

**UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA
ENGENHARIA ELÉTRICA**

HOLYVERS VINÍCIUS MORAIS DE OLIVEIRA

**DESENVOLVIMENTO DE UM SISTEMA DE MONITORAMENTO
REMOTO PARA OTIMIZAÇÃO DA COLETA SELETIVA EM MANAUS**

MANAUS

2023

HOLYVERS VINÍCIUS MORAIS DE OLIVEIRA

**DESENVOLVIMENTO DE UM SISTEMA DE MONITORAMENTO
REMOTO PARA OTIMIZAÇÃO DA COLETA SELETIVA EM MANAUS**

Projeto de pesquisa desenvolvido durante a disciplina de Trabalho de Conclusão de Curso II e apresentado à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Fábio de Sousa Cardoso

Manaus

2023

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST

Reitor:

André Luiz Nunes Zogahib

Vice-Reitor:

Kátia do Nascimento Coureiro

Diretora da Escola Superior de Tecnologia:

Ingrid Sammyne Gadelha Figueiredo

Coordenador do Curso de Engenharia Elétrica:

Israel Gondres Torné

Banca Avaliadora composta por:

Data da defesa: 31/08/2023.

Prof. Fábio de Sousa Cardoso, Dr. (Orientador)

Prof. Antonio Luiz Alencar Pantoja, Dr.

Prof. Jozias Parente de Oliveira, Dr.

CIP – Catalogação na Publicação

Oliveira, Holyvers Vinícius Morais de

Desenvolvimento de um sistema de monitoramento remoto para otimização da coleta seletiva em Manaus / Holyvers Vinícius Morais de Oliveira; [orientado por] Fábio de Sousa Cardoso. – Manaus: 2023. 63 p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica). Universidade do Estado do Amazonas, 2023.

1. IoT. 2. LoRa. 3. Sistemas Embarcados.
- I. Cardoso, Fábio de Sousa.

HOLYVERS VINÍCIUS MORAIS DE OLIVEIRA

DESENVOLVIMENTO DE UM SISTEMA DE MONITORAMENTO REMOTO PARA
OTIMIZAÇÃO DA COLETA SELETIVA EM MANAUS


Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheiro Eletricista.

Nota obtida: 10,0 (dez vírgula zero)

Aprovada em 31/08/2023

Área de concentração: Internet das Coisas

BANCA EXAMINADORA



Orientador: Fábio de Sousa Cardoso, Dr.

Antonio Luiz Alencar Pantoja Assinado de forma digital por Antonio
Luiz Alencar Pantoja
Dados: 2023.09.05 16:47:24 -04'00'

Avaliador: Antônio Luiz Alencar Pantoja, Dr.

JOZIAS PARENTE DE Assinado de forma digital por JOZIAS
PARENTE DE OLIVEIRA:44545142249
Dados: 2023.09.05 18:50:09 -03'00'
OLIVEIRA:44545142249

Avaliador: Jozias Parente de Oliveira, Dr.

Manaus
2023

Dedico este trabalho a Jesus, o Rei dos Reis e Senhor dos Senhores que está comigo desde antes de nascer e que tem me sustentado até hoje, sem ele não teria chegado aqui.

AGRADECIMENTOS

Agradeço primeiramente a Deus que por meio do seu filho Jesus e do consolador Espírito Santo, me deu por bondade e amor, a vida que eu não merecia e que me sustenta desde o dia do meu nascimento e que me deu forças para chegar até aqui apesar das derrotas, fracassos e tristezas vivenciadas, me concedeu vitórias extraordinárias. À ele seja dada a glória, honra, força, sabedoria, riquezas, poder e louvor para todo o sempre amém.

Agradeço à minha mãe Leila Maria Moraes de Lira, que cuida de mim desde o ventre e que tem sido a minha melhor amiga para quem eu posso compartilhar qualquer situação da minha vida para sempre me dar apoio e ajuda na tomada de decisões.

Agradeço ao meu pai José Inácio de Oliveira, que me forneceu a oportunidade de estudos desde cedo através do apoio financeiro, transporte e incentivo para que eu alcançasse a educação que eu tenho hoje. Também agradeço por ele me passar mesmo que indiretamente, experiências da vida sobre o que se deve ou não fazer.

Agradeço ao meu querido irmão Victor Holyvers Moraes de Oliveira que me acompanha nessa jornada desde tenra idade, seja nas brincadeiras ou nos pensamentos e questões filosóficas da vida, nas quais me encontro admirado em observar a sua notória inteligência e raciocínio sobre tais assuntos.

Agradeço aos amigos que conheci na EST os quais jamais imaginaria que encontraria na vida. Eles fizeram minha experiência acadêmica ser divertida e não somente presa aos estudos. Muitos dos cafés, almoços e lanches no RU, passagens de tempo nas salas dos blocos A, B, C ou D, seja estudando, jogando conversa fora ou dormindo ficarão guardadas em minhas memórias. Obrigado por serem as pessoas que mais se fizeram presentes na minha vida acadêmica, desde o ensino fundamental até o ensino superior. São eles: Rubem Silas, Ana Clara Begnini, Paulo Henrique, Paulo Victor, Isabel Priscila, Marcus Cunha, Felipe Moraes, Josué Alves, Caio Bentes, Raissa Vasconcelos, dentre outros.

Agradeço ao meu orientador Fábio de Sousa Cardoso que aceitou orientar o meu projeto e agradeço ao corpo docente e discente da Escola Superior de Tecnologia que me forneceu experiências que me forjaram e me trouxeram a maturidade.

RESUMO

À medida que a produção de resíduos em áreas urbanas aumenta, o processo de coleta segue ineficiente. Isso se deve à falta de políticas públicas na gestão dos resíduos o que pode ser visto no planejamento das rotas de coleta. Geralmente, as rotas são realizadas sem haver preocupação se as lixeiras estão cheias ou não naquele instante, consequentemente, ocorre desperdício de combustível e detrimento à mobilidade urbana por parte dos transportes coletores. Além disso, por não terem sido recolhidos no tempo devido, tem-se o transbordamento dos coletores de resíduos degradando o meio ambiente pela emissão de gases poluentes e gerando impactos negativos na sociedade como a transmissão de doenças, proliferação de insetos e ampliação de gastos com a limpeza pública. O presente trabalho tem como objetivo o desenvolvimento do protótipo de um sistema de monitoramento remoto voltado para a coleta seletiva de resíduos recicláveis aplicado à cidade de Manaus localizada no estado do Amazonas. Isso foi realizado por meio de tecnologia de comunicação LoRa, protocolo de comunicação MQTT e um sensor ultrassônico para monitoramento do nível de enchimento desses resíduos presentes em uma lixeira e da sua localização. A partir dos dados obtidos, enviaram-se essas informações para uma plataforma IoT conhecida como TagoIO com o intuito de apresentá-las em *Dashboards*. Para representação dos resíduos recicláveis no projeto, utilizaram-se materiais para o papel, plástico, metal e vidro, os quais constituíram-se em folhas de papel, garrafas plásticas, latas de metal e copos de vidro, respectivamente. Ao final são apresentadas 4 tabelas em que cada uma contém as informações obtidas de 5 casos de testes realizados para cada tipo de material reciclável. Além disso, são exibidos 4 *Dashboards* nos quais cada um mostra os dados referentes a um caso de teste para determinado resíduo reciclável expostos nas tabelas apresentadas com a finalidade de uma análise geral desses dados.

Palavras-chave: Coleta Seletiva, Monitoramento Remoto, IoT, LoRa.

ABSTRACT

As urban areas continue to witness an increase in waste production, the efficiency of the waste collection process lags behind. This is attributed to the lack of public policies in waste management, which is evident in the planning of collection routes. Generally, routes are executed without considering whether the bins are full at that moment, resulting in fuel wastage and urban mobility challenges for waste collection vehicles. Furthermore, due to the delay in collection, waste bins overflow, leading to environmental degradation through the emission of pollutants and negative societal impacts such as disease transmission, insect proliferation, and increased public cleaning expenses. The present study aims to develop a prototype of a remote monitoring system specifically tailored for the selective collection of recyclable waste in the city of Manaus, located in the Amazonas state. This endeavor was achieved through the utilization of LoRa communication technology, the MQTT communication protocol, and an ultrasonic sensor for monitoring the fill level of these waste bins and their location. Based on the collected data, this information was transmitted to an IoT platform known as TagoIO with the purpose of presenting it in Dashboards. To represent recyclable waste materials in the project, materials such as paper, plastic, metal, and glass were used, which consisted of pieces of paper, plastic bottles, metal cans, and glass cups, respectively. At the end, 4 tables are presented, each containing the information obtained from 5 test cases conducted for each type of recyclable material. Additionally, 4 Dashboards are displayed, with each one showing the data related to a test case for a specific recyclable waste type, as presented in the tables, for an overall analysis of this data.

Keywords: Selective Collection, Remote Monitoring, IoT, LoRa.

Lista de Figuras

1	Coletores para coleta seletiva.	17
2	Representação do conceito de Internet das Coisas.	18
3	Diagrama de operação do sensor ultrassônico.	19
4	Comparação de LPWAN com outras redes usadas em IoT.	19
5	Arquitetura LPWAN.	20
6	Representação do Chirp.	22
7	Espectro de uma transmissão LoRa.	22
8	Frame LoRa.	23
9	Exemplos de Topologias de Rede.	24
10	Relação do modelo OSI com o TCP/IP.	26
11	Arquitetura MQTT.	26
12	TagoIO Logo.	28
13	Espressif Systems Logo.	28
14	Arduino Logo.	29
15	Arquitetura Geral do Sistema.	30
16	Sensor de distância HCSR-04.	31
17	Módulo ESP32 Heltec Wifi LoRa (V2).	32
18	Esquema de conexão do Transmissor.	33
19	Fluxograma do Algoritmo do Transmissor.	35
20	Coletor Transmissor.	38
21	Materiais utilizados para representação dos resíduos recicláveis.	39
22	Preparação do coletor para os testes.	40
23	Receptor.	41
24	Fluxograma do Algoritmo do Receptor.	43
25	Tela inicial da TagoIO.	45
26	Tela inicial após o <i>Sign In</i>	45
27	Tela para adição de um dispositivo.	46
28	Tela de inserção do nome do dispositivo na TagoIO.	46
29	Tela de confirmação da criação do dispositivo.	47

30	Tela de visão geral do dispositivo.	47
31	Tela de informações do <i>Token</i>	48
32	Tela de visão geral <i>Action</i>	49
33	Tela de visão geral dos Tópicos MQTT.	50
34	Tela de visão geral para adição dos <i>Dashboards</i>	50
35	Tela de visão geral para adição do <i>template</i> dos <i>Dashboards</i>	51
36	Tela de visão geral para adição da localização do coletor.	52
37	Transmissor em funcionamento.	53
38	Receptor em funcionamento.	53
39	Exibição dos dados na TagIO.	54
40	<i>Dashboard</i> referente ao papel.	57
41	<i>Dashboard</i> referente ao plástico.	58
42	<i>Dashboard</i> referente ao vidro.	59
43	<i>Dashboard</i> referente ao Metal.	60

Lista de Tabelas

1	Principais parâmetros do Sensor HCSR-04	31
2	Especificações técnicas do Heltec Wifi LoRa 32 (V2)	32
3	Bibliotecas utilizadas no Algoritmo implementado no Transmissor	34
4	Bibliotecas utilizadas no Algoritmo implementado no Receptor	42
5	Dados obtidos para o material Papel	55
6	Dados obtidos para o material Plástico	55
7	Dados obtidos para o material Vidro	56
8	Dados obtidos para o material Metal	56

LISTA DE ABREVIATURA

3GPP	<i>3rd Generation Partnership Project</i>
AM	Amazonas
BLE	<i>Bluetooth Low Energy</i>
CRC	<i>Cyclic Redundancy Check</i>
CSS	<i>Chirp Spread Spectrum</i>
FSK	<i>Frequency Shift Keying</i>
GSM	<i>Global System Mobile</i>
I2C	<i>Inter-Integrated Circuit</i>
IDE	<i>Integrated Development Environment</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
IPTU	Imposto Predial e Territorial Urbano
LoRa	<i>Long Range</i>
LoRaWAN	<i>Long Range Wide Area Network</i>
LPWAN	<i>Low Power Wide Area Network</i>
LTE-M	<i>Long Term Evolution for Machines</i>
M2M	<i>Machine to Machine</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
OSI	<i>Open Systems Interconnection</i>
RSU	Resíduos Sólidos Urbanos
SF	<i>Spreading Factor</i>
SPI	<i>Serial Peripheral Interface</i>
TCP	<i>Transmission Control Protocol</i>

SUMÁRIO

INTRODUÇÃO	14
1 REFERENCIAL TEÓRICO	16
1.1 COLETA SELETIVA DE LIXO	16
1.2 INTERNET DAS COISAS	17
1.3 SENSORES	18
1.4 LPWAN	19
1.4.1 LORA	21
1.5 MQTT	25
1.6 TAGOIO	27
1.7 ESP32	28
1.8 ARDUINO IDE	28
2 METODOLOGIA	30
3 IMPLEMENTAÇÃO	31
3.1 DESENVOLVIMENTO DO TRANSMISSOR	31
3.1.1 SENSOR HCSR-04	31
3.1.2 ESP32 HELTEC WIFI LORA (V2)	32
3.1.3 CONEXÃO DO SENSOR AO ESP32	33
3.1.4 DESENVOLVIMENTO DO SOFTWARE DO TRANSMISSOR	34
3.1.5 MONTAGEM DO COLETOR	37
3.2 DESENVOLVIMENTO DO RECEPTOR	40
3.2.1 DESENVOLVIMENTO DO SOFTWARE DO RECEPTOR	41
3.3 CADASTRO DOS DISPOSITIVOS NA TAGOIO	45
4 RESULTADOS E DISCUSSÕES	53
CONCLUSÃO	61
REFERÊNCIAS BIBLIOGRÁFICAS	62

INTRODUÇÃO

Resíduos Sólidos Urbanos (RSU) são aqueles originários de atividades domésticas em residências urbanas e os originários da varrição, limpeza de logradouros e vias públicas e outros serviços de limpeza urbana (SINIR, 2020). Entre 2010 e 2019, a geração de RSU no Brasil registrou considerável incremento, passando de 67 milhões para 79 milhões de toneladas por ano (ABRELPE, 2020). Durante o ano de 2020, em decorrência da pandemia, a geração de resíduos sólidos urbanos no Brasil sofreu um aumento chegando a cerca de 82,5 milhões de toneladas (ABRELPE, 2021). Até 2050, o Brasil observará um aumento de quase 50% no montante de RSU, em comparação ao ano base de 2019 (ABRELPE, 2020).

O descarte de lixo se não recolhido adequadamente, gera impactos tanto no meio ambiente como na sociedade. Desde a contaminação de rios e solos até disseminação de doenças, se torna imprescindível que haja uma coleta de resíduos feita com eficácia. Esse processo é constantemente ineficiente, pois existe um *déficit* em sua gestão pelas partes administrativas. As rotas de coleta são estáticas, ou seja, os caminhões de coleta seguem rotas previamente planejadas, desconsiderando o fato dos coletores de resíduos (lixeiras) se encontrarem realmente cheios ou não (PARDINI et al., 2019). Esta atividade acaba resultando em desperdício de combustível, aumento na emissão de gases poluentes e prejuízo à mobilidade urbana.

Diferente do cenário de coleta tradicional, a Coleta Inteligente de Resíduos abrange o monitoramento de coletores e planejamento da rota de coleta (FEDCHENKOV et al., 2017). Por meio de Internet das Coisas ("*Internet of Things*" - IoT), diversos trabalhos foram conduzidos visando isso, como o de Cruz et al. (2021) que tem o uso das tecnologias LoRa (*Long Range*) e LoRaWAN (*LoRa Wide Area Network*) na coleta de resíduos para a *Smart City* de Lisboa. Anh Khoa et al. (2020) propôs um sistema inteligente utilizando LoRa e sensores ultrassônicos para otimizar a coleta de resíduos na Universidade *Ton Duc Thang* (Vietnã), resultando em economia de custo, mão de obra e tempo de serviço.

Diante disso, se torna viável uma abrangência na implementação de meios para monitoramento do processo de coleta dos resíduos através do uso de tecnologias que envolvem conceitos de Internet das Coisas. Através da interconexão de dispositivos inteligentes e a coleta de dados em tempo real, é possível obter informações sobre os coletores como os níveis de enchimento desses recipientes e outras variáveis relevantes.

Com isso, o presente trabalho tem como objetivo desenvolver um sistema capaz de monitorar remotamente o volume dos resíduos recicláveis presentes em uma lixeira e identificar sua respectiva localização, por meio de um sensor ultrassônico para a coleta dos dados, tecnologia LoRa e protocolo de comunicação MQTT para transmissão das informações coletadas à uma plataforma IoT denominada TagoIO para análise dos dados

a partir da criação de *Dashboards* referentes a estes, visando otimizar o processo de coleta seletiva urbano.

Dentre os objetivos específicos encontram-se o estabelecimento da comunicação confiável usando tecnologia LoRa e protocolo MQTT na transmissão dos dados, o desenvolvimento de uma interface intuitiva na plataforma TagoIO para fácil compreensão dos *Dashboards* e a realização de testes práticos em campo para validar a eficácia do sistema em diferentes cenários.

