

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA

DE MINAS GERAIS - *CAMPUS* FORMIGA

BACHARELADO EM ENGENHARIA ELÉTRICA

Vanessa Gorete da Silva

**DESENVOLVIMENTO DE PROTÓTIPO PARA MONITORAMENTO DE INSUMOS  
VIA REDE SEM FIO COM TECNOLOGIA DE COMUNICAÇÃO LORA® E  
PLATAFORMA THINGSPEAK™**

Formiga - MG

2022

VANESSA GORETE DA SILVA

**DESENVOLVIMENTO DE PROTÓTIPO PARA MONITORAMENTO DE INSUMOS  
VIA REDE SEM FIO COM TECNOLOGIA DE COMUNICAÇÃO LORA® E  
PLATAFORMA THINGSPEAK™**

Trabalho de Conclusão de Curso apresentado  
ao Curso de Bacharelado em Engenharia  
Elétrica do Instituto Federal de Minas Gerais  
- *Campus* Formiga para obtenção do grau de  
Bacharel em Engenharia Elétrica.

Orientador: Prof. Me. Marco Antônio Silva  
Pereira

Formiga

2022

Silva, Vanessa Gorete da

S586d          Desenvolvimento de protótipo para monitoramento de insumos via rede sem fio com tecnologia de comunicação LoRa® e plataforma ThingSpeak™ / Vanessa Gorete da Silva -- Formiga : IFMG, 2022.

75p. : il.

Orientador: Prof. MSc. Marco Antônio Silva Pereira

Trabalho de Conclusão de Curso – Instituto Federal de Educação,  
Ciência e Tecnologia de Minas Gerais – *Campus* Formiga.

1. Internet das coisas. 2. LoRa®. 3. ThingSpeak™.  
4. Comunicação. 5. Insumos. I. Pereira, Marco Antônio Silva. II. Título.

CDD 621.3

VANESSA GORETE DA SILVA

**DESENVOLVIMENTO DE PROTÓTIPO PARA MONITORAMENTO DE INSUMOS  
VIA REDE SEM FIO COM TECNOLOGIA DE COMUNICAÇÃO LORA® E  
PLATAFORMA THINGSPEAK™**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Engenharia Elétrica do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – *Campus Formiga* como requisito para obtenção do título de bacharel em Engenharia Elétrica.

Orientador: Prof. Me. Marco Antônio Silva Pereira

Avaliado em 29 de novembro de 2022.

Nota: 96,0

**BANCA EXAMINADORA**

Marco Antônio Silva Pereira

Prof. Me. Marco Antônio Silva Pereira (Orientador)

Patrick Santos de Oliveira

Prof. Dr. Patrick Santos de Oliveira

Rayane Rainer Leal

Prof. Ma. Rayane Rainer Leal

## AGRADECIMENTOS

Agradeço primeiramente a Deus, por estar comigo, me auxiliando em cada momento e me trazendo à memória o que me traz esperança, me fazendo perseverar para que este dia chegasse. Ele e eu sabemos como desejei este momento, e o quanto sou grata por me permitir vivenciar experiências que, se não fosse por Ele, não aconteceriam. A Ele, meu melhor e mais fiel amigo, toda a minha mais profunda gratidão.

Agradeço aos meus pais, Nivaldo e Marina, pelo carinho e ensinamentos na vida, e por me ajudarem em diversas ocasiões, me incentivando a continuar buscando um futuro melhor. À minha mãe, um dos grandes amores da minha vida, que permaneceu viva para ver esse momento. Ao meu pai, pelas incontáveis vezes que precisei de ajuda e ter com quem contar.

À minha irmã Patrícia, que desempenhou como sempre um excelente papel de irmã mais velha, expressei toda a minha gratidão, por sempre me estimular a continuar, mesmo diante das dificuldades, e me mostrar que sempre há uma luz no fim do túnel, por vibrar comigo nas conquistas, por cada gesto de amor e carinho, por cuidar de mim e me acolher sempre que precisei, por me lembrar onde estou e onde quero chegar e especialmente por me dar uma rosa vermelha quando terminei as disciplinas.

À minha irmã Poliane, agradeço por sempre me trazer uma perspectiva de futuro, por acreditar em mim mesmo quando nem eu mesma acreditei, e por me fazer enxergar além das circunstâncias. Também por passar horas me ouvindo falar sobre o curso, sobre meus projetos, por estar comigo em todos os momentos dessa trajetória e por cada gesto de cuidado, por tentar me fazer mais feliz, seja com uma palavra, um puxão de orelha ou um docinho.

À minha princesa, amiga, conselheira e motivação diária, Emily, por ser uma das minhas pessoas e um dos estímulos a querer buscar um futuro melhor com os estudos. Sou grata por cada palavra e cada silêncio, e se estou aqui, é por querer ser um exemplo, e um porto seguro quando for a sua vez de escolher o caminho, segurando sua mão, assim como me senti segura por simplesmente ter você. “É o tempo que dedicaste a tua rosa que a fez tão importante”.

Agradeço a Bia, minha mais que tia preferida, mais que amigas: friends, por todos os conselhos e por colocar meus pés no chão quando não sabia o que fazer. Por cada madrugada de conversa, pelo cuidado e amor, e pelos momentos divertidos que me proporcionou. A caminhada se tornou mais leve com suas inúmeras palavras. Ao Guilherme, João Paulo e

Júlio, mais que primos e tio, uma família pra mim.

Cadu, minha estrelinha, talvez você não possa me ouvir e nem ler este trabalho, mas a você o dedico. Por ser quem você foi, pelo seu brilho, por ter sido sempre tão amável. Agradeço por cada elogio que me fez e por cada memória. Jamais vou me esquecer de você, e prometo ser a melhor Engenheira que puder. Obrigada pela sua passagem pela Terra, e por ter feito minha vida mais feliz no tempo em que esteve por aqui. A Varinha das Varinhas vai estar guardada comigo pra sempre.

Gratidão ao meu orientador, Marco Antônio Silva Pereira, pela disponibilidade e atenção, e por sempre tirar dúvidas, buscar soluções e pela orientação durante toda a jornada deste trabalho de conclusão de curso.

À professora Danielle Cristina Teles Ferreira, minha eterna gratidão. Há pessoas que nos inspiram, e ela é, sem dúvidas, uma das minhas maiores inspirações.

Agradeço a cada professor do IFMG por contribuírem ativamente para minha formação, compartilharem seu conhecimento e serem solícitos e compreensivos em muitos momentos de dificuldade que enfrentei durante a graduação. Vivenciei gestos que guardarei para sempre comigo.

Agradeço as minhas amigas e amigos, Stéfane, Gê, Peu, Jaderson, pelos bons momentos.

Aos melhores amigos que encontrei durante a graduação e espero levar por toda a vida: Lorena, Roberta, Dandarah, Whester e Ana Luíza pelos incontáveis bons momentos na faculdade, e por passarmos juntos por momentos memoráveis, amo vocês. Eu sei que palavras jamais serão capazes de expressar a gratidão que sinto por ter tido o presente de conhecer vocês. Se hoje estou aqui, vocês são grande parte do motivo para que isso acontecesse. Especialmente à Lorena, minha dupla e companheira em trabalhos e em situações da vida, saiba que jamais vou me esquecer de cada palavra, gesto e cada apoio, você sempre será muito importante pra mim. Ao Whester, com quem dividi muitos momentos, trabalhos, ideias e conversas, por toda a ajuda durante todo o curso e por toda a compreensão. Você sempre será meu Engenheiro Eletricista preferido.

Gratidão aos outros amigos que conquistei durante essa jornada. Eduardo, Thaymara, e André, obrigada pela parceria, e pelos momentos em que compartilhamos experiências e sentimentos. Vocês foram incríveis.

Por fim, agradeço a todos os que contribuíram direta ou indiretamente para este momento.

“A felicidade pode ser encontrada até nas horas mais difíceis, se você se lembrar de acender a luz.”

J. K. Rowling

## RESUMO

Dentre os pilares da Quarta Revolução Industrial, destaca-se a tecnologia de *IoT* (do inglês *Internet das Coisas*), apresentando atualmente um dos conceitos mais importantes no que diz respeito ao avanço tecnológico. Permitindo a conexão entre dispositivos diversos com a rede, a *Internet of Things* (IoT) possibilita a otimização de recursos aplicados a ambientes residenciais, industriais, hospitalares, agropecuários, dentre outros. Neste sentido, diferentes tecnologias são desenvolvidas, a fim de suprir necessidades dos mais diversos nichos mercadológicos. Sob este viés, a comunicação sem fio traduz-se como uma importante ferramenta no que tange à implementação de metodologias que viabilizem a otimização na execução de processos distintos. Neste contexto, este trabalho tem como objetivo apresentar o desenvolvimento de um protótipo para monitoramento de insumos utilizando conceitos de *IoT*. O sistema proposto emprega a tecnologia de radiofrequência *LoRa*® (*Long Range*), baseado na técnica de espalhamento de espectro, apresentando baixo consumo energético e capacidade de comunicação a longas distâncias, possibilitando assim o monitoramento e controle de dispositivos alocados em regiões com acesso precário à *Internet*. Para o desenvolvimento do protótipo foram utilizados dois módulos *Heltec Wi-fi ESP32 LoRa* (V2) para transmissão de dados via *LoRa*® e *Wi-Fi* e dois sensores ultrassônicos HC-SR04. A tecnologia de radiofrequência é responsável por comunicar os dois módulos, visto que estes operam como centrais conectivas, comunicando o chamado *Sender* ao denominado *Receiver*. Recebidos os dados, estes são enviados para a plataforma *IoT Thingspeak*™ e tratados para retornar ao usuário informações pertinentes. Os resultados obtidos são demonstrados neste documento, detalhando os recursos utilizados na implementação.

**Palavras-chave:** *Internet das Coisas*; *LoRa*®; *Thingspeak*™.

## ABSTRACT

Among the pillars of the Fourth Industrial Revolution, IoT (Internet of Things) technology stands out, currently presenting one of the most important concepts of technological advancement. Allowing the connection between different devices with the network, the Internet of Things (IoT) allows the optimization of resources applied to residential, industrial, hospital, and agricultural environments, among others. In this sense, different technologies are developed to meet the needs of the most diverse market niches. Under this bias, wireless communication is an essential tool in terms of the implementation of methodologies that make possible the optimization of the execution of different processes. In this context, this work aims to present the development of a prototype for monitoring inputs using IoT concepts. The proposed system employs the LoRa® (Long Range) radiofrequency technology, based on the spread-spectrum technique, presenting low energy consumption and communication capacity over long distances, thus enabling the monitoring and control of devices located in regions with precarious access to the Internet. For the development of the prototype, two Heltec Wi-fi ESP32 LoRa (V2) modules were used for data transmission via LoRa® and Wi-Fi, and two HC-SR04 ultrasonic sensors. Radiofrequency technology is responsible for communicating the two modules, since they operate as connective centers, communicating the so-called Sender to the Receiver. Once the data is received, it is sent to the IoT Thingspeak™ platform and processed to return relevant information to the user. The results obtained are shown in this document, detailing the resources used in the implementation.

**Keywords:** Internet of Things; LoRa®; Thingspeak™.

## LISTA DE ABREVIATURAS

ANATEL – Agência Nacional de Telecomunicações

BNDES – Banco Nacional de Desenvolvimento Econômico e Social

BW – *Band Width*

CPU – *Central Processing Unit*

CR – *Code Rate*

GSM – *Global System for Mobile*

IBGE – Instituto Brasileiro de Geografia e Estatística

IOT – *Internet of Things*

ISM – *Industrial Scientific and Medical*

ITU – *International Telecommunication Union*

LORA® – *Long Range*

LORAWAN - *Long Range Wide Area Network*

LPWAN – *Low Power Wide Area Network*

MCTIC – Ministério da Ciência, Tecnologia, Inovações e Comunicações

RAM – *Random Access Memory*

RFID – *Radio-Frequency Identification*

ROM – *Read Only Memory*

RSSI – *Received signal strength indication*

SF – *Spreading Factor*

USB – *Universal Serial Bus*

## LISTA DE FIGURAS

Figura 1 - Sensor ultrassónico HC-SR04 .....	15
Figura 2 - Arquitetura de um microcontrolador .....	16
Figura 3 - Heltec ESP32 LoRa (V2) .....	16
Figura 4 - Arduino IDE .....	19
Figura 5 - Propriedades da tecnologia LoRa® .....	20
Figura 6 - Comparação entre Wi-Fi, Bluetooth e LoRa® .....	21
Figura 7 - Modulação utilizando pulsos Chirp .....	22
Figura 8 - Frequência de banda utilizada por dispositivos que empregam tecnologia LoRa® em diferentes locais do planeta.....	23
Figura 9 - Evolução da Internet .....	24
Figura 10 - Diagrama de aplicação IoT.....	25
Figura 11 - Pinout da placa Heltec ESP32 LoRa (V2) .....	26
Figura 12 - Comunicação entre dispositivos .....	28
Figura 13 - Definição de preferências .....	29
Figura 14 - Instalação do pacote Heltec ESP32 Series Dev-boards .....	30
Figura 15 - Instalação da biblioteca Heltec ESP32.....	30
Figura 16 - Seleção da placa Heltec ESP32 LoRa (V2) .....	31
Figura 17 - Fluxograma do funcionamento do sistema de leitura de dados.....	32
Figura 18 - LoRa Sender em funcionamento.....	33
Figura 19 - Lora Receiver aguardando comunicação.....	34
Figura 20 - Dispositivo LoRa Receiver recebendo dados.....	35
Figura 21 - Fluxograma do funcionamento do dispositivo Receiver.....	36
Figura 22 - Fluxograma de sistema de aquisição de dados pelo ThingSpeak™, aplicativo e Widget .....	38
Figura 23 - Proposta do sistema .....	39
Figura 24 - Protótipo final.....	40
Figura 25 - Esquemático LoRaSender .....	41
Figura 26 - Saída SerialArduino IDE - Comunicação via rádio inicializada .....	41
Figura 27 - Saída SerialArduino IDE - Conexão com a rede .....	42
Figura 28 - Trajeto percorrido - Teste 1.....	42
Figura 29 - Trajeto percorrido - Teste 2.....	43

Figura 30 - Trajeto percorrido - Teste 3.....	43
Figura 31 - Interface inicial em plataforma IoT .....	44
Figura 32 - Visualização gráfica –Variação da Capacidade do Reservatório .....	45
Figura 33 - Visualização gráfica - Decréscimo da Capacidade do Estoque.....	46
Figura 34 - Visualização gráfica - Consumo do estoque em porcentagem.....	47
Figura 35 - Visualização gráfica – Custo .....	48
Figura 36 - Visualização numérica do custo .....	48
Figura 37 - Configurações do recurso React .....	50
Figura 38 - Preenchimento de dados do canal .....	51
Figura 39 - Tela de confirmação de dados do canal .....	52
Figura 40 - Informações de reservatório e estoque.....	53
Figura 41 - Informações de consumo e custo .....	54
Figura 42 - Configurações Widget.....	55
Figura 43 - Visualização numérica e gráfica em Widget .....	56
Figura 44 - Sistema de alerta .....	57

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1	Justificativa	11
1.2	Objetivos	13
1.2.1	<i>Objetivos gerais</i>	13
1.2.2	<i>Objetivos específicos</i>	13
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>14</b>
2.1	Dispositivos eletrônicos e sensores	14
2.2	Microcontroladores	15
2.3	Arduino IDE	18
2.4	Comunicação	20
2.4.1	<i>LoRa®</i>	20
2.5	<i>Internet das Coisas</i>	23
2.6	<i>ThingSpeak™</i>	25
<b>3</b>	<b>METODOLOGIA</b>	<b>26</b>
3.1	<i>Heltec ESP32 LoRa (V2)</i>	26
3.2	Arquitetura do projeto	27
3.3	Desenvolvimento do sistema	28
3.3.1	<i>Configuração do ambiente de programação</i>	28
3.3.2	<i>Leitura de dados referentes ao reservatório e estoque</i>	31
3.3.3	<i>LoRa Sender</i>	32
3.3.4	<i>LoRa Receiver</i>	33
3.3.5	<i>Aquisição de dados Pelo ThingSpeak™</i>	36
3.4	Considerações finais sobre o sistema proposto	38
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>40</b>
4.1	<i>LoRa Sender</i>	40
4.2	<i>LoRa Receiver</i>	41
4.3	Teste de alcance <i>LoRa®</i>	42
4.4	Plataforma <i>IoT- ThingSpeak™</i>	44
4.4.1	<i>Análise de dados</i>	49
4.5	<i>Dashboard aplicativo ThingView – ThingSpeak viewer</i>	50
4.6	<i>Dashboard IoT ThingSpeak Monitor Widget</i>	55
4.7	Análise de custos e viabilidade técnica	57
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>59</b>

<b>REFERÊNCIAS</b>	<b>60</b>
<b>ANEXOS A – CÓDIGO <i>SENDER</i></b>	<b>65</b>
<b>ANEXOS B – CÓDIGO <i>RECEIVER</i></b>	<b>67</b>
<b>ANEXOS C – CÓDIGO <i>MATLAB ANALYSIS</i></b>	<b>70</b>

## 1 INTRODUÇÃO

A evolução tecnológica enquanto fator de progresso, impulsionada por demandas globais de interações socioeconômicas produziu nas últimas décadas uma série de avanços em diferentes esferas, entre os quais está inserida a criação e evolução da *Internet*. Sob pilares fundamentais, surge em 2011 na feira de *Hannover* (Alemanha) o termo Indústria 4.0, proveniente de um projeto de estratégias governamentais alemãs para o desenvolvimento tecnológico no âmbito industrial (ABREU et al., 2017). Segundo Zawadzki e Zywicki (2016), a Indústria 4.0 é caracterizada pelas conquistas tecnológicas dos últimos anos, buscando um futuro com sistemas de produção inteligentes e automatizados, qualificada como a Quarta Revolução Industrial.

Sob este viés, caracterizado como um dos pilares fundamentais desta nova era tecnológica, o termo *Internet of Things* diz respeito à integração de dispositivos com a *Internet*, personalizando a interação de usuários com diversos objetos e possibilitando inovações tecnológicas, maior poder de gestão de processos, melhoria na execução de processos e autonomia, aplicados a diversos domínios, incluindo diversas arquiteturas, padronizações e protocolos (ČOLAKOVIĆ; HADŽIALIĆ, 2018). Diversas tecnologias empregando fundamentos da *Internet* das Coisas vem sendo desenvolvidas, e a busca por ferramentas que permitam a conexão entre usuário e dispositivo de maneira satisfatória e segura segue crescente.

Segundo o Relatório Global de Conectividade de 2022 publicado pela ITU (*International Telecommunication Union* – União Internacional de Telecomunicações), 2,9 bilhões de pessoas não têm acesso à *Internet* no mundo. Ainda segundo a ITU, a conectividade em áreas urbanas é o dobro em comparação a áreas rurais. No Brasil, em áreas urbanas o acesso à *Internet* abrange 92,3% das residências, enquanto em áreas rurais 74,7% dos domicílios têm acesso à rede (IBGE, 2022). Visto isto, tecnologias que abranjam locais remotos e de difícil acesso à rede vem sendo desenvolvidas, a fim de otimizar processos inerentes a estas localidades, tal qual o objeto deste estudo, que se trata do controle e monitoramento de insumos via rede sem fio.

