



UNIVERSIDADE PRESBITERIANA MACKENZIE

Faculdade de Computação e Informática



Sistema de Irrigação Automático

Felipe Carvalho, Felipe Kazuyoshi, Wilian França Costa

¹Universidade Presbiteriana Mackenzie (UPM)

Rua da Consolação, 930 Consolação, São Paulo - SP, 01302-907 – Brazil

{fccruz00@gmail.com, Felipekkdasilva@hotmail.com, wilian.costa@mackenzie.br

Abstract. *This article describes a project in development of IoT and its application in agriculture. After researching the difficulty that farmers suffer in irrigating their plantations, we came up with the idea of automating this process through a low-cost architecture, notifying the farmer of the humidity of the soil, the air humidity and the temperature. This project will allow for easy control of the farmer's land in real time without the need for labor, it will only be necessary to monitor the system to gain insight of its land, and in case of any defect in any of the systems which a maintenance may be necessary.*

Resumo. *Este artigo descreve um projeto em desenvolvimento de IoT e sua aplicação em agricultura. Depois de pesquisar a dificuldade que agricultores sofrem em irrigar suas plantações, surgimos com a ideia de automatizar este processo através de uma arquitetura de baixo custo, notificando o agricultor a umidade do solo, do ar e a temperatura. Este projeto permitira a facilidade do controle das terras do agricultor em tempo real sem necessidade de mão-de-obra, apenas será necessário o monitoramento do sistema para insights sobre o seu terreno e caso haja algum defeito em algum dos sistemas e seja necessário a manutenção do mesmo.*

1. Introdução

O IoT (Internet of Things) vem mudando como a computação e vista e utilizada no dia a dia das pessoas, com seu poder de automação essa tecnologia vem impactando na vida de diversos setores e de diversas pessoas. Apesar de ser uma tecnologia muito poderosa, além de ser custoso para implementar um projeto com sucesso, requer uma infraestrutura muito complexa de ser implementada (envolvendo conexão de rede, protocolos utilizados, segurança, arquitetura).

Levando tudo isso em conta, e o fato que um dos maiores setores consumidores dessa tecnologia ser a agricultura, resolvemos aceitar o desafio de um grande problema deste setor, que seria a irrigação. A proposta é explorar os diversos sensores relacionados a umidade do ar, solo, temperatura para poder identificar se há necessidade de irrigar o solo e notificar o agricultor, utilizando o protocolo mqtt para estabelecer uma conexão entre o sistema e o aplicativo mobile (onde todos os dados serão disponibilizados para visualização em tempo real). Podendo, assim, automatizar um processo que requer muita mão-de-obra com o menor custo e com a menor necessidade de manutenção do sistema.

2. Materiais e Métodos

Agora nós vamos entrar com mais detalhes na questão de como todo o projeto foi realizado, incluindo quais dispositivos foram utilizados, protocolos utilizados, softwares utilizados, código desenvolvido e como foram colocados juntos para criar a sinergia necessária para entregar a solução.

2.1 Materias Utilizados

Começando pelos dispositivos e produtos utilizados para realizar a montagem, nós fizemos uma lista com tudo que foi utilizado, suas especificações mais importantes junto com o link para poder entrar mais detalhes nas características do produto e para dar uma referência de onde comprá-los, segue:



Figura 1. WiFi / Bluetooth ESP32S CP2102 30 pinos

Microcontrolador que possui conectividade bluetooth e WiFi já integrados na placa que será utilizado para estabelecer a conexão com wifi/mqtt e substituir o Arduino uno. Para referência

do dispositivo, pode ser encontrado no link a seguir: <https://www.saravati.com.br/placa-esp32-wifi-bluetooth-esp32s-cp2102-30-pinos>

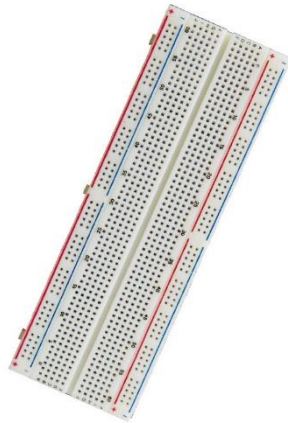


Figura 2. Protoboard 830 pontos MB-102

A protoboard serve como uma facilitadora na montagem dos circuitos eletrônicos, ajudando na fase de desenvolvimento do projeto. Usaremos ela para anexar o sensor de umidade e temperatura do ar. Para referência do dispositivo, pode ser encontrado no link a seguir: <https://www.baudaeletronica.com.br/protoboard-830-pontos-mb-102.html>



