



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
CAMPUS CARAÚBAS
CENTRO MULTIDISCIPLINAR DE CARAÚBAS
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

JANNESON JOSÉ FERREIRA DE LIMA

**PROCESSAMENTO DE DADOS DE ESTAÇÕES METEOROLÓGICAS DE BAIXO
CUSTO APLICADO NA AGRICULTURA DE PRECISÃO**

CARAÚBAS – RN

2024

JANNESON JOSÉ FERREIRA DE LIMA

**PROCESSAMENTO DE DADOS DE ESTAÇÕES METEOROLÓGICAS DE BAIXO
CUSTO APLICADO NA AGRICULTURA DE PRECISÃO**

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia Elétrica
do Centro Multidisciplinar de Caraúbas da
Universidade Federal Rural do Semi-Árido,
como requisito parcial à obtenção do grau de
bacharel em Engenharia Elétrica.

Orientador: Prof. Dr. Francisco de Assis
Brito Filho

Co-Orientador: Prof. Dr. Reginaldo Gomes
Nobre

CARAÚBAS – RN

2024

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

L732p Lima, Janneson José Ferreira de .
Processamento de dados de estações
meteorológicas de baixo custo aplicado na
agricultura de precisão / Janneson José Ferreira
de Lima. - 2024.
64 f. : il.

Orientador: Francisco de Assis Brito Filho.
Coorientador: Reginaldo Gomes Nobre.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Engenharia
Elétrica, 2024.

1. Irrigação de Precisão. 2. Evapotranspiração
de Referência e de Cultura - Penman-Monteith-FAO.
3. Internet das Coisas. I. Brito Filho, Francisco
de Assis , orient. II. Nobre, Reginaldo Gomes,
co-orient. III. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade
com AACR2 e os dados fornecidos pelo(a) autor(a).
Biblioteca Campus Caraúbas
Bibliotecária: Dalvanira Brito Rodrigues
CRB: 15/700

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

JANNESON JOSÉ FERREIRA DE LIMA

PROCESSAMENTO DE DADOS DE ESTAÇÕES METEOROLÓGICAS DE BAIXO CUSTO
APLICADO NA AGRICULTURA DE PRECISÃO

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia Elétrica
do Centro Multidisciplinar de Caraúbas da
Universidade Federal Rural Do Semi-Árido,
como requisito parcial à obtenção do grau de
bacharel em Engenharia Elétrica.

Aprovada em: 17 / 04 / 2024

BANCA EXAMINADORA

FRANCISCO DE ASSIS
BRITO
FILHO:

Assinado de forma digital por
FRANCISCO DE ASSIS BRITO
FILHO:
Dados: 2024.04.26 09:59:38 -03'00'

Prof. Dr. Francisco de Assis Brito Filho (Orientador)
– UFERSA
Presidente



Documento assinado digitalmente
REGINALDO GOMES NOBRE
Data: 23/04/2024 22:01:31-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Reginaldo Gomes Nobre
UFERSA



Documento assinado digitalmente
SILVAN FERREIRA DA SILVA JUNIOR
Data: 26/04/2024 09:34:18-0300
Verifique em <https://validar.iti.gov.br>

Me. Silvan Ferreira da Silva Junior

UFRJ



Documento assinado digitalmente
NADISON FRANCISCO DA SILVA
Data: 26/04/2024 17:49:55-0300
Verifique em <https://validar.iti.gov.br>

Eng. Nadison Francisco da Silva
Externo

AGRADECIMENTOS

Agradeço à Deus por ter permitido eu chegar até aqui e ter me feito capaz de buscar realização dos meus sonhos.

Agradeço ao meu pai Edson, minha mãe Gilvanete, meu irmão Jefferson e sua esposa Joyce que são meus pontos de apoio.

Ao meu professor e orientador Brito e também a Nadison por terem confiado a produção deste trabalho à mim.

Agradeço aos meu amigos, John Lennon, Davi Euclides, Edson, Fábio, Eric por toda companhia. Às minhas amigas Geovanna Nunes, Kelly Dantas, Vanessa Alves por todo cuidado que têm por mim.

Agradeço aos meus irmãos de outras mães: Júnior Oliveira, Moisés Campos, Vinícius Fernandes, Gabriel José e Evando Raykaard por todo apoio.

“Everyday is a new day
I’m thankful for every breath I take
I won’t take it for granted
So I learn from my mistakes”

(Alive - P.O.D)

RESUMO

A irrigação de precisão é uma alternativa para uma agricultura sustentável, considerada como parte integrante da Indústria 4.0. Esse avanço passa pela utilização de novas tecnologias direcionadas a atividades de plantio, irrigação, manejo, colheita, dentre outros. Estudos apontam que em torno de 70% da água do mundo é destinada à agricultura e que uma parcela considerável desse montante é desperdiçada, devido a aplicação de volumes superiores ao requerido pelas culturas, uso indevido de sistemas de irrigação e vazamentos. Visando promover sustentabilidade dos recursos hídricos é que se intensifica o desenvolvimento de sistemas inteligentes que evitem o desperdício de água, para isso, é apresentado o conceito de Internet das Coisas, no qual a ligação entre esse conceito e a Indústria 4.0 promovem inovação para as atividades do campo, como por exemplo as estações meteorológicas feitas com Arduino ou ESP32, que fornecem dados de condições climáticas. Aliado a isso, este trabalho objetiva utilizar dados de uma estação meteorológica de baixo custo para aplicar comandos de irrigação de precisão em que os dados dessa estação são enviados periodicamente para um servidor web. O método escolhido para desenvolver o processamento desses dados foi o de Penman-Monteith-FAO, que é um método padronizado e indicado pela FAO (*Food and Agriculture Organization*) para a determinação de evapotranspiração de referência (ET_o) de um determinado local. Com a ET_o, é possível verificar a quantidade de água diária necessária para irrigar uma determinada espécie de planta através do seu respectivo coeficiente de cultivo (K_c) que resulta na evapotranspiração de cultura (ET_c). Para a irrigação, foi criado um sistema embarcado utilizando um microcontrolador NodeMCU ESP8266, um relé de acionamento e uma válvula solenoide, que, por sua vez, é conectada aos tubos de irrigação da plantação e que seu papel no sistema é permitir ou não a passagem de água. Esse sistema embarcado busca os dados no servidor web e faz todo o processamento até chegar em um coeficiente de irrigação que seja suficiente para atender de forma sustentável a espécie agrícola. Outrossim, nesse trabalho destaca-se como foi feito a busca e o tratamento dos dados, bem como o funcionamento do sistema.

Palavras-chave: Irrigação de Precisão. Evapotranspiração de Referência e de Cultura - Penman-Monteith-FAO. Internet das Coisas.

ABSTRACT

Precision irrigation is an alternative for sustainable agriculture, considered as an integral part of Industry 4.0. This advancement involves the use of new technologies aimed at planting, irrigation, management, harvesting activities, among others. Studies point out that around 70% of the world's water is destined for agriculture and that a considerable portion of this amount is wasted, due to the application of volumes greater than that required by the crops, misuse of irrigation systems and leaks. Aiming to promote sustainability of water resources is that the development of intelligent systems that avoid waste of water, for this, the concept of Internet of Things is presented, in which the link between this concept and Industry 4.0 promotes innovation for rural activities, such as weather stations made with Arduino or ESP32, which provide weather conditions data. In addition, this work aims to use data from a station low-cost weather forecast to apply precision irrigation commands where data from this station are periodically sent to a web server. The method chosen for developing the processing of these data was the Penman-Monteith-FAO method, which is a standardized and indicated by FAO (Food and Agriculture Organization) for the determination of reference evapotranspiration (ET_o) of a given location. With ET_o, you can check the amount of daily water needed to irrigate a given plant species through of their respective crop coefficient (K_c) which results in crop evapotranspiration (ET_c). For irrigation, an embedded system was created using a NodeMCU microcontroller ESP8266, a drive relay and a solenoid valve, which, in turn, is connected to the plantation irrigation pipes and that their role in the system is to allow or not the passage of water. This embedded system searches the data on the web server and does all the processing until it reaches in an irrigation coefficient that is sufficient to sustainably serve the species agricultural. Furthermore, this work highlights how the search and processing of data was carried out, as well as the functioning of the system

Keywords: Precision Irrigation. Reference Reference and Crop Evapotranspiration - Penman-Monteith-FAO Internet of Things.

LISTA DE ILUSTRAÇÕES

Figura 1 – Estação Meteorológica de Baixo Custo.	14
Figura 2 – Uso da água no Brasil (retirada e consumo) por setor (m ³ /s).	17
Figura 3 – Irrigação de precisão por gotejamento.	19
Figura 4 – Indústria 4.0.	20
Figura 5 – Ilustração da evapotranspiração.	21
Figura 6 – Coeficientes de cultivo único Kc.	22
Figura 7 – Blocos básicos da IoT.	23
Figura 8 – Estação Meteorológica de baixo custo.	25
Figura 9 – Visão Geral do Projeto.	32
Figura 10 – Pinagem NodeMCU - ESP8266.	34
Figura 11 – Válvula Solenoide.	35
Figura 12 – Módulo relé.	36
Figura 13 – Diagrama Esquemático.	37
Figura 14 – Esquema Elétrico.	38
Figura 15 – IDE Arduino.	39
Figura 16 – Janela "Preferências" da IDE.	40
Figura 17 – Passo 1.	43
Figura 18 – Passo 2.	44
Figura 19 – Passo 3.	44
Figura 20 – Passo 4.	45
Figura 21 – Teste de API Google.	46
Figura 22 – Tela de telemetria.	47
Figura 23 – Verificação de API.	48
Figura 24 – Requisição HTTP e Documentação JSON.	49
Figura 25 – Seleção de variáveis.	50
Figura 26 – Código de seleção de variáveis a partir do JSON.	51
Figura 27 – Variáveis desserializadas.	51
Figura 28 – Cálculo de ETo e ETc.	52
Figura 29 – Tempo de Irrigação.	53
Figura 30 – Definição de momento de irrigação - irrigará.	53
Figura 31 – Definição de momento de irrigação - não irrigará.	54

Figura 32 – Foto do circuito montado 1.	55
Figura 33 – Foto do circuito montado 2	56
Figura 34 – Experimento montado 1.	57
Figura 35 – Experimento montado 2.	58
Figura 36 – Experimento montado 3.	59

LISTA DE QUADROS

Quadro 1 – Trabalhos Relacionados.	27
Quadro 2 – Bibliotecas.	41
Quadro 3 – Quadro de Métodos HTTP.	41
Quadro 4 – Códigos de Verificação HTTP.	49

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVOS	13
1.1.1	Objetivo Geral	13
1.1.2	Objetivos Específicos	14
1.1.3	Apresentação do Projeto	14
1.1.4	Estrutura do trabalho	15
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	DESPERDÍCIO DE ÁGUA NA AGRICULTURA IRRIGADA	17
2.2	IRRIGAÇÃO DE PRECISÃO	18
2.3	MÉTODO DE PENMAM-MONTHEITH-FAO	20
2.4	INTERNET DAS COISAS	23
2.5	ESTAÇÕES METEOROLÓGICAS	24
3	TRABALHOS RELACIONADOS	26
4	MATERIAIS E MÉTODOS	28
4.1	EVAPOTRANSPIRAÇÃO PELO MÉTODO PENMAN-MONTEITH	28
4.2	VISÃO GERAL DO FUNCIONAMENTO DO PROTÓTIPO	31
4.3	COMPONENTES DO PROTÓTIPO - HARDWARE	33
4.3.1	NODEMCU - ESP8266	33
4.3.2	VÁLVULA SOLENOIDE	34
4.3.3	MÓDULO RELÉ	36
4.3.4	DIAGRAMA ESQUEMÁTICO	37
4.4	FERRAMENTAS DE SOFTWARE	38
4.4.1	LINGUAGEM C	38
4.4.2	AMBIENTE DE DESENVOLVIMENTO	39
4.4.3	BIBLIOTECAS E FUNÇÕES	40
4.4.4	PROTOCOLOS HTTP E HTTPS	41
4.4.5	ARDUINO JSON	42
4.4.6	INSOMNIA	45
5	RESULTADOS E DISCUSSÕES	48
5.1	VERIFICAÇÃO DA API	48
5.2	DESSERIALIZAÇÃO DO DOCUMENTO JSON	50

5.3	CÁLCULO DE ETO E ETC	52
5.4	A IRRIGAÇÃO	52
5.5	MONTAGEM DO CIRCUITO	54
5.6	MONTAGEM DO SISTEMA DE IRRIGAÇÃO	56
6	CONCLUSÃO	60
	REFERÊNCIAS	61

1 INTRODUÇÃO

O desperdício de água no agronegócio é algo que vem sendo comentado há bastante tempo e, não é difícil perceber que essa luta é incansável quando se analisa algumas pesquisas. Isso é evidenciado pelo exemplo de Bernardo, Soares e Mantovani (2006) que relatam que “A agricultura irrigada está associada a um elevado nível tecnológico, mas no Brasil esse paralelo não é bem atendido porque se aplica de forma inadequada, desperdiçando muita água”. Esse pensamento entra em concordância com Coin (2022) que afirma que a produção de alimentos está estreitamente relacionada com a disponibilidade de água e que alguns dos alimentos exigem muito mais água para serem produzidos do que outros. O autor relata que o crescimento populacional também é um fator que aumenta o consumo de água, e esse consumo é algo essencial para a vida humana.

Santos *et al.* (2022) afirmam que a preocupação com o uso da água deve-se ao fato de que a agricultura é, de longe, o maior usuário em âmbito mundial de água, representando, em média, 70% da demanda, contra 23% da indústria e 7% do abastecimento humano. Sabendo que o problema da água tem sido tratado de maneira que sempre causa preocupação à sociedade ano após ano, os esforços não devem ser medidos para resolução do problema de desperdício d’água. Deve-se usar as tecnologias existentes para a redução desse problema de maneira que preserve o meio ambiente e reduza a utilização desnecessária de água.

O desenvolvimento do agronegócio está intrinsecamente ligado à adoção de novas tecnologias, que desempenham papel fundamental na promoção do uso sustentável da água, no aprimoramento do cultivo agrícola e na melhoria de sua qualidade. Nesse contexto, a irrigação de precisão pode contribuir tanto com a segurança alimentar quanto com a conservação dos recursos hídricos e ao investigar mais fundo, é perceptível que diversas técnicas dessa abordagem podem ser associadas a conceitos de Internet das Coisas (IoT - do inglês *Internet of Things*) (SANTOS *et al.*, 2022).

Essas tecnologias auxiliam no consumo sustentável de água, bom desenvolvimento e qualidade das culturas. Souza (2021) afirma que a contabilização de água é feita de forma demasiada e há poucos registros da quantidade de água utilizada na agricultura. Aliando-se a esse pensamento, se faz necessário a criação de novas metodologias que sejam relacionadas a irrigação de forma a ser ter um uso de água somente da forma necessária e que se tenha a melhor qualidade da cultura.

O uso de tecnologias é defendido por Paolinelli, Neto e Mantovani (2022) uma vez que

ele apresenta a utilização de Internet das Coisas (IoT) como uma principais tendências para a irrigação 4.0 (Irrigação de Precisão). Os autores apresentam uso de sistemas renováveis dentro do agronegócio e as ferramentas utilizadas como multifuncionais para a realização tarefas de irrigação, fornecimento de água às plantas e aos animais e auxílio nas tomadas de decisão sobre as atividades produtivas.

Além disso, é inerente falar sobre a utilização de estações meteorológicas para auxílio das atividades agrícolas. O cérebro de algumas dessas estações podem ser microcontroladores, mais especificamente ESP-32, onde podem ser conectados à Wi-Fi. Com isso é possível aplicar métodos de irrigação e conseguir monitorar a quantidade de água utilizada.

Assim, esta pesquisa apresenta uma maneira de interligar esses dois conceitos. Através de um método de determinação da evapotranspiração para uso na irrigação de precisão conhecido como Penman-Monteith, que calcula a evapotranspiração de referência (ET_o), é possível aplicar essa ferramenta em estações meteorológicas utilizando os dados colhidos por um microcontrolador, como por exemplo, ESP-32, como apresentado no trabalho de Silva (2021).