Sabendo do baixo nível de acesso à rede em localidades rurais, são imprescindíveis tecnologias que possibilitem a otimização de processos. Neste sentido, tecnologias de comunicação por radiofrequência têm se mostrado promissoras. A tecnologia de comunicação

*LoRa*®, capaz de transferir dados a longas distâncias com baixo consumo energético mostra-se como uma próspera metodologia de aplicação *IoT*, visto sua capacidade de comunicação com módulos *LoRa*®, e posterior possibilidade de agregar dados à *Internet*.

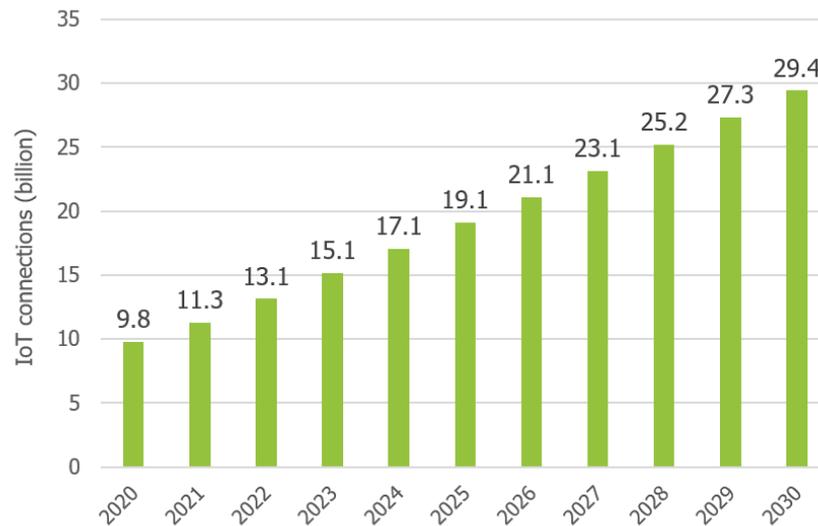
Alguns módulos, tais como o *Heltec ESP32 LoRa (V2)* permitem a comunicação e transmissão de dados a longas distâncias por radiofrequência, *Bluetooth* e *Wi-Fi*, e, devido a tais características permite a transferência de dados para a rede. Neste trabalho, dados provenientes da medição de grandezas físicas de um sistema de teste, relacionados especificadamente com a quantidade de insumo em recipientes, são disponibilizados na rede através da plataforma *ThingSpeak*<sup>TM</sup>, que viabiliza a visualização dos dados recebidos, bem como fornece ferramentas para o tratamento e análise destes através de programação em *MATLAB*® (MIRE; ARAMICE, 2020).

## 1.1 Justificativa

Segundo o relatório da *Transforma Insights*, intitulado *Global IoT Forecast Report, 2021-2030*, estima-se que, em 2021, 11,3 bilhões de dispositivos *IoT* estavam ativos. Ainda segundo o relatório, estima-se que em 2030 29,4 bilhões de dispositivos *IoT* estarão ativos, movimentando cerca de US\$ 621 bilhões no mercado da *Internet of Things*. O Gráfico 1 ilustra a previsão de evolução do uso de dispositivos *IoT* ativos até o ano de 2030.

Gráfico 1 - Previsão de conexões *IoT* 2020-2030**IoT connections forecast 2020-2030**

[Source: Transforma Insights TAM Forecasts, 2022]

Fonte: Extraída de (Transforma *Insights*, 2022)

Em um estudo conduzido pelo BNDES em parceria com o MCTIC evidenciou-se que no âmbito rural, o potencial lucrativo com a utilização de dispositivos que integrem tecnologia *IoT* será de aproximadamente US\$ 350 bilhões em 2025. Segundo Silva e Espejo (2020), no setor agropecuário, a *Internet das Coisas (IoT)* viabiliza a eficiência produtiva, segurança alimentícia e redução de custos com produtos e insumos. Ainda segundo Dinesh e Ramesh, o uso de *IoT* no agronegócio propicia a aplicação no controle, monitoramento, predição e logística no agronegócio, possibilitando melhorias nas condições de gestão no setor primário da economia.

Dados do Censo Agropecuário de 2017 demonstram que apenas 9,5% dos estabelecimentos agropecuários no Brasil têm acesso à *Internet*. Para Souza, Gomes e Alves (2020), a concentração da produção está intrinsecamente ligada à tecnologia. Deste modo, novas metodologias têm tomado espaço em diversas localidades, tais quais em locais de setores primários da economia. Neste sentido, as propriedades da tecnologia *LoRa*® conferem a ela possíveis aplicações em sistemas baseados em *IoT*, principalmente por sua capacidade de comunicação a longas distâncias por radiofrequência, operação com baixo consumo energético, e custo acessível.

## 1.2 Objetivos

### 1.2.1 Objetivos gerais

Apresentar um protótipo de monitoramento de insumos em áreas sem acesso à *Internet*, utilizando conceitos e aplicações relacionados a comunicação *LoRa*®, demonstrando suas aplicações em sistemas baseados em *IoT* para realização de análise de dados, exibindo resultados enviados via *Wi-Fi* à plataforma *ThingSpeak*™, assim como a apresentação de recursos vinculados à esta plataforma.

### 1.2.2 Objetivos específicos

Os objetivos específicos deste trabalho de conclusão de curso se baseiam no desenvolvimento de um protótipo para dispor ao usuário um sistema de visualização de dados alusivos ao monitoramento de insumos, viabilizando a verificação do consumo, possíveis necessidades de ações a serem executadas e monitoramento de gastos.

Para desenvolvimento do protótipo proposto, foi necessário estabelecer as seguintes metas:

- Realizar a leitura do nível de recipientes utilizando sensores ultrassônicos do tipo HC-SR04;
- Trafegar informações entre dois módulos *Heltec ESP32 LoRa (V2)*, via comunicação *Lora*®;
- Configurar plataforma *ThingSpeak*™ para recebimento de informações via *Wi-Fi* de um dos módulos *Heltec ESP32 LoRa (V2)*.
- Utilizar recursos de *React* para alertar ao usuário quanto à baixa de insumos.
- Elaborar gráficos para monitoramento do custo associado à utilização dos insumos.

## 2 FUNDAMENTAÇÃO TEÓRICA

De forma a auxiliar o entendimento referente a metodologia utilizada neste trabalho, o referencial teórico elucida os principais conceitos sobre o tema estudado, apresentando os dispositivos utilizados para confecção do protótipo para monitoramento de insumos, bem como plataforma utilizada para recepção e tratamento de dados.

### 2.1 Dispositivos eletrônicos e sensores

Para a execução de tarefas específicas, pode-se utilizar componentes eletrônicos associados a sensores e atuadores. Esta seção apresenta informações básicas sobre os sensores utilizados na implementação do protótipo proposto.

O sensor ultrassônico HC-SR04 utiliza o envio de sinais ultrassônicos para aferir a medida da posição de um objeto em determinado instante de tempo, utilizando, para tal, o envio do sinal de uma onda mecânica de alta frequência, determinando a distância percorrida pelo tempo em que a onda é emitida e retorna para a origem quando colide com algum obstáculo (TIRELLI, 2022). Por possuir seu princípio de funcionamento baseado na velocidade do som, que é constante no meio (340 m/s), a distância percorrida pela onda através da Equação 1.

$$d = \frac{t}{2v_s} \quad (1)$$

em que “d” é a distância percorrida, “t” o tempo em que a medição foi efetuada, “ $v_s$ ” a velocidade do som no ar, e a razão de  $\frac{1}{2}$  se refere ao movimento de ida e volta da onda.

O sensor ultrassônico HC-SR04 possui 4 pinos, sendo o pino de VCC, destinado à alimentação (5 Volts), GND (comum), *Trigger* (responsável por emitir o sinal ultrassônico. Recebe um sinal alto de 10 $\mu$ S para enviar pulsos), e o pino *Echo* (responsável por receber o retorno da onda). Quando energizado com a tensão de 5V, o pino *Trigger* envia o sinal, energizando o pino *Echo*, que, ao detectar o retorno da onda é desenergizado pelo sistema. Assim, a distância será calculada através do tempo em que o *Echo* permanece em nível alto, dividindo o resultado por 2 (DA SILVA et al., 2018). A corrente de trabalho do sensor é de 15 mA, mensurando distâncias que variam de 2 cm a 400 cm, com margem de erro de 3 mm, segundo seu *Datasheet*. Este sensor está ilustrado pela Figura 1, e é utilizado neste trabalho

em duas unidades, em que um sensor é responsável por realizar medições relacionadas a um reservatório de insumos e outro para aferir medições relacionadas a um estoque deste mesmo insumo.

Figura 1 - Sensor ultrassônico HC-SR04

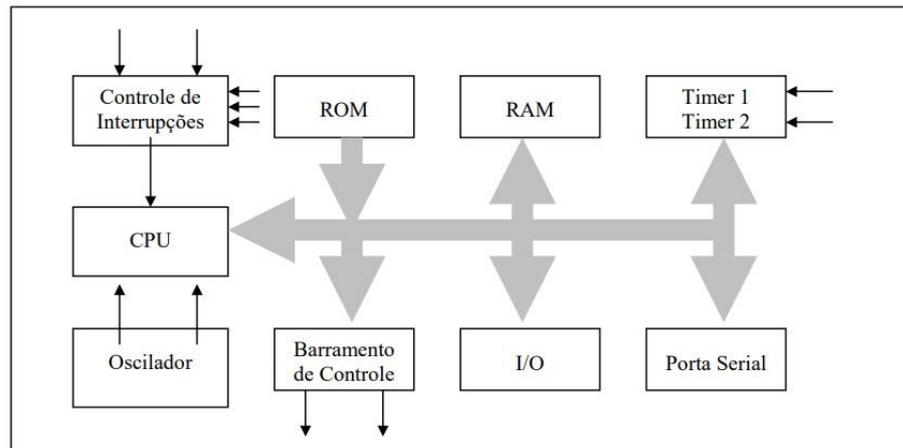


Fonte: Extraída de (TECNOTRONICS, 2022)

## 2.2 Microcontroladores

Os microcontroladores são caracterizados por serem dispositivos compostos por uma unidade de processamento central (*CPU*) e um conjunto de periféricos alocados em um único circuito integrado que envolve unidade de portas de entradas e saídas, memória *RAM* (memória volátil, utilizada para armazenar informações temporariamente), memória *ROM* (memória de leitura, onde as informações relacionadas a aplicação estão contidas), memória *FLASH* externa (interface para comunicação serial, temporizadores, conversores analógico/digitais, e são programados para executar tarefas específicas (KERSCHBAUMER, 2013). Desta forma, o microcontrolador se diferencia do PC devido ao encapsulamento dos componentes periféricos, acessíveis como posições convencionais de memória, sendo que, o *software* responsável por compilar informações também é caracterizado como seu sistema operacional (HUBERT, 2001). A Figura 2 expõe a arquitetura genérica de um microcontrolador.

Figura 2 - Arquitetura de um microcontrolador



Fonte: Extraído de (HUBERT, 2001)

Deste modo, com aplicações que vão desde automações residenciais, aplicações em ambientes hospitalares, indústrias, agronegócio, dentre outros, a escolha do microcontrolador ideal para aplicação em um projeto deve passar por uma criteriosa análise, a fim de atender as especificações e necessidades do sistema, tal qual a inserção do módulo *LoRa*® em outros microcontroladores como o Arduino e Raspberry Pi, avaliando suas necessidades de implantação de módulos *Wi-Fi* para comunicação com a plataforma *IoT*.

Para contemplar as demandas do projeto proposto neste trabalho, a placa desenvolvida pela empresa *Heltec Automation*, *Heltec ESP32 LoRa (V2)* foi escolhida devido a sua capacidade de processamento, integração *Wi-Fi*, *LoRa*®, *Bluetooth*, o que a torna conveniente para comunicação sem fio via radiofrequência e comunicação com a *Internet*. A Figura 3 ilustra a placa.

Figura 3 - *Heltec ESP32 LoRa (V2)*

Fonte: Extraído de (HELTECAUTOMATION, 2018)

A placa é composta por um microcontrolador ESP32, componentes referentes à tecnologia *LoRa*®, *display* e antena. Para melhor compreensão dos aspectos construtivos, bem como aspectos operacionais, as especificações da placa são elucidadas a seguir de maneira separada, por meio de tabelas. A Tabela 1 contém especificações do ESP32.

Tabela 1 - Especificações ESP32

<b>Placa Heltec ESP32 LoRa (V2)</b>	
<b>ESP 32</b>	
Fabricante	<i>Espressif</i>
Chip Principal	ESP32-D0WQD6
Tensão de alimentação	2,7 a 3,6 VDC
Processador	<i>Xtensa 32-Bit LX6 Dual Core</i>
Clock	80 a 240 MHz
Memória ROM	448Kb
Memória SRAM	520Kb
Memória FLASH	4Mb
Wi-Fi	802.11
Bluetooth	V4.2 BR/EDR e BLE(Low energy)
Temperatura de trabalho	-40° C a +85° C

Fonte: Adaptado de (HELTEC AUTOMATION, 2018) e (CASA DA ROBÓTICA, 2022)

. A Tabela 2 informa dados referentes à tecnologia *LoRa*® contidas na placa.

Tabela 2 - Especificações *LoRa*

<b>Placa Heltec ESP32 LoRa (V2)</b>	
<b>LoRa®</b>	
Chip Principal	SX1276
Tensão de alimentação	2,2 a 3,6 VDC
Frequência de operação	868MHz a 915MHz
Potência máxima TX	19 dB ± 1 dB
Sensibilidade do receptor	-135 dBm
Taxa de transferência	0,0018 à 37,5 Kbps
Interface de comunicação	SPI
Temperatura de trabalho	-40° C à +85° C

Fonte: Adaptado de (HELTEC AUTOMATION, 2018) e (CASA DA ROBÓTICA, 2022)

É possível escolher a frequência de operação da placa através da definição desta no

código compilado na IDE Arduino.

A Tabela 3 elucida dados referentes ao *display* da placa.

Tabela 3 - Especificações do Display

<b>Placa Heltec ESP32 LoRa (V2)</b>	
<b>Display</b>	
Tecnologia do <i>Display</i>	<i>OLED</i>
Tensão de operação	3,3V a 5V
Tamanho da tela	0,96"
Resolução	128x64 <i>pixels</i>
Ângulo de visão	> 160 °
Interface	I2C
Cor da escrita	Branca

Fonte: Adaptado de (*HELTEC AUTOMATION*, 2018) e (*CASA DA ROBÓTICA*, 2022)

A Tabela 4 demonstra especificações da antena contida na placa.

Tabela 4 - Especificações da antena

<b>Placa Heltec ESP32 LoRa (V2)</b>	
<b>Antena</b>	
Tipo de conector	<i>Ipex</i>
Frequência de banda	868MHz a 915MHz
Relação de onda estacionária	$\leq 1,5$
Ganho	3dBi
Potência máxima	10W

Fonte: Adaptado de (*HELTEC AUTOMATION*, 2018) e (*CASA DA ROBÓTICA*, 2022)

### 2.3 Arduino IDE

O Arduino é uma plataforma *open-source* de prototipagem eletrônica com *hardware* e *software* flexíveis e fáceis de usar, destinado a artistas, *designers*, *hobbistas* e qualquer pessoa interessada em criar objetos ou ambientes interativos. A Arduino IDE se refere ao *software* de código aberto que torna possível a escrita e *upload* do código que será compilado

para a placa (*hardware*). Pela Arduino IDE é possível escrever instruções para diversas placas, desde que as bibliotecas utilizadas atendam suas especificações. Com *layout* amigável e intuitivo, a IDE viabiliza a programação de placas de maneira simples e direta (ARDUINO, 2022). A Figura 4 demonstra a tela inicial da IDE Arduino, em que as funções *setup* (executada uma única vez) e a função *loop* (executada repetidamente) são predefinidas a fim de garantir o bom funcionamento do código (FONSECA, 2021).

Figura 4 - Arduino IDE



Fonte: Extraída pela autora, 2022

## 2.4 Comunicação

A transmissão efetiva de dados entre dispositivos conectados através da *Internet* é fruto de diversos parâmetros, tais quais: segurança, confiabilidade, gestão e escalabilidade. Um conjunto de descrições formais compõem as propriedades que regem as regras referentes à sintaxe, semântica e sincronização da comunicação, operando como uma linguagem universal entre dispositivos e uniformizando sua comunicação (JUNIOR, 2021).

### 2.4.1 LoRa®

Na era da *Internet* das Coisas, um extenso número de objetos está conectado por meio de tecnologias, tais quais o *Wi-Fi*, *Bluetooth*, *ZigBee*, *GSM*. Entretanto, suas características restritivas no que se refere a eficiência energética inviabilizam sua aplicação em projetos que demandam baixo consumo energético. Neste sentido, pode-se citar a rede *LPWAN* (*Low Power Wide Area Network*) que oferece cobertura por rádio adaptando a taxa de transferência, potência e modulação, a fim de viabilizar a troca de informações entre dispositivos com baixo consumo energético, através da tecnologia de comunicação por radiofrequência *LoRa®* (AUGUSTIN et. al., 2016). Algumas propriedades desta tecnologia podem ser visualizadas pela Figura 5.

Figura 5 - Propriedades da tecnologia *LoRa®*

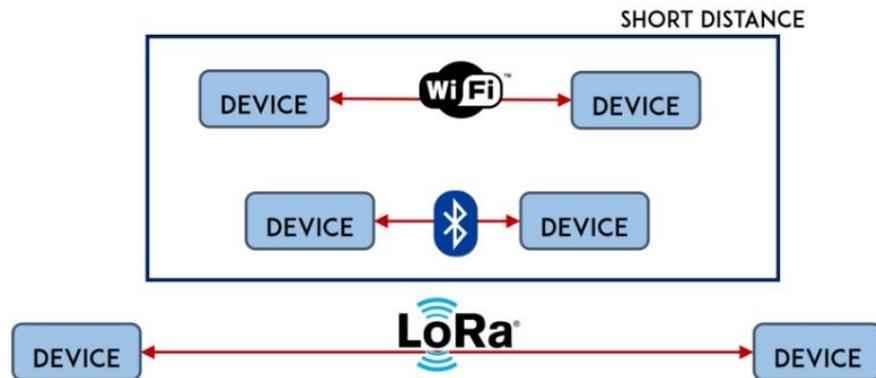


Fonte: Adaptada de (PINTO, 2019)

Ao contrário de tecnologias como *Wi-Fi* e *Bluetooth* que suportam a comunicação a pequenas distâncias – *Wi-Fi* na ordem de 40m e *Bluetooth* na ordem de 10m -, dois dispositivos *LoRa®* podem trocar informações a distâncias consideravelmente longas – na

ordem de Km -, conforme ilustrado pela Figura 6.

