



UNIVERSIDADE FEDERAL DO AMAZONAS
FACULDADE DE TECNOLOGIA
ENGENHARIA ELÉTRICA

SISTEMA DE SENSORIAMENTO EM IOT APLICADO A PEQUENAS PLANTAÇÕES
UTILIZANDO ESP32 E LORA

YASMIM PEREIRA TORRES

MANAUS-AM
2021

Ficha catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Torres, Yasmim Pereira

T693s Sistema de sensoriamento em IoT aplicado a pequenas
plantações utilizando ESP32 e LoRa / Yasmim Pereira Torres.
2021

50 f.: il. color; 31 cm.

Orientador: Waldir Sabino da Silva Júnior

TCC de Graduação (Engenharia Elétrica - Telecomunicações) -
Universidade Federal do Amazonas.

1. Sistema de monitoramento. 2. Umidade do ambiente. 3.
Temperatura do ambiente. 4. Umidade do solo. I. Silva Júnior,
WaldirSabino da. II. Universidade Federal do Amazonas III. Título

Yasmim Pereira Torres

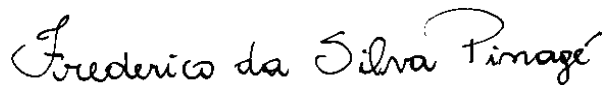
Sistema de sensoriamento em IoT aplicado a pequenas plantações utilizando ESP32 e LoRa

Monografia apresentada ao corpo docente do curso de Engenharia Elétrica da Universidade Federal do Amazonas, como parte dos requisitos necessários para obtenção do título de Engenheira Eletricista.

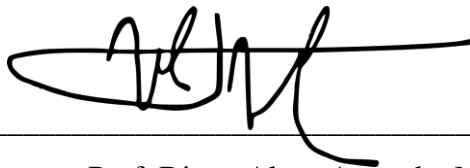
Aprovada em 01 de dezembro de 2021.



Prof. Waldir Sabino da Silva Júnior, D.Sc.



Prof. Frederico da Silva Pinagé, D.Sc.



Prof. Diego Alves Amoedo, M.Sc

MANAUS-AM

2021

Agradecimentos

Agradeço a Deus pelo seu infinito amor e misericórdia pois sem Ele eu nada alcançaria. Aos meus pais, Gilmar Torres e Raimunda Maria, que sempre me apoiaram de todas as formas nos momentos mais difíceis, ao meu irmão, Victor Lorenzo e ao meu noivo, Gabriel de Medeiros, que esteve ao meu lado e sempre me incentivou.

Agradeço aos meus amigos e colegas da UFAM, Luiz Felipe, Jones Castro, Rafael Furtado, Lucas Tribuzy, Jean Braga, Lucas Gonçalves, Clayssa Picanço e Tatiana da Costa que estiveram comigo ao longo dessa jornada e me disponibilizaram um ombro amigo quando precisei.

A todos os professores da UFAM, em especial ao meu orientador Prof. Dr. Waldir Sabino por toda paciência, disposição, ajuda, apoio e conselhos que me fazem crescer em vários aspectos da vida.

*“Entrega o teu caminho ao
Senhor, confia nele, e ele agirá”*

(Salmo 37,5)

Resumo

Este projeto consiste no desenvolvimento de um sistema de monitoramento remoto de fatores importantes no setor de agricultura: umidade do ambiente, temperatura do ambiente e umidade do solo, através da utilização de sensores sem fio, sendo esses fatores disponibilizados em tempo real para acompanhamento em uma aplicação web disponível a qualquer usuário com acesso à internet. Também foi implementado um circuito que visa prevenir, determinada área de cultivo, contra invasão de animais, mantendo o plantio em segurança, principalmente em épocas frutíferas. O sistema desenvolvido foi testado dentro de três cenários de teste, variando os fatores de temperatura, umidade e distância em experimentos que envolvem uma fonte de calor, solo em estado seco, úmido e molhado e aproximação de um obstáculo que simula a presença de um predador no local.

Palavras-chave: Sistema de monitoramento, umidade do ambiente, temperatura do ambiente, umidade do solo.

Abstract

This project consists in the development of a remote monitoring system for important factors in the agriculture sector: ambient humidity, ambient temperature and soil humidity, through the use of wireless sensors, these factors being available in real time for monitoring in a web application available to any user with internet access. A circuit was also implemented that aims to prevent, in a certain area of cultivation, against the invasion of animals, keeping the planting safe. The developed system was tested within three test scenarios, varying the temperature, humidity and distance factors in experiments involving a heat source, soil in a dry, wet and wet state and an obstacle approach that simulates the presence of a predator on site.

Keywords: Monitoring system, ambient humidity, ambient temperature, soil humidity.

Lista de Figuras

2.1 Representação de uma rede LoRa de comunicação.....	8
3.1 Diagrama em blocos do sistema de monitoramento do ambiente e do solo.....	14
3.2 Sensor DHT11.....	15
3.3 Representação do diagrama de pulsos do DHT11.....	15
3.4 Sensor de umidade do solo.....	17
3.5 Pinagem ESP32 TTGO T-beam versão 1.1.....	19
3.6 Esquemático do circuito de monitoramento formado pela placa TTGO T-beam, DHT11 e sensor de umidade do solo.....	20
3.7 Cadastro do usuário na plataforma.....	21
3.8 Aquisição de um plano ATC LoRaWAN.....	21
3.9 Customização do dispositivo.....	22
3.10 Acesso e configuração das informações do dispositivo.....	22
3.11 Trecho do código de monitoramento.....	23
3.12 Criando as variáveis de umidade e temperatura na plataforma.....	23
3.13 Trecho do código de monitoramento mostrando que os mesmos <i>Alias</i> criados na plataforma ProIoT são iguais aos <i>Alias</i> do código principal.....	24
3.14 Trecho do código de monitoramento mostrando o tamanho do payload e as informações em formato json.....	24
3.15 Configurando as opções de payload.....	24
3.16 Configurando o widget de temperatura do ambiente.....	25
3.17 Configurando o widget <i>Alerta de luz</i> para umidade do solo.....	26
3.18 Configurando o widget <i>Gráfico de linhas</i> para visualizar a variação das informações recebidas nas últimas 24 horas.....	26
3.19 Diagrama em blocos do sistema de prevenção contra predadores.....	27
3.20 Sensor ultrassônico HC-SR04.....	27
3.21 Pinagem ESP32-WROOM-32.....	29
3.22 Buzzer ativo.....	30
3.23 Esquemático do circuito de prevenção contra predadores formado pela placa ESP32-WROOM-32, sensor ultrassônico e buzzer.....	30
4.1 Sensor de umidade do solo inserido em solo molhado.....	31
4.2 Umidade do solo molhado sendo mostrada no dashboard da ProIoT.....	32
4.3 Sensor de umidade do solo inserido em solo úmido.....	32
4.4 Umidade do solo úmido sendo mostrada no dashboard da ProIoT.....	32
4.5 Sensor de umidade do solo inserido em solo seco.....	33

Lista de Figuras	9
4.6 Umidade do solo seco sendo mostrada no dashboard da ProIoT.....	33
4.7 DHT11 captando valores de umidade e temperatura em ambiente real e aberto.....	34
4.8 DHT11 captando valores de umidade e temperatura após ser submetido à ação de uma fonte externa de calor.....	34
4.9 Gráfico de linhas com a variação dos componentes no tempo.....	35
4.10 Circuito completo de monitoramento.....	35
4.11 Dashboard completo.....	36
4.12 Dados em tempo real recebidos pela plataforma ProIoT.....	36
4.13 Sensor ultrassônico e ESP32 captando a presença de um predador a uma distância menor que 35cm.....	37
4.14 Serial do ambiente de desenvolvimento integrado Arduino mostrando a distância calculada à medida que o obstáculo se aproxima.....	38

Lista de Tabelas

2.1 Fatores de propagação do protocolo LoRaWAN.....	9
---	---

Lista de Abreviaturas e Siglas

ABP	Activation by Personalization
ADC	Analog to Digital Converter
ADR	Adaptative Data Rate
AppEUI	Application Extended Unique Identifier
AppKey	Application Key
AppSKey	Application Session Key
CSS	Chirp Spread Spectrum
DevAddr	Device Address
DevEUI	Device Extended Unique Identifier
DHT11	Digital Humidity and Temperature
FSK	Frequency Shift Keying
GPIO	General Purpose Input-Output
IoT	Internet Of Things
LoRA	Long Range
LoRaWAN	Long Range Wide Area Network
LPWAN	Low Power Wide Area Networks
MAC	Media Access Control
NetID	Network Identification
NTC	Negative Temperature Coeficiente
NwkSKey	Network Session Key
OTAA	Over the Air Activation
RPMA	Random Phase Multiple Access
RSSI	Received Signal Strength Indication
UNB	Ultra Narrowband

Sumário

Introdução	1
1.1 Objetivo Geral	2
1.2 Objetivos Específicos	2
Fundamentação Teórica	3
2.1 DHT11	3
2.2 Sensor de umidade do solo	3
2.3 Sensor ultrassônico	4
2.4 Internet das coisas (IoT)	4
2.5 <i>Low Power Wide Area Network (LPWAN)</i>	5
2.6 Princípios básicos da tecnologia LoRa	5
2.6.1 A modulação CSS	6
2.7 Protocolo LoRaWAN	7
2.7.1 Topologia de rede	7
2.8 Classes de dispositivos LoRaWAN	9
2.9 Modos de ativação	10
2.10 Gateway	12
2.11 Servidor de rede	12
Sistema Proposto	14
3.1 Visão geral	14
3.2 Bloco DHT11	14
3.3 Bloco sensor de umidade do solo	17
3.4 Bloco ESP32 com LoRa	18
3.5 Blocos gateway, servidor de rede e servidor de aplicação	20
3.6 Visão geral	26
3.7 Bloco sensor ultrassônico	27
3.8 Bloco ESP32-WROOM-32	28
3.9 Bloco buzzer	29
Cenários de teste e resultados	31
4.1 Cenário 1 – Teste de variação da umidade do solo	31
4.2 Cenário 2 – Teste de variação da temperatura e da umidade do ar	33
4.3 Análise dos dados recebidos	34
4.3 Cenário 3 – Variação da distância do predador	37
Conclusão	39
Referências Bibliográficas	40

Capítulo 1

Introdução

A crescente demanda por dispositivos móveis e pelos benefícios que eles promovem, demonstra a importância dos sistemas de comunicação sem fio dentro da sociedade. Nessa tecnologia, a comunicação é realizada por ondas eletromagnéticas, onde qualquer informação pode ser transmitida rapidamente, atendendo um grande número de usuários e sem a necessidade de cabos. Os sistemas de comunicação sem fio incluem uma enorme quantidade de aplicações dentro da área de telecomunicações e da computação, como wi-fi, bluetooth, NFC, comunicação via satélite, infravermelho *etc.* Devido às vantagens dos diversos métodos e padrões da comunicação sem fio, tem crescido a área de Internet das Coisas (IoT) que incorpora sensores, softwares e outras tecnologias para conectar dispositivos e equipamentos à internet. Alguns setores da atualidade se beneficiam fortemente das aplicações em IoT, como os setores de assistência médica, automotivo, segurança e agricultura. No setor da agricultura, um sistema em IoT é construído, por exemplo, para monitorar o campo de cultivo com a ajuda de alguns sensores como os de umidade, temperatura e umidade do solo para que assim sejam tomadas medidas mais assertivas quanto ao cuidado com a plantação, preservação do cultivo e otimização de insumos e tratamentos. Um dos maiores objetivos do monitoramento de fatores como umidade do ar, temperatura do ar e umidade do solo é fazer a irrigação do solo de forma localizada e automatizada, evitando o desperdício de água e diminuindo custos. Dentro dessa realidade, o presente projeto propõe implementar um sistema de sensoriamento em IoT para fazer o monitoramento remoto da umidade, temperatura e umidade do solo de pequenas plantações, disponibilizando essas informações em tempo real em um dashboard disponível pela plataforma da ProIoT, o circuito montado inclui um sensor DHT11 capaz de captar a umidade e a temperatura do ambiente, um sensor de umidade do solo e um placa TTGO T-beam com chip LoRa, que junto ao protocolo LoRaWAN serão alguns dos itens principais abordados neste projeto. Este trabalho propõe também a implementação de um circuito de prevenção da área de cultivo, contra invasão de animais que põem em risco toda a plantação, principalmente em épocas frutíferas, o circuito montado nesta parte inclui um sensor ultrassônico que, com o auxílio do microcontrolador ESP32-WROOM-32, calcula a distância do obstáculo à frente, acompanhando sua aproximação até que ela seja menor ou igual à 35cm para que assim seja emitido um sinal sonoro, através de um buzzer, afugentando o predador.

1.1 Objetivo Geral

Fazer o monitoramento remoto de alguns fatores climáticos e do solo, de uma pequena plantação e desenvolver um sistema de prevenção contra predadores para manter o cultivo em segurança.

1.2 Objetivos Específicos

- 1) Implementar um sistema de monitoramento em IoT para acompanhar fatores como umidade e temperatura do ambiente, em uma pequena área de cultivo.
- 2) Desenvolver um sistema de monitoramento da umidade do solo que possa especificar as condições do solo ao longo do dia, aplicado a uma pequena área de cultivo.
- 3) Construir um circuito de prevenção contra aproximação de possíveis predadores a uma pequena área de cultivo, emitindo um sinal sonoro quando a distância de aproximação for menor que a especificada.