O trabalho proposto se justifica pois o monitoramento remoto dos coletores de resíduos possibilita a transmissão de informações em tempo real para as empresas e autoridades responsáveis pelo recolhimento diário do lixo. Usando esses dados pode-se ter a implementação de rotas eficientes visando áreas com necessidade urgente de limpeza prioritária, reduzindo o número de coletas perdidas em unidades de descarte que ainda possuem espaço para preenchimento, evitando transbordamento de lixeiras, economizando gastos com combustível, diminuindo emissão de gases poluentes pelos transportes coletores e otimizando o fluxo do tráfego de veículos melhorando a mobilidade urbana.

Além disso, esta pesquisa pode servir como incentivo ao governo e empresas na ampliação de políticas públicas com foco na gestão dos resíduos. Somado a isso, tratando-se de lixo destinado à coleta seletiva, ao passo em que é recolhido devidamente, coopera-se na reciclagem e sustentabilidade ambiental. Fora isso, a coleta de resíduos se enquadra no contexto de Cidades Inteligentes dado que este tema engloba desenvolvimento sustentável e econômico.

Por fim, o desenvolvimento deste projeto envolve a aplicação de diversos conhecimentos adquiridos nas disciplinas estudadas do curso de Engenharia Elétrica, podendo citar: Sistemas de Telecomunicações, Redes de Comunicação de Dados, Linguagem de Programação, Microcontroladores, Introdução às Ciências do Ambiente, Informática Industrial e Estágio Supervisionado. Além do incentivo à comunidade acadêmica em realizar pesquisas científicas no uso de Internet das Coisas para resolver desafios na sociedade.

1 REFERENCIAL TEÓRICO

Neste capítulo serão abordados os fundamentos e conceitos teóricos relacionados aos assuntos relacionados ao projeto. Inicialmente, na seção 1.1 é abordado sobre a coleta de lixo no Brasil voltado para o processo de coleta seletiva, em seguida, na seção 1.2 é tratado sobre o conceito de Internet das Coisas. Na seção 1.3 discorre-se o funcionamento dos sensores enfatizando o ultrassônico, já na seção 1.4 é apresentada a LPWAN, onde na subseção 1.4.1 é direcionado para os fundamentos da tecnologia específica LoRa. Na seção 1.5 é tratado sobre o protocolo de troca de mensagens MQTT com uma descrição do protocolo TCP/IP o qual atua em conjunto com o mesmo, na seção 1.6 é abordado sobre a plataforma de internet das coisas conhecida como TagoIO. Nas seções 1.7 e 1.8 é dada uma descrição geral do microcontrolador ESP32 seguido da apresentação do ambiente de desenvolvimento denominado Arduino IDE respectivamente.

1.1 COLETA SELETIVA DE LIXO

A coleta de lixo consiste no serviço de recolher os resíduos gerados, o qual pode ser prestado por uma empresa pública ou privada. Na maioria dos países, a coleta de lixo é feita pelo poder público, onde no Brasil, o processo é realizado pelas prefeituras ou empresas especializadas contratadas para essa finalidade, sendo sustentado por uma taxa retirada do IPTU (Imposto Predial e Territorial Urbano), cobrado de quem possui qualquer tipo de imóvel, ou seja, propriedade construída dentro de uma área urbana (CONTEMAR, 2020).

De acordo com a Lei de Política Nacional de Resíduos Sólidos (Lei nº 12.305/2010), são alguns dos objetivos: a proteção da saúde pública e da qualidade ambiental, não geração, redução, reutilização, reciclagem e tratamento dos resíduos sólidos, bem como disposição final ambientalmente adequada dos rejeitos e a adoção, desenvolvimento e aprimoramento de tecnologias limpas como forma de minimizar impactos ambientais.

Em Manaus, os serviços de Coleta e Transporte de Resíduos Sólidos Urbanos (RSU) são divididos em cinco modalidades. Inicialmente, a Coleta Domiciliar é responsável por recolher resíduos de domicílios, pequenas indústrias, comércio, bancos, escolas, e outros locais seguindo roteiros previamente definidos. Já a Remoção Mecânica trata dos resíduos que não podem ser recolhidos de forma manual e que não sejam domiciliares enquanto que a Remoção Manual recolhe resíduos depositados fora do horário de coleta regular e pontos de difícil acesso localizados na cidade. A Coleta Seletiva é caracterizada por cinco estratégias de execução: Coleta nos Pontos de Entrega Voluntária (PEVs), no Centro, Agendada, no Galpão da Logística Reversa e nas Associações e Cooperativas, e, a Coleta de Poda é executada após os serviços de poda e roçagem (SEMULSP, 2021).

A Coleta seletiva de lixo é um processo que consiste na separação e recolhimento dos

resíduos descartados. Desta forma, os materiais que podem ser usados para a reciclagem são separados do lixo orgânico (restos de carne, frutas, verduras e outros alimentos). Este último tipo de lixo é descartado em aterros sanitários ou usado para a fabricação de adubos orgânicos (PRIMA, 2023).

Reciclagem é o processo em que há a transformação do resíduo sólido que não seria aproveitado, com mudanças em seus estados físico, físico-químico ou biológico, de modo a atribuir características ao resíduo para que se torne novamente matéria-prima ou novos produtos (ECYCLE, 2022). A importância da coleta seletiva é justamente contribuir com a conservação do meio ambiente e o consumo sustentável, colaborando com redução dos impactos ambientais causados pelo exagero do consumo e o descarte incorreto de resíduos. A separação do lixo evita a contaminação dos materiais reaproveitáveis, aumentando o valor agregado destes e diminuindo os custos de reciclagem. A prática da coleta seletiva é um dos pilares da sustentabilidade (AMBIPAR, 2022).

Figura 1 – Coletores para coleta seletiva.



Fonte: Strutura Locações (2021).

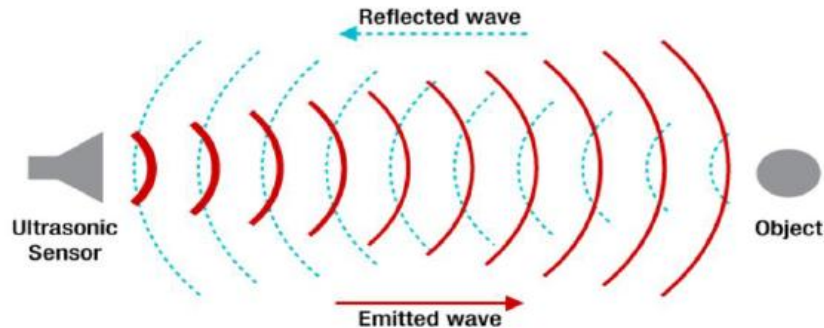
Para facilitar a coleta seletiva e a reciclagem, existe uma cor para representar cada material reciclado. Estas cores são geralmente usadas, junto com os símbolos de reciclagem, nos recipientes destinados à coleta dos materiais. As cores mais comuns são: verde para vidros, amarelo para metais, vermelho para plásticos e azul para papéis, conforme visto na figura 1.

1.2 INTERNET DAS COISAS

A Internet das Coisas (*"Internet of Things"* - IoT) permite objetos "verem", "ouvirem", "pensarem" e executarem trabalhos fazendo com que eles interajam entre si, para compartilhar informações e coordenar decisões. Esses objetos são transformados de tradicionais para inteligentes tendo como base tecnologias de comunicação, dispositivos embarcados, redes de sensores e protocolos de Internet (AL-FUQAHA et al., 2015). Na figura 2, essa ideia descrita é apresentada de forma ilustrativa.

entre o envio e recebimento do eco. Esse intervalo é computado por um controlador para determinar a distância do objeto.

Figura 3 – Diagrama de operação do sensor ultrassônico.

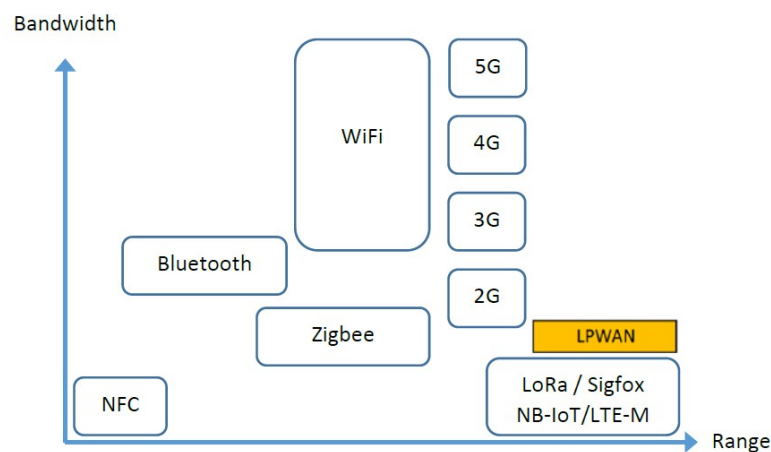


Fonte: Ahmad (2020).

1.4 LPWAN

LPWAN - *Low Power wide area network* ("Rede de área ampla de baixa potência") consiste numa tecnologia de rede *Wireless* ("Sem fio") que interconecta largura de banda baixa e dispositivos alimentados por bateria com baixas taxas de transmissão sobre longas distâncias. Criada para tecnologias IoT e *Machine-to-Machine - M2M* ("Máquina a Máquina"), isto é, com o uso de dispositivos capazes de se comunicarem entre si sem uma intervenção humana, opera em baixo custo com uma eficiência energética melhor do que as redes móveis tradicionais. Além disso, suportam um grande número de dispositivos conectados sobre uma área extensa, LPWAN não é uma única tecnologia mas um conjunto de várias tecnologias de rede de área ampla e baixa potência, as quais assumem diversas formas. Podem usar frequências licenciadas ou não licenciadas e incluem opções de uso de padrão aberto ou proprietárias (SHEA, 2017).

Figura 4 – Comparação de LPWAN com outras redes usadas em IoT.



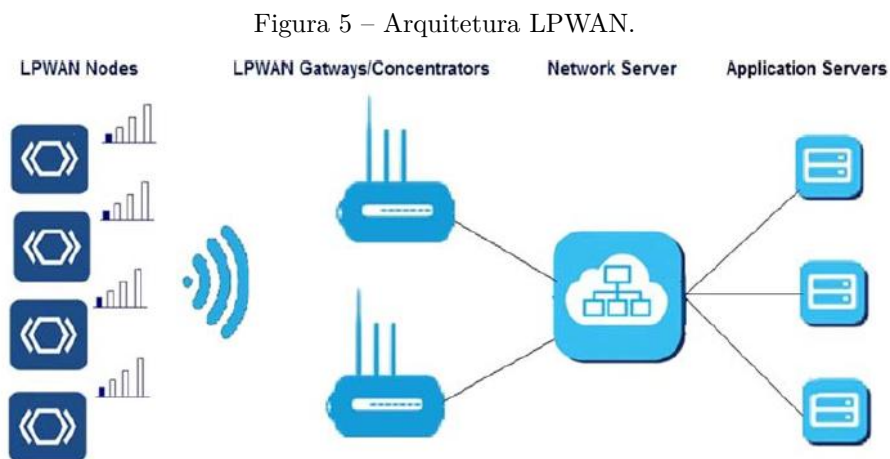
Fonte: Montagny (2022).

Na figura 4 é feito um comparativo relacionando a largura de banda (*"bandwidth"*) requerida, com a capacidade de alcance (*"range"*) de diversas tecnologias de comunicação. Tratando-se das LPWANs, observa-se a LoRa primeiramente desenvolvida pela empresa Cycleo em 2009 na Grenoble (França) e posteriormente comprada pela *Semtech* (USA) sendo em 2015 padronizada pela *LoRa-Alliance* e empregada em 42 países. A seguir se encontra a NB-IoT, baseada em tecnologia de rádio *narrowband* (*"banda estreita"*) e padronizada pela *3rd generation partnership project - 3GPP* (*"Projeto de Parceira da 3ª Geração"*), com especificações dadas em 2016 (CHAXEL et al., 2019).

Analisando ainda a figura 4 é apresentada a *Sigfox*, o trabalho de Chaxel et al. (2019) descreve que essa tecnologia foi desenvolvida em 2010 pela empresa de mesmo nome em *Toulouse* (França), a qual oferece uma solução de conectividade IoT *end-to-end* (*"ponta a ponta"*), isto é, a capacidade de conectar dispositivos de Internet das Coisas desde a coleta de dados até a entrega ao cliente, baseada nas suas patentes.

Por fim, na mesma figura em LPWAN verifica-se a seguinte rede denominada como *Long-Term Evolution for Machines - LTE-M* (*"Evolução de Longo Prazo para Máquinas"*). É um padrão de comunicação celular baseado na tecnologia LTE usado para comunicação de dados móveis de alta velocidade e opera em espectro licenciado, além de participar do padrão 3GPP.

A arquitetura básica LPWAN necessita de acesso sem fio e conectividade com a Internet e nuvem, uma tecnologia LPWAN específica requer um conjunto de entidades arquitetônicas (CHAUDHARI; ZENNARO, 2020). Na figura 5 é apresentado a estrutura com essas entidades de uma típica rede LPWAN.



Fonte: Zennaro et al. (2020).

Analisando a figura 5, os LPWAN *nodes* (*"nós"*) são dispositivos que coletam dados e respondem às entradas da rede LPWAN sendo geralmente sensores. Os dados coletados são enviados através de um *link* de rádio específico, ou seja, por meio de uma conexão sem fio até uma estação de acesso em uma rede IoT. Essa, fornece um *link* de rádio para

gerenciamento de dispositivos bem como a troca de tráfego, e interfaceia com o LPWAN *Gateways/Concentrators* ("Portão de entrada/Concentradores") também chamados de *core* ("núcleo") em alguns casos (CHAUDHARI et al., 2020).

De acordo com Zennaro et al. (2020), o *Gateway* é responsável por lidar com o controle e o tráfego do plano do usuário. Fornece um caminho para troca de informações entre a estação de acesso e a rede IoT e a translação entre os protocolos suportados pela estação de acesso de um lado e pela rede do outro. Já o *Network Server* ("Servidor de Rede"), realiza o provisionamento, registro e operações para as entidades LPWAN. Podendo compartilhar ou aumentar funcionalidades tais como: roteamento de tráfego, segurança e tratamento prioritário com o Núcleo. Posteriormente, o *Application Servers* ("Servidores de Aplicação") juntamente com a nuvem dão o suporte à rede LPWAN em administrar, o gerenciamento do *database* ("banco de dados") que contém as mensagens recebidas de todos os dispositivos conectados.

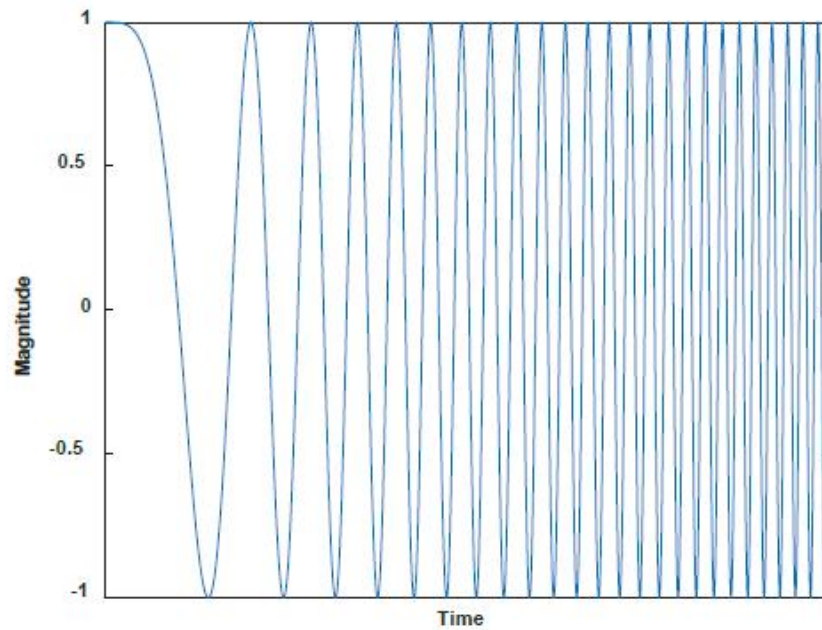
A partir desse processo uma análise dos dados pode ser realizada pelo respectivo cliente da rede LPWAN, e com isso uma ação, tarefa ou decisão é executada. Geralmente essas atividades são voltadas para os campos de *Smart Grids* ("Redes Inteligentes"), Energia, Segurança, Automotivo e logística, Automação Industrial, *Smart Agricultures* ("Agricultura Inteligente"), *Smart Homes* ("Casas Inteligentes"), entre outros.

1.4.1 LORA

LoRa - *Long Range* ("Longo Alcance"), é uma tecnologia de radiofrequência onde sinais eletromagnéticos são utilizados nas faixas de frequências de rádio para comunicação. É composta por uma camada ("*layer*") física na qual se realiza a modulação *wireless* ("Sem fio") utilizada para criar um *link* ("conexão") de comunicação de longo alcance. Utiliza tecnologias com tal capacidade para isso, como por exemplo *gateways* ou estações base (estações de acesso), os quais conseguem cobrir cidades inteiras ou centenas de quilômetros quadrados. A eficiência do alcance depende do ambiente ou de obstruções encontradas em uma certa localização (ALLIANCE, 2015).

Sua modulação, isto é, a forma de transmitir um sinal de informação modificado por meio de um canal de comunicação, mantém características da modulação FSK - *Frequency Shift Keying* ("Chaveamento de Frequência"). Basea-se em *Spread Spectrum* ("Espectro de Espalhamento"), no qual os dispositivos transmitem ao mesmo tempo e no mesmo canal, a partir de uma estrutura de sinal específica chamada *Chirp*. Daí, a modulação *LoRa* ser chamada de modulação *Chirp Spread Spectrum* (CSS), o termo *Chirp* vem do fato dele ser usado em tecnologia de radar (MONTAGNY, 2022).

