

ETEC– CÔNEGO JOSÉ BENTO
CURSO DE REDES DE COMPUTADORES

ALESSANDRO DA SILVA BARBOSA
LUCIANO SOARES GONÇALVES
MARIA HELENA SIQUEIRA LEITE

SISTEMA DE MONITORAMENTO DE PLANTAS UTILIZANDO O ESP32

JACAREÍ-SP
2025

ALESSANDRO DA SILVA BARBOSA
LUCIANO SOARES GONÇALVES
MARIA HELENA SIQUEIRA LEITE

SISTEMA DE MONITORAMENTO DE PLANTAS UTILIZANDO O ESP32

Trabalho de conclusão de curso apresentado ao
Curso de Redes de Computadores, ETEC Cônego
José Bento

Orientador: Flavio A. Guardia

Jacareí-SP
2025

DEDICATÓRIA

Dedicamos esse trabalho primeiramente a Deus, por ser essencial em nossas vidas, nosso guia, nunca nos desamparar e aos nossos familiares por ser a nossa rede de apoio nos momentos difíceis.

AGRADECIMENTO

Agradecemos sobre tudo a Deus por nos conceder força e sabedoria para concluir essa etapa de nossas vidas. Aos nossos colegas de turma por fazer essa jornada ser mais leve. Ao professor e orientador Flávio A. Guardia pelo apoio, orientação e incentivo durante todo o desenvolvimento do projeto e aos demais professores que tivemos nessa trajetória por toda sua dedicação e sabedoria ensinada.

EPÍGRAFE

“O modo como você reúne, administra e usa a informação determina se vencerá ou perderá.” (Bill Gates).

RESUMO

O presente trabalho tem como objetivo desenvolver um sistema de monitoramento de plantas utilizando o microcontrolador ESP32, sensores e o software Blynk, visando auxiliar agricultores no gerenciamento eficiente de recursos hídricos e nutricionais. A solução busca minimizar os impactos gerados pelos períodos das secas, comuns no agronegócio brasileiro, através do monitoramento e identificação das necessidades do cultivo. Os resultados esperados incluem o aumento da produtividade e a redução de perdas, contribuindo para uma agricultura mais sustentável e tecnológica.

Palavras-chave: ESP32, monitoramento de plantas, recursos, agricultura sustentável, sensores, Blynk.

ABSTRACT

This study aims to develop a plant monitoring system using the ESP32 microcontroller, sensors, and Blynk software to assist farmers in the efficient management of water and nutritional resources. The solution seeks to mitigate the impacts of droughts, a common challenge in Brazilian agribusiness, by monitoring and identifying crop needs in real time. Expected outcomes include increased productivity, reduced losses, and a contribution to more sustainable and technology-driven agriculture.

Keywords: ESP32, plant monitoring, resources, sustainable agriculture, sensors, Blynk.

LISTA DE ILUSTRAÇÕES

Figura 1 - ESP32 Wroom	13
Figura 2 - Sensor de Raio Ultravioleta	Erro! Indicador não definido.
Figura 3 - Sensor de Umidade de Solo	15
Figura 4 - Regulador de Tensão.....	Erro! Indicador não definido.
Figura 5 -Controlador Solar	17
Figura 6 -Mini Pannel de Energia Solar.....	18
Figura 7 - Bateria de Li-Ion.....	18
Figura 8 - Sensor de Chuva	19
Figura 9 - Logotipo Blynk	20
Figura 10 - Logotipo ARDUINO IDE.....	20
Figura 11 - Logotipo C++	21
Figura 12 - Logotipo Wi-Fi	22
Figura 13 – Fluxograma do projeto	23
Figura 14 - Protoboard com o ESP	24
Figura 15 - Protoboard com Sensor de Umidade de Solo	25
Figura 16 - Protoboard com Sensor de Chuva	25
Figura 17 - Sensores Conectados a Protoboard	26
Figura 18 - Processo de Soldagem	26
Figura 19 - Alimentação de Energia	27
Figura 20 - Esquema Completo.....	27
Figura 21 - Circuito.....	28
Figura 22 - Template do Projeto.....	30
Figura 23 - Datastream do Projeto	31
Figura 24 - Datastreams Configurados	31
Figura 25 - Configuração dos Widgets	32
Figura 26 - Web Dashboard	33
Figura 27 - Web Dashboard no Smartphone.....	33
Figura 28 - Notificação na Plataforma	34
Figura 29 - Destinatário das Notificações.....	35
Figura 30 - Intervalo de Notificações.....	35
Figura 31 - Notificações no Smartphone	36
Figura 32 - Teste com a Planta	37

Figura 33 - Dados Coletados.....	37
----------------------------------	----

SUMÁRIO

1	INTRODUÇÃO	11
1.1	OBJETIVO GERAL	12
1.2	OBJETIVOS ESPECÍFICOS.....	12
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	ESP32-WROOM.....	13
2.2	SENSOR DE RAIO ULTRAVIOLETA UV-UVM30A	14
2.3	SENSOR DE UMIDADE DE SOLO	14
2.4	REGULADOR DE TENSÃO SDB628 STEP UP	15
2.5	CONTROLADOR SOLAR PWM 14V/24V	16
2.6	MINI PAINEL DE ENERGIA SOLAR FOTOVOLTAICA 12V.....	17
2.7	BATERIA LI-ION.....	18
2.8	MÓDULO SENSOR DE CHUVA.....	19
2.9	BLYNK.....	19
2.10	ARDUINO IDE.....	20
2.11	C++.....	21
2.12	WI-FI.....	21
3	DESENVOLVIMENTO	23
3.1	ARQUITETURA DO SISTEMA	24
3.2	DESENVOLVIMENTO DO SISTEMA.....	24
3.3	CÓDIGO	28
3.3.1	BIBLIOTECAS	28
3.3.2	DEFINIÇÃO DOS PINOS E CREDENCIAIS DO PROJETO.....	29
3.3.3	LEITURA DOS SENSORES.....	29
3.4	BLYNK.....	29
3.4.1	CRIAÇÃO DO PROJETO.....	29
3.4.2	PINOS VIRTUAIS BLYNK.....	30
3.4.3	DATASTREAM	30
3.4.4	WEB DASHBOARD	32
3.4.5	LÓGICA DE ALERTA	34
3.5	TESTES E RESULTADOS	36
3.5.1	SENSOR DE UMIDADE DO SOLO	38
3.5.2	SENSOR UV	38
3.5.3	SENSOR DE CHUVA	38
4	CONCLUSÃO	39

4.1	TRABALHOS FUTUROS.....	39
	REFERÊNCIAS.....	41
	APÊNDICE A - BRAINSTORM	43
	APÊNDICE B - MAPIFY	44
	APÊNDICE C – CÓDIGO	45

1 INTRODUÇÃO

Atualmente, na sociedade brasileira, o agronegócio se tornou um dos mercados mais potentes e cruciais economicamente do país. Um dos principais motivos disso, deve-se ao aumento considerável da produção agrícola nos últimos anos. Segundo Mesquita (2024).” [...] a atividade agropecuária cresceu 15,1% de 2022 para 2023 [...]”. Através dessa informação é notável o avanço desse mercado, que vem se tornando um comércio fundamental e batendo recordes em 2024, de acordo com o site "Governo Federal" (2024). “Quantidade supera o recorde de novas expansões para produtos agrícolas no comércio internacional em um único ano, de acordo com os registros da série histórica.”.

Entretanto, um dos grandes obstáculos para quem trabalha nessa área, são as secas, que acabam impactando algumas regiões do país e prejudicando o desenvolvimento e o cultivo das plantações, como aponta Cerri (2024): “A seca prolongada, por exemplo, resulta na inviabilidade das sementes, incapazes de germinar devido à escassez de água.”. Nesses períodos, a falta de chuva causa escassez ao abastecimento de água, e com isso, as plantas acabam se desenvolvendo em menor escala e com menor qualidade, afetando financeiramente os produtores (JORNAL DA USP, 2024).

Diante desse cenário, a tecnologia tornou-se uma grande aliada do agronegócio. A inserção tecnológica no meio do agro, trouxe a oportunidade de acompanhar e gerenciar a semeadura, além de reduzir grandes impactos no meio ambiente e principalmente aumentar a produtividade na área, como é referenciado por Neto (2023): “A automação de tarefas, o uso de maquinário avançado e a implementação de sistemas de informação permitem que os agricultores realizem suas atividades com rapidez e precisão, otimizando os processos e aumentando a quantidade e qualidade dos produtos.”. Devido ao grande desenvolvimento dessa ciência, é possível pensar em alternativas viáveis para contribuir e reduzir o impacto das secas. Através de integrações de softwares, sistemas de gestão e equipamentos, há a possibilidade de permitir que as produções e a qualidade do plantio sejam mais eficientes, de maneira que possa conter o efeito desfavorável das grandes secas.

Pensando nisso, o presente trabalho tem como objetivo desenvolver um sistema que possa atender a essas exigências e minimizar seus impactos, explorando

tais vulnerabilidades e visando conceder aos profissionais da área a oportunidade de monitorar e identificar possíveis déficits no cultivo através da tecnologia.