Capítulo 2

Fundamentação Teórica

2.1 DHT11

O sensor digital de umidade e temperatura (DHT11, do inglês, *digital humidity and temperature*) é capaz de, em um único componente, medir dois fatores importantes do ambiente: a temperatura e a umidade relativa do ar. Esse componente possui saída digital e é conhecido por seu baixo custo, baixo consumo de potência e tamanho reduzido [1]. O elemento interno responsável pela medição da temperatura, muda sua resistência elétrica com a variação da temperatura externa e é um termistor com coeficiente de temperatura negativo (NTC, do inglês, *negative temperature coefficient*), o que indica que a resistência e a temperatura são inversamente proporcionais, esse elemento normalmente é feito de cerâmica ou polímero semiconductor para se obter um valor maior de resistência [2]. O elemento responsável pela umidade no DHT11, é um sensor de umidade capacitivo, que possui dois eletrodos com um dielétrico, entre eles, para criar um campo elétrico. A umidade do ar se acumula naquela região causando mudança na condutividade (resistência) do meio. O DHT11 também possui um controlador de 8 bits, responsável por converter o sinal de umidade e temperatura em um sinal serial e enviá-lo ao microcontrolador através do seu pino de dados. Esse sensor, permite medir temperatura na faixa de 0° a 50° Celsius, com precisão de $\pm 2\text{ }^{\circ}\text{C}$, e umidade entre 20% a 90% com precisão de $\pm 5\%$. Ao contrário do DHT22, que é uma versão mais cara e mais robusta, o valor retornado pelo DHT11 é um valor inteiro e ao utilizá-lo, o ideal é que se façam medidas com um intervalo mínimo de 1 segundo pois sua taxa de amostragem é de 1Hz, sua tensão operacional é de 3V a 5V e a corrente máxima pode atingir 2,5mA [3].

2.2 Sensor de umidade do solo

O sensor de umidade do solo faz a medição do nível de umidade do solo e é formado pelos seguintes componentes: dois condutores em forma de garfo, que são introduzidos no solo, e um módulo eletrônico que os conecta ao microcontrolador. O sensor atua através da variação da resistência elétrica entre os condutores, que implica no valor da corrente que percorre esses condutores a cada instante. A resistência elétrica do sensor é inversamente proporcional a quantidade de água no solo, quanto mais úmido, menor é a resistência e maior é a condutividade

entre as pontas de prova. Com isso, o módulo do sensor de umidade do solo produz um sinal de saída que é enviado à saída analógica A_0 , o mesmo sinal é enviado ao comparador LM393, que digitaliza o sinal e o disponibiliza na saída D_0 . Assim, tem-se duas formas de se trabalhar com esse sensor, através da saída analógica, de maior precisão, que pode facilmente ser lida e manipulada por um microcontrolador ou através da saída digital que é influenciada por um potenciômetro responsável por ajustar a sensibilidade da saída. Ao definir, pelo potenciômetro, um limite no nível de umidade, o sensor retorna nível lógico 0, enquanto esse limite não for atingido, e nível lógico 1 ao ultrapassar esse limite. O módulo contém dois leds, um deles indica quando a saída digital está em nível lógico baixo e o outro é o led de energia que acende quando o sensor é alimentado por uma tensão operacional na faixa de 3,3V a 5,5V [4].

2.3 Sensor ultrassônico

O sensor ultrassônico é muito utilizado em projetos eletrônicos e na robótica como um detector de obstáculos, sejam eles objetos, animais ou pessoas, sem entrar em contato com eles e não sendo prejudicado por fatores como formato, acabamento, transparência e cor. Seu princípio de funcionamento é baseado no cálculo da distância entre emissor (sensor) e receptor (obstáculo) quando o sensor envia uma onda sonora, na faixa de ultrassom, e aguarda o recebimento do sinal refletido, o eco, levando em consideração o tempo em que a onda leva para atingir o obstáculo e retornar desde que o obstáculo seja capaz de produzir a reflexão necessária. O sensor é capaz de manter esse registro de tempo e, sabendo o valor da velocidade do som ($340,29 \text{ m/s}$), a distância é facilmente encontrada por simples equação matemática [5]:

$$d = v \cdot t \quad (1)$$

2.4 Internet das coisas (IoT)

Com o crescimento em larga escala do número de dispositivos que necessitam conexão com a internet, rapidez no processamento e eficiência de custo surgiu a Internet das Coisas (IoT, do inglês, *internet of things*) em 1999 após Kevin Ashton mencionar o termo pela primeira vez no *RFID Journal*. A Internet das Coisas mostrou-se como uma forma de facilitar a conexão e a comunicação de equipamentos eletrônicos, dos mais diversos setores, com usuários e servidores disponíveis. Algumas empresas, através de plataformas, facilitam esse processo de comunicação, disponibilizando sua estrutura de rede IoT de forma parcial ou completa para que os usuários possam configurar e utilizar essa tecnologia conectando seus projetos IoT à rede. Com isso, aumenta o número de empresas engajadas em investir em pesquisas e desenvolver

novas plataformas de hardware para dispositivos que serão aplicados em soluções inteligentes para o ramo da agricultura, saúde, segurança etc [6].

2.5 Low Power Wide Area Network (LPWAN)

As redes de longa distância de baixa potência (LPWAN, no inglês, *low power wide area networks*) foram boas opções para aplicações com baixa taxa de transmissão, quando surgiu a necessidade de mitigar a alta procura por tecnologias de baixo custo, baixo consumo, longo alcance, segurança na rede e robustez. Atualmente, há redes LPWANs trabalhando em bandas de frequência licenciadas e não licenciadas, a maior parte atua nas bandas não licenciadas chamadas de bandas industrial, científica e médica (ISM, do inglês, *industrial, scientific and medical*) de 2,4GHz, 868/915 MHz, 433 MHz e 169 MHz, conforme a região em que opera. Duas delas, as redes Sigfox e Ingenu, serão abordadas nesta seção e a rede LoRa, escolhida para ser utilizada neste trabalho, será abordada de forma mais abrangente em outra seção. Juntas elas são as são as tecnologias mais importantes das LPWANs [7].

- a) Rede Sigfox: A rede Sigfox surgiu em 2009 e agora está presente em mais de 70 países, foi a primeira tecnologia LPWAN proposta no mercado de IoT. Essa rede opera em sua camada física com a modulação sem fio de banda ultra estreita (UNB, do inglês, *ultra narrowband*) porém, os protocolos da camada de rede não são divulgados, escolha feita pela Sigfox que atua como uma operadora de serviços IoT e possui o registro da patente sobre essa tecnologia. A comunicação bidirecional é suportada e, de acordo com a empresa, cada gateway (equipamento central da rede) pode suportar até 1 milhão de conexões [7].
- b) Rede Ingenu: A rede Ingenu é uma marca comercial da empresa ON-Ramp Wireless que desenvolveu a tecnologia de acesso múltiplo de fase aleatória (RPMA, do inglês, *Random Phase Multiple Access*) implantada na rede Ingenu e em diversas outras, atuando na banda de 2,4GHz e permitindo um longo alcance devido a um design de camada física robusta. A desvantagem é que essa rede não prevê comunicação bidirecional, não sendo viável para algumas aplicações em IoT [7].

2.6 Princípios básicos da tecnologia LoRa

Uma das soluções LPWAN mais importantes é a tecnologia de longo alcance (LoRa, do inglês, *long range*) que se trata de um nome comercial de uma tecnologia desenvolvida em 2010 pela startup francesa Cycleo e adquirida em 2012 pela empresa Semtech Corporation, responsável por fornecer os chips de rádio LoRa, além de ser a fundadora da LoRa Alliance,

uma aliança de diversas empresas com interesse em aplicações IoT que se encarrega da padronização do protocolo LoRaWAN.

2.6.1 A modulação CSS

A modulação por trás do LoRa é patenteada e a documentação a respeito de processamento de sinal e equações não é divulgada. Essa técnica de modulação é baseada no espalhamento do espectro por chirp (CSS, do inglês, *Chirp Spread Spectrum*), onde chirps são sinais da portadora cuja frequência aumenta ou diminui com o tempo, possibilitando uma transmissão de dados a longas distâncias com baixo consumo de energia até que seja demodulado, mesmo que sua potência seja de até 20dB mais baixa que o nível de ruído. A modulação CSS determina que cada símbolo seja espalhado por vários chirps e o número de chirps por símbolo é chamado de fator de espalhamento (do inglês, *spreading factor*). O fator de espalhamento é o responsável por alterar a sensibilidade do rádio LoRa, ele influencia na ortogonalidade do sinal e, conseqüentemente, na propagação do mesmo, quanto mais sensível maior é a capacidade de demodular sinais onde o nível ruído é muito mais presente que o sinal real [8]. Pacotes que são transmitidos simultaneamente, com diferentes fatores de espalhamento, são ortogonais entre si e não se colidem. O fator de espalhamento pode variar de 7 a 12, quanto maior seu valor, maior é o tempo de propagação no ar e menor é a taxa de dados. Para o fator de espalhamento 7, a transmissão de dados é maior e o dispositivo consome menos energia pois ocupa por menos tempo o espectro de frequência. O ajuste do fator de espalhamento precisa ser feito levando em consideração a distância a ser percorrida pelo sinal, fatores de espalhamento como 10, 11 e 12 são usados, por exemplo, para transmissões em longas distâncias. O desempenho da comunicação LoRa depende de diversos outros elementos, como largura de banda, taxa de codificação, potência de transmissão e frequência da portadora.

- a) Largura de banda: Larguras de banda maiores possibilitam a transmissão de dados em taxas mais altas, mas reduzem a sensibilidade do receptor. Para uma largura de banda BW , normalmente sendo 125kHz ou 500kHz, a taxa de chirp R_c é dada por:

$$R_c = BW \text{ chips/s} \quad (3)$$

- b) Taxa de codificação: Para aumentar a robustez do sinal transmitido e a resistência a bits corrompidos, LoRa suporta um esquema de correção de erro com um número variável de bits redundantes, para cada quatro bits enviados um quinto bit de paridade é enviado. A taxa de codificação (do inglês, *coding rate*) é representada pelas seguintes relações: 4/5, 4/6, 4/7 ou 4/8, ela deve ser maior à medida que a

quantidade de interferência aumenta, para evitar ao máximo a recepção de pacotes malsucedidos, no entanto, isso gera um custo maior de energia pois quanto mais bits são transmitidos na mensagem maior é o tempo de propagação da mensagem no ar.

- c) Potência de transmissão: A potência de transmissão é uma característica que pode ser alterada pelo transceptor LoRa, ela está diretamente ligada à energia necessária para transmitir um pacote. Para potências acima de 17dBm, as limitações de hardware e as regulamentações legais, limitam o ciclo de trabalho do rádio LoRa a um máximo de 1%.
- d) Frequência da portadora: Os transceptores LoRa fazem uso de frequências na faixa de mega hertz, dentro de bandas não licenciadas, como 433MHz, 868MHz e 915MHz. No Brasil, na Austrália e nos Estados Unidos, é utilizada a frequência de 915MHz [9].

2.7 Protocolo LoRaWAN

Ao contrário da modulação LoRa, que é patenteada, o protocolo de rede LoRaWAN (do inglês, *long range wide area network*) é de código aberto e suas especificações são mantidas pela LoRa Alliance. A modulação LoRa é responsável pelo link de comunicação de longo alcance e faz parte da camada física da estrutura LoRaWAN, que também pode ser implementada com modulação por chaveamento de frequência, (FSK, do inglês, *frequency shift keying*), enquanto a camada de controle de acesso ao meio (MAC, do inglês, *media access control*) é definida pelo protocolo LoRaWAN. O protocolo LoRaWAN é responsável por diversas funções dentro da arquitetura do sistema, tais como: segurança e gerenciamento da rede, conexão com o servidor de rede, ativação dos dispositivos e confirmação de mensagens [10].

2.1.7 Topologia de rede

Grande parte das redes de comunicação sem fio fazem uso da arquitetura de rede em malha, chamada *mesh*, entretanto essa arquitetura possui desvantagens significativas ao aumentar a distância de comunicação. O LoRaWAN foi estruturado com base na arquitetura estrela, e divide-se em quatro partes: dispositivos finais, gateways, servidores de rede e servidores de aplicação. Os dispositivos finais são responsáveis pelo envio dos dados coletados aos gateways, dados como de temperatura, umidade, distância ou pressão, por exemplo. Os gateways são equipamentos embarcados com rádios potentes, responsáveis por retransmitir as

mensagens entre os dispositivos finais e o servidor de rede através de conexões IP [10]. Os dispositivos finais fazem uso de comunicação LoRa para trocar mensagens para um ou mais gateways que possuem a habilidade de receber pacotes de diversos dispositivos simultaneamente, ou seja, o gateway é capaz de receber mensagens com diferentes taxas de dados no mesmo canal e ao mesmo tempo, possibilitando que os pacotes de informação fiquem no ar o menor tempo possível e consequentemente deixando mais canais livres para serem usados por outros dispositivos, visando a economia de bateria e o menor consumo de energia possível. Ao aumentar a quantidade de gateways, a taxa de dados varia, a latência diminui e o dimensionamento da rede pode aumentar até oito vezes mais. Os servidores de rede são responsáveis pelo gerenciamento da rede, filtragem dos pacotes, segurança e mensagem de confirmação, eles se comunicam com os servidores de aplicação que fazem o manuseio e análise dos dados de aplicação, nessa etapa é onde são construídas as aplicações que vão interpretar e apresentar os dados ao usuário final, o servidor de aplicação formata os dados que serão enviados e fornece o *payload* de downlink da camada de aplicação para os dispositivos finais, tudo isso de forma autônoma [11]. A Figura 2.1 ilustra a arquitetura estrela da rede LoRa e seus respectivos componentes.