Esse projeto se dá no aproveitamento dos dados que uma estação meteorológica capta, de modo a processar em um microcontrolador de forma que seja aplicado o método de irrigação citado acima e acionar, em função do tempo, uma válvula solenoide que permitirá a passagem de água. O tempo que esta válvula permitirá a passagem d'água está em função da ET_o, do tipo de cultura que será irrigado e de parâmetros locais da aplicação como área a ser irrigada e vazão dos irrigadores.

1.1 OBJETIVOS

Os objetivos deste trabalho se dividem em objetivos gerais e específicos:

1.1.1 Objetivo Geral

Aproveitar dados de uma estação meteorológica já existente para aplicação de controle de sistema de irrigação de precisão pelo método de Penman-Monteith-FAO, além de fazer o monitoramento e detectar quando há ou não a necessidade de irrigar, de modo a se aplicar na agricultura de precisão.

1.1.2 Objetivos Específicos

- Realizar o processamento de dados de estações meteorológicas de baixo custo;
- Apresentar os principais métodos usados para a irrigação de precisão;
- Apresentar o método de coleta de dados;
- Apresentar o método de evapotranspiração adotado;
- Apresentar um comando inteligente de irrigação;
- Realizar um levantamento bibliográfico sobre a utilização de irrigação de precisão em estações meteorológicas.

1.1.3 Apresentação do Projeto

A estação Meteorológica (Figura 1), é responsável por verificar diversas variáveis meteorológicas através do uso de sensores específicos. Esse sistema embarcado envia informações via Internet para uma plataforma chamada *ThingsBoard*, onde são guardados os dados em formato de planilha e também possibilita o acesso aos dados via *dashboard* na página dos usuários (SILVA, 2021).

Figura 1 – Estação Meteorológica de Baixo Custo.



Fonte: (SILVA, 2021)

Para atender as funcionalidades da estação foram especificadas as seguintes variáveis que

essa estação monitora (SILVA, 2021).

- Pluviosidade [mm];
- Temperatura do ar [°C];
- Umidade do ar [%];
- Índice de calor do ar [°C];
- Radiação Solar UV [mW/cm²]
- Luz ambiente [Lux];
- Pressão atmosférica [atm];
- Altitude [m];
- Direção do vento;
- Umidade superficial da folha[%];
- Temperatura do solo [°C];
- Umidade do solo [%];
- Nível de bateria [%]

Essa estação não aplica nenhum comando atuador, isso quer dizer que sua especificidade é monitoramento e leitura de sensores como já visto acima. Assim, quando se acrescentar dispositivos de que sejam atuadores, como relés e bombas, fará com que a estação apresente mais utilidades disponíveis para outras aplicações da agricultura.

Assim, o presente trabalho é um projeto desenvolvido em conjunto com o trabalho de Silva (2021), onde o incremento das funções dessa estação é aplicar um comando de irrigação, para diversas culturas, buscando um método de irrigação de precisão que facilite a exploração agrícola sustentável. O método de irrigação escolhido será apresentado no capítulo de fundamentação teórica. Os novos dispositivos acrescentados serão especificados na sessão de materiais utilizados.

1.1.4 Estrutura do trabalho

O presente trabalho está estruturado em 5 capítulos. No Capítulo 2 é apresentada a fundamentação teórica do trabalho, relatando uma visão sobre o que é a irrigação de precisão e sua importância, bem como fazer um paralelo sobre conceitos referentes às tecnologias abordadas no projeto, que são Internet das Coisas e Sistemas Embarcados aplicado a estações meteorológicas de baixo custo.

No Capítulo 3 são apresentados os trabalhos relacionados ao tema de: soluções tecnológicas na agricultura, sistemas embarcados aplicados no contexto de irrigação de precisão.

No Capítulo 4 são abordados os Materiais e Métodos, no qual são apresentados os materiais utilizados, a visão geral da prototipação, os hardwares e os softwares utilizados na incrementação desse projeto. Além disso, é mostrado como é realizado método de evapotranspiração de referência e cálculos necessários para a aplicação da irrigação.

O Capítulo 5 é destinado aos resultados do desenvolvimento do código e também apresentação do sistema em funcionamento.

No Capítulo 6 são apresentadas as conclusões do trabalho realizado.

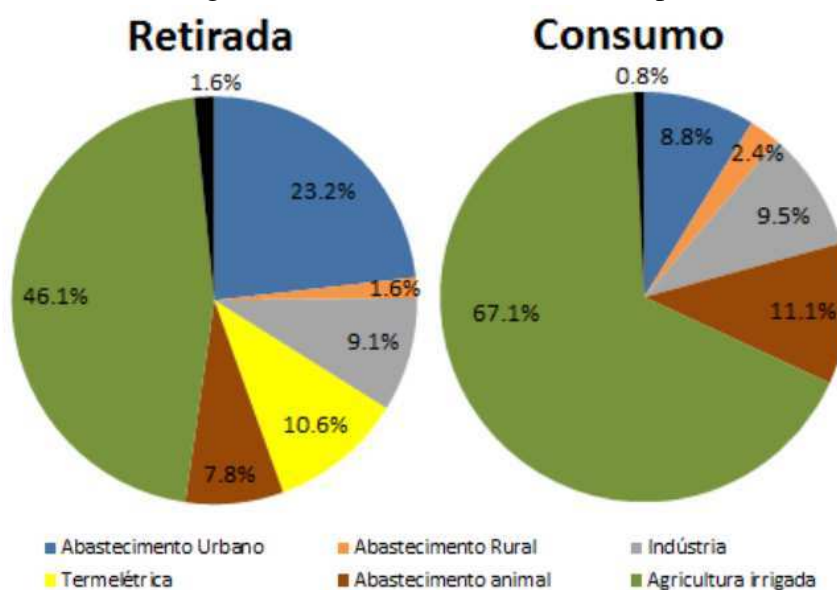
2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados os conceitos teóricos para desenvolvimento do trabalho.

2.1 DESPERDÍCIO DE ÁGUA NA AGRICULTURA IRRIGADA

A agricultura irrigada é um uso consuntivo da água, ou seja, altera suas condições na medida em que é retirada do ambiente e a maior parte é consumida pela evapotranspiração das plantas e do solo, não retornando diretamente aos corpos hídricos. De acordo com Oliveira (2018) os resultados do levantamento atual da irrigação, a atividade é responsável pela retirada de 969 mil litros por segundo ($969 \text{ m}^3/\text{s}$) e pelo consumo de 745 mil litros por segundo ($745 \text{ m}^3/\text{s}$). Considerando os demais usos consuntivos levantados pela ANA (Agência Nacional de Águas e Saneamento Básico), esses valores correspondem a 46% da retirada ($2,105 \text{ m}^3/\text{s}$) e 67% da vazão de consumo ($1,110 \text{ m}^3/\text{s}$). Como mostra a Figura 2.

Figura 2 – Uso da água no Brasil (retirada e consumo) por setor (m^3/s).



Fonte: Oliveira (2018)

Alguns autores afirmam que cerca de 50% da água é realmente aproveitadas pelas plantas, o resto é desperdiçado. Apesar de pesquisas e evoluções terem levado benefícios na agricultura irrigada, é necessário ações para o aumento da eficiência e a consequente redução do desperdício. Isso deve ter tratado como meta constante a ser atingida, uma vez que, o desperdício de água com manejo da irrigação no Brasil é ainda um desafio (OLIVEIRA, 2018).

O desperdício de água contribui de maneira significativa no agravamento de crises hídricas, que podem ser relacionadas ao uso inadequado da água, e como já visto, a agricultura detém do maior consumo de água doce o que faz com que se analise uma alternativa que se mantenha o cenário de produção agrícola mas que também se tenha um uso adequado da água para essas atividades. Galdino *et al.* (2018) alerta que em um cenário de desperdício de água traz consigo a necessidade de desenvolver tecnologias que sejam capazes de reverter o problema da escassez de água, como por exemplo a utilização de tecnologias como Sistemas Embarcados e IoT.

2.2 IRRIGAÇÃO DE PRECISÃO

Gundim *et al.* (2023) definem irrigação de precisão como uma ciência voltada para a otimização dos recursos hídricos em áreas irrigadas é chamada de irrigação de precisão, cujo objetivo é desenvolver tecnologias adequadas para aumentar a produtividade hídrica na agricultura irrigada por meio da aplicação de água em quantidades precisas e exatas no momento certo. Isso faz com que a agricultura de precisão desempenhe um papel significativo na gestão dos insumos envolvidos na produção agrícola.

Nas últimas décadas pesquisas voltadas à irrigação de precisão, impulsionadas pelas mudanças climáticas, desenvolveram uma infinidade de estratégias, métodos e tecnologias para reduzir o consumo de água em projetos de irrigação, para adaptação à crescente ocorrência de escassez de água, secas agrícolas e competição entre os setores agrícolas e industriais pelo uso da água. Nesta conjuntura, a adoção de práticas de economia e aplicação de água, implica em uma abordagem multidisciplinar para a quantificação precisa das necessidades de água das culturas, sob diversas práticas de disponibilidade e manejo da água (GUNDIM *et al.*, 2023).

Diferente da irrigação tradicional a irrigação de precisão (ou irrigação por gotejamento) trabalha a lâmina d'água variando de forma espacial e temporal. A irrigação tradicional entrega uma lâmina d'água uniforme para toda a área irrigada. O termo “irrigação de precisão” também tem sido utilizado para se referir à irrigação por gotejamento como ilustrado na Figura 3. Batizam-se, também, no contexto da Agricultura 4.0, outros termos e expressões bastante frequentes e que se relacionam de certa forma com a irrigação de precisão como “Irrigação 4.0” e o “uso inteligente da água na agricultura” (BRITO, 2023).

Figura 3 – Irrigação de precisão por gotejamento.



Fonte: Blog Agromove (2020)

Agricultura 4.0 permite a geração, o registro, o monitoramento, o armazenamento e a interpretação em tempo real de um grande volume de dados referentes a esses três campos – solo, planta e clima. A integração dessas tecnologias na agricultura, combinando inovação com práticas agronômicas, possibilita decisões mais precisas sobre o momento e a quantidade adequada de água para irrigação. Isso resulta em aumentos significativos na produtividade, ao mesmo tempo em que reduz os custos relacionados à água e energia (SANCHES; CARVALHO; GOMES, 2019). As propriedades rurais que adotam tecnologias da indústria 4.0 em seus processos produtivos e de gestão são conhecidas como "fazendas inteligentes"(smart farms).

Isso pode ser analisado na Figura 4 no qual percebe-se que o que compõe a Agricultura 4.0 é um conjunto de atividades que são centradas no gerenciamento de sistemas, onde englobam monitoramento de dados meteorológicos através de sensores, otimizações de plantio que envolve desenvolver benefícios desde as sementes, passando pela irrigação até a colheita. Compõe também o avanço da indústria 4.0 o controles de irrigação tendo como base “onde, quando e quanto irrigar”. Indo além, também é possível envolver o sistemas de equipamentos agrícolas de maior porte como tratores e ferramentas para manejo de solo.

Figura 4 – Indústria 4.0.



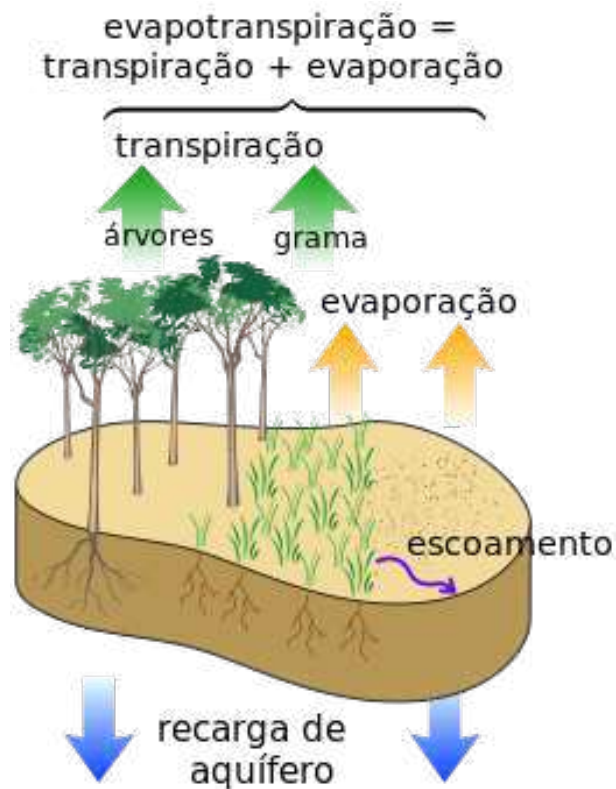
Fonte: Blog Agromove (2020)

2.3 MÉTODO DE PENMAM-MONTHEITH-FAO

Segundo Jin *et al.* (2023) a evapotranspiração pode ser compreendida como a transpiração dos seres vivos vegetais e a evaporação da água proveniente do solo e da água consumida pelas plantas, ou seja, é o somatório da transpiração da vegetação, na qual o autor afirma que pode corresponder a aproximadamente 60% da evapotranspiração total de toda a superfície do globo terrestre, e a evaporação não proveniente da vegetação.

Um dos mecanismos que influenciam na evapotranspiração é chamado de demanda evaporativa, no qual abrange a radiação solar, temperatura do ar, umidade relativa do ar e velocidade do vento, a transpiração das plantas e recuo aquífero (JIAO *et al.*, 2019). Para exemplificação desse conceito uma ilustração é apresentada a Figura 5.

Figura 5 – Ilustração da evapotranspiração.



Fonte: Wikipedia (2024)

A evapotranspiração de referência (ET_0) é utilizada em diversos estudos, tais como em modelos de produção, balanços hídricos e zoneamentos agroclimáticos, bem como no manejo e no dimensionamento de sistemas de irrigação. Atualmente o método considerado padrão para se estimar a ET_0 é o de Penman-Monteith parametrizado pela FAO (*Food and Agriculture Organization*), que é a organização das Nações Unidas para Alimentação e a Agricultura, que trabalha para melhorias no desenvolvimento agrícola e segurança alimentar. Apesar de ser um modelo antigo, segundo Fernandes *et al.* (2012) a própria ONU evidencia ser o modelo mais adequado para a determinação de evapotranspiração.

Jin *et al.* (2023) afirmam que existem dois tipos de evapotranspiração, a de referências e a real. A evapotranspiração de referência (ET_0) é definida como a taxa de evapotranspiração de uma superfície hipotética com características específicas, que representa a transferência de água do solo-planta para a atmosfera. Por outro lado, a evapotranspiração real (ET) (ou evapotranspiração de cultura (ET_c)) considera todas as condições de uma determinada superfície vegetada, sendo a quantidade de água transferida para a atmosfera por evaporação e transpiração. A ET é calculada multiplicando-se a ET_0 pelo coeficiente da cultura (K_c), que leva em conta apenas as condições ideais para o desenvolvimento da cultura, excluindo problemas reais como

pragas, variações na fertilidade do solo e fornecimento de água.

Alguns coeficientes de cultura podem ser vistos na Tabela apresentada no trabalho de Albuquerque e Coelho (2021), tal que esses valores foram padronizados pela FAO em condições padrões. Na Figura 6 os valores Kc1, Kc3 e Kc5 são, respectivamente, coeficientes de cultura da fase inicial, reprodutiva e de maturação. De acordo com o autor o primeiro valor (Kc1) é para colheita fresca e o segundo (Kc3) é para colheita de grãos secos. Já o primeiro valor para Kc5 é para colheita com alta umidade nos grãos. O segundo valor para Kc5 é para cultura colhida após o grão estar seco. O h mostra a altura que as culturas pode alcançar.

Figura 6 – Coeficientes de cultivo único Kc.

