	Não
Linguagem de programação	Python, MicroPython, C, e C++.	C e C++
Frequência de operação	240 MHz	16 MHz
Memória Flash	16MB	4MB
Entradas Analógicas	18	6
GPIOs	48	20, (6 são PWM)
Conversor ADC	12 bits	10 bits
Interface USB	Micro-USB	USB Tipo-B
Suporte a comunicação I2C, SPI e UART	Sim	Sim

Fonte: Adaptado de Eletronic (2024).

Observa-se, portanto, que o ESP32 tem mais recursos para projetos de automação e coleta de dados devido à sua conectividade WiFi e Bluetooth integrados, permitindo sua conexão a uma rede doméstica e a comunicação com outros dispositivos, além de maior capacidade de processamento e quantidade de memória para armazenamento. Na Figura 6 tem-se a pinagem do ESP32.

Figura 6 – Pinagem do ESP32.



Fonte: Armijos (2025).

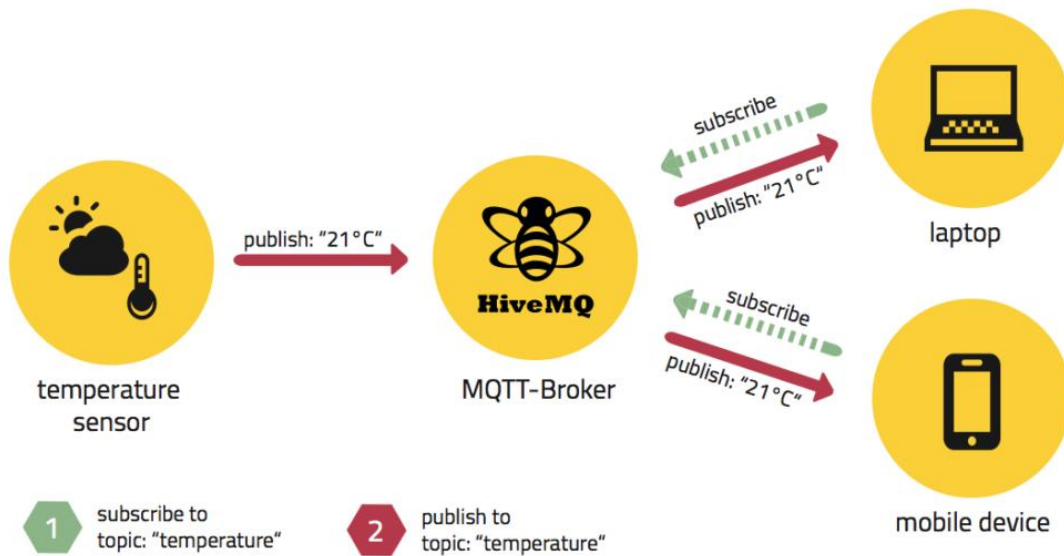
4.3 Protocolo de Comunicação MQTT

Outro conceito importante tratado nesta pesquisa é o MQTT (*Message Queuing Telemetry Transport*) protocolo de comunicação máquina para máquina (M2M - *Machine to Machine*) para IoT que funciona com base no protocolo TCP/IP. Um sistema MQTT se baseia na comunicação entre cliente e servidor, no estilo *Publish-Subscribe* (publicar-assinar). O protocolo se popularizou pela simplicidade, baixo consumo de dados e pela possibilidade de comunicação bilateral (NERI; LOMBA; BULHÕES, 2019).

O MQTT surgiu em 1999 quando engenheiros na indústria de petróleo e gás necessitavam de um protocolo de comunicação robusto e de baixo consumo de energia e dados, feito para baixas larguras de bandas e sistemas de alta latência, visto que as plataformas são *offshore* e a comunicação é feita via satélite.

O MQTT funciona como cliente/broker (servidor), sendo o *broker* responsável por receber todas as mensagens enviadas pelos clientes e entregá-las aos clientes que se inscreveram (assinaram) nos tópicos correspondentes. Os clientes são os dispositivos que publicam ou assinam mensagens nos tópicos do *broker*. Comparando o MQTT com o funcionamento dos e-mails, pode-se entender os tópicos como os "assuntos" dos e-mails e as mensagens publicadas como o conteúdo do e-mail, um exemplo da utilização do protocolo MQTT é mostrado na Figura 7.

Figura 7 – Exemplo de utilização do protocolo MQTT.



Fonte: Basílio (2018).

Neste exemplo o sensor de temperatura é um cliente assinado como “Publisher” no tópico “*temperature*”, ou seja, a temperatura medida será publicada no tópico. O laptop e o celular são clientes assinando o tópico como “Subscriber”, o que significa que eles terão acesso aos dados publicados pelo sensor. É importante ressaltar que os assinantes dos tópicos não precisam se conhecer (ou estar conectados), basta conectar-se ao broker, o qual recebe as requisições e as responde.

4.4 Plataforma de Visualização e Análise de Dados

A análise dos dados por meio da rede é indispensável para o monitoramento e controle de sistemas. Existem diversas plataformas que permitem a visualização dos dados e até controle de atuadores remotamente, algumas dessas plataformas são Google Cloud IoT, Thingspeak, Home Assistant, Azure IoT Central e outras, das quais se destacam o Thingspeak e o Node-Red.

4.4.1 Thingspeak

O Thingspeak (Mathworks, 2025) é uma plataforma IoT que permite visualizar e analisar dados diretamente da rede. Pode ser utilizada tanto no celular quanto no computador, basta que o dispositivo ligado à plataforma tenha conexão com a internet para monitorar os dados, remotamente, de qualquer lugar do mundo.

Existe a versão paga e a versão gratuita, porém a versão gratuita possui algumas limitações, tais como, a quantidade de canais e o tempo de upload de dados mínimo de 15 segundos. Um exemplo de visualização do *dashboard* no Thingspeak é mostrado na Figura 8 .

Figura 8 – Visualização de dados na plataforma do Thingspeak



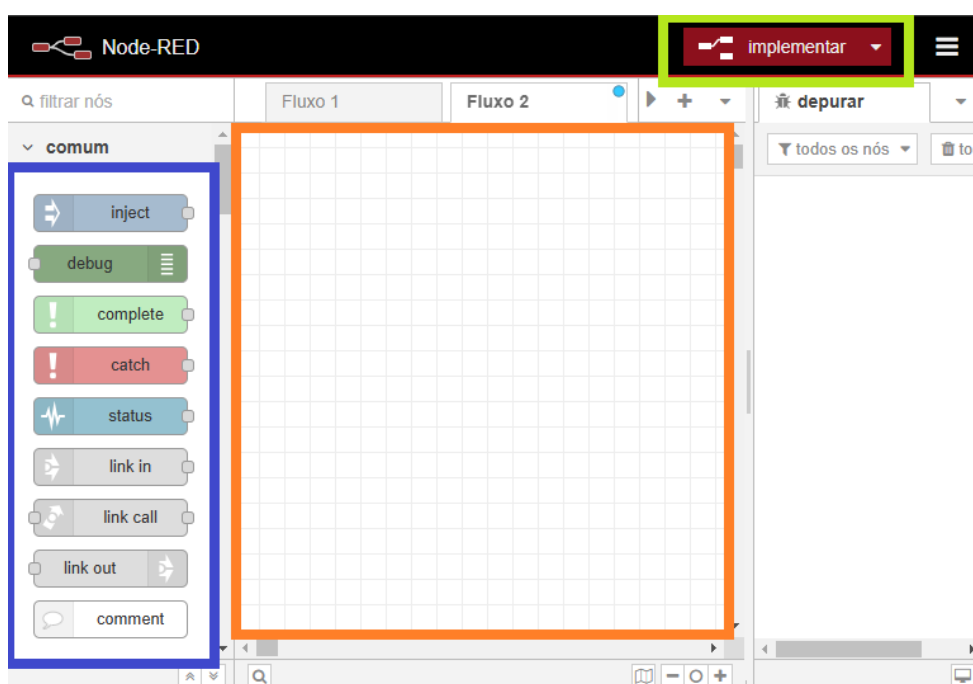
Fonte: Santiago (2025).

Na Figura 8, nos gráficos superiores são mostrados o valor atual de temperatura e umidade e os gráficos inferiores, as ondas mostram os valores ao longo do dia. O Thingspeak, por mais que seja uma ferramenta muito versátil de visualização de dados, não permite que sejam feitos acionamentos por meio da sua plataforma.

4.4.2 Node-Red

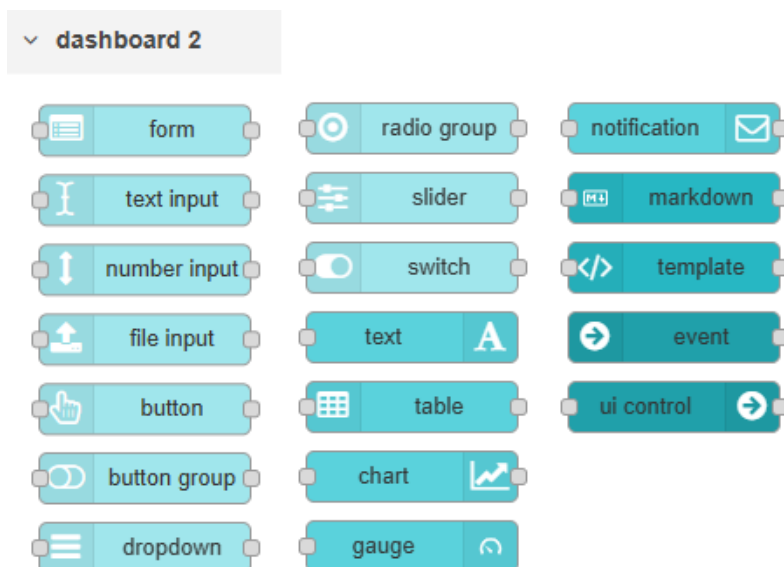
O Node-Red é uma ferramenta *open source* para criação de aplicações IoT que utiliza programação visual, cuja interface é mostrada na Figura 9. Para executar uma determinada tarefa blocos de códigos pré-definidos, conhecidos como "nós" (que estão em azul na imagem) são conectados. Os nós conectados, geralmente são uma combinação de nós de entrada, nós de processamento e nós de saída, quando ligados juntos, formam um "fluxo", os fluxos são montados na área contornada em laranja. O Node-red não salva automaticamente então toda vez que uma mudança nos fluxos é feita o botão "implementar" (contornado em verde) deve ser pressionado.

Figura 9 - Interface do Node-Red.



Fonte: Própria do autor.

Existem tipos diferentes de nós que podem ser utilizados nos fluxos. É chamado de 'Paleta' o conjunto de nós que possuem utilização parecida. Por padrão, vem instalados as paletas de nós comuns, de funções, redes, sequência e armazenamento. Porém, o usuário pode obter outras paletas e nós diferentes, por meio das bibliotecas. Neste trabalho foi utilizada a paleta *dashboard* 2.0, os nós que compõem esta paleta são mostrados na Figura 10.

Figura 10 - Nós da paleta *dashboard* 2.0.

Fonte: Própria do autor.

Outro nó importante e versátil é o nó de função que permite a implementação de lógicas personalizadas por meio de código JavaScript. Com ele, é possível manipular os dados que trafegam pelo fluxo, criar condições, realizar cálculos, formatar mensagens e até mesmo controlar a lógica de encaminhamento entre diferentes partes do sistema.

Assim como no ThingSpeak, pode-se monitorar a leitura de sensores em tempo real. O grande diferencial do Node-red encontra-se no acionamento de atuadores via interface web manualmente, por botões implementados no *dashboard* ou por acionamentos automáticos programados e também a liberdade que o usuário possui para criar os fluxos da maneira que achar necessário (CELESTRINI, 2021).

Para este trabalho, o Node-Red funcionará como um cliente subscrevendo os tópicos em que o ESP32 publicará, tratando os dados analógicos e apresentando a visualização das grandezas ao usuário em uma *dashboard* por meio de gráficos.

4.5 Sensores

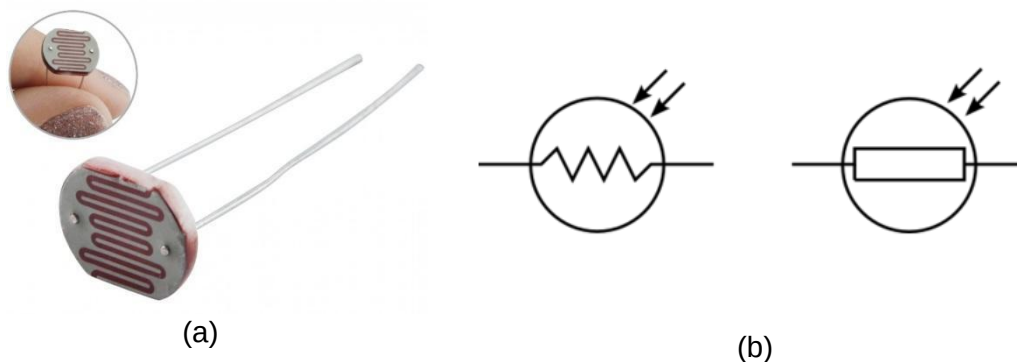
Os sensores são responsáveis pela interação da plataforma de controle com o ambiente em que está inserido um sistema. Eles respondem a estímulos externos e produzem um sinal que pode ser mensurado. Existem vários dispositivos que respondem a diferentes estímulos, tais como pressão, temperatura, umidade, presença

de campo magnético, entre outros. Nesta pesquisa foram usados os sensores LDR, DHT22, umidade relativa do ar e umidade do solo.

4.5.1 Sensor LDR

O LDR (*Light Dependent Resistor* - “Resistor dependente da luz”) é um bipolo sem polaridade, cuja resistência varia de acordo com a intensidade da luz. É composto por duas células fotocondutivas de sulfeto de cádmio (CdS) ou sulfeto de chumbo (PbS) com respostas do espectro visível semelhantes à do olho humano. Quanto mais luz incidir sobre o componente, menor a resistência. Por este motivo ele é classificado como um dispositivo fotoelétrico ou fotoresistor, ou ainda fotossensor, pois altera as suas propriedades elétricas devido à incidência de luz. O sensor e sua simbologia são mostrados na Figura 11.

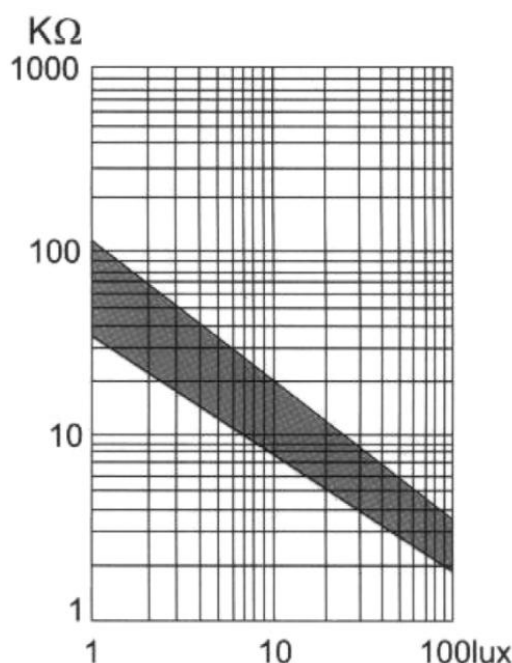
Figura 11 – (a) Sensor LDR. (b) Simbologia



Fonte: Meneses (2017).

Em estado normal o material do LDR possui uma alta resistência. Quando as partículas de luz (fótons) incidem sobre a superfície do sensor, os elétrons que estão no material semiconductor são liberados, dessa forma a condutividade do LDR aumenta e a sua resistência diminui (MENESES, 2017). É importante ressaltar que o LDR é adequado para medir ausência e presença de luminosidade, e não para medir a quantidade de iluminância (em Lux) com precisão e exatidão (MARKERHERO, 2024). Na Figura 12 mostra-se em escala logarítmica, a curva característica de iluminância (Lux) x Fotoresistência do sensor LDR (medida em Ohms, Ω).

Figura 12 - Foto resistência do sensor LDR x iluminância



Fonte: Datasheet Photoconductive Cells (2025).

Nota-se que no gráfico a resistência do LDR diminui à medida que a incidência de luz aumenta. Em baixa luz (1 lux), a resistência do LDR pode ser superior a 100 kΩ, com mais luz (10 lux), a resistência reduz para valores entre 10 e 100 kΩ e em ambientes muito iluminados (100 lux ou mais), a resistência pode cair para menos de 10 kΩ (MARKERHERO, 2024)

Ainda neste gráfico, a área sombreada representa a faixa típica de variação da resistência para diferentes LDRs, indicando que sensores similares podem ter pequenas diferenças nas características. Nessa pesquisa o LDR é utilizado para medir a luminosidade externa da estufa.