1.1 OBJETIVO GERAL

Implementar um sistema de monitoramento de plantas utilizando um ESP32 e alguns sensores, visando monitorar o desenvolvimento do plantio, destacando o amadurecimento e as necessidades hídricas e nutricionais da vegetação.

1.2 OBJETIVOS ESPECÍFICOS

- **Integrar os sensores ao microcontrolador** para realizar a coleta dos dados;
- **Fazer o código** para capturar os dados gerados pelos sensores.
- **Fazer integração do sistema físico ao Blynk pelo código** para exibir as informações do sistema de maneira gráfica ao usuário;
- **Configurar o Blynk** para exibir os widgets e unidades de medida corretos.

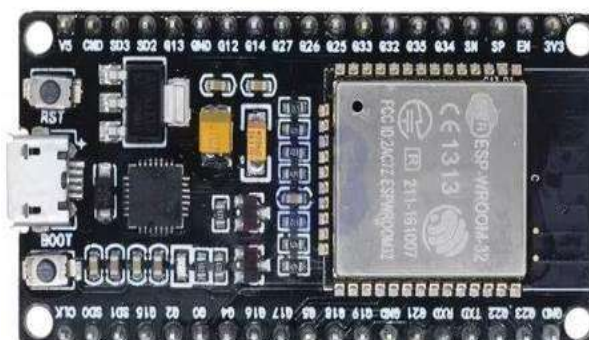
2 FUNDAMENTAÇÃO TEÓRICA

Esse capítulo tem como objetivo referenciar os componentes utilizados no projeto, trazendo seus principais conceitos e responsabilidades para a realização do planejamento. Para compreender a estrutura e o funcionamento do sistema, foram utilizados mapas mentais, conforme consta no Apêndice A e no Apêndice B, como forma de organizar os principais conceitos e componentes envolvidos.

2.1 ESP32-WROOM

O ESP32-WROOM se trata de um microcontrolador eletrônico que possui conectividade sem fio e de baixo custo, com uma alta capacidade para realizar aplicações eficientes aos sistemas embarcados. Conforme descrito pelo autor Neil (2017): "O ESP32 é mais do que apenas um microcontrolador; é uma plataforma que permite a criação de dispositivos inteligentes que podem interagir com o mundo físico e digital de maneiras inovadoras.". Dentro do projeto, o ESP32 é responsável por desempenhar um papel central no sistema, atuando como o microcontrolador responsável por processar os dados coletados pelos sensores e transmiti-los via Wi-Fi para o aplicativo de monitoramento. A figura 1 mostra, como visto, o ESP32-WROOM utilizado no projeto.

Figura 1 - ESP32 Wroom



Fonte: <https://www.mercadolivre.com.br>

2.2 SENSOR DE RAIO ULTRAVIOLETA UV-UVM30A

O Módulo Sensor De Raio Ultravioleta é um dispositivo que detecta a radiação solar por meio do chip UVM-30A, operando na faixa de 200 a 370nm. Ele converte a intensidade da radiação UV em um sinal elétrico proporcional. Segundo Pereira (2018): “[...] é um dispositivo eletrônico que mede a intensidade do sinal luminoso na faixa de 240 a 370 nm, [...]”. Compatível com tensões de 3 a 5V, o módulo pode ser utilizado com microcontroladores como Arduino, *ESP* e *Raspberry PI*. No projeto, ele permite avaliar a exposição solar ao longo do dia, fator importante para o desenvolvimento saudável da vegetação. Como ilustrado na figura 2, o componente do sensor é apresentado.

Figura 2 - Sensor de Raio Ultravioleta



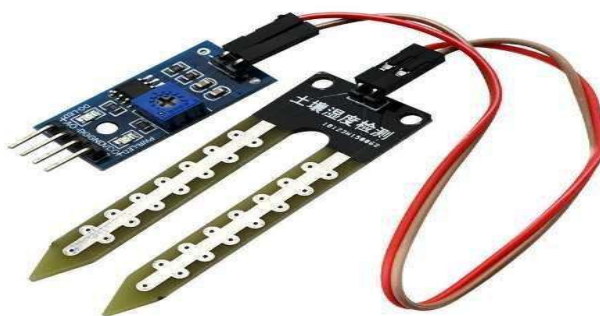
Fonte: <https://www.eletrogate.com>

2.3 SENSOR DE UMIDADE DE SOLO

O Sensor de Umidade de Solo é um dispositivo responsável por medir e identificar as variações de umidade da terra, sendo fundamental para que as plantas recebam a quantidade ideal de água e permaneçam bem nutridas. Esse dado é essencial para alertar o usuário quando a planta apresenta necessidades hídricas, como reforça Cherlinka (2024): “Os sistemas de sensores de umidade do solo provam

ser vitais, uma vez que o cultivo de culturas é um processo dinâmico que requer manutenção regular”. O sensor é apresentado na figura 3.

Figura 3 - Sensor de Umidade de Solo



Fonte: <https://www.mercadolivre.com.br>

2.4 REGULADOR DE TENSÃO SDB628 STEP UP

O regulador de tensão é um dispositivo usado para projetos eletrônicos que necessitam de diferentes valores de tensão para acionamento, opera na função de entrada de 2 a 24V oferecendo tensões de saída entre 2,5 e 28V, sendo ideal para utilização em circuitos eletrônicos, microcontroladores, relés, entre outros. No projeto, ele estabiliza a alimentação do ESP32 e demais módulos, promovendo maior eficiência e proteção ao sistema. A figura 4 representa o regulador de tensão utilizado no sistema.

Figura 4 - Regulador de tensão sdb628 step up

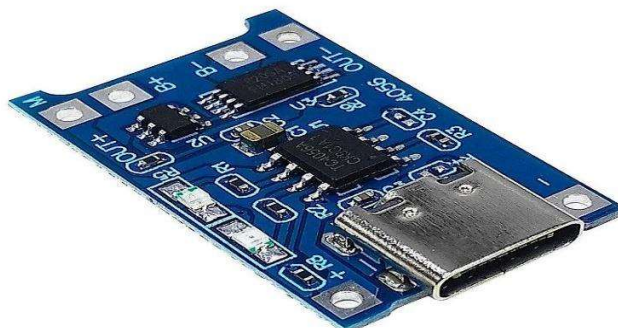


Fonte: <https://www.mercadolivre.com.br>

2.5 CONTROLADOR SOLAR PWM 14V/24V

Controlador Solar é um dispositivo eletrônico que regula a carga da bateria em sistemas solares, gerenciando a vida útil da bateria para que ela não seja sobrecarregada ou descarregada excessivamente, permitindo a maximização do sistema e otimizando a geração de energia. Conforme destacado pela empresa Go Power! (2019): “Um Controlador Solar (ou Controlador/Regulador de Carga) é um componente essencial do seu sistema fotovoltaico. O controlador preserva a vida útil da bateria protegendo-a contra sobrecarga.”. O componente é representado na figura 5.

Figura 5 -Controlador Solar

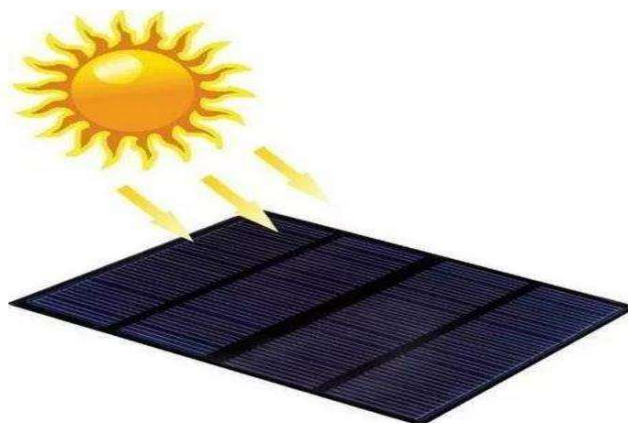


Fonte: <https://shopee.com.br/>

2.6 MINI PAINEL DE ENERGIA SOLAR FOTOVOLTAICA 12V

O painel de energia solar converte diretamente a energia solar em eletricidade através do efeito fotovoltaico, isso permite que o sistema funcione de forma autônoma e sustentável, mesmo em locais sem acesso à rede elétrica. Essa tecnologia é geralmente utilizada em projetos para carregar baterias ou alimentar sistemas eletrônicos, atuando como uma fonte de energia prática. Como destaca Silvério (2021): “A Energia Solar Fotovoltaica (ESF) é alcançada através da conversão direta da luz do sol em eletricidade, denominada Efeito Fotovoltaico”. A figura 6 permite visualizar o funcionamento do mini painel solar em questão.

Figura 6 -Mini PaineL de Energia Solar.



Fonte: <https://www.amazon.com.br/>

2.7 BATERIA LI-ION

A bateria Li-íon é amplamente utilizada para recarregar equipamentos eletrônicos portáteis, devido a sua capacidade de armazenamento e confiabilidade, garantindo o funcionamento do sistema mesmo em períodos sem luz solar. Essa bateria é composta por íons de lítio e não viciam, porém, sua vida útil, geralmente entre 300 e 500 ciclos completos, seja menor quando comparada a outras tecnologias como baterias de chumbo-ácido. Segundo a Iberdrola (s.d), essas baterias são capazes de armazenar mais energia em menos espaço que as demais, por isso serão fundamentais para o futuro do armazenamento de energia. A figura 7 exibe o componente em questão.