Cultura	Kc1	Kc3	Kc5	Altura máxima da cultura - h (m)
Amendoim		1,15	0,60	0,4
Ervilha		1,15 ²	1,10–0,30 ¹	0,5
Feijão	0,40	1,05–1,15 ¹	0,90–0,35 ¹	0,4
Feijão-caupi		1,05	0,60–0,35 ¹	0,4
Grão-de-bico		1,00	0,35	0,4
Lentilha		1,10	0,30	0,5
Soja		1,15	0,50	0,5–1,0
Algodão		1,15–1,20	0,70–0,50	1,2–1,5
Girassol		1,00–1,15 ³	0,35	2,0
Mamona		1,15	0,55	2,0
Arroz	1,05	1,20	0,90–0,60	1,0
Aveia		1,15	0,25	1,0
Cevada		1,15	0,25	1,0
Milheto		1,00	0,30	1,5
Milho		1,20	0,60–0,35 ⁴	2,0
Sorgo		1,00–1,10	0,55	1,0–2,0
Trigo		1,15	0,25–0,40 ⁵	1,0
Cana-de-açúcar	0,40	1,25	0,75	3,0

Fonte: Albuquerque e Coelho (2021)

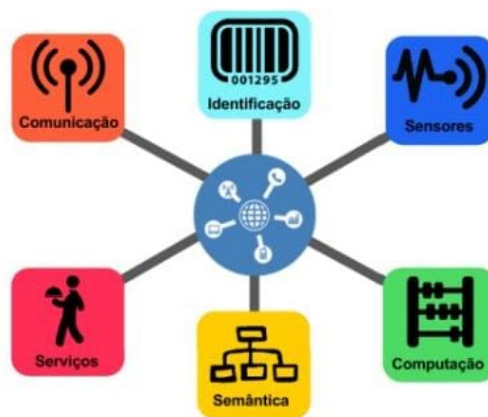
Apesar de contar com modelos onde utilizados dados de temperatura do ar, velocidade do vento, umidade relativa do ar e radiação, nem sempre essas informações estão disponíveis, assim é necessário fazer uma estimativa de uma ou mais variáveis, onde essa estimativa é possível se embasar em trabalhos como o de Conceição (2006) onde ele relaciona metodologias padrão para essas estimativas. Essas tratativas serão mostradas no capítulo de materiais e metodologia.

2.4 INTERNET DAS COISAS

Segundo Santos *et al.* (2016), a Internet das Coisas pode ser definida como a conexão de coisas com a internet e ser controlado ou monitorado via WiFi, que permite fazer comunicações sem fio. Esse avanço foi considerável para muitas áreas, como por exemplo setor residencial, hospitalar e inclusive a agricultura. Por meio da IoT é possível controlar remotamente qualquer objeto que se utilize desta tecnologia. Uma vez que estes objetos se somam a outros recursos disponíveis, uma variedade de aplicações surgem como *Smartwatch*, *SmartTV*, *Smartphones*, aplicações de irrigação inteligente, monitoramento de consumo de água e energia e entre outros.

Santos *et al.* (2016) ainda afirmam que ao conectar objetos com diferentes recursos a uma rede, permite-se a abertura de caminho surgimento de novas aplicações, criando um ciclo contínuo de inovação. Esse processo possibilita o desenvolvimento de novas ideias e avanços tecnológicos, impulsionando ainda mais o progresso na área. O IoT se divide em 6 blocos básicos, que são "identificação", "sensores", "comunicação", "computação", "serviços" e "semântica", conforme pode ser visualizado na Figura 7.

Figura 7 – Blocos básicos da IoT.



Fonte: Santos *et al.* (2016)

Para Santos *et al.* (2016), os blocos mostrados na Figura 7 são descritos da seguinte forma:

- A identificação trata de identificar os objetos e comunicar com a internet, para isso são utilizadas tecnologias como o endereçamento IP (Protocolo da Internet):
- Sensores coletam informações (sensor de presença, umidade, temperatura, radiação e entre outros) e encaminham para os centros de armazenamentos.
- A conexão dos objetos inteligentes se dá pela Comunicação, através de *Wifi*, *Bluetooth* e

etc.;

- A computação compõe a utilização de plataformas micro controladas e processadores, por exemplo, que responsáveis por executar algoritmos locais nos objetos;
- Os serviços contém várias classes com diversas aplicações e são responsáveis pelo mapeamento das entidades físicas (interesse do usuário) para as entidades virtuais, como por exemplo, a temperatura de um local físico;
- A semântica é a extração de informação útil a partir do objetos IoT, atribuindo a descoberta e aplicação dos conhecimentos da IoT para gerar um serviço.

O conceito de internet das coisas é aplicado na solução dos problemas relacionados ao desperdício de água e consequentemente contribuindo para o contexto de irrigação de precisão, como é possível ver no trabalho de Borges (2021). Existem épocas no cenário nordestino que se passam longos períodos sem chuvas que sejam suficientes para a plantação e consumo humano, e também já foi visto que a agricultura consome a maior quantidade de água potável.

Com base nisso, é possível aliar os pontos que mostram que tecnologias como IoT são viáveis para a diminuição do desperdício de água, e uma alternativa é apontada por Kadar, Rafee e Sameon (2018) no qual os autores defendem que esta tecnologia seja usada pelo governo e indústrias, que pode ser proposto como programas sociais e apoio e investimento às pesquisas tecnológicas dentro da própria universidade e no mercado de trabalho.

2.5 ESTAÇÕES METEOROLÓGICAS

Uma estação meteorológica é composta por sensores que captam dados do ambiente em que está inserido e os enviam a uma unidade responsável por armazenar os dados e disponibilizá-los aos usuários. Esse tipo de estação é chamado de estações automáticas, quando todos os dados são gerados por equipamentos que não necessitem da intervenção humana. As estações automáticas podem inclusive transmitir e armazenar os dados automaticamente, agilizando e diminuindo a probabilidade de erros em todo o processo (SILVA *et al.*, 2016).

As estações meteorológicas de baixo custo são compostas por: pluviômetro, sensor de umidade do ar, sensor de umidade do solo, anemômetro e biruta. Geralmente, essas estações que visam ser acessíveis economicamente são feitas utilizando a plataforma Arduino e suas estruturas modeladas e impressas em impressora 3D (DACAL *et al.*, 2019). Estes são materiais fáceis de se adquirir e baratos, porém é necessário ter conhecimento mínimo de programação em linguagem C e de manuseio de impressoras 3D. Um exemplo é mostrado na Figura 8.

Figura 8 – Estação Meteorológica de baixo custo.



Fonte: Silva (2021)

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados alguns trabalhos relacionados ao sistema embarcado desenvolvido neste trabalho.

Através de um levantamento bibliográfico, foi possível encontrar trabalhos relacionados ao tema, isso mostra viabilidade de novos projetos para o meio acadêmico. Baseado nisso, foi pesquisado palavras chaves que sejam referentes à pesquisa como por exemplo: "estações meteorológicas de baixo custo", "métodos de irrigação de precisão", "aplicações de irrigação de precisão com Arduíno ou ESP32". Também foi tentado direcionar essas buscas em aplicações que sejam referente a agricultura brasileira, apesar de haver trabalhos que remetem a outros países também. A procura por trabalhos em foi feita na ferramenta de busca Google Scholar e na plataforma IEEE Xplore. Algumas referências são apresentadas no Quadro 1.

A pesquisa revelou alguns trabalhos que tinham aplicações acadêmicas na educação básica, o que se mostra como uma ferramenta bastante interessante para o desenvolvimento de novos estudos. Trata-se de realizar aplicações em sala de aula de produção da sua própria estação e entender sobre as questões meteorológicas desde a base escolar. Além disso, também gera a experiência em manusear sensores e plataformas micro controladas.

No Quadro 1 são apresentados alguns trabalhos sobre irrigação de precisão e utilização de estações meteorológicas, bem a utilização de plataformas micro controladas como Arduino, NodeMCU. A utilidade desse quadro é mostrar que existem trabalhos que são parecidos com nossa proposta, que é ter uma estação totalmente conectada à internet e que ela envia comando de irrigação para uma determinada plantação.

Este trabalho destaca a importância de promover a sustentabilidade dos recursos hídricos na agricultura por meio do desenvolvimento de sistemas inteligentes que evitam o desperdício de água. Utilizando o conceito de Internet das Coisas e a metodologia de determinação de evapotranspiração de Penman-Monteith-FAO, o trabalho visa otimizar a irrigação de culturas ao aplicar comandos de precisão baseados em dados meteorológicos coletados por estações de baixo custo. A implementação de um sistema embarcado, composto por um microcontrolador NodeMCU ESP8266, um relé de acionamento e uma válvula solenoide, automatiza o processo de irrigação, garantindo a entrega adequada de água às plantas de forma sustentável e eficiente. Esse trabalho destaca a importância da inovação e da padronização metodológica na busca por práticas agrícolas mais sustentáveis e eficazes.

Quadro 1 – Trabalhos Relacionados.

Título	Resumo
Swamp: uma plataforma para irrigação de precisão baseada na Internet das Coisas.(KAMIENSKI; VISOLI, 2018).	O projeto Swamp desenvolve métodos para o gerenciamento inteligente de água na irrigação de precisão, com um foco experimental que está sendo testado em quatro pilotos, dois no Brasil e dois na Europa (Itália e Espanha). Este artigo apresenta uma visão geral do Swamp, sua arquitetura e plataforma computacional, os pilotos e o processo de desenvolvimento baseado em cenários adotado no projeto..
Confecção de uma estação meteorológica de baixo custo para fins educacionais. (CONSIGLIERI, 2023)	Este trabalho teve como objetivo criar uma estação meteorológica de baixo custo com foco em educação, utilizando Arduino e materiais reutilizados. A estação foi capaz de coletar de forma automática variáveis meteorológicas dentro das expectativas quando comparadas a de uma estação comercial.
Desenvolvimento de uma estação meteorológica de baixo custo localizada no IFBA - Campus Parauapebas. (LUZ <i>et al.</i> , 2023).	Este trabalho apresenta a criação de uma estação meteorológica de baixo custo. Os dados recolhidos são armazenados em tempo real numa base de dados através de uma rede informática. Os principais equipamentos utilizados foram Arduino, módulo ethernet e sensores de temperatura, umidade e luz. A estação meteorológica permite coletar dados precisos sobre as condições ambientais do campus Parauapebas-PA do Instituto Federal de Educação, Ciência e Tecnologia do Pará, enquanto o banco de dados serve como repositório para armazenar e gerenciar os dados coletados em um minuto.
Proposta de estação meteorológica baseada em internet das coisas de código aberto e baixo custo como ferramenta para articulação entre novas tecnologias digitais da informação e comunicação no ensino de ciências/física na educação básica (MANNRICH; JR, 2021).	Neste trabalho foi proposto a construção e utilização de estações meteorológicas de baixo custo baseadas na tecnologia de Internet das Coisas por meio de plataformas de código aberto. No estudo foi utilizado o microcontrolador NodeMCU ESP8266 12E, o protocolo MQTT e o servidor Mosquitto Client.
Sistema para Automação e Manejo da Irrigação de baixo custo. (BORGES, 2021)	Este trabalho apresenta como objetivo a construção de um protótipo de automação de irrigação com sistema de controle de malha fechada, visando a determinação do manejo da irrigação para facilitar o trabalho cotidiano dos pequenos produtores rurais, através do uso do microcontrolador NodeMCU v3 Lolin. Este trabalho utilizou o método de irrigação Hangreves, que é um método bastante utilizado também.
Desenvolvimento de um sistema embarcado de baixo custo para automatização do manejo de irrigação localizada (PEREIRA; SANTOS, 2022).	Trata-se de um sistema que faz a aquisição e processamento de variáveis climáticas, obtidas por meio de uma estação meteorológica integrada e, a partir da inserção da localização geográfica, dados da cultura e sistema de irrigação, utiliza o método Penman-Monteith para determinar a evapotranspiração de referência (ET _o), a evapotranspiração da cultura (ET _c), o tempo de irrigação e o acionamento automático e programado do conjunto motobomba para manejo da irrigação localizada por gotejamento ou microaspersão.

Fonte: Autoria própria (2021)

4 MATERIAIS E MÉTODOS

Neste capítulo são apresentados os tópicos referentes aos materiais e métodos utilizados no desenvolvimento do sistema embarcado. Além disso, a metodologia adotada na determinação de evapotranspiração e do tempo de irrigação.

4.1 EVAPOTRANSPIRAÇÃO PELO MÉTODO PENMAN-MONTEITH

A evapotranspiração de referência (ET_o) é utilizada em diversos estudos, tais como em modelos de produção, balanços hídricos e zoneamentos agroclimáticos, bem como no manejo e no dimensionamento de sistemas de irrigação. O roteiro adotado para aplicação do presente trabalho está indicado no trabalho de Conceição (2006), onde esse padrão é parametrizado pela FAO. A seguir é apresentado o roteiro de cálculo.

$$ET_o = \frac{0.408 \cdot \Delta \cdot (R_n - G) + \gamma \cdot \frac{900}{T+273} \cdot U \cdot (e_s - e_a)}{\Delta + \gamma \cdot (1 + 0.34 \cdot U)} \quad (4.1)$$

- ET_o é a evapotranspiração (em mm/dia);
- Δ é a declividade da curva de pressão de vapor em relação à temperatura (em kPa/°C);
- R_n é o saldo de radiação diário (em MJ/m²/dia);
- G é o fluxo total diário de calor no solo (em MJ/m²/dia);
- γ é o coeficiente psicrométrico (em kPa/°C);
- T é a temperatura média do ar (em °C);
- U é a velocidade do vento a 2 m de altura (em m/s);
- e_s é a pressão de saturação de vapor T (em kPa);
- e_a é a pressão de saturação de vapor (em kPa).

Para calcular a pressão de saturação de vapor de água (e_s) em kPa, usou-se a equação de Magnus-Tetens:

$$e_s = 0.6108 \cdot \exp \left(\frac{17.27 \cdot T}{T + 237.3} \right) \quad (4.2)$$

- Onde T é a temperatura do ar em °C.

Para calcular a inclinação da curva de saturação do vapor de água (Δ) em kPa/°C, usou-se a seguinte equação:

$$\Delta = \frac{4098 \cdot e_s}{(T + 237.3)^2} \quad (4.3)$$

Substituindo a equação de e_s , temos:

$$\Delta = \frac{4098 \cdot (0.6108 \cdot \exp(\frac{17.27 \cdot T}{T+237.3}))}{(T + 237.3)^2} \quad (4.4)$$

Para calcular a pressão real de vapor de água do ar (e_a) em kPa, usou-se a seguinte fórmula:

$$e_a = \frac{UR \cdot 0.6108 \cdot \exp(\frac{17.27 \cdot T}{T+237.3})}{100} \quad (4.5)$$

Onde:

- e_a é a pressão real de vapor de água do ar (em kPa);
- UR é a umidade relativa do ar (em %);
- T é a temperatura do ar (em °C).

O coeficiente " γ " é calculado pela expressão:

$$\gamma = 0,665 \times 10^{-3} \times P \quad (4.6)$$

Onde:

- P é a pressão atmosférica.

A equação 4.1 resulta em uma unidade de medida de milímetros por dia [mm/dia], na qual é comumente usada para medir taxas de precipitação, ou seja, a quantidade de chuva que cai em milímetros durante um período de um dia. A Evapotranspiração é um processo pelo qual a água é transferida da superfície terrestre para a atmosfera, tanto por evaporação da água da superfície do solo quanto pela transpiração das plantas. É uma combinação dos processos de evaporação, que é a mudança de água líquida para vapor, e transpiração, que é a liberação de vapor de água pelas plantas durante a fotossíntese (YANG *et al.*, 2023).

Tendo isso, é possível fazer uma aplicação de irrigação para uma determinada cultura. Analisando pelo método de Conceição (2006), isso é dado pelo Kc.

$$ETc = ETo \times Kc \quad (4.7)$$

Onde:

- ETo é a evapotranspiração;
- Kc é o coeficiente de cultura.

A determinação de lâmina d'água pode ser seguida o modelo encontrado em Daré (2013), onde ele descreve as equações que disponibiliza a lâmina d'água e o tempo de irrigação através dos valores de evapotranspiração de cultura. Com isso, pode-se iniciar descrevendo a Eq. 4.8 onde irá definir a lâmina d'água de acordo com a evapotranspiração e a precipitação do dia.

$$ITN_{dia} = \frac{ET_c - P_e}{E_i} \quad (4.8)$$

em que:

- ITN_{dia} = lâmina total de irrigação necessária diária (mm);
- ET_c = evapotranspiração de cultura (mm/dia);
- P_e = é a precipitação de chuva no local;
- E_i = Eficiência de irrigação (%).