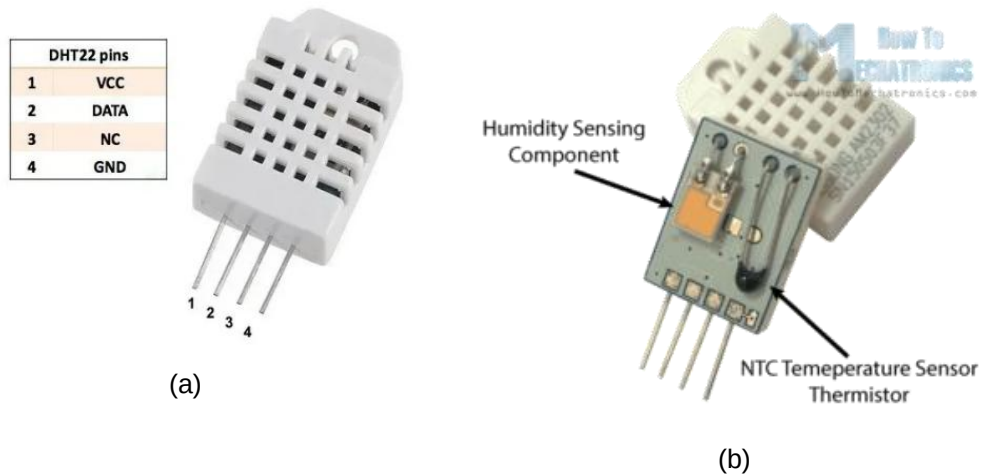
4.5.2 Sensor DHT22

O sensor DHT22 possui um sensor de umidade capacitivo (componente sensível à umidade) e um termistor (NTC-Temperature Sensor Thermistor) interno para medir a temperatura. Converte os valores em um sinal digital e os envia pelo único pino de dados. Como a sua atualização é feita a cada 2 segundos, é considerado de comunicação relativamente lenta, não sendo ideal para medições em tempo real que

exijam rapidez. Esse sensor tem boa precisão, uma faixa de medição de temperatura abrangente, baixo consumo de energia e baixo custo.

Na Figura 13 (a) apresenta-se a pinagem do sensor DHT22 e seus componentes internos na Figura 13 (b). No Quadro 3 são resumidas as características técnicas do DHT22.

Figura 13 - (a) Pinagem (b) Sensores internos



Fonte: DHT22-AM2302-Datasheet (2024)

Quadro 3 – Características técnicas do DHT22.

Características Técnicas	
Tensão de operação:	3,0 – 5,0V (5,5V máximo)
Faixa de medição de umidade:	0 a 100% RH
Faixa de medição de temperatura:	-40° a 80°C
Corrente:	máx de 2,5 mA em uso 100µA a 150µA em modo de espera
Precisão de umidade de medição:	± 2,0% UR
Precisão de medição de temperatura:	± 0,5°C
Tempo de resposta:	< 2 segundos
Dimensões:	25 mm x 15 mm x 7 mm (sem terminais)

Fonte: DHT22-AM2302-Datasheet (2024).

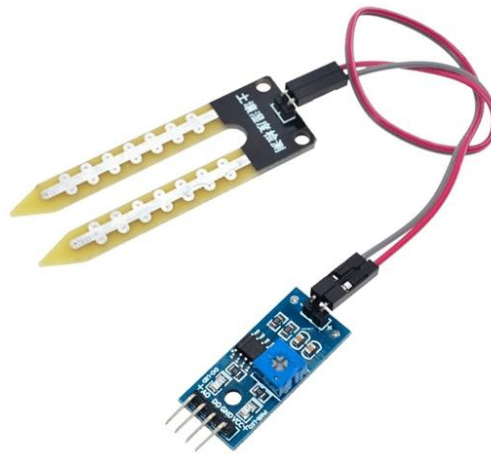
4.5.3 Sensor HL- 69

O sensor HL-69 mostrado na Figura 14 mede a umidade do solo, e é do tipo resistivo que inclui um comparador LM393 para garantir precisão nas leituras. Fornece

tanto a saída digital quanto a analógica, permitindo uma integração flexível com sistemas diversos. Sua sensibilidade é ajustável por um potenciômetro, adaptando-se às características específicas do solo.

Além disso, possui LEDs indicadores de tensão (luz vermelha) e para a saída digital (luz verde), facilitando a visualização do status operacional. Com estas características, este sensor é uma ferramenta confiável para monitorar e controlar a umidade do solo em uma variedade de aplicações.

Figura 14 - Sensor de umidade do solo



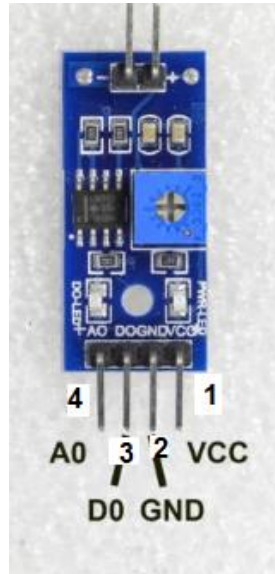
Fonte: Makerhero (2016).

O sensor de umidade do solo resistivo opera com base na variação da resistência elétrica do solo conforme sua umidade. Consiste em duas sondas metálicas que são inseridas no solo. Quando o solo está úmido, sua resistência elétrica é menor devido à presença de água, permitindo que a corrente elétrica flua mais livremente entre as sondas. Por outro lado, quando o solo está seco, sua resistência aumenta, dificultando o fluxo de corrente entre as sondas. O sensor mede essa resistência elétrica e a converte em um sinal elétrico proporcional ao nível de umidade. Esse sinal pode ser lido por um microcontrolador como o Arduino ou outro qualquer que tenha características similares, para monitorar e controlar a irrigação ou outras aplicações relacionadas à umidade do solo.

O módulo de condicionamento do sinal do sensor de umidade do solo possui quatro pinos: A0 (saída analógica), D0 (saída digital), VCC para alimentação e GND conectado à terra. A0 é usado para enviar os dados analógicos do sensor para o

microcontrolador e o pino D0 envia dados digitais. Além disso, possui os pinos + e – que são conectados aos pinos + e – da sonda resistiva, conforme visto na Figura 15.

Figura 15 – Pinos do sensor de umidade do solo, módulo HL- 69.



Fonte: Própria do autor.

Além disso, o módulo possui um potenciômetro para um ajuste fino do limite entre o seco (nível lógico 1) e úmido (nível lógico 0) da saída digital. A saída analógica funciona como no sensor LDR, vai de 0 a 1023 (resolução de 10bits), quanto mais seco, o Arduino mostrará valores mais perto de 1023 (resistência alta, baixa condutividade). No Quadro 4 tem-se um resumo de suas características.

Quadro 4 – Características técnicas do sensor de umidade do solo, HL- 69.

Tipo:	resistivo
Tensão de operação:	3.3V ~ 5V;
Compatível com:	Arduino, PIC, AVR, ARM, etc;
Comparador	LM393
Saídas	Digital (D0 e analógica (A0);
Sensibilidade ajustável	
Dimensões:	sensor com sondas (CxL): ~60x20mm; circuito com trimpot (CxL): ~31x14mm;

Fonte: Própria do autor.

Para a determinação da umidade do solo é necessário a sua calibração para o tipo de solo. Gomes *et al.* (2017) fizeram um levantamento da precisão da calibração do

HI-69, cujos os erros obtidos no processo de calibração do sensor são resumidos na Tabela 1.

Tabela 1 - Resumo de erros de calibração do sensor

Características	Erro (%)		
	0-10 cm	10-20 cm	20-40 cm
Erro máximo absoluto	5,7	2,3	5,9
Histerese	3,5	3,8	3,2
Repetitividade - incerteza (95%)	1,7	1,2	1,9
Repetitividade - desvio máximo	3,2	1,8	3,4
Não-linearidade	9,9	2,7	10,5

Fonte: Gomes *et al.* (2017).

Conforme a área de contato do sensor que é considerada pequena é necessário obter um maior número de avaliações ao longo do perfil do solo para melhor representatividade da umidade do solo. Mas, em relação a outros sensores do mesmo tipo, e desde que seja calibrado corretamente, o sensor HI-69 apresenta um bom custo-benefício (GOMES *et al.*, 2017).

4.6 Atuadores

Neste item apresenta-se o módulo relé, um importante componente usado para ligar ou desligar dispositivos elétricos atuadores, como interruptores, motores, lâmpadas, ventiladores entre outros em sistemas de automação, visando a proteção dos dispositivos de baixa tensão, tais como os microprocessadores/microcontroladores. Plataformas de controle como o Arduino ou ESP32 suportam tensões de 5V/3,3V, respectivamente, para atuar com componentes que precisam de uma alimentação maior, como 12V ou 110V. Apresenta-se a seguir, o módulo relé utilizado nesta pesquisa.

4.6.1 Relé

Um relé é um interruptor operado eletricamente que pode ser ligado para permitir a passagem da corrente ou desligado e projetados para serem controlados com baixas tensões como 3.3V ou 5V. Desta maneira com uma tensão baixa é possível ligar ou desligar dispositivos de alta tensão. (ELETRÔNICA, 2025). Na Figura 16 é apresentado um módulo relé 5V, 10A de um canal, podendo acionar um dispositivo de corrente

alternada, de até 10A, como, por exemplo, lâmpadas, portões eletrônicos, ventiladores, etc.. No Quadro 5 são apresentadas suas características e especificações.

Figura 16 – Módulo relé 5V , 10A de um canal



Fonte: Própria do autor

Quadro 5 - Resumo das características do relé

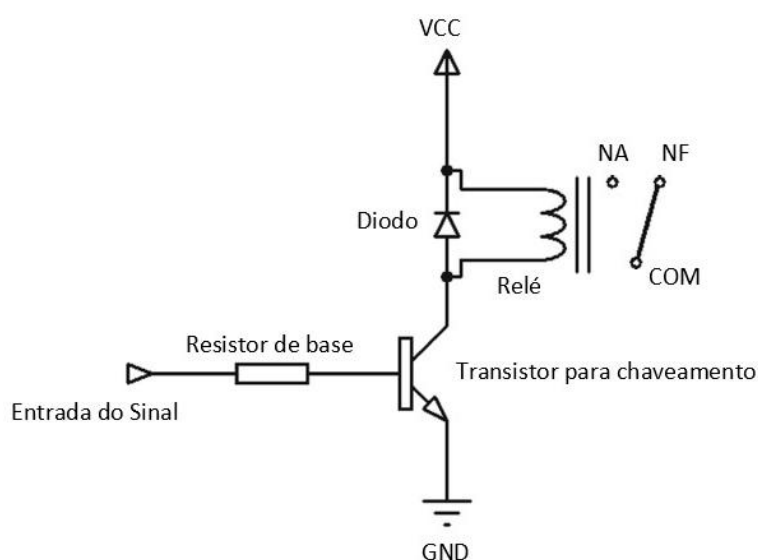
Características	
Módulo Relé 5V 1 Canal :	Modelo: JQC3F-C;
Compatível com:	Arduino, ESP32, PIC, ARM, Raspberry PI, etc.;
Controle AC/DC com LED indicador de status;	
Tensão de operação:	5VDC (VCC);
Pinos de entradas:	SIG/VCC/GND;
Tensão de sinal:	TTL 5VCD;
Pinos - Carga nominal:	10A 250VAC/ 10A 125VAC/ 10A 30VDC/ 10A 28VDC;
Saídas:	Contato reversível NA (normal aberto), NF (normal fechado), C (comum);
Corrente por canal:	até 10A;

Fonte: Usinainfo (2022).

O funcionamento de um relé é semelhante ao de um interruptor, se a carga for conectada à saída NF, ela será ativada quando receber um sinal digital nível baixo de um microcontrolador e será desligada quando receber um sinal digital de nível alto. Analogamente, se a carga for conectada à saída NA, ela será ativada quando receber um sinal digital nível alto do microcontrolador e será desligada quando receber um sinal digital nível baixo. Conforme visto na tabela de características do relé, o sinal de acionamento (5V) deve ser conectado na entrada SIG.

O circuito interno de um relé básico conta com um transistor responsável por comutar o circuito, conforme mostrado na Figura 17. A ativação (ou acionamento) ocorre por meio de uma pequena corrente de base (entrada do sinal), que pode ser gerada por um dispositivo como um microcontrolador, alterando o estado do relé. O diodo conectado em paralelo e invertido em relação ao relé tem a função de proteger os componentes responsáveis pelo acionamento, como transistores (ou SCRs). Isso acontece porque, ao desligar uma carga indutiva (como um motor ou o próprio relé), o campo magnético gerado se contrai, resultando em uma alta tensão reversa que pode danificar o transistor de controle. O diodo, por estar invertido, não conduz durante a operação normal, mas entra em ação no momento do desligamento, dissipando a corrente induzida e protegendo o transistor (ou SCR. (ELETRÔNICA, 2025).

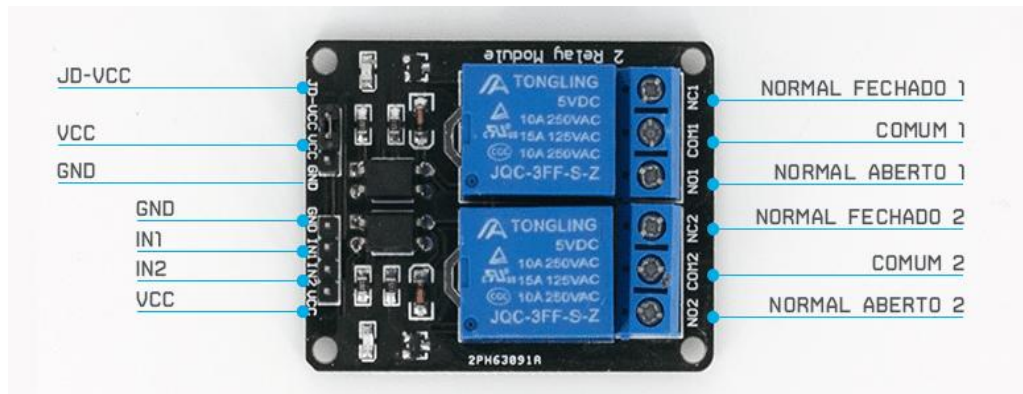
Figura 17 - Circuito interno de um relé básico



Fonte: Eletrônica (2025).

Atualmente, existem vários módulos relé, sendo um deles apresentado na Figura 18, módulo relé 5V, 10A de dois canais com optoacopladores, podendo acionar dois dispositivos diferentes.

Figura 18 – Módulo relé de dois canais 5V com optoacoplador



Fonte: Handsontec (2025).

Este exemplar é constituído por dois relés apresentando três pinos de saída, o normalmente aberto (NA), o comum (C) e o normalmente fechado (NF); possui 2 pinos para a alimentação, VCC (5V) e um GND; as entradas do sinal In1 e In2, servem para acionar ou desligar os relés, portanto, devem ser conectados à alguma das portas digitais de um microcontrolador, essa porta digital deverá ser definida na programação como saída, para que os sinais (alto e baixo) possam ser enviados ao módulo. Este módulo é opticamente isolado do lado de alta tensão e também previne o *loop* de terra (ground loop) quando interfaceado com um microcontrolador (VIDAL, 2017).

Este módulo Relé (SRD-05VDC-SL-C) com optoacopladores, permite a integração com microcontroladores como Arduino, AVR, PIC, ARM. A partir de suas saídas digitais pode-se, por meio do relé, controlar cargas maiores e dispositivos como motores AC ou DC, eletroímãs, solenoides e lâmpadas incandescentes. O módulo é equipado com um relé de alta qualidade, baixo ruído com carga nominal 10A/250VAC, 10A/125VAC, 10A/30VDC e cada canal possui um LED para indicar o estado da saída do relé. No Quadro 6 tem-se o resumo de suas características.

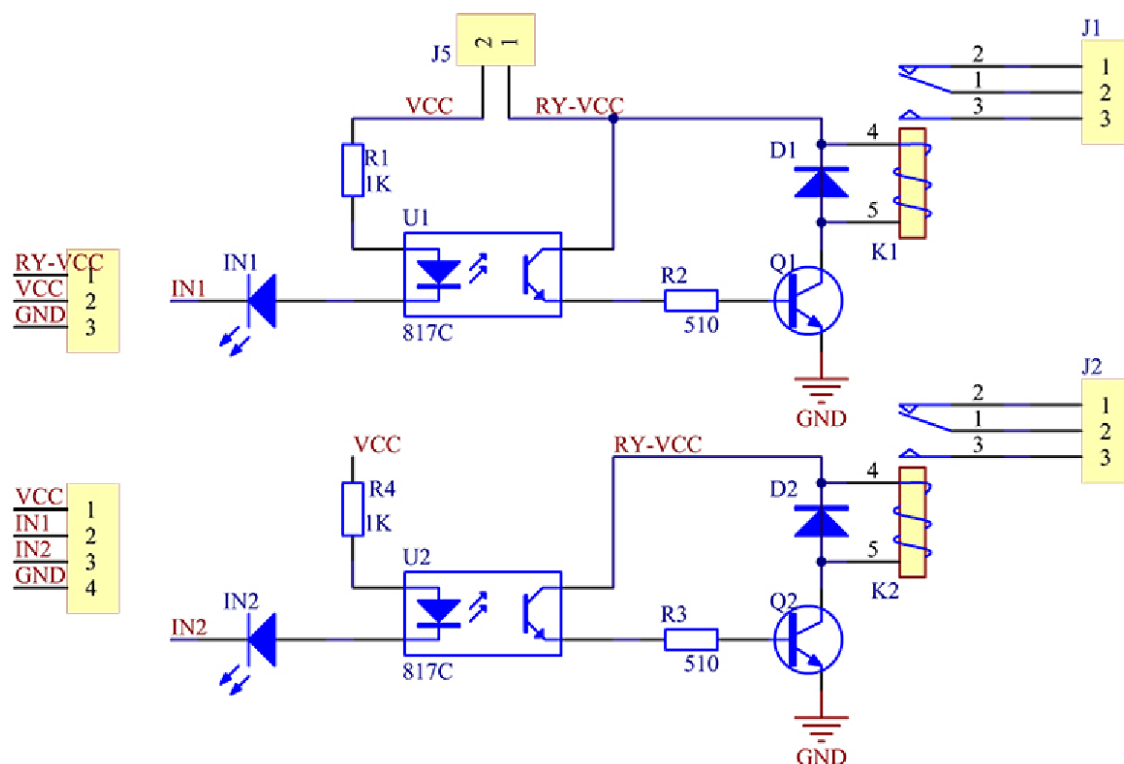
Quadro 6 - Módulo Relé 2 Canais 5V com optoacoplador

Especificações técnicas	
Tensão de operação:	5VDC (VCC e GND)
Tensão de sinal:	5VDC (IN1 e IN2)
Corrente típica de operação :	15 a 20mA
Cada relé possui :	3 terminais :contato NA, 1NF e o C (comum)
Contato do relé permite tensão de saída:	até 250VAC/10A , DC30V /10A
Tempo de resposta:	5 a 10ms
Indicador	LED

Fonte: Handsontec (2025).