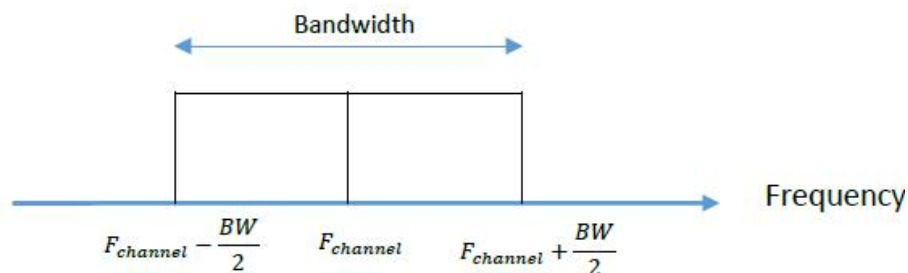
Figura 6 – Representação do Chirp.



Fonte: Montagny (2022).

Na figura 6 é apresentada a forma básica de um *Chirp* variado no tempo ("*Time*") em função da magnitude. Em termos de frequência, quando esse sinal varia sobre o espectro de forma crescente, ou seja, da frequência mínima até a máxima tem-se um *Up-Chirp* ("Chirp Ascendente"), quando o inverso ocorre e o sinal modulado varia sua frequência máxima para a mínima gradualmente ao longo do tempo, há o que é conhecido como *Down-Chirp* ("Chirp Descendente").

Figura 7 – Espectro de uma transmissão LoRa.



Fonte: Montagny (2022).

Para um entendimento melhor do espectro na transmissão a figura 7 que representa a modulação *LoRa* no domínio da frequência é analisada. De acordo com Montagny (2022), a frequência inicial é a frequência do canal ($F_{channel}$) menos a largura de banda dividida por dois ($BW/2$), enquanto que a frequência final é a frequência do canal mais a largura de banda dividida por dois. Onde:

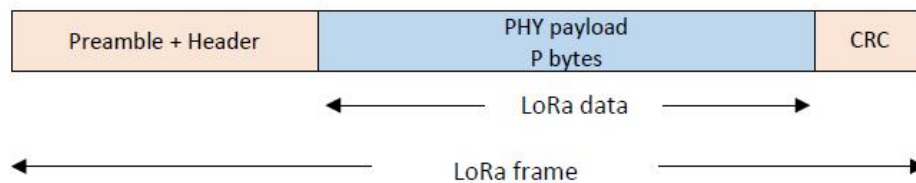
- ($F_{channel}$) é a frequência central.
- A banda ocupada ao redor de ($F_{channel}$) é a largura de banda ("*Bandwidth*").

Cada símbolo representa um número de *bits - binary digity* ("dígito binário") transmitidos. Montagny (2022) apresenta a seguinte regra na qual o número de bits transmitidos em um símbolo é igual ao *Spreading Factor - SF* ("Fator de Espalhamento"), o qual nada mais é do que um parâmetro que indica quantas vezes um símbolo é espalhado ao longo do espectro de frequência. Na emissão, os *bits* são agrupados em pacotes de *SF bits*, onde cada pacote é representado de possíveis formas dadas por 2^{SF} chips .

Para exemplificar, considerando um símbolo representado por 1 0 0 1 0 0 0 1 1 0, este possui 10 *bits* em seu pacote que podem ser codificados de 2^{SF} combinações binárias possíveis, logo, podem ser representados por $2^{10} = 1024$ chips.

Uma importante parte a se mencionar é o *Frame* LoRa. *Frame* vem do Inglês que significa algo como ("estrutura") no contexto dessa tecnologia, representa a unidade de transmissão de dados em LoRa contendo informações importantes a serem enviadas do dispositivo transmissor para o receptor. A figura 8 apresenta sua estrutura básica:

Figura 8 – Frame LoRa.



Fonte: Montagny (2022).

Examinando a figura 8, é feito um detalhamento dessa estrutura. Primeiramente há o *preamble* ("preâmbulo") que permite a sincronização do receptor com o transmissor seguido do *Header* ("Cabeçalho"), o qual contém informações relevantes como o endereço do dispositivo de destino e o tipo de mensagem a ser enviada.

O *PHY payload*, *physical layer - PHY* ("camada física") e *payload* ("carga útil"), seria o conteúdo útil que se deseja transmitir. Em outras palavras, são os dados LoRa que podem ser as informações de sensoriamento, como temperatura, umidade ou volume, bem como dados de localização como coordenadas GPS. *Padding bytes* mostrado como *P bytes* na figura, seriam os ("*bits* de preenchimento") que ocupam os espaços vazios do *Frame* para que as mensagens possam ser transmitidas e interpretadas corretamente. E finalmente, o campo *CRC - Cyclic Redundancy Check* ("Verificação de Redundância Cíclica") é responsável por verificar a integridade do *Frame* de maneira que os dados enviados pelo remetente, sejam idênticos aos dados recebidos pelo destinatário (MONTAGNY, 2022).

A partir disso, tem-se o conceito de Taxa de *bits* em LoRa que consiste na taxa máxima de transferência de dados que um dispositivo LoRa pode suportar. Definido matematicamente:

$$\text{Taxa de bits} = \text{SF} \cdot \frac{\text{Bandwidth}}{2^{\text{SF}}} \quad (1)$$

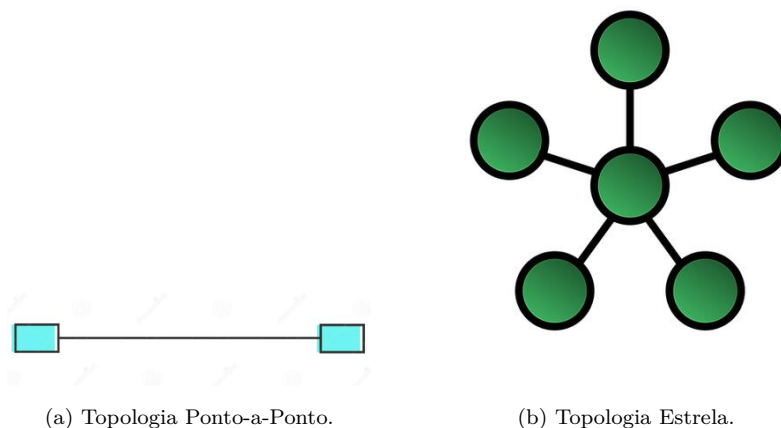
De acordo com a expressão acima, quanto maior o *Spreading Factor* (SF), menor será essa taxa enquanto que quanto maior a largura de banda ("*bandwidth*"), maior será a taxa de transmissão.

LoRa pode ser utilizada na forma de uma rede em um determinado tipo de topologia. A definição de topologia de rede, consiste na forma como os elementos são organizados em uma rede de comunicação. Nessa estrutura, os dispositivos de comunicação são modelados como nós e as conexões entre os dispositivos são modeladas como *links* ou linhas entre os nós. Ao mapeá-los graficamente, gera-se como resultado algumas formas geométricas que podem ser usadas para descrever diferentes topologias (INTERNATIONALIT, 2022).

No quesito topologia de rede, LoRa pode funcionar na topologia ponto-a-ponto em que dois dispositivos estão diretamente conectados entre si, formando uma comunicação direta. Além disso, os dados são transmitidos de um dispositivo para o outro sem a necessidade de compartilhá-los com recursos externos da rede.

Uma outra forma de configuração aplicável ao LoRa é a topologia estrela. Nesta, a rede é organizada de forma que os nós sejam conectados a um dispositivo central ou *gateway*, que gerencia a transmissão de dados até chegar em seu destino. Um exemplo do que foi descrito, seria o encaminhamento de informações de um ou mais dispositivos remotos ("*Senders*") até um dispositivo central ("*Receiver*"), responsável por enviar os dados para o destino final podendo ser um servidor na nuvem, banco de dados ou uma plataforma IoT para fins de análise, armazenamento ou outras ações. Essas duas topologias descritas podem ser observadas geometricamente na figura 9.

Figura 9 – Exemplos de Topologias de Rede.



Fonte: InternationalIT (2021).

Outra questão a se analisar é o fluxo de dados, que refere-se à maneira como os dados são transmitidos e recebidos. É uma característica importante a considerar ao projetar sistemas de comunicação e processamento de dados que envolvem a transferência e manipulação de informações. Quanto aos tipos de fluxos de dados permitidos, LoRa é bem versátil, permitindo tráfego de dados em três formas distintas (BERTOLETI, 2019):

- *Simplex*

Somente um dispositivo ("nó") transmite dados e os outros N dispositivos ("nós") da rede apenas recebem dados. É uma comunicação em via única entre o transmissor e o receptor.

- *Half-duplex*

Neste tipo de fluxo, todos os dispositivos da rede transmitem e recebem dados, porém nunca há uma transmissão e recepção simultâneas. Ou seja, num instante de tempo, um ("nó") pode estar transmitindo ou recebendo dados, nunca ambos.

- *Full duplex*

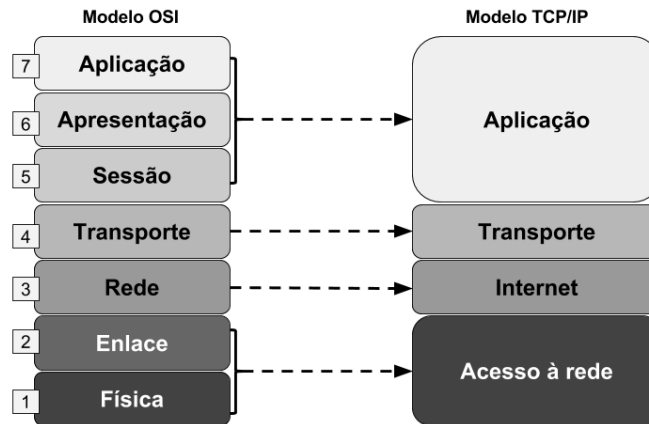
Todos os dispositivos da rede transmitem e recebem dados, podendo haver transmissão e recepção simultâneas.

1.5 MQTT

O protocolo MQTT - *Message Queuing Telemetry Transport* ("Fila de mensagens para transporte de telemetria"), é um protocolo de mensagens leve e simples que é frequentemente utilizado para conectar dispositivos de Internet das Coisas (KAZIENKO et al., 2019). É construído sobre o protocolo TCP/IP, este tem o seu nome originado por outros dois protocolos importantes: *Transmission Control Protocol* - TCP ("Protocolo de Controle de Transmissão") e *Internet Protocol* - IP ("Protocolo de Internet"), e seu principal objetivo consiste em construir uma interconexão de redes ("*Internetwork*"), a qual provê serviços de comunicação sobre redes físicas heterogêneas (PARZIALE et al., 2006; ROSSELOT et al., 2006).

O TCP/IP é modelado em camadas ("*layers*"). Essa representação leva ao termo pilha de protocolo, o qual se refere à pilha de camadas no conjunto do protocolo, podendo ser usada para posicionar o TCP/IP em relação a outros modelos de referência tal como o modelo OSI ("*Open System Interconnection*") que seria o modelo de Interconexão de Sistemas Abertos (PARZIALE et al., 2006). Essa comparação é vista na figura 10.

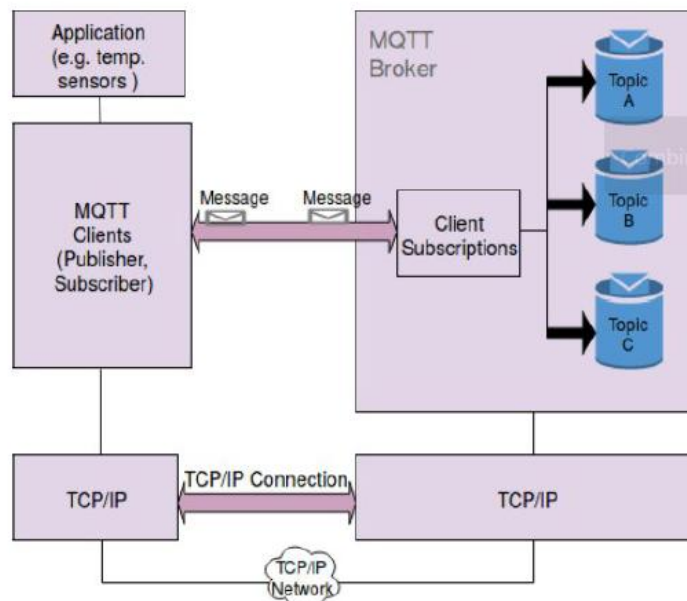
Figura 10 – Relação do modelo OSI com o TCP/IP.



Fonte: Kazienko et al. (2019).

O MQTT atua na camada de aplicação, esta é responsável por prover serviços ao usuário final tendo como principal importância a capacidade de fornecer serviços inteligentes com uma alta qualidade. Atua em cima do protocolo TCP utilizando sua camada de transporte para assegurar a entrega de dados de forma confiável pois, seus clientes e servidores devem suportar o uso de um ou mais protocolos de transporte que forneçam um fluxo de bytes ordenado e sem perdas (TUBINO et al., 2019). Na figura 11, é apresentada a arquitetura do MQTT com seus principais componentes e conceitos básicos que serão explicados detalhadamente.

Figura 11 – Arquitetura MQTT.



Fonte: Kazienko et al. (2019).

De acordo com Kazienko et al. (2019), o protocolo MQTT é baseado em um paradigma que permite a transmissão de mensagens para agrupamentos específicos de clientes, conhecido como *Publisher-Subscriber* ("Publicador-Assinante"). Na figura 11, verifica-se esse paradigma onde o *Publisher* publica mensagens em tópicos ("*Topics*") que são os canais de comunicação formando o assunto da mensagem. Daí, o *Subscriber* se inscreve nesses tópicos particulares para receber todas as mensagens publicadas neles. Cada um desses tópicos contém assinaturas do cliente ("*Client Subscriptions*") permitindo-o coletar os dados desejados (MAKWANA; SONI, 2017).

Esse processo é feito em um servidor central chamado *Broker* MQTT. Este é responsável pelo controle da distribuição da informação e principalmente, receber as mensagens do *Publisher*, filtrá-las, decidir quem tem interesse nelas e então enviá-las para todos os clientes inscritos (MAKWANA et al., 2017). Um *Broker* pode conter múltiplos tópicos, onde cada um deles permite a recepção de mensagens de diferentes dispositivos publicadores e as entrega aos múltiplos dispositivos assinantes daquele tópico (QUINCOZES et al., 2019).

Como exemplos de modelos *Publisher-Subscriber* do MQTT em um cenário de IoT, um sensor de umidade em um plantação que envia esses dados para o servidor seria o *Publisher* enquanto que um sistema de irrigação que recebe essas informações atuando para irrigar a plantação quando necessário seria o *Subscriber*. Em outro caso, um sensor de monitoramento de lixeira seria o *Publisher* que envia as informações para um servidor MQTT, enquanto que um sistema de gerenciamento de coleta de lixo, que as usa para otimizar as rotas dos caminhões de coleta, levando em consideração a quantidade de lixo em cada lixeira se classificaria como *Subscriber*.

1.6 TAGOIO

TagoIO é uma plataforma IoT completa que conecta sensores, empresas e pessoas. É capaz de armazenar e processar dados e disponibilizá-los para os usuários finais, através da conexão de um ou milhares de sensores, e da criação de *Dashboards* ("Painéis") e relatórios (TAGOIO, 2023).

Dessa forma, é possível construir e desenvolver uma solução IoT. O processo de criação da solução IoT acontece em três etapas. A primeira envolve a conexão de um dispositivo por Wi-Fi, LoRa, Sigfox, LTE, BLE, ZigBee, entre outras, que realize a leitura dos ambientes e envio dos dados para a plataforma. Depois, é necessário escolher um *template* ("modelo") para iniciar a construção da aplicação e integrar as soluções, combinando os dados com sistemas externos. Para finalizar o processo, é preciso criar e gerenciar os usuários e dispositivos, escalando a aplicação de forma rápida. Tais aplicações podem acontecer em diversos segmentos, como: automação industrial, irrigação inteligente, localização interna de depósitos, composição, refrigeração e telemática (KHOMP, 2019).

Figura 12 – TagoIO Logo.



Fonte: TagoIO (2023).

1.7 ESP32

O ESP32 é um microcontrolador (MCU) de baixo custo, com conectividade *Wifi* e *Bluetooth* integradas. Desenvolvido pela empresa chinesa *Espressif Systems* (Figura 12), é amplamente utilizado em projetos de Internet das Coisas, automação residencial, controle remoto, robótica, entre outros (ESPRESSIF, 2023). Embora seja tecnicamente apenas o chip, os módulos e as placas de desenvolvimento que possuem esse chip também costumam ser chamados de “ESP32” pelo fabricante, onde os módulos ESP32 são os módulos montáveis na superfície que contêm o chip. O benefício de adquiri-los é que podem ser facilmente montados em um MCU no processo de produção. Já o outro termo mencionado consiste nas placas de desenvolvimento IoT MCU que possuem os módulos contendo o chip ESP32 pré-instalado (HÜBSCHMANN, 2020).

O ESP32 pode ser programado usando a linguagem de programação C++ e isso é feito, por meio de várias ferramentas de desenvolvimento, incluindo a IDE do Arduino, o ESP-IDF (*Espressif IoT Development Framework*) e outras plataformas, tornando-se compatível com uma grande variedade de bibliotecas e módulos externos, o que facilita a criação de projetos IoT mais complexos.

Figura 13 – Espressif Systems Logo.



Fonte: Espressif (2023).