Figura 7 - Bateria de Li-Ion



Fonte: <https://produto.mercadolivre.com.br/>

2.8 MÓDULO SENSOR DE CHUVA

O Sensor de Chuva é responsável por monitorar as condições climáticas e identificar a presença de água. Sua integração no projeto ajuda a evitar irrigação desnecessária durante períodos de chuva. Como é referenciado por Moreira (2019), “Este sensor pode ser usado para monitorar uma variedade de condições climáticas, como gotas de chuva ou neve. O sensor verifica se há ou não presença de chuva.”. Utilizando uma placa de circuito impresso (PCB) como um sensor resistivo, ele mede e gerencia a quantidade de umidade presente. O componente é ilustrado na figura 8.

Figura 8 - Sensor de Chuva



Fonte: <https://shopee.com.br/>

2.9 BLYNK

O Blynk é uma plataforma IoT que permite o gerenciamento de dispositivos remotos por intermédio de uma interface gráfica intuitiva. Através dela, o usuário pode visualizar informações como umidade, radiação UV e presença de chuva em um aplicativo intuitivo. Como destacado em sua documentação oficial, seu principal diferencial está na facilidade de uso, onde os aplicativos criados com o Blynk estão prontos para os usuários finais que podem facilmente baixar o aplicativo, conectar seus dispositivos e começar a usá-los (BLYNK.IO, 2024).

Figura 9 - Logotipo Blynk



Fonte: <https://blynk.io/>

2.10 ARDUINO IDE

O Arduino IDE (*Integrated Development Environment*) é um software que permite um ambiente de programação para microcontroladores da plataforma Arduino. Utilizando como linguagem de programação C/C++, foi a ferramenta utilizada para programar o ESP32. Conforme Guse (2024), grande vantagem da Arduino IDE é sua simplicidade de uso, permitindo que até quem não possui experiência em programação possa criar e experimentar projetos de eletrônica.

Figura 10 - Logotipo ARDUINO IDE



Fonte: <https://www.arduino.cc/>

2.11 C++

É uma linguagem de programação compilada e orientada a objetos, foi utilizada para a construção do código embarcado. Criada em 1990, é uma das linguagens mais populares e utilizada em diversas aplicações, como destacam Feliciano e Lamego (2012) “[...] conclui-se que C + + foi criada para ser orientada a objetos e é uma linguagem super utilizada desde programas do Microsoft Office o quanto em editores de imagens, aplicativos do Facebook, além de muito comum também em jogos e sistemas operacionais presentes no cotidiano de várias pessoas”.

Figura 11 - Logotipo C++



Fonte: <https://en.wikipedia.org/wiki/>

2.12 WI-FI

Wi-Fi (*Wireless Fidelity*), se trata de uma tecnologia de rede sem fio utilizada para realizar a conexão de dispositivos à internet por meio de ondas de rádio em redes locais, sem a necessidade de utilizar cabos. Segundo Júnior (2016), “é uma tecnologia muito prática por permitir mobilidade”, o que torna ideal para aplicações que exigem flexibilidade e acesso remoto, como no projeto, onde os dados são enviados para o usuário em tempo real via aplicativo.

Figura 12 - Logotipo Wi-Fi

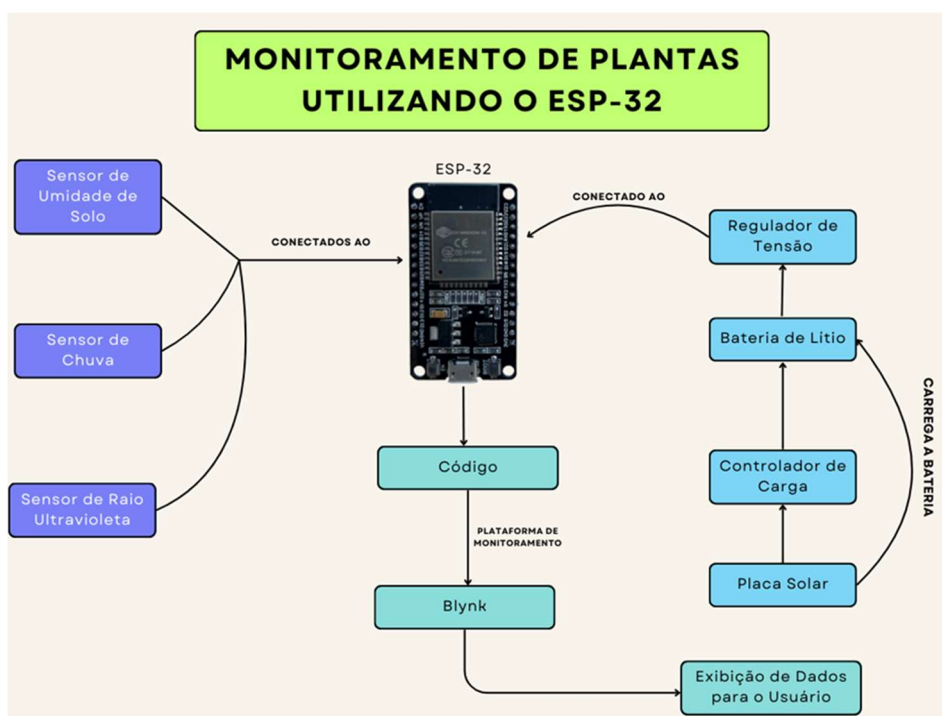


Fonte: <http://en.wikipedia.org/wiki/wi-fi>

3 DESENVOLVIMENTO

O projeto foi desenvolvido com o objetivo de monitorar plantas por meio de um sistema composto pelo microcontrolador ESP32, sensores ambientais e a plataforma Blynk, permitindo que os usuários visualizem os dados de maneira gráfica e intuitiva. A seguir, são descritas as etapas de desenvolvimento do sistema.

Figura 13 – Fluxograma do projeto



Fonte: Elaborado pelos Autores

Conforme o fluxograma representado na figura 13, o ESP32 é responsável por captar as informações dos sensores e enviar os dados para o Blynk através do código, que irá exibir as informações ao usuário de maneira gráfica dentro da plataforma. A alimentação do sistema é feita de forma sustentável, com uma placa solar que carrega a bateria de lítio por meio de um controlador de carga. A energia armazenada passa por um regulador de tensão antes de alimentar o ESP32, garantindo o funcionamento contínuo mesmo em locais sem acesso à rede elétrica.

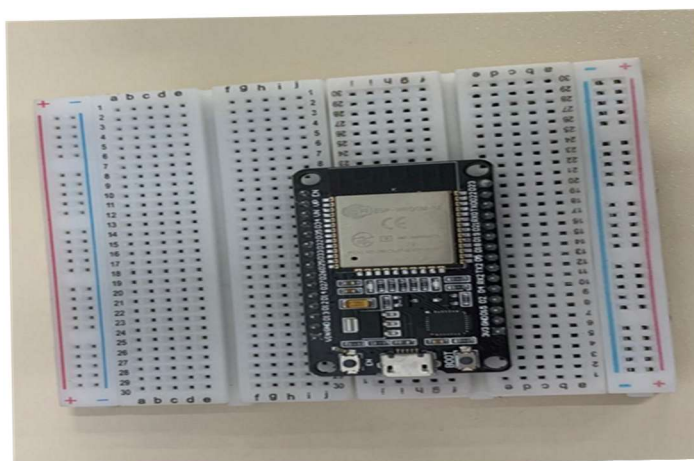
3.1 ARQUITETURA DO SISTEMA

A Arquitetura do sistema, consiste na integração dos sensores ao microcontrolador ESP32, que realiza a captura desses dados e através do código desenvolvido, conforme o Apêndice C, envia os dados através de uma conexão Wi-Fi ao aplicativo Blynk.

3.2 DESENVOLVIMENTO DO SISTEMA

Conforme a Figura 14, o ESP 32 WROOM foi inserido ao protoboard.