Figura 6 - Comparação entre *Wi-Fi*, *Bluetooth* e *LoRa*®

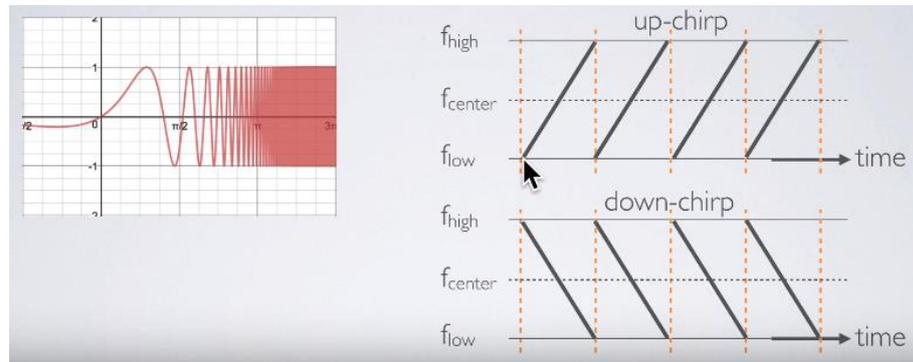


Fonte: Extraída de (RANDOM NERD, 2018)

O *LoRa*® é uma tecnologia *LPWAN* de camada física, desenvolvido pela *Clyceo de Genobre*, e adquirido pela *SemTech* em 2012 que possibilita a comunicação ponto a ponto através da radiofrequência. A tecnologia *LoRa*® utiliza como técnica de modulação o método de espalhamento por frequência *CSS* (do inglês *Chirp Spread Spectrum*), garantindo alta imunidade a ruídos e sinais de outra natureza, propagado em faixas de frequência *ISM* não licenciadas, utilizando pulsos *chirp* (ABEL, 2019).

Um pulso *chirp* aumenta (*Up-Chirp*) ou diminui (*Down-Chirp*) a frequência com o tempo. Logo, esta técnica de espalhamento de espectro é um método pelo qual um sinal é espalhado no domínio da frequência, de modo a ser transmitido em curtas rajadas, pulando entre frequências e proporcionando robustez ao sistema. A Figura 7 ilustra um exemplo de modulação utilizando pulsos *chirp*.

Figura 7 - Modulação utilizando pulsos *Chirp*



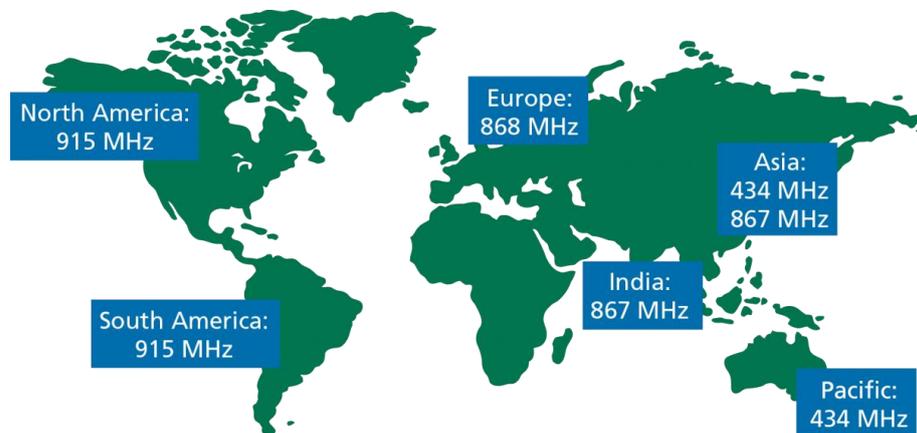
Fonte: Extraído de (GTA UFRJ, 2019)

À esquerda da Figura 7, observa-se um objeto em que a frequência aumenta com o tempo. À direita é possível observar o funcionamento do *Up-Chirp Down-Chirp*, em que a frequência aumenta com o tempo, variando entre a frequência mais baixa e frequência máxima que, quando alcançada, realiza o salto para a frequência mais baixa, recomeçando o processo, para o primeiro caso, e invertendo o processo para o segundo.

Logo, a base *chirp* cuja frequência inicia-se com a frequência mínima  $f_{min} = -\frac{BW}{2}$  e termina com a frequência máxima  $f_{max} = +\frac{BW}{2}$ , sendo BW definido como a largura de banda de espalhamento do sinal, e modulam *chirps* de base deslocados ciclicamente no tempo, é denominado *Up-Chirp*. O *chirp* que começa com a frequência  $f_1 = +\frac{BW}{2}$  e termina com  $f_2 = -\frac{BW}{2}$  refere-se ao *Down-Chirp*, e trata-se praticamente do conjugado complexo do *chirp* de base. Deste modo, os sinais *chirp* possuem amplitude constante, e variam a frequência ao longo do tempo de maneira linear, realizando a varredura de toda a largura de banda (HAXHIBEQIRI et. al., 2018).

A definição referente a comunicação entre dois pontos utilizando a tecnologia por radiofrequência *LoRa*® requer a definição de parâmetros, tais quais frequência de operação, fator de espalhamento (SF), largura de banda (BW), e taxa de código (CR). A Figura 8 ilustra as frequências de operação do rádio *LoRa*® para algumas regiões do planeta.

Figura 8 - Frequência de banda utilizada por dispositivos que empregam tecnologia LoRa® em diferentes locais do planeta



Fonte: Extraído de (GTA UFRJ, 2019)

No Brasil, o órgão regulamentador para frequências na faixa *ISM* (do inglês *Industrial Scientific and Medical*) é a ANATEL. Em 2017 o ato 14448 regulamentou a frequência de banda para o *LoRa®* entre 902 MHz e 928 MHz, com frequência típica de 915 MHz (TOZETTO; SILVA; FREIRE, 2021; ANATEL, 2017).

Embora exista a possibilidade de implementar projetos com o padrão de rede *LoRaWAN®*, protocolo de comunicação da camada superior da comunicação *LoRa®*, é possível elaborar projetos que utilizem a arquitetura ponto a ponto, sem necessidade de um intermediário. Este trabalho de conclusão de curso foi desenvolvido seguindo a arquitetura ponto a ponto.

Outro importante fator é o indicador de força do sinal recebido – *RSSI*-, que se refere a potência do sinal recebido em mW e é medido em dBm e trata-se de um valor negativo. O valor é usado para mensurar a qualidade do sinal recebido por um dispositivo. Quanto mais próximo de 0 é o valor do *RSSI*, melhor é o sinal (*LORA*, 2018).

## 2.5 Internet das Coisas

Introduzido em 1999, o termo *Internet of Things* foi proposto por Kevin Ashton em uma apresentação na *Protecter & Gamble*, se referindo a sistemas em que o mundo físico está

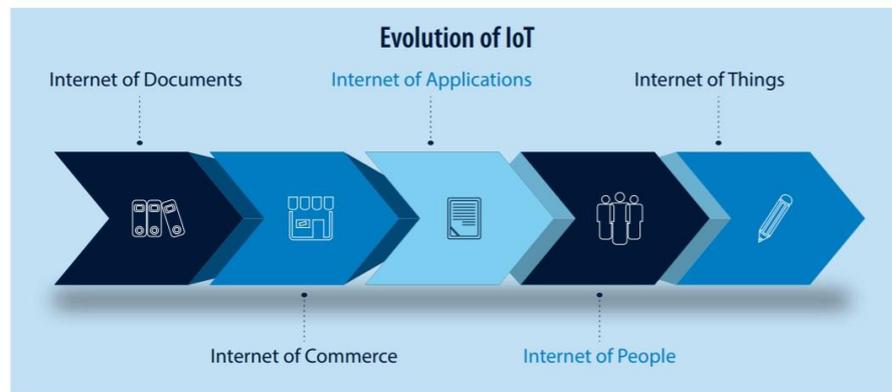
conectado à *Internet* por meio de sensores e tecnologia RFID (GOHKALE; BHAT; BHAT, 2018). Sendo assim, a *Internet das Coisas* se refere à extensão da *Internet* atual, conectando dispositivos à rede, modernizando e personalizando assim o acesso a diversos equipamentos.

Segundo Ashton (2009), a sociedade moderna é dependente de dados gerados por pessoas, que, uma vez limitadas no que tange à atenção e precisão, a captura de dados do mundo real pode ser prejudicada. Ainda de acordo com Kevin, o ambiente físico no qual vive o homem é baseado em coisas, e, a capacidade de criar dispositivos que coletem e analisem dados sem interferência humana, será possível reduzir custos, desperdícios e perdas, caracterizando a *Internet das Coisas* como potencial responsável de reorganizar o mundo.

Desta forma, à rede de objetos físicos interconectados pela rede, coletam e transmitem informações, oportunizando o controle com maior precisão e interação sem intervenção humana, aplicáveis residências, indústrias, saúde, automóveis, dentre outros, dá-se o nome de *Internet das Coisas* (GOHKALE; BHAT; BHAT, 2018).

Segundo Gohkale, Bhat e Bhat (2018), a evolução da *Internet* se dá sob 5 eras, tais quais evidenciadas pela Figura 9.

Figura 9 - Evolução da Internet



Fonte: Extraída de (AL REYAMI, 2020)

Deste modo, para os autores, a evolução da *Internet* passou pela era de bibliotecas eletrônicas e páginas *web* baseadas em documentos (*Internet of Documents*), ascensão do *e-commerce* e *sites* de negociações (*Internet of Commerce*), relação de aplicativos baseados em tecnologia da informação na *Web 2.0* (*Internet of Applications*), uso de redes sociais (*Internet of People*), e dispositivos e máquinas conectados (*Internet of Things*).

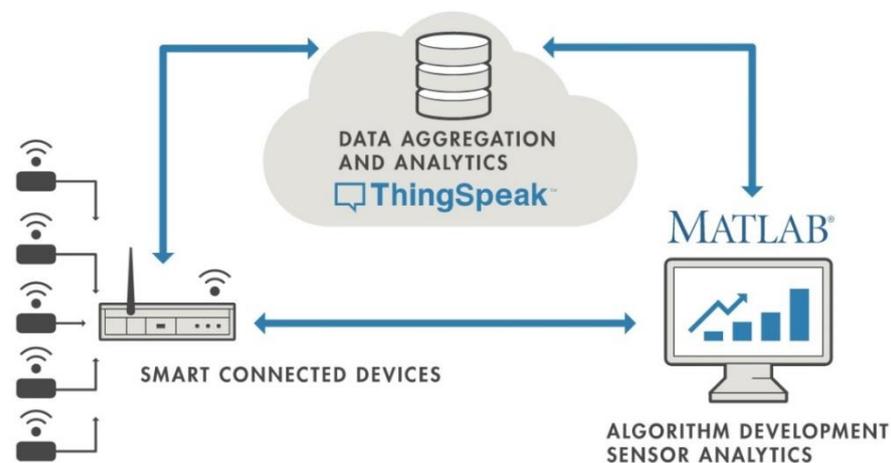
## 2.6 ThingSpeak™

Caracterizado como um *API* (do inglês *Application Programming Interface*) para serviços *IoT* com interface de código aberto, o *ThingSpeak™* é descrito como um *Middleware* para *Internet* das Coisas, possibilitando a construção de aplicações em torno de dados coletados (MAUREIRA; OLDENHOF, 2011).

Funcionando como plataforma de recebimento e envio de dados, o *ThingSpeak™* permite a visualização gráfica ao longo do tempo de valores numéricos enviados a partir de dispositivos eletrônicos, propiciando o monitoramento de tais grandezas, bem como análise e interação com os dados enviados à plataforma (*THINGSPEAK*, 2022).

O uso da versão gratuita da plataforma comporta recursos aceitáveis para implementações com menor quantidade de dados, visto que as informações recebidas são depositadas em canais, e cada canal permite o armazenamento de dados em até 8 campos. A plataforma dispõe recursos de análise de dados a partir do *MATLAB®*, com possibilidade de agendamento para execução de códigos, visualizações, ações sobre os dados (*MATHWORKS*, 2022). Devido às suas características, a aplicação do *ThingSpeak™* em soluções *IoT* vêm se destacando. Na Figura 10 é possível visualizar um diagrama de sistemas *IoT* que implementam suas aplicações na plataforma.

Figura 10 - Diagrama de aplicação *IoT*



Fonte: Extraída de (*THINGSPEAK*, 2022)

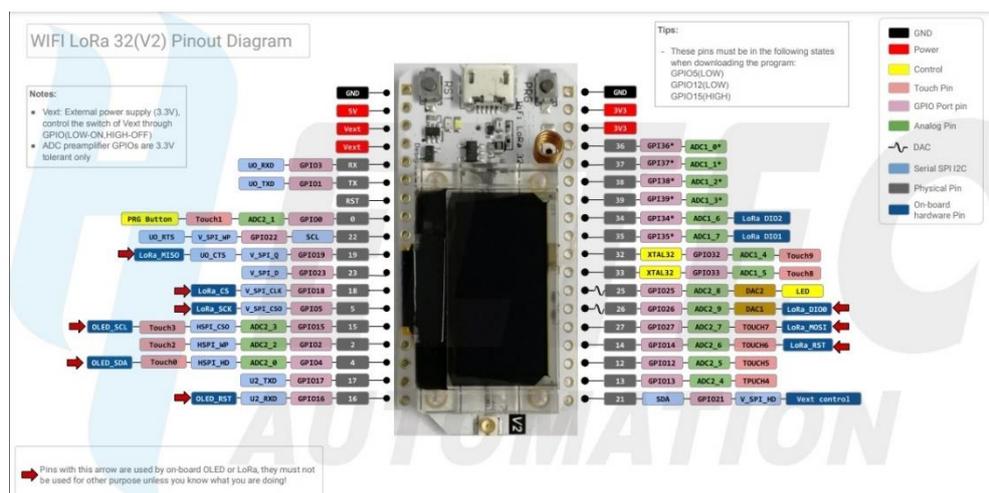
### 3 METODOLOGIA

Este t3pico aborda a metodologia empregada para implementa33o do prot3t3po proposto. Inicialmente, demonstra-se o m3dulo escolhido para compor o sistema, e, posteriormente, o princ3pio de funcionamento 3 exposto.

#### 3.1 Heltec ESP32 LoRa (V2)

Utilizaram-se duas placas *Heltec ESP32 LoRa (V2)* devido as suas caracter3sticas, citadas anteriormente. No m3dulo denominado “*LoRa Sender*” foram conectados dois sensores ultrass3nicos HC-SR04. O m3dulo denominado “*LoRa Receiver*” 3 respons3vel pela recep33o dos dados enviados pelo *LoRa Sender*, que se referem 3s medi33es provenientes dos sensores convertidas em porcentagem do n3vel de preenchimento de recipientes (reservat3rio e estoque). Na Figura 11 3 poss3vel visualizar a distribui33o de pinos (*pinout*) da placa *Heltec ESP32 LoRa (V2)*.

Figura 11 – Pinout da placa *Heltec ESP32 LoRa (V2)*



Fonte: Extra3da de (*HELTEC AUTOMATION*, 2022)

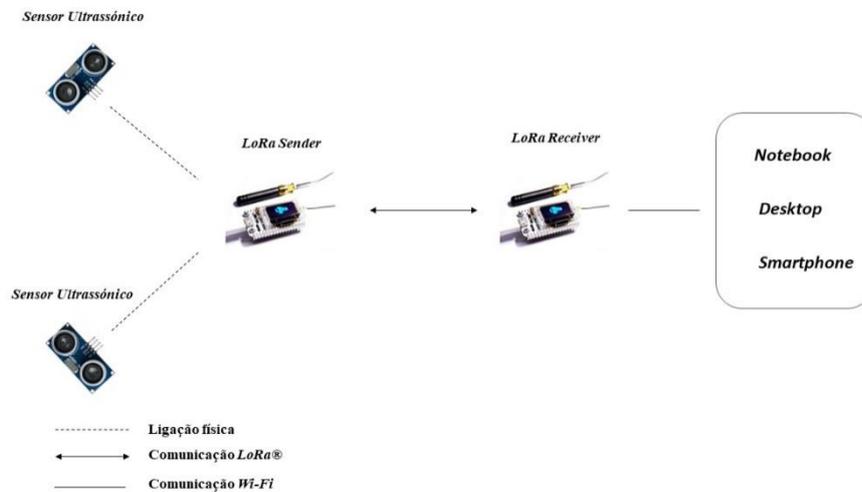
### 3.2 Arquitetura do projeto

A proposta de monitoramento de dados contempla a utilização de recipientes associados ao estoque e reservatório para uso contínuo de determinado insumo. Para a comunicação entre os dispositivos, foi utilizado a tecnologia de comunicação por radiofrequência *LoRa*®, enviando do dispositivo transmissor (*LoRa Sender*) para o dispositivo receptor (*LoRa Receiver*) os valores lidos pelos sensores. A fim de implementar um sistema baseado em *IoT*, o dispositivo denominado *Receiver* envia os dados recebidos para a plataforma *ThingSpeak*™. Como recursos adicionais para a visualização dos dados recebidos pela plataforma *IoT* foram utilizadas as seguintes ferramentas:

1. Aplicativo para *smartphones* *ThingView* – *ThingSpeak viewer*: permite a visualização dos canais vinculados à plataforma *IoT* de maneira rápida;
2. *IoT ThingSpeak Monitor Widget*: permite a visualização de valores contidos nos campos vinculados ao *ThingSpeak*™, e criação de alertas para valores escolhidos pelo usuário.

Para o funcionamento adequado do circuito, os microcontroladores foram programados na Arduino IDE com códigos apresentados no Anexo A – *Sender*, Anexo B – *Receiver*. O código referente ao cálculo de custo, efetuado em *MATLAB*® é apresentado no Anexo C. Além das conexões físicas implementadas, também é necessário estabelecer a conexão do dispositivo receptor via *Internet* para a plataforma *IoT* citada anteriormente. A Figura 12 ilustra o diagrama de conexões no protótipo final.

Figura 12 – Comunicação entre dispositivos



Fonte: Elaborada pela autora, 2022

### 3.3 Desenvolvimento do sistema

Esta seção apresenta o funcionamento do sistema proposto de maneira mais detalhada, explicando como foram programados os microcontroladores a fim de executar as tarefas sugeridas. As leituras dos sensores, o envio de dados entre os rádios *LoRa*®, e a comunicação para a plataforma *ThingSpeak*™, que retorna os dados enviados pelo dispositivo microcontrolado para uma *dashboard*, bem como as etapas referentes a configuração do aplicativo e *Widgets* para melhor visualização dos dados, são explanadas a seguir. Para a comunicação via rede sem fio *LoRa*®, há a necessidade de dois dispositivos que suportem a tecnologia. É possível programar as placas como transmissoras e receptoras, ou programá-las de maneira que um módulo transmita informações e outro apenas as receba. Neste trabalho cada placa desempenha uma função.