1.8 ARDUINO IDE

O Arduino IDE - *Integrated Development Environment* (“Ambiente de Desenvolvimento Integrado”), é uma plataforma utilizada para escrever, compilar e carregar códigos para as placas compatíveis com o Arduino. É uma ferramenta *open source* (“código aberto”), disponível para *Windows*, *Mac* e *Linux*. Seu objetivo é simplificar o processo de programação das placas Arduino, permitindo que desenvolvedores possam criar projetos e experimentos eletrônicos de forma fácil e rápida, suporta uma linguagem de programação baseada em C++, com algumas funções simplificadas e específicas para o ambiente Arduino. Oferece uma grande variedade de bibliotecas e exemplos de código prontos para

uso, que podem ser facilmente instalados e adaptados para atender às necessidades de diferentes projetos. Tem a facilidade de se conectar com diferentes placas e microcontroladores compatíveis, o que inclui desde as placas mais simples até as mais avançadas, como o ESP32, tornando esse ambiente uma opção popular para uma grande variedade de projetos de Internet das Coisas (ARDUINO, 2023).

Figura 14 – Arduino Logo.



Fonte: Arduino (2023).

2 METODOLOGIA

O trabalho proposto foi uma pesquisa aplicada, e teve como finalidade a realização da pesquisa exploratória sobre o material bibliográfico adquirido sobre o assunto. Foi utilizado o método de abordagem hipotético-dedutivo e o de procedimento monográfico em sua elaboração. Para a coleta de dados, foram utilizadas documentação indireta com auxílio de documentos primários e secundários, onde que a análise e a interpretação dos dados ocorreram de forma quantitativa.

A implementação do sistema de monitoramento foi feita através da montagem do coletor, no qual foram colocados um sensor ultrassônico e um microcontrolador ESP32 embutido com tecnologia LoRa sendo este conjunto o Transmissor, responsável por enviar as informações referentes aos resíduos via LoRa para um outro ESP32 atuando como o Receptor. Após este ter recebido os dados, os mesmos foram enviados através do protocolo MQTT para a plataforma TagoIO para análise dos dados em *dashboards*. Na figura 15, é apresentada a arquitetura do sistema desenvolvido.

Figura 15 – Arquitetura Geral do Sistema.



Fonte: Própria.

A razão pela qual o tipo de sensor que foi escolhido tenha sido o ultrassônico, é devido a sua capacidade de realizar leituras de medições de distância em relação aos objetos com uma boa precisão. Isso contribuiu para a implementação do projeto, pois como os objetos foram os resíduos recicláveis (papel, vidro, plástico e metal), ao serem colocados no interior do coletor, as distâncias entre os resíduos e o topo do coletor foram lidas pelo sensor. Então, esses mesmos valores foram convertidos em volume gerando a informação do *status* do coletor, ou seja, o seu nível de enchimento.

O ESP32 foi utilizado pela facilidade de integração de sensores a um sistema IoT, flexibilidade de programação, além do que algumas de suas variantes vêm com LoRa integrado. Por fim, a TagoIO foi usada como a plataforma IoT devido a conectividade acessível, suportando uma variedade de protocolos de comunicação, incluindo MQTT, permitindo conectar uma ampla gama de dispositivos.

3 IMPLEMENTAÇÃO

Neste tópico são descritas as etapas utilizadas para o desenvolvimento do sistema proposto.

3.1 DESENVOLVIMENTO DO TRANSMISSOR

Nesta parte são apresentados os passos para a configuração da lixeira, desde os modelos dos componentes utilizados no projeto, até a montagem do coletor transmissor.

3.1.1 SENSOR HCSR-04

O sensor ultrassônico HCSR-04 é um dispositivo que possui um circuito pronto com emissor e receptor acoplados e 4 pinos: *Vcc* para alimentação, *Trigger* para o disparo do pulso, *Echo* para o eco do pulso e o *Gnd* atuando como o terra. Este módulo de sensor foi escolhido para o projeto, na figura 16 e tabela 1, são apresentados este componente bem como os parâmetros importantes de funcionamento.

Figura 16 – Sensor de distância HCSR-04.



Fonte: Eletrogate (2022).

Tabela 1 – Principais parâmetros do Sensor HCSR-04

Tensão de operação	5 V (DC)
Corrente de operação	15 mA
Frequência de operação	40 kHz
Alcance de medição (mín ~ máx)	2 cm ~ 4 m

Fonte: Própria.

3.1.2 ESP32 HELTEC WIFI LORA (V2)

O ESP32 Heltec WiFi LoRa (V2) é um módulo de desenvolvimento avançado que combina as capacidades do microcontrolador ESP32 com a tecnologia LoRa e conectividade *WiFi*. Equipado com um transceptor LoRa SX1276, este módulo oferece comunicação de longo alcance, conectividade local e acesso à internet. Além disso, possui um *display* OLED integrado e uma antena LoRa otimizada para melhor desempenho. Com essa ampla gama de funcionalidades, este *Hardware* foi selecionado para a configuração do transmissor possibilitando o monitoramento remoto, conforme visto na figura 17.

Figura 17 – Módulo ESP32 Heltec Wifi LoRa (V2).



Fonte: Heltec (2018).

Na tabela 2, é detalhado algumas das especificações deste componente.

Tabela 2 – Especificações técnicas do Heltec Wifi LoRa 32 (V2)

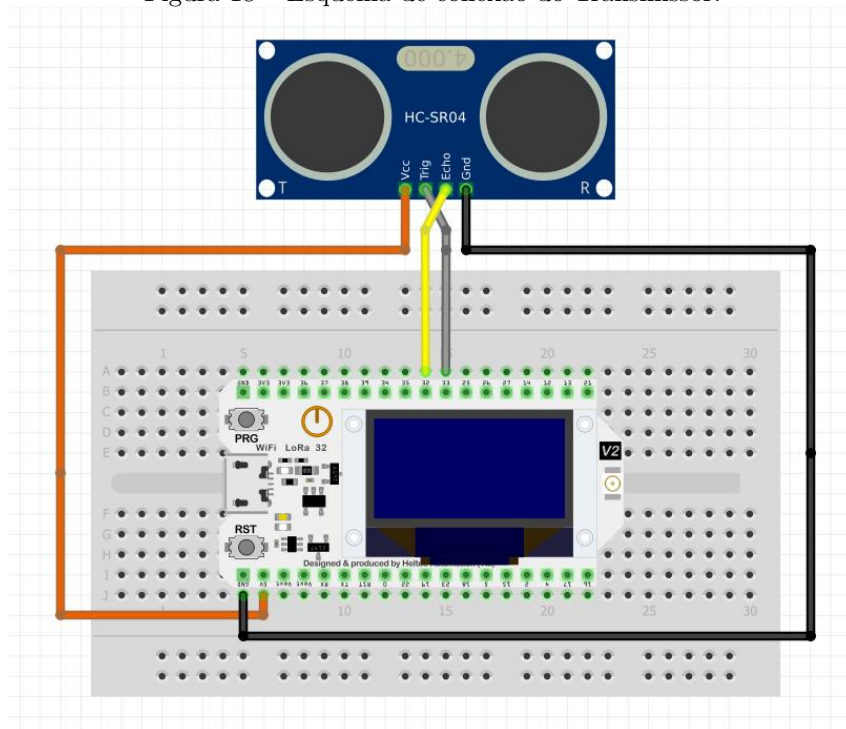
Características	Funcionalidades
Microprocessador	ESP32 (MCU de 32 bits <i>dual-core</i>) e o <i>chip</i> LoRa SX1276/SX1278
Interface Micro USB	Possui regulador de tensão e proteção contra curto-circuito
<i>WiFi</i> a bordo, antena de metal 3D e interface IPEX (U.FL)	Conectividade a rede <i>WiFi</i> , <i>Bluetooth</i> e ao uso LoRa, respectivamente
<i>Display</i> OLED matriz de pontos 128*64 de 0.96 polegadas a bordo	Exibe informações de depuração, energia da bateria e outras informações
<i>Chip</i> USB CP2102 integrado	Programação somado a depuração de dados

Fonte: Própria.

3.1.3 CONEXÃO DO SENSOR AO ESP32

Para que fosse possível realizar a leitura dos dados feita pelo sensor HCSR-04 foi necessária a configuração dos pinos *Vcc*, *Trigger*, *Echo* e o *Gnd* aos pinos do ESP32. Tendo como referência o diagrama de pinagem deste módulo fornecido pela *Heltec Automation*, tornou-se viável a conexão dos pinos entre os dois componentes, a qual é apresentada na figura 18.

Figura 18 – Esquema de conexão do Transmissor.



Fonte: Própria.

Analisando a figura 18, nota-se um esquemático desenvolvido no *Fritzing*, uma plataforma de software *Open Source* que pode ser usada para criação de protótipos virtuais de projetos eletrônicos. Por meio de cabos ("jumpers") e uma placa *protoboard* de 400 pinos a seguinte ligação foi feita:

- O pino *Vcc* do HCSR-04 foi conectado ao pino *Vcc* com entrada 5V do Heltec ESP32;
- O pino *Trig* (*Trigger*) do HCSR-04 foi conectado ao pino de numeração 33 do Heltec ESP32;
- O pino *Echo* do HCSR-04 foi conectado ao pino de numeração 32 do Heltec ESP32;
- O pino *Gnd* (*Ground*) do HCSR-04 foi conectado ao pino *Gnd* do Heltec ESP32;

3.1.4 DESENVOLVIMENTO DO SOFTWARE DO TRANSMISSOR

O algoritmo implementado no módulo Heltec ESP32 LoRa para monitoramento do sensor foi construído no ambiente do Arduino IDE, feito nas linguagens de programação C/C++ e baseado no repositório <http://bit.do/eX7er> do *GitHub*. Utilizaram-se as seguintes bibliotecas para auxílio conforme a Tabela 3.

Tabela 3 – Bibliotecas utilizadas no Algoritmo implementado no Transmissor

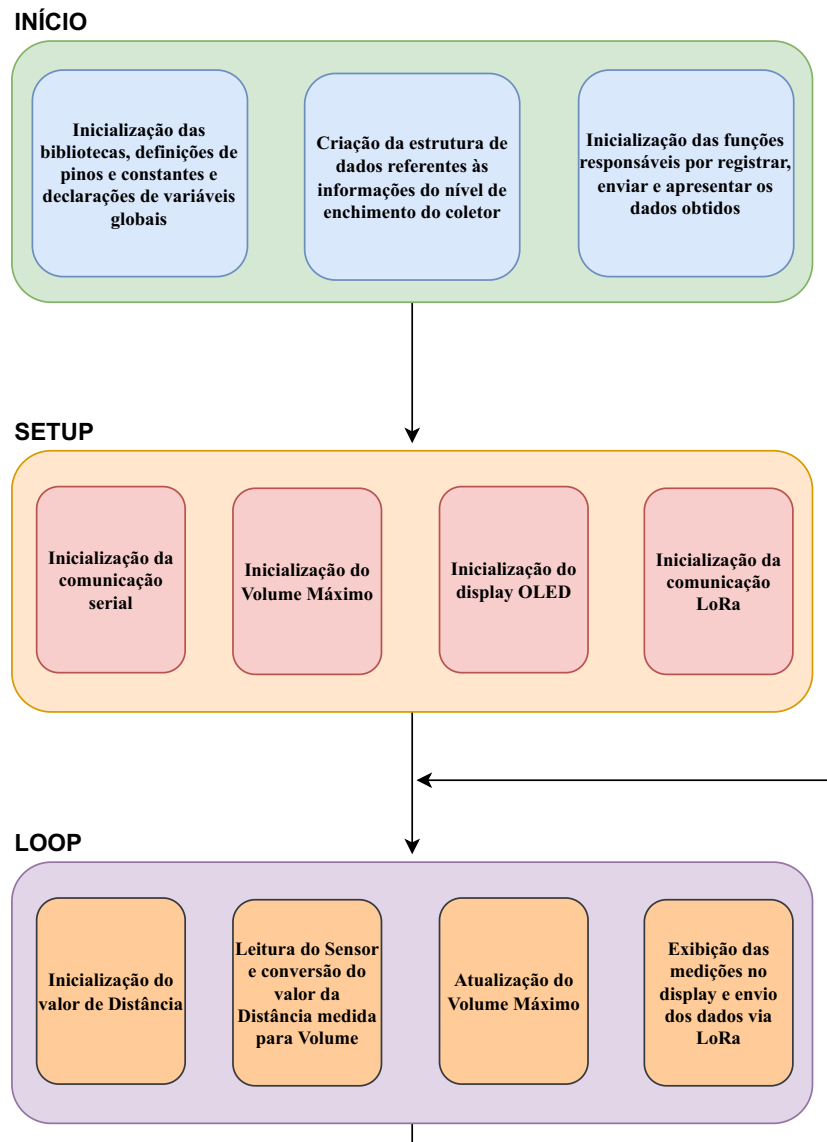
Biblioteca	Descrição
LoRa	Fornece suporte para a comunicação LoRa.
SPI	Permite a comunicação serial periférica entre os dispositivos.
Ultrasonic	Fornece suporte para configuração de sensores ultrassônicos.
<i>Wire</i>	Permite comunicação I2C entre dispositivos.
Adafruit_GFX	Fornece funções de texto e elementos visuais no <i>display</i> OLED.
Adafruit_SSD1306	Fornece suporte para <i>displays</i> OLED do controlador SSD1306.

Fonte: Própria.

Relacionando as bibliotecas da Tabela 3 com o transmissor, a biblioteca LoRa permitiu configurar e utilizar o módulo de comunicação LoRa para enviar dados, enquanto que a SPI - *Serial Peripheral Interface* ("Comunicação Serial Periférica") permitiu a comunicação entre o ESP32 e o sensor, a biblioteca *Ultrasonic* facilitou a utilização do sensor HCSR-04, oferecendo funções para a configuração e sua respectiva leitura.

A comunicação I2C - *Inter-Integrated Circuit* ("Circuito Inter-Integrado") que a biblioteca *Wire* permite, foi usada para estabelecer a comunicação entre o microcontrolador ESP32 e o *display* OLED através do protocolo I2C. Adafruit_GFX e Adafruit_SSD1306 com suas funções descritas, permitiram a visualização dos dados dos resíduos no *display* do módulo Heltec ESP32. Desta forma, fez-se a lógica do código a qual será apresentada e explicada com detalhes em um fluxograma visto a seguir.

Figura 19 – Fluxograma do Algoritmo do Transmissor.



Fonte: Própria.

Na figura 19 observa-se o funcionamento do algoritmo dividido em blocos interligados e sua análise é descrita abaixo:

1. INÍCIO:

Neste bloco, as bibliotecas listadas na Tabela 3 foram inicializadas e os pinos do sensor HCSR-04, do *display* OLED e do rádio para comunicação LoRa presente no módulo foram estabelecidos.

Definiram-se as constantes que se referem ao coletor de resíduos recicláveis. São estas: "distancia_lida" que informava a atual distância lida pelo sensor, "volume_lido" que mostrava o atual valor dessa distância convertido para volume em termos de porcentagem, "volume_max" que fornecia o máximo volume obtido durante um período de funcionamento do coletor e "profundidade_coletor" que consiste na variável que diz res-

peito a própria profundidade física que a lixeira possui em centímetros (cm).

Com as declarações das variáveis globais no bloco inicial, criaram-se os seguintes objetos: *ultrasonic* e *display* para representar no algoritmo o sensor ultrassônico e o *display* contido no módulo. Vale ressaltar uma importante definição de uma constante realizada para configurar a banda de frequência de operação do dispositivo LoRa especificando a faixa de frequências que iria operar. No caso, esse valor foi 915 referindo-se à faixa de frequência de 915 MHz.

Somado a isso, houve a criação de uma estrutura de dados a qual em programação, consiste em uma construção para criar um tipo de dado personalizado que pode conter diversos campos agrupando as informações relacionadas. No algoritmo isso foi feito usando a sintaxe *typedef struct*, usada para criar a estrutura chamada TDadosLora, que contém três campos:

- *distancia_lida* (Campo usado para armazenar a distância lida pelo sensor ultrassônico);
- *volume_lido* (Campo que armazena o volume lido pela lixeira);
- *volume_max* (Campo que armazena o valor máximo de volume registrado);

Com essa estrutura de dados criada facilitou-se o empacotamento e o transporte desses dados de forma organizada. Por fim, nesse processo foram programadas funções auxiliares que desempenharam uma tarefa única contribuindo para a inicialização, coleta, comunicação e exibição do sistema.

2. SETUP:

Esta etapa representa um bloco de código executado uma única vez quando o microcontrolador era inicializado. o que ocorria consistia no seguinte: Havia a inicialização da comunicação serial para estabelecer a comunicação serial entre o microcontrolador (ESP32) e um dispositivo externo (*notebook*). Junto com isso, a variável *volume_max* era inicializada com o valor do volume atual lido, isso servia como ponto de partida para registrar o volume máximo alcançado pelo coletor. Por conseguinte, iniciava-se o *display* OLED, caso isso fosse bem-sucedido, o *display* era configurado e preparado para mostrar informações e tinha-se a inicialização da comunicação com o módulo LoRa.

3. LOOP:

Na terceira fase do processo era onde a execução das tarefas ocorria em *loop* contínuo após a etapa anterior ter sido concluída. No contexto descrito, as seguintes ações eram realizadas: O sensor ultrassônico era acionado por cerca de 20 vezes em cada medição, e a distância média era calculada para aumentar a precisão da leitura. Essa distância ("distancia_lida"), nada mais era do que a distância entre a superfície formada pelo acúmulo dos resíduos no interior do lixeira e a tampa da lixeira.