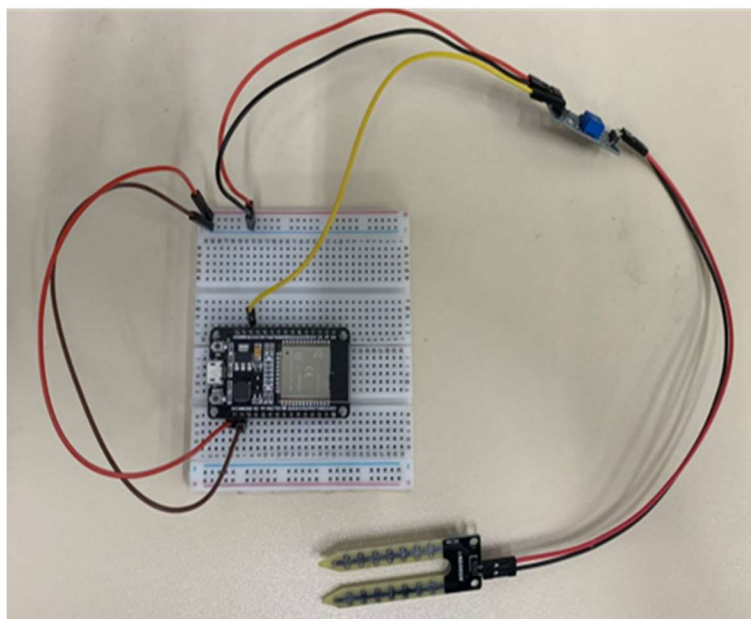
Figura 14 - Protoboard com o ESP



Fonte: Elaborado pelos Autores

Foi inserido na saída 3V3 e GND dos ESP os fios de alimentação dos sensores, e subsequente alimentado o sensor de umidade de solo na protoboard com sua saída D0 conectada ao pino D13 do ESP, demonstrado na figura 15.

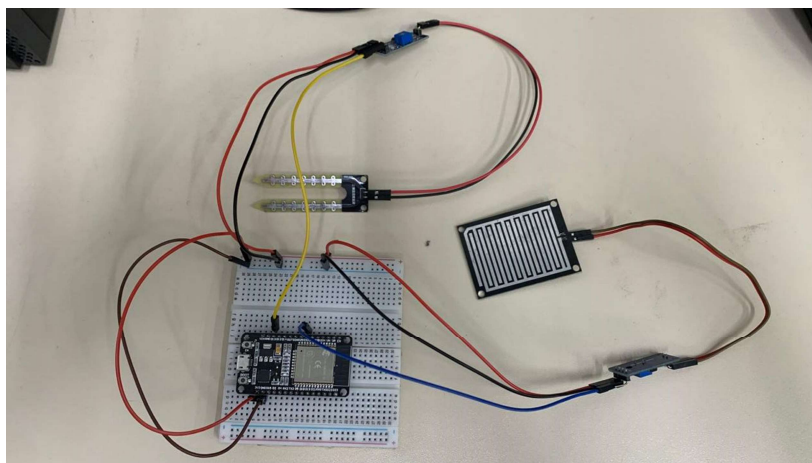
Figura 15 - Protoboard com Sensor de Umidade de Solo



Fonte: Elaborado pelos Autores

Após o processo mencionado anteriormente, foi adicionado o sensor de chuva com sua alimentação ligada no protoboard e sua saída D0 no pino D33 do ESP, como ilustra a figura 16.

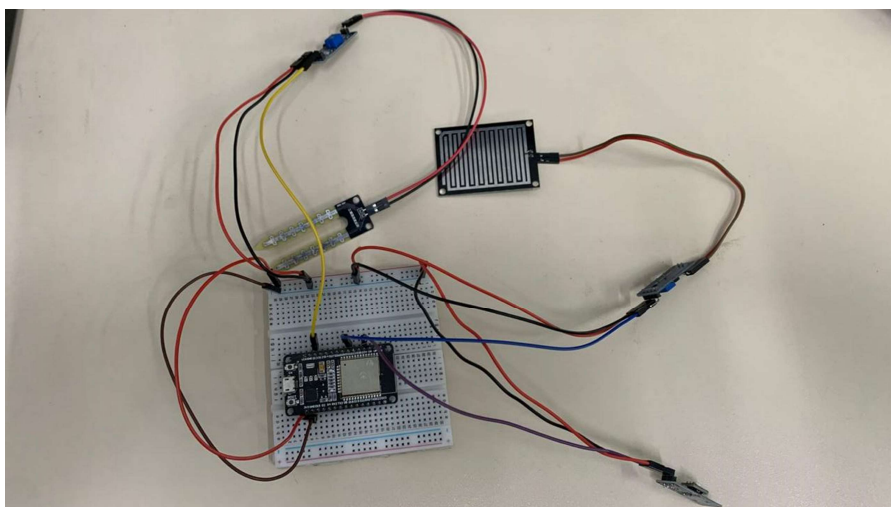
Figura 16 - Protoboard com Sensor de Chuva



Fonte: Elaborado pelos Autores

Ao finalizar a ligação dos sensores, conforme os demais, sua alimentação também foi inserida no protoboard, com seu pino D0 conectado ao pino D34 do ESP, como mostra a figura 17.

Figura 17 - Sensores Conectados a Protoboard



Fonte: Elaborado pelos Autores

Para alimentar o circuito, os fios foram soldados na placa solar conforme a Figura 18.

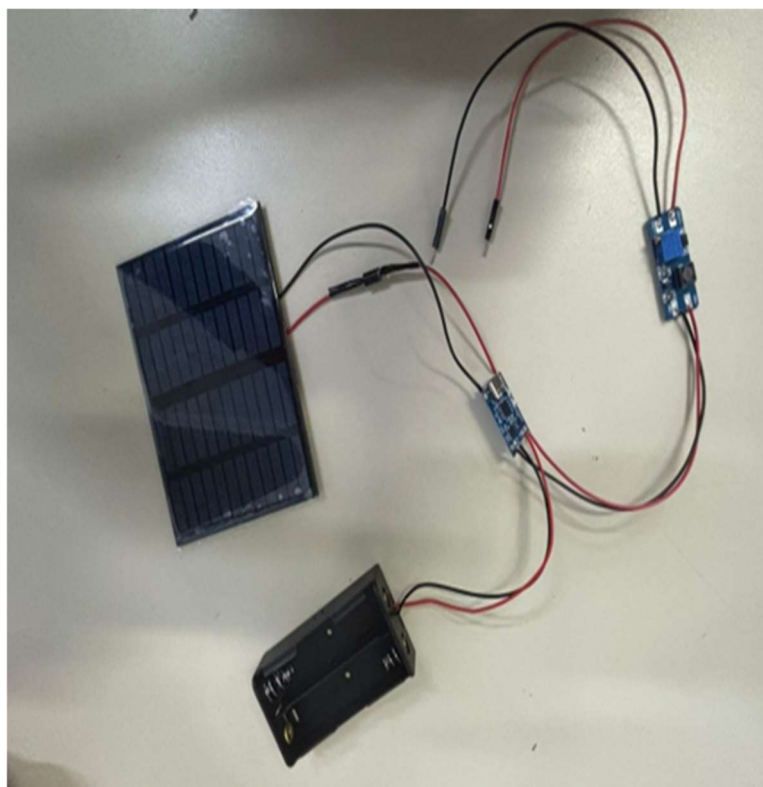
Figura 18 - Processo de Soldagem



Fonte: Elaborado pelos Autores

Após a soldagem, a alimentação fica conforme é ilustrada na figura 19, com a alimentação da placa solar ligada ao controlador de carga que alimenta a bateria, e a saída do controlador de carga está conectada ao regulador de tensão.

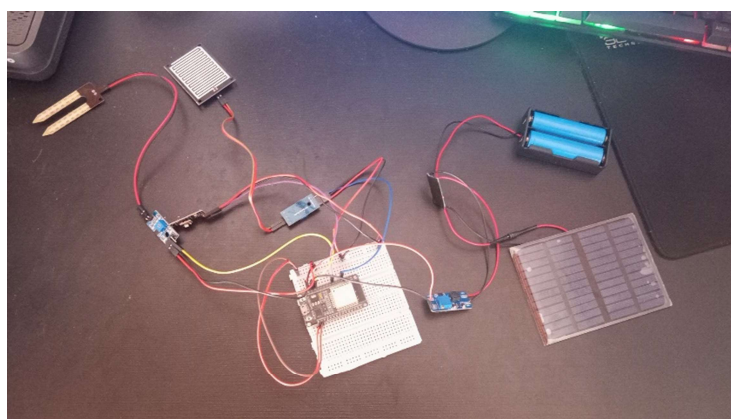
Figura 19 - Alimentação de Energia



Fonte: Elaborado pelos Autores

Por fim, é adicionado o circuito elétrico ao microcontrolador com o positivo no Vin e o negativo no GND do esp, o esquema do projeto fica conforme a Figura 20.