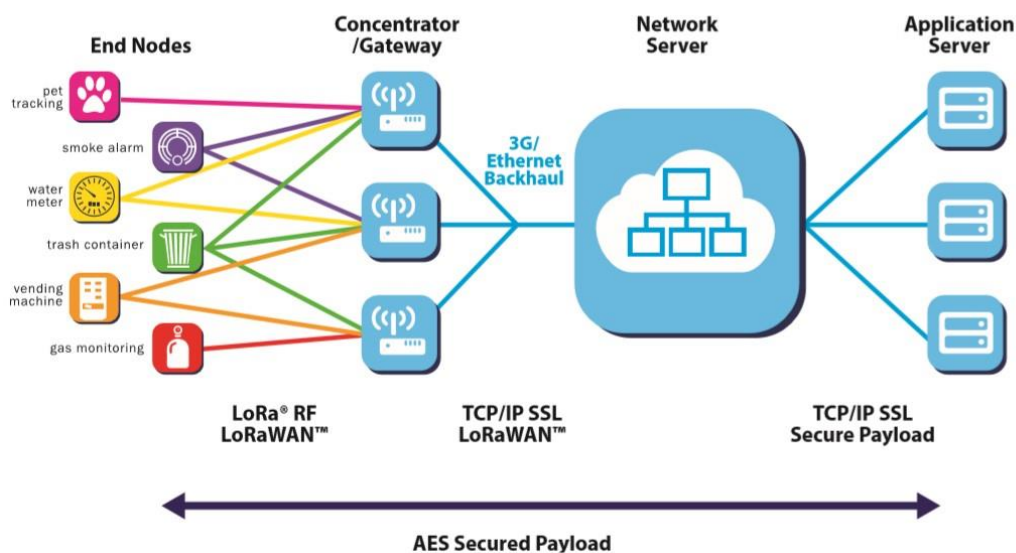


Figura 2.1: Representação de uma rede LoRa de comunicação.

Fonte: [11].

A arquitetura estrela possui bom desempenho ao relacionar comunicação de longo alcance, número de antenas e vida útil da bateria dos dispositivos. São utilizados diferentes canais de frequência e taxa de dados para comunicar dispositivos finais e gateways, o LoRa pode permitir taxas de dados de 300bps a 5kbps ao utilizar uma largura de banda de 125kHz, por exemplo. A rede LoRa consegue alcançar alta capacidade utilizando a taxa de dados

adaptável (ADR, *adaptive data rate*) e através de um transceptor multicanal no gateway permitindo as mensagens simultaneamente recebidas [11]. A taxa de dados adaptável gerencia as taxas de dados dos nós e a saída de rádio frequência de cada dispositivo conectado, otimizando o processo e diminuindo o tempo de uso das antenas e o consumo médio de energia. Algumas regras precisam ser aplicadas para que os dispositivos finais possam transmitir em qualquer canal, a qualquer hora e fazendo uso de qualquer taxa de dados disponível, são elas:

- a) A grande quantidade de trocas de canal de forma aleatória pelos dispositivos finais alcança um sistema mais robusto contra interferências.
- b) Os dispositivos finais devem obedecer às regulamentações locais relacionadas ao tempo máximo de transmissão na sub-banda que costuma ser 400ms [12].

A Tabela 2.1 mostra a taxa de dados em função da distância e do fator de espalhamento, um o gateway faz uso dessa propriedade de dependência para receber diferentes taxas de dados no mesmo canal e ao mesmo tempo.

Tabela 2.1: Fatores de propagação do protocolo LoRaWAN.

Fator de espalhamento (125kHz)	Taxa de bits	Alcance (depende das condições de propagação)	Tempo no ar (ms) (para 10 bytes)
SF7	5470 bps	2 km	56 ms
SF8	3125 bps	4 km	100 ms
SF9	1760 bps	6 km	200 ms
SF10	980 bps	8 km	370 ms
SF11	440 bps	11 km	740 ms
SF12	290 bps	14 km	1400 ms

Fonte: [12].

Quanto mais próximo ao gateway o dispositivo final estiver, menor deve ser o fator de espalhamento, para que um dispositivo final mais distante possa transmitir com um fator de espalhamento mais alto pois necessita de um maior ganho de processamento e maior sensibilidade de recepção, embora a taxa de dados seja menor.

2.8 Classes de dispositivos LoRaWAN

Existem três classes diferentes de perfis de comunicação disponíveis para redes LoRa, cada classe é responsável por atender a necessidades de diferentes aplicativos com diferentes especificações. As principais diferenças entre as classes A, B e C estão relacionadas

principalmente a escolha da latência de comunicação e consumo de energia. As características que levam à divisão em classes, são:

- a) Classe A: Os dispositivos permitem comunicação bidirecional e é a classe suportada por todos os dispositivos. Nesta classe, tem-se duas janelas de tempo de recepção de downlink a cada uplink enviado, o servidor pode responder com um downlink fazendo uso de uma das janelas. É a classe com maior eficiência energética devido ao agendamento do tempo das janelas de acordo com a necessidade do dispositivo e é a opção mais indicada para aplicações que necessitam apenas de uma curta resposta de downlink do servidor logo após o uplink ter sido enviado [10,11].
- b) Classe B: É uma derivação da classe A, não existe obrigatoriedade de ser suportada por todos os dispositivos e permite comunicação bidirecional. A diferença para a classe A é que além das janelas para uplinks, a classe B possui janelas programadas em intervalos fixos de tempo para downlink que são selecionadas pelo servidor [11].
- c) Classe C: É uma derivação da classe A e B, não existe obrigatoriedade de ser suportada por todos os dispositivos e permite comunicação bidirecional, mas diferente das classes A e B, o dispositivo fica em modo de recepção até que uma mensagem de uplink seja enviada novamente. A desvantagem é que existe uma diminuição na latência e um aumento no consumo de energia quando comparado aos dispositivos de outras classes. Dispositivos que implementam comunicação de classe C são usados em aplicações com energia suficiente disponível e com isso não precisam de janelas programadas de recepção [12].

2.9 Modos de ativação

O documento com especificações do protocolo de comunicação LoRaWAN descreve os parâmetros recomendados de um sinal de transmissão LoRa para as regiões de regulamentação espalhadas pelo mundo e pode sofrer alterações a cada versão lançada pela LoRa Alliance. O protocolo LoRaWAN dispõe de seis versões lançadas desde 2015, sendo a última a versão 1.0.4, lançada em 2020. As versões do protocolo LoRaWAN envolvem métodos de ativação conhecidos como ativação por personalização (ABP, do inglês, *activation by personalization*) e ativação pelo ar (OTAA, do inglês, *over the air activation*). Para um dispositivo que faz uso do ABP não é necessário possuir uma permissão de acesso à rede pois ele já vem de fábrica conectado a uma rede específica com identificação e chaves personalizadas, logo o dispositivo está pronto para uso assim que é ligado, mas é considerado menos seguro em comparação ao

OTAA. No modo de ativação OTAA o dispositivo solicita permissão de acesso e pode se conectar a múltiplas redes, nesse modo de ativação o fabricante do dispositivo disponibiliza as informações necessárias para conexão e as chaves de segurança podem ser renovadas sempre que for necessário [12]. Existem duas chaves raiz (do inglês, *root keys*) referentes à segurança do protocolo, criadas com o objetivo de manter a integridade e confidencialidade das mensagens, são elas: chave de aplicação (AppKey, do inglês, *application key*) e chave de rede (NwkKey, do inglês, *network key*). Essas chaves, além de serem chaves específicas para cada dispositivo, devem ser armazenadas pela rede onde o dispositivo está conectado. A Appkey deriva outras duas chaves, a chave de sessão de aplicação (AppSKey, do inglês, *application session key*) e a chave de sessão de rede (NwkSKey, do inglês, *network session key*), a AppSKey é usada para criptografar e descriptografar o *payload* que carrega os dados de aplicação. A NwkSKey é utilizada tanto pelo dispositivo final quanto pelo servidor de rede para gerar e verificar o código de integridade da mensagem, assegurando a integridade e criando uma assinatura específica para cada dispositivo [13].

Conforme foi citado, o LoRaWAN define dois modos de ativação, o OTAA e o ABP, pelo menos um deles deve ser usado para que o dispositivo final ingresse na rede. No modo ABP, o dispositivo já possui todas as chaves necessárias para dar início a uma comunicação com a rede, o endereço do dispositivo (DevAddr, do inglês, *device address*), NwkSKey e AppSKey são armazenados no dispositivo final desde sua fabricação, a desvantagem é a possibilidade dessas chaves serem descobertas por terceiros, e com isso eles façam cópias ou descriptografem as mensagens trocadas. O modo de ativação OTAA requer três outros fatores importantes: um código de identificação do dispositivo (DevEUI, do inglês, *device extended unique identifier*), um código de aplicação (AppEUI, do inglês, *application extended unique identifier*) e a chave de aplicação AppKey, esse modo é considerado a forma mais segura de autenticação pois cada vez que o dispositivo é conectado à rede, uma chave de sessão de rede é gerada, isso facilita o *roaming* entre as redes de provedores diferentes, além disso, com duas chaves é mais difícil que os dados sejam lidos mesmo que uma das chaves esteja comprometida. Ao iniciar a comunicação com a rede, o dispositivo final envia uma mensagem chamada *join-request*, solicitando conexão, a mensagem inclui AppEUI, DevEUI e um valor aleatório usado para rejeitar *join-request* inválidos, chamado DevNonce. O servidor recebe a solicitação de conexão e valida ou não o pedido, enviando a mensagem *join-accept* caso a solicitação tenha sido aceita. A mensagem *join-accept* carrega a identificação da rede (NetID, do inglês, *network identification*), o endereço do dispositivo DevAddr e um AppNonce, gerado pelo DevNonce.

Por fim, o dispositivo gera as chaves de sessão que serão armazenadas na rede, permitindo a troca de mensagens entre o servidor e o dispositivo [12,13].

2.10 Gateway

Os gateways de uma rede LoRaWAN operam na camada física e verificam a integridade dos dados de cada mensagem LoRa de entrada, eles podem transmitir sinais alcançando uma distância de até 5km em ambientes urbanos e até 15km em ambientes rurais. A capacidade de uma rede LoRaWAN, tanto em quantidade de mensagens quanto em número de conexões, depende de quantos gateways estão operantes na rede. Um gateway de oito canais pode suportar, em 24 horas, milhares de mensagens, com dez gateways a rede pode suportar cem mil dispositivos conectados e um milhão de mensagens se cada dispositivo enviar dez mensagens por dia. O Brasil, segue o regulamento *AU915-928MHz ISM Band* do documento de parâmetros regionais lançado pela LoRa Alliance, esse documento especifica que cada gateway pode ter até 64 canais de uplink para uma largura de banda de 125kHz, iniciando a partir do canal de frequência de 915,2MHz, incrementando linearmente a 200kHz até 927,8MHz. Para uma largura de banda de 500kHz são reservados 8 canais de uplink iniciando a partir do canal de frequência de 915,9MHz, incrementando linearmente a 1,6MHz até 927,1MHz e para downlink nessa mesma largura de banda, são definidos 8 canais iniciando a partir do canal de frequência de 923,3MHz, incrementando linearmente a 600kHz até 927,5MHz. Ao invés de cabeamento, a comunicação entre dispositivo final e gateway é feita pelo ar através de rádio frequência, (RF, do inglês, *radio frequency*), se a integridade da mensagem recebida estiver comprometida, ela é descartada, se correta o gateway a encaminha para o servidor de rede junto com um indicador de intensidade do sinal (RSSI, do inglês, *received signal strength indication*) e com as informações de data e hora (opcional). Cada pacote de uplink modulado passa por todos os gateways que estão ao alcance. Os gateways encaminham, por tráfego IP, as mensagens ao servidor de rede que é quem controla e filtra os pacotes redundantes gerados a partir disso. A comunicação entre gateway e servidor de rede pode via wi-fi, ethernet com fio, satélite ou por meio de uma conexão celular [11,12].

2.11 Servidor de rede

O servidor de rede LoRaWAN faz o gerenciamento e controle dos parâmetros da rede, para que haja o transporte de dados de forma segura desde o dispositivo final até as aplicações de usuário na nuvem, assim como o controle do tráfego entre dispositivo final e servidor de rede mesmo sem poder ver ou acessar os dados. Ele também agenda confirmações através do

gateway e põe em prática a taxa de dados adaptável [12,13]. Os servidores de rede LoRaWAN são conhecidos pelas seguintes funções:

- a) Verifica o endereço do dispositivo.
- b) Notifica quando as mensagens são recebidas.
- c) Faz uso da taxa de dados adaptável.
- d) Responde todas as solicitações da camada MAC.
- e) Encaminha o *payload* de informação para o servidor de aplicação.
- f) Organiza o *payload* de downlink do servidor de aplicação para encaminhar para qualquer dispositivo da rede.
- g) Encaminha mensagem de *join-accept*.

Capítulo 3

Sistema Proposto

O Capítulo 3 detalha os procedimentos técnicos executados no desenvolvimento do sistema de monitoramento do ambiente e do solo, usando a placa TTGO T-beam, e no sistema de prevenção contra predadores utilizando a placa ESP32-WROOM-32, descrevendo a função, implementação e os materiais utilizados em cada bloco.

3.1 Visão geral

O sistema proposto do projeto de monitoramento do ambiente e do solo é formado por seis blocos, duas entradas e uma saída. Para os três primeiros blocos, foram adquiridos os seguintes componentes: um sensor de umidade e temperatura do ambiente (DHT11), um sensor de umidade do solo e uma placa TTGO T-beam com chip LoRa. O circuito visa captar esses fatores do ambiente e do solo (entradas) para que possam ser enviados para um dashboard (saída), que será atualizado a cada 5 minutos, sendo possível de ser acessado de qualquer lugar e por qualquer usuário que tenha acesso à internet e à plataforma ProIoT. Os blocos gateway, servidor de rede e servidor de aplicação fazem parte da infraestrutura LoRaWAN que não foi implementada neste projeto, mas foi contratada através de um plano anual de R\$34,00, oferecido pela plataforma ProIoT, escolhida por ter um custo acessível e por ter mais recursos disponíveis que as plataformas gratuitas como a *The Things Network*, por exemplo. A Figura 3.1 ilustra o diagrama em blocos que demonstra cada etapa e conexão do sistema.