O volume de água demandado pela cultura irrigada é dado pela fórmula 4.9.

$$V_{cultura} = 10 \times \frac{ITN \times A}{1000} \quad (4.9)$$

em que:

- $V_{cultura}$ = volume de água requerida pela cultura (m^3);
- ITN = lâmina total de irrigação necessária (mm);
- A = área de irrigação (m^2).
- 10 é a constante presente na fórmula;
- 1000 é para transformar mm em m.

E o tempo de irrigação pode ser dado pela fórmula 4.10.

$$TC = \frac{V_{cultura}}{Vazao} \quad (4.10)$$

em que:

- TC = tempo calculado (h);
- $V_{cultura}$ = volume de água requerida pela cultura (m^3);
- $Vazao$ = vazão de água disponível (m^3/h).

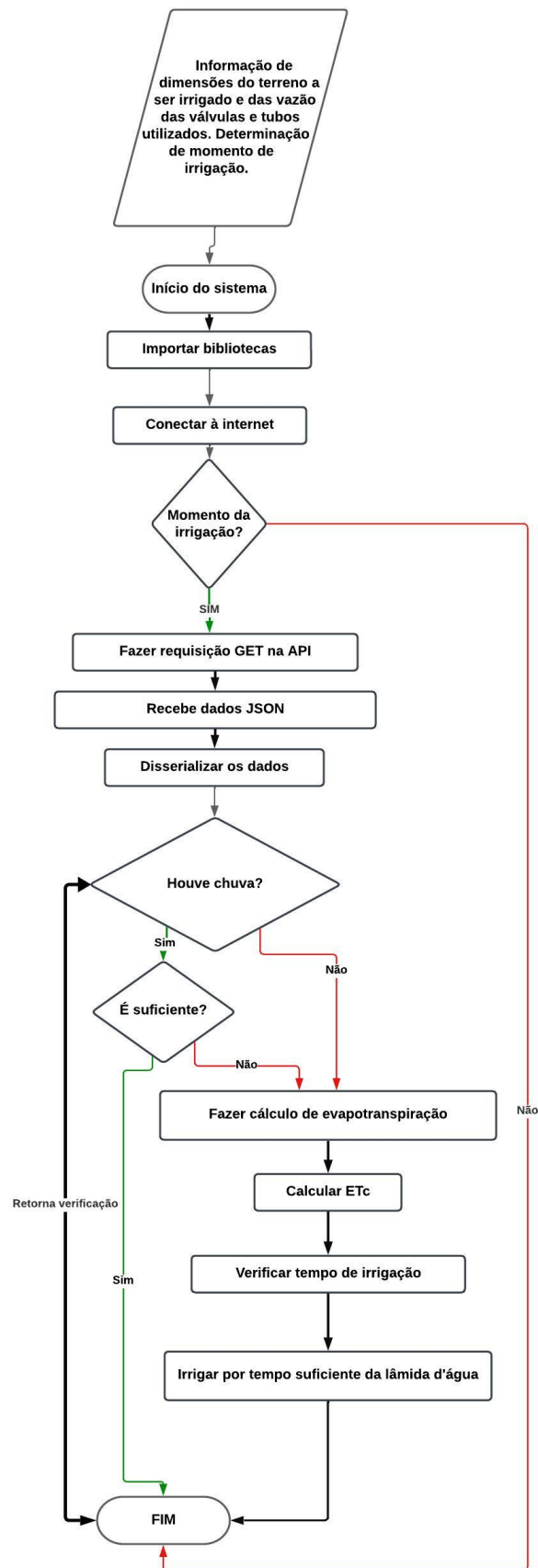
4.2 VISÃO GERAL DO FUNCIONAMENTO DO PROTÓTIPO

O sistema embarcado para aplicação em irrigação de precisão tem suas funcionalidades divididas nas seguintes etapas:

1. Verificação dos parâmetros para irrigação;
2. Leitura dos dados da estação meteorológica;
3. Cálculo de evapotranspiração de referência;
4. Cálculo para a evapotranspiração de cultura;
5. Cálculo de tempo de irrigação.

Todas essas etapas incluem realizar os cálculos mostrados na subseção 4.1. Isso vai de encontro ao objetivo do trabalho de realizar um comando de irrigação de precisão. Na Figura 9 é ilustrado um fluxograma do funcionamento geral do sistema embarcado.

Figura 9 – Visão Geral do Projeto.



Fonte: Autoria própria (2024)

Observando a Figura 9, pode-se observar que é necessário que o usuário informe as dimensões do terreno a ser irrigado como área e profundidade do solo média. Também será requisitado que se informe a espécie agrícola e o respectivo coeficiente de cultivo a qual está submetido o protótipo para irrigação. O depósito desses dados é proposto seja adicionado pelo próprio usuário na plataforma *ThingsBoard* desenvolvida no trabalho Silva (2021), que é o local onde está armazenado todos os dados colhidos pela estação meteorológica.

Com isso, dá-se início ao processo do sistema embarcado. Ao ESP8266 ser iniciado, ele irá importar as bibliotecas utilizadas, onde irá conectar à internet, verificar se está na hora definida pela usuário para irrigação. Após isso, se estiver em um horário de irrigação, o microcontrolador irá requisitar ao servidor os dados meteorológicos naquele momento. Recebendo os dados em formato JSON (*JavaScript Object Notation*), o algoritmo irá desserializar - transformar dados em um formato que possa ser facilmente armazenado, transmitido e reconstruído posteriormente - os dados para fazer verificações se o solo está úmido (o que pode indicar que choveu e não é necessário aplicar a irrigação) e também realizar a manipulação das variáveis de forma a calcular a evapotranspiração de referência e de cultura. Feito isso, calcula-se o tempo necessário para irrigar uma determinada cultura.

4.3 COMPONENTES DO PROTÓTIPO - HARDWARE

A escolha por estes componentes foi realizada com o intuito manter o projeto viável economicamente e fácil de ser construído e reprogramado. Além de acrescentar mais utilidades para a estação meteorológica criada. Tendo em vista isso, para a construção do protótipo de acionamento de irrigação foram necessários os seguintes componentes de hardware:

1. NodeMCU ESP8266;
2. Válvula Solenoide;
3. Módulo relé;

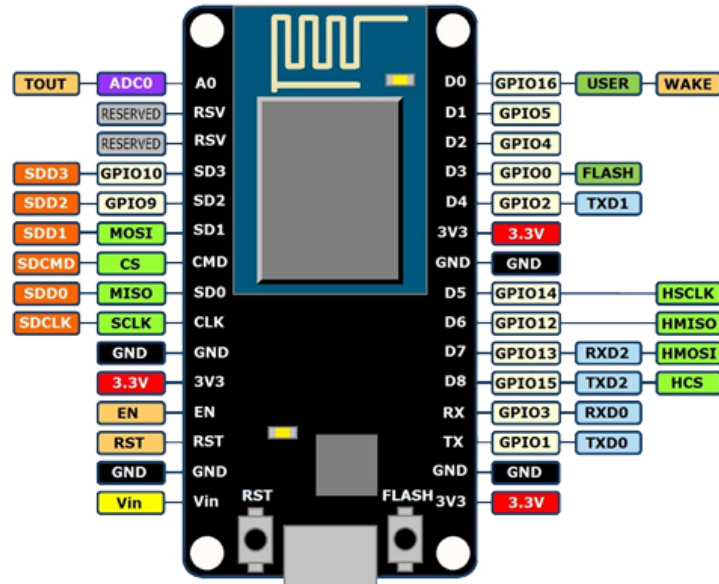
Estes serão descritos nas subseções a seguir e também será mostrado um diagrama esquemático da montagem do sistema na seção 4.3.4.

4.3.1 NODEMCU - ESP8266

O projeto foi desenvolvido utilizando o microcontrolador NodeMCU, um *firmware* que permite programar módulos ESP8266 (FERNANDES *et al.*, 2017). O módulo ESP8266 envia

dados de forma serial pela rede em que é conectado. O servidor pode conter uma página *web* com informações dos componentes ligados ao ESP8266. Uma visão geral do NodeMCU e sua pinagem pode ser vista na Figura 10.

Figura 10 – Pinagem NodeMCU - ESP8266.



permitindo ou interrompendo o fluxo de fluido. O controle do fluxo é alcançado por meio da aplicação e remoção de corrente elétrica na bobina, realizada por sistemas de controle externo. Amplamente utilizadas em diversas aplicações, as válvulas solenoides são reconhecidas por sua rápida operação, confiabilidade e precisão no controle do fluxo de fluidos (ANDRADE, 2020).

Figura 11 – Válvula Solenoide.



Fonte: Disponível em: Mercado Livre (2024)

As características dadas pelo fabricante são as seguintes:

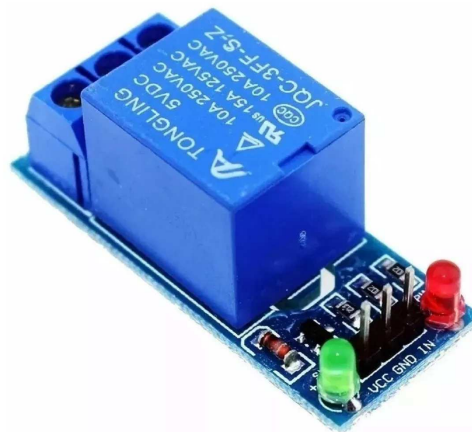
- Tensão de operação: 12V DC;
- Corrente de operação: 0,4 A;
- Funcionamento: Como uma chave (aberta ou fechada);
- Modelo: Normalmente Fechada
- Filtro: Para permitir somente a passagem de fluídos;
- Rosca de Entrada/Saída: 3/4";
- Pressão de operação: 0,02 Mpa à 0,8 Mpa
- Pressão máxima: 1,2 Mpa por 5 min, sem deformação ou vazamento.
- Vazão mínima: 2L/min (à 0,02 Mpa);
- Vazão máxima: 20L/min (à 0,8Mpa);
- Dimensões: 73x57x30mm;
- Peso: 86g.

Será necessário uma fonte 12 V no circuito para acionamento da válvula solenoide.

4.3.3 MÓDULO RELÉ

É um dispositivo que controla o fluxo de corrente elétrica em um circuito ao receber um sinal elétrico de entrada. Ao ser energizado, o relé aciona um interruptor que pode abrir ou fechar os contatos do circuito. Este tipo de módulo é amplamente utilizado em aplicações de automação para controlar dispositivos elétricos, como lâmpadas, motores, eletrodomésticos, válvulas solenoides e entre outros. A tensão de 5V refere-se à tensão de alimentação necessária para o funcionamento do relé. Compatível com diferentes sistemas de controle, incluindo microcontroladores como o ESP8266, o módulo relé oferece uma solução simples e eficaz para automatizar e controlar dispositivos elétricos de forma remota.

Figura 12 – Módulo relé.



Fonte: Disponível em: Mercado Livre (2024)

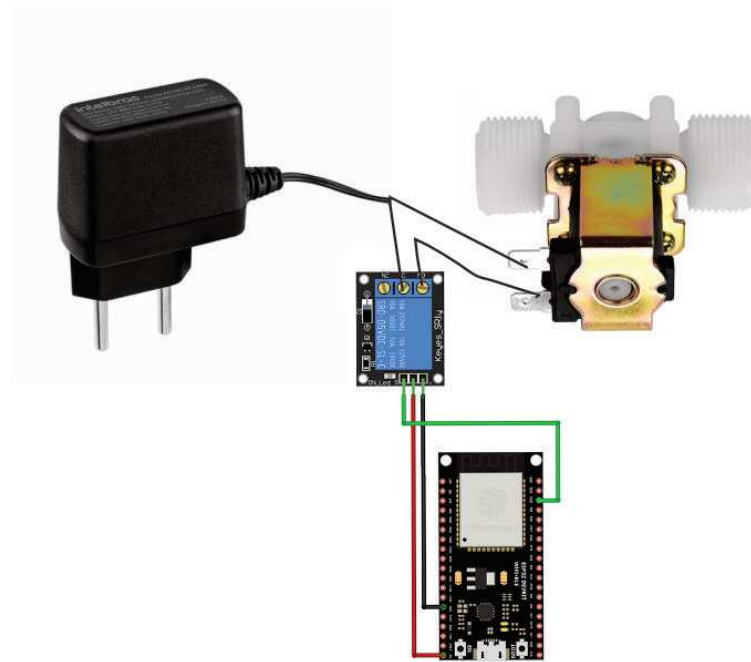
As especificações do fabricante são mostradas a seguir:

- Tensão de operação: 5 V;
- Tensão de entrada recomendado: 220 V;
- Dimensões: 10 mm x 10 mm x 10 mm;
- Frequência de Relógio: 60 Hz;
- Peso: 20 g;
- Quantidade de pinos digitais de entrada e saída: 4.

4.3.4 DIAGRAMA ESQUEMÁTICO

A construção do protótipo deste projeto utilizou os componentes detalhados nas subseções 4.2.1 à 4.2.5. Conforme pode ser visualizado no diagrama esquemático do sistema embarcado apresentado na Figura 13, foi conectado nas *GPIO*'s do NodeMCU, o relé de acionamento por conseguinte a válvula solenoide com a alimentação de 12 V.

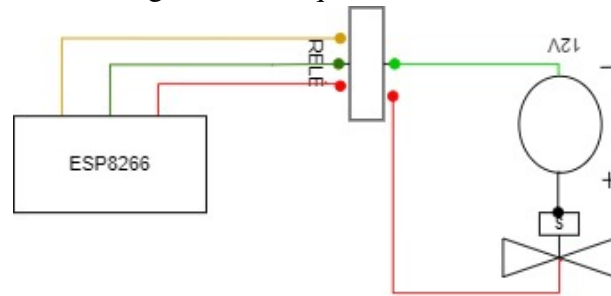
Figura 13 – Diagrama Esquemático.



Fonte: Autor (2024)

No diagrama esquemático representado na Figura 13 a GPIO conecta o fio verde no relé, que fio de pulso para acionamento do mesmo. Os demais fios do sensor representam o GND (terra), fio preto, e o fio vermelho a alimentação (5 Vcc). No relé tem-se um fio comum que leva da fonte à válvula solenoide, esse fio se conecta a uma chave normalmente aberta que quando for acionada irá abrir a chave interna da válvula solenoide. Destaca-se que o contato da válvula solenoide é normalmente fechado, isso quer dizer que a válvula quando energizada permite a passagem do fluido.

Figura 14 – Esquema Elétrico.



Fonte: Autor (2024)

Na Figura 14 é mostrado um esquema do circuito elétrico com a presença do ESP8266, relé, fonte e válvula solenoide.

4.4 FERRAMENTAS DE SOFTWARE

Nesta seção, são descritas as ferramentas de software empregadas no desenvolvimento do sistema embarcado deste projeto. Isso inclui a linguagem de programação e o ambiente de desenvolvimento utilizados no microcontrolador, bem como o foi realizado o protocolo de comunicação com a internet-hardware, bem como será apontado os softwares utilizados para testes de API e desserialização de dados JSON.

4.4.1 LINGUAGEM C

A linguagem de programação foi desenvolvida inicialmente por Dennis Ritchie na década de 1970 na Bell Labs, sediada em Murray Hill, Nova Jérсия. Essa linguagem é conhecida por sua eficiência, portabilidade e flexibilidade, sendo amplamente utilizada em sistemas operacionais, compiladores, dispositivos embarcados e aplicativos de software de baixo nível (BERTOLINI *et al.*, 2019). C é uma linguagem de médio nível, o que significa que oferece recursos de linguagens de alto nível, como abstração de dados e funções, enquanto também permite manipulação direta de hardware e memória.

Para isso, o autor ressalta dois conceitos intrínsecos desse método, que são: Sintaxe e Semântica. Sendo análogo ao português, sintaxe é forma de escrita da sentença e semântica é o significado. Além disso, para escrever um programa, necessita-se de um compilador, que é um tradutor da linguagem de programação que permita que o computador entenda e execute a linguagem objeto.