Apresenta-se na Figura 19 o circuito esquemático do módulo relé de dois canais com optoacoplador. Neste, VCC e RY-VCC são tensões de alimentação do módulo e caso seja necessário alimentar uma carga maior, basta retirar a tampa do jumper e conectar uma alimentação extra ao RY-VCC para alimentar o relé; conectar VCC a 5V da placa do microcontrolador para fornecer sinais de entrada.

Figura 19 – Circuito esquemático para o módulo relé de dois canais com optoacoplador



Fonte: Handsontec (2025).

A isolação se completa conectando “VCC “para +5V do microcontrolador, mas o GND deve ser separado . Remover o jumper VCC para o JD-VCC. Conectar uma fonte de alimentação separada de +5V ao JD-VCC e o GND, fornecendo energia ao transistor e às bobinas do relé . Se o isolamento do relé for suficiente para a aplicação em questão, conecta-se o microcontrolador em +5 V e o GND, deixando o jumper VCC no JD-VCC (HANDSONTEC, 2025).

Às vezes é possível usar esse relé com sinais de 3,3 V, se o JD-VCC (Relay Power) for fornecido por uma fonte de +5 V e o jumper VCC para JD-VCC for removido. Essa fonte de relé de 5 V pode ser totalmente isolada do dispositivo de 3,3 V ou ter um aterramento comum, se o opto-isolamento não for necessário. Se usado com sinais

isolados de 3,3 V, o VCC (na entrada do opto-isolador, próximo aos pinos IN) deve ser conectado à fonte de +3,3 V do dispositivo de 3,3 V. Se desse modo não atuarem, pode ser necessário alterar o valor de R1 de 1000 ohms para próximo de 220 ohms ou fornecer +5 V para a conexão VCC (HANDSONTEC, 2025).

4.6.2 Bomba d'água e aspersor

Uma bomba de água é um dispositivo essencial quando se fala em transferir a água de um lugar para o outro, utilizada para o abastecimento das caixas d'água residenciais e também em sistemas de irrigação, que é o interesse desse trabalho. Basicamente, para levar água de um lugar ao outro, o dispositivo é interligado a canos que viabilizam esse processo de passagem e pressurização da água — Opera por meio de um mecanismo que cria vácuo ou pressão, fazendo a água fluir através de tubos ou mangueiras.

Neste trabalho foi utilizada uma mini bomba d'água Sapo Submersa Jt100, apresentada na Figura 20, operando em tensões baixas de 3V a 5V DC, 120l e preço reduzido. Esta mini bomba d'água é projetada para ser utilizada em projetos que necessitam que ela fique submersa, com garantia de total vedação da parte elétrica, em águas de profundidade de até 3m, pois possui proteção IP68¹. Sua vazão pode chegar a 120 litros por hora, porém depende da tensão e corrente que for alimentada. No Quadro 7 são resumidos os seus dados técnicos.

Figura 20 – Mini bomba d'água Jt100.



Fonte: Usinainfo (2025).

¹ **Proteção IP68**- Resistente à submersão em água significa que ela oferece a máxima proteção contra a

Quadro 7 - Dados técnicos da mini bomba d'água.

Modelo	JT100
Alimentação	3V - 5V DC
Corrente de trabalho	130 - 220mA
Potência	0,4 - 1,5W
Vazão de água	~1 a 1,5 l/m
Alto fluxo	Impulsiona até 1,5 litros/ min;
Total vedação com IP68;	Utilização apenas com água.
Elevação máxima	1m
Dimensões (CxLxA):	44x23,5x31mm
Peso	28g

Fonte: Usinainfo (2025).

Este modelo foi criado para o desenvolvimento de projetos de automação residencial (domótica) e protótipos robóticos baseados em plataformas microcontroladoras, tais como Arduino e Raspberry Pi.

Apresenta um motor compacto, capaz de impulsionar entre 1000ml a 1500ml por minuto, com eficiência e precisão durante sua execução em conjunto com o Arduino, por exemplo. A Mini Bomba de Água Submersa JT100 pode ser aplicada no desenvolvimento de robôs hidráulicos, irrigadores automáticos no caso de automação residencial, aquários, etc., foi muito aplicada em totens e sistemas automáticos de distribuição de álcool líquido. Por seu tamanho reduzido e baixo peso pode ser aplicada na grande maioria dos projetos, permitindo elevação máxima de até 1 metro (ALVES, 2025).

Para fazer um sistema de irrigação é utilizado também, juntamente com a bomba, um mini aspersor tipo gotejador, mostrado na Figura 21. Esse dispositivo, de baixa vazão, é projetado para liberar a água de forma localizada e gradual diretamente na base das plantas, o que resulta em economia de água e evita o encharcamento da terra. O modelo apresentado possui encaixe padrão para mangueiras de irrigação e uma tampa regulável que permite o controle da vazão, adaptando-se às necessidades

específicas de cada cultivo. Existem outros tipos mais sofisticados usados para a irrigação (HORTA VIVA, 2025).

Figura 21 – Mini aspersor



Fonte: Horta Viva (2025).

4.7. Fonte de alimentação

Em qualquer projeto, seja de automação ou outro sistema eletroeletrônico, um dos problemas cruciais é dimensionar uma fonte de tensão adequada às tensões e correntes para alimentar um sistema, ou seja, fornecer a energia necessária para o seu funcionamento.

Uma fonte de tensão contínua (CC), (*Direct Current* - DC, em inglês) fornece energia elétrica em uma única direção, com um fluxo de corrente constante. A corrente contínua não muda de direção, enquanto que a corrente alternada (AC) muda. Uma fonte de alimentação CC é um dispositivo que converte energia elétrica de corrente AC em corrente CC.

Fontes de tensão contínua são utilizadas em diversos equipamentos eletrônicos e industriais, como pilhas, baterias e conversores AC/DC. recarregar baterias e emular painéis fotovoltaicos. A fonte de potência CC, também chamada de fonte de corrente CC, fornece uma tensão (diferença de potencial) constante a sua carga e sua principal função é converter a corrente elétrica recebida conforme a tensão, corrente e frequência, corretas para alimentar uma carga.

Uma fonte chaveada de tensão baixa, ou seja, com saída em voltagens como 5V, 12V ou 24V, é um tipo de fonte de alimentação que utiliza um processo de conversão eletrônica para reduzir a tensão de entrada (geralmente da rede elétrica) para uma tensão mais baixa e contínua. Essa tecnologia é eficiente e compacta, permitindo o uso

de menos componentes em comparação com fontes lineares. De interesse deste trabalho, duas fontes de tensão são apresentadas, uma chaveada e outra simples.

4.7.1 Módulo MB-102

O módulo MB102 é um módulo de fonte de alimentação para matriz de contatos fabricado pela empresa YwRobot, na China. O módulo possui fonte 12V ajustável para 5,0/3,3 V para *protoboard* apresentado na Figura 22, é uma placa desenvolvida para instalação em um *protoboard*. Possui entrada de alimentação por plug P4 e USB, por meio de conexão do cabo com tensão de 6,5 a 12VCC ou cabo USB para a *protoboard* que o módulo fornece e tensão de saída ajustável de 3.3V e/ou 5V. É uma alternativa importante para os primeiros testes com um protótipo, uma vez que um *protoboard* é uma matriz de contatos desenvolvida para montagens de circuitos experimentais de projetos eletrônicos.

Figura 22 - Módulo Fonte 12V ajustável para *protoboard*

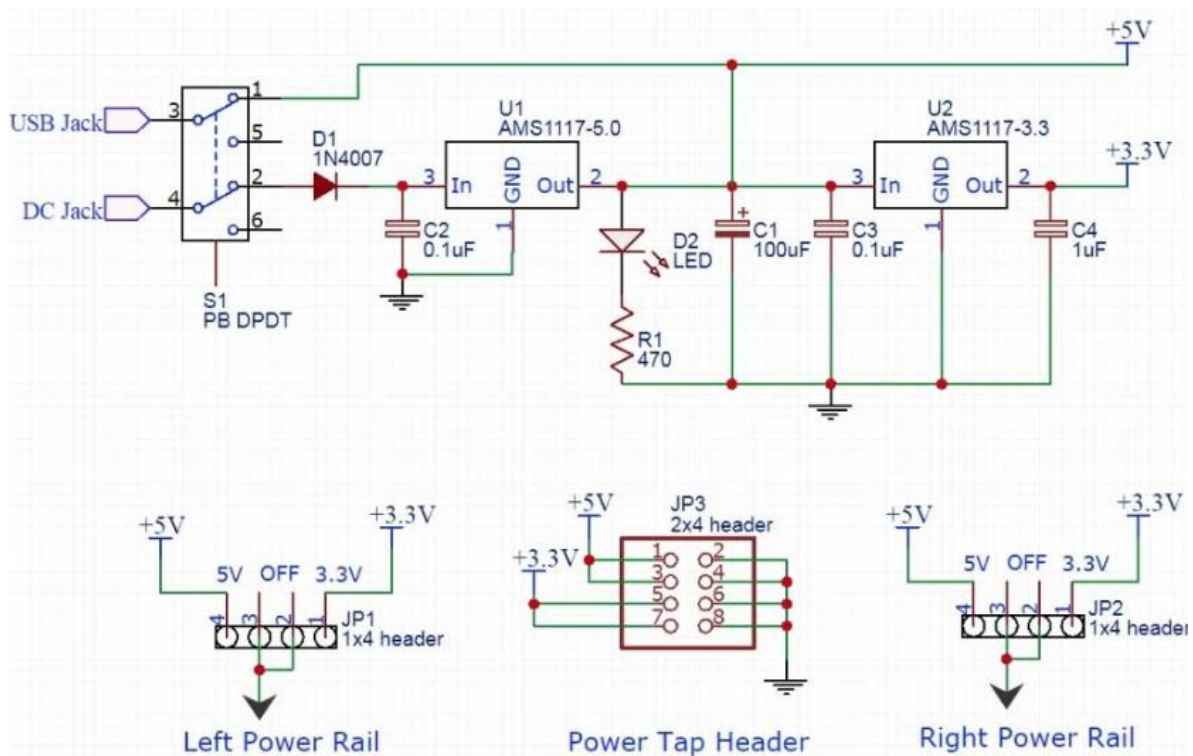


Fonte: Smartkits (2025).

A entrada USB também pode ser usada como saída para alimentar circuitos, uma vez que oferece em seus terminais 5V, permitindo ao projetista alimentar diferentes dispositivos. Também apresenta um interruptor liga/desliga que auxilia na montagem de um sistema, tornando desnecessário o desligamento do plug de alimentação AC.

A Fonte ajustável para *protoboard* tem um ajuste manual da tensão de saída (3,3V e/ou 5V) por meio de jumpers, e um LED indicador de funcionamento, sendo útil no desenvolvimento de projetos, conforme mostra-se no circuito esquemático, Figura 23.

Figura 23 – Circuito esquemático do Módulo Fonte 12V .



Fonte: Protosuplies (2025).

As fontes de tensão da esquerda e da direita podem ser configuradas de forma independente. A seleção da tensão de saída é feita pela colocação de jumpers no pino correspondente, um LED indica se está ligada. A placa não será alimentada, se os dois jumpers estiverem na posição “OFF”. Suas características técnicas são resumidas no Quadro 8.

Quadro 8 – Características técnicas da fonte de alimentação MB -102.

Modelo Fonte	MB-102
Tensão de entrada: e	6,5V a 12V (DC) ou via USB
Tensão de saída:	3,3V ou 5V (DC), ajustadas por jumper
Corrente máxima de saída :	700 mA
Chave liga / desliga	
LED indicador de operação	

Fonte: Protosuplies (2025).

4.7.2 Fonte Chaveada 12V (Tipo Colméia)

A Fonte Chaveada Colmeia 12V, 5A, 60W , ilustrada pela Figura 24, é geralmente utilizada na alimentação de circuitos eletrônicos, tais como, projetos de automação residencial, câmeras de segurança, CFTV (Circuito Fechado de Televisão), fitas de LEDs, entre outros, por ser um produto que apresenta alta eficiência.

Figura 24 – Fonte chaveada Colmeia 12V, 5A, 60W .



Fonte: Usinainfo (2025).

É dita ser fonte chaveada porque possui um circuito controlador interno que realiza o chaveamento da corrente elétrica. Esse processo consiste em ligar e desligar rapidamente a energia, com o objetivo de estabilizar a tensão de saída, o que é importante para muitos dispositivos eletrônicos sensíveis.

A fonte utilizada no projeto é de 127V e sua saída de 12VDC com corrente máxima de 5A possibilita alimentar vários dispositivos simultaneamente, desde que o limite de corrente de 5 A não seja ultrapassado.

Essa fonte apresenta uma construção metálica com partes em alumínio, o que auxilia na dissipação do calor gerado durante o funcionamento. A blindagem externa também contribui para a redução de interferências eletromagnéticas em outros equipamentos próximos. Outra característica importante é o peso que é muito inferior ao de um transformador, devido a aplicação de maior tecnologia.

Uma curiosidade, o termo “fonte colmeia” se refere a uma fonte de alimentação chaveada eletrônica que tem a aparência semelhante a uma colmeia de abelhas, com várias saídas de conexão em um único bloco de energia. As fontes de alimentação chaveadas são muito usadas porque são eficientes, compactas em comparação com outras fontes de alimentação, boa relação custo x benefício (USINAINFO, 2025b). No Quadro 9, são resumidas suas características e especificações.

Quadro 9 – Características técnicas da fonte de alimentação chaveada.

Modelo Fonte	Modelo: MS60-12
Tensão de Entrada:	100 a 240VAC 1.18A (Bivolt);
Tensão de saída estável e Corrente máxima de saída:	12VDC e 5A; 60W
Pode alimentar um ou mais circuitos:	SIM
Chave para regulagem mínima da tensão de saída	SIM
LED indicador de status	SIM
Peso	190g
Dimensões (CxLxA):	97x99x36mm ou 78x110x35mm;

Fonte: Usinainfo (2025b).

Uma fonte chaveada converte corrente alternada em corrente contínua, fornecendo tensões elétricas com boa capacidade de corrente sem necessitar de um transformador,

ou seja, substituem as fontes lineares, que necessitam de transformadores, uma importante vantagem, pois não usam um grande transformador, pesado e caro. Constitui um componente compacto e muito poderoso, indicado para aparelhos que necessitam de corrente contínua para funcionar e que possui uma ótima eficiência energética (ALVES, 2018).

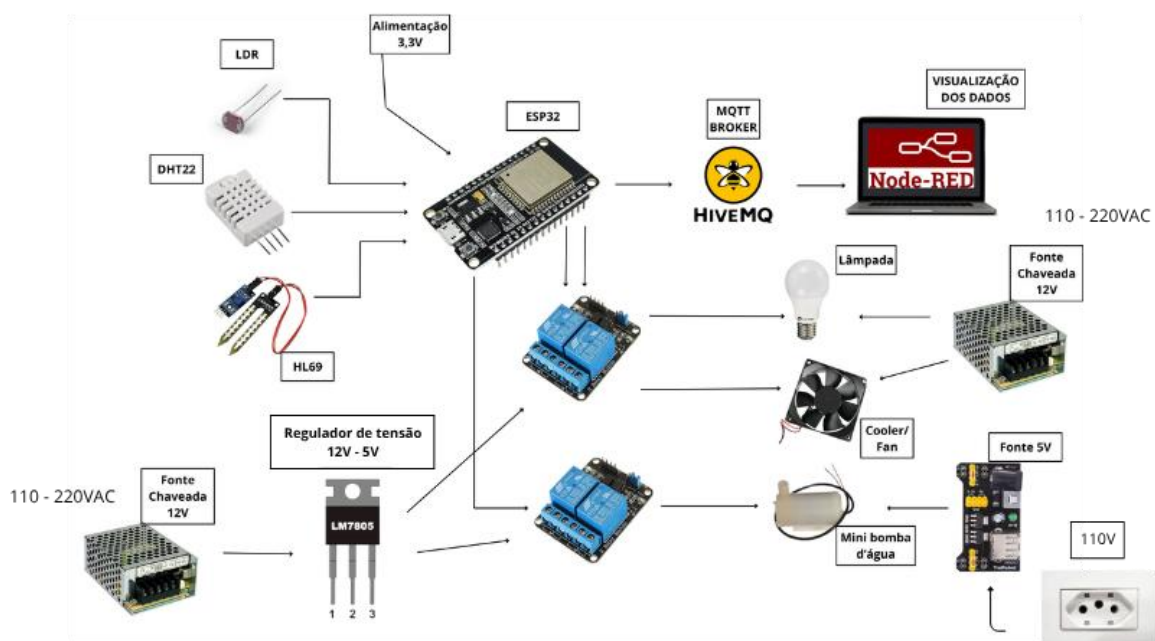
5. DESENVOLVIMENTO DO PROJETO DE CONTROLE E MONITORAMENTO DA ESTUFA

Após a apresentação dos conceitos e das ferramentas necessárias para a execução do projeto de controle e monitoramento em tempo real de uma estufa, visando o cultivo de hortaliças, neste tópico são mostrados os procedimentos de desenvolvimento, iniciados pelo diagrama esquemático ilustrativo da metodologia implementada e o seu funcionamento, o fluxograma do programa e a interface com o computador para a visualização dos dados.