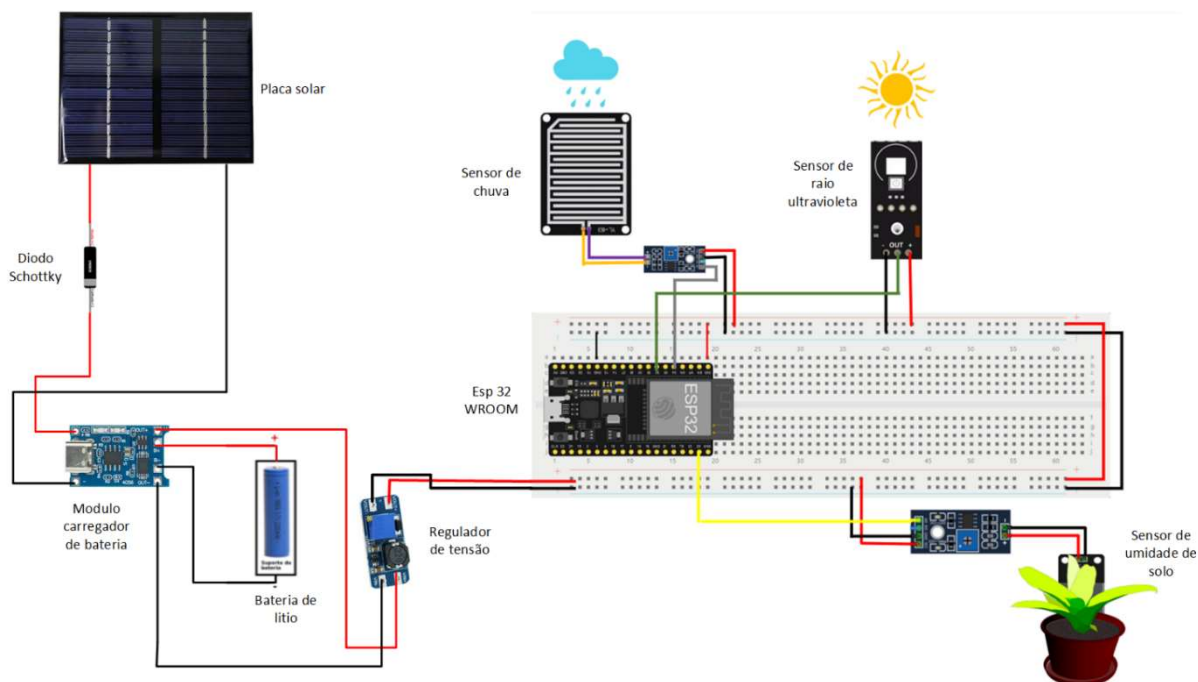
Figura 20 - Esquema Completo



Fonte: Elaborado pelos Autores

Após todo processo de desenvolvimento do sistema físico, a ilustração da figura 21 demonstra o resultado final do projeto.

Figura 21 - Circuito



Fonte: Elaborado pelos Autores

3.3 CÓDIGO

O código foi desenvolvido utilizando a linguagem C++ no ambiente de desenvolvimento Arduino IDE e estruturado para realizar a leitura dos dados capturados pelos sensores e enviar para a plataforma Blynk.

3.3.1 BIBLIOTECAS

Foram utilizadas bibliotecas específicas para facilitar a integração dos sensores, microcontrolador e a plataforma, como mostra o Apêndice C.

As bibliotecas utilizadas foram:

- **Arduino.h:** utilizada para o controle dos pinos e lógica do programa.
- **WiFi.h:** usada para gerenciar a conexão do ESP32 à rede Wi-Fi.
- **BlynkSimpleEsp32.h:** biblioteca responsável por realizar a comunicação entre o ESP32 e a plataforma Blynk. Através dela, foi possível enviar os

dados capturados pelos sensores para os widgets, além de programar uma lógica de alerta ao usuário quando determinada condição estiver abaixo do ideal.

3.3.2 DEFINIÇÃO DOS PINOS E CREDENCIAIS DO PROJETO

Conforme o Apêndice C, para realizar a leitura correta das informações obtidas, é necessário definir os pinos utilizados na integração entre o microcontrolador e os sensores. Além disso, para que os dados sejam enviados aos widgets do Blynk é necessário declarar as credenciais do projeto criado na plataforma (token de autenticação, id do projeto e nome).

3.3.3 LEITURA DOS SENSORES

Após realizar as configurações iniciais do projeto, foi desenvolvido a lógica para realizar a leitura dos dados fornecidos pelos sensores. Conforme o Apêndice C, o sistema realiza leituras constantes das informações captadas pelos sensores por meio de entradas analógicas e digitais. Esses dados são enviados a plataforma Blynk por meio de pinos virtuais, permitindo o monitoramento remoto em tempo real.

3.4 BLYNK

O projeto foi configurado na plataforma Blynk, onde foi criado um template com os widgets correspondentes aos sensores utilizados no projeto.

3.4.1 CRIAÇÃO DO PROJETO

Foi criado um template (modelo base do projeto) na plataforma, realizada através do painel web, onde foi escolhido o nome (Monitoramento de Plantas), tipo de dispositivo (ESP32) e a conexão por Wi-Fi. Após a criação, a plataforma gerou automaticamente um token de autenticação ao projeto, na qual é utilizada no código para realizar a comunicação entre o microcontrolador e o aplicativo. A figura 22 demonstra as configurações definidas.

Figura 22 - Template do Projeto

NAME

Monitoramento de Plantas 24 / 50

HARDWARE CONNECTION TYPE

ESP32 WiFi

TEMPLATE DESCRIPTION (OPTIONAL)

Este template foi desenvolvido para monitorar em tempo real as condições de uma planta.

87 / 128

HOTSPOT PREFIX

Hotspot Prefix

Created At 7:49 PM Mar 26, 2025

Fonte: Elaborado pelos Autores

3.4.2 PINOS VIRTUAIS BLYNK

Foram definidos os pinos virtuais que serão utilizados na plataforma Blynk. Eles serão responsáveis pela comunicação entre o aplicativo e o ESP32, permitindo a troca dos dados coletados pelos sensores de maneira mais flexível, para que seja possível vinculá-los aos DataStreams e definir parâmetros para exibi-los nos widgets da interface. Dessa forma, o usuário pode visualizar as informações captadas pelos sensores.

3.4.3 DATASTREAM

Datastreams são responsáveis por conectar os dados do ESP32 com a interface da plataforma Blynk.

Cada sensor foi vinculado a um datastream específico por meio dos pinos virtuais do Blynk, definidos no código. Com essa configuração, é possível definir o tipo

de dado, unidades de medida e limites numéricos de máximo e mínimo, conforme a figura 23 demonstra.

Figura 23 - Datastream do Projeto

Virtual Pin Datastream

General Expose to Automations

NAME: Umidade do Solo ALIAS: Umidade do Solo

PIN: V3 DATA TYPE: Double

UNITS: Percentage, %

MIN: 0 MAX: 100 DECIMALS: #,## DEFAULT VALUE: Default Value

☒ Enable history data

Cancel Save

Fonte: Elaborado pelos Autores

Na figura 24, é mostrado todos os datastreams do projeto configurados.

Figura 24 - Datastreams Configurados

Monitoramento de Plantas

Cancel Save And Apply

Datastreams

Search datastream

+ New Datastream

3 Datastreams

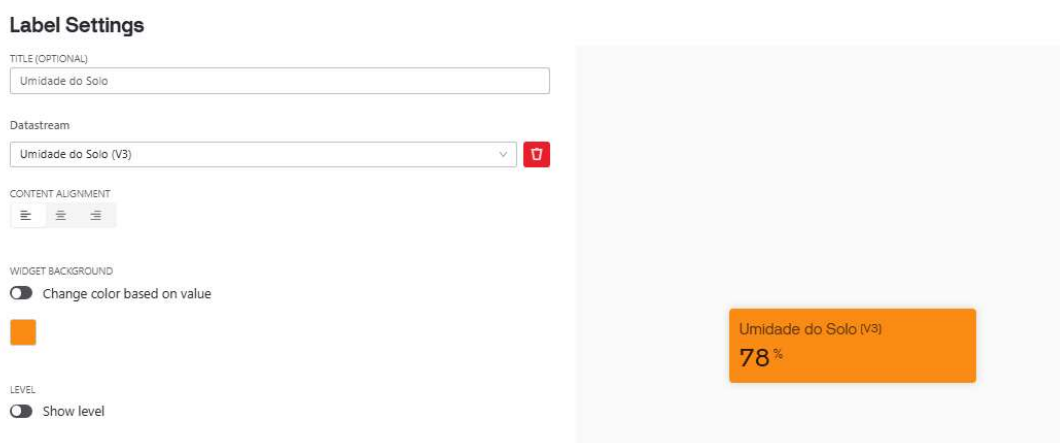
ID	Name	Pin	Color	Data Type	Units	Is Raw	Min	Max	Decimals	Default Value	Auto	Actions
4	Umidade do Solo	V3		Double	%	false	0	100	#,##			Sense
5	Sensor de Chuva	V4		Integer	%	false	0	100	--	0		Switch
6	Índice UV	V5		Double		false	0	11	#,##			Sense

Fonte: Elaborado pelos Autores

3.4.4 WEB DASHBOARD

Após a configuração dos datastreams, foi feita a vinculação dos pinos virtuais aos widgets para exibir os dados captados de maneira gráfica, como ilustra a figura 25.