Com a distância medida, o volume ocupado no coletor era calculado pela seguinte forma:

$$\text{volume_lido (\%)} = 100 - \left(100 \cdot \frac{\text{distancia_lida}}{\text{profundidade_coletor}} \right) \quad (2)$$

A partir da expressão acima afirma-se que com base na relação entre a distância lida e a profundidade física da lixeira, obtinha-se o volume do espaço vazio do coletor que era multiplicado por 100 para o valor ser transformado em termos de porcentagem. Considerando o espaço total do coletor de 100 % e subtraindo-o da porcentagem do espaço vazio obtinha-se o volume_lido, ou seja o volume atual de enchimento do coletor.

Sendo a profundidade do coletor medida em 18,45 cm, é realizada a exemplificação do que foi descrito anteriormente: Para uma distância lida de 4 cm, isso indica que o espaço vazio entre os resíduos e a tampa era de aproximadamente 22 %, fazendo a diferença de $100 \% - 22 \% = 78 \%$, resultando em um volume lido de 78 %.

Caso o volume obtido fosse maior do que o volume máximo registrado até então, o valor do volume máximo era atualizado. Após a realização desses procedimentos, essas informações eram exibidas no *display* OLED, fornecendo um *feedback* visual do estado atual. Por fim, os dados eram empacotados e enviados via LoRa através do módulo LoRa para transmissão a um receptor remoto.

3.1.5 MONTAGEM DO COLETOR

Com os componentes adquiridos e o algoritmo construído foi realizada a configuração do coletor. O módulo Heltec ESP32 LoRa foi fixado ao lado do coletor em um *protoboard* de 400 pontos com o auxílio de abraçadeiras e para alimentá-lo, foi utilizada uma fonte de bateria portátil de 5V, conforme visto na figura 20.

Figura 20 – Coletor Transmissor.



(a) Vista frontal



(b) Vista lateral



(c) Vista superior

Fonte: Própria.

Tratando-se ainda da figura acima, conectaram-se os pinos do sensor ao módulo por meio de *jumpers* e o sensor ultrassônico HCSR-04 foi anexado na tampa do coletor e tinha como referência para as medições a própria base do coletor. Para que o sensor pudesse ser fixado, foram feitas duas circunferências ao centro da tampa para encaixá-lo. Por fim, nomeou-se o coletor de "TCCLIXEIRA01".

Para representação dos resíduos recicláveis no projeto, utilizaram-se materiais para o papel, plástico, metal e vidro, os quais constituíram-se em folhas de papel, garrafas plásticas, latas de metal e copos de vidro respectivamente. Essa escolha de materiais permitiu uma representação visualmente identificável dos diferentes tipos de resíduos re-

cicláveis que pode ser vista na figura 21.

Figura 21 – Materiais utilizados para representação dos resíduos recicláveis.



(a) Folhas de papel



(b) Garrafas de refrigerante 200 ml



(c) Latas de refrigerante 269 ml



(d) Copos de vidro

Fonte: Própria.

De posse dos materiais, foi iniciada a preparação para os testes onde em cada execução, enchia-se a lixeira com um dos 4 tipos de materiais específicos citados anteriormente. Isso simbolizava as lixeiras destinadas para a coleta seletiva. Os estados de como o coletor se encontrava conforme realizava-se o procedimento é mostrado na figura 22.

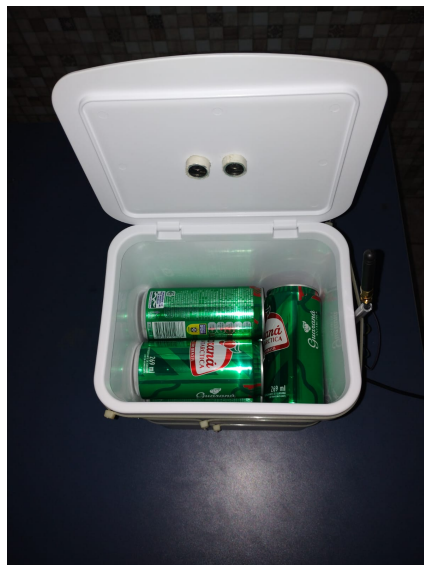
Figura 22 – Preparação do coletor para os testes.



(a) Coletor preenchido com papel.



(b) Coletor preenchido com plástico.



(c) Coletor preenchido com metal.



(d) Coletor preenchido com vidro.

Fonte: Própria.

Com o desenvolvimento do transmissor finalizado, a próxima etapa do projeto consistia na configuração do receptor do sistema IoT. Essa outra parte do processo que inclui desde a montagem da parte física e implementação do algoritmo será explicada com detalhes no tópico a seguir.

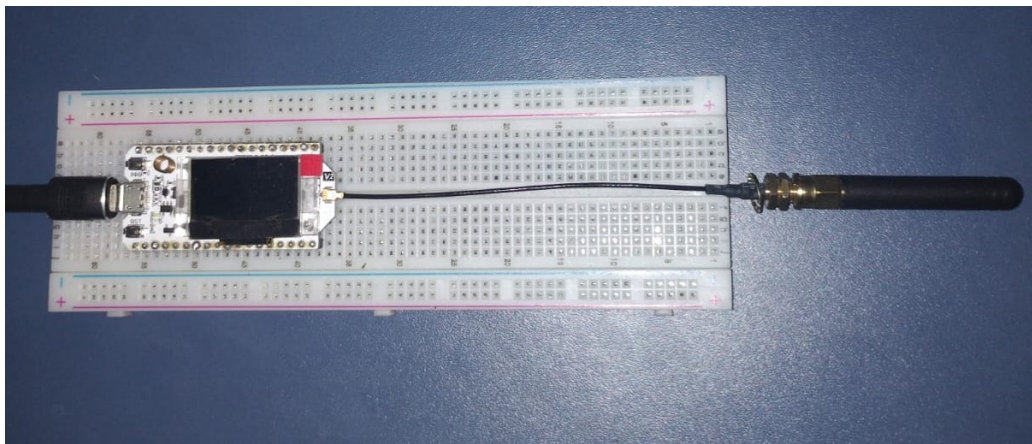
3.2 DESENVOLVIMENTO DO RECEPTOR

Nesta etapa uma análise foi realizada para determinar a escolha do *Hardware* que fosse capaz de receber e processar as informações enviadas pelo coletor. Considerando que o dispositivo precisaria ter comunicação LoRa para que fosse possível a recepção dos dados

do transmissor e capacidade de implementação do protocolo MQTT para a transferência dos dados coletados para a plataforma TagoIO, selecionou-se um outro módulo da Heltec ESP32 WiFi LoRa (V2) para atuar como o receptor do sistema.

Essa escolha tornou-se viável pois, conforme o tópico 3.1.2, com este componente possuindo um transceptor LoRa em sua placa, possibilitou a sua comunicação com o ESP32 presente no coletor e tendo tecnologia de conectividade *WiFi*, proporcionou o envio das mensagens recebidas via MQTT para a plataforma IoT. Para sua implementação, o módulo foi anexado em uma placa *protoboard* de 830 pinos, de acordo com a figura 23.

Figura 23 – Receptor.



Fonte: Própria.

Sua entrada micro-USB era energizada pela entrada USB do *notebook*. Além disso, foi construído o algoritmo de funcionamento do receptor o qual será descrito a seguir.

3.2.1 DESENVOLVIMENTO DO SOFTWARE DO RECEPTOR

O algoritmo implementado no módulo-receptor foi construído no Arduino IDE, programado nas linguagens C/C++ e baseado no repositório <http://bit.do/eX7er> do *GitHub*. Para isso, as seguintes bibliotecas foram utilizadas, conforme a Tabela 4.

Tabela 4 – Bibliotecas utilizadas no Algoritmo implementado no Receptor

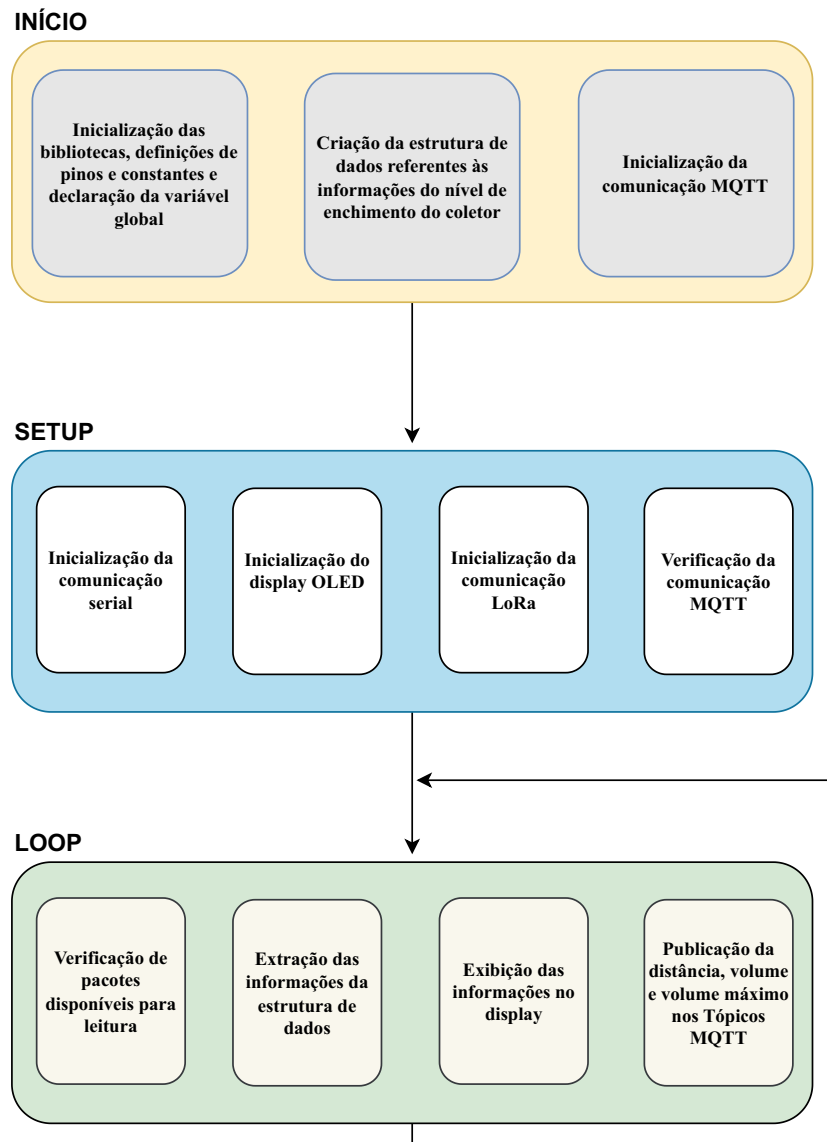
Biblioteca	Descrição
LoRa	Fornece suporte para a comunicação LoRa.
SPI	Permite a comunicação serial periférica entre dispositivos.
<i>Wire</i>	Permite comunicação I2C entre dispositivos.
Adafruit_GFX	Fornece funções de texto e elementos visuais no <i>display</i> OLED.
Adafruit_SSD1306	Fornece suporte para <i>displays</i> OLED do controlador SSD1306.
<i>WiFi</i>	Fornece suporte a redes <i>WiFi</i> , promovendo acesso a Internet.
EspMQTTClient	Simplifica a implementação do MQTT em dispositivos ESP32.

Fonte: Própria.

De forma similar a descrição do transmissor, é feita uma relação das bibliotecas da Tabela 4 com o receptor. A biblioteca LoRa permitiu configurar e utilizar o módulo de comunicação LoRa para receber dados, já a SPI configurou os pinos do ESP32-receptor para funcionar como linhas de comunicação. *Wire*, *Adafruit_GFX* e *Adafruit_SSD1306* possuíram utilidades da mesma maneira explicada no tópico 3.1.4.

Por fim, com a biblioteca *WiFi* através da configuração de parâmetros como o SSID - ("*Service Set Identifier*"), um nome único que identifica uma rede sem fio, e a senha, possibilitou-se o acesso a internet, para que por meio da *EspMQTTClient*, fosse facilitada a troca de mensagens MQTT entre o dispositivo ESP32-receptor e o servidor contido na TagoIO. Com o auxílio dessas bibliotecas fez-se a lógica do código do receptor conforme o fluxograma a seguir.

Figura 24 – Fluxograma do Algoritmo do Receptor.



Fonte: Própria.

A análise do algoritmo apresentado em blocos conectados na figura 24 é descrita abaixo.

1. INÍCIO:

Neste bloco, as bibliotecas listadas na Tabela 4 foram inicializadas e os pinos do *display* OLED e do rádio para comunicação LoRa foram estabelecidos. Com a declaração da variável global, criou-se o objeto do *display* para representar no algoritmo o *display* contido no módulo receptor. Novamente, foi realizada a definição da constante para configurar a banda de frequência de operação do dispositivo sendo o mesmo valor do transmissor, ou seja, 915 MHz.

Houve a implementação da estrutura de dados feita anteriormente chamada de TDadosLora, para que a compatibilidade e a interpretação dos dados vindos do trans-

missor pudessem ser realizadas corretamente no receptor. Além disso, foi inicializada a comunicação MQTT, criando-se o objeto `EspMQTTClient` caracterizado pelos seguintes parâmetros:

- `HOLYVERS 1`: Nome da rede *Wifi* que o ESP32 se conectou, sendo o identificador (SSID);
- `48484848483235313332313137`: Senha da rede *Wifi* correspondente ao SSID acima;
- `mqtt.tago.io`: Endereço do servidor *Broker* MQTT onde o dispositivo se conectou para o envio das mensagens;
- `esp32tcc`: Nome de usuário ("*Username*") usado para identificação do dispositivo por meio do *Token* na plataforma MQTT;
- `d6f637bf-d6ef-4704-b512-a63962a0d53b`: Código do *Token* ("Símbolo") usado para verificar a identidade do dispositivo correspondente ao usuário na TagoIO;
- `HVMO/TccEsp32`: Cliente que publicou e assinou os tópicos no *Broker* MQTT;
- `1883`: Número da porta padrão utilizada para a comunicação MQTT;

Com essa estrutura criada, estabeleceu-se a conexão MQTT entre o dispositivo e o servidor *Broker* na TagoIO, permitindo as interações de publicação e assinatura dos tópicos.

2. SETUP:

Havia a inicialização da comunicação serial para estabelecer a comunicação serial entre o microcontrolador (ESP32) e o *notebook*. Inicializavam-se o *display* OLED e o rádio LoRa, e a comunicação MQTT introduzida anteriormente era verificada. Isto era feito por meio de uma funcionalidade do protocolo MQTT chamada de "última vontade", que permite definir uma mensagem para ser enviada automaticamente para um tópico específico no servidor *Broker* no caso de uma desconexão não planejada.

3. LOOP:

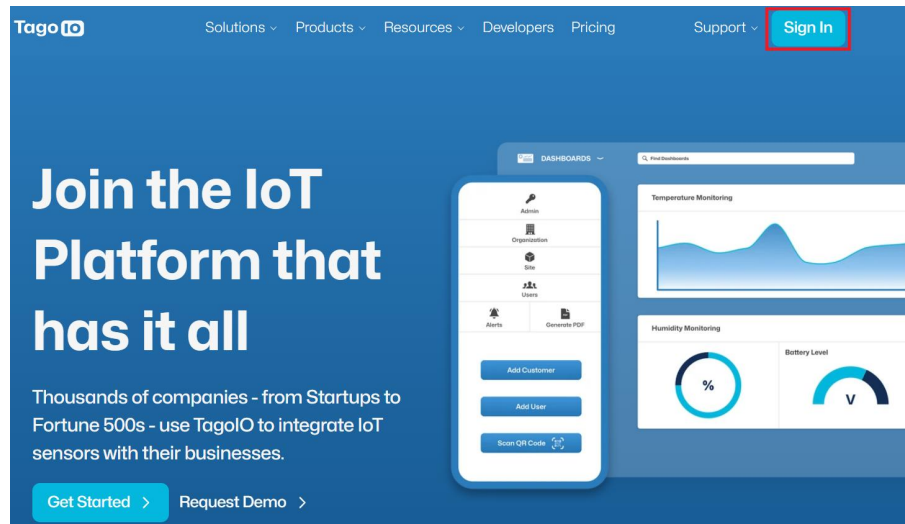
Nesta parte do processo, o algoritmo verificava se existiam pacotes disponíveis para leitura no rádio LoRa. Havendo um pacote, os dados recebidos eram extraídos da estrutura `TDadosLora` e as informações eram exibidas no *display* OLED do receptor. Com isso, os tópicos enviados para o *Broker* MQTT na TagoIO que continham a distância, volume e volume máximo, constituíam-se da seguinte maneira:

- `Dados/Mqtt/distancia_lida`: Tópico para a distância lida pelo sensor;
- `Dados/Mqtt/volume_lido`: Tópico para o volume em porcentagem da respectiva distância lida;
- `Dados/Mqtt/volume_maximo_registrado`: Tópico para o registro do máximo volume obtido durante determinado tempo de funcionamento;

3.3 CADASTRO DOS DISPOSITIVOS NA TAGOIO

Nesta subseção são apresentados os passos utilizados para o registro do transmissor e do receptor na plataforma TagoIO. Dessa forma, foi criada uma conta nesse site para visualização dos dados coletados no desenvolvimento do projeto. A figura 25 exibe a tela inicial do site com a opção de *Sign In* ("Entrar").