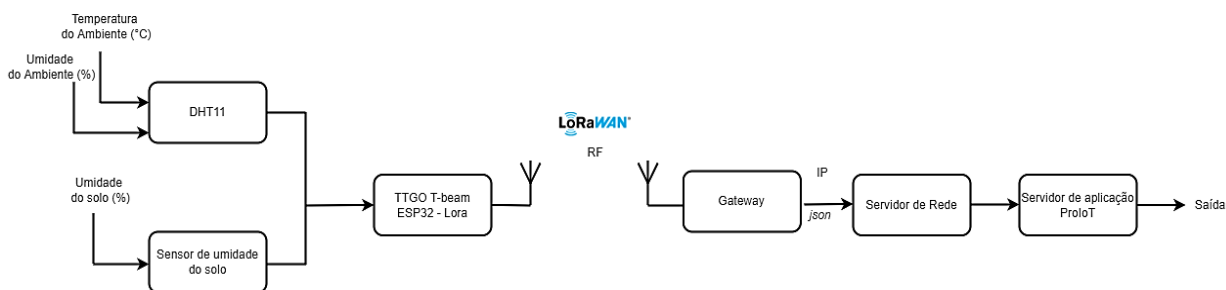


Figura 3.1: Diagrama em blocos do sistema de monitoramento do ambiente e do solo.

Fonte: Próprio autor.

3.2 Bloco DHT11

O bloco DHT11 refere-se unicamente ao sensor DHT11, esse sensor é um dos principais componentes utilizados neste projeto, pois é responsável por captar a temperatura e a umidade do ambiente em que é submetido. O DHT11 está disponível no mercado como um sensor de 4

pinos ou como um módulo de 3 pinos, a diferença entre eles é um resistor de pull-up e um led de ativação que já vem embarcado no módulo. O sensor utilizado é ilustrado na Figura 3.2:

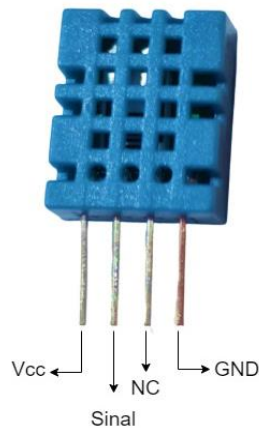


Figura 3.2: Sensor DHT11.

Fonte: Próprio autor.

Neste projeto, foi utilizado o sensor de 4 pinos, sendo eles: V_{cc} , GND , $Sinal$, e um pino não conectado e descartado para uso chamado NC . A comunicação com o DHT11 consiste em 3 etapas: a primeira é enviar uma solicitação ao DHT11 através de um pulso, em seguida o DHT11 envia um pulso de resposta e então ele começa a enviar dados com um total de 40 bits para o microcontrolador. O estímulo inicial dado ao DHT11 é um pulso em nível lógico baixo por pelo menos $18ms$ enviado pelo microcontrolador e em seguida um pulso em nível lógico alto, ambos aplicados ao pino de dados do sensor. Após esse estímulo, o DHT11 enviará um pulso de resposta ao microcontrolador caracterizado por ser um pulso de nível lógico baixo por $54\mu s$ e, em seguida, nível lógico alto $80\mu s$. A Figura 3.3 mostra a representação do diagrama de pulsos do DHT11:

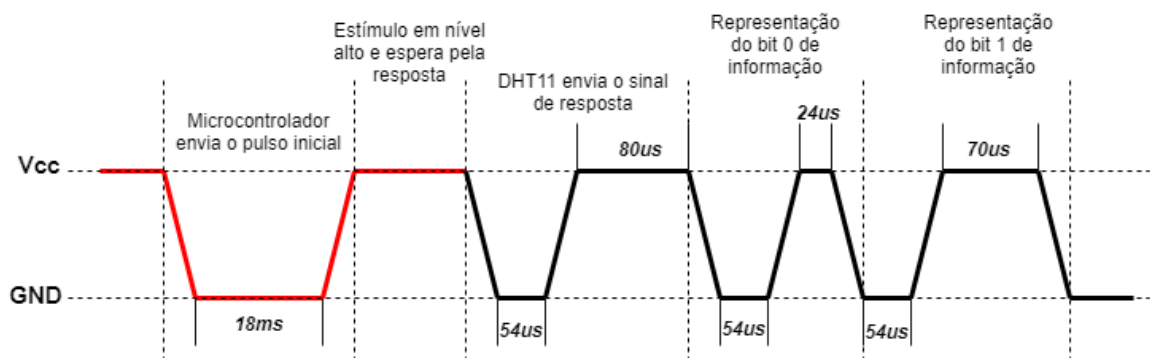


Figura 3.3: Representação do diagrama de pulsos do DHT11.

Fonte: Próprio autor.

Com isso, o DHT11 envia os dados contendo o valor de umidade e temperatura junto com a soma de verificação, totalizando 40 bits de comprimento que são divididos em 5 bytes

por segmento, cada byte com 8 bits, os primeiros dois segmentos contêm a parte inteira e a parte decimal da umidade captada, bem como os próximos dois segmentos contêm a parte inteira e a parte fracionária da temperatura. O último segmento é a soma de verificação cujo conteúdo é o somatório dos 4 primeiros segmentos, para que o microcontrolador possa fazer a comparação com os primeiros bytes recebidos e verifique se os dados estão corretos [14]. Então, o DHT11 entra em modo de economia de bateria até que o microcontrolador envie novamente o pulso de início.

As especificações técnicas desse sensor são descritas a seguir [15]:

Especificações:

- ✓ Dimensão: 23mm x 12mm x 5mm (incluindo terminais).
- ✓ Tensão de alimentação: 3V ~ 5,5V.
- ✓ Corrente: 200 μ A a 500mA, em stand by de 100 μ A a 150 μ A
- ✓ Faixa de medição de temperatura: 0°C ~ 50°C.
- ✓ Precisão de medição de temperatura: ± 2 C°.
- ✓ Faixa de medição de umidade: 20% ~ 90%.
- ✓ Precisão de medição de umidade: $\pm 5\%$.

Durante a implementação do bloco DHT11, optou-se por alimentá-lo com uma tensão de 5V, aplicada ao pino V_{cc} , que é um valor de tensão que está de acordo com sua especificação técnica. Foi adicionado um sensor de pull-up de 10k Ω que é ligado do pino *Sinal* até o pino V_{cc} com a finalidade de garantir que as entradas se ajustem em níveis lógicos esperados (nível alto e nível baixo), prevenindo estados desconhecidos. O pino *Sinal* foi ligado à GPIO 4 (do inglês, *general purpose input-output*) do microcontrolador ESP32 com LoRa que será explorado com mais detalhes na Seção 3.4 e o pino *GND* do sensor foi ligado ao pino *GND* do ESP32, o circuito foi montado em um protoboard de 400 furos. Para o funcionamento do sensor foi desenvolvido um código em linguagem C no ambiente de desenvolvimento integrado Arduino (Arduino IDE, do inglês, *arduino integrated development environment*) versão 1.18.15 instalado em um computador com sistema operacional Windows. Nesse software de código aberto é possível escrever ou fazer upload de códigos em linguagem C ou C++ para embarcar em placas compatíveis com Arduino, tendo acesso à correção de erros, destaque de sintaxe, inclusão de bibliotecas, visualização de resultados através do monitor serial e diversas outras funcionalidades. Na implementação do código foi utilizada a biblioteca *DHTesp.h*,

especializada na leitura de sensores de umidade e temperatura da linha DHT, e duas funções importantes para obter a temperatura em graus celsius e a umidade relativa em porcentagem, são elas: *dht.getTemperature* e *dht.getHumidity* responsáveis por dar início à leitura desses parâmetros do sensor se a última leitura for anterior ao tempo mínimo de atualização do sensor [16].

3.3 Bloco sensor de umidade do solo

O bloco sensor de umidade do solo diz respeito à implementação e funcionalidade desse sensor dentro do projeto de monitoramento. Esse sensor é formado por dois condutores em forma de garfo e um módulo eletrônico que os conecta ao microcontrolador. O módulo possui 4 pinos: A_0 , D_0 , GND e V_{cc} , os pinos A_0 e D_0 indicam duas formas de se trabalhar com esse sensor, fazendo uso de sua saída analógica ou digital, como mencionado na Seção 2.2. Durante o desenvolvimento deste bloco, optou-se por utilizar apenas a saída analógica para que se tenha maior liberdade para fazer manipulações dos dados retornados. O sensor foi alimentado com uma tensão de 5V aplicada ao pino V_{cc} que foi ligado à saída de 5V do microcontrolador no qual está conectado, o pino GND do sensor também foi ligado ao pino GND do ESP32 com LoRa. A saída A_0 foi conectada à porta 0 do ESP32 pois a mesma apresenta um conversor analógico-digital (ADC, do inglês, *analog to digital converter*) necessário para converter os dados provenientes de A_0 para que sejam interpretados pelo microcontrolador. O circuito foi montado no mesmo protoboard do DHT11. O sensor de umidade do solo utilizado é ilustrado na Figura 3.4:



Figura 3.4: Sensor de umidade do solo.

Fonte: Próprio autor.

As especificações técnicas desse sensor são descritas a seguir [17]:

Especificações:

- ✓ Dimensão: 60mm x 20mm x 3mm.
- ✓ Modelo: Módulo e sonda.
- ✓ Tensão de alimentação: 3,3V ~ 5,5V.
- ✓ Corrente máxima: 35mA.
- ✓ Comprimento do cabo: 21cm.
- ✓ Peso: 15g.
- ✓ Led indicador de tensão: vermelho.
- ✓ Led indicador de saída digital: verde.

Para o funcionamento do sensor foi desenvolvido um código em linguagem C no Arduino IDE e nesse do código foi utilizada a função *analogRead (pinSensorA)*, onde *pinSensorA* é referente ao pino A_0 que está conectado à porta 0 do microcontrolador. Essa função retorna valores dentro da faixa de 4095, quando o sensor não está inserido no solo, e 3224, quando o sensor está inserido em um solo totalmente submerso na água. Então, faz-se uso da função *map* para ajustar os valores captados dentro de uma escala entre 0 e 100, com isso, obtém-se o valor final que representará a umidade do solo naquele instante de medição.

3.4 Bloco ESP32 com LoRa

Esse bloco refere-se à comunicação dos sensores com a placa TTGO T-beam utilizada neste projeto, que é uma placa de desenvolvimento da fabricante Espressif, versão 1.1, com chip ESP32 de 4MB de memória flash. Essa placa integra um transceptor SemTech SX1276 necessário para a comunicação LoRaWAN na banda 868/915MHz, um receptor de GPS U-Blox Neo-6M, um chip de energia AXP192, antena 3D e GPIOs que podem ser aplicadas em diversas finalidades e aplicações, dependendo de qual módulo ou função se usa primeiro. Por exemplo, a GPIO 0 é usada para definir um canal ADC mas também pode ser utilizada para definir um canal PWM. O chip SX1276 faz uso da técnica de modulação LoRa podendo alcançar uma sensibilidade de mais de $-148dBm$ que combinada com um amplificador de potência de $+20dBm$ torna-o ideal em aplicações de longo alcance [18, 19]. A placa TTGO T-beam versão 1.1 utilizada neste projeto, junto ao esquemático dos pinos, pode ser vista na Figura 3.5:

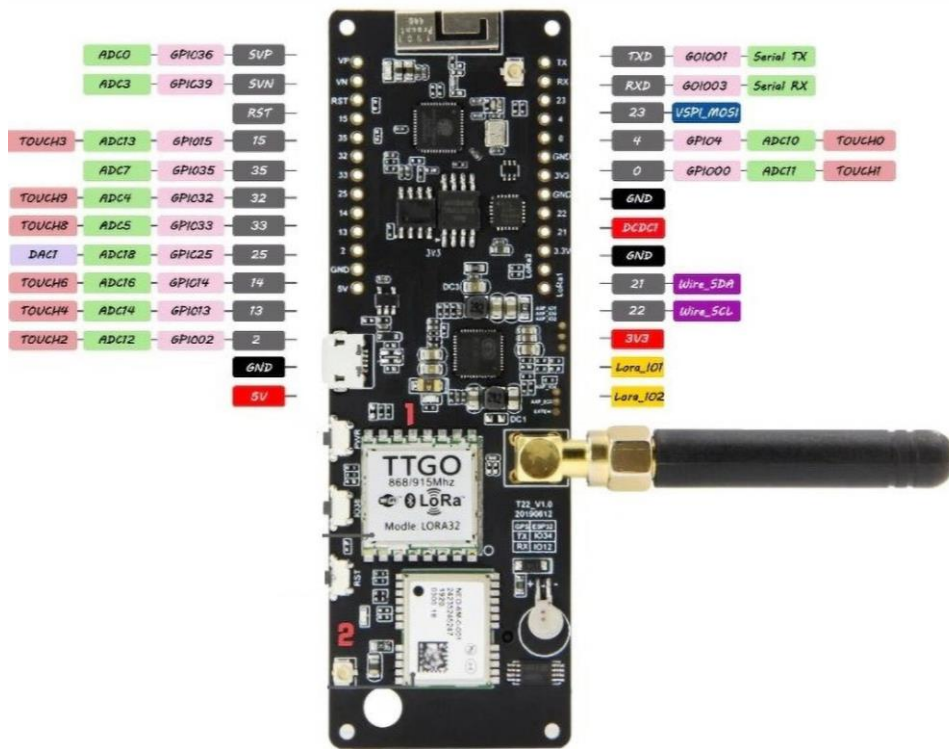


Figura 3.5: Pinagem ESP32 TTGO T-beam versão 1.1.