5.1 Diagrama esquemático ilustrativo do projeto

O projeto de automação foi montado conforme o diagrama esquemático ilustrativo do projeto, apresentado na Figura 25. Este projeto foi desenvolvido em laboratório, em ambiente simulado de pequena escala e usando os parâmetros de interesse de uma estufa real. Nesta automação usa-se o conceito de IoT e para isso foi utilizado o ESP 32 que possui *wifi* embutido, fazendo a coleta de dados remotamente, e enviando para um notebook. A tela do notebook deve proporcionar a visualização dos dados, via Node-red.

Figura 25 – Diagrama esquemático ilustrativo do projeto



Fonte: Própria do autor.

Inicialmente, a partir deste diagrama esquemático que traduz a metodologia utilizada no projeto, com o protótipo são adquiridos os dados do sensor de luminosidade, o fotodetector, LDR, medido em uma resistência; o sensor de temperatura e a umidade do ar, DHT22 e a umidade do solo, sensor HL69, todos esses parâmetros são saídas de tensões analógicas, que vão para as entradas analógicas do ESP32, com valores medidos na faixa de 0 a 3,3 V. A conectividade ao *wifi* é necessária para implementar a coleta de dados e mostrar na tela do computador, uma vez que o ESP32 envia esses dados para um *broker* MQTT, (conforme mostrado no diagrama esquemático) para que possam ser visualizados em formato da *dashboard* do Node-Red.

Com estas informações o ESP32 pode acionar os atuadores se necessário, publicando os dados nos tópicos via MQTT para o *broker* HiveMQ, armazenando estes dados na nuvem. Na plataforma Node-Red os nós estão montados de forma a se inscreverem nos tópicos do *broker* e montar na *dashboard* de forma que o usuário possa visualizar os gráficos gerados de luminosidade, umidade do ar, temperatura e a umidade do solo.

No lado dos atuadores tem-se a lâmpada e um mini ventilador acionados pelo ESP32 usando relés de dois canais para proteção do ESP32. Os dois (lâmpada e Mini ventilador) têm alimentação de 12V, dada pela fonte chaveada de 12V (110-220VAC), tipo Colmeia. Apesar da bomba d'água operar em 3,3V não é conveniente ser ligada diretamente ao ESP32, por causa da corrente. O sensor de umidade do solo tem saída digital, em nível alto, indicando solo seco e ao contrário, úmido, em nível baixo. O sistema deve funcionar como segue:

- Quando há um aumento de temperatura no local, um valor pré-estabelecido como ideal para o cultivo de determinada hortaliça, o ventilador é acionado pelo comando do Node-Red usando a plataforma ESP32 que por sua vez, aciona o ventilador;
- No caso da lâmpada, só será ligada quando a temperatura baixar muito, de forma a evitar o frio intenso e as geadas, que estragam as hortaliças;
- Para o sensor de luminosidade, é deixado fora da estufa para observar quando o tempo escurece, portanto acendendo a lâmpada a noite por exemplo, quando pode-se acelerar o crescimento das plantas ;

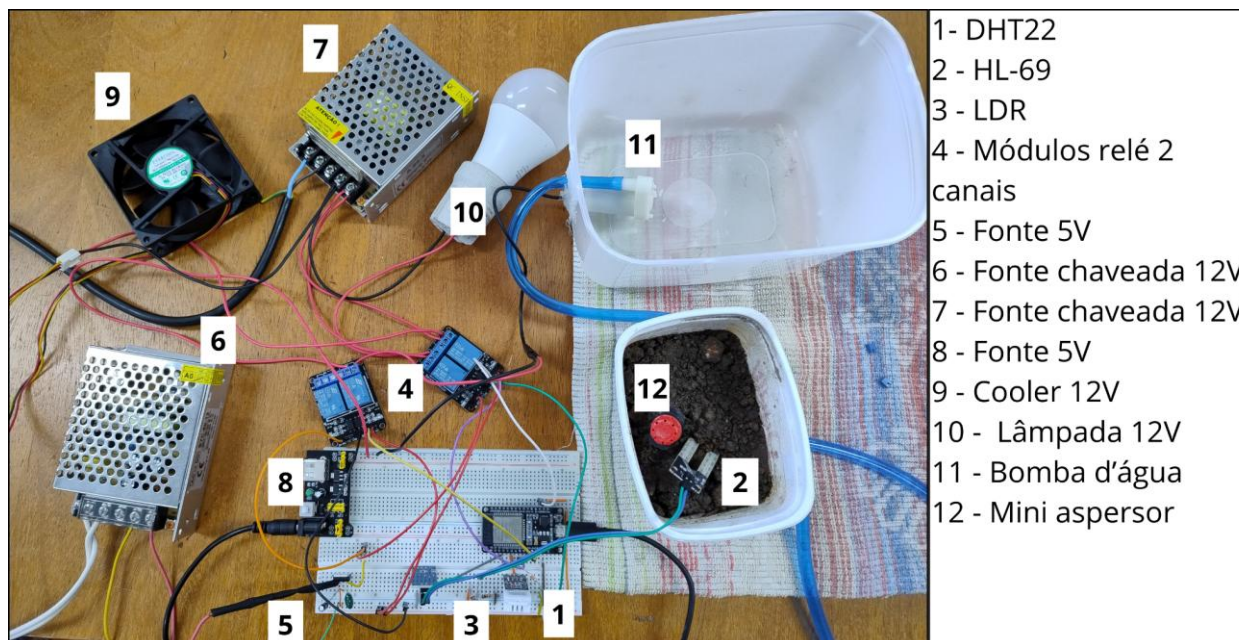
- Finalmente, se o sensor de umidade acusar solo seco, a bomba d'água é acionada pelo ES32 para solo seco (nível alto), caso contrário para de mandar água.

A criação do protótipo tem a finalidade de facilitar a manutenção do ambiente dentro das estufas que podem ser facilmente monitoradas e controlado os seus parâmetros, visando favorecer o desenvolvimento adequado da cultura e de seu manejo.

5.2 Montagem do projeto

Seguindo o esquema apresentado, foi montado o hardware em laboratório, utilizando somente *protoboard*, conforme mostrado na Figura 26 e com os componentes descritos anteriormente em fundamentação teórica. A montagem que aparece na figura teve seus componentes numerados visando facilitar o entendimento.

Figura 26 – Montagem do projeto

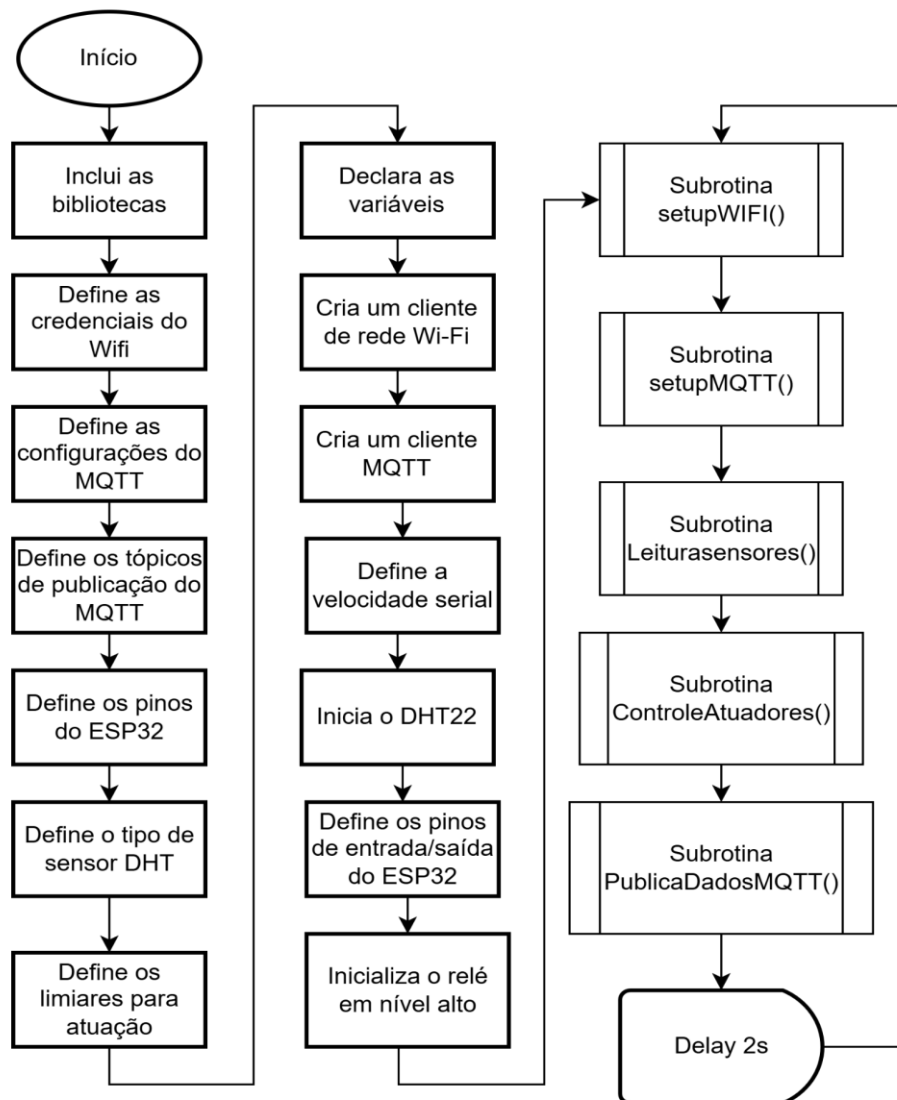


Fonte: Própria do autor.

Na Figura 27 é apresentado o circuito elétrico, desenhado no Proteus (PROTEUS DESIGN SUITE, 2022), software composto do Módulo Ares e o Módulo Isis, para o desenvolvimento de placas PCB (Printed Circuit Board, ou Placa de Circuito

Também foi necessário definir as variáveis inteiras e reais nos programas, os pinos analógicos e digitais utilizados no ESP32 e algumas outras especificações importantes para o funcionamento correto do ESP32. Na Figura 28 apresenta-se o fluxograma do programa principal para o sistema da estufa.

Figura 28 – Fluxograma principal com a programação do sistema da estufa



Fonte: Própria do autor.

O fluxograma criado para o controle e monitoramento da estufa, cinco sub-rotinas. Inicia pela inclusão das bibliotecas do componente DHT22, credenciais do *wifi*, configurações do MQTT - PubSubClient, que permite a conexão MQTT, como ID, *broker*

e porta de atuação. O broker utilizado foi o HiveMQ.m e a coleta de dados. Os limites (limiares) são para gerenciar os atuadores.

De forma que o mini-ventilador não fique ligando e desligando a toda hora, há um limite na faixa de atuação, ou seja, o estado (Ligado/Desligado) do atuador apenas mudará se a temperatura cair abaixo de 23°C ou se subir mais que 25°C, valores esses que podem ser mudados conforme a necessidade do usuário ou do cultivo que está na estufa. As outras variáveis que guardam os dados lidos pelos sensores, também foram declaradas e os clientes wifi e MQTT foram criados para que o ESP32 tenha acesso ao *wifi* e MQTT, respectivamente.

Em seguida, dentro do “void setup”, mostrado na Figura 29 (executado apenas uma vez, quando o sistema é iniciado) que consta em todos os programas do Arduino, é definida a velocidade serial do ESP32 e inicia o DHT22, Os pinos utilizados que foram declarados anteriormente são definidos como saída ou entrada e os pinos de saída são colocados no nível alto de forma que o relé seja iniciado. Isso porque o módulo relé funciona “ao contrário” do que seria intuitivo, quando o sinal é nível lógico alto o relé fica no estado normal, onde o NF está fechado e o NA está aberto. Quando ele recebe um sinal de nível lógico baixo, o NF abre e o NA fecha, assim acionando os atuadores (que estão conectados ao NA).

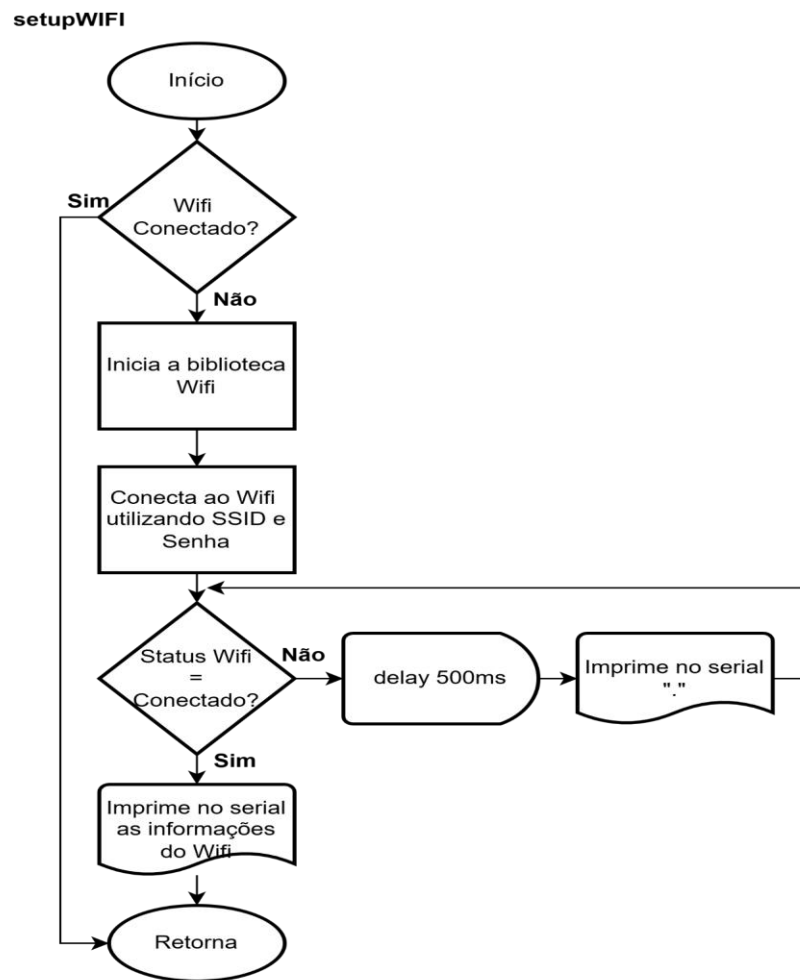
Figura 29 – Trecho “void setup”

```
void setup()
{
    Serial.begin(9600);
    dht.begin();
    pinMode(rele, OUTPUT);
    digitalWrite(rele, HIGH);
    ethernet.setup(mac, ip, port);
}
```

Fonte: Própria do autor.

Para auxiliar no entendimento o programa principal contém 5 subrotinas , que são: sub-rotina setupWIFI(), setupMQTT(), Leitura sensores(), Controleatuadores() e Publicadados MQTT(). O programa principal chama primeiro a subrotina setupWIFI, que é responsável por se conectar ao *wifi*, cujo fluxograma é apresentado na Figura 30.