Figura 25 - Configuração dos Widgets



Fonte: Elaborado pelos Autores

Nessa sessão, é selecionado o tipo de elemento gráfico em que será utilizado para representar cada valor do sensor, além de ser possível estiliza-los. Os widgets utilizados no projeto foram:

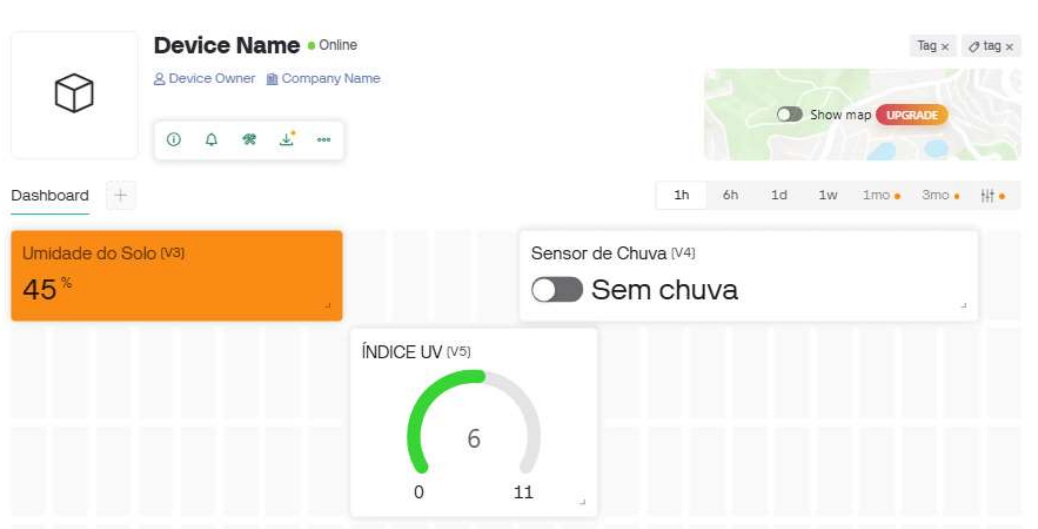
3 **Gauge:** utilizado para exibir o valor do índice UV. Foi configurado com um intervalo entre 0 e 11 e vinculado ao pino virtual V7.

4 **Label:** usado para mostrar informações sobre os sensores de umidade (V2), umidade do solo (V3), radiação UVA (V5), radiação UVB(V6).

5 **Switch:** utilizado para exibir os dados do sensor de chuva, vinculado ao pino V4. Trabalha com os valores 0 (ligado) e 1(desligado). Assim que a chuva é identificada, o botão passa do valor 1 para o 0, e é ativado.

Na figura 26, é mostrado a web dashboard do projeto.

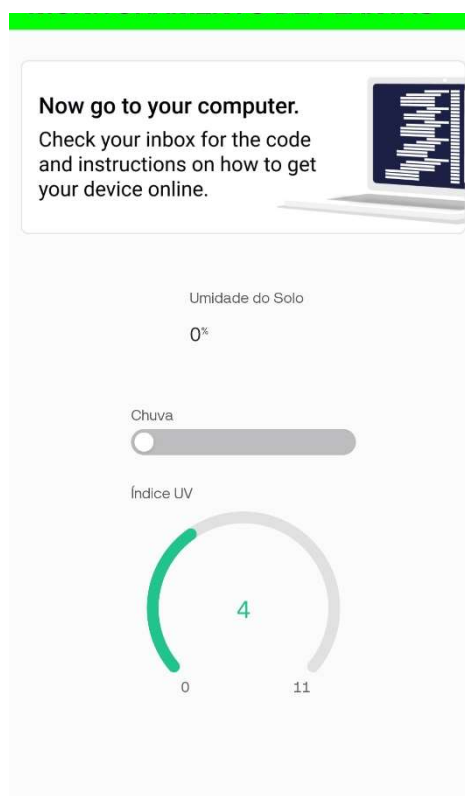
Figura 26 - Web Dashboard



Fonte: Elaborado pelos Autores

Além da configuração dos widgets pelo Desk top, também foram inseridos os mesmos pelo smartphone, como é mostrado na figura 27.

Figura 27 - Web Dashboard no Smartphone



Fonte: Elaborado pelos Autores

3.4.5 LÓGICA DE ALERTA

Foram definidos alguns limites para cada parâmetro monitorado para que, assim que um valor estiver abaixo do ideal, o sistema notifique o usuário sobre tal condição. Isso inclui um índice UV superior a 7.0 (Tevini & Teramura, 1989) e um percentual de umidade de solo inferior a 30%, conforme as diretrizes de irrigação da FAO (*Food and Agriculture Organization*) e Embrapa. Para isso, foi utilizado a função `Blynk.logEvent()` que têm como função registrar eventos personalizados e enviar notificações automáticas.

Além disso, foi configurado também dentro da plataforma do Blynk, inserindo o nome, o evento declarado no código, tipo de notificação e uma descrição sobre a condição, como ilustra a figura 28.

Figura 28 - Notificação na Plataforma

The screenshot shows the Blynk notification configuration interface. At the top, the title is "uv_alto". Below it are four tabs: "General" (selected), "Notifications", "Settings", and "Expose to Automations". Under the "General" tab, there are two input fields: "EVENT NAME" with the value "uv_alto" and "EVENT CODE" with the value "uv_alto". To the right of these fields are two icons: a warning triangle and an orange square. Below these fields is a "TYPE" section with four buttons: "Info", "Warning" (selected), "Critical", and "Content". At the bottom, there is a "DESCRIPTION (OPTIONAL)" text area containing the text "A exposição ao sol está muito alta. Tome precauções." and a character count "52 / 300". A "Cancel" button is located at the bottom right of the form.

Fonte: Elaborado pelos Autores

Na figura 29, na aba "Notifications", foi configurado o destinatário das notificações e por qual endereço elas chegariam (SMS e Email)

Figura 29 - Destinatário das Notificações

uv_alto

General **Notifications** Settings Expose to Automations

☒ Enable notifications

Recipients
Users can set up the notification preferences by going to the Device > Notifications & Events > Notification Settings in Console, or they can also disable them from the Left sidebar > Settings in the mobile app.

E-MAIL TO
Device Owner

PUSH NOTIFICATIONS TO
Device Owner

SMS TO
Select contact

Settings

☐ Enable notifications management
When turned ON, end-users will access advanced notification management for this event

☒ Deliver push notifications as alerts
When turned on, push notifications will use critical alert sounds. End-users will need to turn this setting on in their app settings. They can also change a sound.

Cancel

Fonte: Elaborado pelos Autores

Por fim, na aba “Settings”, foi declarado o intervalo do envio de notificações ao destinatário e a disponibilidade de visualização tanto pela plataforma, quanto pelas notificações do smartphone, conforme demonstra a figura 30.

Figura 30 - Intervalo de Notificações

uv_alto

General Notifications **Settings** Expose to Automations

Limit

Every message will trigger the event

Event will be sent to user only once per

Show in Notifications & Events section

☒ Device page (app and console)

☒ Home screen (app only)

Other

Apply tag when this event is recorded

☐ Allow users with Edit Timeline permission to archive this event

Cancel

Fonte: Elaborado pelos Autores

Na figura 31, é demonstrado as notificações recebidas no smartphone.

Figura 31 - Notificações no Smartphone



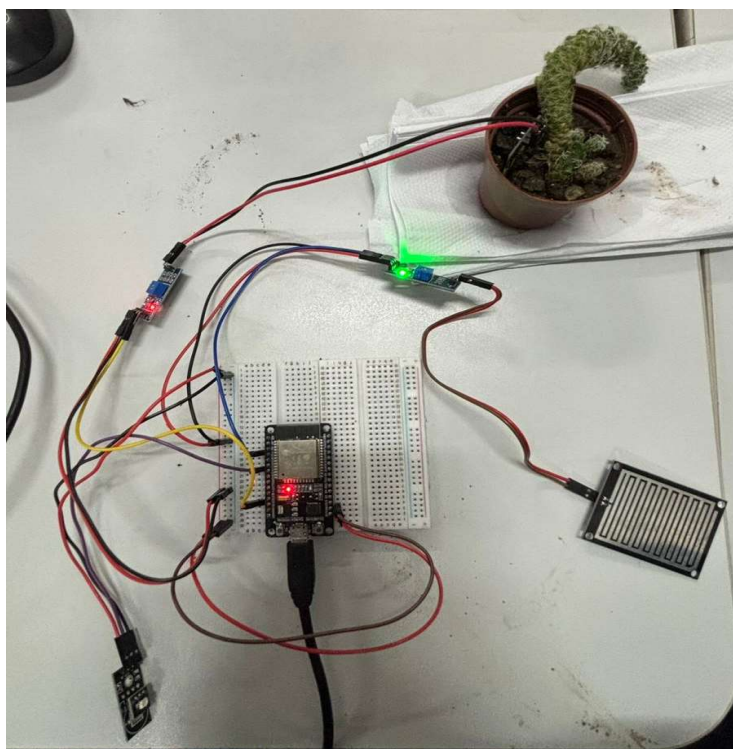
Fonte: Elaborado pelos Autores

3.5 TESTES E RESULTADOS

Nesta seção, são apresentados os testes feitos para validar o funcionamento do sistema.

Conforme a figura 32, após toda a configuração realizada, o teste foi feito com uma planta.