Figura 3. Fios Jumper Premium 40p x 20cm

Fios eletrônicos usados para conectar dois pontos de um circuito eletrônico, usaremos os fios para conectar o ESP32 ao protoboard, aos sensores e ao relé. Para referência do dispositivo, pode ser encontrado no link a seguir: Opção macho/macho - <https://www.baudaeletronica.com.br/jumper-premium-40p-x-20cm-macho-macho.html>; Opção macho/fêmea - <https://www.baudaeletronica.com.br/jumper-premium-40p-x-20cm-macho-femea.html>; Opção fêmea/ fêmea - <https://www.baudaeletronica.com.br/jumper-premium-40p-x-20cm-femea-femea.html>



Figura 4. Sensor de Umidade do Solo LM393 3,3V-5V

Uma das partes principais do projeto, serve para medir a umidade do solo, vai ser conectado ao ESP32 usando os fios jumper. Para referência do dispositivo, pode ser encontrado no link a seguir: <https://www.baudaeletronica.com.br/sensor-de-umidade-do-solo.html>



Figura 5. Sensor de Temperatura e Umidade do Ar DHT22

Sensor utilizado para medir a temperatura e a umidade do ar. Para referência do dispositivo, pode ser encontrado no link a seguir: https://www.usinainfo.com.br/sensor-de-umidade-arduino/sensor-de-umidade-e-temperatura-am2302-dht22--2813.html?search_query=dht22&results=5



Figura 6. Mangueira Tubo Cristal Transparente Pvc 8mm 5/16

A detailed view of a white, multi-ported mechanical component, likely a pump or valve, with a black cylindrical section and a black handle. The component has a complex, multi-faceted design with several ports and a central black handle. The text "usmairto" is visible in the background.

Figura 9. Fonte de Alimentação Chaveada 12VDC 1A

Fonte de alimentação, influencia diretamente na quantidade e força de água deslocada, recomendado de 9 a 15V. Para referência do dispositivo, pode ser encontrado no link a seguir: <https://www.usinainfo.com.br/fonte-de-alimentacao/fonte-de-alimentacao-chaveada-12vdc-1a-3082.html>



Figura 10. Adaptador J4 Fêmea 2,1mm para Borne

Adaptador utilizado na fonte de alimentação chaveada para poder conectar os fios no relé e na mini bomba de água. Para referência do dispositivo, pode ser encontrado no link a seguir: <https://www.baudaeletronica.com.br/adaptador-j4-femea-2-1mm-para-borne.html>



Figura 11. Cabo flexível 4mm

Cabo utilizado para conectar o relé e bomba de água no adaptador j4 fêmea 2,1 mm para Borne. Para referência do dispositivo, pode ser encontrado no link a seguir: https://www.leroymerlin.com.br/cabo-flexivel--4mm-15metros-verde-750v-lexman_87934070

2.2 Montagem

Tendo os produtos que utilizamos em mente, para realizar a montagem nós utilizamos um total de 5 portas, sendo elas: GND; 3,3V; D22 que está conectado no sensor DHT22; D23 que está conectado no Módulo Relé; VP que está conectado no sensor de umidade de solo, realizando o papel de um pino ADC (Analog Input). Nós possuímos dois diagramas, um de arquitetura e outro de circuito, para podermos entrar em mais detalhes de como foi realizada a montagem, segue:

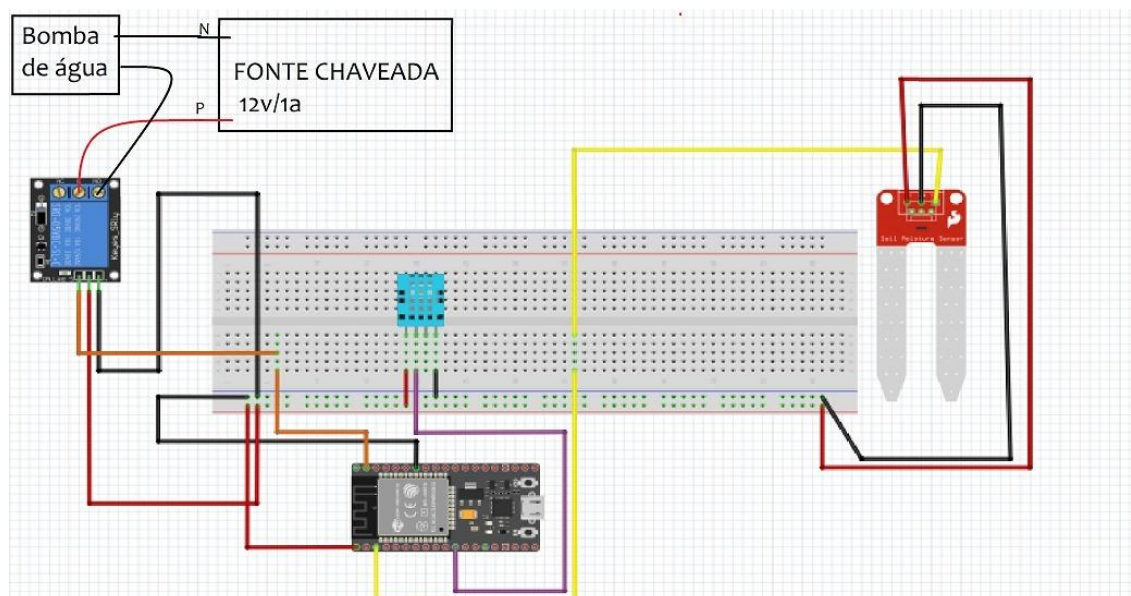


Figura 12. Diagrama de Arquitetura

Diagrama de arquitetura da montagem onde demonstra como foi realizada a montagem do sistema, notando que não há uma lealdade na forma como você vai conectar os sensores no ESP32, seja direto no dispositivo ou utilizando a protoboard, contanto que você nas portas mencionadas acima, você devera ter o mesmo resultado.

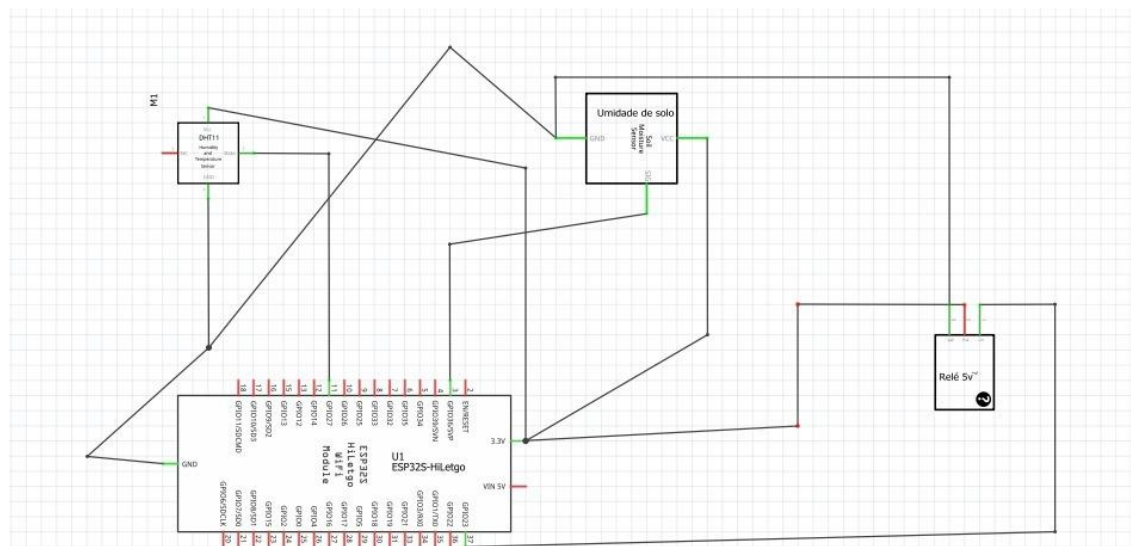


Figura 13. Diagrama de Circuito

Diagrama de circuito que mostra graficamente a representação do circuito elétrico da Figura 12.

2.3 Softwares, Protocolos utilizados

Nós utilizamos o protocolo TCP/IP para estabelecer a comunicação entre o dispositivo ESP32 e o WIFI, e utilizamos o protocolo MQTT para estabelecer a conexão entre o ESP32 e o Broker possibilitando o envio de dados para um dashboard com melhor visualização.