Figura 30 – Fluxograma da subrotina setupWifi()

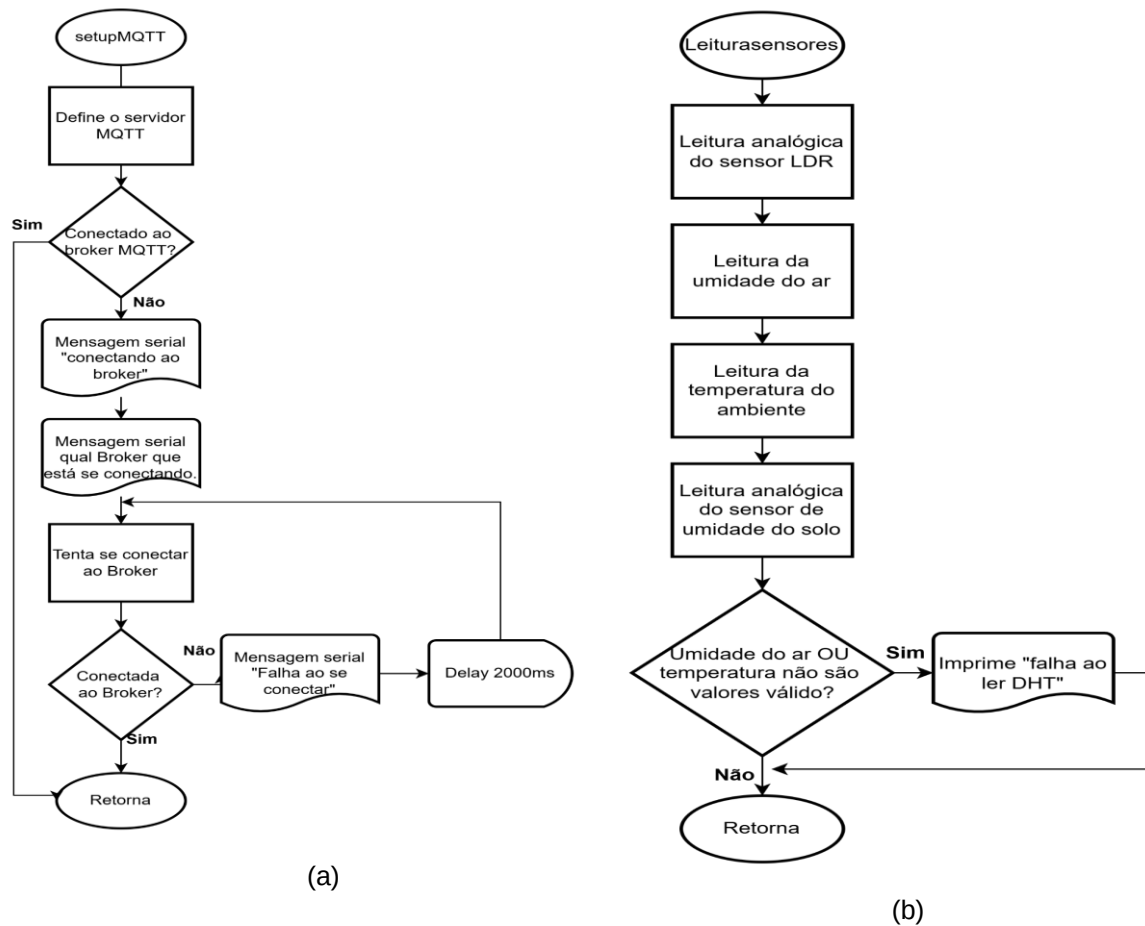


Fonte: Própria do autor.

Se não houver uma conexão *wifi*, o programa entra em um loop, de forma que não avance, enquanto não houver uma conexão. Se houver uma conexão *wifi*, o programa vai para a próxima etapa, chamando a subrotina, `setupMQTT()`, e se conecta ao *broker* HiveMQ, . o fluxograma da subrotina `setupMQTT()` é apresentado na Figura 31(a). Essa subrotina tem o funcionamento semelhante a anterior, onde é verificado se há uma conexão com o broker MQTT, se houver, o programa avança para o próximo passo e se não houver, entra em um loop até ser estabelecida uma conexão.

O próximo passo é a aquisição de dados pelos sensores, subrotina `Leiturasensores()`, nela, os valores recebidos pelos pinos do ESP32 são armazenados nas suas respectivas variáveis, cujo fluxograma é mostrado na Figura 30 (b).

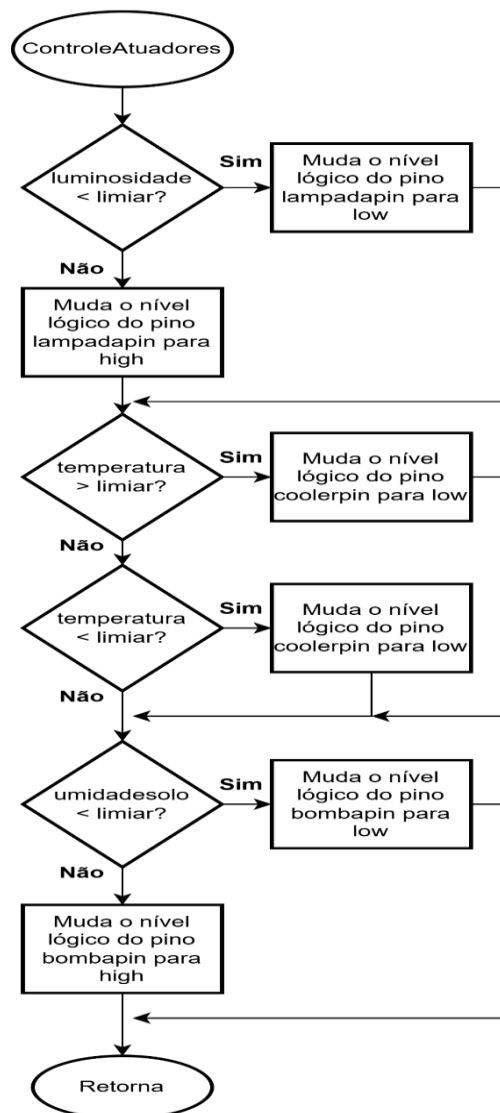
Figura 31 – Fluxogramas: a) Subrotina setupMQTT().b) Subrotina Leiturasensores().



Fonte: Própria do autor.

Por fim, tem-se a última subrotina, ControleAtuadores, cujo fluxograma é apresentado na Figura 32.

Figura 32 – Fluxograma da subrotina ControleAtuadores



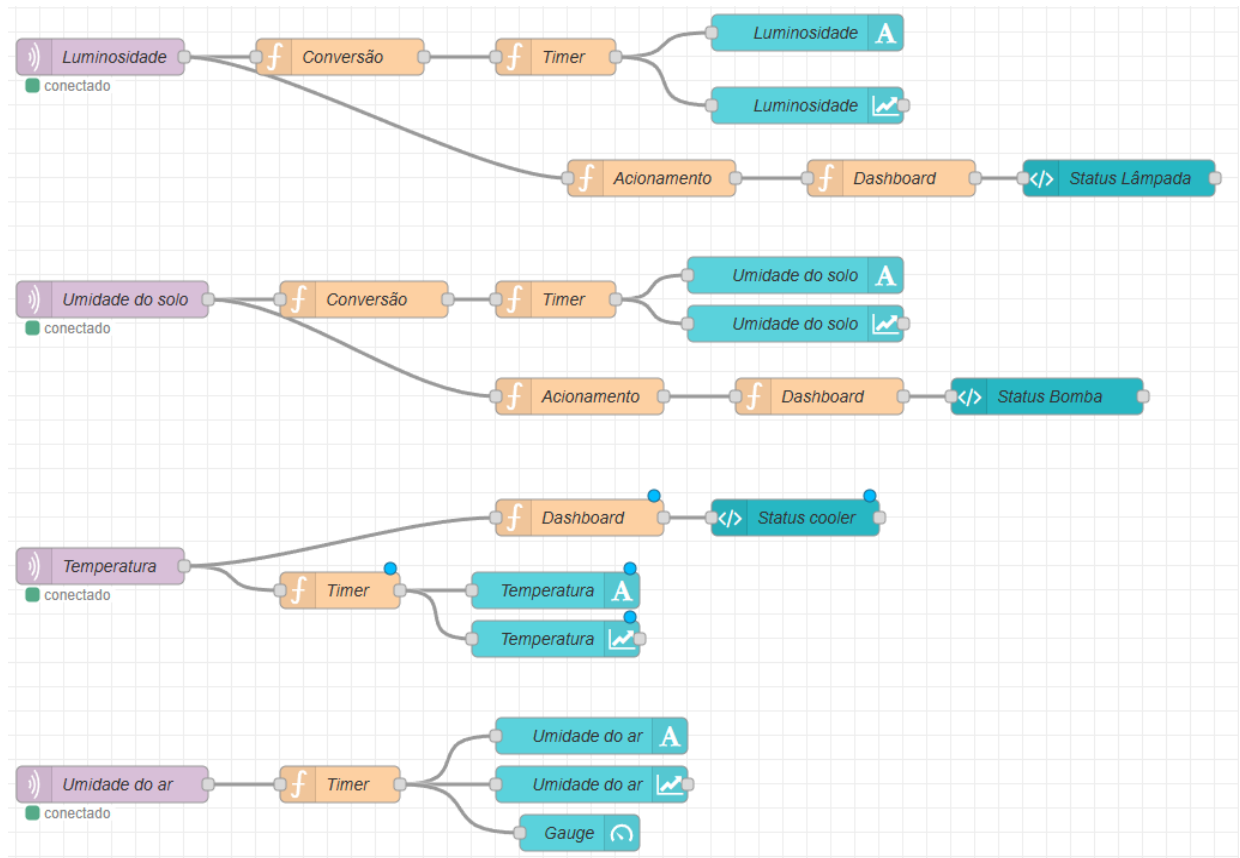
Fonte: Própria do autor.

Nesta etapa do programa, os valores obtidos dos sensores são comparados com os limiares definidos no início, de forma que, a depender dos dados ele aciona (com nível lógico baixo - *low*) ou desliga (com nível lógico alto - *high*) os atuadores. Por fim, é chamada a subrotina PublicaDadosMQTT, que tem como objetivo publicar os dados obtidos nos tópicos MQTT para serem acessados no Node-red. O *delay* de dois segundos, adicionado ao final é o tempo necessário para que o DHT22 possa realizar outra leitura.

5.4 Plataforma do Node-Red

Neste item, apresenta-se o fluxo de configuração do *dashboard* do sistema da estufa criado, Figura 33.

Figura 33 – Fluxo de configuração do Node-Red.



Fonte: Própria do autor.

As cores diferentes dos nós se referem à diferentes funções, sendo o roxo comunicação/conexão, amarelos, funções e os azuis de visualização no *dashboard*. Existem quatro diferentes tipos de nós de função, a começar com os nós de “conversão”, os valores recebidos via MQTT são enviados sem nenhum tratamento, assim, como os sensores com saídas analógicas, o ESP32 faz a conversão (Analog-Digital Converter-ADC) com resolução de 12 bits (escala de 0 a 4096), portanto os valores recebidos pelo Node-red aparecem nessa escala, ou seja, os nós de conversão estão entre 0 -100%. Na Figura 34 para esclarecer melhor, mostra-se o conteúdo do nó de conversão.

Figura 34 – Programa criado para o *nó de conversão*.

```

1  // Defina os limites mínimo e máximo da escala original
2  let inputMin = 0;
3  let inputMax = 4095;
4
5  // Defina os limites da nova escala (0 a 100)
6  let outputMin = 0;
7  let outputMax = 100;
8
9  let value = Number(msg.payload);
10
11 let scaledValue = (value * outputMax) / inputMax ;
12 scaledValue = Math.max(outputMin, Math.min(outputMax, scaledValue));
13 scaledValue = parseFloat(scaledValue.toFixed(2));
14 msg.payload = scaledValue;
15
16 return msg;

```

Fonte: Própria do autor.

No início do programa são definidos os limites de cada escala, que o valor recebido é numérico e processa a transformação do valor conforme a Equação 1,

$$Y = Y_{min} + \frac{(X - X_{min}) * (Y_{max} - Y_{min})}{X_{max} - X_{min}} \quad (1)$$

sendo X, o valor recebido no Node-red; Y é o valor na escala nova; Ymin e Ymax são os valores mínimos e máximos da escala nova ; Xmin e Xmax são os valores mínimos e máximos da escala antiga.

Simplificando a equação (1) com os valores dados no programa, tem-se a Equação (2),

$$Y = \frac{X * 100}{4095} \quad (2)$$

O *nó de timer* atua como um cronômetro que envia os dados para a *dashboard* a cada intervalo de tempo. O timer foi colocado nesta etapa, para ser adicionado delays muito grandes no código do ESP32, pois pode interferir no funcionamento do programa. Além disso, o *timer* filtra as amostras, de forma que os gráficos não registrem todas as variações que chegam ao Node-red , uma vez que os resultados das medidas podem ter um espaçamento maior, considerando que são grandezas físicas que não se alteram

tão abruptamente. Caso o usuário ache necessário, o valor de timer pode ser alterado, na Figura 35 mostra-se o programa *nó de timer*.

Figura 35 – Programa criado para o *nó de timer*.

```

1  // Obtém o timestamp atual
2  let now = new Date().getTime();
3
4  // Recupera o último tempo salvo (se existir)
5  let lastMsg = context.get("lastMsg") || 0;
6
7  // Define o intervalo de tempo (10 minutos = 600000 ms)
8  let interval = 600000;
9
10 if (now - lastMsg > interval) {
11     context.set("lastMsg", now);
12     return msg;
13 }
14 return null;

```

Fonte: Própria do autor.

Nesta parte do programa o *timestamp* (data e hora) atual é e o último tempo salvo, o intervalo de tempo é definido e por meio de comparação, é verificado se já passou o tempo necessário. Se sim, retorna a mensagem para publicação, se não passou o tempo necessário, descarta a mensagem.

Outro nó importante é o *nó de acionamento* que foi utilizado para apontar, na *dashboard*, o status dos atuadores. O valor recebido dos atuadores é comparado com os limiares definidos (têm que ser os mesmos do programa do ESP32 para que não haja erros) e então retorna a mensagem conforme o estado do atuador. O programa é mostrado na Figura 36 (a). No nó de *dashboard* é adicionada a imagem do status do atuador e definida a cor, conforme mostrado na Figura 36 (b).

Figura 36 –a) Nó de Atuador. b) Nó de dashboard.

<pre> 1 // Define o valor de referência para comparação 2 let limite = 2047; 3 4 let valor = parseFloat(msg.payload); 5 6 // Faz a comparação e define o status 7 if (valor < limite) { 8 msg.payload = "1"; // Acionado 9 } else { 10 msg.payload = "2"; // Desligado 11 } 12 13 return msg; </pre> (a)	<pre> 1 const value = msg.payload 2 let icon = "" 3 let color = "" 4 5 if (value === 1) { 6 icon = "engine" 7 color = "green" 8 } 9 10 else { 11 icon = "engine-off" 12 color = "red" 13 } 14 15 msg.payload = { 16 "icone": icon, 17 "cor": color 18 } 19 return msg; </pre> (b)
---	--

Fonte: Própria do autor.

6. RESULTADOS E DISCUSSÕES

Após apresentada a metodologia utilizada para montagem do sistema de monitoramento e controle de uma estufa e a sua programação, neste tópico são apresentados os resultados dos testes individuais para cada sensor e o software implementado no ESP32 junto ao Node-red os resultados finais obtidos pelo sistema

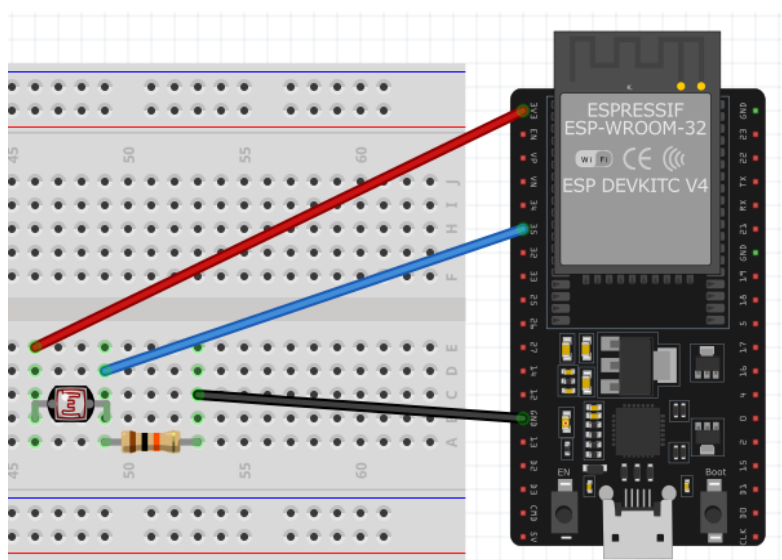
6.1 Testes dos sensores

Inicia-se a apresentação dos testes individuais dos sensores pelo sensor LDR

6.1.1 Teste para o sensor LDR

Apesar de existirem outros sensores que têm propriedades semelhantes, optou-se pelo LDR que é de fácil manuseio e baixo custo, adequado para este trabalho, O circuito montado para realizar os testes com o sensor LDR é apresentado na Figura 37, trata-se de um divisor de tensão, sendo a tensão de saída do circuito medida na resistência R, e utilizando o ESP32.

Figura 37 – Esquemático para os testes com o LDR e o ESP32.



Fonte: Própria do autor

O LDR é ligado em série com o resistor, R e o sinal da tensão de saída está conectado ao pino 35 (fio azul) do ESP32, sendo VCC (fio vermelho) e o GND (fio preto). Os valores obtidos dependem da tensão que varia de 0V (um ambiente muito escuro) a

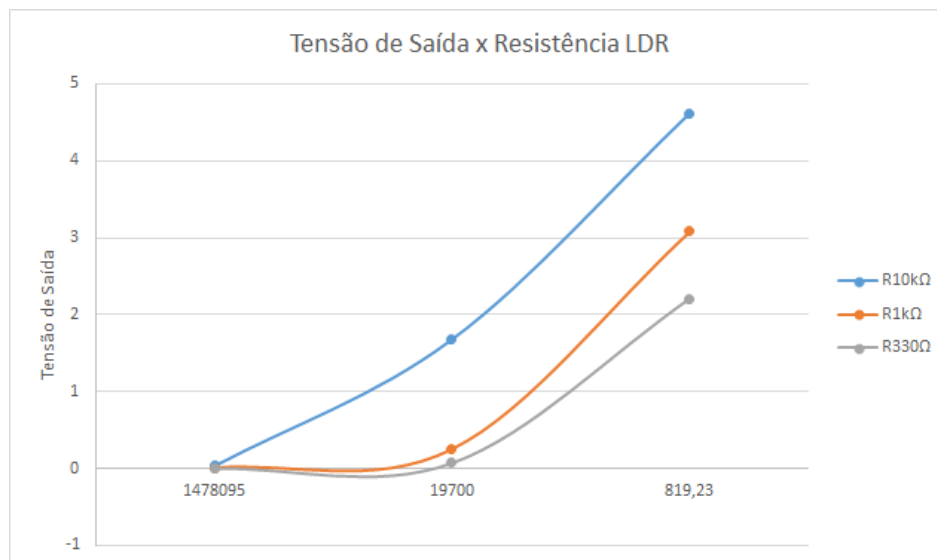
3,3V (um ambiente bem iluminado). O ESP32 converte esse valor de tensão equivalente, com resolução de 12bits, ou seja, fornece valores entre 0 e 4095.