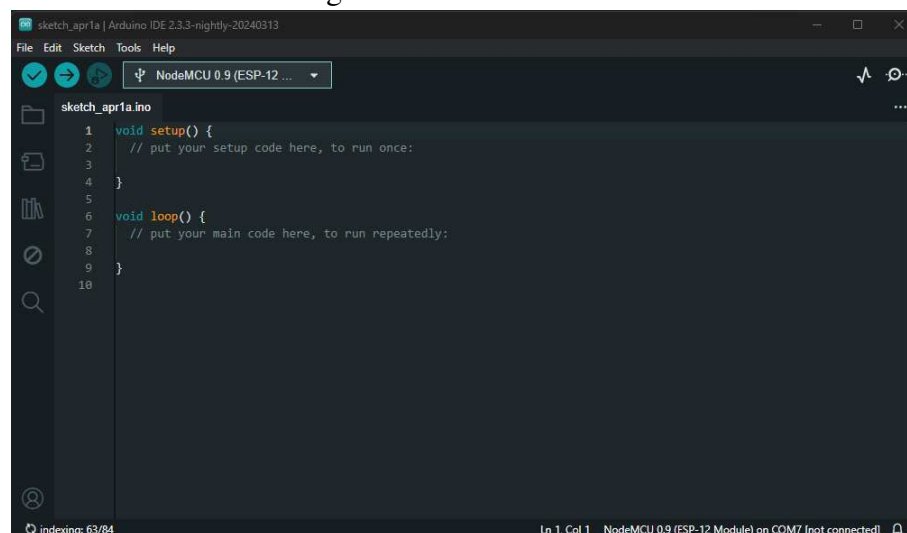
4.4.2 AMBIENTE DE DESENVOLVIMENTO

O ESP8266 é um microcontrolador compatível com o IDE do Arduino, logo, pode-se usar o mesmo *software* para a programação de componentes. Assim, o Ambiente Integrado de Desenvolvimento Arduino (IDE - *Integrated Development Environment*) é uma plataforma que permite escrever e fazer upload de programas nas linguagens C e C++ em placas compatíveis com esta plataforma. Estes programas são escritos no próprio editor de texto contido na IDE, sendo salvos em arquivos .ino, formando o que chamam de *sketches* (RIBEIRO; PORTARI, 2020). Existem duas funções essenciais para o desenvolvimento de um *sketch*, "*void setup* " e "*void loop*".

A função *void setup* é executada uma vez. Realiza a iniciação de componentes ou variáveis, além de configurar as GPIOs e declarar bibliotecas.

A função *void loop* é uma função que não retorna nada e é executada em um loop infinito. Isso significa que o código dentro dessa função será executado continuamente, repetindo-se indefinidamente, a menos que explicitamente seja interrompido por algum comando dentro do código ou por uma condição de parada. Estas funções são apresentadas na Figura 15.

Figura 15 – IDE Arduino.



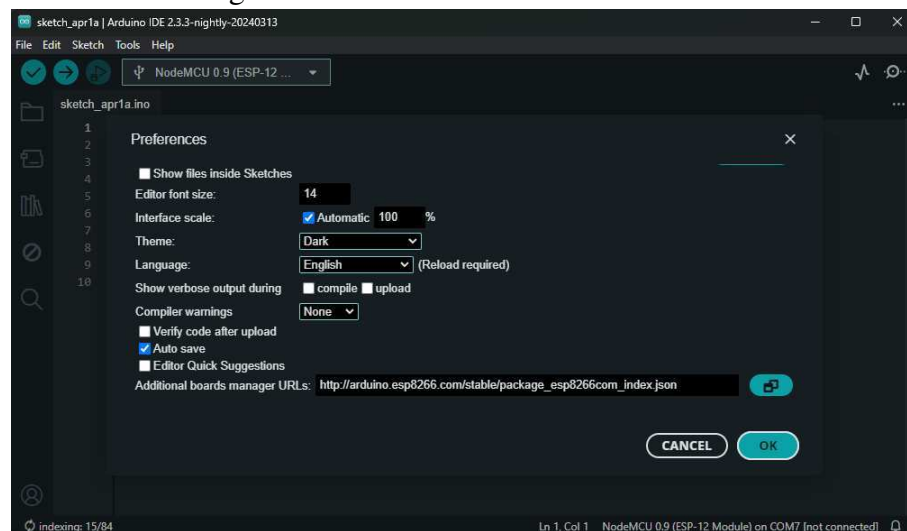
Fonte: (ARDUINO, 2024)

O Monitor Serial do Arduino é uma janela da IDE que é responsável por apresentar e receber os dados em formato de texto. Sendo possível a verificação do código através de códigos direcionador a mostrar algum conteúdo no Monitor Serial.

Para a utilização do microcontrolador ESP8266 na IDE do Arduino, existem diversos tutoriais na internet que mostra que é possível programar e compilar qualquer NodeMCU ou

qualquer módulo controlador da família ESP, pois as bibliotecas são compatíveis e permitem a utilização de vários componentes comumente usados pelo Arduino. Para preparar o ambiente para utilizar o ESP8266, realizam-se os seguintes comandos: Arquivo > Preferências > Configurações > URL Adicionais para Gerenciadores de Placas e adicionar a URL <http://arduino.esp8266.com/stable/package_esp8266com_index.json>. A janela "Preferências" é apresentada na Figura 16.

Figura 16 – Janela "Preferências" da IDE.



Fonte: Autoria própria (2024)

4.4.3 BIBLIOTECAS E FUNÇÕES

Para a realização da programação do microcontrolador NodeMCU ESP8266, necessita-se de incluir bibliotecas que servem tanto para comunicação com a internet como operações como tratamentos de dados de entradas/saídas. As bibliotecas utilizadas nesse trabalho são apresentadas no Quadro 2.

Quadro 2 – Bibliotecas.

Biblioteca	Descrição
ESP8266WiFi.h	Esta biblioteca fornece rotinas específicas de Wi-Fi ESP8266 que é utilizada para se conectar à rede.
ESP8266HTTPClient.h	Fornece os métodos para enviar solicitações HTTP.
ArduinoJson.h	A biblioteca ArduinoJson.h simplifica a manipulação de dados JSON permitindo análise de dados estruturados em dispositivos embarcados com recursos limitados.
math.h	Fornece funções matemáticas para operações como trigonometria, exponenciação, logaritmos, raiz quadrada e outras, sendo essencial para cálculos complexos e avançados em programas em C e C++.
NTPClient.h	é uma ferramenta para sincronização de tempo utilizando o protocolo Network Time Protocol (NTP).
WiFiUDP.h	é utilizada em conjunto com a NTPClient.h para estabelecer a comunicação UDP necessária para sincronizar o tempo do dispositivo com um servidor NTP. O protocolo NTP utiliza o UDP para troca de mensagens entre o dispositivo e o servidor NTP.

Fonte: (ARDUINO, 2024)

4.4.4 PROTOCOLOS HTTP E HTTPS

O protocolo HTTP foi utilizado para a integração do sistema embarcado com o sistema *web*. Esse protocolo é a base de comunicação na web e é usado para transferir uma variedade de recursos, como páginas da web, imagens, vídeos, arquivos, APIs. Ele oferece alguns métodos de trocas de informações, como por exemplo o *GET*, *POST*, *PUT* e entres outras. No Quadro 3 são apresentados a descrição dos métodos mencionados (WEILER; OTTEQUIR; BECKER, 2017).

Quadro 3 – Quadro de Métodos HTTP.

Método	Descrição
<i>GET</i>	Requisita dados de um recurso específico.
<i>POST</i>	Submete diversos dados para serem processados por um recurso específico.
<i>PUT</i>	Adiciona (ou modifica) um recurso na URI (Identificador Uniforme de Recurso) passada.
<i>DELETE</i>	Exclui um recurso específico.
<i>HEAD, OPTIONS, TRACE</i>	Mesma requisição de um GET, retorno de cabeçalhos HTTP e dentre outros.

Fonte: Weiler, Ottequir e Becker (2017)

Na sua estrutura de trocas de informações, são padronizados alguns códigos de status

que são indicados para resultados de solicitações. Alguns desses podem ser vistos as seguir:

- 200 OK: Indica que a solicitação foi bem-sucedida;
- 404 Not Found: Indica que o recurso solicitado não foi encontrado no servidor;
- 500 Internal Server Error: Indica que ocorreu um erro no servidor ao processar a solicitação;
- 302 Found: Indica que a solicitação foi redirecionada para outro URL.

É necessário que haja uma padronização entre o *servidor* e o microcontrolador, assim, o ESP8266 realiza uma requisição HTTP que o fará receber os dados, o método para isso segundo Schmitz *et al.* (2017) é o GET. Uma vez que está recebido, o microcontrolador será preparado para executar as informações de irrigação, que representa o objetivo deste trabalho.

O HTTPS é essencial para garantir a segurança das comunicações que utilizam o protocolo HTTP na Internet. HTTPS (*Hypertext Transfer Protocol Secure*) é um protocolo de comunicação usado para transferir dados pela internet de forma segura. Ele é uma extensão do HTTP, mas com uma camada adicional de segurança fornecida pelo protocolo SSL/TLS (*Secure Sockets Layer/Transport Layer Security*), os certificados SSL garantem aos usuários que estão se comunicando com o site legítimo e não com um impostor (FIORENZA *et al.*, 2020).

4.4.5 ARDUINO JSON

JSON é um padrão de formatação de dados para troca de informações entre sistemas. Consiste em uma coleção de pares chave/valor. Utiliza uma integração de esquemas, que é o casamento das equivalências dos dados em análise. Esse processo é dividido em fases e atividade e também é definido artefatos de entrada e de saída. São exemplos de fases "pré-processamento dos documentos, a análise das similaridades textuais dos elementos, verificação das equivalências das instâncias e a abordagem da estrutura hierárquica dos elementos contidos nos documentos JSON."(PADILHA, 2020).

Aliado a isso, a plataforma Arduino detém de um aplicativo assistente para "desserializar"o documento JSON. A desserialização é o processo de converter dados serializados em uma estrutura de dados utilizável em um programa de computador. Quando se recebe dados serializados, eles geralmente estão em um formato que não é diretamente utilizável pelo seu programa. A desserialização é o processo de interpretar esses dados e convertê-los em uma forma que o programa possa entender e manipular (ARDUINO, 2024). Para isso, utiliza-se uma biblioteca do Arduíno para ESP8266 chamada ArduinoJson.h.

Para realizar esse processo, configura-se o aplicativo para o tipo de microcontrolador que

está sendo utilizado, no qual pode-se escolher o modo de desserialização, uma vez que pode-se filtrar apenas para as variáveis escolhidas e o tipo de entradas de documentos. Esse processo de configuração é mostrado na Figura 17.

Figura 17 – Passo 1.

The image shows the 'Step 1: Configuration' screen of the ArduinoJson Assistant. At the top, there is a progress bar with four steps: 1. Configuration (highlighted in teal), 2. JSON, 3. Size, and 4. Program. Below the progress bar, the main configuration area has a teal header 'Step 1: Configuration'. It contains three dropdown menus: 'Processor' set to 'ESP32', 'Mode' set to 'Deserialize and filter', and 'Input type' set to 'Stream'. Below the 'Input type' dropdown, there is a note: 'This is the most memory efficient option, but not the fastest; see this if your program is slow.' At the bottom of the configuration area, there is a green box with a lightbulb icon and the text: 'This is the Assistant for ArduinoJson 6.17.3. Make sure the same version is installed on your computer.' In the bottom right corner, there is a teal button labeled 'Next: JSON'.

Fonte: (ARDUINO, 2024)

O próximo passo é colar o documento JSON, na qual na aplicação do presente trabalho é lido no serial monitor do Arduino uma vez que realiza o método GET em comunicação com o servidor. Um exemplo de documentação JSON é mostrado na Figura 18.

Figura 18 – Passo 2.

Step 2: JSON

Examples: [OpenWeatherMap](#), [Reddit](#)

Input

```
{
  "sensor": "gps",
  "time": 1351824120,
  "data": [
    48.75608,
    2.302038
  ]
}
```

Input length: 87 [Prettify](#)

[Previous](#) [Next: Size](#)

Fonte: (ARDUINO, 2024)

O passo 3 refere-se a dados que o próprio aplicativo retorna como estrutura dos dados, formato do dados, o filtros utilizados e a memória que será utilizada do microcontrolador. Como é mostrado na Figura 19.

Figura 19 – Passo 3.

Step 3: Size

Data structures	4496	Bytes needed to stores the JSON objects and arrays in memory
Strings	511	Bytes needed to stores the strings in memory
Filter	11	The parser temporarily stores some ignored keys; this is the size of the largest one.
Total (minimum)	5018	Minimum capacity for the <code>JsonDocument</code> .
Total (recommended)	6144	Including some slack in case the strings change, and rounded to a power of two.

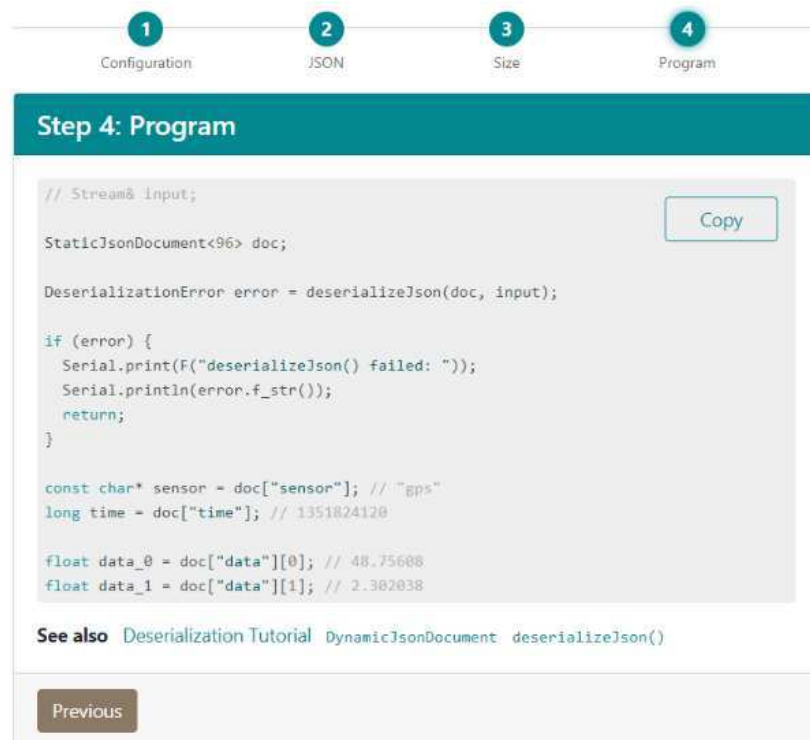
► Tweaks (advanced users only)

[Previous](#) [Next: Program](#)

Fonte: (ARDUINO, 2024)

O último passo é receber o código da desserialização. Com isso, é possível manipular os dados como fazer operações, apresentar no serial monitor e entre outros. É necessário verificar os tipo que está sendo armazenado a documentação. Como por exemplo na Figura 20.

Figura 20 – Passo 4.