A respeito do MQTT, nós utilizamos o Node-Red com versão 1.2.2 acompanhado com a instalação do NodeJS com versão 14.15.0; O Broker que utilizamos foi o mosquitto com versão 1.6.12. Todas as portas utilizadas foram as que já chegam configuradas de forma padrão após a instalação.

Com estas informações em mente, podemos passar para a configuração do Node-Red e como está o deploy para que ele possa receber os Pubs/Subs que enviaremos para aplicação, segue:

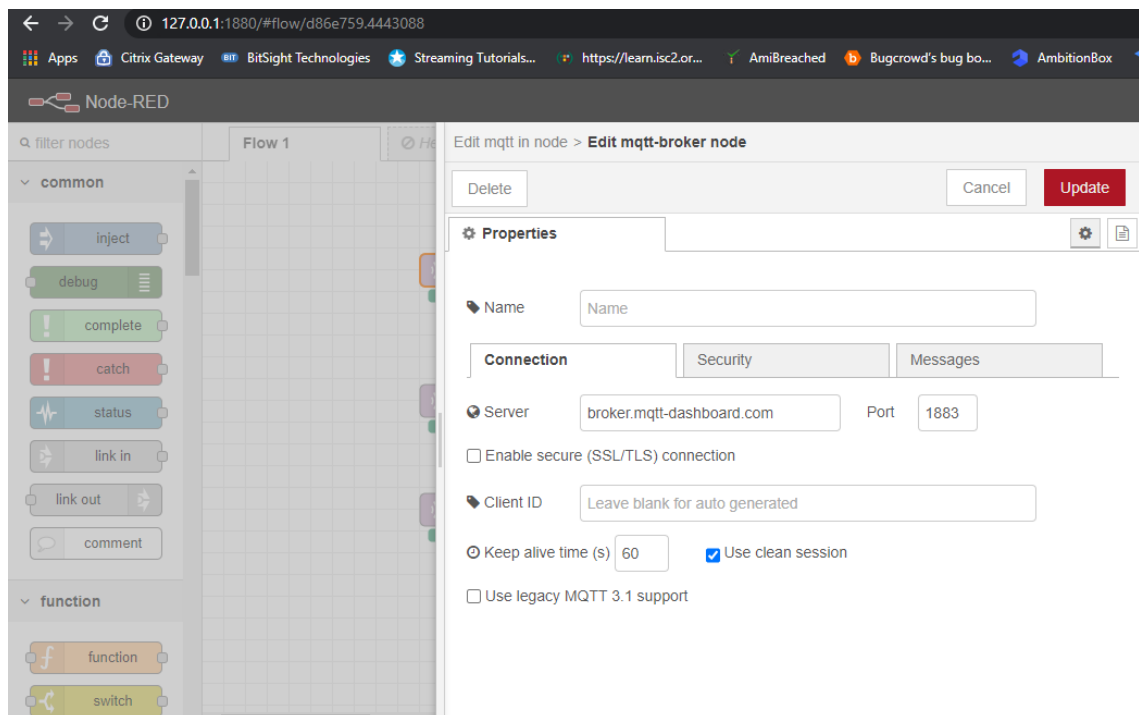


Figura 14. Configuração do Broker

Apesar de estarmos rodando tudo localmente (127.0.0.1), na configuração do nosso Broker nós vamos utilizar o servidor “broker.mqtt-dashboard.com” na porta padrão do Mosquitto 1883, esta configuração será realizada em todos os Subs que utilizaremos podendo assim receber os dados.