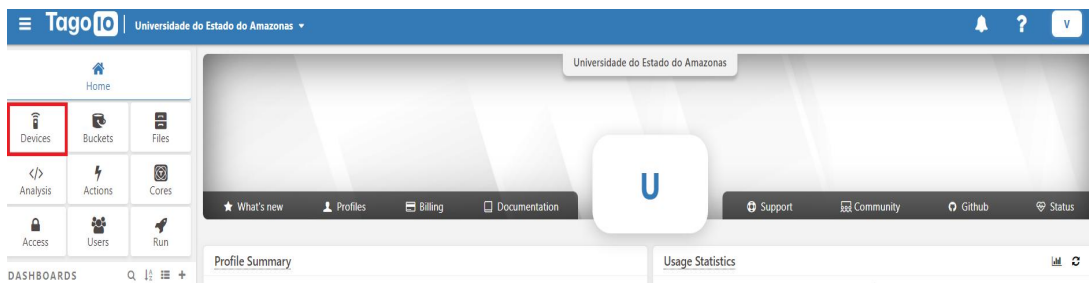
Figura 25 – Tela inicial da TagoIO.



Fonte: Própria.

Ao clicar em "Sign In", é solicitado o *Login* no primeiro acesso. Após a realização disto, é apresentada na tela diversas funções desde recursos para monitoramento de dispositivos conectados, visualização de dados em tempo real, criação de painéis personalizados, configuração de notificações e alertas, análise de dados históricos e integração com outras ferramentas e serviços, como é mostrado na figura 26.

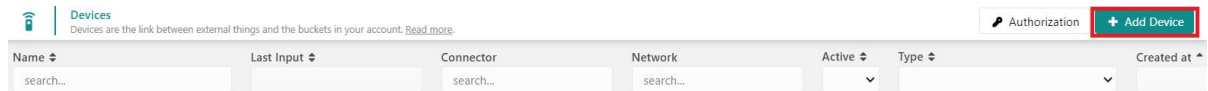
Figura 26 – Tela inicial após o *Sign In*.



Fonte: Própria.

Com relação a figura 26, ao clicar em "Devices", a seguinte tela é mostrada conforme a figura abaixo.

Figura 27 – Tela para adição de um dispositivo.



Fonte: Própria.

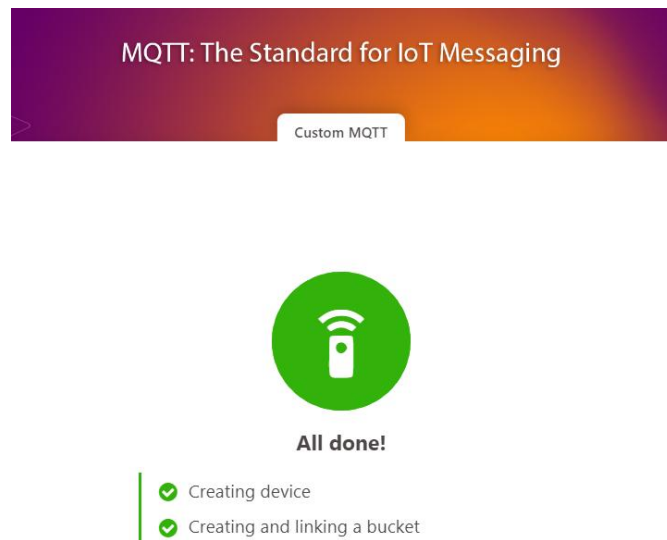
Ao selecionar a opção "Add Device" da figura 27, uma série de tipos de conexões são mostradas, escolheu-se a conexão MQTT já que os dados eram enviados para essa plataforma por meio desse protocolo onde ao selecioná-la foi exibida a imagem abaixo.

Figura 28 – Tela de inserção do nome do dispositivo na TagoIO.

Fonte: Própria.

Na figura 28 é mostrado o nome que foi dado ao dispositivo receptor na TagoIO em "Device name" como "esp32mqtt". Após esse procedimento, criou-se o dispositivo confirmado pela seguinte mensagem na tela apresentada na figura 29.

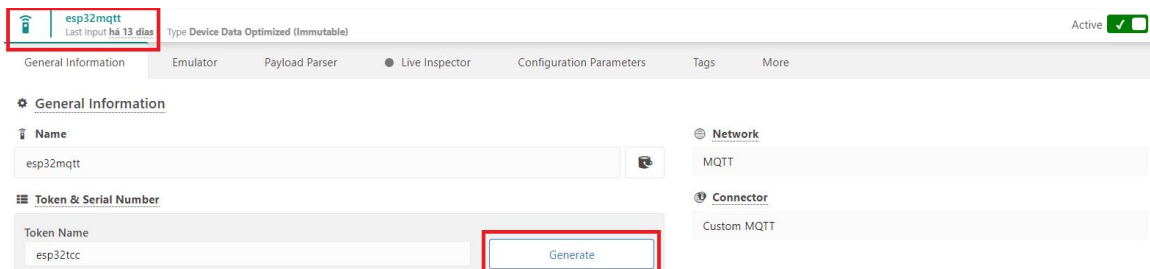
Figura 29 – Tela de confirmação da criação do dispositivo.



Fonte: Própria.

As informações gerais referentes ao receptor como a sua última sessão na plataforma (“*Last Input*”), nome do dispositivo, tipo de rede e conexão, e informações de *Token* são vistas na figura 30.

Figura 30 – Tela de visão geral do dispositivo.



Fonte: Própria.

Tratando-se ainda da figura acima, na opção “*Live Inspector*”, disponibilizavam-se em tempo real as informações dos tópicos MQTT. Clicando-se em “*Generate*” foi gerado o código do *Token* referente ao seu respectivo *Username* “esp32tcc” que é exibido na figura abaixo.

Figura 31 – Tela de informações do *Token*.

Fonte: Própria.

Conforme a figura 31, é apresentado o código do *Token* utilizado no algoritmo do receptor, sendo este a sequência "d6f637bf-d6ef-4704-b512-a63962a0d53b". Este código não possui tempo de expiração conforme verificado na imagem acima, além disso, pode ser fornecido por meio do *QRCode*.

Diante disso, a próxima etapa consistiu na construção dos *Dashboards* contendo as informações de distância, volume e volume máximo obtidas. Isso foi feito clicando na opção "Action" da tela inicial, e depois na opção "Add Action". Ao realizar isso, a seguinte tela era mostrada de acordo com a figura 32.

Figura 32 – Tela de visão geral *Action*.

Add Action

Name
recebe dados

Type of trigger

- Variable**
Triggered when the selected variables meet certain conditions.
- Resource**
Triggered when the selected resources change (devices, buckets, users, ...).
- Schedule**
Triggered based on the selected time interval.
- MQTT Topic**
Triggered based on MQTT topics.

Type of action – Learn more about this Action type
Insert to Device Bucket

Learn more ▾

Cancel **Create my Action**

Fonte: Própria.

Na figura 32, verifica-se as informações colocadas para a criação dos *Dashboards*. Primeiramente foi dado um nome para a ação chamada "recebe dados", além disso foi selecionada em "Type of trigger" a opção "MQTT Topic", e em "Type of action" a opção "Insert to Device Bucket". Por fim, clicou-se em "Create my Action".

Ao entrar na ação criada, exibiu-se uma tela na qual foram solicitadas inserções referentes aos tópicos enviados. Na figura 33 é mostrada a visualização dessa tela com destaque para certas informações que serão explicadas logo a seguir.

Figura 33 – Tela de visão geral dos Tópicos MQTT.

The screenshot shows the configuration page for an MQTT topic. At the top, the action is named "recebe dados" and is of type "MQTT Topic". The trigger is set to "Single device" (watching a single device) with the device name "esp32mqtt". Under the "Subscribe" section, three MQTT topics are defined: "Dados/Mqtt/distancia_lida", "Dados/Mqtt/volume_lido", and "Dados/Mqtt/volume_maximo_registrado". The action type is "Insert to Device Bucket". A "Save" button is located at the bottom right of the configuration area.

Fonte: Própria.

Na figura 33 é exibida o nome da ação "recebe dados", selecionou-se em *Trigger* a opção "Single Device" referindo-se ao receptor, bem como a sua respectiva nomenclatura colocada na TagoIO ("esp32mqtt"). Na parte inferior da tela, na opção *Subscribe* foram requeridos os nomes dos tópicos dos quais pretendiam-se saber os seus conteúdos. Conforme explicado anteriormente e visto na imagem acima, foram colocados os seguintes nomes dos tópicos MQTT: Dados/Mqtt/distancia_lida, Dados/Mqtt/volume_lido e o Dados/Mqtt/volume_maximo_registrado. Após isso, clicou-se em "Save". Diante disso, novamente na tela inicial foi selecionada a opção "Add Dashboards" na qual foi exibida a tela mostrada na figura 34.

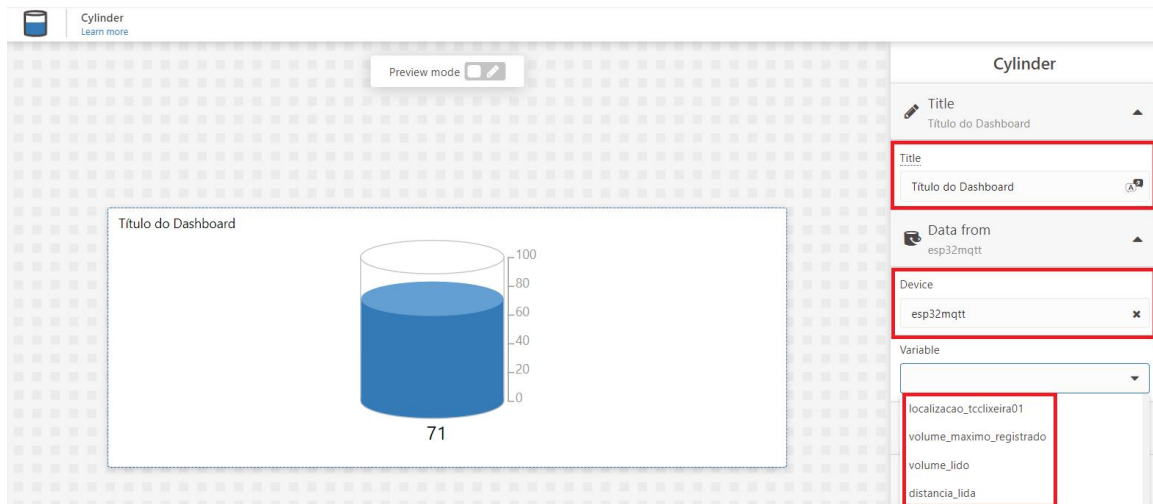
Figura 34 – Tela de visão geral para adição dos Dashboards.

The screenshot shows the "Add Dashboard" dialog. The "Name" field is filled with "Título do Dashboard". Under "Type of dashboard", the "Normal" option is selected. The "Normal" option description is "For quick start. You simply need to define the devices for each widget." The "Blueprint" option description is "Blueprint with live updates, it allows linking widgets to multiple devices dynamically. This is useful when scaling up applications." At the bottom, there are "Cancel" and "Create my Dashboard" buttons.

Fonte: Própria.

Analisando a figura 34 em *Name* foi colocado o título do *Dashboard* construído referente a um determinado resíduo reciclável (papel, plástico, metal e vidro), enquanto que em *Type of dashboard* selecionou-se a opção "Normal". Clicando em "Create my Dashboard" escolheram-se o *template* e as informações que iriam aparecer no *Dashboard*. A imagem a seguir, demonstra o ambiente para a criação dos *Dashboards* na TagoIO.

Figura 35 – Tela de visão geral para adição do *template* dos *Dashboards*.



Fonte: Própria.

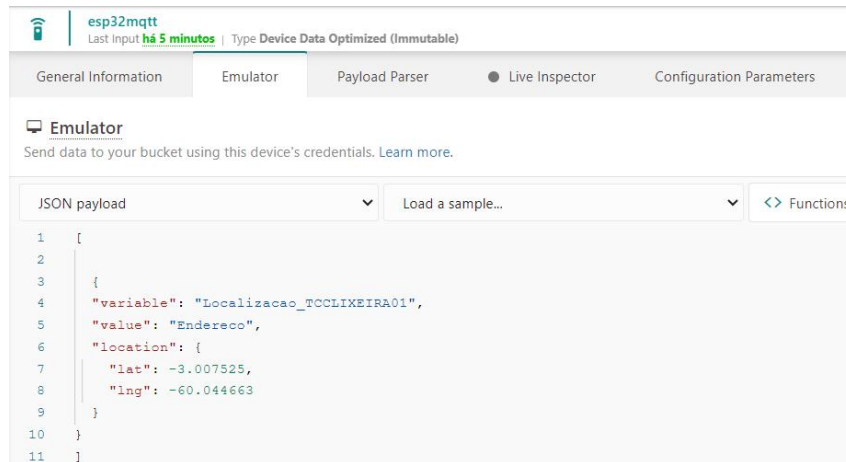
Na figura 35, no lado direito observa-se as informações inseridas para o desenvolvimento do *Dashboards*. Em "Title" há o título escolhido enquanto que em "Device", selecionava-se o dispositivo cadastrado "esp32mqtt" ao qual pretendia-se visualizar os dados. Na opção "Variable", selecionava-se os tópicos MQTT para serem mostrados no *template* o qual exibia os tópicos mais recentes enviados pelo dispositivo. Porém, caso se quisesse observar os tópicos mais antigos no *Dashboard* à medida que os testes iam sendo realizados, bastava inserir os valores destes tópicos dentro da própria configuração do *Dashboard* criado, com os valores dos tópicos antigos armazenados na plataforma na opção "Buckets" exibida na figura 26 ao lado da opção com destaque "Devices".

Além dos tópicos MQTT enviados pelo receptor, foi possível adicionar a localização da lixeira (coletor) nesses *Dashboards* já que a TagoIO possui integração com plataformas de mapeamento como o *MapBox* e *OpenStreetMap* (serviços de mapeamento online que oferecem uma variedade de recursos relacionados a mapas, navegação e localização). A maneira como isso foi realizado será apresentada mais adiante.

Para o cadastro da localização do coletor-transmissor na plataforma TagoIO, após o algoritmo do transmissor ser embarcado no ESP32 fixado na lixeira e com esse dispositivo ser alimentado pela bateria de 5 V, em cada caso de teste o coletor era colocado em diversos pontos cujas localizações eram adquiridas em latitude e longitude, por meio do aplicativo *Google Maps* (serviço de mapeamento semelhante ao *MapBox* e *OpenStreetMap*) em um

celular próximo a esse coletor. Ao acessar o *device* cadastrado na TagoIO, foi possível adicionar essas localizações associadas a cada teste específico. Isso foi feito na parte ”*emulator*” conforme a imagem abaixo.

Figura 36 – Tela de visão geral para adição da localização do coletor.



Fonte: Própria.

Na figura 36 em ”*variable*” tem-se a variável localização para o coletor com o ID ”TCCLIXEIRA01”, enquanto que no parâmetro ”*value*” o endereço de localização do coletor como a rua pode ser adicionado. Por fim, em *location* foi adicionado a latitude e longitude obtidas para cada ocasião de teste podendo serem vistas nos *dashboards*.

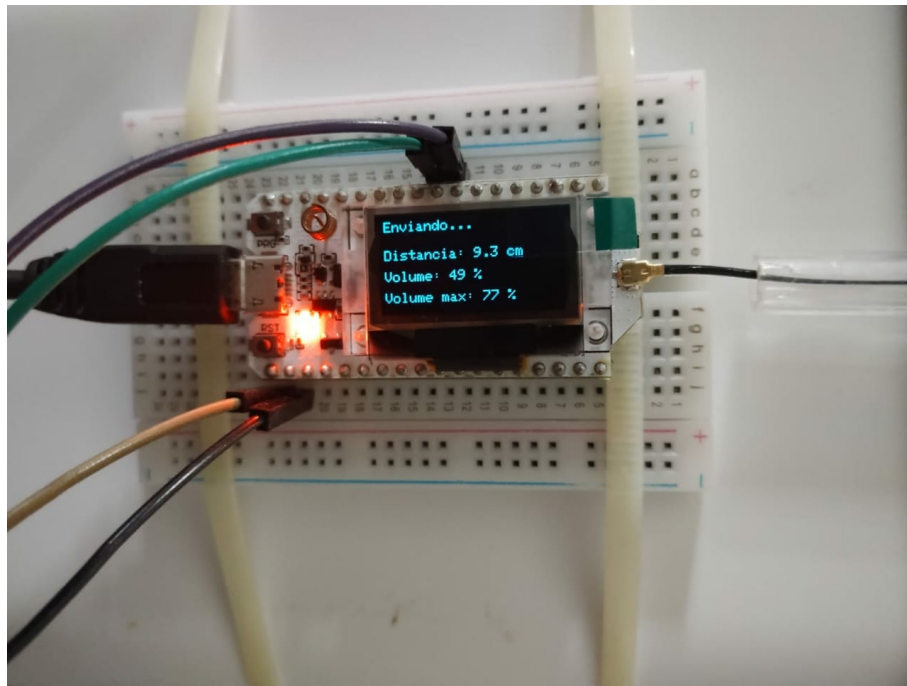
Diante de todos os fatos descritos, o procedimento para a execução dos testes de validação do sistema baseou-se no seguinte: Para cada conjunto de resíduos recicláveis dos tipos papel, plástico, metal e vidro, foram executadas 5 sequências de coleta dos dados em que cada uma, o coletor encontrava-se em um determinado local do bairro Tarumã localizado na cidade de Manaus (AM), próximo à residência na Rua Praia de Jericoaquara, número 186 onde encontrava-se o receptor, localizada no mesmo bairro.

Em cada caso de teste, o nível dos resíduos presentes no coletor eram variados com a finalidade de averiguar o seu comportamento. Os resultados serão apresentados em Tabelas e *Dashboards* na seção a seguir.