Fonte: [18].

A placa possui um conector micro-usb para alimentá-la ao ser conectada ao computador (5V/1A) mas pode ser alimentada pela bateria RONTEK de Li-ion, 3,7V e 2200 mAh. As portas do microcontrolador utilizadas neste projeto, além da saída de 5V e do pino GND, foram as GPIO 4 e GPIO 0. Os pinos 5, 12, 15, 16, 17, 18, 19, 26 e 27 são usados para funções de controle da placa e não devem ser usados para outros fins. O código associado a esse microcontrolador para acionamento da comunicação LoRa, incluindo ganho, chaves de acesso, payload, intervalo de tempo entre as transmissões e integração com os códigos dos sensores, foi desenvolvido no Arduino IDE, em linguagem C, as principais bibliotecas utilizadas foram: *Axp20x.h*, *Ticker.h*, *Limic.h* e *Arduinojson.h*.

A Figura 3.6 ilustra o esquemático do circuito de monitoramento:

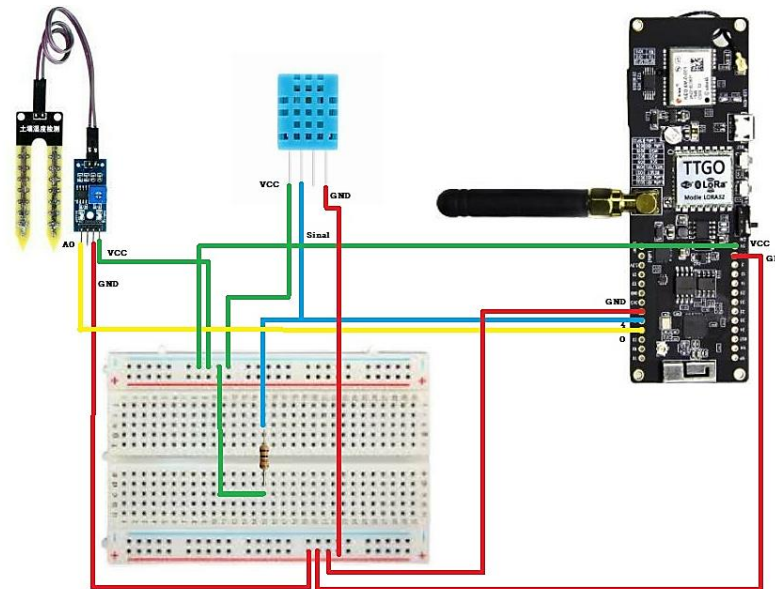


Figura 3.6: Esquemático do circuito de monitoramento formado pela placa TTGO T-beam, DHT11 e sensor de umidade do solo.

Fonte: Próprio autor.

3.5 Blocos gateway, servidor de rede e servidor de aplicação

Neste projeto, toda a parte da estrutura LoRaWAN formada por gateway, servidor de rede e servidor de aplicação foi fornecida pela empresa ProIoT. Essa empresa desenvolveu uma plataforma com um sistema que permite a criação de um ecossistema de dispositivos através da venda de conexões LoRaWAN, oferecendo pacotes de serviços para que usuários tenham acesso a toda estrutura de equipamentos montada no Brasil pela empresa American Tower Corporation (ATC). A ATC foi fundada em 1995 e hoje é uma das principais empresas do mundo de ativos imobiliários de comunicação, com e sem fio, onde um dos serviços oferecidos é o de locação personalizada da sua infraestrutura de torres, antenas e servidores que viabilizam a Internet das Coisas, eliminando a necessidade de investimentos em infraestrutura por parte de usuários interessados em implementações IoT [20]. Com isso, a ProIoT oferece através de uma loja dentro de sua plataforma, planos anuais de conexão com essa infraestrutura. Além de fornecer a interface para gerenciamento e criação de aplicações, podendo ativar, mover, excluir e desativar quantos dispositivos quiser, ela também permite a criação de dashboards através de widgets disponíveis e configuráveis, facilitando a visualização de dados recebidos na plataforma em tempo real [21]. Neste projeto, se fez uso dos serviços oferecidos pela ProIoT e nesta seção serão abordadas as etapas realizadas para conectar o circuito de monitoramento, formado pelos blocos anteriores (DHT11, sensor de umidade do solo e ESP32 com LoRa), com

a rede da ATC e com a plataforma ProIoT, desde a criação de uma conta até a configuração e montagem do dashboard.

Passo 1:

O primeiro passo foi acessar a plataforma ProIoT através do link: <https://app.proiot.com.br/login>, aceitar os termos e condições e criar uma conta, como mostrado na Figura 3.7.

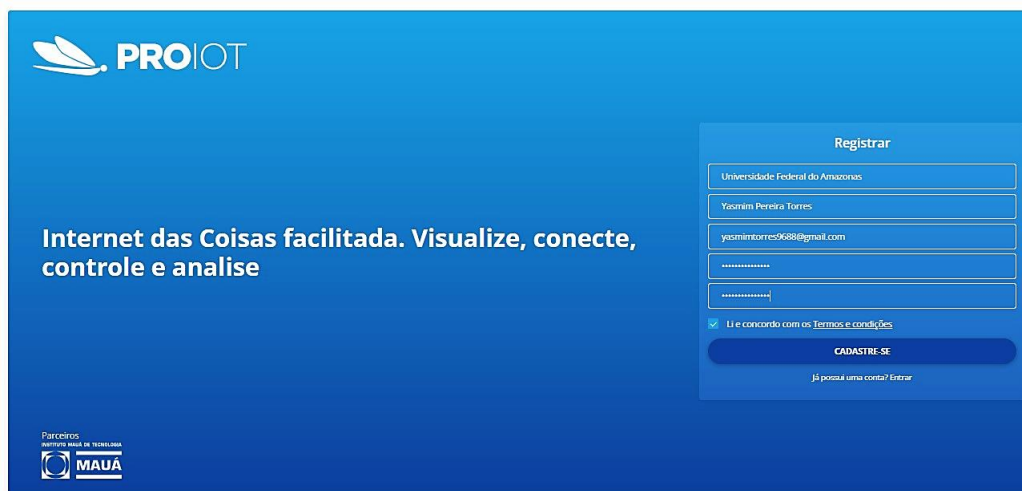


Figura 3.7: Cadastro do usuário na plataforma.

Fonte: Próprio autor.

Passo 2:

Após criar a conta e fazer login, seguiu-se para o menu *Loja*, e depois para a opção *Planos ATC LoRaWAN*. Com isso, foi adquirido o plano com maior quantidade de uplinks possível para que não houvesse interrupção do serviço em pouco tempo caso a quantidade de uplinks fornecida pelo plano contratado tenha sido atingida.

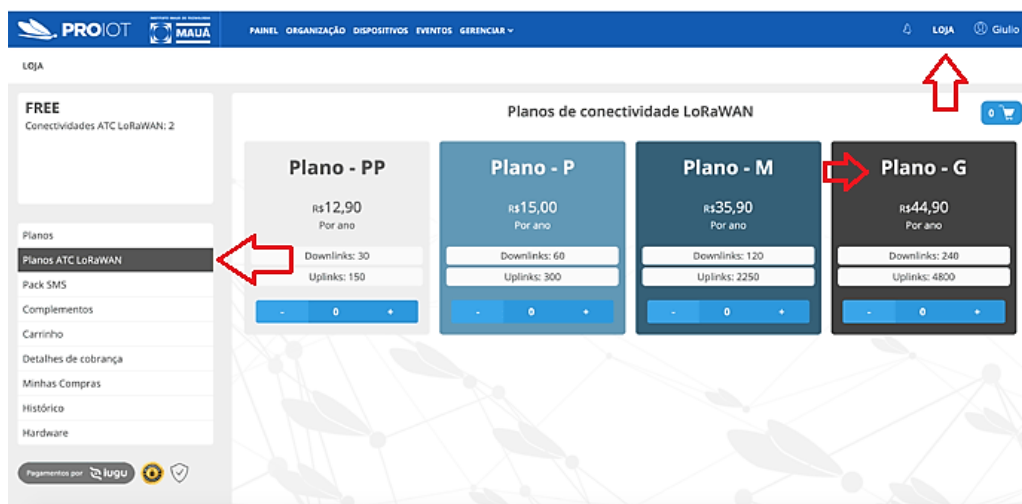


Figura 3.8: Aquisição de um plano ATC LoRaWAN.

Fonte: Próprio autor.

Passo 3:

Então, na aba *Dispositivos* deve-se acessar a opção + *Dispositivo*, em seguida escolher a rede *ATC LoRaWAN* como conector e customizar o dispositivo clicando em *Custom Device*. É obrigatório que os campos sejam preenchidos e que seja gerado um *DevEUI* aleatório ou que seja identificado no microcontrolador o *DevEUI* exclusivo, como mostrado na Figura 3.9

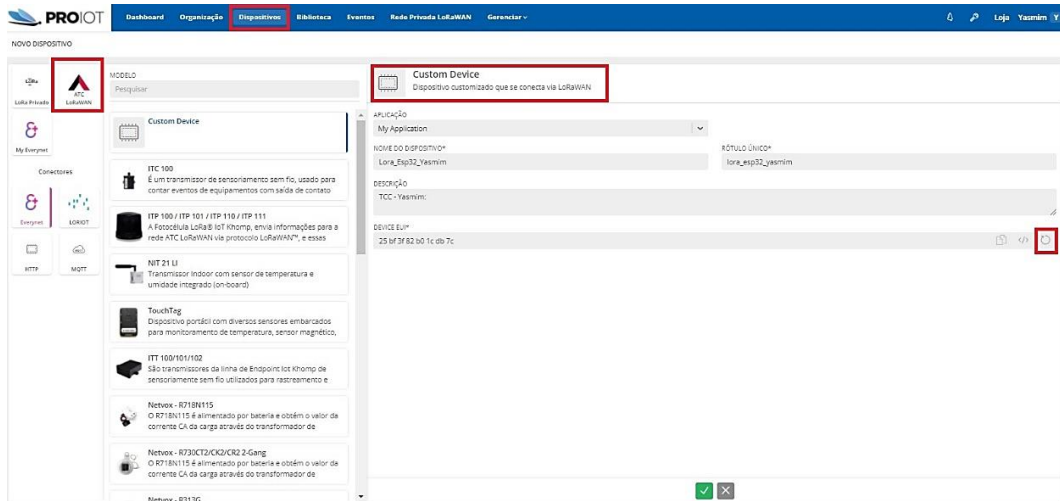


Figura 3.9: Customização do dispositivo.

Fonte: Próprio autor.

Passo 4:

Na aba *Dispositivos* foi acessada a opção do dispositivo recentemente criado. Em *Informações do dispositivo*, foi escolhido como *Ativação de Segurança* o modo *ABP*, por ser um modo mais simples e não volátil, e também foram geradas as seguintes chaves de acesso de forma aleatória: *Application Session Key*, *Network Session Encryption Key* e *Device Address*. Essas mesmas chaves devem ser utilizadas no código de monitoramento do ambiente no formato hexadecimal, habilitando a comunicação da placa TTGO T-beam com a rede LoRaWAN. A Figura 3.10 ilustra esse procedimento e a Figura 3.11 ilustra o trecho referente ao código.

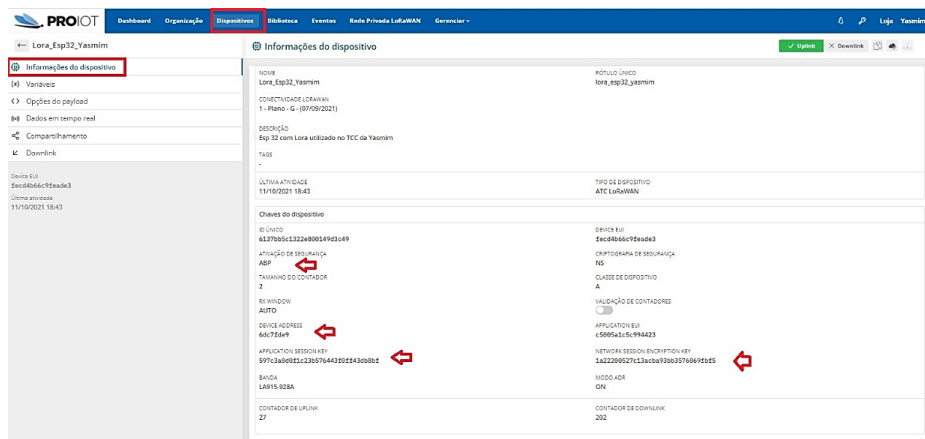


Figura 3.10: Acesso e configuração das informações do dispositivo.

Fonte: Próprio autor.

```

/* - Chaves (Network Keys) - */
static const PROGMEM u1_t NWKSKEY[16] = { 0x1A, 0x22, 0x20, 0x05, 0x27, 0xC1, 0x3A, 0xCB, 0xA9, 0x3B, 0xB3, 0x57, 0x60, 0x69, 0xFB, 0xF5 };
/* - Chaves (Application Keys) */
static const u1_t PROGMEM APPSKEY[16] = { 0x59, 0x7C, 0x3A, 0x0D, 0x0F, 0x1C, 0x23, 0xB5, 0x76, 0x44, 0x3F, 0x0F, 0xF4, 0x3D, 0xB8, 0xBF };
/* - Device Address - */
static const u4_t DEVADDR = 0x6dc7fde9;

```

Figura 3.11: Trecho do código de monitoramento mostrando que as mesmas chaves geradas na plataforma ProIoT foram adicionadas ao código principal.

Fonte: Próprio autor.