Considerando o divisor de tensão e os resistores $R = 10k\Omega$ em série com o sensor LDR, com resistência dada por R_{LDR} e a alimentação de 5 VCC, a tensão de saída no resistor, $V_{saída}$ é dada pela Equação (3),

$$V_{saída} = \frac{R}{R_{LDR} + R} * V_{cc} \quad (3)$$

Foram realizadas quatro condições de testes para a luminosidade, ambiente escuro, luz ambiente, luz direta forte e mudanças graduais durante o dia e três valores distintos de resistor, R , no divisor de tensão, 330Ω , $1k\Omega$ e $10k\Omega$. Para cada resistência do LDR obteve-se o comportamento observado no gráfico da Figura 38, sendo verificado o comportamento aproximadamente linear para o $R = 10k\Omega$, muito maior que o valor da resistência do LDR, portanto adequado ao projeto e com valor de tensão maior (próximas da escala de operação do ADC, resolução de 12 bits).

Figura 38 – Gráfico tensão de saída (V) x Resistência do LDR(Ω).

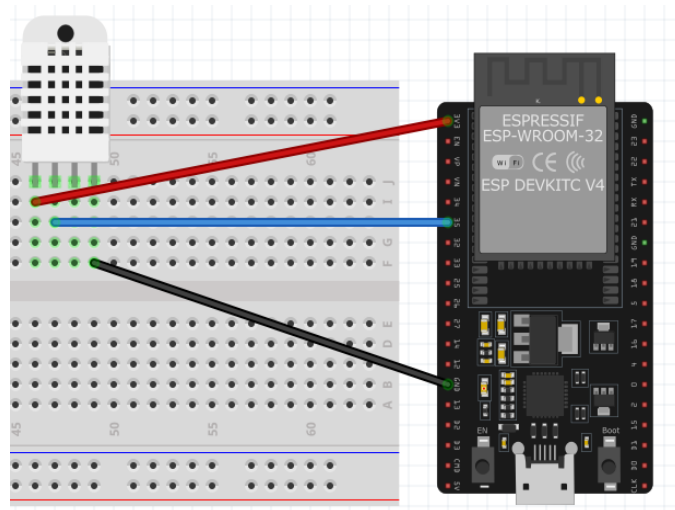


Fonte: Própria do autor.

6.1.2 Teste para o sensor DHT22

Para os testes do sensor que mede temperatura e umidade, DHT22 foi ligado diretamente ao ESP32 com a alimentação de tensão em 3,3V(fio vermelho), e GND(fio preto) , e a saída do sinal (fio azul) conforme ilustrado na Figura 39.

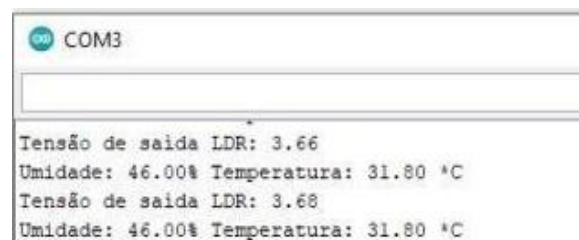
Figura 39 – Circuito de teste para o DHT22.



Fonte: Própria do autor.

Os testes foram realizados para a variação de temperatura ao longo de um dia inteiro e para a comparação dos resultados foi usado como referência às informações do site do Instituto Nacional de Meteorologia (INMET), onde disponibilizam os dados meteorológicos para consulta. Os valores obtidos de temperatura e umidade também foram comparados aos da Estação - Canal CLIMA da UNESP Ilha Solteira, que fica a cerca de 9km do laboratório onde os testes foram realizados. Na figura 40 é possível ver os resultados obtidos.

Figura 40- Resultados dos testes com o DHT22 em laboratório.



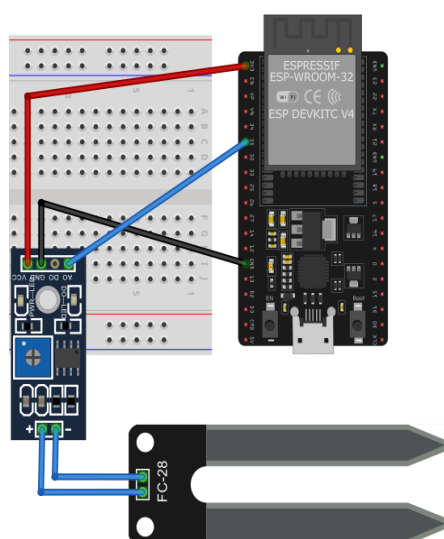
Fonte: Própria do autor.

O DHT22 possui dois sensores internos que fazem as aquisições de dados. Esses sensores geram sinais analógicos, que são convertidos por um conversor A/D interno em valores digitais de 16 bits cada. Esses dados são passados para o ESP32 por um único fio de comunicação serial para serem exibidos.

6.1.3 Teste para o sensor de umidade do solo HL- 69

O sensor HL- 69 foi conectado ao VCC=3,3V(fio vermelho) do ESP32, em GND (fio preto) e um pino analógico para o sinal (azul), conforme mostrado na Figura 41.

Figura 41: Circuito para o teste do HL69.



Fonte: Própria do autor.

Os testes foram efetuados em três situações diferentes, solo seco, depois em solo levemente molhado para verificar a mudança e por fim, o solo foi encharcado. Depois foi feito um teste ao longo do tempo em que foi deixado a terra secar naturalmente e os valores foram medidos ao longo deste tempo. Na Figura 42 pode-se ver o código do programa utilizado para realizar o teste do sensor.

Figura 42 – Programa para o sensor de umidade de solo

```

1  const int sensorPin = 34;
2
3  void setup() {
4      Serial.begin(9600);
5  }
6
7  void loop() {
8      int leitura = analogRead(sensorPin);
9
10     int umidadePercentual = map(leitura, 1023, 0, 0, 100);
11
12     Serial.print("Leitura: ");
13     Serial.print(leitura);
14     Serial.print(" | Umidade: ");
15     Serial.print(umidadePercentual);
16     Serial.println("%");
17
18     delay(1000);
19 }

```

Fonte: Própria do autor.

6.1.4 Testes da bomba d'água JT100

O primeiro teste na bomba d'água foi para determinar a sua vazão. Para isso, foram feitos dois testes, a bomba foi ligada na fonte de alimentação de 3,3V e depois em 5V, neste caso usando um Arduino UNO R3 e o tempo que ela levava para encher 400ml de um recipiente foi cronometrado, conforme apresentado na Figura 43.

Figura 43 – Teste de vazão da bomba d'água.



Fonte: Própria do autor.

Foram coletadas 10 amostras de tempo com a bomba alimentada em cada uma das tensões, 3,3V e 5V , os resultados obtidos são apresentados na Quadro 10.

Quadro 10 – Resultados das medidas e os tempos na bomba d’água.

3,3V			5V	
Medida	Tempo (s)		Medida	Tempo (s)
1	35,22		1	24,63
2	35,51		2	25,42
3	34,32		3	26,02
4	33,24		4	25,76
5	35,06		5	25,44
6	35,07		6	24,95
7	35,56		7	25,47
8	34,55		8	25,34
9	35,82		9	25,55
10	34,58		10	26,16
Valor médio	35,07		Valor médio	25,46

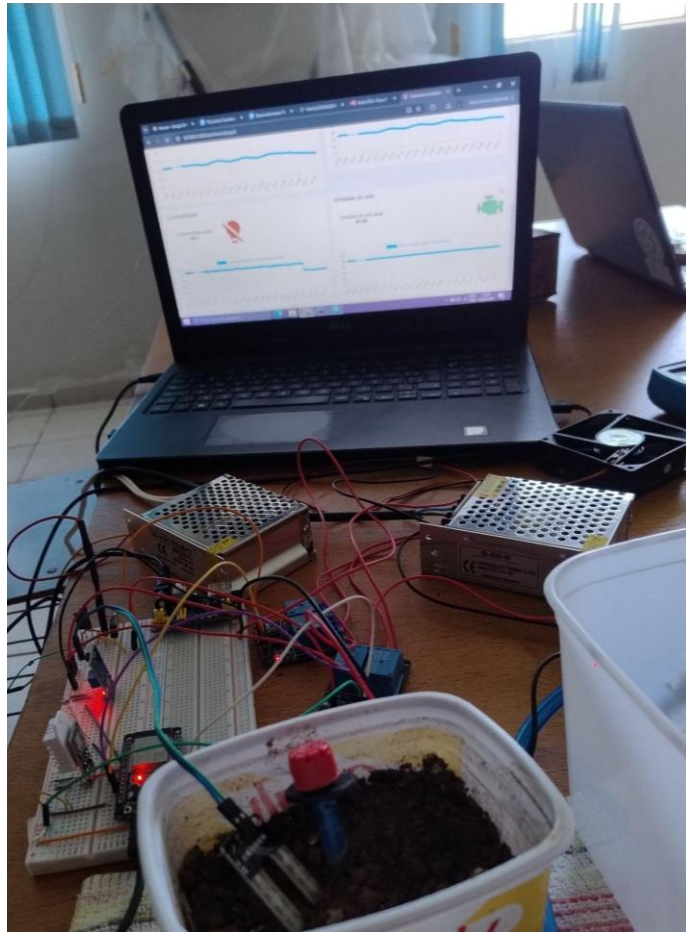
Fonte: Própria do autor.

Assim, para descobrir a vazão por minuto, tem-se o tempo para encher 400ml, portanto, basta fazer uma regra de três e dessa forma, os resultados foram 0,68 l/min em 3,3V e 0,94 L/min em 5V. Portanto, os valores reais são próximos aos valores teóricos apresentados no capítulo 3 e neste projeto a mini bomba d’água é utilizada em 5V.

6.2 Teste final para o hardware completo

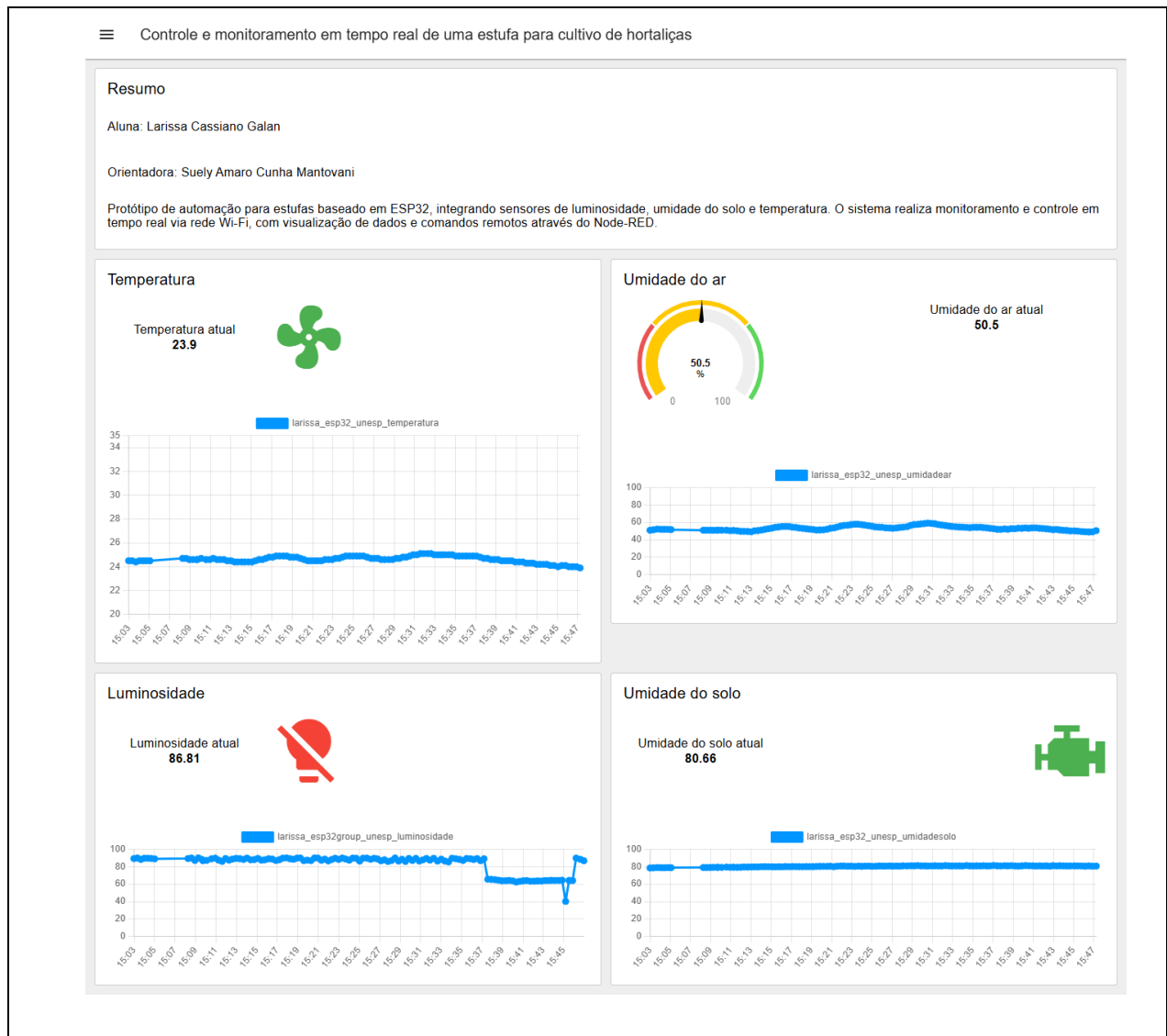
Os testes finais foram realizados no sistema de monitoramento e controle da estufa mostrado na Figura 44. Neste estão conectados todos os sensores e atuadores suas respectivas fontes de alimentação, ligados ao ESP32 que por sua vez está ligado ao notebook que mostra em sua tela os resultados , por meio da plataforma Node-red. Em seguida foram realizados os testes para verificar o funcionamento de todas as variáveis e do programa. A interface da *dashboard* no Node-Red é mostrada nas Figuras 45 e 46.

Figura 44 – Sistema de monitoramento e controle de uma estufa.



Fonte: Própria do autor.

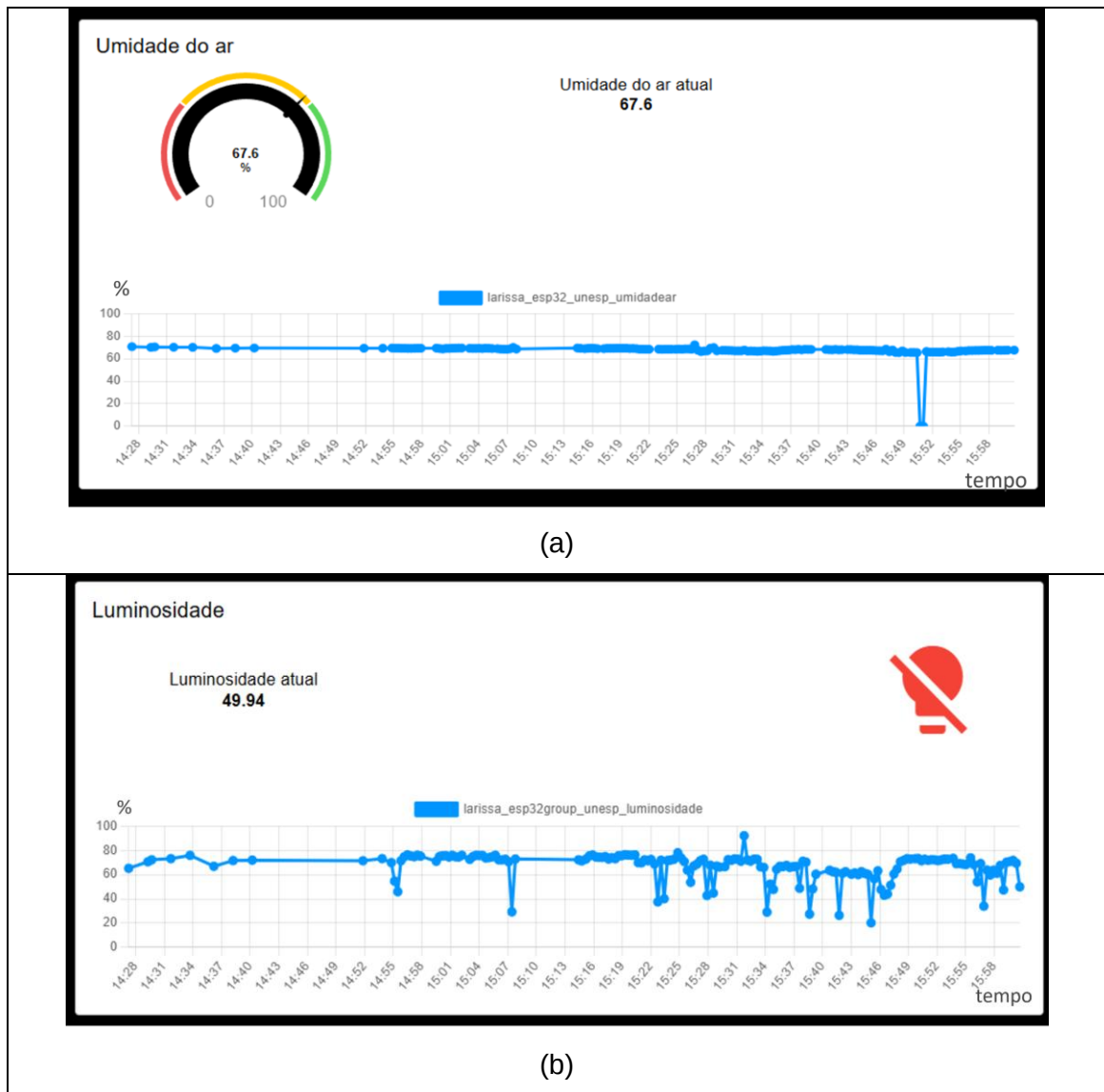
Figura 45 - Interface da *dashboard*: a) Página inteira; b) Gráfico de Umidade do ar; c) Gráfico de luminosidade; d) Gráfico de Temperatura. e) Gráfico de Umidade do solo.