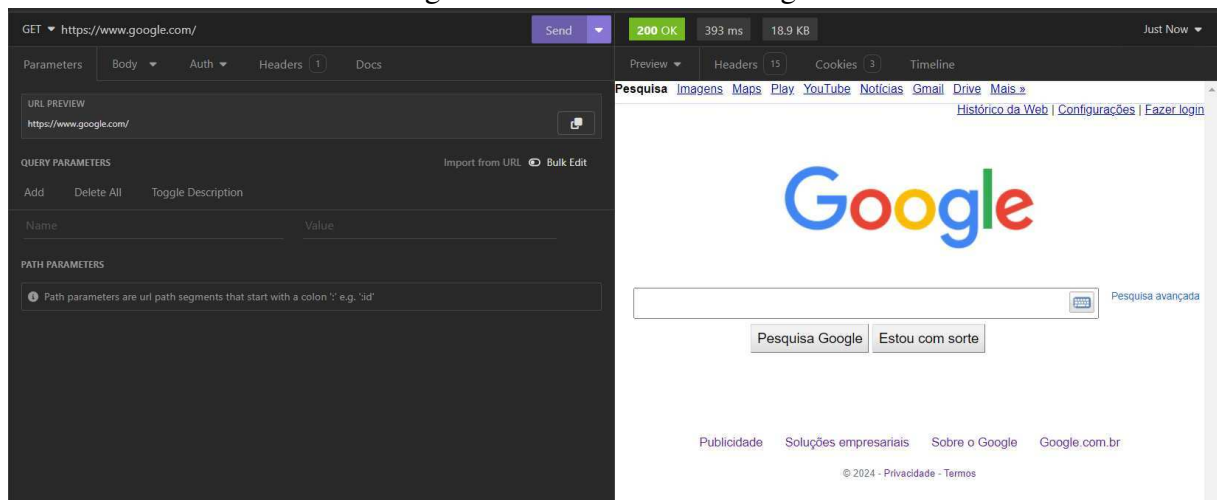


Fonte: (ARDUINO, 2024)

4.4.6 INSOMNIA

O *software* Insomnia tem como serventia automatizar testes manuais de APIs. Sendo um aplicativo de código aberto, ele permite simplificar processos de desenvolvimentos de API (*Application Programming Interfaces*) (REGES, 2021). O Insomnia oferece facilidade na realização de testes em APIs, permitindo que os desenvolvedores criem e enviem solicitações HTTP e visualizem as respostas correspondentes em tempo real, como por exemplo documentações JSON.

Figura 21 – Teste de API Google.



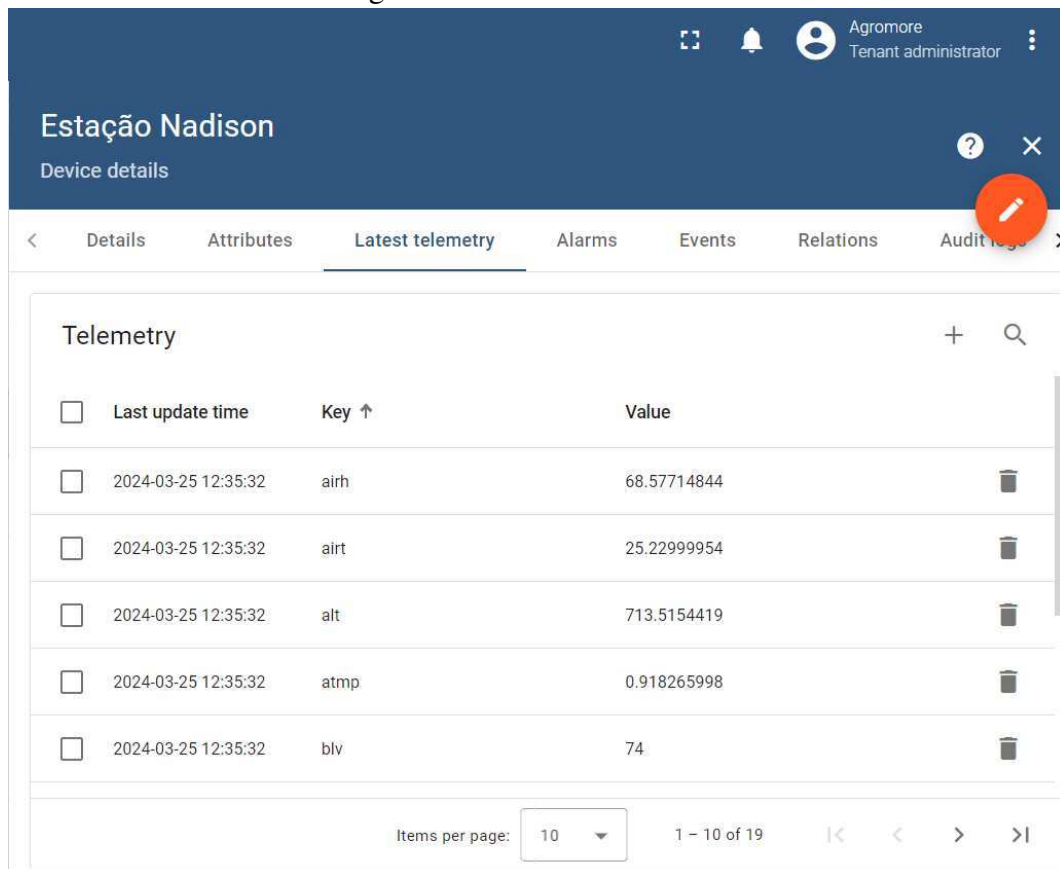
Fonte: Autor (2024)

A Figura 21, mostra que foi possível realizar a requisição GET e receber o conteúdo da API do Google. Além disso, é possível perceber o "200 OK" no canto superior direito, esse número é um resultado padrão das requisições HTTP que apresenta validação na requisição feita. Outras coisas que podem ser destacadas são algumas abas que permitem a construção de uma API, como por exemplo acionar *tokens* e *headers* (ALMEIDA, 2023).

- Tokens: Os tokens são usados para autenticar e autorizar solicitações HTTP em uma API (ALMEIDA, 2023);
- Headers: Os cabeçalhos são partes das solicitações HTTP que fornecem informações adicionais sobre a solicitação ou o cliente que está fazendo a solicitação. Eles incluem informações como o tipo de conteúdo da solicitação, codificação, autenticação, entre outras (ALMEIDA, 2023).

A API era referente ao servidor ThingsBoard que direciona para os dados de telemetria, ou seja, os dados de velocidade do vento, umidade, radiação solar, umidade do solo e entre outros. Na Figura 22 apresenta como esses valores são apresentados na tela do servidor. Esses dados são enviados da estação para o servidor também no formato JSON, então é possível acessá-los também na documentação JSON, como mostrado na subseção anterior. Os dados no servidor são apresentados na Figura 22.

Figura 22 – Tela de telemetria.



☐	Last update time	Key ↑	Value	
☐	2024-03-25 12:35:32	airh	68.57714844	
☐	2024-03-25 12:35:32	airt	25.22999954	
☐	2024-03-25 12:35:32	alt	713.5154419	
☐	2024-03-25 12:35:32	atmp	0.918265998	
☐	2024-03-25 12:35:32	blv	74	

Items per page: 10 1 – 10 of 19

Fonte: Autor (2024)

Para a utilização da API para as requisições, foi utilizado o Ngrok, de forma a se fazer um servidor paralelo para a realização dos testes. O Ngrok é uma ferramenta que permite criar caminhos seguros entre a internet pública e seu ambiente de desenvolvimento local (VIEIRA *et al.*, 2022). Para utilizar o Ngrok, primeiro é copiado o token disponibilizado pelo ThingsBoard, o que permite que seja acessado os dados de telemetria endereçados do servidor. Assim, um outro token é gerado pelo Ngrok para vincular o endereço IP do servidor local ao endereço IP e porta que o usuário poderá acessar via Internet. Após completar essas etapas, o Ngrok entrará em operação, fornecendo uma URL (em HTTP) de acesso público que deverá ser compartilhada com o cliente interessado em acessar a página web. Essa URL é utilizada para fazer as requisições a partir do microcontrolador através da IDE.

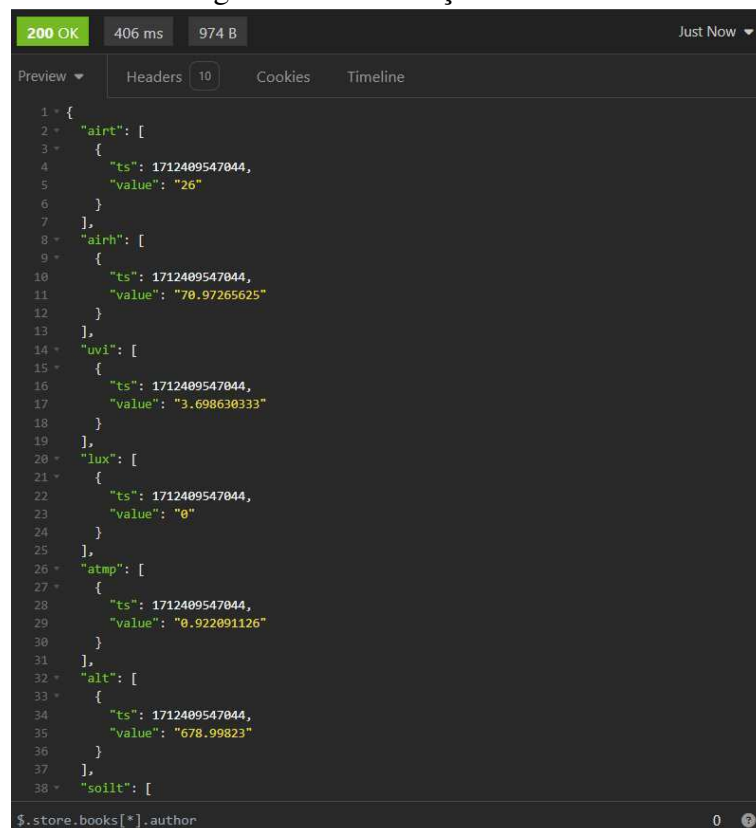
5 RESULTADOS E DISCUSSÕES

Neste Capítulo, são apresentados os resultados obtidos com o funcionamento do sistema de irrigação de precisão desenvolvido, bem como as telas de integração do sistema embarcado e o sistema *web* são apresentadas através da API.

5.1 VERIFICAÇÃO DA API

A verificação da API foi realizada no Insomnia, que serviu para entender quais esquemas JSON são encontrados. Essa verificação foi importante para comparar se os dados enviados para o ESP8266 estão no chaveamento correto. Além disso, serviu para testar momentos em que o servidor ThingsBoard estava fora do ar ou com alguma requisição mal feita. Conforme a Figura 23 vê-se um teste realizado para verificação do envio dos dados do servidor. É importante salientar que os dados de API, tokens e certificações foram preservados.

Figura 23 – Verificação de API.



Fonte: Autor (2024)

A Figura 23 pode ser comparada com a Figura 22 que apresenta os valores das telemetria

feita e apresenta as variáveis que foram monitoradas na Estação Meteorológica. Com isso, é possível utilizar dessa API para as requisições feitas a partir do ESP8266.

Nos testes de conexão com a API, foi percebido alguns códigos que o servidor retornava, esses códigos são resposta às solicitações realizada por um cliente. São padrões das solicitações HTTP, onde auxiliam os desenvolvedores para corrigir erros presentes no projeto tanto do servidor, quanto na forma de comunicação entre o microcontrolador e o servidor. Alguns desses códigos de verificação são apresentados no quadro abaixo.

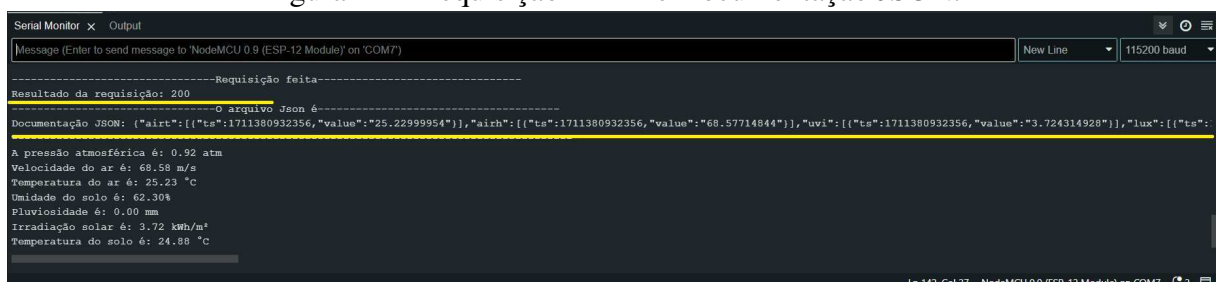
Quadro 4 – Códigos de Verificação HTTP.

Código HTTP	Definição
HTTP Code 200 OK	Envio de dados realizado com sucesso.
HTTP Code 400	Dados inválidos.
HTTP Code 404	Dados não encontrados
HTTP Code 500	Erro interno no servidor
HTTP Code 503	O servidor está temporariamente incapaz de realizar solicitação devido a sobrecarga ou manutenção.

Fonte: Autor (2024)

A comunicação entre o sistema embarcado e o sistema *web* foi implementada por meio da biblioteca "ESP8266HTTPClient.h", a qual permite fazer a coleta dos dados via ESP8266. Inicialmente é inserido a *login* e senha da rede *Wi-Fi* do usuário dentro da biblioteca supracitada. Após isso é verificado se há conexão. Uma vez conectado, foi feita a requisição da documentação JSON através do método GET, essa requisição é feito através da biblioteca HTTPClient.h. Uma imagem dessas requisição é apresentada na Figura 24.

Figura 24 – Requisição HTTP e Documentação JSON.



Fonte: Autor (2024)

Conforme a Figura 24, é possível verificar o resultado "200" da requisição HTTP, sendo possível obter o documento JSON. Essa documentação, contém todas as variáveis que são enviadas da estação para o servidor ThingsBoard, por isso, são recebidas diversas variáveis que seriam necessárias para o código.

5.2 DESSERIALIZAÇÃO DO DOCUMENTO JSON

Assim, fez-se uma breve seleção do que é necessário para a aplicação. Isso pode ser feito com a ajuda do software ArduinoJson, que, ao ser disponibilizado a documentação JSON, é possível selecionar o que é viável para a aplicação. Isso é boa conduta para que não use memória desnecessária do microcontrolador.

Tendo em vista tudo isso que foi comentado, é realizado a seleção das variáveis que serão utilizadas Figura 25.

Figura 25 – Seleção de variáveis.

The screenshot shows the 'Step 2: JSON' interface of the ArduinoJson JSON Filter tool. It consists of three main panels: 'Input', 'Filter', and 'Filtered input'.

- Input Panel:** Contains a text area with a JSON document. The JSON is:


```
{
  "airt": [
    {
      "ts": 1711380932356,
      "value": "25.22999954"
    }
  ],
  "airh": [
    {
      "ts": 1711380932356,
```

 Below the text area, it says 'Input length: 1606'. There is a 'Prettify' button.
- Filter Panel:** Contains a text area for a filter. The filter is:


```
{
  "airt": [
    {
      "value": true
    }
  ],
  "airh": [
    {
      "value": true
    }
  ],
```

 There is a 'Prettify' button.
- Filtered input Panel:** Shows the result of applying the filter. The JSON is:


```
{
  "airt": [
    {
      "value": "25.22999954"
    }
  ],
  "airh": [
    {
      "value": "68.57714844"
    }
  ],
  "uvi": [
```

At the bottom, there are 'Previous' and 'Next: Size' buttons.

Fonte: Autor (2024)

Como mostrado, no quadro esquerdo da Figura 25 estão todas as variáveis completas, onde estão dados que não serão necessário para a utilização que propõe-se. O quadro do meio mostra um filtro, que deseja-se que o JSON extraia somente o valor da variável, como por exemplo o "airt", que é a variável documentada que representa a temperatura do ar, e no nosso filtro, selecionamos como "true" somente o valor que está alocado na variável. O restante das variáveis seguiram o mesmo curso. Feito isso, o quadro na parte direita da imagem mostra como fica o nosso objeto JSON depois da filtragem.

Com isso, recebe-se do software, um código somente com as variáveis que necessitará no projeto onde estará pronto para fazer a desserialização (Figura 26).

Figura 26 – Código de seleção de variáveis a partir do JSON.

Step 4: Program

```
// Stream& input;

StaticJsonDocument<480> filter;
filter["airt"][0]["value"] = true;
filter["airh"][0]["value"] = true;
filter["uvi"][0]["value"] = true;
filter["lux"][0]["value"] = true;
filter["atmp"][0]["value"] = true;
filter["alt"][0]["value"] = true;
filter["soilt"][0]["value"] = true;
filter["rainf"][0]["value"] = true;
filter["soilh"][0]["value"] = true;
filter["wdsp"][0]["value"] = true;

StaticJsonDocument<768> doc;

DeserializationError error = deserializeJson(doc, input, DeserializationOption::Filter(filter));

if (error) {
    Serial.print(F("deserializeJson() failed: "));
    Serial.println(error.f_str());
    return;
}

const char* airt_0_value = doc["airt"][0]["value"]; // "25.22999954"

const char* airh_0_value = doc["airh"][0]["value"]; // "68.57714844"

const char* uvi_0_value = doc["uvi"][0]["value"]; // "3.724314928"

const char* lux_0_value = doc["lux"][0]["value"]; // "0"

const char* atmp_0_value = doc["atmp"][0]["value"]; // "0.918265998"
```

Fonte: Autor (2024)

Posteriormente fez-se algumas alterações nesse código, como por exemplo o tipo de variável que está sendo alocado os valores. Como é possível observar na Figura 26, foram associadas algumas variáveis como ponteiro "const char*", o que não é viável para fazer operações algébricas. Com isso, altera-se o tipo dessas variáveis para o "float". Feito isso, foi possível compilar esse código no ESP8266, o resultado da desserialização é apresentado na Figura 27.