Passo 5:

Após os passos anteriores, ainda na aba *Dispositivos*, foi acessada a opção *Variáveis*. Nesta etapa é possível criar todas as variáveis que serão recebidas pela ProIoT pois, somente assim, a plataforma saberá onde armazenar cada uma das informações que serão recebidas posteriormente: temperatura do ambiente, umidade do ambiente e umidade do solo. Com isso, é necessário preencher todos os campos solicitados, incluindo pontos decimais, tipo e unidade. O item *Alias* é o pseudônimo de cada variável, todos os *Alias* criados são iguais aos que se encontram no código pois o pacote de uplink recebido pela plataforma é formado pelos *Alias* de cada variável e, logo em seguida, pelos valores dos componentes de umidade e temperatura, permitindo que haja a separação correta de cada informação. A Figura 3.12 ilustra esse procedimento e a Figura 3.13 ilustra o trecho referente ao código.

NOME DA VARIÁVEL	ALIAS	TIPO	UNIDADE	ÚLTIMO VALOR
Temperatura	Temp	Temp	°C	33.0°C
Umidade do ar	hum		%	28.0%
Umidade do solo	umidade_porcentual		%	73.0%

Figura 3.12: Criando as variáveis de umidade e temperatura na plataforma.

Fonte: Próprio autor.

```
umidade_percentual = map(umidade, 4095,3224, 0, 100);
Temp = dht.getTemperature();
hum = dht.getHumidity();
```

Figura 3.13: Trecho do código de monitoramento mostrando que os mesmos *Alias* criados na plataforma ProIoT são iguais aos *Alias* do código principal.

Fonte: Próprio autor.

Passo 6:

O próximo passo após a criação das variáveis foi as configurações de payload, o payload funciona como o corpo da informação, ele é quem carrega o conteúdo das informações enviadas em LoRa. No código do projeto o tamanho considerado para o payload foi 45, valor suficiente para transportar todas as informações de monitoramento necessárias. Ainda na aba *Dispositivos*, em *Opções de payload*, escolheu-se a opção *JSON* como o formato do payload e habilitou-se a opção *base 64* para que o payload seja codificado em base 64. A Figura 3.14 ilustra o trecho do código referente ao que foi citado.

```
DynamicJsonDocument json(JSON_OBJECT_SIZE(3));
json["umidade_percentual"] = umidade_percentual;
json["Temp"] = Temp;
json["hum"] = hum;
byte payload[45];
```

Figura 3.14: Trecho do código de monitoramento mostrando o tamanho do payload e as informações em formato json.

Fonte: Próprio autor.

Na parte inferior da tela é possível observar o payload recebido e toda a informação que ele carrega, bem como os dados decodificados no canto inferior direito, como na Figura 3.15.

Figura 3.15: Configurando as opções de payload.

Fonte: Próprio autor.

Passo 7:

Nesta etapa do processo é onde foram adicionados os widgets para criação de um dashboard completo que informa ao usuário, a cada 5 minutos, os devidos valores recebidos de umidade e temperatura do ambiente, bem como o valor da umidade do solo. Na aba *Dashboard*, em *+Widget*, foi escolhida a opção *Medidor*, esse mesmo tipo de widget foi usado para representar todas as outras informações de monitoramento. Com isso, configurou-se o nome do widget, pontos decimais, tamanho da fonte, ângulo, valor mínimo e máximo, quantidade de divisões e subdivisões, cor *etc.* No canto esquerdo da tela, deve ser selecionada a variável que estará associada a ele. Na Figura 3.16 é possível observar as configurações realizadas para este projeto dentro do widget de temperatura do ambiente.

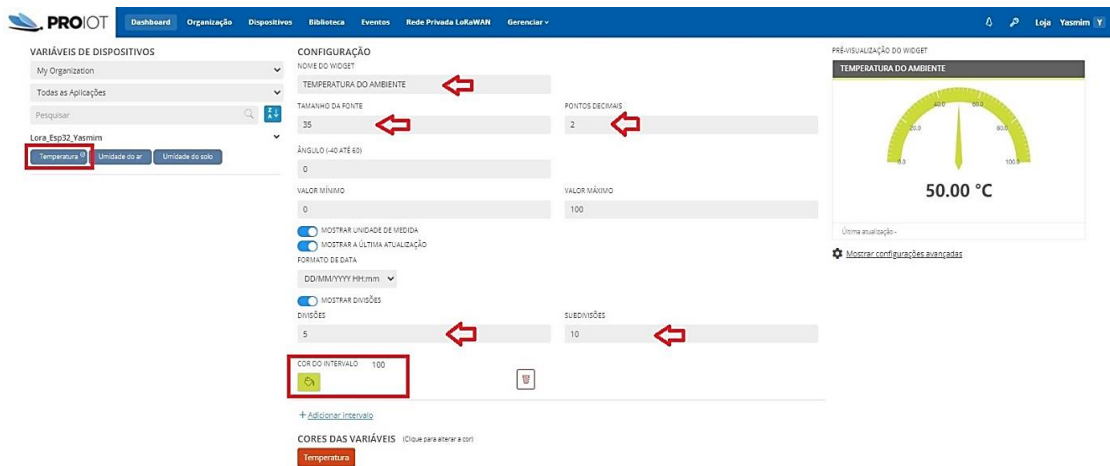


Figura 3.16: Configurando o widget de temperatura do ambiente.

Fonte: Próprio autor.

Outros dois tipos também foram adicionados, os widgets *Alerta de luz*, cuja função é emitir ao usuário uma sinalização, visual e escrita, quando a umidade do solo for considerada seca (entre 0% e 35%), úmida (entre 36% e 65%) ou molhada (entre 66% e 100%) e o widget *Gráfico de linhas*, para constatar a variação dos três componentes em um determinado período de tempo, nesse caso, nas últimas 24 horas. As configurações de cada um deles é mostrada nas Figuras 3.17 e 3.18. O resultado final do dashboard será apresentado na Seção 4.3.

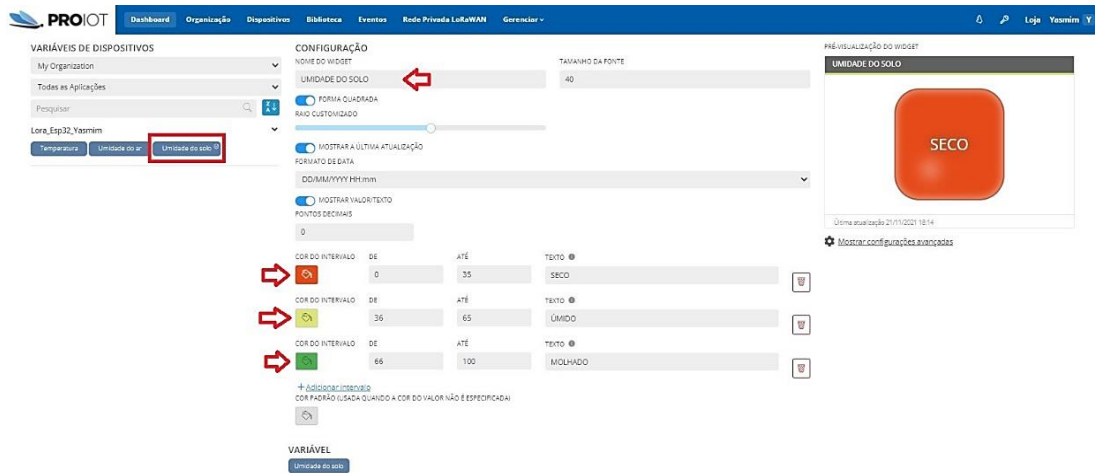


Figura 3.17: Configurando o widget *Alerta de luz* para umidade do solo.

Fonte: Próprio autor.

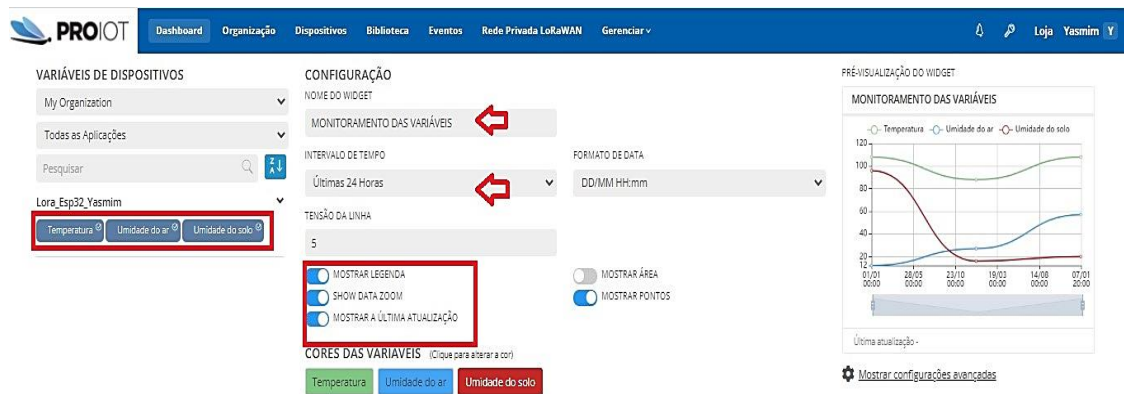


Figura 3.18: Configurando o widget *Gráfico de linhas* para visualizar a variação das informações recebidas nas últimas 24 horas.

Fonte: Próprio autor.

3.6 Visão geral

O sistema proposto do projeto de prevenção contra predadores é formado por três blocos, uma entrada e uma saída. Foram adquiridos três componentes fundamentais para implementação do circuito de prevenção: um sensor ultrassônico, uma placa ESP32-WROOM-32 e um buzzer ativo. O circuito visa detectar a aproximação de um possível predador (entrada) através do cálculo da distância entre o sensor ultrassônico e o predador, acionando um sinal sonoro (saída) enquanto a distância for menor ou igual a 35cm, buscando afastá-lo daquele local. A Figura 3.19 ilustra o diagrama em blocos que demonstra cada etapa e conexão do sistema.



Figura 3.19: Diagrama em blocos do sistema de prevenção contra predadores.

Fonte: Próprio autor.

3.7 Bloco sensor ultrassônico

O bloco sensor ultrassônico apresenta a implementação do sensor HC-SR04 e sua função dentro do projeto de prevenção contra predadores. Ele é responsável por detectar a presença de um animal que se aproxima do local que deve ser mantido em segurança e é formado por três itens principais: o transmissor ultrassônico, que emite as ondas ultrassônicas que serão refletidas, um receptor para identificar o eco do sinal e um circuito de controle responsável pelo cálculo do tempo entre o sinal emitido e o recebido. Esse componente possui 4 pinos, são eles: V_{CC} , GND , $Trigger$ e $Echo$. A Figura 3.20 ilustra o sensor ultrassônico HC-SR04 utilizado neste projeto:

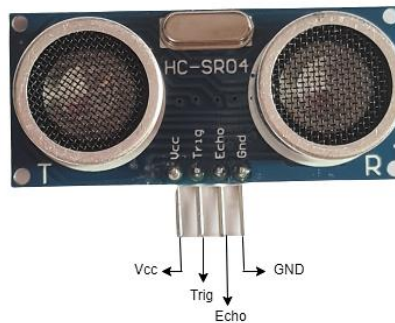


Figura 3.20: Sensor ultrassônico HC-SR04.

Fonte: Próprio autor.

O microcontrolador atrelado a esse componente envia um pulso, em nível alto, de pelo menos $10\mu s$ ao pino $Trigger$, quando o sensor ultrassônico recebe esse sinal, ele envia 8 pulsos de $40kHz$ e fica no aguardo de um sinal de retorno mantendo o pino $Echo$ em nível alto, até que o sinal de retorno seja detectado, o tempo de duração do pino $Echo$ em nível alto é igual ao tempo calculado entre envio e recebimento do sinal. Com isso, a distância entre emissor e receptor pode ser facilmente calculada, como demonstrado na equação (1). O sensor ultrassônico HC-SR04 atua dentro de um ângulo de abertura de 30° podendo detectar obstáculos de 2 centímetros a 4 metros de distância. O circuito foi montado em um protoboard de 830 furos, o pino V_{CC} e GND foram ligados às saídas de $3,3V$ e GND do microcontrolador

ESP32-WROOM-32, bem como os pinos *Trigger* e *Echo* foram ligados, respectivamente, aos pinos 25 e 26, do mesmo componente.

As especificações técnicas do sensor HC SR-04 são descritas a seguir [22]:

Especificações:

- ✓ Dimensão: 40mm x 20mm x 15mm.
- ✓ Peso: 10g.
- ✓ Tensão de alimentação: 3V ~ 5V.
- ✓ Corrente consumida: 15mA
- ✓ Frequência de operação: 40kHz.
- ✓ Distância alcançada: 2cm ~ 4m.
- ✓ Ângulo de medição: 30°.

Para funcionamento do sensor, foi desenvolvido um código em linguagem C no Arduino IDE fazendo uso da biblioteca *Ultrasonic.h*, compatível com o HC-SR04 que juntamente com a função *Ranging(CM)* retorna a distância em centímetros do obstáculo detectado a frente.