4 RESULTADOS E DISCUSSÕES

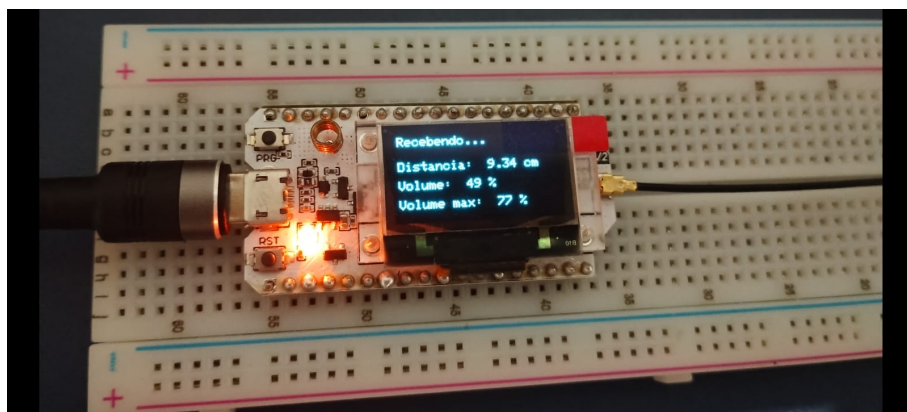
Com o sistema em funcionamento, conforme a realização dos testes, os dados coletados puderam ser exibidos nos *displays* do transmissor e do receptor, respectivamente. As figuras abaixo apresentam algumas dessas visualizações.

Figura 37 – Transmissor em funcionamento.



Fonte: Própria.

Figura 38 – Receptor em funcionamento.



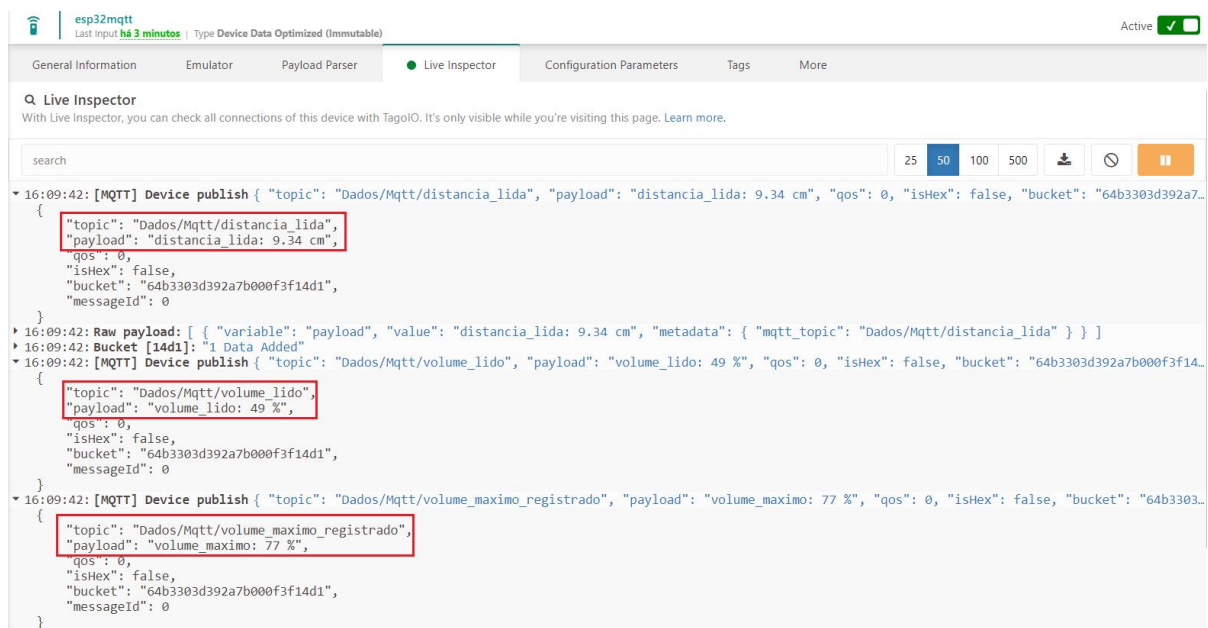
Fonte: Própria.

Na figura 37 é apresentado o transmissor em teste enviando as informações de "Distância" que se refere a distância atual entre os resíduos colocados no coletor e a sua superfície naquele momento sendo esse valor de 9,3 cm, "Volume" que consiste no valor dessa respectiva distância convertida no volume de 49 %, e na parte inferior do *display*

nota-se o máximo volume registrado em "Volume max" nesse período de funcionamento no valor de 77 % o qual obteve-se em uma medição feita anteriormente.

Na figura 38 é exibido o receptor recebendo as informações do transmissor via LoRa e mostrando-as no seu respectivo *display*. Verifica-se que as informações de ambos os dispositivos coincidem sendo o valor da "Distância" no receptor de 9,34 cm o qual arredondando-se vai para o valor de 9,3 cm (o mesmo do transmissor), o valor do "Volume" de 49 % e o do "Volume max" de 77 %. Além disso, essas informações puderam ser enviadas via protocolo MQTT para a TagoIO conforme mostra a figura 39.

Figura 39 – Exibição dos dados na TagoIO.



```

16:09:42: [MQTT] Device publish { "topic": "Dados/Mqtt/distancia_lida", "payload": "distancia_lida: 9.34 cm", "qos": 0, "isHex": false, "bucket": "64b3303d392a7b000f3f14d1", "messageId": 0 }
  {
    "topic": "Dados/Mqtt/distancia_lida",
    "payload": "distancia_lida: 9.34 cm",
    "qos": 0,
    "isHex": false,
    "bucket": "64b3303d392a7b000f3f14d1",
    "messageId": 0
  }
16:09:42: Raw payload: [ { "variable": "payload", "value": "distancia_lida: 9.34 cm", "metadata": { "mqtt_topic": "Dados/Mqtt/distancia_lida" } } ]
16:09:42: Bucket [14d1]: "1 Data Added"
16:09:42: [MQTT] Device publish { "topic": "Dados/Mqtt/volume_lido", "payload": "volume_lido: 49 %", "qos": 0, "isHex": false, "bucket": "64b3303d392a7b000f3f14d1", "messageId": 0 }
  {
    "topic": "Dados/Mqtt/volume_lido",
    "payload": "volume_lido: 49 %",
    "qos": 0,
    "isHex": false,
    "bucket": "64b3303d392a7b000f3f14d1",
    "messageId": 0
  }
16:09:42: [MQTT] Device publish { "topic": "Dados/Mqtt/volume_maximo_registrado", "payload": "volume_maximo: 77 %", "qos": 0, "isHex": false, "bucket": "64b3303d392a7b000f3f14d1", "messageId": 0 }
  {
    "topic": "Dados/Mqtt/volume_maximo_registrado",
    "payload": "volume_maximo: 77 %",
    "qos": 0,
    "isHex": false,
    "bucket": "64b3303d392a7b000f3f14d1",
    "messageId": 0
  }
  
```

Fonte: Própria.

Analisando a figura 39, observam-se os dados publicados nos conteúdos de "[MQTT] Device publish" onde em "topic" são visualizados os tópicos MQTT construídos anteriormente, enquanto que em "payload" as mensagens referentes aos tópicos MQTT são mostradas: "distancia_lida: 9.34 cm", "volume_lido: 49 %" e "volume_maximo: 77 %". A partir disso, confirma-se que os dados enviados pelo coletor puderam ser recebidos pelo receptor que os reenviou à plataforma eficazmente.

A seguir são apresentadas 4 tabelas, cada uma feita para um determinado resíduo reciclável testado com 5 sequências de medições realizadas. Para cada tabela foi criado um *Dashboard* referente a um caso de teste dessas sequências.

Tabela 5 – Dados obtidos para o material Papel

Casos	Distância(cm)	Volume(%)	Volume Máximo(%)	Latitude	Longitude
1º Teste	11.30	38	38	-3.007132	-60.047095
2º Teste	3.74	79	79	-3.007161	-60.047095
3º Teste	3.41	81	81	-3.007087	-60.047027
4º Teste	8.09	56	81	-3.007259	-60.046881
5º Teste	6.72	63	81	-3.007430	-60.046992

Fonte: Própria.

Para a sequência de testes mostrada na tabela 5, o preenchimento de papéis de forma manual se deu de forma crescente do 1º ao 3º caso, esse preenchimento correspondeu aos valores de volume obtidos de 38 %, 79 % e 81 % respectivamente.

A partir do 4º teste optou-se em reduzir o nível dos resíduos no interior da lixeira, porém o nível medido não acompanhou o decaimento sendo que no 5º teste havia menos pedaços de papel do que o 4º mas mesmo assim o volume do 5º teste mostrou-se relativamente superior ao do 4º teste indicando uma certa imprecisão na leitura do sensor. O maior volume registrado foi de 81 % e esse valor continuou sendo registrado corretamente até o final da sequência. Os valores de Latitude e Longitude corresponderam às localizações aproximadas em que lixeira estava posicionada nos mapas do *Dashboard*.

Tabela 6 – Dados obtidos para o material Plástico

Casos	Distância(cm)	Volume(%)	Volume Máximo(%)	Latitude	Longitude
1º Teste	3.30	81	81	-3.006940	-60.047011
2º Teste	4.12	77	81	-3.006959	-60.045892
3º Teste	6.26	66	81	-3.007525	-60.044663
4º Teste	11.21	39	81	-3.007989	-60.047009
5º Teste	14.54	21	81	-3.006026	-60.045801

Fonte: Própria.

Na tabela 6 consta os testes para as garrafas de plástico, onde a execução dos casos realizadas de forma decrescente foram correspondidas nos valores das medições. No 1º teste em que o coletor estava com mais garrafas, o maior volume foi exibido de 81 % e a medida que se ia esvaziando-o, seu respectivo volume decaiu-se até alcançar o valor mínimo de 21 %. O Volume Máximo foi registrado adequadamente do começo ao fim da sequência e os valores de Latitude e Longitude inseridos na TagoIO, mostraram as localizações aproximadas da lixeira.

Tabela 7 – Dados obtidos para o material Vidro

Casos	Distância(cm)	Volume(%)	Volume Máximo(%)	Latitude	Longitude
1º Teste	8.99	51	51	-3.007169	-60.046986
2º Teste	6.14	66	66	-3.007032	-60.046772
3º Teste	5.73	68	68	-3.007047	-60.046977
4º Teste	8.68	52	68	-3.006962	-60.046747
5º Teste	12.44	32	68	-3.007030	-60.046515

Fonte: Própria.

Na tabela 7 consta os dados para o material vidro, onde aumentando a quantidade de copos de vidro, o volume apresentava aumento e reduzindo a quantidade destes o volume decaía conseqüentemente. O Volume Máximo foi registrado em 68 % e as localizações do coletor foram vistas aproximadamente na plataforma de forma adequada.

Tabela 8 – Dados obtidos para o material Metal

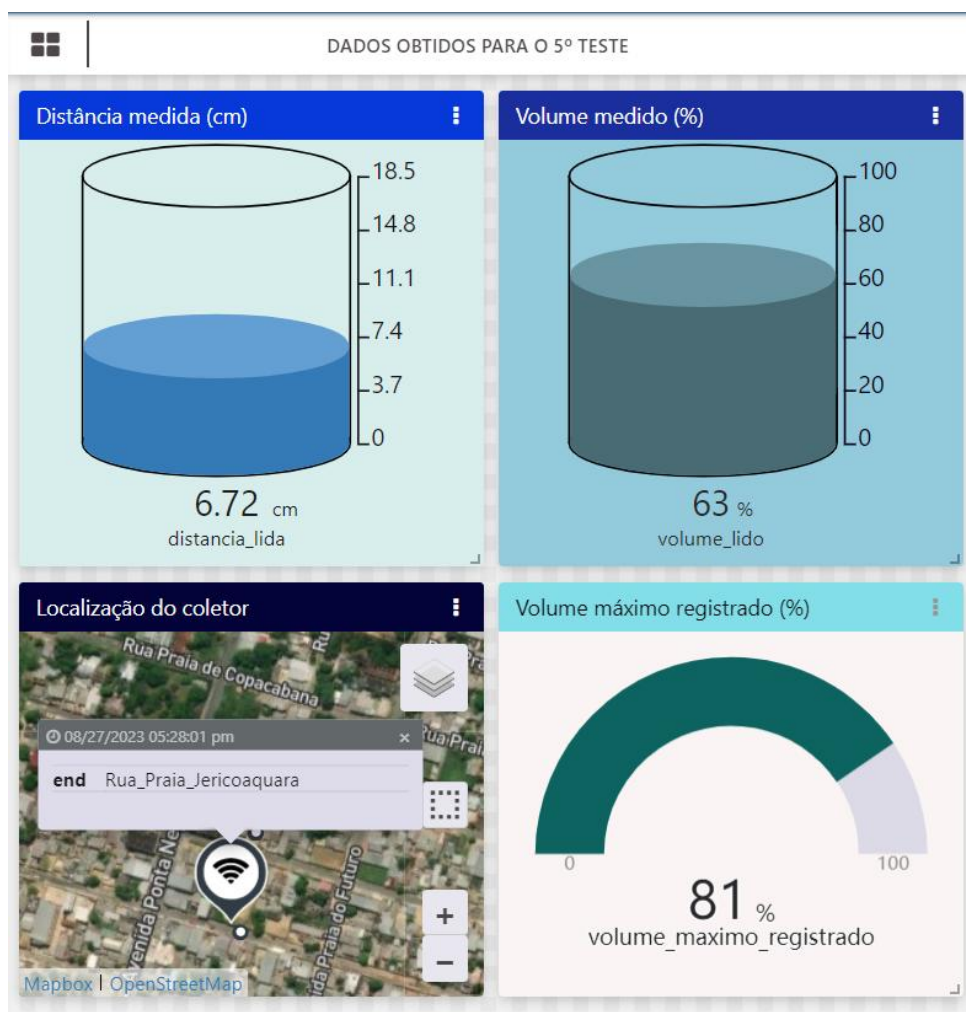
Casos	Distância(cm)	Volume(%)	Volume Máximo(%)	Latitude	Longitude
1º Teste	18.71	0	0	-3.007640	-60.046104
2º Teste	13.98	24	24	-3.007032	-60.046772
3º Teste	9.67	48	48	-3.007648	-60.047114
4º Teste	6.95	62	62	-3.007037	-60.046747
5º Teste	2.98	83	83	-3.007042	-60.047114

Fonte: Própria.

Por fim, para o último material testado o preenchimento do coletor com as latas de metal deu-se de forma crescente porém, para o 1º teste optou-se em deixar o coletor vazio. Pela tabela 8, o valor de volume medido foi de acordo com a ação inicial sendo de 0 %, contudo a distância lida pelo sensor que deveria ser a altura do recipiente medida em 18,45 cm, foi de 18,71 cm apresentando uma imprecisão no funcionamento do sensor.

Para o caso de teste deste material, as medições se apresentaram de forma crescente linear com o volume máximo alterando-se de acordo com o decorrer das leituras. Além disso, foi verificada a localização do coletor na plataforma em cada caso, sendo correspondida aproximadamente com os dados de latitude e longitude obtidos.

Figura 40 – *Dashboard* referente ao papel.

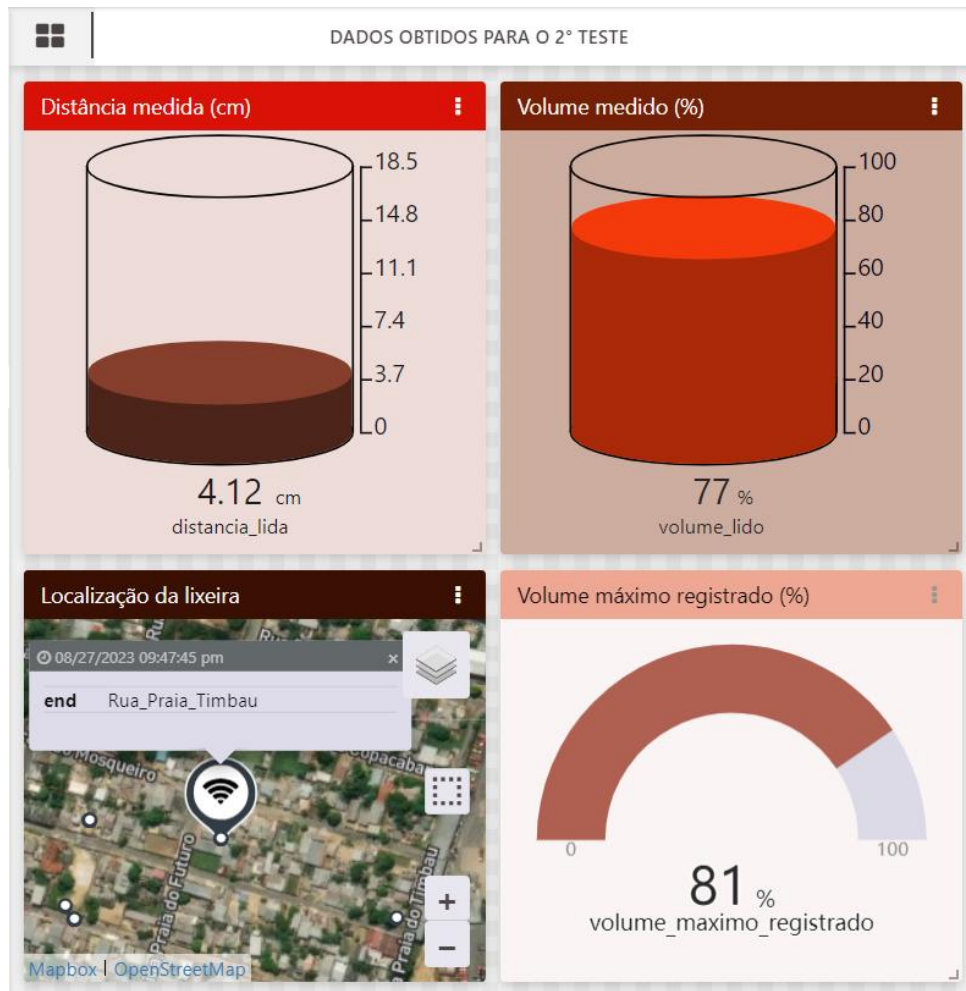


Fonte: Própria.