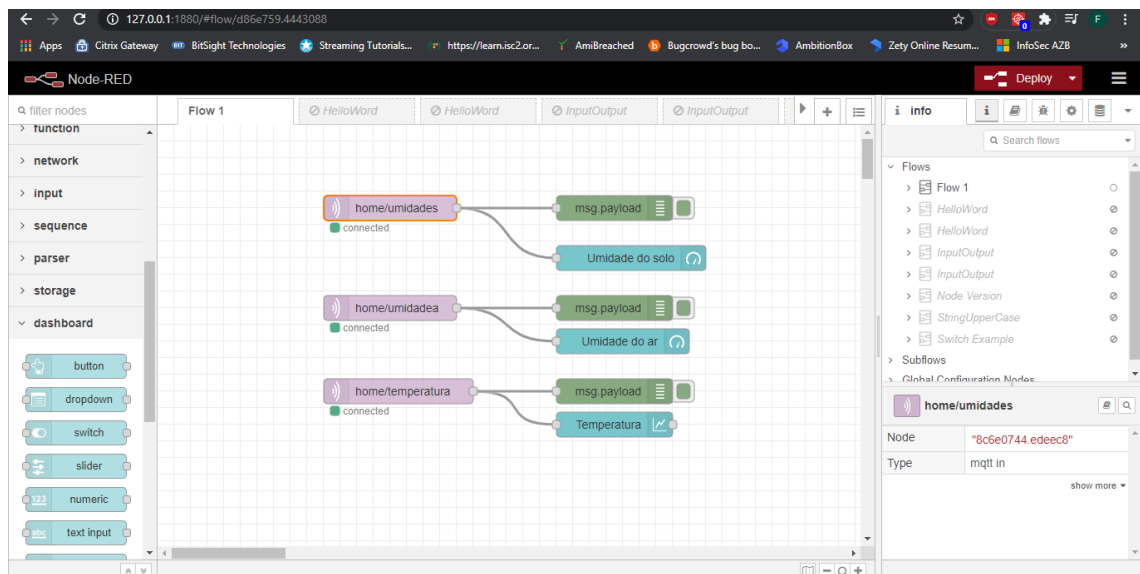


Figura 15. Visão geral do Deploy e como estão conectados os nós

Conforme dito na descrição da Figura 14, todos os Subs possuem a mesma configuração, a única coisa que difere são o nome dos tópicos utilizados, isso se dá a necessidade de enviar as informações para um gráfico apenas, isto será explicado com mais detalhes na parte do código.

2.4 Código desenvolvido

Primeiramente, as bibliotecas utilizadas e que precisam ser instaladas são: DHT.h; WiFi.h; PubSubClient.h. A instalação pode ser realizada pelo próprio Arduino IDE, assim como a placa utilizada do ESP32 cujo a qual utilizamos a seguinte URL: https://dl.espressif.com/dl/package_esp32_index.json.

Desenvolvemos um diagrama com o fluxograma da função principal que é responsável por se conectar ao wifi, pegar os valores dos sensores, imprimir eles em telas, realizar o publish e validar se a planta precisa ou não ser irrigada, os detalhes das variáveis utilizadas e a estrutura do código estarão no repositório do github mencionado na seção 3.

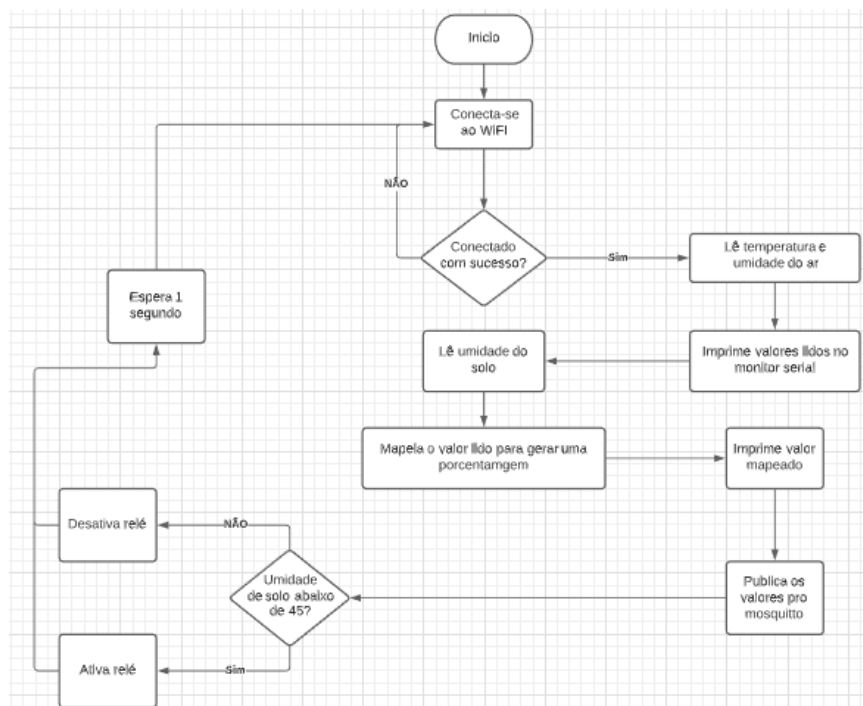


Figura 16. Fluxograma de funcionamento

Este diagrama representa o fluxo de funcionamento do código, que está representado na função `void loop()` do código, mais detalhes podem ser encontrados no código do github.