Figura 27 – Variáveis desserializadas.

```
Serial Monitor x Output
Message (Enter to send message to 'NodeMCU 0.9 (ESP-12 Module)' on 'COM7')
New Line 115200 baud

-----Requisição feita-----
Resultado da requisição: 200
-----O arquivo Json é-----
Documentação JSON: {"airt":[{"ts":1711380932356,"value":"25.22999954"}],"airh":[{"ts":1711380932356,"value":"68.57714844"}],"uvi":[{"ts":1711380932356,"value":"3.724314928"}],"lux":[{"ts":1711380932356,"value":"0"}],"atmp":[{"ts":1711380932356,"value":"0.918265998"}]}

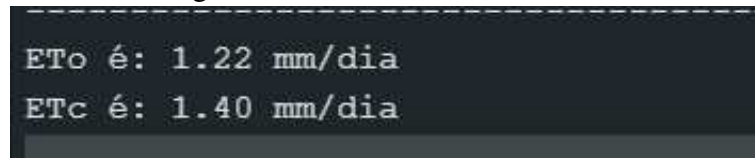
A pressão atmosférica é: 0.92 atm
Velocidade do ar é: 68.58 m/s
Temperatura do ar é: 25.23 °C
Umidade do solo é: 62.30%
Pluviosidade é: 0.00 mm
Irradiação solar é: 3.72 kWh/m²
Temperatura do solo é: 24.88 °C
```

Fonte: Autor (2024)

5.3 CÁLCULO DE ETO E ETC

O cálculo da evapotranspiração de referência e de evapotranspiração de cultura foi realizado no microcontrolador pela equações 4.1 e 4.7 respectivamente. E um exemplo de aplicação é apresentado na Figura 28. Esses valores são calculados a partir de uma última telemetria enviada do servidor da estação. A estação que foi buscada esses dados se encontrava em fase de testes e não estavam no campo buscando os valores reais em sua totalidade visto que alguns sensores não estavam alocados no seu respectivo objeto de sensoramento, por isso os valores apresentados na Figura podem ser diferente dos valores de alguma outra estação alocada no lugar.

Figura 28 – Cálculo de ETo e ETc.



```
ETo é: 1.22 mm/dia
ETc é: 1.40 mm/dia
```

Fonte: Autor (2024)

É válido destacar que algumas unidades da estação meteorológica estavam diferentes da padronização dada por Conceição (2006), isso faz com que seja necessário aplicar conversões de unidades para que o cálculo de ETo e ETc esteja coerente com o roteiro visto a necessidade de precisão nos dados para vislumbrar da economia de água, por exemplo.

5.4 A IRRIGAÇÃO

A partir disso, a irrigação pode ser determinada pela equação 4.10. O momento da irrigação é determinada pelo usuário que irá apresentar valores de horas e minutos para que seja efetuada a irrigação em algum momento do dia, isso será feito de acordo com a necessidade da cultura ser irrigada, pois existem culturas que precisam de mais água que outras. O tempo de irrigação está em função da ETc e também de outros parâmetros que são necessários para esse cálculo como por exemplo a área de irrigação, ou seja, área onde será implantada a cultura e também da vazão dos gotejadores ou aspersores. Uma vez que é feito os cálculos, o microcontrolador faz a verificação do horário e manda o sinal para o relé permitir a passagem de água pelo tempo calculado. Com isso, é mostrado também, no monitor serial o tempo de irrigação (Figura 29).

Figura 29 – Tempo de Irrigação.

```

Pluviosidade é: 0.00 mm
Irradiação solar é: 3.70 kWh/m²
Temperatura do solo é: 25.50 °C
-----
ETo é: 1.20 mm/dia
ETc é: 1.38 mm/dia
-----
Tempo de irrigação: 2 minutos e 54 segundos
15:24:27
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado!
15:24:30
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado!
15:24:33

```

Fonte: Autor (2024)

De acordo com as Figuras 31 e 30 pode-se ver a verificação da hora, que nesse exemplo, para fins de amostragem, foi definida a hora 05h39min. A biblioteca NTP permite que o microcontrolador esteja verificando a horário de algum servidor na internet, o servidor utilizado é um servidor brasileiro. Assim, o microcontrolador, através da biblioteca NTP, verifica o horário e também irá verificar o horário que foi definido para irrigação, com isso, se não é um momento de irrigação escolhido o microcontrolador não irá mandar sinal para o relé, e se for o horário escolhido ele irá avisar no serial monitor que é um momento de irrigação e começa a irrigação pelo tempo calculado.

Figura 30 – Definição de momento de irrigação - irrigará.

```

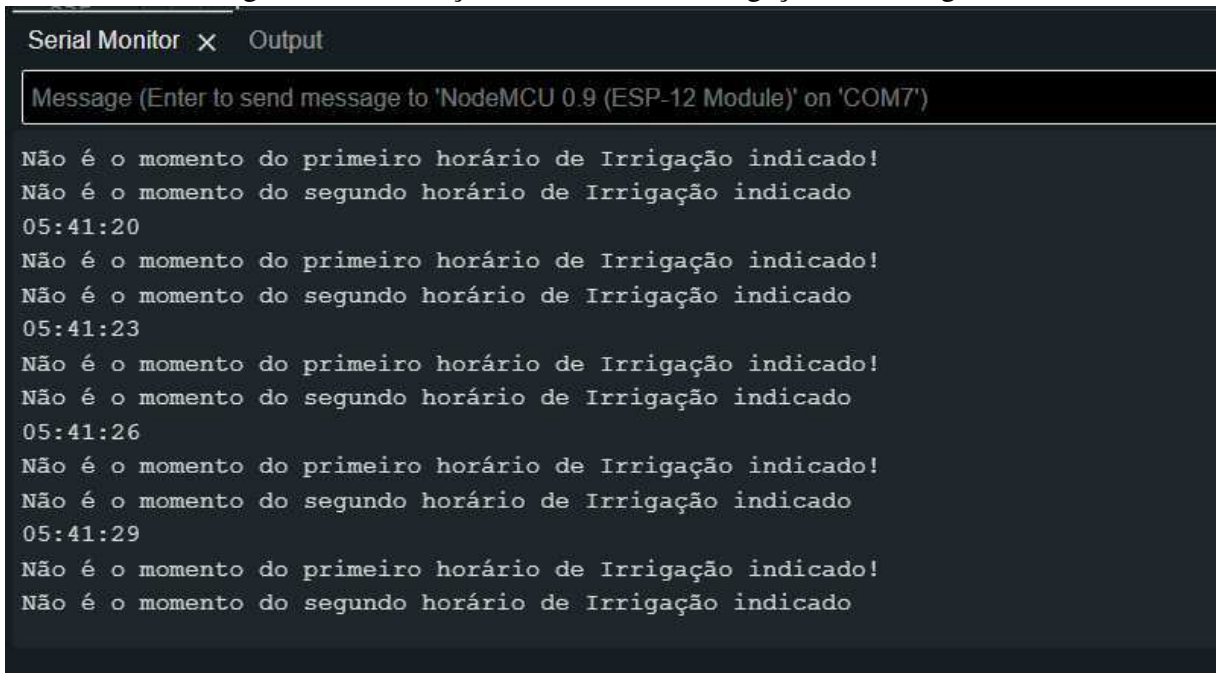
Serial Monitor x Output
Message (Enter to send message to 'NodeMCU 0.9 (ESP-12 Module)' on 'COM7')

Não é o momento do primeiro horário de Irrigação indicado!
Hora da Irrigação!
05:39:08
Não é o momento do primeiro horário de Irrigação indicado!
Hora da Irrigação!
05:39:11
Não é o momento do primeiro horário de Irrigação indicado!
Hora da Irrigação!
05:39:14
Não é o momento do primeiro horário de Irrigação indicado!
Hora da Irrigação!
05:39:17
Não é o momento do primeiro horário de Irrigação indicado!
Hora da Irrigação!

```

Fonte: Autor (2024)

Figura 31 – Definição de momento de irrigação - não irrigará.

The image shows a screenshot of a 'Serial Monitor' application window. The title bar includes 'Serial Monitor' and a close button 'x', followed by 'Output'. Below the title bar is a text input field containing the message: 'Message (Enter to send message to 'NodeMCU 0.9 (ESP-12 Module)' on 'COM7')'. The main area of the window displays a series of text messages in a monospaced font. The messages are: 'Não é o momento do primeiro horário de Irrigação indicado!', 'Não é o momento do segundo horário de Irrigação indicado', and a timestamp '05:41:20'. This sequence of three lines is repeated four times, with the timestamps '05:41:23', '05:41:26', and '05:41:29' appearing after the second message of each repetition. The messages indicate that the current time is not within the scheduled irrigation periods.

```
Serial Monitor x Output
Message (Enter to send message to 'NodeMCU 0.9 (ESP-12 Module)' on 'COM7')
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado
05:41:20
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado
05:41:23
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado
05:41:26
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado
05:41:29
Não é o momento do primeiro horário de Irrigação indicado!
Não é o momento do segundo horário de Irrigação indicado
```

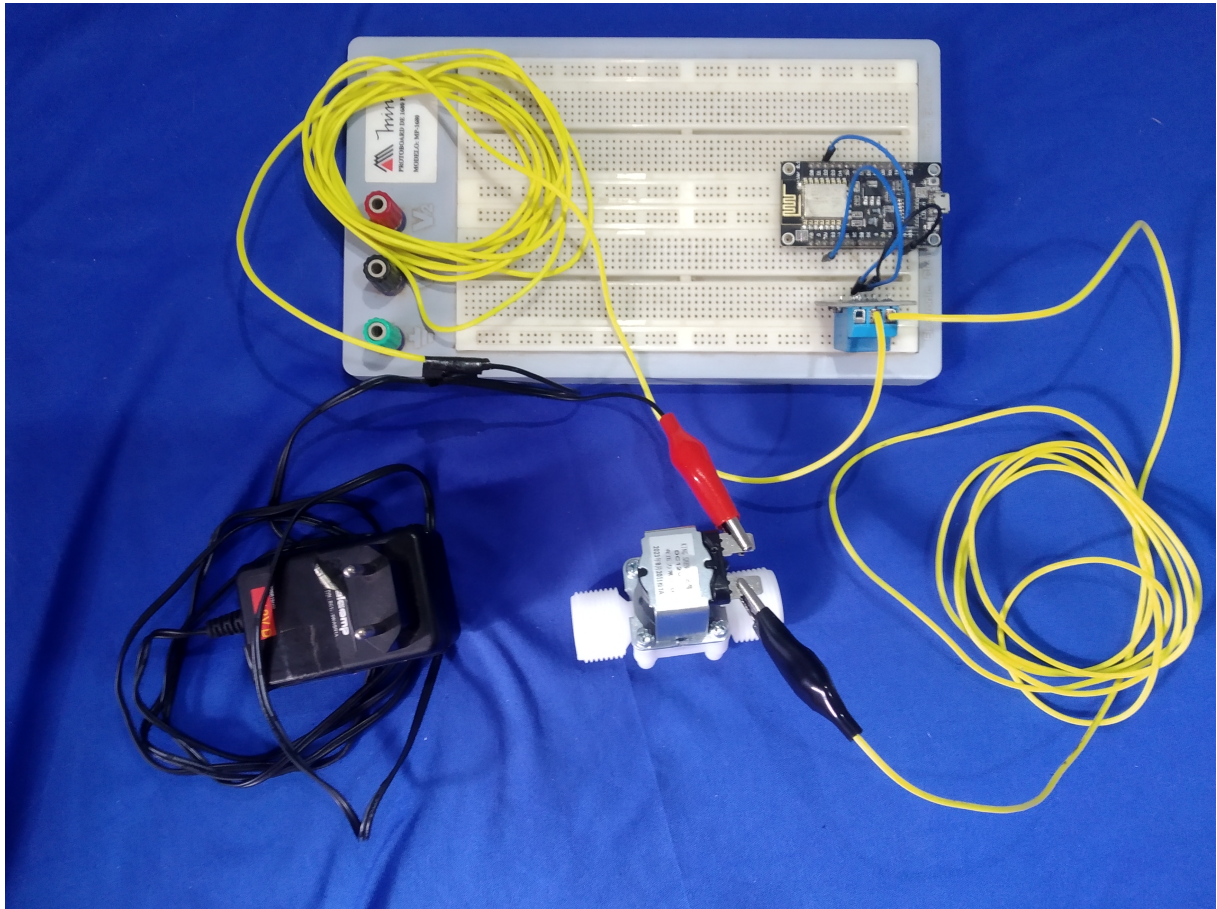
Fonte: Autor (2024)

Nessas figuras, apresentam duas mensagens pois é disponível até dois momento de irrigação no dia, mas é possível aumentar ou diminuir a quantidade de momentos em um dia.

5.5 MONTAGEM DO CIRCUITO

O circuito montado é apresentado nas Figuras 32 e 33. Nota-se a presença do microcontrolador ESP8266, do relé (5 V) e da Válvula Solenoide (12 V). Como já comentado, a válvula necessita de alimentação 12 V para seu funcionamento. O relé pode ser alimentado com até 5V na sua parte de comando e sua parte de força alimentado pelo mesmos 12V da fonte de alimentação.

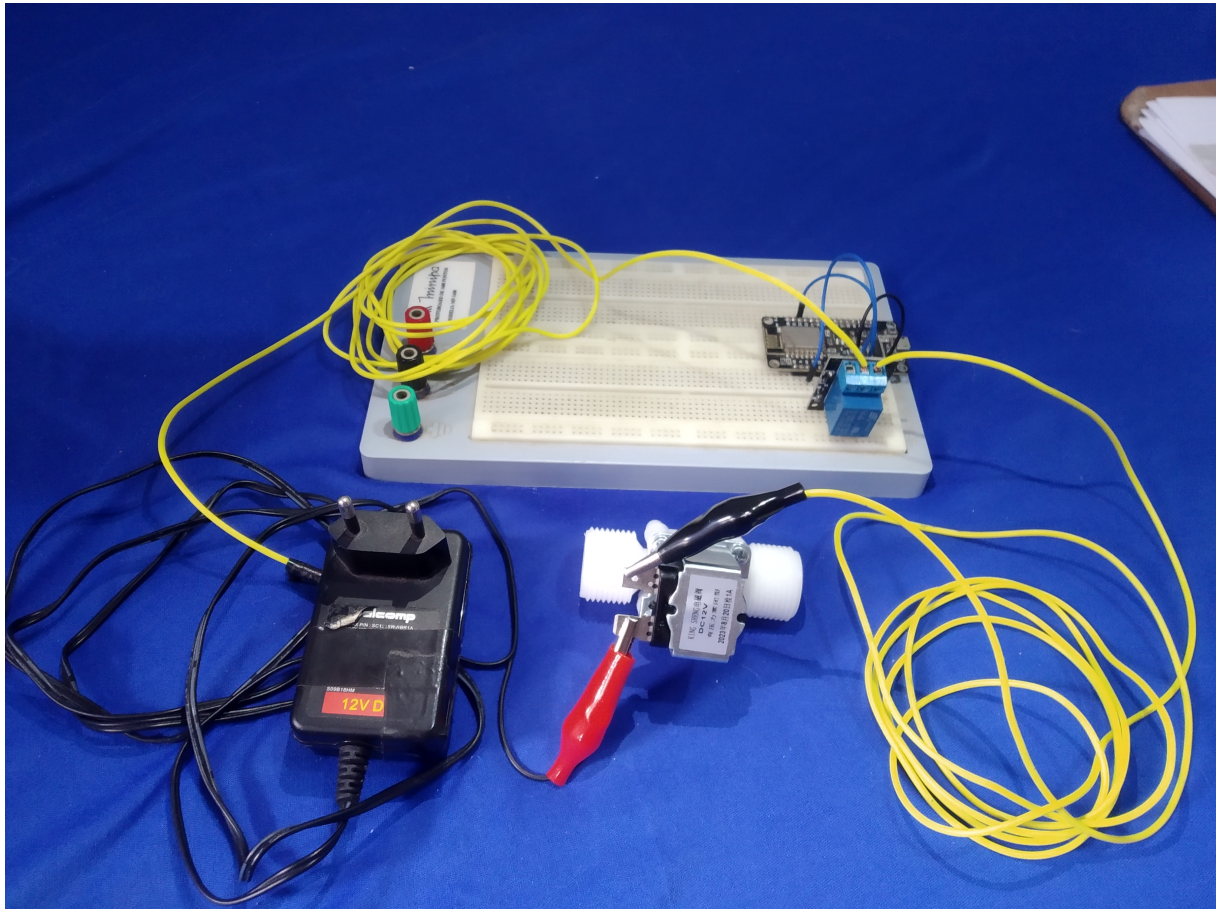
Figura 32 – Foto do circuito montado 1.