3.8 Bloco ESP32-WROOM-32

Esse bloco refere-se à comunicação do sensor ultrassônico com a placa ESP32-WROOM-32 de 4MB de memória flash, que possui microprocessador Xtensa 32 bits LX6 de baixo consumo de energia. Essa placa de desenvolvimento programável é da fabricante Espressif, com módulo NodeMCU. Ela atua na faixa de frequência de 2,4GHz até 2,5GHz e possui um conector micro-usb para alimentá-la ao ser conectada ao computador (5V/1A), não possuindo suporte para bateria [23]. A placa ESP32-WROOM-32 utilizada neste projeto, junto ao esquemático dos pinos, pode ser vista na Figura 3.21:

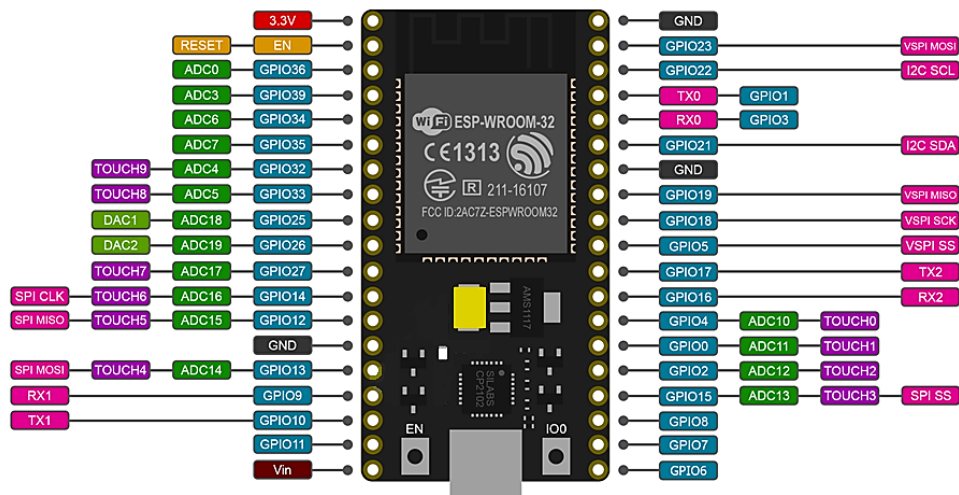


Figura 3.21: Pinagem ESP32-WROOM-32.

Fonte: [24].

O código associado a esse microcontrolador ligado ao sensor ultrassônico, foi desenvolvido no Arduino IDE, em linguagem C, fazendo uso da biblioteca *Ultrasonic.h*, contendo a declaração da variável, *distancia*, que armazena a distância calculada e a definição do pino do buzzer, que será explanado na próxima seção, como pino de saída, para que assim sejam desenvolvidas das condições e os laços de repetição. As portas do microcontrolador utilizadas neste projeto, além da saída de 3,3V e do pino *GND*, foram as portas 25, 26 e 32, ele foi montado no mesmo protoboard de 830 furos que o sensor ultrassônico.

3.9 Bloco buzzer

O bloco buzzer representa o componente buzzer ativo utilizado na implementação do projeto de prevenção contra predadores, com a função de emitir um sinal sonoro sempre que for solicitado pelo microcontrolador ESP-32-WROOM-32. Esse componente é encontrado no mercado como ativo ou passivo, o buzzer ativo, utilizado neste projeto, possui pinos de diferentes tamanhos, o maior sendo o V_{cc} e o menor o *GND*, para seu funcionamento é necessário apenas energizá-lo. O pino V_{cc} foi conectado à porta 32 do microcontrolador e o *GND* foi conectado ao *GND* do mesmo, ele foi montado no mesmo protoboard de 830 furos que o sensor ultrassônico e o ESP-32-WROOM-32. A Figura 3.22 ilustra esse componente:



Figura 3.22: Buzzer ativo.

Fonte: Próprio autor.

No código desenvolvido no Arduino IDE, em linguagem C, o pino V_{cc} do componente foi definido como pino de saída, `pinMode(pino_buzzer, OUTPUT)`, sendo aplicado dentro de um laço de repetição, `void loop()`, submetendo-se a duas condições: quando a variável `distancia` é menor ou igual a `35cm`, ou seja, quando o sensor ultrassônico capta um obstáculo à sua frente e a distância calculada a partir do eco recebido é menor ou igual a `35cm`, a função `digitalWrite(pino_buzzer, HIGH)` é acionada, permitindo que o buzzer fique em nível lógico alto disparando um sinal sonoro para afugentar o predador, impedindo sua aproximação. Quando a distância calculada for maior que `35cm`, a função `digitalWrite(pino_buzzer, LOW)` coloca o buzzer em nível lógico baixo, impedindo-o de emitir o som.

A Figura 3.23 ilustra o esquemático do circuito de prevenção contra predadores:

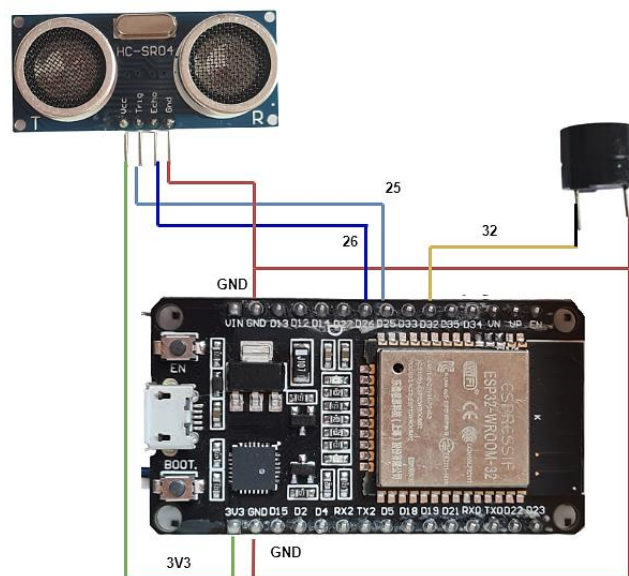


Figura 3.23: Esquemático do circuito de prevenção contra predadores formado pela placa ESP32-WROOM-32, sensor ultrassônico e buzzer.

Fonte: Próprio autor.

Capítulo 4

Cenários de teste e resultados

Neste capítulo, serão apresentados os cenários de teste e resultados dos dados obtidos nas duas implementações deste projeto: circuito de monitoramento do ambiente e do solo formado pelos seguintes componentes: um DHT11, um sensor de umidade do solo e um ESP32 com LoRa, responsável por captar os fatores de temperatura do ambiente, umidade do ambiente e umidade do solo e enviar para a plataforma ProIoT a fim de que as informações sejam divulgadas ao usuário em tempo real e atualizadas a cada 5 minutos. A segunda implementação é o circuito de prevenção contra predadores, formado por um sensor ultrassônico, um ESP32 e um buzzer cujo objetivo é afugentar possíveis predadores que queiram se aproximar do local que deve ser mantido em segurança, emitindo um sinal sonoro pelo buzzer toda vez que a distância calculada pelo ESP32, através do eco obtido pelo sensor ultrassônico, for menor ou igual a 35cm. Esse sinal sonoro é emitido enquanto a distância for mantida dentro dessa condição.

4.1 Cenário 1 – Teste de variação da umidade do solo

Neste cenário de teste, o sensor de umidade do solo é inserido dentro de três opções de umidade: solo seco, solo úmido e solo molhado. Ao inserir o sensor de umidade do solo no solo molhado, conforme é mostrado na Figura 4.1, a umidade captada pelo sensor, modulada por LoRa e mostrada no dashboard na plataforma ProIoT foi 87%, como ilustrado na Figura 4.2.



Figura 4.1: Sensor de umidade do solo inserido em solo molhado.

Fonte: Próprio autor.



Figura 4.2: Umidade do solo molhado sendo mostrada no dashboard da ProIoT.

Fonte: Próprio autor.

Ao inserir o sensor de umidade do solo no solo úmido, como mostra a Figura 4.3, a umidade captada e mostrada no dashboard foi 44%, como indicado na Figura 4.4.



Figura 4.3: Sensor de umidade do solo inserido em solo úmido.

Fonte: Próprio autor.

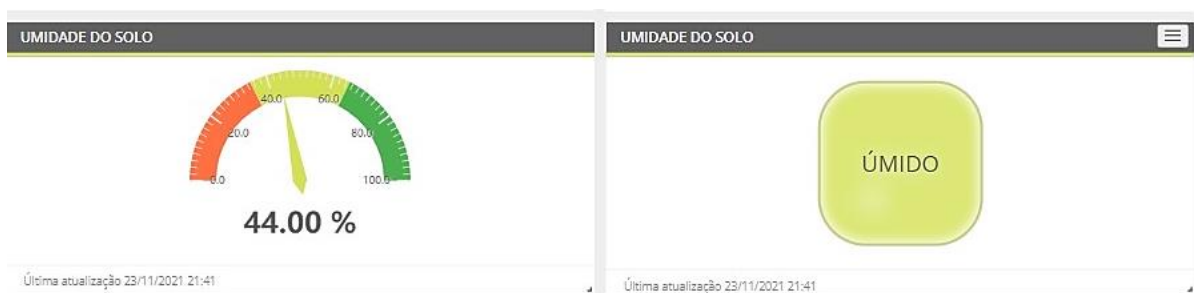


Figura 4.4: Umidade do solo úmido sendo mostrada no dashboard da ProIoT.

Fonte: Próprio autor.

Por fim, ao inserir o sensor de umidade do solo no solo seco, como mostra a Figura 4.5, a umidade captada e mostrada no dashboard foi 0%, como indicado na Figura 4.6.



Figura 4.5: Sensor de umidade do solo inserido em solo seco.

Fonte: Próprio autor.



Figura 4.6: Umidade do solo seco sendo mostrada no dashboard da ProIoT.

Fonte: Próprio autor.

4.2 Cenário 2 – Teste de variação da temperatura e da umidade do ar

No segundo cenário de teste, o sensor de umidade e temperatura do ambiente, DHT11, é responsável por captar os valores de umidade e temperatura do ambiente para que sejam enviados por LoRa e em seguida mostrados no dashboard da ProIoT. A Figura 4.7 ilustra o sensor DHT11, captando os valores de umidade e temperatura do ambiente em ambiente real e aberto, esses valores são 29° de temperatura e 55% de umidade. Na Figura 4.8 o sensor é submetido à ação de uma fonte externa de calor e é possível observar a variação da temperatura e da umidade para 37° e 36% respectivamente. O circuito completo com DHT11, sensor de umidade do solo e ESP32 com LoRa foi montado dentro de uma case com tampa, sendo possível observar o DHT11 parcialmente para o lado de fora através de uma pequena abertura na case.



Figura 4.7: DHT11 captando valores de umidade e temperatura em ambiente real e aberto.

Fonte: Próprio autor.



Figura 4.8: DHT11 captando valores de umidade e temperatura após ser submetido à ação de uma fonte externa de calor.

Fonte: Próprio autor.

4.3 Análise dos dados recebidos

O gráfico de linhas, com a variação dos componentes no tempo, pode ser observado na Figura 4.9. As linhas nas cores verde, azul e vermelho representam a variação da temperatura do ambiente, umidade do ambiente e umidade do solo, respectivamente. A umidade do solo foi o parâmetro que mais variou no intervalo de tempo de 24 horas, se mantendo dentro da faixa de 75% e 95%.



Figura 4.9: Gráfico de linhas com a variação dos componentes no tempo.

Fonte: Próprio autor.

A Figura 4.10 mostra o circuito completo de monitoramento em funcionamento durante os testes executados.

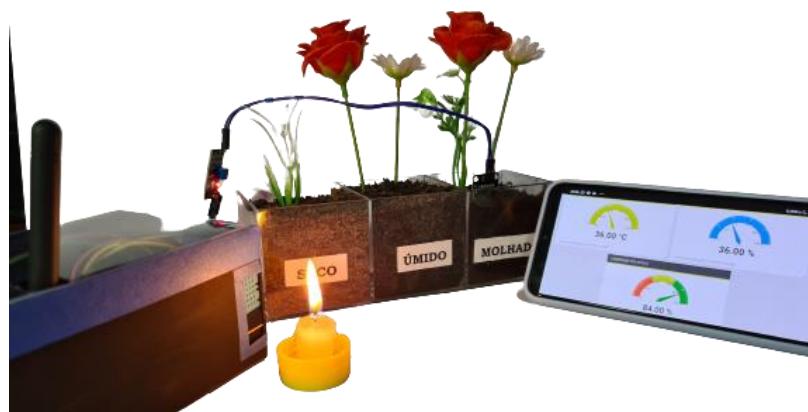


Figura 4.10: Circuito completo de monitoramento.

Fonte: Próprio autor.

Na Figura 4.11 pode-se observar o dashboard completo, é dessa forma que o usuário monitora os fatores ambientais e do solo destacados neste projeto.

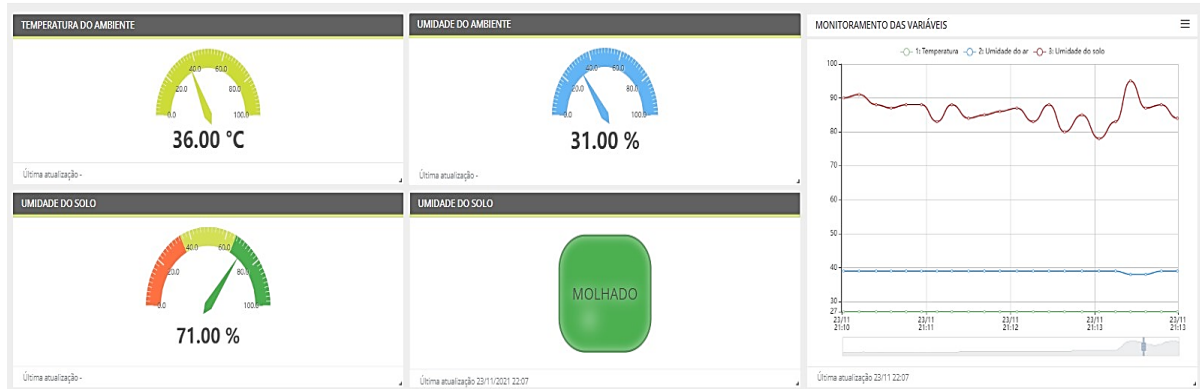


Figura 4.11: Dashboard completo.