#### 3.3.1 Configuração do ambiente de programação

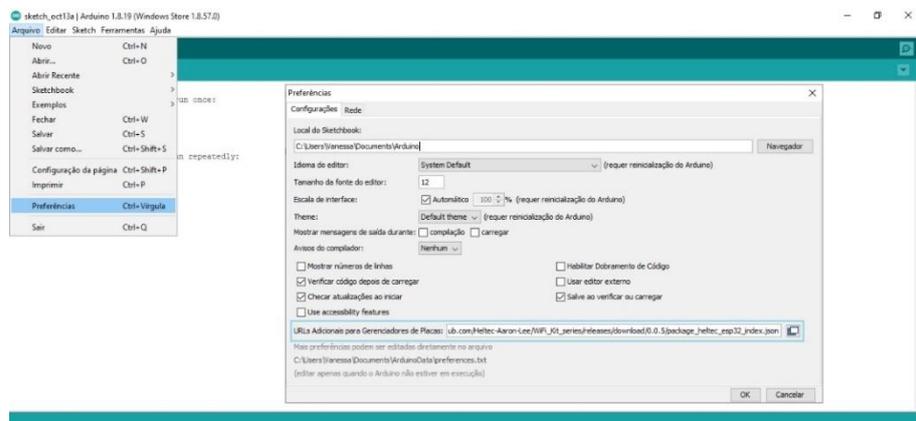
Atualmente estão disponíveis diferentes plataformas de desenvolvimento de aplicações para o ESP32, tais quais *Espruino*, que utiliza a linguagem *Java Script SDK*, o

framework ESP-IDF, *ESPlorer* IDE, que utiliza linguagem Lua e Arduino IDE, que emprega a linguagem C++.

Para o desenvolvimento deste projeto a plataforma escolhida foi a Arduino IDE. A plataforma possui compatibilidade com diversas placas, incluindo a placa *Heltec ESP32 LoRa* (V2). Para programá-la, deve-se instalar e configurar bibliotecas. Para isso, deve-se seguir os seguintes passos:

- a. Ir até Arquivos -> preferências -> URL's adicionais para Gerenciadores de Placas, e adicionar a seguinte URL: [https://github.com/Heltec-Aaron-Lee/WiFi\\_Kit\\_series/releases/download/0.0.5/package\\_heltec\\_esp32\\_index.json](https://github.com/Heltec-Aaron-Lee/WiFi_Kit_series/releases/download/0.0.5/package_heltec_esp32_index.json), conforme ilustrado pela Figura 13.

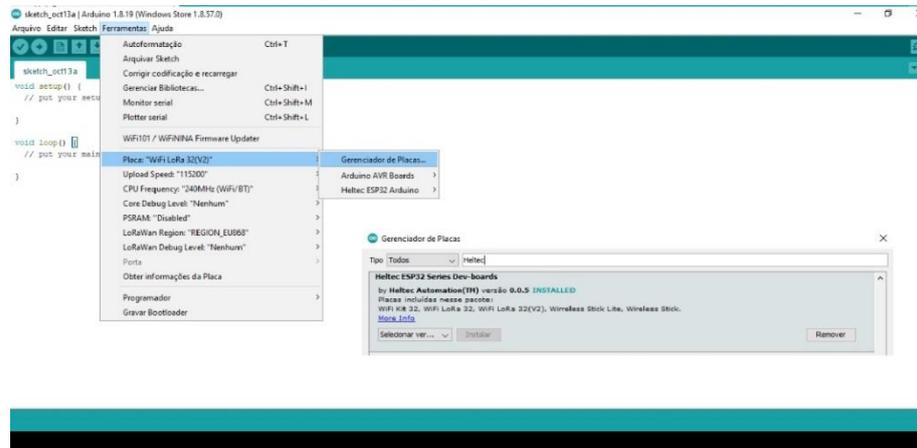
Figura 13 – Definição de preferências



Fonte: Extraída pela autora, 2022

- b. Em seguida deve-se ir até Ferramentas -> Placas -> Gerenciador de Placas, procurar pela placa *Heltec ESP32 LoRa* e realizar a instalação, conforme exposto pela Figura 14.

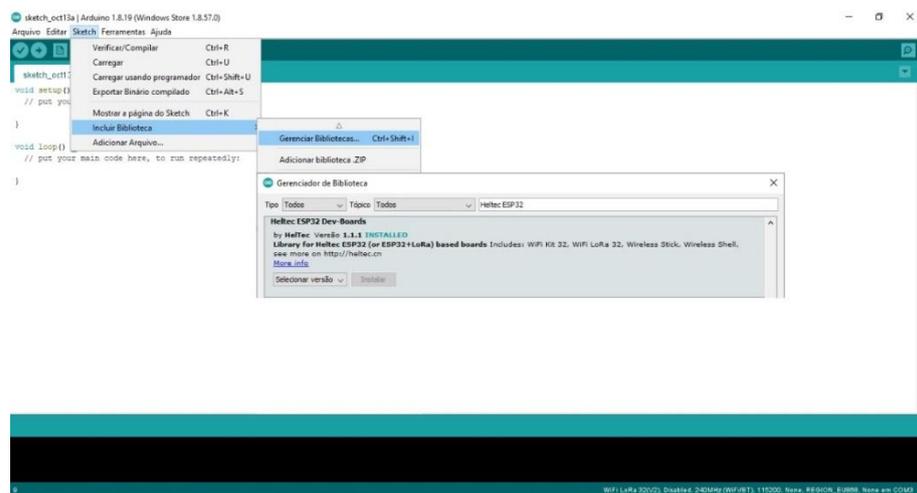
Figura 14 - Instalação do pacote *Heltec ESP32 Series Dev-boards*



Fonte: Extraída pela autora, 2022

- c. Posteriormente, deve-se ir até *Sketch* -> Incluir Biblioteca -> Gerenciador de Biblioteca, procurar pela biblioteca da *Heltec* ESP32, e realizar a instalação, conforme visto pela Figura 15.

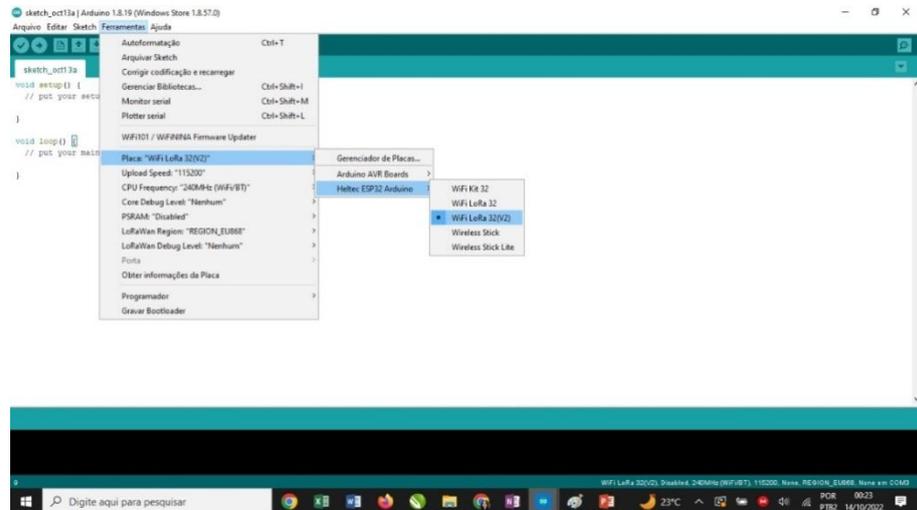
Figura 15 - Instalação da biblioteca *Heltec* ESP32



Fonte: Extraída pela autora, 2022

- d. Por fim, para iniciar a programação deve-se escolher a placa *Heltec* ESP32 *LoRa* (V2), conforme ilustrado pela Figura 16.

Figura 16 - Seleção da placa *Heltec ESP32 LoRa (V2)*



Fonte: Extraída pela autora, 2022

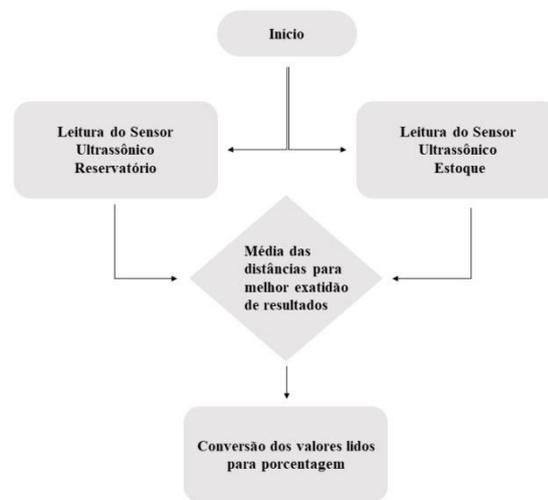
Caso o computador não reconheça a placa automaticamente, deve-se instalar o *drive* CP10xUSB para *UART Bridge Virtual COM Port (VCP)* correspondente ao sistema operacional utilizado pelo usuário. O *download* pode ser feito pelo *link*: <https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=overview>.

Após o *download* deve-se descompactar o conteúdo contido no arquivo, e selecionar a opção *32 bits* ou *64 bits*, conforme a correspondência do sistema. Feito isto, deve-se executar o instalador como administrador, completando a instalação do *drive* e assegurando, desta forma, a comunicação com a placa.

### 3.3.2 Leitura de dados referentes ao reservatório e estoque

Para o sistema de leitura de dados referentes à capacidade do reservatório e capacidade do estoque, foram utilizados dois sensores ultrassônicos HC-SR04, com alimentação em 5V. Um divisor de tensão foi alocado ao sistema, a fim de fornecer 3,3V ao pino do módulo, evitando assim possíveis danificações, conforme o esquemático que pode ser visualizado na Figura 25, na seção 4.1 deste trabalho. Os valores lidos referem-se às distâncias, posteriormente convertidos em unidade de porcentagem para melhor aproximação em uma aplicação real. O fluxograma desta aplicação pode ser visto pela Figura 17.

Figura 17 - Fluxograma do funcionamento do sistema de leitura de dados

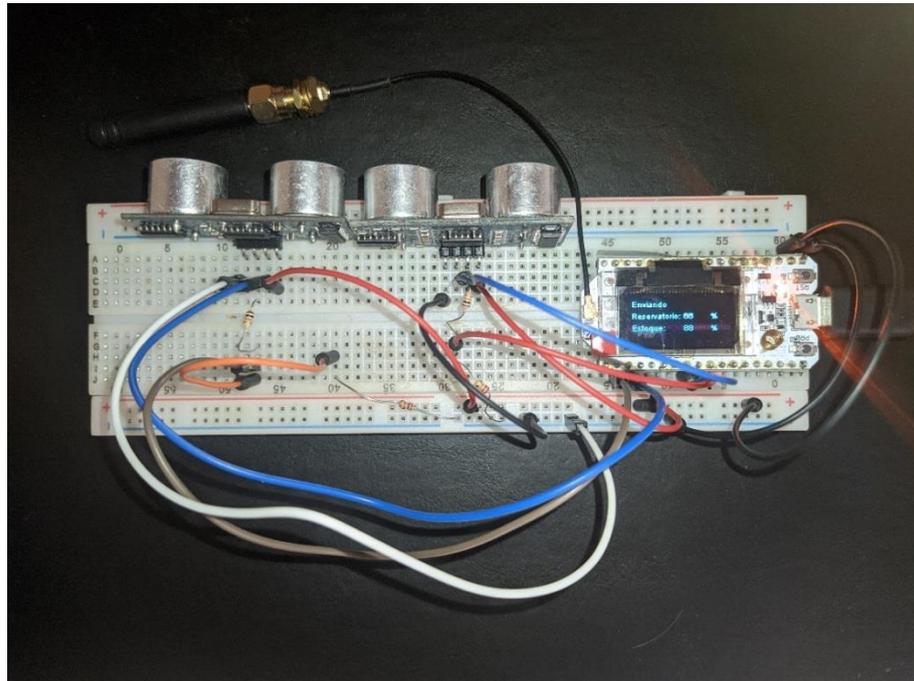


Fonte: Elaborada pela autora, 2022

### 3.3.3 LoRa Sender

O dispositivo responsável por transmitir os dados lidos pelos sensores é denominado *LoRa Sender*. Para aplicação de controle e monitoramento de insumos, as bibliotecas “*heltec.h*” e <*Ultrasonic.h*> são incluídas a fim de possibilitar a execução de instruções relacionadas a tarefas específicas. Deste modo, os sensores ultrassônicos são conectados ao microcontrolador, a fim de realizar as leituras pertinentes ao projeto. Assim, ao habilitar a placa que contém funções que suportam a comunicação por radiofrequência, torna-se possível o envio dos dados síncronos, desde que o dispositivo para recepção de dados esteja habilitado. É possível visualizar tais dados através do *display Oled* contido no módulo. A Figura 18 ilustra este funcionamento.

Figura 18 – *LoRa Sender* em funcionamento



Fonte: Elaborada pela autora, 2022

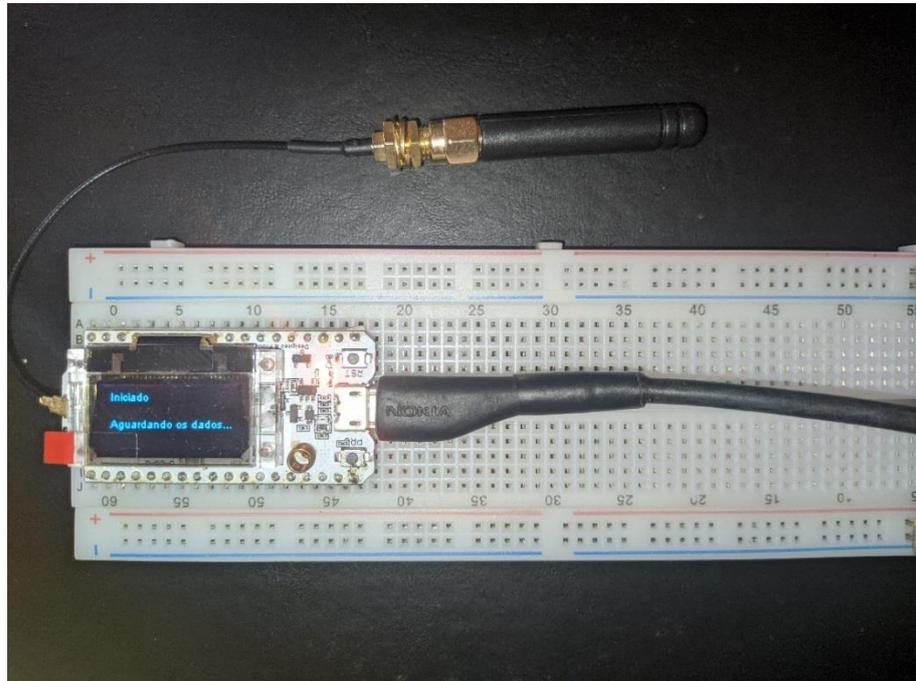
### 3.3.4 *LoRa Receiver*

O dispositivo responsável por receber os dados referentes ao reservatório e ao estoque é denominado *LoRa Receiver*. Para aplicação de controle e monitoramento de insumos as bibliotecas “*heltec.h*”, <*Wi-Fi.h*> e <*Thingspeak.h*> são definidas. Os dados recebidos do dispositivo *Sender* podem ser visualizados pelo *display Oled* contido na placa.

A fim de garantir que o envio e recepção de dados aconteçam pelos dois módulos de interesse, define-se uma palavra de sincronização, evitando assim que pacotes indesejáveis de dados sejam recebidos pelo receptor, uma vez que quaisquer dispositivos *LoRa®* que estejam em operação dentro de seu alcance podem interferir a recepção de dados. Esta palavra está contida nos códigos do transmissor e do receptor, e varia de 0 a 0xFF, funcionando sobre o dispositivo receptor a fim de que este ignore quaisquer pacotes que não contenha essa palavra.

O dispositivo é inicializado, e aguarda a efetivação da comunicação *LoRa®*, exibindo a mensagem que pode ser vista pela Figura 19.

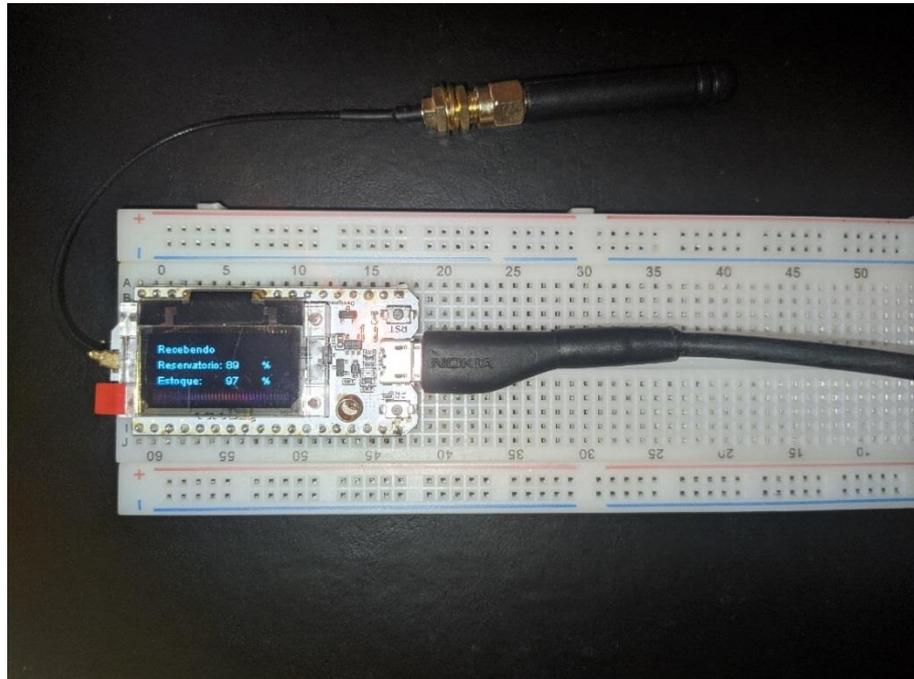
Figura 19 – LoRa Receiver aguardando comunicação



Fonte: Elaborada pela autora, 2022

Deste modo, assim que a comunicação via rádio é estabelecida, o dispositivo *Receiver* inicia a recepção dos dados, conforme visualizado pela Figura 20.