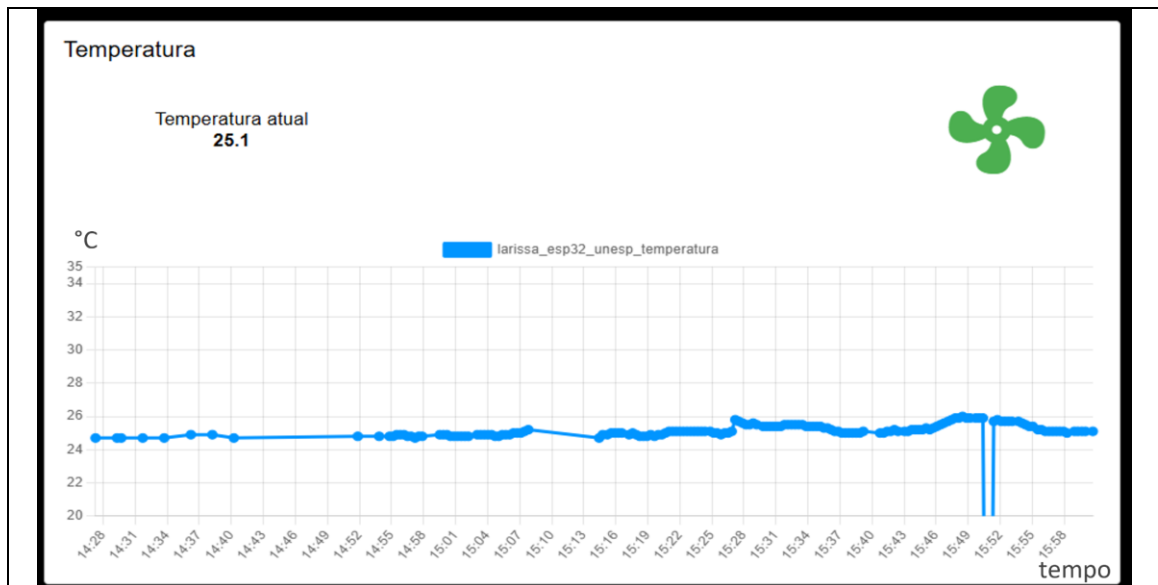


Fonte: Própria do autor.

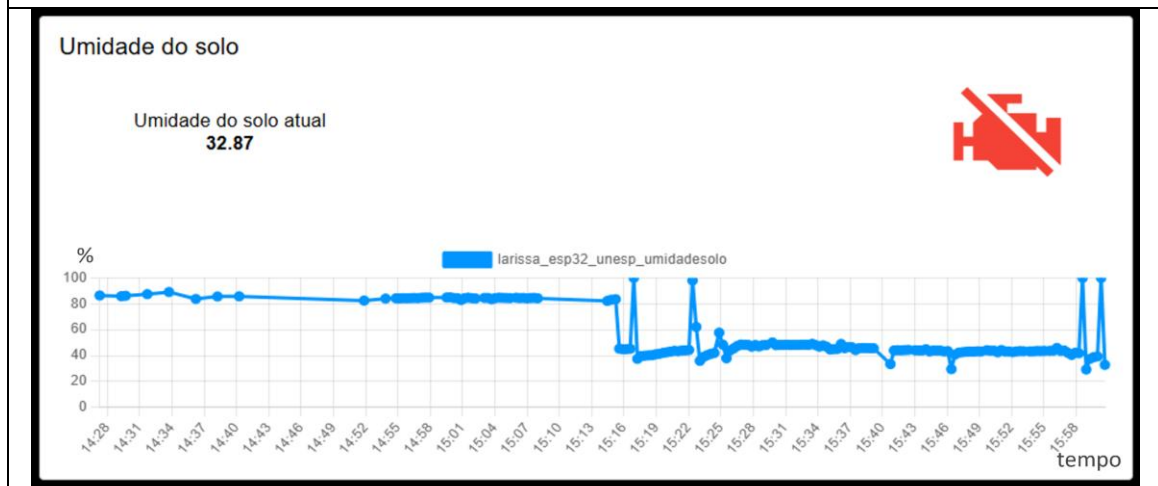
Nesta figura são observados os resultados na *dashboard* para os atuadores, pois, de acordo com o valor recebido do sensor de temperatura, estava mais de 25°C o mini ventilador foi ligado (em verde), a bomba d'água também detectou solo seco, mas a lâmpada estava desligada pois a luminosidade do ambiente estava dentro dos parâmetros determinados no programa.

Figura 46 - Interface da *dashboard*: a) Gráfico da Umidade do ar; b) Gráfico de luminosidade; c) Gráfico da Temperatura. d) Gráfico de Umidade do solo.





(c)



(d)

Fonte: Própria do autor.

Durante os testes não houve muitas variações nos gráficos, como mostrado na figura com os gráficos na *dashboard*, pois os testes foram efetuados dentro de um ambiente controlado, o laboratório. O funcionamento do sistema foi testado variando as condições locais de forma a mostrar a resposta dos sensores e dos atuadores. Para obter a variação da luminosidade foi utilizada a lanterna do celular como fonte de grande incidência de luz e ao contrário, simulando baixa ou pouca luz. Para variar a temperatura ambiente, foi ligado o ar-condicionado, de forma que quando era necessário diminuir a temperatura, o ar-condicionado era ativado em temperatura baixa e vice versa, para aumentar a temperatura, era desligado. O processo de mudança de temperatura é mais lento e com o auxílio do gráfico é possível captar essas variações ao longo do tempo.

Por fim, para verificar a umidade do solo, o sensor de umidade do solo era retirado do solo ou colocado novamente.

Ressalta-se que com este projeto obteve-se os resultados esperados, ou seja, todos os sensores funcionando , assim como seus respectivos atuadores. Foram usados dispositivos atuais e conceitos com objetivo de proporcionar o domínio dessas tecnologias. Para viabilizar a realização do projeto, todo o material foi adquirido pelo próprio aluno, que inicialmente desenvolveu o projeto na sua casa. A lista de componentes utilizados é apresentada no **Apêndice A**.

7. CONCLUSÃO

Nesta pesquisa foi desenvolvido um projeto de controle e monitoramento em tempo real, de uma estufa, para o cultivo de hortaliças. Para isso foram utilizados sensores para medir luminosidade, temperatura, umidade do ar e do solo. Com essas medidas pode-se monitorar as condições ideais de um ambiente, remotamente, por meio do conceito de IoT e novas tecnologias de hardware e software, de maneira a obter um aumento na produtividade em qualquer cultura que seja necessário o cultivo abrigado. Com estes parâmetros torna-se possível atuar no sistema utilizando componentes de fácil manuseio e de baixo custo, e proporcionar boas condições para o desenvolvimento da planta, com economia de água, energia e mão de obra, em outras palavras uma agricultura sustentável.

A motivação em criar este projeto se baseou na literatura que apresenta as estufas como solução e alternativa importante para que os agricultores desenvolvam suas culturas com proteção contra agentes externos, amenizando problemas de variação climática que acometem algumas regiões do Brasil. Além disso, aliam-se a isso, as novas tecnologias que facilitam o controle e o monitoramento dos parâmetros, por meio de sensores e plataformas de hardware e software com grandes recursos.

Foram realizados testes nos sensores utilizados visando à calibração e o levantamento da sua curva de funcionamento. A partir de uma determinada medida da temperatura foi simulado a atuação do mini ventilador, ligando /desligando quando necessário. A mesma simulação foi feita para a luminosidade para ser observada a atuação na lâmpada, também ligando e desligando para uma medida de luminosidade arbitrária, especificada conforme o cultivo. Os mesmos testes foram feitos para o caso da umidade do ar e do solo e o uso da bomba d'água com o aspersor para irrigação.

O protocolo de comunicação MQTT que faz a comunicação do ESP32 com o Node-Red, foi utilizado para permitir a visualização dos parâmetros do sistema pelo produtor, em um computador e/ou smartphones. Ao acessar a página de monitoramento no Node-Red o usuário obtém os valores em tempo real, de cada sensor e consegue tomar as decisões e fazer os ajustes necessários para a sua plantação, tais como, adubo se tiver faltando, abastecimento da caixa d'água se não estiver cheio o tanque, etc.

É importante destacar que soluções profissionais de automação de estufas disponíveis no mercado podem demandar investimentos elevados, variando entre R\$

15.000,00 a mais de R\$ 60.000,00, dependendo da complexidade e das tecnologias envolvidas. Frente a esse cenário, o projeto desenvolvido nesta representa uma alternativa acessível, especialmente para pequenos produtores e que empregam tecnologias de automação e conceitos de IoT de forma simplificada e com bom custo-benefício.

Nesta pesquisa procurou-se abordar um sistema geral de monitoramento e controle de estufas sem se importar com uma planta específica que se desenvolve bem em determinada temperatura e umidade, mas entende-se que o sistema proposto possa ser adequado com facilidade às especificações de qualquer cultura. Dos resultados obtidos conclui-se que o sistema atende aos objetivos esperados.

Neste trabalho, além de todo o conhecimento adquirido, o seu desenvolvimento proporcionou ao aluno uma publicação de um resumo no XXXIII Congresso de Iniciação Científica - CIC da UNESP, apresentado no **Apêndice B**.

Como proposta de melhoramento e sugestões para futuros trabalhos pode-se criar uma base de dados com a finalidade de fornecer informações para o pequeno agricultor programar seus plantios e colheitas. Outra sugestão é a confecção de uma placa em circuito impresso para tornar o sistema um produto comercializável.

REFERÊNCIAS BIBLIOGRÁFICAS

AGROADVANCE (org.). IoT na agricultura: 5 aplicações de internet das coisas que podem aumentar a rentabilidade agrícola.. 2024. Elaborado por Mariana Colli. Disponível em: <https://agroadvance.com.br/blog-iot-na-agricultura/>. Acesso em: 07 mar. 2025.

ALBUQUERQUE, PAIVA DE OLIVEIRA, R.; BASTOS FILHO, C. Uma Arquitetura de Microserviços de Internet das Coisas para Casas Inteligentes. **Revista de Engenharia e Pesquisa Aplicada**, v. 2, n. 2, 27 jul. 2017.

ALVES, F. Fonte chaveada: como funciona. **Athos Electronics**, 15 Junho 2018. Disponível em: <https://athoselectronics.com/fonte-chaveada/>. Acesso em: 20 mai 2022.

ALVES, Jeferson CURTOCIRCUITO (blog). **Como usar a Bomba de Água Diafragma**. 2023. Disponível em: <https://curtocircuito.com.br/blog/post/como-usar-a-bomba-de-agua-diafragma>. Acesso em: 02 fev. 2025

ARMIJOS, Víctor Asanza. **Especificaciones del módulo ESP32**. Disponível em: <https://vasanza.blogspot.com/2021/07/especificaciones-del-modulo-esp32.html>. Acesso em: 10 mar. 2025.

BASÍLIO, Shirley. **Protocolo MQTT**: publish/subscriber (parte - 01). Publish/Subscriber (parte – 01). 2018. Publicado em Master Walkershop. Disponível em: <https://blogmasterwalkershop.com.br/embarcados/esp8266/protocolo-mqtt-publish-subscriber-parte-01>. Acesso em: 10 mar. 2025.

BRITO, D. C.; GOTTSCHALL, C. R.; CUNHA, R. T.; SILVA, W. L. **Monitoramento e controle de uma estufa para cultivo abrigado de tomates**. In: MOSTRA NACIONAL DE ROBÓTICA, 8., 2018, João Pessoa - PB. Anais 978-65-89682-02-8: MNR, 2018. v. 01, p. 661-665. Disponível em: <https://www.mnr.org.br/anais-da-8a-mostra-nacional-de-robotica-mnr-2018/>. Acesso em: 23 abr. 2022.

CELESTRINI, Jordano R.. **Desenvolvimento de aplicações integrando serviços web, fontes de dados e dispositivos IoT com o uso do Node-Red**. Espírito Santo: Ufes, 2021. 32 slides, color. Disponível em: <https://www.inf.ufes.br/~zegonc/material/Redes%20de%20Sensores%20sem%20Fio/Minicurso%20Node-RED.pdf>. Acesso em: 10 mar. 2025.

COLLI, Mariana. 2023. Disponível em: <https://agroadvance.com.br/blog-iot-na-agricultura/> Acesso em : 30 abr 2025.

DATASHEET PHOTOCONDUCTIVE CELLS. Disponível em: <https://cdn.sparkfun.com/datasheets/Sensors/LightImaging/SEN-09088.pdf>. Acesso em: 11 mar. 2025.

DHT22-AM2302-Datasheet.Disponível em: <https://www.makerhero.com/img/files/download/DHT22-AM2302-Datasheet.pdf> Acesso em : 10 abr de 2024.

ELETRONIC, Blikai. **ESP32 vs Arduino**:: an in-depth contrast. An In-Depth Contrast. 2024. Disponível em: <https://www-blikai-com.translate.goog/blog/featured->

products/esp32-vs-arduino-an-in-depth-contrast/blog/featured-products/esp32-vs-arduino-an-in-depth-contrast?_x_tr_sl=en&_x_tr_tl=pt&_x_tr_hl=pt&_x_tr_pto=tc&_x_tr_hist=true. Acesso em: 10 mar. 2025.

ELETRÔNICA, Sta (org.). **COMO UTILIZAR O MÓDULO RELÉ COM ARDUINO**. Disponível em: <https://www.sta-eletronica.com.br/artigos/arduinos/como-utilizar-o-modulo-rele-com-arduino>. Acesso em: 11 mar. 2025.

FERNANDES, D. G. **Sistema automatizado de controle de estufas para cultivo de hortaliças**. 2017. 32 f. TCC (Graduação) - Curso de Bacharelado em Sistemas de Informação, Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, Frederico Westphalen - RS, 2017. Disponível em: https://repositorio.ufsm.br/bitstream/handle/1/12958/TCCG_SIFW_2017_FERNANDES_DOUGLAS.pdf?sequence=1&isAllowed=y. Acesso em: 14 mai 2022.

FERNANDES, D. G.; PREUSS, E.; SILVA, T. L. Sistema automatizado de controle de estufas para cultivo de hortaliças. 2017. Trabalho de Conclusão de Curso (Graduação em sistemas de Informação) - Universidade Federal de Santa Maria, Frederico Westphalen, 2017. Disponível em: https://repositorio.ufsm.br/bitstream/handle/1/12958/TCCG_SI_FW_2017_FERNANDES_DOUGLAS.pdf?sequence=1&isAllowed=y. Acesso em: 14 mai. 2022

FURLANI, P.R.; SILVEIRA, L.C.P.; BOLONHEZI, D; FAQUIN, V. **Cultivo Hidropônico de Plantas**: Parte 1 - Conjunto hidráulico. 2009. Artigo em Hypertexto. Disponível em: <http://www.infobibos.com/Artigos/2009_1/hidroponiap1/index.htm>. Acesso em: 19 mai 2022.

GOMES, Flávio Henrique Ferreira *et al.* CALIBRAÇÃO DE UM SENSOR DE UMIDADE DO SOLO DE BAIXO CUSTO. **Revista Brasileira de Agricultura Irrigada**, Fortaleza, v. 11, n. 4, p. 1509-1516, 17 abr. 2017. Disponível em: https://www.researchgate.net/profile/Flavio-Henrique-Ferreira-Gomes/publication/318666453_CALIBRACAO_DE_UM_SENSOR_DE_UMIDADE_DO_SOLO_DE_BAIXO_CUSTO/links/5aec463eaca2727bc003fa1a/CALIBRACAO-DE-UM-SENSOR-DE-UMIDADE-DO-SOLO-DE-BAIXO-CUSTO.pdf. Acesso em: 11 mar. 2025.