Fonte: Próprio autor.

Na plataforma ProIoT, na aba *Dados em tempo real*, é possível acompanhar parâmetros importantes de todos os pacotes de uplink recebidos na plataforma, por exemplo, largura de banda, fator de espalhamento, taxa de codificação, relação sinal-ruído, intensidade do sinal, canal de frequência, payload com as informações em base 64, latitude, longitude e altura onde o gateway, relacionado a esse pacote, foi implantado, conforme ilustra a Figura 4.12.

(00) Dados em tempo real

23/NOV 22:07:09	Uplink	Frequency: 915.6 Signal quality: -8.2dB/-117dB
23/NOV 22:06:57	Uplink	Frequency: 916 Signal quality: -4.2dB/-112dB
23/NOV 22:06:46	Uplink	Frequency: 916.4 Signal quality: -13.8dB/-119dB
<pre> "modulation": "Q", "bandwidth": 125000, "type": "LORA", "spreading": 12, "coderate": "4/5" }, "hardware": { "status": 1, "chain": 1, "tnst": 356414308, "snr": -13.8, "rssi": -119, "channel": 6, "gps": { "lat": -3.098999977118164, "lon": -50.0619614831543 } } </pre>		
23/NOV 22:06:34	Uplink	Frequency: 916.6 Signal quality: 4.5dB/-99dB
23/NOV 22:06:23	Uplink	Frequency: 916 Signal quality: -3.5dB/-114dB
23/NOV 22:06:12	Uplink	Frequency: 916.2 Signal quality: -4dB/-109dB
23/NOV 22:06:00	Uplink	Frequency: 915.8 Signal quality: -8.2dB/-117dB
23/NOV 22:05:48	Uplink	Frequency: 916.6 Signal quality: -4.8dB/-112dB

Figura 4.12: Dados em tempo real recebidos pela plataforma ProIoT.

Fonte: Próprio autor.

Nesse pacote de uplink, por exemplo, a largura de banda foi de 125kHz , com fator de espalhamento 12, taxa de codificação de $4/5$, relação sinal ruído de $-13,8\text{dB}$, intensidade do

sinal de $-119dB$, canal de frequência de $916,4MHz$, latitude do gateway de $-3,0989999771118164$, longitude do gateway de -59.9410514831543 e altura do gateway na torre de transmissão de $99m$, totalizando uma distância de $8,8km$ entre transmissor e receptor, além disso, o payload carregava as seguintes informações de monitoramento: umidade e temperatura de 32% e 37° e umidade do solo de 85% .

4.3 Cenário 3 – Variação da distância do predador

O último cenário de teste, refere-se ao segundo circuito implementado neste projeto, o circuito de prevenção contra predadores, onde o sensor ultrassônico e o ESP32 trabalham juntos para descobrir a distância do obstáculo que se aproxima colocando o buzzer em nível alto, ou seja, emitindo um sinal sonoro, enquanto a distância for menor ou igual a $35cm$. A Figura 4.13 ilustra o circuito montado dentro de uma case com tampa transparente, com o sensor ultrassônico parcialmente para fora, ilustra também a representação do predador à uma distância menor que $35cm$, ou seja, com buzzer emitindo um sinal sonoro. No código desenvolvido, a distância calculada em centímetros é mostrada pela Serial do Arduino IDE, como mostra a Figura 4.14.

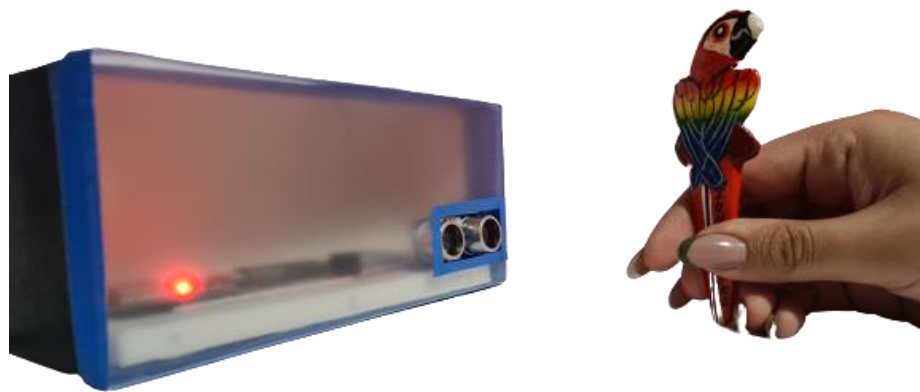
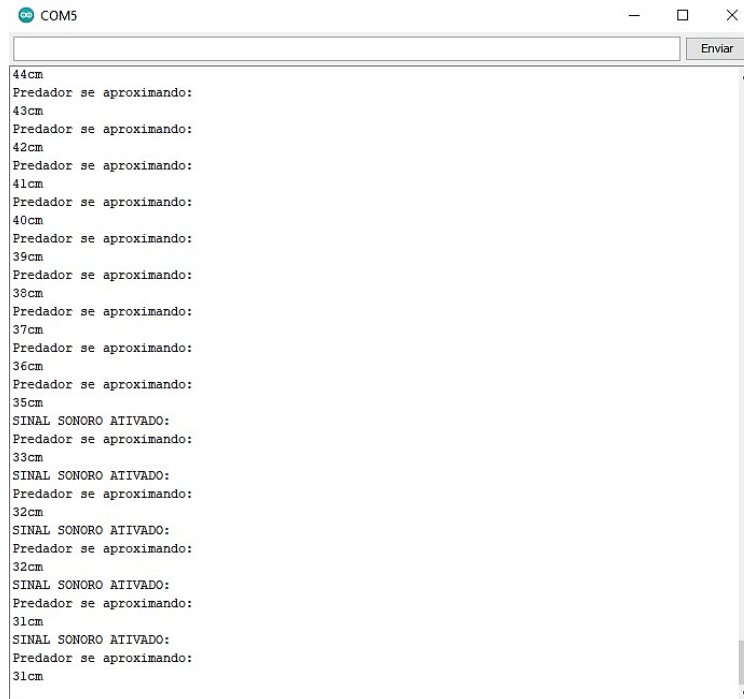


Figura 4.13: Sensor ultrassônico e ESP32 captando a presença de um predador a uma distância menor que $35cm$.

Fonte: Próprio autor.



```
COM5
Enviar
44cm
Predador se aproximando:
43cm
Predador se aproximando:
42cm
Predador se aproximando:
41cm
Predador se aproximando:
40cm
Predador se aproximando:
39cm
Predador se aproximando:
38cm
Predador se aproximando:
37cm
Predador se aproximando:
36cm
Predador se aproximando:
35cm
SINAL SONORO ATIVADO:
Predador se aproximando:
33cm
SINAL SONORO ATIVADO:
Predador se aproximando:
32cm
SINAL SONORO ATIVADO:
Predador se aproximando:
32cm
SINAL SONORO ATIVADO:
Predador se aproximando:
31cm
SINAL SONORO ATIVADO:
Predador se aproximando:
31cm
```

Figura 4.14: Serial do ambiente de desenvolvimento integrado Arduino mostrando a distância calculada à medida que o obstáculo se aproxima.

Fonte: Próprio autor.

Capítulo 5

Conclusão

A Internet das Coisas se baseia na comunicação entre sensores e atuadores com toda uma estrutura de rede inteligente, permitindo a conexão de dispositivos com a internet. A tecnologia LoRa revolucionou o conceito de IoT, por ser caracterizada pelo seu baixo custo, baixo consumo e longo alcance na transmissão dos pacotes de dados. Os conceitos apresentados neste trabalho, voltados para a IoT, LoRa e LoRaWAN colaboraram para um melhor entendimento a respeito do tema, possibilitando a implementação do projeto com base no que foi estudado de forma teórica. O desenvolvimento do código, a escolha dos componentes e a montagem dos circuitos garantiram um aprofundamento prático em relação ao monitoramento de fatores importantes no setor da agricultura. Com os testes realizados, percebeu-se a viabilidade do sistema como um todo ao detectar umidade, temperatura e distância com um tempo de resposta satisfatório e uma quantidade de erros mínima, mas principalmente pelo fato de que as informações podem ser acessadas em tempo real, de forma remota, em uma interface de fácil interpretação pelo usuário. Com isso, pode-se considerar que o projeto é promissor, mas que algumas melhorias o tornariam mais robusto e atrativo ao mercado agrícola como, por exemplo, a adição de uma válvula solenoide para irrigação do plantio de forma automatizada, o uso de um sensor de umidade do solo com maior resistência à corrosão e a junção dos dois circuitos apresentados, o de monitoramento e o de prevenção, fazendo uso de um só microcontrolador. Deseja-se que este projeto incentive trabalhos futuros na área de Internet das Coisas voltados não só para o setor agrícola, mas para tantos outros que demandam aplicações em IoT para o melhor funcionamento de seus dispositivos e fornecimento de serviços à sociedade.

Referências Bibliográficas

- [1] SRIVASTAVA, D., et al., *Measurement of Temperature and Humidity by using Arduino Tool and DHT11*. International Research Journal of Engineering and Technology, [S.l], v.5, p.1-2, 2018.
- [2] HOW TO MECHATRONICS., *DHT11 & DHT22 Sensors Temperature and Humidity Tutorial using Arduino*. [S. l.], 2016. Disponível em: <https://howtomechatronics.com/tutorials/arduino/dht11-dht22-sensors-temperature-and-humidity-tutorial-using-arduino/>. Acesso em: 10 out. 2021.
- [3] NARAYAN, R., *Measuring Humidity Using Sensor (DHT11)*. [S. l.], 2019. Disponível em: <https://www.instructables.com/Measuring-Humidity-Using-Sensor-DHT11/>. Acesso em: 11 out. 2021.
- [4] ZHANG, X., et al., *Research and application of a new soil moisture sensor*. MATEC Web of Conferences, [S. l.], p. 2-3, 2018.
- [5] ELECFREAKS, *Ultrasonic Ranging Module HC - SR04*. [S. l.], 2015. Disponível em: <https://www.robocore.net/sensor-robo/sensor-de-distancia-ultrassonico-hc-sr04>. Acesso em: 13 out. 2021.
- [6] MARINO, D., et al. *Jardim Inteligente IoT - JIIOT*. Revista Tecnologia, [S. l.], v. 38, n. 1, p. 39-41, 2017.
- [7] CENTENARO, M., et al., *Long-Range Communications in Unlicensed Bands: The Rising Stard in The IoT and Smart City Scenarions*. IEEE Wireless Communications , [S. l.], n. 1, p. 60-63, 2016.
- [8] CUNHA, L., VALIM, P., *Rede de Sensores Sem Fio para Monitoramento de Variáveis de Ambiente*. XI Computer on the Beach, Balneário Camboriú, p. 2-3, 2 set. 2020.
- [9] Cattani, M., et al., *An Experimental Evaluation of the Reliability of LoRa Long-Range Low-Power Wireless Communication*. Journal of sensor and actuator networks. [S.l], p.4-6, 2017.
- [10] DUCROT, N., RAY, D., et al., *LoRa Device Developer Guide*. Orange IoT Solutions, [S.l], p.3-4, 2017.
- [11] ALLINCE, L., *LoRaWAN What is it? A technical overview of LoRa and LoRaWAN*. LoRa Alliance Technical Marketing Workgroup, [S.l], p.8-10, 2015.
- [12] SEMTECH, C., *LoRa and LoRaWAN: A Technical Overview*. Semtech Corporation, [S.l], p.8-12, 2019

- [13] ARAS, E., et al., *Exploring The Security Vulnerabilities of LoRa*. IEE, Belgium, p.8-12, 2019.
- [14] AOSONG, *Temperature and humidity module DHT11 Product Manual*. [S. l.], 2018. Disponível em: <https://www.robocore.net/sensor-ambiente/sensor-de-temperatura-dht11>. Acesso em: 29 out. 2021.
- [15] GAY, W., *Mastering the Raspberry Pi*. [S.l.]: Apress, 2014. ISBN 978-1-4842-0182-4
- [16] GITHUB, *DHTesp*. [S.l.], 2021. Disponível em: <https://github.com/beegee-tokyo/DHTesp>. Acesso em: 2 nov. 2021.
- [17] YU, L., et al., *Review of research progress on soil moisture sensor technology*. Int J Agric & Biol Eng, [S.l.], v.14, p.3-4, 2021.
- [18] YEFREMOV, Y., *TTGO T-Beam*. [S. l.], 2021. Disponível em: https://doc.riotos.org/group_boards__esp32__ttgo-t-beam.html. Acesso em: 1 nov. 2021.
- [19] SEMTECH, C., *Semtech SX1276*. [S. l.]. Disponível em: <https://www.semtech.com/products/wireless-rf/lora-core/sx1276>. Acesso em: 5 nov. 2021.
- [20] AMERICAN T., *Empresa*. [S. l.]. Disponível em: <https://americantower.com.br/pt/company/>. Acesso em: 5 nov. 2021.
- [21] PROIOT, *Sobre*. [S. l.]. Disponível em: <https://americantower.com.br/pt/company/>. Acesso em: 5 nov. 2021.
- [22] MORGAN, E., *HC-SR04 Ultrasonic Sensor*. [S. l.]. Disponível em: <https://datasheetspdf.com/datasheet/search.php?sWord=hc-sr04>. Acesso em: 8 nov. 2021.
- [23] ESPRESSIF, *ESP32-WROOM-32 Datasheet*. [S. l.], v.3.2. Disponível em: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf. Acesso em: 11 nov. 2021.
- [24] KOYANAGI, F., *ESP32: Detalhes internos e pinagem*. [S. l.]. Disponível em: <https://www.fernandok.com/2018/03/esp32-detalhes-internos-e-pinagem.html>. Acesso em: 11 nov. 2021.