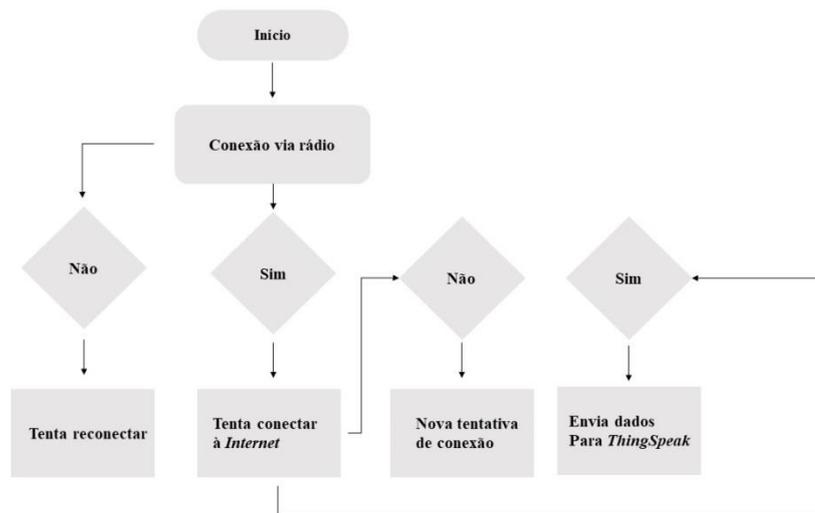
Figura 20 - Dispositivo *LoRa Receiver* recebendo dados



Fonte: Elaborada pela autora, 2022

Uma vez que a comunicação via rádio é bem sucedida, o dispositivo inicia a tentativa de conexão com a rede de *Internet* via *Wi-Fi*. Quando conectado à rede, o dispositivo dá início ao envio dos dados recebidos pelo dispositivo *Sender* para a plataforma *IoT ThingSpeak™*, apresentando os dados através de uma *Dashboard*. O funcionamento do sistema pode ser visto pela Figura 21.

Figura 21 - Fluxograma do funcionamento do dispositivo *Receiver*



Fonte: Elaborada pela autora, 2022

### 3.3.5 Aquisição de dados Pelo *ThingSpeak*<sup>TM</sup>

Para esta aplicação, necessita-se que os dados sejam enviados pelo dispositivo *Receiver* para a plataforma *IoT*. O sistema foi configurado para enviar informações ao *ThingSpeak*<sup>TM</sup> a cada 2 minutos, obedecendo a quantidade total de dados diários da plataforma, de 8 mil dados. Deste modo, a plataforma fornece a visualização gráfica instantânea das informações, possibilitando a realização de análises de dados, bem como alocação de outros recursos através do campo *App*.

Para que seja possível conectar-se ao *ThingSpeak*<sup>TM</sup> é necessário realizar um cadastro no *site*, e, posteriormente ativar o *login* por intermédio do *link* recebido por *e-mail*. Em seguida, deve-se criar um novo canal, e configurá-lo por meio do campo *Channel Settings*. Caso utilizado na versão gratuita, o programa oferta a possibilidade de envio e visualização de dados em 8 campos distintos para cada canal, com possibilidade de torná-los públicos, privados ou compartilhados com uma lista de usuários específicos.

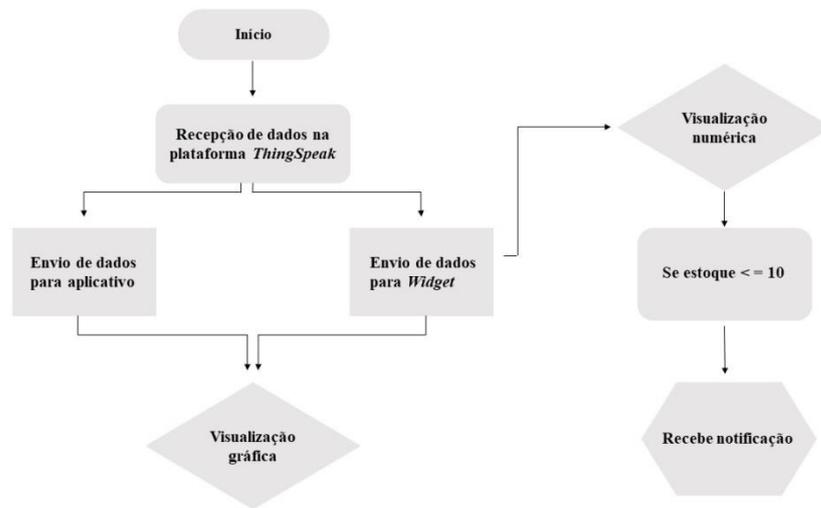
Para que seja possível publicar no *ThingSpeak*<sup>TM</sup>, a plataforma disponibiliza um número de identificação único (*Channel ID*), chaves *API* de escrita (*Write API Key*) e leitura (*Read API Key*), e chaves para envio e requisição de dados (*API Requests*), elementos essenciais para movimentação de dados. Há suporte para publicações com os protocolos de comunicação *HTTP* (do inglês *Hypertext Transfer Protocol*) e *MQTT* (do inglês *Message*

Queuing Telemetry Transport). Para sua utilização de maneira mais simples, pode-se empregar a biblioteca <*Thingspeak.h*> no Arduino IDE, importando automaticamente todos os recursos necessários para aplicações na plataforma.

Neste trabalho foram utilizados dois recursos adicionais vinculados à plataforma: o aplicativo *ThingView – ThingSpeakviewer* e o *Widget IoT ThingSpeak Monitor Widget*. O aplicativo foi instalado em dispositivo *Android*, e para viabilizar a visualização de dados, deve-se inserir o número de identificação do canal para canais públicos, e, para canais privados deve-se adicionar a chave de leitura nas configurações. O aplicativo foi escolhido devido a possibilidade de visualizar informações do canal configurado sem que haja necessidade de fazer *login*, tornando de fácil acesso o monitoramento das informações.

De forma similar, para usufruir dos recursos do *Widget*, deve-se inserir o número de identificação do canal e chave de leitura. Para cada implementação do recurso, é possível visualizar valores numéricos e gráficos de até dois campos. A interface permite criar notificações referentes aos valores apresentados, exibindo mensagem de alerta e vibração do aparelho vinculado até que a notificação de mensagem seja devidamente lida pelo usuário. Neste trabalho, definiu-se um alerta para limite inferior em relação a capacidade do estoque. Para tal, o valor definido varia entre igual ou menor a 10 unidades (em percentual), com notificação insistente a cada 30 minutos, permanecendo neste estado até que haja mudança superior na capacidade do estoque. A Figura 22 ilustra o funcionamento deste sistema.

Figura 22 - Fluxograma de sistema de aquisição de dados pelo *ThingSpeak™*, aplicativo e *Widget*



Fonte: Elaborada pela autora, 2022

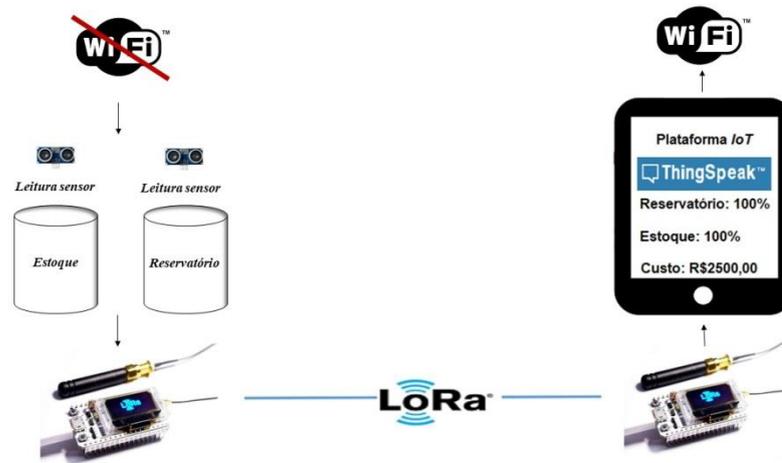
### 3.4 Considerações finais sobre o sistema proposto

A proposta do sistema consiste no monitoramento de insumos com auxílio de prototipagem eletrônica. Embora a comunicação via rádio possa ser executada em ambientes conectados à rede de *Internet*, tais quais ambientes hospitalares, estações de controle meteorológico, indústrias, residências, dentre outros, a proposta deste trabalho visa sua aplicação em locais de difícil acesso à *Internet*.

Deste modo, em uma aplicação real, o dispositivo *Sender* seria instalado em um local remoto, coletando dados a partir de sensores e enviando-os através da comunicação *LoRa®* para o dispositivo *Receiver*, instalado em local com acesso à rede, a fim de enviar via *Wi-Fi* os dados para a plataforma *IoT*. O sistema de alerta notifica o usuário sobre a necessidade de reposição do estoque, e através de recursos utilizados no *ThingSpeak™* o consumidor pode estimar os custos temporalmente.

A Figura 23 ilustra o princípio de funcionamento do sistema proposto.

Figura 23 - Proposta do sistema



Fonte: Elaborada pela autora, 2022

## 4 RESULTADOS E DISCUSSÕES

Nesta seção serão apresentados os resultados obtidos com a estruturação do protótipo, que pode ser visto na Figura 24. Na Figura é possível observar como foram organizados os sensores e microcontroladores. O funcionamento geral do sistema foi satisfatório, viabilizando uma possível aplicação real com os mesmos princípios deste sistema.

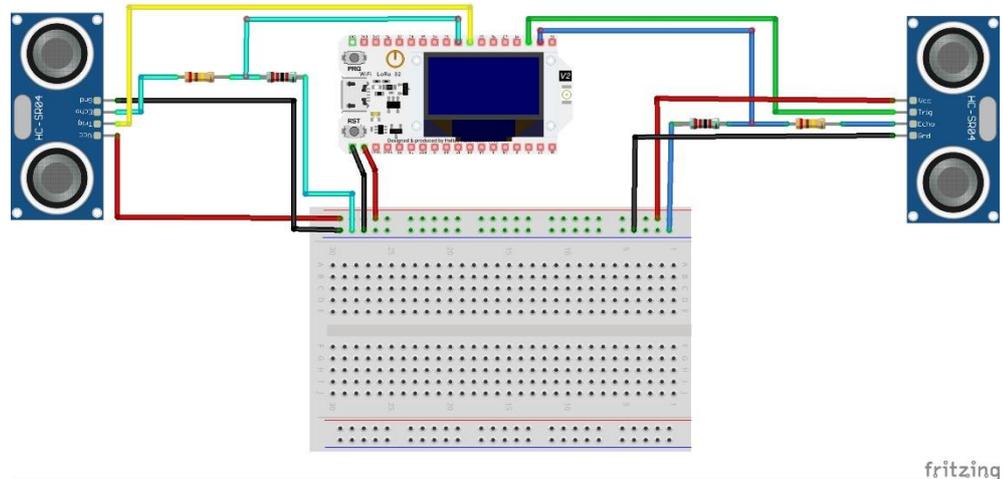
Figura 24 - Protótipo final



Fonte: Elaborada pela própria autora, 2022

### 4.1 LoRa Sender

A conexão física dos componentes eletrônicos constituintes do circuito do dispositivo transmissor foi realizado conforme esquemático desenvolvido pelo *software Fritzing*, conforme demonstrado pela Figura 25.

Figura 25 - Esquemático *LoRa Sender*

fritzing

Fonte: Elaborada pela autora, 2022

## 4.2 *LoRa Receiver*

Ao módulo receptor nenhum componente foi conectado, haja vista ser necessário apenas que este seja alimentado para o funcionamento deste projeto. Como exposto anteriormente, esta placa é responsável por receber os dados via comunicação sem fio *LoRa*®, e conectar-se à *Internet* a fim de enviar os dados recebidos para a plataforma de *Internet* das Coisas. É possível visualizar pelo serial do Arduino IDE o instante em que a comunicação via rádio é efetivada, conforme ilustra a Figura 26.

Figura 26 – Saída Serial Arduino IDE - Comunicação via rádio inicializada



Fonte: Elaborada pela autora, 2022

De forma similar, pela Serial Arduino é possível visualizar o momento em que o módulo se conecta à rede via *Wi-Fi*, bem como a intensidade do sinal e valores referentes ao reservatório e ao estoque, conforme a Figura 27.

Figura 27 - Saída Serial Arduino IDE - Conexão com a rede

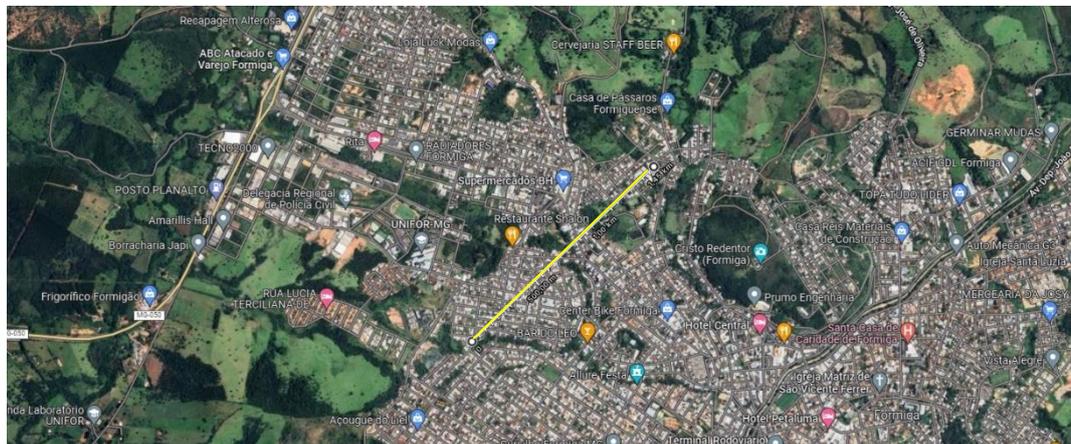


Fonte: Elaborada pela autora, 2022

### 4.3 Teste de alcance *LoRa*®

Os testes de alcance *LoRa*® foram realizados em ambiente urbano, na cidade de Formiga/MG. Em um teste inicial, o dispositivo *LoRa Sender* foi alocado próximo a uma janela dentro de uma residência, enquanto o dispositivo *LoRa Receiver*, alimentado por *Powerbank*, permaneceu em movimento, cursando o percurso entre a residência e o Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – *Campus Formiga*, como é apresentado pela Figura 28.

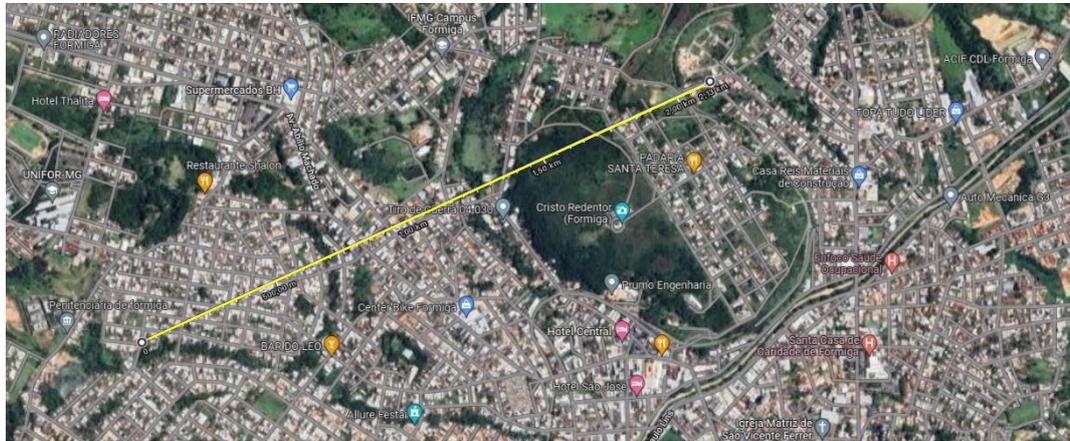
Figura 28 - Trajeto percorrido - Teste 1



Fonte: Extraída de (GOOGLE MAPS, 2022)

Para o segundo teste de alcance, o dispositivo *LoRa Sender* foi alocado no mesmo ponto do teste anterior, enquanto o módulo *Receiver*, alimentado por *Powerbank*, percorreu um trajeto entre a residência e determinado ponto da Rua João Balduíno, Formiga/MG. É possível visualizar o trajeto através da Figura 29.

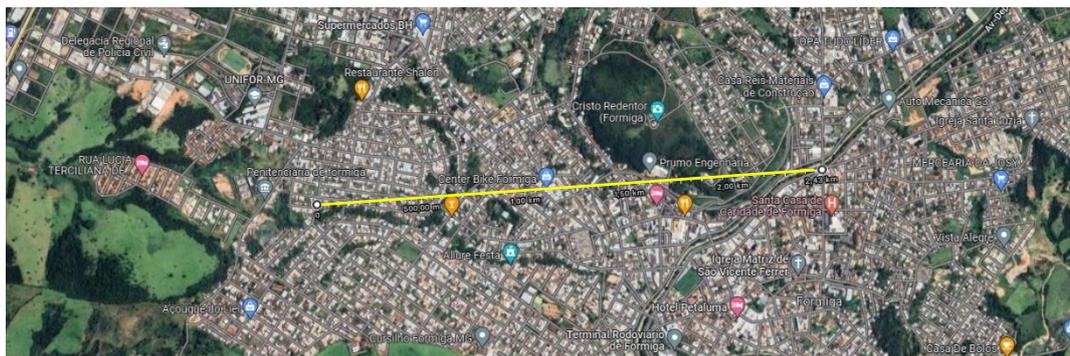
Figura 29 - Trajeto percorrido - Teste 2



Fonte: Extraída de (GOOGLE MAPS, 2022)

Para realização do terceiro teste, o dispositivo transmissor foi alocado na área externa da residência, enquanto o dispositivo *Sender*, alimentado pela porta *USB* de um *notebook*, cursou um caminho desde a residência, até determinado ponto da Avenida Rio Branco. Neste ponto a comunicação foi perdida e o valor *RSSI* era de -126 dBm. A Figura 30 ilustra o trajeto percorrido.

Figura 30 - Trajeto percorrido - Teste 3



Fonte: Extraída de (GOOGLE MAPS, 2022)

A Tabela 5 demonstra as distâncias alcançadas na comunicação via rádio para os testes realizados.

Tabela 5 - Resultados obtidos em testes de alcance

Testes de distância			
	Teste 1	Teste 2	Teste 3
<b>Distância (Km)</b>	1.45	2.13	2.43

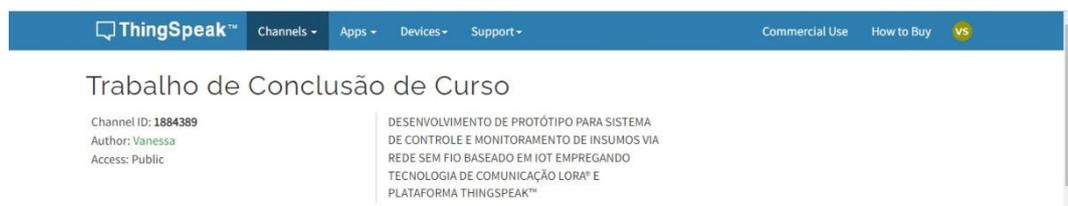
Fonte: Elaborada pela autora, 2022

Visto que os testes de distância realizados aconteceram em ambiente urbano todos os percursos apresentavam obstáculos, tais quais: grande volume de residências, estabelecimentos comerciais e veículos. Sabe-se que a presença de barreiras interfere diretamente a capacidade de transmitir e receber dados a determinadas distâncias, por se tratar de ondas eletromagnéticas que são influenciadas pelo meio.