3. Resultados

Nosso projeto inicialmente tinha sido planejado para ser um sistema de irrigação automático, onde usaríamos um Arduino uno como base. Apesar de termos enfrentados dificuldade e acabamos tendo que realizar a troca para o dispositivo ESP32, e após a troca nós conseguimos continuar com o desenvolvimento do projeto, e o que havia sido planejado inicialmente, foi implementado.

Chegamos a realizar testes em diversas etapas do projeto, começamos pelos sensores para validar se os dispositivos estavam funcionando sem nenhum defeito de fábrica, após isto passamos a testar a conexão com WiFi e foi nesta parte onde travamos e achamos a melhor opção substituir o Arduino uno pelo ESP32. Após a conexão de sucesso com o novo dispositivo, nós passamos a testar a conexão entre o Node-Red e o Broker mosquito, foi a parte do projeto que consumiu mais tempo devido o estabelecimento de conexão entre o ESP32, Node-Red e Mosquitto.

Link do Github: <https://github.com/FelipeCruz00/IoT>

Link do Youtube: https://youtu.be/2hV_ceq1XZc

4. Conclusões

Os resultados indicam que, apesar do nosso projeto ter sido um sucesso devido a forma que foi projetada inicialmente, ainda há muito que melhorar para se tornar uma solução viável para uma implementação em massa¹. A vantagem é que esse projeto é uma implementação relativamente barata e poder ser realizada por pessoas que possuem uma horta caseira e poderá manter suas plantas irrigadas mesmo estando ausente.

Porém a parte negativa do projeto é que o código tem uma necessidade de ser alterado dependendo da necessidade do cliente, devido especificações de solo e clima. Não só isso, como também há a necessidade de modelar uma plataforma que torne o circuito contra água (chuva por exemplo).

Com ambas as visões podemos concluir que, apesar de ser um ótimo projeto caseiro e que você terá uma base profunda a respeito do assunto, há alguns ajustes para poder tornar em uma solução que possa ser implantada em massa para clientes como agricultores. Então a partir de um projeto bem sucedido, nós podemos já começar a pensar em um novo projeto com as melhorias implementadas e tornando ela viável para o mercado de agricultores.

5. Referências

SAFRA. Safra, 2014. Ementa (Site que originou a ideia do projeto e nos motivou para criar uma solução do problema devido ao projeto ter uma menor quantia de equipamentos para irrigar e um maior alcance). Disponível em: <http://revistasafra.com.br/problemas-com-irrigacao-vao-alem-da-falta-de-agua/>. Acesso em: 8, 10 e 2020.

ESP32LEARNING. ESP32 learning, 2020. Ementa (Site utilizado com referência para implementar o sensor de umidade de solo no esp 32). Disponível em: <http://www.esp32learning.com/code/esp32-and-soil-moisture-sensor-example.php>.

Acesso em: 11, 11 e 2020

RANDOM NERD TUTORIALS. Random Nerd Tutorials, 2019. Ementa (Site utilizado com referência para implementar o sensor DHT22 no esp 32). Disponível em: <https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-sensor-arduino-ide/>. Acesso em: 11, 11 e 2020

BLOG DA CURTO. Curto Circuito, 2018. Ementa (Site utilizado como referência para implementar a conexão com o node-red e mosquito, apesar deles utilizarem hivemq como broker). Disponível em: <https://www.curtocircuito.com.br/blog/esp32:-node-red-editor-de-fluxo-on-line>. Acesso em: 17, 11 e 2020

BAÚ DA ELETRÔNICA. Baú da eletrônica, 2017. Ementa (Site utilizado para referência de como implementar um sensor de umidade em um Arduino uno). Disponível em: <http://blog.baudaeletronica.com.br/sensor-de-umidade-de-solo/?amp>. Acesso em: 30, 10 e 2020

ADILSON THOMSEN. FilipeFlop, 2013. Ementa (Site utilizado para referência de como implementar um DHT em um Arduino uno). Disponível em: <https://www.filipeflop.com/blog/monitorando-temperatura-e-umidade-com-o-sensor-dht11/>. Acesso em: 30, 10 e 2020

EARTH OBSERVING SYSTEM. Earth Observing System, 2020. Ementa (Site visitado que nós ajudou a projetar a o projeto). Disponível em: <https://eos.com/pt/blog/umidade-do-solo/>. Acesso em: 30, 10 e 2020