GOMES, Lucas Matheus de Carvalho *et al.* ARDUINO: automatização da coleta de dados na irrigação. **Revista Conexão Eletrônica**, Três Lagoas, v. 13, n. 112, p. 2297-2312, 2016. Anual. Disponível em: <http://revistaconexao.aems.edu.br/wp-content/plugins/download-attachments/includes/download.php?id=993>. Acesso em: 19 maio 2022.

HANDSONTEC, 2025. Disponível em : www.handsontec.com. Acesso em : 20 mar 2025

HEXAGON. **Como a automação agrícola está mudando o agronegócio**. 2021. Disponível em: <https://hexagon.com/pt/resources/resource-library/how-farming-automation-changing-agriculture>. Acesso em: 08 abr. 2022.

HORTAVIVA. **Bico Gotejador Regulável 0-40 Litros/ Hora**. Disponível em: https://www.hortavivasementes.com.br/irrigacao/gotejadores/bico-gotejador-regulavel-0-40-litros-hora?parceiro=4549&gad_source=1&gbraid=0AAAAADlt8629XZ_F4Y2pCwhTiffHLOsS&

gclid=Cj0KCQjwoNzABhDbARIsALfY8VMQRcXffbW0WcC1DQZw8Oxb085qlfw6P3q7RcMZ2o1UijmVxcg-WvAaAgwrEALw_wcB. Acesso em: 04 mai 2025.

PROTEUS DESIGN SUITE. Disponível em: <https://www.labcenter.com/>. Acesso em: 15 maio 2022.

LONGO, Augusto Kesting; ABREU, Bruno Sabino. **ESTUFA AUTOMATIZADA::** controle automático para o cultivo de hortaliças. 2021. 87 f. TCC (Graduação) - Curso de Engenharia Elétrica, Universidade do Sul de Santa Catarina, Tubarão, 2021. Disponível em: <https://repositorio-api.animaeducacao.com.br/server/api/core/bitstreams/3f024378-918a-4f6d-8320-8a5fd28c226a/content>. Acesso em: 16 fev. 2025

MAKERHERO. **Monitore sua planta usando Arduino**. 2016. Elaborado por Adilson Thomsen. Disponível em: https://www.makerhero.com/blog/monitore-sua-planta-usando-arduino/?srsltid=AfmBOorh6Uy_uLfmti7US6AONpkxSryvQp7YT0XtaqOctzYZSLQ6Wrid. Acesso em: 11 mar. 2025.

MAKERHERO. **O que é LDR? Funcionamento, tipos e uso com Arduino**. 2024. Elaborado por Pedro Bertoleti. Disponível em: <https://www.makerhero.com/blog/o-que-e-ldr/?srsltid=AfmBOopwwP676KrfhwwhNFGGe3H98VblywoKTJch7QuJHOEWJf2empBRA>. Acesso em: 11 mar. 2025.

MENESES, André. **Tudo sobre LDR**. 2017. Disponível em: <http://mundoengenharia.com.br/tudo-sobre-ldr-resistor-dependente-da-luz/>. Acesso em: 23 abr. 2022.

MOUHA, R. (2021) Internet of Things (IoT). *Journal of Data Analysis and Information Processing*, **9**, 77-101.

NERI, Renan; LOMBA, Matheus; BULHÕES, Gabriel. **Protocolo MQTT: redes 1**. Redes 1. 2019. Departamento de Eletronica - Escola Politecnica UFRJ. Disponível em: <https://www.gta.ufrj.br/ensino/eel878/redes1-2019-1/vf/mqtt/>. Acesso em: 10 mar. 2025.

PROTOSUPPLIES. **Breadboard Power Supply Module**. 2025. Disponível em: <https://protosupplies.com/product/ywrobot-breadboard-power-supply/>. Acesso em: 25 abr. 2025.

SABERHORTIFRUTI. **O cultivo de hortaliças e a segurança do que comemos**. Disponível em: <https://saberhortifruti.com.br/cultivo-de-hortalicas/>. Acesso em: 19 mai 2022.

SANTIAGO. **San Diego - Estación Meteorológica**. 2025. San Diego, Cerro Largo, Uruguay Estación Meteorológica Solar (Temp, Hum, Presion, Lluvia, Viento). ESP8266, UNO R3, BME 680 Update Interval - 15 seg <https://clima.santiago.ovh/>. Disponível em: <https://thingspeak.mathworks.com/channels/1293177>. Acesso em: 25 abr. 2025.

SANTOS, Lorena Beatriz. **Sistema automatizado para o controle de umidade e temperatura em cultura de morangos aplicados aos pequenos produtores**. 2014. 110 f. TCC (Graduação) - Curso de Engenharia da Computação, Faculdade de Tecnologia e Ciências Sociais Aplicadas, Brasília, 2014. Disponível em: <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwj94jx9ez3AhVyBLkGHR8HByMQFnoECAgQAQ&url=https%3A%2F%2Fr>

epositorio.uniceub.br%2Fjspui%2Fbitstream%2F235%2F5938%2F1%2F21016773.pdf&usg=AOvVaw0BrKNQUsZNPsmjOSZKD0Da. Acesso em: 19 mai 2022.

SBAJOVEM. **Sistemas Embarcados**. 2007. Escrito por Otavio Chase. Disponível em: https://www.maxpezzin.com.br/aulas/6_EAC_Sistemas_Embarcados/1_SE_Introducao.pdf. Acesso em: 08 mar. 2025.

SMARTKITS. **Fonte Ajustável para Protoboard**. Disponível em: <https://www.smartkits.com.br/fonte-ajustavel-para-protoboard>. Acesso em: 25 abr. 2025.

USINAINFO. Disponível em: <https://www.usinainfo.com.br/rele-arduino/modulo-rele-5v-10a-1-canal-jqc3f-para-arduino-e-esp32-4763.html>. Acesso em: 26 mai 2022b.

USINAINFO. https://www.usinainfo.com.br/bombinha-de-agua-e-ar/mini-bomba-de-agua-submersa-3v-5v-jt100-15lmin4927.html?gad_source=1&gad_campaignid=17181939790&qbraid=0AAAAAD4ryP-MpMzChkRrDmRSmkunhfEU-&gclid=Cj0KCQjw2tHABhCiARIsANZzDWp60eqJqYP5F1qhm_l3NFbxOgnOjkZrQUD3T1j46nIROyEDznDEpeAaAuMmEALw_wcB. Acesso em: 20 mar 2025a

USINAINFO. **Fonte Chaveada Colmeia 12V 5A 60W Bivolt - MS-60-12**. Disponível em: <https://www.usinainfo.com.br/fonte-colmeia/fonte-chaveada-colmeia-12v-5a-60w-bivolt-ms-60-12-2609.html>. Acesso em: 04 mai 2025b.

VIDAL, Vitor. **Módulo Relé para automação residencial com Arduino**. 2017. Disponível em: <https://blog.eletrogate.com/modulo-rele-para-automacao-residencial-com-arduino/>. Acesso em: 23 abr. 2022.

ZANNATA. **Produção em estufa agrícola**: conhecendo essa vantajosa estrutura. 2020. Disponível em: <https://www.zanatta.com.br/producao-em-estufa-agricola/>. Acesso em: 23 abr. 2022..

APÊNDICE A - Valores gastos no projeto em reais

Unidade	Descrição	Valor
1	ESP32	R\$ 39,00
1	Sensor de umidade do solo	R\$ 15,68
1	LDR	R\$ 1,50
1	DHT22	R\$ 12,00
1	Cooler 12V	R\$ 9,92
2	Módulo relé 5V - 2 canais	R\$ 15,90
2	Fonte chaveada	R\$ 54,68
1	Mini bomba d'água	R\$ 11,00
1	Lâmpada 12V	R\$ 10,00
	Total	R\$ 169,68

Fonte: Própria do autor.

APÊNDICE B - Resumo apresentado no XXXIII CIC da UNESP



Controle e monitoramento em tempo real de uma estufa para cultivo de hortaliças.

Larissa Cassiano Galan, Suely C. A. Mantovani, UNESP, Campus de Ilha Solteira, curso de Engenharia Elétrica, larissa.cassiano@unesp.br, ICSB.

Palavras Chave: Estufa, Monitoramento, Arduino.

Introdução

Uma solução viável para problemas como alterações climáticas, amplitudes térmicas, pragas e doenças nas lavouras é a utilização de estufas. Elas mantêm a produção contínua de hortaliças em todas as épocas do ano para isso, necessitam que os parâmetros luminosidade, temperatura e irrigação se mantenham em níveis adequados. Por este motivo, sistemas automatizados que auxiliem no controle das estufas, vêm se tornando cada vez mais comuns¹.

Objetivo

Desenvolver um sistema de monitoramento e controle de uma estufa utilizando a plataforma Arduino, sensores e um módulo para comunicação com a web, facilitando a consulta pelo usuário.

Material e Métodos

O projeto de automação de uma estufa é constituído de elementos de hardware e software, que interagem utilizando o Arduino. O monitoramento, em tempo real, por computador é realizado pelo módulo de ethernet ENC20J60, conforme esquema mostrado na Figura 1.

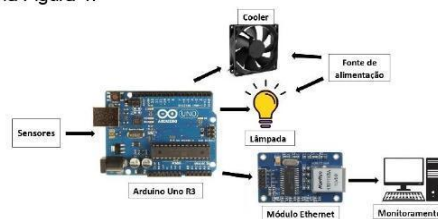


Figura 1. Esquema do projeto.

Os sensores estão ligados ao Arduino Uno R3, que controla, por meio de relés, um cooler e uma lâmpada de LED, acionados conforme a leitura de temperatura e luminosidade, respectivamente. Utiliza-se o LDR (*Ligh* *Dependent* *Resistor*) como sensor de luminosidade e o DHT11 de temperatura, que mede também a umidade do ar, além desses o sensor ESP8266, de umidade do solo. O módulo ethernet é conectado a uma rede *wireless* e ao Arduino possibilitando a visualização da leitura dos sensores em uma página da web. No desenvolvimento (programação e testes) do projeto utiliza-se apenas a IDE do Arduino.

Resultados e Discussão

O circuito montado para testes e a página da web criada para o monitoramento estão na Figura 2.

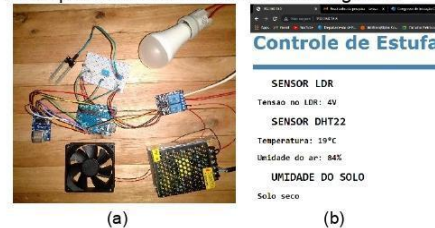


Figura 2. (a) Circuito. (b) Página da Web.

Uma hortaliça cultivada em uma estufa necessita de uma temperatura adequada e uma quantidade específica de luz. Dependendo do valor da temperatura, o Arduino aciona o relé que liga o cooler para resfriar o ambiente da estufa e, caso tenha baixa luminosidade, acende a lâmpada, testes esses realizados na Figura 3.

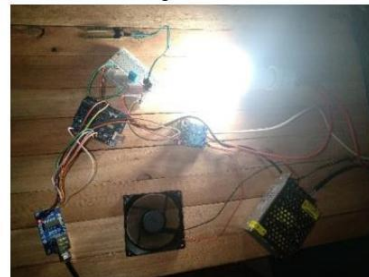


Figura 3. Lâmpada e cooler em funcionamento.

Conclusão

A automação e o monitoramento de parâmetros de uma estufa, utilizando componentes de fácil manuseio e de baixo custo, com a visualização por computadores e/ou smartphones, visa auxiliar o pequeno produtor, constituindo em um objetivo importante para uma agricultura sustentável e um bom aprendizado para o desenvolvimento científico do aluno.

¹ Brito, D. Costa; Gottschall, C. Ramos; Cunha, R. Teixeira; Silva, W. Lacerda. Monitoramento e controle de uma estufa para cultivo abrigado de tomates. MNR 2018. Pós-graduação, 2018.

Apêndice C - Programas em linguagem C++

```

#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"

// Credenciais Wifi
#define SSID "Larissa"
#define PASSWORD "Larissa24"

// Configurações MQTT
#define MQTT_ID "larissa-esp32group-unesp-dee"
#define MQTT_BROKER "hivemq.org"
#define MQTT_PORT 1883

//Tópico de Publicação
#define MQTT_LUMINOSITY_TOPIC "larissa_esp32group_unesp_luminosidade"
#define MQTT_TOPIC_TEMPERATURA "larissa_esp32_unesp_temperatura"
#define MQTT_TOPIC_UMIDADEAR "larissa_esp32_unesp_umidadear"
#define MQTT_TOPIC_UMIDADESOLO "larissa_esp32_unesp_umidadesolo"

// Definição de pinos
#define LDRPIN 35
#define DHTPIN 32
#define SOLOPIN 34
#define COOLERPIN 25
#define LAMPADAPIN 15
#define BOMBAPIN 14

// Definição DHT
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// Definição dos limites
#define LIMIAR_LUMINOSIDADE 2500
#define LIMIAR_MAX_TEMPERATURA 23
#define LIMIAR_MIN_TEMPERATURA 20
#define LIMIAR_UMIDADE_SOLO 2047

// Variáveis dos sensores
int luminosidade = 0;
float temperatura = 0.0;
float umidadear = 0.0;

```

```

int umidadesolo = 0;

// Network e MQTT clients
WiFiClient espClient;
PubSubClient MQTT(espClient);

void setupWIFI() {
    if (WiFi.status() == WL_CONNECTED) return;

    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(SSID);

    WiFi.begin(SSID, PASSWORD);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }

    Serial.println("\nWiFi connected");
    Serial.print("IP Address: ");
    Serial.println(WiFi.localIP());
}

void setupMQTT() {
    MQTT.setServer(MQTT_BROKER, MQTT_PORT);

    while (!MQTT.connected()) {
        Serial.print("Connecting to MQTT Broker: ");
        Serial.println(MQTT_BROKER);

        if (MQTT.connect(MQTT_ID)) {
            Serial.println("Connected to MQTT Broker");
        } else {
            Serial.println("Failed to connect. Retrying in 2 seconds...");
            delay(2000);
        }
    }
}

void Leiturasensores() {
    luminosidade = analogRead(LDRPIN);
    umidadeair = dht.readHumidity();
    temperatura = dht.readTemperature();
}

```

```

umidadesolo = analogRead(SOLOPIN);

// Teste de erro do DHT
if (isnan(temperatura) || isnan(umidadeear)) {
    Serial.println("Erro ao ler DHT!");
    return;
}
}

void PublicaDadosMQTT() {
    MQTT.publish(MQTT_LUMINOSITY_TOPIC, String(luminosidade).c_str());
    MQTT.publish(MQTT_TOPIC_TEMPERATURA, String(temperatura).c_str());
    MQTT.publish(MQTT_TOPIC_UMIDADEAR, String(umidadeear).c_str());
    MQTT.publish(MQTT_TOPIC_UMIDADESULO, String(umidadesolo).c_str());
}

void ControleAtuadores() {
    Serial.print("Temperatura: "); Serial.println(temperatura);
    Serial.print("Umidade do Solo: "); Serial.println(umidadesolo);
    Serial.print("Luminosidade: "); Serial.println(luminosidade);

    // Controle da lâmpada
    if (luminosidade < LIMIAR_LUMINOSIDADE) {
        Serial.println("Lâmpada ligada");
        digitalWrite(LAMPADAPIN, LOW); // Liga a lâmpada
    } else {
        Serial.println("Lâmpada desligada");
        digitalWrite(LAMPADAPIN, HIGH); // Desliga a lâmpada
    }

    // Controle do cooler
    if (temperatura > LIMIAR_MAX_TEMPERATURA) {
        Serial.println("Cooler ligado");
        digitalWrite(COOLERPIN, LOW);
    } else if (temperatura < LIMIAR_MIN_TEMPERATURA) {
        Serial.println("Cooler desligado");
        digitalWrite(COOLERPIN, HIGH);
    }

    // Controle da bomba
    if (umidadesolo > LIMIAR_UMIDADE_SOLO) {
        Serial.println("Bomba ligada");
        digitalWrite(BOMBAPIN, LOW);
    } else {
        Serial.println("Bomba desligada");
    }
}

```

```

        digitalWrite(BOMBAPIN, HIGH);
    }
    Serial.println("-----"); // Separador para
    facilitar leitura no monitor serial
}

void setup() {
    Serial.begin(115200);

    dht.begin();
    setupWIFI();
    setupMQTT();

    pinMode(LDRPIN, INPUT);
    pinMode(SOLOPIN, INPUT);
    pinMode(COOLERPIN, OUTPUT);
    pinMode(LAMPADAPIN, OUTPUT);
    pinMode(BOMBAPIN, OUTPUT);

    digitalWrite(LAMPADAPIN, HIGH);
    digitalWrite(COOLERPIN, HIGH);
    digitalWrite(BOMBAPIN, LOW);
}

void loop() {
    if (WiFi.status() != WL_CONNECTED) setupWIFI();
    if (!MQTT.connected()) setupMQTT();

    MQTT.loop();

    Leiturasensores(); // Lê os sensores
    ControleAtuadores(); // Controla os atuadores automaticamente
    PublicaDadosMQTT(); // Publica os valores lidos

    delay(2000); // Aguarda 2 segundos antes da próxima leitura
}

```