Desta forma, a distância de comunicação *LoRa*® em meio rural geralmente é maior devido à ausência de obstáculos comumente presentes em ambientes urbanos. Entretanto, deve-se avaliar a aplicação da tecnologia em regiões rurais com características de relevo diferentes, a fim de avaliar o desempenho da comunicação via radiofrequência nestes locais.

#### 4.4 Plataforma *IoT- ThingSpeak*™

A fim de disponibilizar a visualização de dados para o usuário, a plataforma *IoT ThingSpeak*™ foi utilizada. Ao adicionar os tópicos referentes aos sensores, consumo e cálculo de custos, foi possível monitorar o sistema através da *dashboard* do recurso. A Figura 31 ilustra a interface inicial na plataforma.

Figura 31 - Interface inicial em plataforma *IoT*

Fonte: Elaborada pela autora, 2022

Foram configurados 4 campos (*Fields*) dentro do canal, sendo estes:

- a. *Field 1*: Capacidade do Reservatório (%);
- b. *Field 2*: Capacidade do Estoque (%);
- c. *Field 3*: Consumo (%);
- d. *Field 4*: Custo (R\$);

Os campos foram configurados para exibir os últimos 6 valores no formato de gráficos de colunas, para melhor visualização das informações. É possível a verificação numérica de dados ao selecionar a coluna desejada.

O Campo 1 recebe do dispositivo receptor os dados referentes a capacidade em porcentagem do reservatório. Em uma aplicação real, o reservatório é destinado ao espaço onde o insumo é disponibilizado ao usufrutuário final. Deste modo, seu nível de preenchimento se altera em função do consumo, havendo decréscimo no nível na medida em que o insumo é utilizado, e acréscimo quando o recipiente destinado ao reservatório é preenchido pelo recipiente definido como estoque. A Figura 32 ilustra a publicação gráfica dos dados ao longo tempo.

Figura 32 - Visualização gráfica –Variação da Capacidade do Reservatório

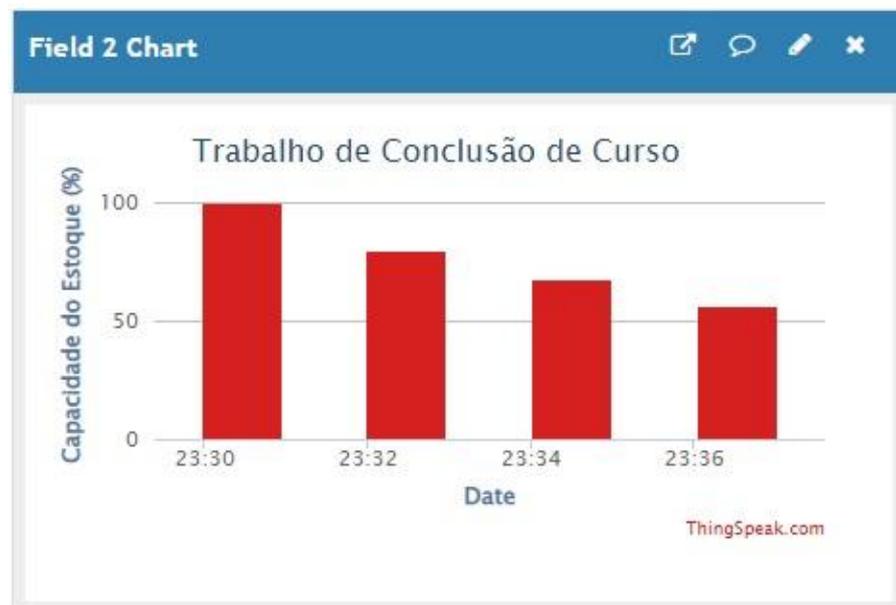


Fonte: Elaborada pela autora, 2022

O Campo 2 recebe do dispositivo receptor dados referentes a capacidade em

porcentagem do estoque. Sua capacidade é diminuída gradativamente na medida em que há demanda no reservatório. Neste trabalho definiu-se como alerta de necessidade de reposição de estoque quando a porcentagem atingida é menor ou igual a 10%. A Figura 33 ilustra a publicação gráfica dos dados ao longo do tempo, viabilizando a visualização do esvaziamento gradativo do espaço destinado ao estoque de insumos.

Figura 33 - Visualização gráfica - Decréscimo da Capacidade do Estoque



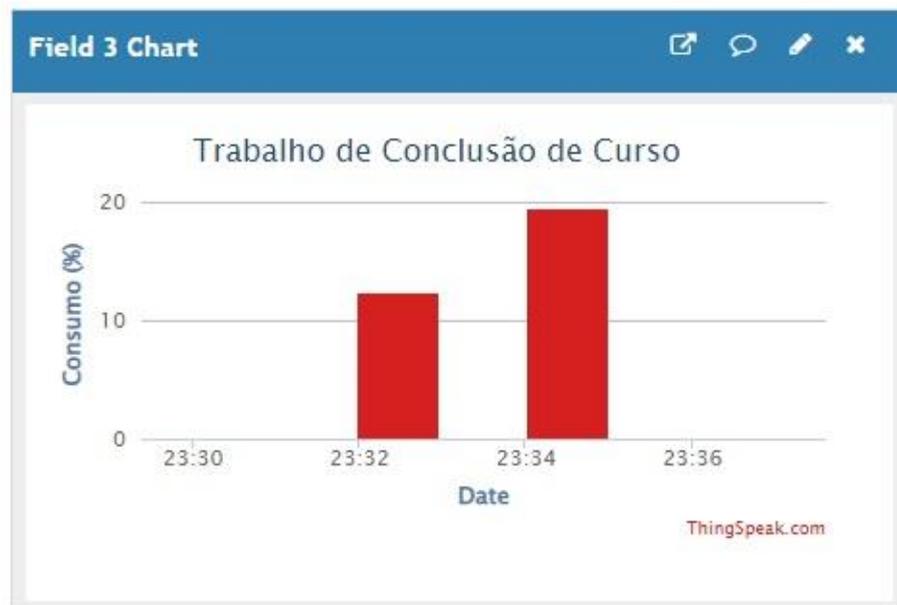
Fonte: Elaborada pela autora, 2022

O Campo 3 exibe informações enviadas pelo *LoRa Receiver* referentes ao consumo do estoque em porcentagem, conforme verificado pela Figura 34.

O cálculo de consumo considera dados relacionados ao nível percentual atual de estoque e nível percentual anterior de estoque, subtraindo o valor lido do estoque anterior ao valor lido do estoque atual.

Desta forma, se há alteração nos dados referentes ao nível de estoque, o cálculo é efetuado e a informação é enviada à plataforma. Se não há alteração do nível percentual do estoque, o microcontrolador envia ao *ThingSpeak™* o valor 0.

Figura 34 - Visualização gráfica - Consumo do estoque em porcentagem



Fonte: Elaborada pela autora, 2022

O Campo 4 recebe dados da ferramenta *MATLAB Analysis* disposta como recurso de análise de dados no *ThingSpeak™*. A Figura 35 demonstra a visualização gráfica deste campo, que se refere à estimativa de custo associado ao consumo do insumo, resultado da multiplicação do Campo 3 por um preço configurável. O valor configurável refere-se a um valor fictício com objetivo de simular os custos, que pode ser ajustado conforme o insumo empregado.

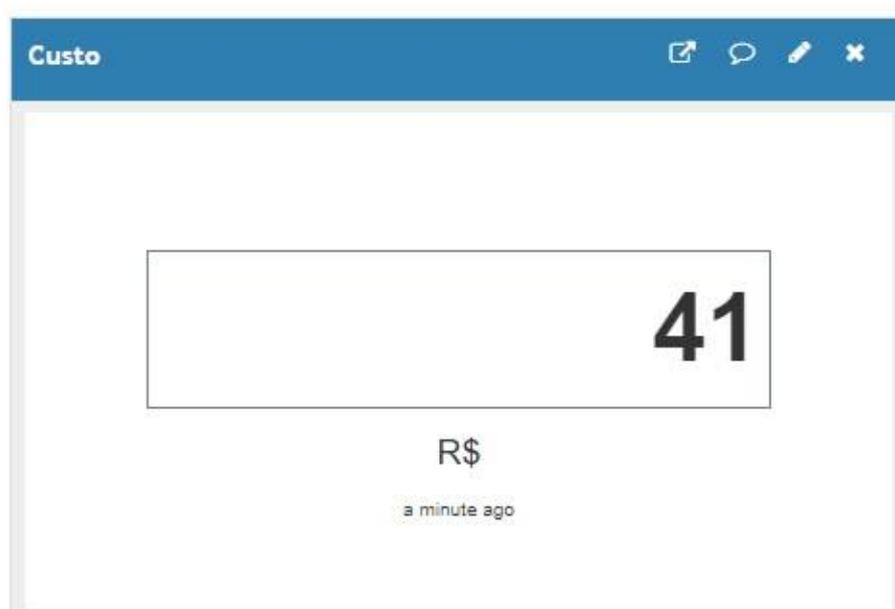
Figura 35 - Visualização gráfica – Custo



Fonte: Elaborada pela autora, 2022

Para otimizar a visualização do valor de custo, um *Widget* disponível na plataforma *IoT* foi vinculado ao Campo 4, ilustrando numericamente o valor outrora calculado, conforme demonstra a Figura 36.

Figura 36 - Visualização numérica do custo



Fonte: Elaborada pela autora, 2022

#### 4.4.1 Análise de dados

Uma das ferramentas disponíveis no *ThingSpeak*<sup>TM</sup> trata-se de um aplicativo para preparo, filtragem e análises de dados, o *MATLAB Analysis*. O instrumento foi utilizado neste trabalho com objetivo de calcular os custos do usuário com os insumos. O código foi implementado no espaço denominado *MATLAB Code*, e, para possibilitar a leitura e escrita de dados em um mesmo canal, o mesmo foi colocado em visualização pública.

Deste modo, para que seja possível analisar os dados contidos no canal, deve-se utilizar a identificação numérica deste, a chave de leitura *Read API Key* e chave de escrita *Write API Key*. Ao utilizar-se a função “*thingSpeakRead (ChannelID, 'ReadKey', readAPIKey, 'Field', 3)*” foi possível realizar a leitura de dados contidos no campo 3. O cálculo de custos foi efetivado, e através da função “*thingSpeakWrite (writeChannelID, custo, 'WriteKey', writeAPIKey, 'Field', 4)*” o resultado foi publicado no Campo 4.

Visto que é de interesse do usuário ter ciência de seus gastos, a ferramenta *React* foi utilizada, com objetivo de executar o código sob condições específicas. A reação foi configurada para ser executada a cada inserção de dados com valores maiores do que 0 no Campo 3 – Consumo (%) -, executando o código do *MATLAB Analysis* sempre que a condição é satisfeita. A Figura 37 demonstra as configurações do *React*.

Figura 37 - Configurações do recurso *React*

Apps / React / Reação custos TCC / Edit

**React Name**

**Condition Type**

**Test Frequency**

**Condition** If channel

field

**Action**

Code to execute

**Options**  Run action only the first time the condition is met  
 Run action each time condition is met

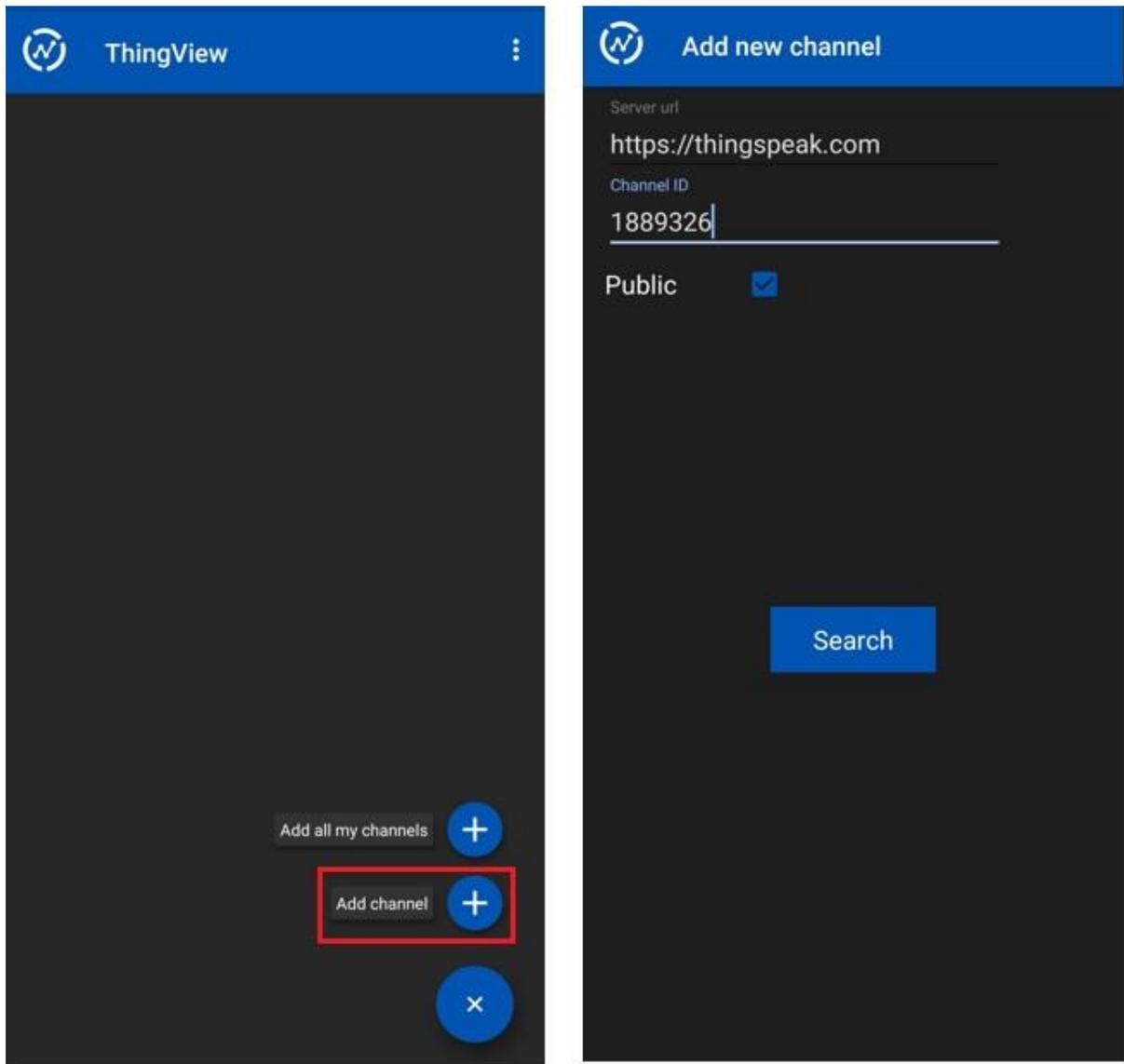
Fonte: Elaborada pela autora, 2022

#### 4.5 Dashboard aplicativo *ThingView* – *ThingSpeak viewer*

Outra forma de interface com o usuário foi empregada a partirdo aplicativo para *smarthphones Android ThingView – ThingSpeak viewer*. O aplicativo é uma ferramenta prática de visualização de dados publicados da plataforma *ThingSpeak™*, visto que após a

inserção dos dados necessários para usufruir seus recursos, não é necessário realizar um novo *login*. Para tornar possível a visualização dos dados do canal, deve-se adicionar um novo canal (indicado em vermelho) e preencher o número de identificação deste, conforme ilustra a Figura 38.

Figura 38 - Preenchimento de dados do canal



Fonte: Elaborada pela autora, 2022

Posteriormente, uma tela para conferência de dados relacionados ao canal selecionado será exibida, conforme exibido pela Figura 39.

Figura 39 - Tela de confirmação de dados do canal

**Confirm channel**

**Channel**  
Trabalho de Conclusão de Curso

**Description**  
DESENVOLVIMENTO DE PROTÓTIPO  
PARA MONITORAMENTO DE INSUMOS  
VIA REDE SEM FIO COM TECNOLOGIA DE  
COMUNICAÇÃO LORA® E PLATAFORMA  
THINGSPEAK™

**Fields**

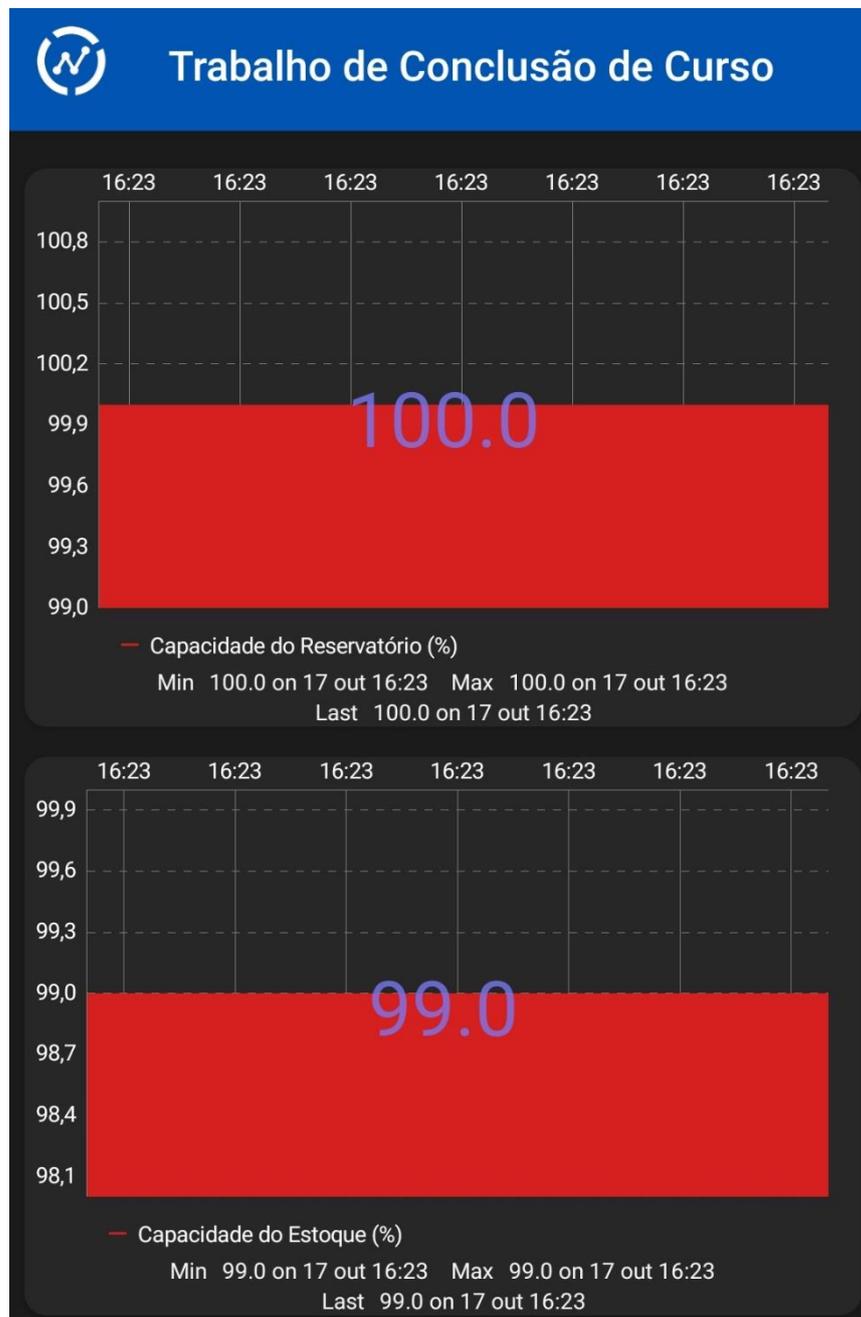
- Capacidade do Reservatório (%)
- Capacidade do Estoque (%)
- Consumo (%)
- Custo (R\$)

**Done**

Fonte: Elaborada pela autora, 2022

Desta forma, é possível visualizar gráfica e numericamente as informações contidas no canal. A Figura 40 exibe informações referentes ao reservatório e ao estoque.