Fonte: Autor (2024)

A parte de comando do relé é feita pelo microcontrolador, que, quando receber um sinal alto (5 Vcc) ele irá induzir uma tensão na bobina para que ela feche o contato normalmente aberto, na parte de força do relé, fazendo com que feche o circuito referente a fonte de tensão e relé.

Figura 33 – Foto do circuito montado 2



Fonte: Autor (2024)

5.6 MONTAGEM DO SISTEMA DE IRRIGAÇÃO

Essa montagem primou-se em conectar um cano com gotejadores a uma caixa d'água no qual a passagem da mesma depende da válvula solenoide. Essa montagem foi com caráter de experimento, uma vez que os dados que fazem os cálculos de ET_o e ET_c são de uma outra localidade. Para a realização, foram aproximados valores de vazão do gotejamento, o valor de K_c para o cultivo de feijão e um dado horário (Figuras 34 e 35).

Figura 34 – Experimento montado 1.



Fonte: Autor (2024)

Foi percebido a necessidade de se calcular a pressão dentro do gotejador a fim de detectar um balanço entre vazão e pressão para que o sistema não fique tão sobrecarregado com forças internas. Para isso, existem dispositivos que são autocompensadores de pressão que poderá ser colocado no sistema.

Figura 35 – Experimento montado 2.



Fonte: Autor (2024)

O sistema em si precisa de uma alimentação de tensão própria para o ESP8266 e a válvula solenoide, isso pode ser suprido com a utilização de baterias de 5 e 12 V, de forma a evitar a utilização de notebook para aplicação (Figura 36). Destaca-se que implementando esses sistema dentro da própria estação meteorológica, uma vez que ela é alimentada com energia solar com um mini módulo fotovoltaico de 9 V de tensão, com isso é possível fazer utilizando reguladores de tensão como conversores CC-CC, que possam aumentar ou diminuir a tensão CC para a alimentação dos dispositivos.

Figura 36 – Experimento montado 3.



Fonte: Autor (2024)

6 CONCLUSÃO

O presente trabalho foi desenvolvido no contexto de irrigação de precisão, no intuito de fornecer mais uma utilidade para a estação meteorológica desenvolvida para evitar desperdício de água. A aplicação de um método de determinação da evapotranspiração de referência padronizado pela FAO permite uma segurança na validação do processo de irrigação e que seja de uma forma inteligente; o cliente precisa somente moldar os padrões necessários para funcionamento de acordo com as necessidades da espécie agrícola cultivada, os dados que precisão ser informados são: disponibilidade hídrica, área a ser irrigada, coeficiente de cultivo da espécie agrícola e horário do dia para irrigação.

Além disso, destaca-se que o acréscimo dos novos dispositivos não levarão aumentos significativos no custo do produto final e permite a viabilidade de cliente adquirir o produto, com um valor médio de R\$ 60,00. Isso mantém a essência do trabalho de Silva (2021), uma vez que sua aplicação visou ser monitoramento de baixo custo para ser acessível ao agricultor.

Na realização do trabalho foi notado que o sistema embarcado fica muito vulnerável ao servidor na qual ele se comunica, essa vulnerabilidade é notada em momentos que o servidor fica fora do ar, ou até mesmo momentos em que a rede de internet sofre com oscilações e queda, fazendo com que o ESP8266 não realize suas atividades por indisponibilidade de internet. Assim, faz-se necessário que os códigos que contém os cálculos, comandos e monitoramento do horário sejam compilados dentro da própria estação meteorológica desenvolvida.

Para isso, necessita modificar a estrutura da estação meteorológica para poder alimentar a válvula solenoide com 12 V. A própria estação tem uma alimentação por geração de um módulo solar e um banco de baterias, no entanto, as duas alimentações são de 9 V e não é suficiente para o acionamento da válvula. Além disso, motivo desse projeto não ter sido desenvolvido na própria estação foi que um dos modelos que estavam disponíveis para testes de modificações apresentou problemas em seu funcionamento, por isso, os testes realizados foram com dados que estavam sendo enviados de outro modelo de estação que estavam em testes em um local diferente do no local onde foram feitos os testes de irrigação aplicado nesse trabalho.

Como trabalhos futuros, indica-se a criação de um novo design de estação, onde seja projetada para comando de irrigação e que seja alocados componentes que permitam alimentar com a tensão necessária para acionamento da válvula. Ademais, os dados podem ser processados no próprio servidor e o microcontrolador da estação receber apenas o comando de quanto tempo é necessário para irrigação, o que diminui a quantidade de memória exigida do dispositivo.

REFERÊNCIAS

- ALBUQUERQUE, P. de; COELHO, E. Planilha para obtenção de coeficiente de cultura (kc) para culturas de ciclo anual, segundo método fao, para as condições climáticas brasileiras. Sete Lagoas: Embrapa Milho e Sorgo., 2021.
- ALMEIDA, E. Programação orientada a serviços. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, 2023.
- ANDRADE, F. S. Irrigação automatizada: sistemas de irrigação temporizado. 196, 2020.
- ARDUINO. 2024. Disponível em: <<https://www.arduino.cc>>.
- BERNARDO, S.; SOARES, A. A.; MANTOVANI, E. C. **Manual de irrigação**. Viçosa: UFV, 2006.
- BERTOLINI, C.; PARREIRA, F. J.; CUNHA, G. B. d.; MACEDO, R. T. Linguagem de programação i. Brasil, 2019.
- Blog Agromove. **Irrigação de precisão: como funciona e quais são suas vantagens**. 2020. Disponível em: <<https://blog.agromove.com.br/irrigacao-precisao/>>.
- BORGES, T. d. S. Sistema para automação e manejo da irrigação de baixo custo. 2021.
- BRITO, J. S. Sistema automatizado e expansível para irrigação rural. Cachoeiro de Itapemirim, 2023.
- COIN, J. P. **Os fluxos de água virtual nas exportações dos setores do agronegócio brasileiro**. 2022.
- CONCEIÇÃO, M. A. Roteiro de cálculo da evapotranspiração de referência pelo método de penman-monteith-fao. **Embrapa Uva e Vinho. Circular Técnica**, v. 65, 2006.
- CONSIGLIERI, L. M. Confecção de uma estação meteorológica de baixo custo para fins educacionais. Universidade Estadual Paulista (Unesp), 2023.
- DACAL, T. M. L. A.; CATALDI, M.; SOARES, A. B.; ALT, A. N.; SALCEDO, I. L.; LIMA, L. G. de; VINHA, J. V. M. V. da; ROSILLO, E. C.; GUIMARÃES, T. M. Estação meteorológica de baixo custo. **Biodiversidade Brasileira**, v. 9, n. 1, p. 199–199, 2019.
- DARÉ, J. C. **Caracterização da demanda de água em sistemas de irrigação na Bacia do Córrego Sossego em Itarana - ES**. 2013. 28 nov. Disponível em: <biblioteca.incaper.es.gov.br>.
- FERNANDES, A.; MAGALHÃES, I. N.; SOUZA, M.; JUNIOR, A. C.; MOREIRA, C. Sistema de aquisição de sinais ecg processado pelo labview com comunicação wi-fi por meio do módulo esp8266. **Revista Principia - Divulgação Científica e Tecnológica do IFPB**, v. 1, n. 34, p. 62–68, 2017.
- FERNANDES, D. S.; HEINEMANN, A. B.; PAZ, R. L. F.; AMORIM, A. d. O. Calibração regional e local da equação de hargreaves para estimativa da evapotranspiração de referência. **Revista Ciência Agronômica**, SciELO Brasil, v. 43, p. 246–255, 2012.
- FIORENZA, M. M.; KREUTZ, D.; ESCARRONE, T.; TEMP, D. Uma análise da utilização de https no brasil. In: SBC. **Anais do XXXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos**. [S.l.], 2020. p. 966–979.

GALDINO, S. M.; SOUSA, R. M. R. de; COSTA, S. M. G.; FERREIRA, J. P. C. Estudo prospectivo de tecnologias potenciais para a solução da crise hídrica no Brasil. **Cadernos de Prospecção**, v. 11, p. 198, 2018.

GUNDIM, A. d. S.; MELO, V. G. M. L. d.; COELHO, R. D.; SILVA, J. P. d.; ROCHA, M. P. A. d.; FRANÇA, A. C. F.; CONCEIÇÃO, A. M. P. d. Tendências e perspectivas da irrigação de precisão: uma revisão. **Ciência Rural**, v. 53, 2023.

JIAO, L.; LU, N.; FANG, W.; LI, Z.; WANG, J.; JIN, Z. Determining the independent impact of soil water on forest transpiration: A case study of a black locust plantation in the loess plateau, China. **Journal of hydrology**, v. 572, p. 671–681, 2019.

JIN, Z.; YOU, Q.; ZUO, Z.; LI, M.; SUN, G.; PEPIN, N.; WANG, L. Weakening amplification of grassland greening to transpiration fraction of evapotranspiration over the Tibetan plateau during 2001–2020. **Agricultural and Forest Meteorology**, v. 341, p. 109661, 2023.

KADAR, H. H.; RAFEE, P. A. A.; SAMEON, S. S. Internet of things (IoT) and water crisis. In: **2018 4th International Conference on Computer and Information Sciences (ICCOINS)**. [S.l.: s.n.], 2018. p. 1–6.

KAMIENSKI, C.; VISOLI, M. Swamp: uma plataforma para irrigação de precisão baseada na Internet das Coisas. Fonte, Belo Horizonte, v. 15, n. 20, p. 76–84, dez. 2018., 2018.

LUZ, D. R. RAMOS da; SOUSA, J. M. D.; SILVA, A. de F.; SOUZA, A. M. FERNANDES de; BARRETO, A. C. S.; SANTOS, P. R.; ROCHA, T. A. L. C. G.; OST, A. Desenvolvimento de uma estação meteorológica de baixo custo localizada no IFPA-campus paraúpebas. **Revista Foco (Interdisciplinary Studies Journal)**, v. 16, n. 9, 2023.

MANNRICH, J. P.; JR, O. da R. N. Proposta de estação meteorológica baseada em Internet das Coisas de código aberto e baixo custo como ferramenta para articulação entre novas tecnologias digitais da informação e comunicação no ensino de ciências/física na educação básica. **Revista Novas Tecnologias na Educação**, v. 19, n. 2, p. 323–334, 2021.

OLIVEIRA, P. J. D. d. Irrigação de precisão para a cultura de cebola. Universidade Estadual Paulista (Unesp), 2018.

PADILHA, R. J. Um processo para casamento de esquemas de documentos JSON baseado na estrutura e nas instâncias. Universidade Federal de Santa Maria, 2020.

PAOLINELLI, A.; NETO, D. D.; MANTOVANI, E. C. **Irrigação de pastagens**. Piracicaba: ESALQ, 2022. 397 p. Il. Disponível em: <https://www.esalq.usp.br/biblioteca/pdf/4.Agricultura_Irigada_no_Brasil-ci%C3%Aancia_e_tecnologia.pdf>.

PEREIRA, L. B.; SANTOS, M. R. dos. Desenvolvimento de um sistema embarcado de baixo custo para automatização do manejo de irrigação localizada. **Revista Velho Chico**, v. 2, n. 1, p. 50–56, 2022.

REGES, L. C. M. Api mossoró+: Api REST para uma plataforma voltada à centralização de eventos educacionais. UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE – UERN, 2021.

RIBEIRO, T. B.; PORTARI, S. C. J. Tomada inteligente, reativa à sensores, com utilização da ESP8266 para IoT. 2020.

SANCHES, B. C.; CARVALHO, E. S.; GOMES, F. F. B. A indústria 4.0 e suas contribuições à sustentabilidade. **Revista Engenharia e Tecnologia Aplicada-UNG-Ser**, v. 2, n. 1, p. 48–55, 2019.

SANTOS, B. P.; SILVA, L. A.; CELES, C.; BORGES, J. B.; NETO, B. S. P.; VIEIRA, M. A. M.; VIEIRA, L. F. M.; GOUSSEVSKAIA, O. N.; LOUREIRO, A. Internet das coisas: da teoria à prática. **Minicursos SBRC-Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos**, v. 31, 2016.

SANTOS, M. D.; SILVA, F. LIMA da; REIS, M. F. d.; WALKER, R. A.; QUINTAL, R. S. Desenvolvimento de um modelo para armazenamento hídrico: Contribuições da engenharia para o uso racional da água na agricultura. **Tecno-Lógica**, v. 26, n. 1, 2022.

SCHMITZ, M. A. *et al.* Comunicação bidirecional para plataforma embarcada do protegemed. Universidade de Passo Fundo, 2017.

SILVA, A. C. M. da; SILVA, M. A. da; MIRANDA, J. P. L. de; PINTO, M. C.; FACEROLI, S. T. *et al.* Estação meteorológica automática de baixo custo-low cost automatic meteorological station. **Multiverso: Revista Eletrônica do Campus Juiz de Fora-IF Sudeste MG**, v. 1, n. 1, p. 46–56, 2016.

SILVA, N. F. d. Desenvolvimento de uma estação meteorológica de baixo custo para agricultores do semiárido nordestino. Universidade Federal Rural do Semi-Árido, 2021.

SOUZA, S. A. **Avaliação do potencial de aumento de produtividade e economia de água e energia na irrigação de precisão**. 109 p. Dissertação (Dissertação) — Universidade Federal de Viçosa, Viçosa, 2021. Mestrado em Engenharia Agrícola.

SRIVASTAVA, P.; BAJAJ, M.; RANA, A. S. Iot based controlling of hybrid energy system using esp8266. In: **2018 IEEMA Engineer Infinite Conference (eTechNxT)**. [S.l.: s.n.], 2018. p. 1–5.

THOMSEN, A. 2016. Disponível em: <<https://www.filipeflop.com/blog/esp8266-nodemcu-como-programar/>>.

VIEIRA, F.; COSTA, M.; CAMPOS, G.; OLIVEIRA, F. Utilização das tecnologias flask, ngrok e arduino no acionamento remoto de dispositivos eletroeletrônicos através da internet. **Anais do Encontro de Computação do Oeste Potiguar ECOP/UFERSA (ISSN 2526-7574)**, n. 6, p. 17–20, 2022.

WEILER, D.; OTTEQUIR, D.; BECKER, F. W. Protótipo para controle de nível e vazão de água utilizando conceito de internet das coisas em reservatórios hídricos de condomínios. Faculdade Senac Blumenau, 2017.

Wikipedia. **Evapotranspiração**. 2024. Disponível em: <<https://pt.wikipedia.org/wiki/Evapotranspira%C3%A7%C3%A3o>>.

YANG, Y.; RODERICK, M. L.; GUO, H.; MIRALLES, D. G.; ZHANG, L.; FATICHI, S.; LUO, X.; ZHANG, Y.; MCVICAR, T. R.; TU, Z. Evapotranspiration on a greening earth. **Nature Reviews Earth & Environment**, Nature Publishing Group UK London, v. 4, n. 9, p. 626–641, 2023.