Na figura 40 é exibido o *Dashboard* feito para o 5º caso de teste do material papel, onde os valores mostrados são vistos na Tabela 5. Um fato a se destacar é a localização do coletor informada no *Dashboard* sendo definida no endereço da Rua Praia de Jericoaquara. Com o volume máximo registrado em 81 % e o volume atual sendo de 63 % pode indicar uma situação em que o coletor já havia sido recolhido anteriormente mas com o seu

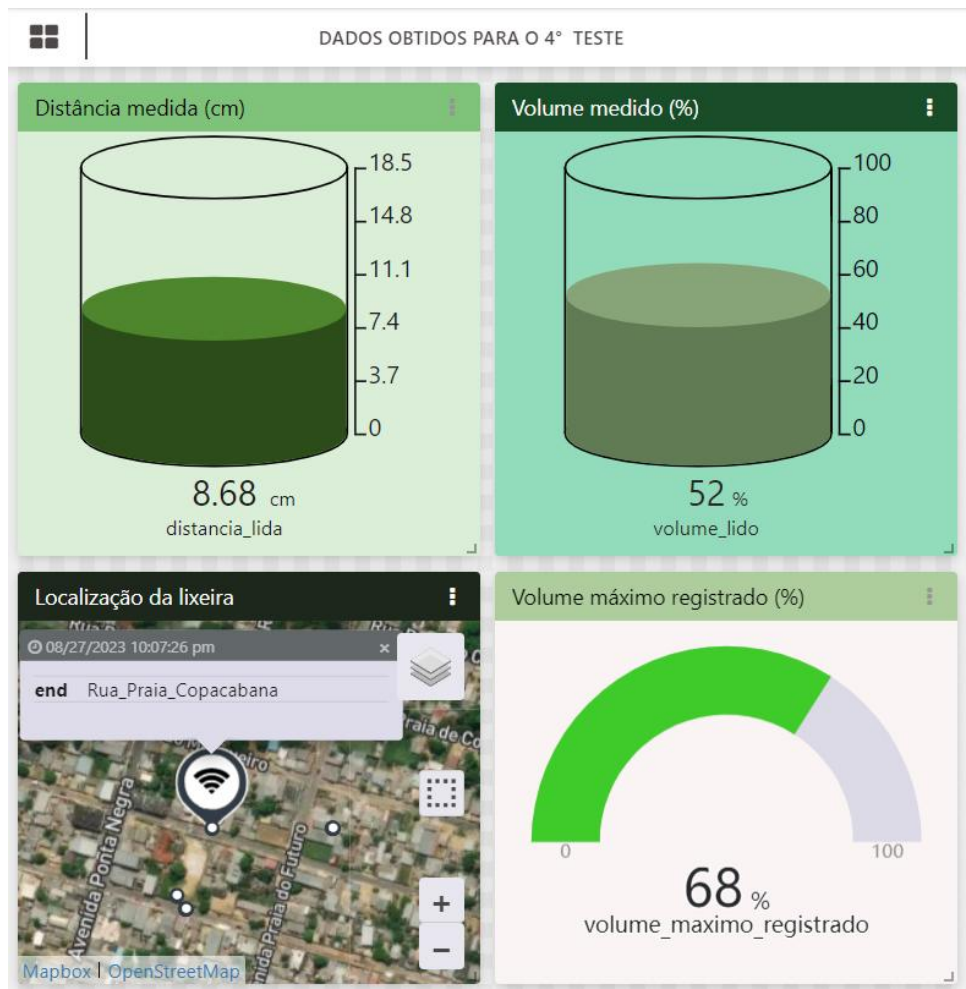
enchimento alcançando 63 % representa uma situação em que se deveria atentar para a realização da próxima coleta.

Figura 41 – *Dashboard* referente ao plástico.



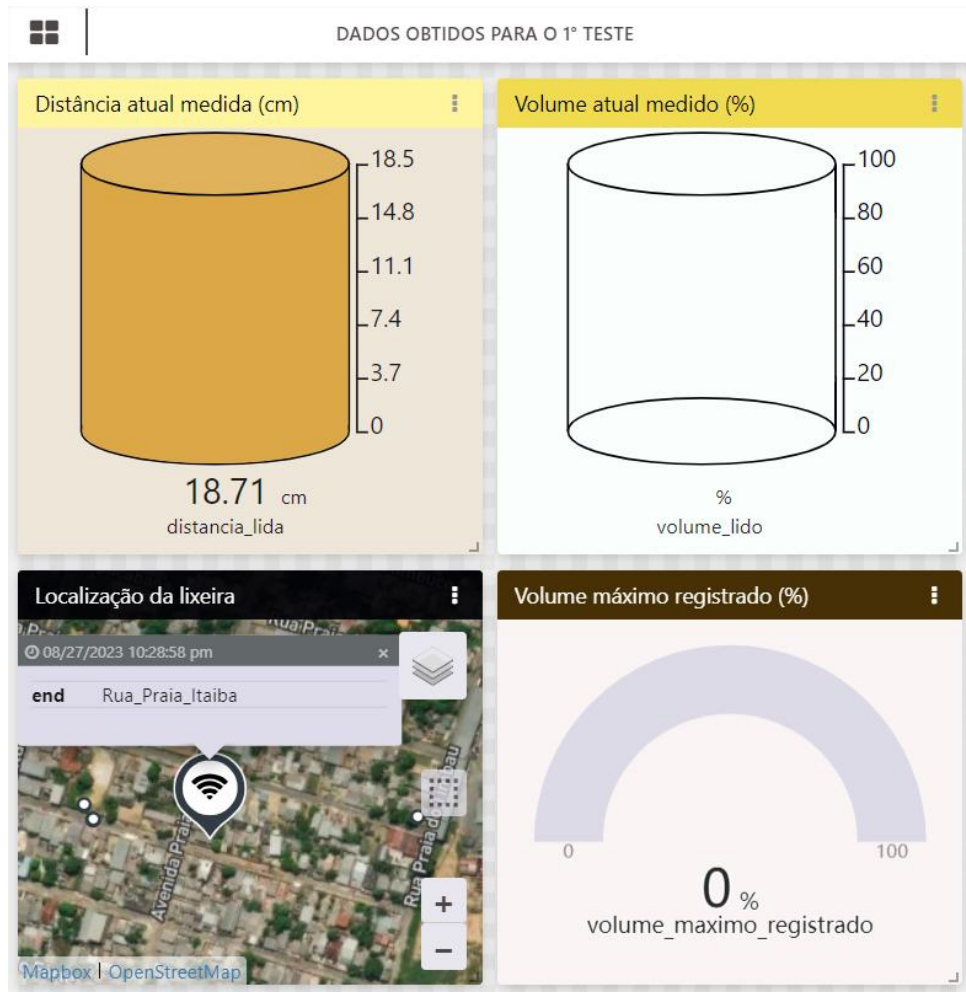
Fonte: Própria.

Na figura 41 é visualizado o *Dashboard* feito para o 2º caso de teste do material plástico, onde os valores mostrados são vistos na Tabela 6. A localização do coletor informada no *Dashboard* consiste no endereço da Rua Praia Timbau. Com o volume máximo registrado em 81 % e o volume atual sendo de 77 % pode indicar uma situação em que o coletor já havia sido recolhido anteriormente, mas com o seu atual enchimento sendo de 77 % representa uma situação crítica em que brevemente necessitaria do recolhimento.

Figura 42 – *Dashboard* referente ao vidro.

Fonte: Própria.

Na figura 42 é apresentado o *Dashboard* feito para o 4° caso de teste do material vidro, onde os valores mostrados são vistos na Tabela 7. A localização do coletor informada mostra-se no endereço da Rua Praia Copacabana. Com o volume atual medido sendo de 52 % indica uma situação em que o coletor estaria cheio pela metade aproximadamente.

Figura 43 – *Dashboard* referente ao Metal.

Fonte: Própria.

Na figura 43 é mostrado o *Dashboard* feito para o 1º caso de teste do material metal, onde os valores mostrados são vistos na Tabela 8. A localização do coletor identifica-se pela Rua Praia Itaiba e nesta ocasião, verifica-se que o coletor ilustrado em "Distância atual medida" encontra-se com coloração amarela em destaque, enquanto que o coletor visto em "Volume atual medido" mostra-se sem coloração. Essas observações estão de conformidade com os valores obtidos. Além disso, não havia um volume máximo registrado para o metal já que esse era o 1º teste para o material em questão.

Diante de todos os resultados apresentados, é importante observar que as informações mais relevantes visualizadas consistem no volume, volume máximo registrado e localização da lixeira pois expressam informações mais claras para se compreender o estado de como o coletor se encontrava. Porém decidiu-se mostrar no *display* e enviar as informações das distâncias lidas pelo sensor antes de serem convertidas em volume, para fins de complementação tanto dos dados como na criação dos *Dashboards*.

CONCLUSÃO

Este trabalho tem o intuito de desenvolver o protótipo de um sistema capaz de realizar o monitoramento remoto dos dados referentes a uma lixeira, também chamada de coletor, por meio de sensoriamento das variáveis e as suas respectivas transmissões por meio da tecnologia de comunicação LoRa e do protocolo de mensagens MQTT. Com a implementação do transmissor ao decorrer da montagem do coletor, bem como a do receptor realizadas, observa-se que o sistema apresenta um bom funcionamento, fornecendo resultados condizentes em sua maioria com os cenários de testes analisados, ou seja, os dados informam de maneira aceitável o *status* da lixeira.

Em relação aos dados mencionados anteriormente, admite-se que dentre os 4 tipos de informações do estado do coletor, aquelas que se referem ao volume e localização são as mais relevantes, pois em um contexto mais abrangente, torna-se mais fácil de se compreender o volume em porcentagem do coletor e o seu respectivo endereço do que a distância lida pelo sensor entre a superfície formada pelo acúmulo dos resíduos e a parte superior do coletor.

Vale ressaltar que embora as medições refletem de forma aceitável o estado da lixeira, essas não representam o nível de enchimento da lixeira com precisão em sua totalidade. Isto se deve ao fato do sensor utilizado no sistema ser o ultrassônico o qual aplica-se melhor em situações onde se realizam medições sobre superfícies regulares. Contudo, como as superfícies formadas pelo preenchimento dos resíduos são irregulares, um sensor de volume é o mais adequado para o projeto. Porém este sendo mais custoso, optou-se pelo sensor ultrassônico no desenvolvimento deste protótipo, pela sua acessibilidade no mercado. O fato é que os dados podem ser transmitidos via tecnologia de comunicação LoRa e protocolo MQTT, independentemente do tipo de resíduo contido na lixeira. De posse desses dados bem como a respectiva localização do coletor, é possível o planejamento diário do recolhimento, otimizando as rotas de coleta seletiva realizadas diariamente.

Como opções de melhoria para o avanço do projeto desenvolvido, sugere-se o monitoramento de uma maior quantidade de coletores com resíduos mais variados não restringindo-se somente aos recicláveis, e da utilização de sensores de volume no sistema. Além disso, pode-se estudar a possibilidade de implementação de uma tecnologia de comunicação mais robusta como o protocolo LoRaWAN para o envio de informações de coletores em larga escala por meio de *gateways* e de outras formas de compartilhamento de suas respectivas localizações.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABRELPE. *Panorama dos Resíduos Sólidos no Brasil*. 2020. Disponível em: <https://abrelpe.org.br/panorama-2020/>.
- ABRELPE. *Panorama dos Resíduos Sólidos no Brasil*. 2021. Disponível em: <https://abrelpe.org.br/panorama-2021/>.
- AHMAD, Fares. *Ultrasonic Sensors*. 2020. Disponível em: <https://www.researchgate.net/publication/341055286>.
- AL-FUQAHA, A. e. a. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communication Surveys Tutorials*, v. 17, n. 4, p. 2347–2376, 2015.
- ALLIANCE, L. A. *LoRaWAN What is it? A technical overview of LoRa and LoRaWAN*. California: SI, 2015.
- AMBIPAR. *O que é coleta seletiva e sua importância*. 2022. Disponível em: <https://www.brasilcoleta.com.br/o-que-e-coleta-seletiva-e-sua-importancia/#>.
- ARDUINO. *Overview of the Arduino IDE 1*. 2023. Disponível em: <https://docs.arduino.cc/software/ide-v1/tutorials/Environment>.
- BERTOLETI, P. *Projetos com ESP32 e LoRa*. [S.l.: s.n.], 2019.
- BRASIL, Lei nº 12.305. 2010. Disponível em: https://www.planalto.gov.br/ccivil/_03/_ato2007-2010/2010/lei/l12305.htm.
- CAETANO, Lauro et al. *Redes Veiculares: Tendências e Estudo de Caso*. 2016. Disponível em: https://www.researchgate.net/publication/310193861_Redes_Veiculares_Tendencias_e_Estudo_de_Caso.
- CHAUDHARI BHARAT S.; ZENNARO, M. *LPWAN Technologies for IoT and M2M Applications*. s.n: Academic Press, 2020.
- CHAXEL, F. e. a. A comparative study of lpwan technologies for large-scale iot deployment. *Science*, v. 5, p. 1–7, 2019.
- CONTEMAR. *Quais s ao os tipos de coleta de lixo e seu destino?*. 2020. Disponível em: <http://blog.contemar.com.br/coleta-de-lixos-tipos-e-destino/>.
- CRUZ, N. e. a. Lorawan and urban waste management—a trial. *Sensors 2021 "IoT Enabling Technologies for Smart Cities: Challenges and Approaches"*, v. 21, p. 1–18, 2021.
- ECYCLE. *Reciclagem: o que é e qual a sua importância?* 2022. Disponível em: <https://www.ecycle.com.br/reciclagem/>.
- ELETROGATE. *Módulo Sensor de Distância Ultrassônico HC-SR04*. 2022. Disponível em: <https://www.eletrogate.com/modulo-sensor-de-distancia-ultrassonico-hc-sr04>.
- ESPRESSIF. *About Espressif*. 2023. Disponível em: <https://www.espressif.com/en/company/about-espressif>.
- FEDCHENKOV, P. e. a. Supporting data communications in iot-enabled waste management. *Springer In Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, p. 163–174, 2017.

- HELTEC. *WIFI LoRa 32 (V2) - Heltec Automation*. 2023. Disponível em: <https://heltec.org/project/wifi-lora-32/>.
- HÜBSCHMANN, Ida. *ESP32 for IoT: A Complete Guide*. 2020. Disponível em: <https://www.nabto.com/guide-to-iot-esp-32/>.
- INTERNATIONALIT. *Topologia de Rede: Conheça os principais tipos*. 2022. Disponível em: <https://www.internationalit.com/post/topologia-de-rede-conhe%C3%A7a-os-principais-tipos>.
- KAZIENKO J.; QUINCOZES, S. T. E. Mqtt protocol: Fundamentals, tools and future directions. *IEEE Latin America Transactions*, v. 17, n. 9, p. 1439–1448, 2019. Disponível em: <https://ieeexplore.ieee.org/document/8931137>.
- KHOA, Ahn et al. *Waste management system using iot-based machine learning in university*. 2020.
- KHOMP. *Conheça a TagoIO, a primeira ferramenta cloud para desenvolvimento de solução IoT homologada Khomp*. 2019. Disponível em: <https://www.khomp.com/pt/tagoio-solucao-iot/>.
- MAKWANA ASHWIN; SONI, D. A survey on mqtt: A protocol of internet of things(iot). 2017. Disponível em: <https://www.researchgate.net/publication/316018571>.
- MONTAGNY, S. *LoRa - LoRaWAN and Internet of Things for beginners*. [S.l.: s.n.], 2022.
- MORÓN, Z. Sensors - what and how should we teach? *Joint IMEKO TC-1 XXXIV MKM Conference Wroclaw*, p. 8–12, 2002. Disponível em: <https://www.imeko.org/publications/tc1-2002/IMEKO-TC1-2002-021.pdf>.
- PARDINI, A. e. a. *Iot-based solid waste management solutions: A survey*. [S.l.: s.n.], 2019.
- PARZIALE L.; ROSSELOT, N. e. a. *TCP/IP Tutorial and Technical Overview*. [S.l.: s.n.], 2006.
- PRIMA. *Coleta Seletiva*. 2023. Disponível em: <https://prima.org.br/coleta-seletiva/>.
- SEMULSP. *Coleta e transporte dos res iduos sólidos*. 2021. Disponível em: <https://semulsp.manaus.am.gov.br/servicos/>.
- SHEA, S. *LPWAN low-power wide area network*. 2017. Disponível em: <https://www.techtarget.com/iotagenda/definition/LPWAN-low-power-wide-area-network>.
- SINIR. *Resíduos Sólidos Urbanos*. 2020. Disponível em: <https://sinir.gov.br/informacoes/tipos-de-residuos/residuos-solidos-urbanos/>.
- TAGOIO. *TagoIO: IoT Cloud Platform*. 2023. Disponível em: <https://tago.io/>.