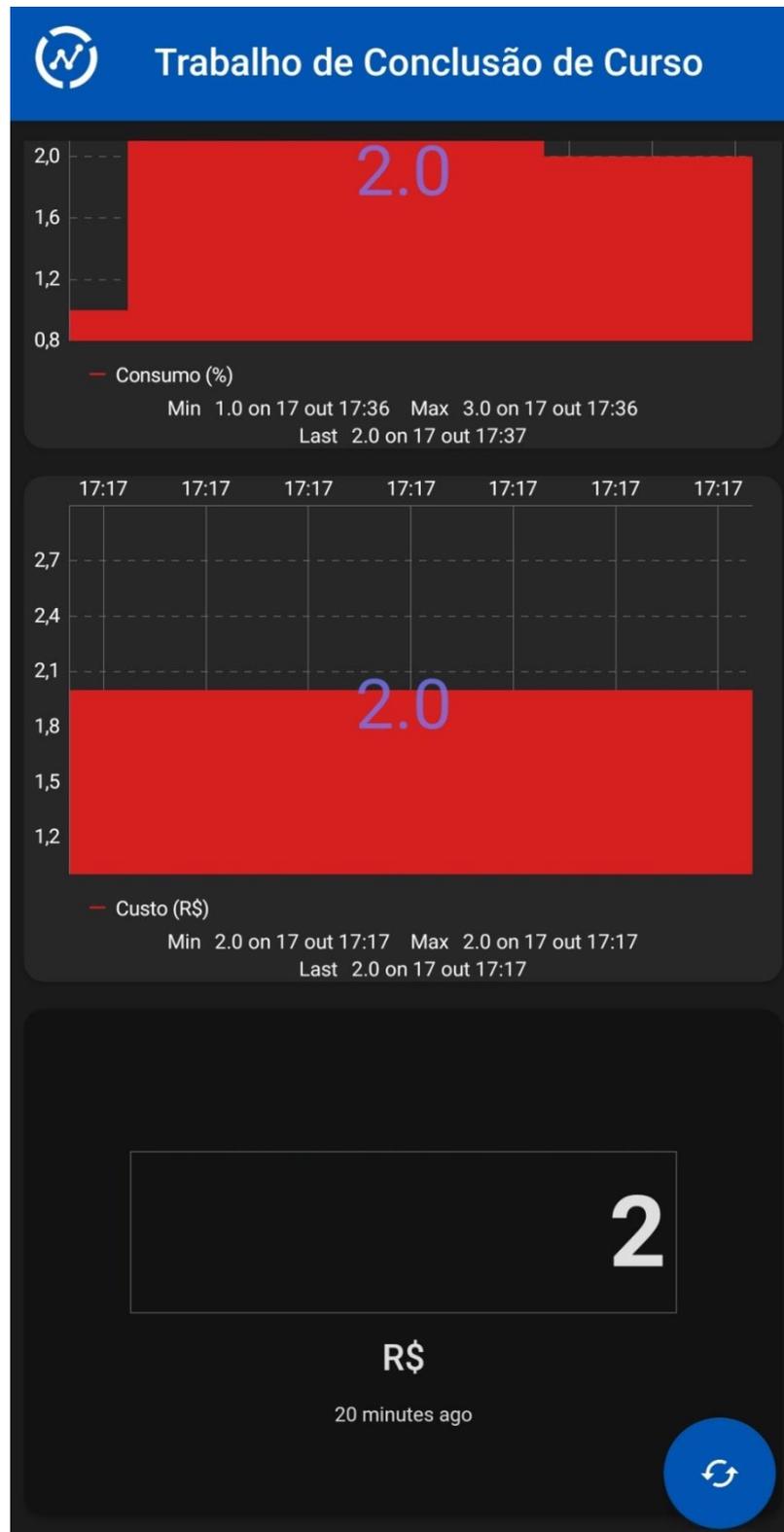
Figura 40 - Informações de reservatório e estoque



Fonte: Elaborada pela autora, 2022

A Figura 41 exibe informações relacionadas ao consumo e custos.

Figura 41 - Informações de consumo e custo



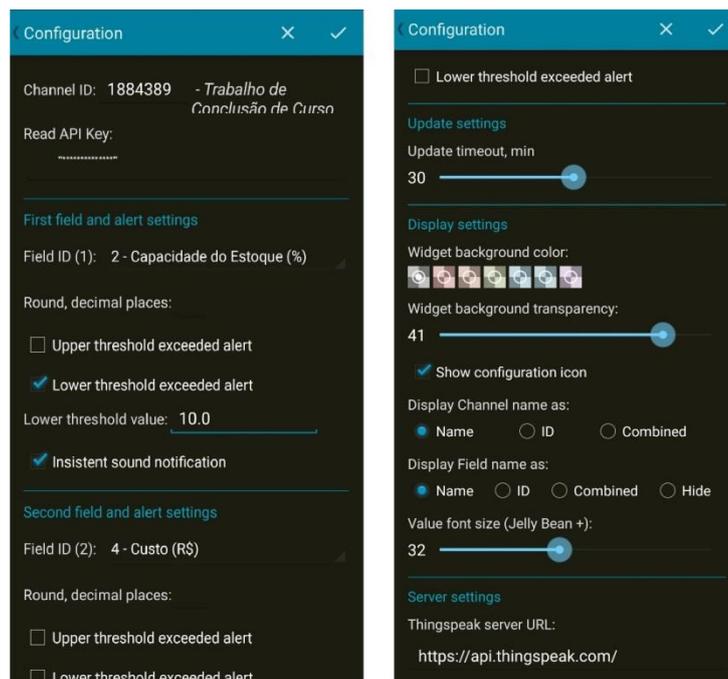
Fonte: Elaborada pela autora, 2022

#### 4.6 Dashboard IoT ThingSpeak Monitor Widget

Para criar um sistema de notificações de alerta, foi utilizado o *IoT ThingSpeak Monitor Widget*, que possibilita, além de visualizações gráficas e numéricas das informações contidas na plataforma, a aplicação de alertas para valores pré-definidos. Inicialmente, deve-se configurar o recurso inserindo o número de identificação do canal (*Channel ID*) e a chave de escrita (*Read API KEY*). Posteriormente, é necessário indicar até dois campos para visualizações. Os campos apontados nesta aplicação foram a capacidade do estoque e os custos.

Igualmente, com auxílio da ferramenta é possível estabelecer um sistema de alertas, que para este trabalho, foi definido como alerta de limite inferior excedido do valor 10.0 referente à capacidade do estoque, com notificação insistente. Desta forma, ao atingir 10% de capacidade, alertas são enviados ao *smartphone* através de vibrações. O sistema permanece ativo até que o usuário leia a notificação, e prossegue emitindo notificações até que o estado da capacidade de estoque mude para um limite superior. O recurso permite configurar a interface para exibição personalizada em relação ao nome, cor e transparência do *Widget*, conforme exhibe a Figura 42.

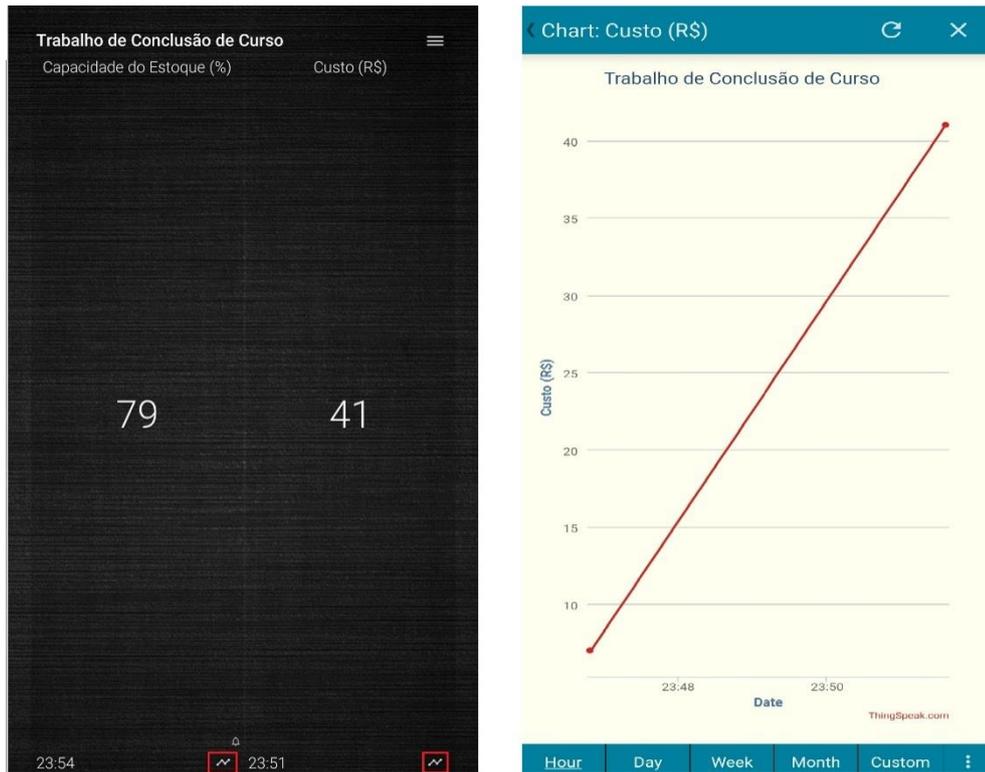
Figura 42 - Configurações *Widget*



Fonte: Elaborada pela autora, 2022

A exibição dos dados acontece de forma numérica, sendo possível a visualização gráfica em horas, dia, semana, mês e ano, ao selecionar a opção no canto inferior direito, destacado em vermelho, conforme ilustrado pela Figura 43.

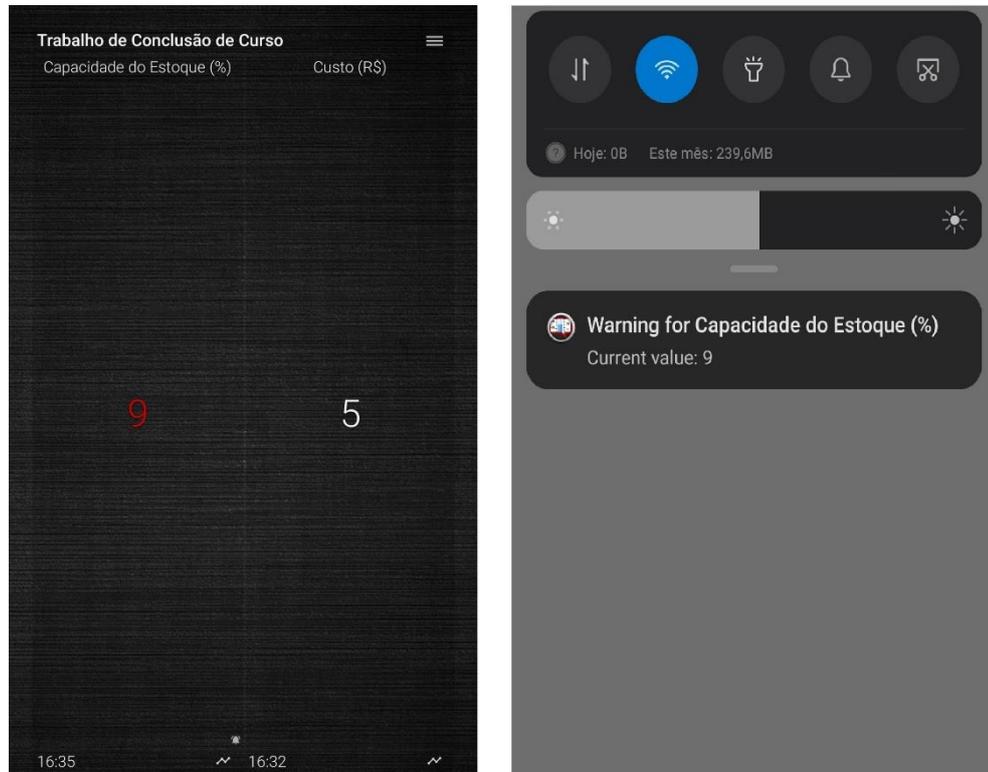
Figura 43 - Visualização numérica e gráfica em *Widget*



Fonte: Elaborada pela autora, 2022

Ademais, quando o sistema de alerta é ativado, a exibição numérica do campo adquire cor vermelha, e a mensagem de alerta é exibida pela barra de notificação, conforme ilustrado pela Figura 44.

Figura 44 - Sistema de alerta



Fonte: Elaborada pela autora, 2022

#### 4.7 Análise de custos e viabilidade técnica

Para a implementação do protótipo proposto, os custos com os materiais utilizados foram de R\$741,16, conforme demonstrado pela Tabela 6. Para uma aplicação real os custos relacionados aos materiais e instalação em ambiente físico seriam mais elevados, considerando-se os custos adicionais de instalação. A implantação de um sistema cujo princípio de funcionamento é o mesmo que o proposto neste trabalho deve ser avaliado para atender cada especificidade, tal qual localidades em que há acesso à *Internet* com históricos recorrentes de falhas, a fim de conferir maior segurança nas aplicações. Entretanto, devido a proposta deste trabalho, a viabilidade econômica mostrou-se satisfatória, haja vista se tratar da implantação de um sistema que usufrui de tecnologias para melhoria das aplicações sugeridas. É importante ressaltar que, para obter uma análise de viabilidade econômica precisa, um estudo de caso deve ser realizado, a fim de determinar quais elementos devem compor o circuito, avaliando as exigências da implementação.

Tabela 6 - Tabela de custos

<b>Número</b>	<b>Material</b>	<b>Quant.</b>	<b>Custo unit.</b>	<b>Custo total</b>	<b>Fonte</b>
1	<i>Heltec</i> ESP32 LoRa V2	2	314.99	629.98	Casa da robótica
2	Sensor ultrassônico HC-SR04	2	14.49	28.98	Casa da robótica
3	Protoboard 830 pontos	2	25.11	50.22	Amazon
4	Fonte de alimentação para protoboard	2	15.99	31.98	Casa da robótica
<b>Total</b>				<b>741.16</b>	

Fonte: Elaborada pela autora, 2022

## 5 CONSIDERAÇÕES FINAIS

A tecnologia de comunicação *LoRa*® possui vantagens indispensáveis para aplicações que demanda a transmissão de dados a longas distâncias quando não há acesso à *Internet*. Através deste trabalho foi possível implementar um sistema para monitoramento de insumos via conexão sem fio utilizando tecnologia de comunicação *LoRa*® e envio de dados para plataforma *IoT ThingSpeak*™. Desta forma, pode-se inferir que é possível suprir as necessidades tecnológicas referentes a transmissão e recepção de dados de estações inseridas em locais de difícil acesso à *Internet*.

Através das tecnologias empregadas foi possível monitorar, em tempo real, os dados referentes à capacidade do reservatório, capacidade do estoque, consumo e custo de uma possível estação. Observou-se por meio dos testes de alcance *LoRa*® um rendimento altamente satisfatório, visto que as avaliações foram realizadas em ambiente urbano, passível de diversas interferências. Desta forma, espera-se que para ambientes remotos o alcance *LoRa*® seja significativamente maior.

Esta pesquisa viabiliza que a solução seja implementada em estudos mais amplos, aplicando o princípio a locais de difícil acesso à *Internet*. Deste modo, a utilização deste tipo de tecnologia mostra-se viável e adaptável a outras unidades de controle e monitoramento, tais quais estabelecimentos agropecuários, visto seu baixo acesso à *Internet*, possibilitando assim o desenvolvimento tecnológico destas unidades.

Os objetivos do trabalho foram alcançados, uma vez que a tecnologia de comunicação *LoRa*® e o sistema de envio de dados para plataforma *IoT ThingSpeak*™ obtiveram resultados satisfatórios, demonstrando sua viabilidade em aplicações reais. Desta forma, as tecnologias empregadas nesta aplicação evidenciam sua importância no que tange ao constante desenvolvimento da Indústria 4.0.

Para trabalhos futuros, sugere-se a implementação de uma *dashboard* que viabilize o cálculo de custos adicionando um campo para inserção do valor do insumo pelo usuário, desafixando desta forma o valor anexado neste trabalho, assim como a realização de estudos de caso envolvendo ambientes relevantes para a aplicação proposta.

## REFERÊNCIAS

- ABEL, Matheus George. **Protocolo de comunicação de topologia mesh utilizando radiofrequência por modulação LoRa**. 2019. Trabalho de Conclusão de Curso.
- ABREU, Cleyde Evangelista Maia et al. Indústria 4.0: Como as empresas estão utilizando a simulação para se preparar para o futuro. **Revista de Ciências Exatas e Tecnologia**, v. 12, n. 12, p. 49-53, 2017.
- AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES. **Ato nº 14448, de 04 de dezembro de 2017**. Disponível em: <http://www.ncc.org.br/img/Ato%2014448.pdf>. Acesso em: 30 set. 2022.
- AMAZON. **Protoboard 830 pontos**. Disponível em: [https://www.amazon.com.br/Protoboard-Pontos-Prot%C3%B3tipos-Arduino-Raspberry/dp/B07GJ6XP6R/ref=asc\\_df\\_B07GJ6XP6R/?tag=googleshopp00-20&linkCode=df0&hvadid=379787216837&hvpos=&hvnetw=g&hvrnd=9016164906340802647&hvppone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=1031631&hvtargid=pla-1728651315726&psc=1](https://www.amazon.com.br/Protoboard-Pontos-Prot%C3%B3tipos-Arduino-Raspberry/dp/B07GJ6XP6R/ref=asc_df_B07GJ6XP6R/?tag=googleshopp00-20&linkCode=df0&hvadid=379787216837&hvpos=&hvnetw=g&hvrnd=9016164906340802647&hvppone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=1031631&hvtargid=pla-1728651315726&psc=1). Acesso em: 18 set. 2022.
- ARDUINO. **Downloads**. Disponível em: <https://www.arduino.cc/en/software>. Acesso em: 23 set. 2022.
- ASHTON, Kevin et al. That ‘internet of things’ thing. **RFID journal**, v. 22, n. 7, p. 97-114, 2009.
- AUGUSTIN, A. et al. **A study of Lora**: Long range & low power networks for the internet of things. *Sensors (Switzerland)*, v. 16, n. 9, p. 1–18, 2016.
- CASA DA ROBÓTICA. **ESP32 V2 LoRa Display Oled Wi-Fi 915MHz HeltecBle com antena**. Disponível em: <https://www.casadarobotica.com/placas-embarcadas/esp/placas/esp32-v2-lora-display-oled-wifi-915mhz-heltec-ble-com-antena>. Acesso em: 22 set. 2022.
- CASA DA ROBÓTICA. **Fonte de alimentação 3,3V 5V para protoboard**. Disponível em: <https://www.casadarobotica.com/fonte-e-conversores/fontes/outros/fonte-de-alimentacao-3-3v-5v-para-protoboard>. Acesso em: 18. Out. 2022.