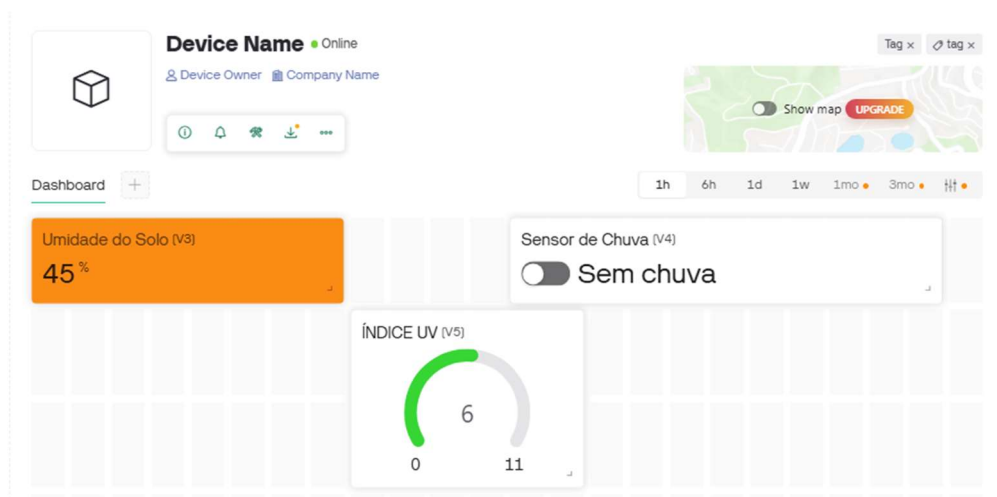
Figura 32 - Teste com a Planta



Fonte: Elaborado pelos Autores

Na figura 33, é demonstrado os dados coletados pelos sensores.

Figura 33 - Dados Coletados



Fonte: Elaborado pelos Autores

3.5.1 SENSOR DE UMIDADE DO SOLO

O sensor de Umidade do Solo conectado ao pino analógico 33 do ESP32, durante os testes feitos, apresentou aproximadamente 45% de umidade, demonstrando que a planta estava com uma quantidade moderada de água, geralmente o suficiente para se manterem saudáveis, como comprova Cherlinka (2022): “A gama ideal de teor de umidade do solo para as culturas depende das espécies vegetais específicas, mas a faixa para a maioria das culturas está entre 20% e 60%.”.

3.5.2 SENSOR UV

O sensor UV conectado ao pino analógico 32, configurado para retornar valores de 0 a 11, demonstrou um índice 6, o que é relativamente alto. Para algumas espécies, pode ser benéfico, mas a exposição a um alto índice de UV pode prejudicar espécies mais sensíveis, afetando seu crescimento e fotossíntese (Tevini & Teramura, 1989).

3.5.3 SENSOR DE CHUVA

O sensor de chuva conectado ao pino digital 13 permaneceu em estado alto (HIGH), indicando ausência de chuva.

Os resultados indicam que o sistema faz leituras coerentes com o teste. Os índices apresentados pelos sensores possibilitam a visualização dos dados em tempo real, permitindo um monitoramento efetivo. Dessa forma, é possível tomar decisões visando o manejo adequado das plantas.

4 CONCLUSÃO

Este trabalho teve como objetivo desenvolver um sistema de monitoramento de plantas utilizando um microcontrolador ESP32 e a plataforma Blynk, com o propósito de exibir os dados captados pelos sensores de maneira simples e intuitiva.

Para isso, foram integrados sensores de umidade de solo, radiação ultravioleta e de chuva ao ESP32 além do desenvolvimento do código, para a leitura dos dados capturados pelos sensores e envio das informações para a plataforma Blynk.

Após a realização dos testes, o sistema demonstrou um bom desempenho e conseguiu fazer a captura dos dados de maneira eficiente, possibilitando o monitoramento em tempo real dos dados, o que pode contribuir no modo de cultivo identificando possíveis necessidades hídricas e nutricionais, além de minimizar os impactos negativos dos períodos de seca de acordo com tais carecimentos reconhecidos, que acabam defasando o solo e afetando o ciclo de vida das plantas.

Apesar dos resultados positivos, o projeto apresenta algumas limitações, como a dependência de uma conexão via Wi-Fi. Por mais que seja possível utilizar uma conexão 4G, algumas áreas de cultivo possuem um sinal fraco ou inexistente, o que pode afetar o desempenho do sistema. A limitação do plano gratuito da plataforma Blynk também é um desafio, já que impossibilita o compartilhamento do projeto com múltiplos usuários, restringindo o acesso ao sistema.

Dessa forma, o trabalho atingiu seu objetivo principal, contribuindo para o uso de tecnologia no setor agropecuário ao implementar um sistema de monitoramento efetivo e com potencial para futuras melhorias.

4.1 TRABALHOS FUTUROS

O presente trabalho, embora tenha apresentado bons resultados, há diversas possibilidades de melhorias futuras para torná-lo mais completo. Entre elas, destacam-se:

- Inclusão de um **sistema de irrigação automática** ao detectar o solo seco.

- Integração com **banco de dados** contendo especificações do tipo de cuidado que cada espécie de planta deve ter e armazenamento dos dados coletados.
- **Aprimoramento de interface no Blynk** permitindo compartilhamento do projeto com múltiplos usuários.
- **Aplicação de técnicas de Inteligência Artificial** para aprimorar a tomada de decisões no sistema de monitoramento de plantas.

Essas melhorias tornariam o sistema ainda mais eficiente e completo para realizar monitoramentos de maneira precisa.

REFERÊNCIAS

BLYNK.IO. **Official Documentation**. 2024. Disponível em: <https://docs.blynk.io/>. Acesso em: 25 abr. 2025.

CHERLINKA, Vasyi. **Sensor De Umidade Do Solo E Seu Funcionamento**. Disponível em: <https://eos.com/pt/blog/sensor-de-umidade-do-solo/>. Acesso em: 23 abr. 2025.

CHERLINKA, Vasyi. **Umidade Do Solo: Nível Ótimo E Controle De Déficit**. Disponível em: <https://eos.com/pt/blog/umidade-do-solo/>. Acesso em: 26 jun. 2025.

FAO. **Optimizing Soil Moisture for Plant Production**, 2003. Disponível em: <https://www.fao.org/4/y4690e/y4690e00.htm>. Acesso em: 26 jun. 2025.

FELICIANO, Patrícia; LAMEGO, Guilherme Cesar. **Linguagem de Programação C++**. Disponível em: http://www.ceavi.udesc.br/arquivos/id_submenu/387/patricia_feliciano____guilherme_cesar_lamego.pdf. Acesso em: 02 abr. 2025.

GO POWER!. **GP-PWM-30-UL User Manual**. 2019. Disponível em: <https://forestriverinc.com/files/Component-Manuals/Electrical/Go%20Power%20-%20Solar%20Controller%20Model%20GP-PWM-30-UL%20User%20Manual.pdf>. Acesso em: 23 abr. 2025.

GOVERNO FEDERAL. **Brasil bate recorde e abre 100 novos mercados para o agro em 2024**. Disponível em: <https://www.gov.br/secom/pt-br/assuntos/noticias/2024/08/brasil-bate-recorde-e-abre-100-novos-mercados-para-o-agro-em-2024>. Acesso em: 25 out. 2024.

GUSE, Rosana. 2024. **O que é Arduino IDE? Como baixar e instalar a nova versão**. Disponível em: <https://www.makerhero.com/blog/arduino-ide/?srsltid=AfmBOopPKygKXrmaONjKjeDiXHTDMLsIHbQB5gkEbwcnyCP3SEBRbi>. Acesso em: 25 abr. 2025.

IBERDROLA. **Baterias de íon-lítio: o que são e como funcionam?**. s.d. Disponível em: <https://www.iberdrola.com/inovacao/baterias-ion-litio>. Acesso em: 25 de abr. 2025.

JORNAL DA USP. **Mudanças climáticas afetam a agricultura e prejudicam a produção de alimentos**. Disponível em: <https://jornal.usp.br/campus-ribeirao-preto/mudancas-climaticas-afetam-a-agricultura-e-prejudicam-a-producao-de-alimentos/>. Acesso em: 26 jun. 2025.

KOLBAN, Neil. **Getting Started with ESP32**. Texas, USA: Self-published. Disponível em:

https://roberthart56.github.io/SCFAB/SC_lab/Electronics/Microcontrollers/ESP32/kolban-ESP32.pdf Acesso em: 12 out. 2024.

MESQUITA, Igor. **Crescimento da economia brasileira é impulsionado pela alta de 15% da agropecuária em 2023**. Disponível em: <https://www.gov.br/agricultura/pt-br/assuntos/noticias/crescimento-da-economia-brasileira-e-impulsionado-pela-alta-de-15-da-agropecuaria-em-2023#:~:text=Puxando%20o%20crescimento%20da%20economia,R%24%2010%2C9%20trilh%C3%B5es..> Acesso em: 25 out. 2024.

MOREIRA, Esdras Emanuel Mariano. **DESENVOLVIMENTO DE UMA MINIESTAÇÃO METEOROLÓGICA DE BAIXO CUSTO BASEADA NA PLATAFORMA ARDUÍNO**, 2019. Trabalho de Conclusão de Curso (Graduação em Redes de Computadores) - Universidade Federal do Ceará, Quixadá, 2019. Disponível em: https://repositorio.ufc.br/bitstream/riufc/50705/1/2019_tcc_eemmoreira.pdf. Acesso em: 23 abr. 2025.

NETO, Nikolly. **Tecnologia no agronegócio: entenda o impacto e a importância**. Disponível em: <https://www.sagri.com.br/tecnologia-no-agronegocio/>. Acesso em: 24 jun. 2025.

PEREIRA, Tiago Quartiero. **Aplicação da metodologia de taxonomia de bloom revisada no ensino de física a parte da análise de dados de estações meteorológicas**. 2018. 178 f. Tese (Pós-Graduação em Física) – Universidade Federal de Santa Catarina, Araranguá 2018. Acesso em: 12 mar. 2025.

SILVÉRIO, Loren Ramos. **VIABILIDADE DE IMPLANTAÇÃO DE ENERGIA SOLAR EM INDÚSTRIAS DE ALIMENTOS DE GOIÁS**, 2021. Trabalho de Conclusão de Curso (Engenharia de Alimentos) - Universidade Federal de Goiás, Goiânia, 2021. Disponível em: https://files.cercomp.ufg.br/weby/up/72/o/TCC_Loren_final_corrigido.pdf. Acesso em: 23 abr. 2025.

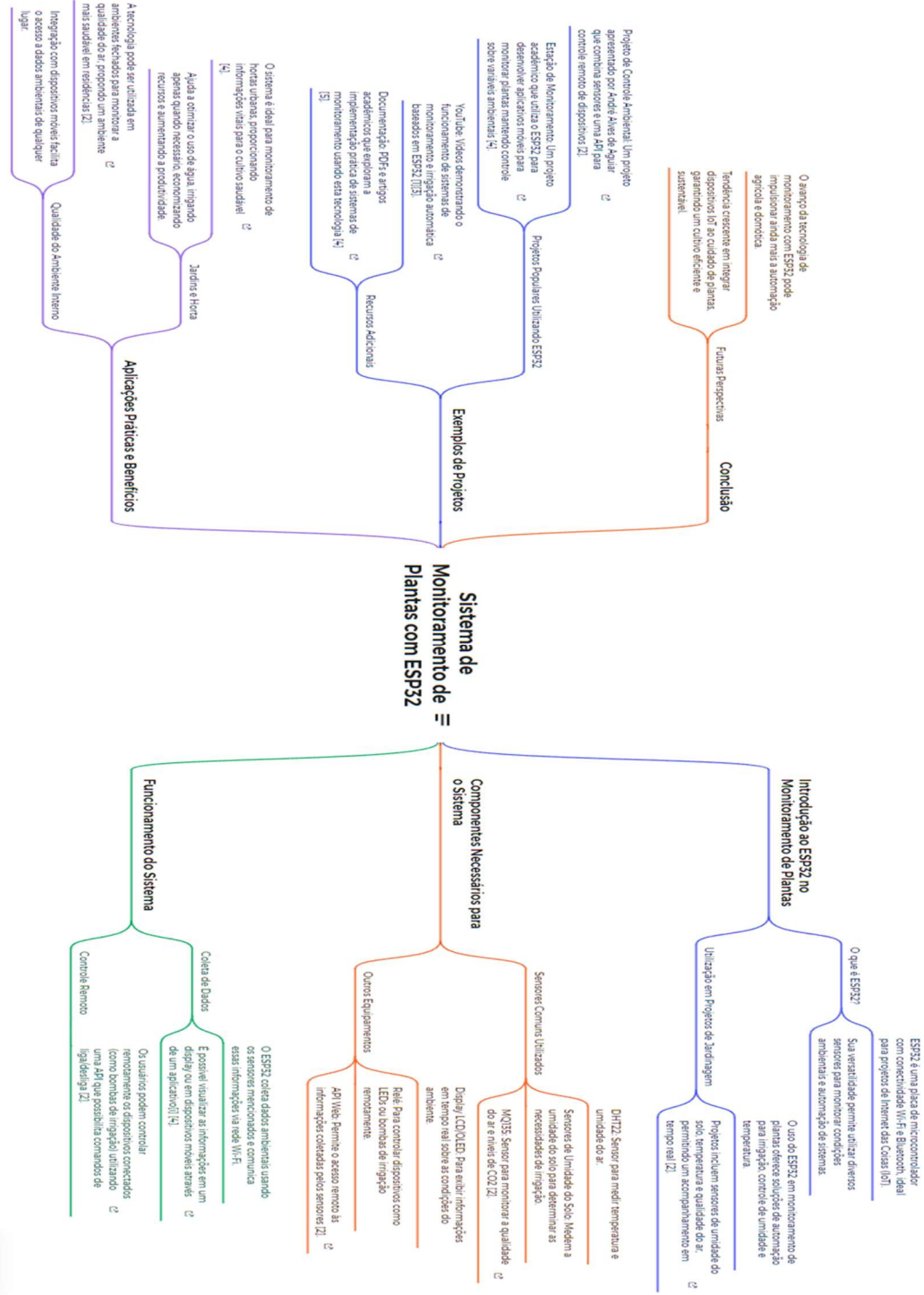
TEVINI, M.; TERAMURA, A. H. **UV-B effects on terrestrial plants. Photochemistry and Photobiology**, [S. l.], v. 50, n. 4, p. 479–487, 1989. Disponível em: 10.1111/j.1751-1097.1989.tb05552.x. Acesso em: 26 jun. 2025.

APÊNDICE A - BRAINSTORM



<https://mm.tt/app/map/3431088179?t=KtRCamK1HE>

APÊNDICE B - MAPIFY



APÊNDICE C – CÓDIGO

```
#define BLYNK_TEMPLATE_ID "TMPL2ekXuChfZ"
#define BLYNK_TEMPLATE_NAME "MONITORAMENTO DE PLANTAS"
#define BLYNK_AUTH_TOKEN "se6Umdlnc2xIJqrqZ2pHi3JXdgS4gW14"

#include <Arduino.h>
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>

// =====
//  CONFIGURAÇÃO WIFI
// =====
char ssid[] = "SSID";
char pass[] = "PASSWORD";

// =====
//  CONFIGURAÇÃO DOS PINOS
// =====
#define UV_SENSOR_PIN 32
#define SOIL_MOISTURE_PIN 33
#define RAIN_SENSOR_PIN 13

// =====
//  PINOS VIRTUAIS BLYNK
// =====
#define VPIN_SOIL V3
#define VPIN_CHUVA V4
#define VPIN_UVA V5
#define VPIN_UVI V7

BlynkTimer timer;
```

```
// =====
//     FUNÇÃO DE ALERTAS
// =====

void checarAlertas(float indiceUV, int soloPercent, int valorChuva) {
  if (indiceUV > 7.0) {
    Blynk.logEvent("uv_alto", "Índice UV elevado!");
  }

  if (soloPercent < 30) {
    Blynk.logEvent("solo_seco", "Umidade do solo muito baixa!");
  }

  if (valorChuva == LOW) {
    Blynk.logEvent("chuva_detectada", "Chuva detectada!");
  }
}

// =====
//     SETUP
// =====

void setup() {
  Serial.begin(115200);
  Serial.println("Iniciando o sistema de monitoramento com Blynk...");

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);

  pinMode(RAIN_SENSOR_PIN, INPUT_PULLUP);

  timer.setInterval(10000L, loopSensores);

  Serial.println("Setup concluído.");
}

// =====
```

```

//  FUNÇÃO DE LEITURA
// =====
void loopSensores() {
  // --- Umidade do Solo ---
  int leituraSolo = analogRead(SOIL_MOISTURE_PIN);
  int soloPercent = 0;

  const int soloMolhado = 1200;
  const int soloSeco = 3200;

  if (leituraSolo < 100 || leituraSolo > 3900) {
    Serial.println("Sensor de solo desconectado ou leitura inválida.");
    soloPercent = 0;
  } else {
    soloPercent = map(leituraSolo, soloSeco, soloMolhado, 0, 100);
    soloPercent = constrain(soloPercent, 0, 100);
  }

  Serial.printf("Umidade do Solo: %d%%\n", soloPercent);
  Blynk.virtualWrite(VPIN_SOIL, soloPercent);

  // --- Sensor de Chuva ---
  int valorChuva = digitalRead(RAIN_SENSOR_PIN);
  Serial.print("Chuva: ");
  Serial.println(valorChuva);
  Blynk.virtualWrite(VPIN_CHUVA, valorChuva);

  // --- Sensor UV ---
  int uvRaw = analogRead(UV_SENSOR_PIN);
  float uvIndex = map(uvRaw, 0, 4095, 0, 11);
  uvIndex = constrain(uvIndex, 0, 11);

  Serial.printf("Valor UV: %.0f\n", uvIndex);
  Blynk.virtualWrite(VPIN_UVA, uvIndex);

```

```
Blynk.virtualWrite(VPIN_UVI, uvIndex);

// --- Alertas ---
checarAlertas(uvIndex, soloPercent, valorChuva);
}

// =====
//      LOOP
// =====
void loop() {
  Blynk.run();
  timer.run();
}
```