CASA DA ROBÓTICA. **Heltec ESP32 LoRa V2**. Disponível em:

[https://www.casadarobotica.com/placas-embarcadas/esp/placas/esp32-v2-lora-display-oled-wifi-915mhz-heltec-ble-com-antena?parceiro=3259&gclid=CjwKCAjw-rOaBhA9EiwAUkLV4sIUkjJ5EhJqJFzV9TbxH8ommSv-1F1BbJx2b6CoqAswbAcn5CybBoChs8QAvD\\_BwE](https://www.casadarobotica.com/placas-embarcadas/esp/placas/esp32-v2-lora-display-oled-wifi-915mhz-heltec-ble-com-antena?parceiro=3259&gclid=CjwKCAjw-rOaBhA9EiwAUkLV4sIUkjJ5EhJqJFzV9TbxH8ommSv-1F1BbJx2b6CoqAswbAcn5CybBoChs8QAvD_BwE). Acesso em 18 out. 2022.

CASA DA ROBÓTICA. **Sensor ultrassônico HC-SR04**. Disponível em:

<https://www.casadarobotica.com/sensores-e-modulos/sensores/movimento-e-proximidade/sensor-ultrassonico-hc-sr04-sensor-de-distancia>. Acesso em: 18 out. 2022.

ČOLAKOVIĆ, Alem; HADŽIALIĆ, Mesud. Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues. **Computer networks**, v. 144, p. 17-39, 2018.

DA SILVA, AgdaAloha Prazeres et al. LUVA GUIA PARA DEFICIENTES VISUAIS, 2018.

DATASHEET. HC-SR04 Ultrasonic Sensor. Disponível em:

<https://cdn.datasheetspdf.com/pdf/1380136/ETC/HC-SR04/1>. Acesso em: 21 set. 2021.

DINESH, E.; RAMESH, L. Analysis of Cloud Services in Agriculture Through Internet of Things and Big Data. **Indian Journal of Science and Technology**, v. 11, n. 19, p. 1-4, 2018.

FILIFELOP. **Comunicação LoRa ponto-a-ponto com Módulos ESP32 LoRa**. Disponível em: <https://www.filieflop.com/blog/comunicacao-lora-ponto-a-ponto-com-modulos-esp32-lora/>. Acesso em: 02 out. 2022.

FONSECA, GUILHERME EDUARDO. Desenvolvimento de sistema para automação residencial baseado em IOT com protocolo de mensagens MQTT. **Instituto Federal de Ciência, Educação e Tecnologia de Minas Gerais**, 2021.

GLOBAL CONNECTIVITY REPORT 2022. International Telecommunication Union. Disponível em: <https://www.itu.int/hub/publication/d-ind-global-01-2022/>. Acesso em: 21 set. 2022.

GOKHALE, Pradyumna; BHAT, Omkar; BHAT, Sagar. Introduction to IOT. **International Advanced Research Journal in Science, Engineering and Technology**, v. 5, n. 1, p. 41-44, 2018.

HAXHIBEQIRI, Jetmir et al. A survey of LoRaWAN for IoT: From technology to application. **Sensors**, v. 18, n. 11, p. 3995, 2018.

HELTEC AUTOMATION. **Wi-Fi LoRa 32 (V2.1) Pasheout**. Disponível em: <https://heltec.org/project/wifi-lora-32/>. Acesso em: 22 set. 2022.

HUBERT, MARCO KASDORF. O protocolo CAN como solução para aplicações distribuídas, baseadas em objetos, entre PCs e microcontroladores. **Universidade Federal de Pelotas**, 2001.

IBGE. **Censo Agropecuário de 2017**. Disponível em: <https://sidra.ibge.gov.br/pesquisa/censo-agropecuario/censo-agropecuario-2017>. Acesso em: 21 set. 2022.

INTERNET JÁ É ACESSÍVEL EM 90,0% DOS DOMICÍLIOS DO PAÍS EM 2021. IBGE. Disponível em: <https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/34954-internet-ja-e-acessivel-em-90-0-dos-domicilios-do-pais-em-2021#:~:text=De%202019%20a%202021%2C%20o%20percentual%20de%20domic%C3%A9lios%20com%20conex%C3%A3o,%25%20para%2083%2C5%25>. Acesso em: 21 set. 2022.

KERSCHBAUMER, R. Engenharia de controle e Automação Microcontroladores. 2013.

LORA. **LoRa**. Disponível em: <https://lora.readthedocs.io/en/latest/>. Acesso em: 30 set. 2022.

MATHWORKS. ThingSpeak. Disponível em: <https://www.mathworks.com/help/thingspeak/>. Acesso em: 02 out. 2022.

MAUREIRA, Marcello A. Gómez; OLDENHOF, Daan; TEERNSTRA, Livia. ThingSpeak— an API and Web Service for the Internet of Things. **World Wide Web**, 2011.

MAZER DEV. **Protocolos de rede – O que é**. 2021. Disponível em: <https://mazer.dev/protocolos-de-rede-o-que-e/>. Acesso em: 26 set. 2022.

MIRY, Abbas Hussien; ARAMICE, Gregor Alexander. Water monitoring and analytic based ThingSpeak. **International Journal of Electrical and Computer Engineering**, v. 10, n. 4, p. 3588, 2020.

PINTO, Pedro. **LoRa: Tecnologia de Radiofrequência de baixo consumo energético**. Disponível em: <https://pedropinto.medium.com/lora-tecnologia-de-r%C3%A9dio->

frequ%C3%A2ncia-de-baixo-consumo-energ%C3%A9tico-54bef93ecb74. Acesso em: 30 set. 2022.

RANDOM NET TUTORIALS. **ESP32 with LoRa using Arduino IDE – Getting Started**. Disponível em: <https://randomnerdtutorials.com/esp32-lora-rfm95-transceiver-arduino-ide/>. Acesso em: 10 out. 2022.

REZENDE, J. A. D.; MAITELLI, André Laurindo. Um esquema de neurocontrole com treinamento em tempo real aplicado ao posicionamento de um servomotor. **Simpósio Brasileiro de Automação Inteligente**, 1999.

SILVA, Eduardo Corneto; ESPEJO, Márcia Maria dos Santos Bortolucci. INTERNET OF THINGS (IOT) NO AGRONEGÓCIO. **Encontro Internacional de Gestão, Desenvolvimento e Inovação (EIGEDIN)**, v. 4, n. 1, 2020.

SOUZA, Geraldo da Silva et al. Função de produção com base nos microdados do Censo Agropecuário de 2017. **Revista de Política Agrícola**, v. 29, n. 4, p. 65, 2020.

TECNOTRONIS. **Sensor ultrassônico HC-SR04**. Disponível em: <https://www.tecnotronics.com.br/sensor-ultra-sonico-de-distancia-hc-sr04.html>. Acesso em: 21 set. 2022.

THINGSPEAK. Learn More About ThingSpeak. Disponível em: [https://thingspeak.com/pages/learn\\_more](https://thingspeak.com/pages/learn_more). Acesso em: 02 out. 2022.

TIRELLI, Guilherme et al. Determinação do coeficiente de atrito cinético com o uso de Arduino e do sensor ultrassônico HC-SR04. 2022.

TOZETTO, Diego F.; SILVA, Rodrigo V.; FREIRE, Roberto Z. Sistema de Comunicação LoRa: Uma solução para Transmissão de Dados no Agronegócio envolvendo Longas Distâncias e Baixo Custo. In: **Simpósio Brasileiro de Automação Inteligente-SBAI**. 2021.

TRANSFORMA INSIGHTS. **Global IoT Forecast Report, 2021-2030**. Disponível em: <https://transformainsights.com/news/global-iot-connections-294>. Acesso em: 21 set. 2022.

UFRJ. **LoRa**. Disponível em: <https://www.gta.ufrj.br/ensino/eel878/redes1-2019-1/vf/lora/modulacao.html>. Acesso em: 28 set. 2022.

ZAWADZKI, Przemysław; ŻYWICKI, Krzysztof. Smart product design and production control for effective mass customization in the Industry 4.0 concept. **Management and production engineering review**, 2016.

## ANEXOS A – CÓDIGO *SENDER*

```

#include "heltec.h" /* Biblioteca da placa ESP32 LoRa (V2) Heltec */
#include <Ultrasonic.h> /* Biblioteca do sensor ultrassônico */

#define BAND 915E6 /* Frequência de banda utilizada no Brasil */

Ultrasonic ultrasonic_reservatorio(33, 32); /* Inicializa o sensor do reservatório nos pinos
definidos TRIGGER e ECHO, respectivamente */
Ultrasonic ultrasonic_estoque(12, 13); /* Inicializa o sensor do estoque nos pinos definidos por
TRIGGER E ECHO, respectivamente */

String packet;
int porcentagem_reservatorio;
int porcentagem_estoque;
float profundidade_reservatorio = 50.0;
float profundidade_estoque = 150.0;
int contador = 0;

/* Protótipo da função */
void leitor_dist(); /* Função de leitura dos dados do sensor ultrassônico para o reservatório */
void leitor_estoque(); /* Função de leitura dos dados do sensor ultrassônico para o estoque */
void sendPacket(); /* Função relacionada ao envio de dados LoRa */

void leitor_dist(){

    float distancia_reservatorio = 0;

    for (int i = 0; i < 10; i++){ /* Conta e soma 10 vezes a distância */
        ultrasonic_reservatorio.measure();
        distancia_reservatorio += ultrasonic_reservatorio.get_cm();
        delay(50);
    }

    distancia_reservatorio = distancia_reservatorio / 10; /* Média das distâncias para melhor
exatidão nos resultados */

    Serial.println("Leitura do Sensor de Distancia do reservatório em cm:");
    Serial.println(distancia_reservatorio);
    porcentagem_reservatorio = 100 - ( 100 * (distancia_reservatorio - 5) /
profundidade_reservatorio);

    Serial.println("Capacidade do reservatório em porcentagem: ");
    Serial.println(porcentagem_reservatorio);

    if(porcentagem_reservatorio < 0){ /* Condição para que a leitura de dados não apresente
valores indesejáveis apresentadas pelo sensor quando fora da margem estipulada */
        porcentagem_reservatorio = 0;
    }
    if(porcentagem_reservatorio > 100){
        porcentagem_reservatorio = 100;
    }
}

void leitor_estoque(){
float distancia_estoque = 0;

    for (int j = 0; j < 10; j++){
        ultrasonic_estoque.measure();
        distancia_estoque += ultrasonic_estoque.get_cm();
        delay(50);
    }
    distancia_estoque = distancia_estoque / 10;
    porcentagem_estoque = 100 - ( 100 * (distancia_estoque - 5) / profundidade_estoque);

    if(porcentagem_estoque < 0){ /* Condição para que a leitura de dados não apresente valores
indesejáveis apresentadas pelo sensor quando fora da margem estipulada */
        porcentagem_estoque = 0;
    }
    if(porcentagem_estoque > 100){
        porcentagem_estoque = 100;
    }
}
Serial.println("Leitura do Sensor de Distancia do estoque em cm:");
Serial.println(distancia_estoque);

```

```

        Serial.println("Capacidade do estoque em porcentagem: ");
        Serial.println(porcentagem_estoque);
    }

    void sendPacket() /* Função para envio do pacote LoRa para o receptor */
    {
        LoRa.beginPacket();
        LoRa.print(porcentagem_reservatorio);
        LoRa.println(" ");
        LoRa.print(porcentagem_estoque);
        LoRa.endPacket();
    }

    void setup() /* Função principal */
    {
        Serial.begin (115200);

        /* Habilita a placa */
        Heltec.begin(true /*Habilita o Display*/, true /*Heltec.Heltec.Heltec.LoRa Disable*/, true /
        *Habilita debug Serial*/, true /*Habilita o PABOOST*/, BAND /*Frequência BAND*/);

        LoRa.setSyncWord(0xF33); /* Palavra de sincronização - Garante a comunicação entre os dois
        dispositivos LoRa de interesse */

        Heltec.display->init();
        Heltec.display->flipScreenVertically();
        Heltec.display->setFont(ArialMT_Plain_16);
        Heltec.display->clear();
        Heltec.display->drawString(33, 5, "Iniciado");
        Heltec.display->display();
        delay(1000);
    }

    Heltec.display->init();
    Heltec.display->flipScreenVertically();
    Heltec.display->setFont(ArialMT_Plain_16);
    Heltec.display->clear();
    Heltec.display->drawString(33, 5, "Iniciado");
    Heltec.display->display();
    delay(1000);
}

void loop()
{
    leitor_dist();
    leitor_estoque();
    contador = contador + 1;
    Serial.print("Enviando pacote ");
    Serial.println(contador); /* verifica quantos pacotes foram enviados */

    Heltec.display->clear();
    Heltec.display->setTextAlignment(TEXT_ALIGN_LEFT);
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->drawString(5, 5, "Enviando");
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->drawString(5, 20, "Reservatorio:");
    Heltec.display->drawString(70, 20, (String)porcentagem_reservatorio);
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->drawString(100, 20, " % ");
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->drawString(5, 35, "Estoque:");
    Heltec.display->drawString(70, 35, (String)porcentagem_estoque);
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->drawString(100, 35, " % ");
    Heltec.display->display();
    sendPacket(); /* Envia dados via LoRa*/
}

```

## ANEXOS B – CÓDIGO *RECEIVER*

```

#include <WiFi.h>
#include "ThingSpeak.h"
#include "heltec.h"

#define BAND 915E6 /* Frequência de banda utilizada no Brasil */

int estoque_anterior = 0;
int consumo = 0;
int aux_envio = 0;
int contador = 0;

const char minha_rede[] = "*****"; /* Nome da rede Wi-Fi que deseja-se conectar */
const char senha[] = "*****"; /* Senha da rede Wi-fi que deseja-se conectar */

WiFiClient client;

const long CANAL = 1884389; /* Número do canal que deseja-se conectar ao ThingSpeak */
const char *CHAVE_ESCRITA = "*****"; /* Chave de escrita restrita ao canal do ThigSpeak */

String packSize = "--";
String packet ;

long tempo_atual_envio = 0;
int periodo_envio = 120000; /* Intervalo entre envio de dados para o ThingSpeak */
int reservatorio, estoque;

/* Protótipo da função */
void LoRa_Print_Tamanho(); /* Função para imprimir o tamanho e a distância lidos pelos sensor */

void recebe_dados(int packetSize); /* Função para receber os dados via LoRa e armazenar na
variável Packet */

void LoRa_Print_Tamanho(){
  Heltec.display->clear();
  Heltec.display->setTextAlignment(TEXT_ALIGN_LEFT);
  Heltec.display->setFont(ArialMT_Plain_10);

  Heltec.display->drawString(0, 1, packet);
  Heltec.display->display();
  Heltec.display->clear();
}

void recebe_dados (int packetSize){
  int j = 0, aux = 0;
  packet = "";
  packSize = String(packetSize,DEC);
  for (int i = 0; i < packetSize; i++) {
    packet += (char) LoRa.read(); /* Atribui um caractere por vez a váriavel packet */
    //Serial.println(packet);
    aux = packet.toInt(); /* Transforma em inteiro */
    if (" "){
    }
    if (i == 0 && aux == 1){
      j = 1;
    }
    if (i == 1 && aux == 0){
      j = 1;
    }
    if (i == 2 && aux == 0){
      j = 1;
    }
    if(i == 3){
      j=1;
    }
  }

  String Sreservatorio;
  String Sestoque;

```

```

if(j==1){
  Sreservatorio = packet.substring(0,3);
  Sestoque = packet.substring(3);
} else {
  Sreservatorio= packet.substring(0,2);
  Sestoque = packet.substring(2);
}
Serial.print(" Sreservatorio = ");
Serial.println(Sreservatorio);
Serial.print(" Sestoque = ");
Serial.println(Sestoque);
LoRa_Print_Tamanho();

reservatorio = Sreservatorio.toInt(); /* Transforma em valor inteiro */
estoque = Sestoque.toInt(); /* Transforma em valor inteiro */

Heltec.display->clear();
Heltec.display->setTextAlignment(TEXT_ALIGN_LEFT);
Heltec.display->setFont(ArialMT_Plain_10);
Heltec.display->drawString(5, 5, "Recebendo");
Heltec.display->setFont(ArialMT_Plain_10);
Heltec.display->drawString(5, 20, "Reservatorio:");
Heltec.display->drawString(70, 20, (String)reservatorio);
Heltec.display->setFont(ArialMT_Plain_10);
Heltec.display->drawString(100, 20, " % ");
Heltec.display->setFont(ArialMT_Plain_10);
Heltec.display->drawString(5, 35, "Estoque:");
Heltec.display->drawString(70, 35, (String)estoque);
Heltec.display->setFont(ArialMT_Plain_10);
Heltec.display->drawString(100, 35, " % ");
Heltec.display->display();
}

void enviar_para_nuvem(){
  if (WiFi.status() != WL_CONNECTED) {
    Serial.print("Tentando conectar com minha_rede: ");
    Serial.println(minha_rede);
    while (WiFi.status() != WL_CONNECTED) {
      WiFi.begin(minha_rede, senha);
      delay(5000);
    }
    Serial.println("\nConectado.");
  }

  if (millis() - tempo_atual_envio > periodo_envio) {
    /* Define campos e variáveis a serem lidas no ThingSpeak */
    ThingSpeak.setField(1, reservatorio); /* Envio do valor do reservatório lido para o ThingSpeak */
    /*
    ThingSpeak.setField(2, estoque); /* Envio do valor do estoque lido para o ThingSpeak */

    if (estoque < estoque_anterior){
      consumo = estoque_anterior - estoque; /* Cálculo de consumo */

      ThingSpeak.setField(3, consumo); /* Envio do consumo para o ThingSpeak quando há alteração
no estoque */
    }

    else {
      consumo = 0;
      ThingSpeak.setField(3, consumo); /* Envio do consumo para o ThingSpeak quando não há
alteração no estoque */
    }

    estoque_anterior = estoque;

    /* Escreve para o canal no ThingSpeak */
    int x = ThingSpeak.writeFields(CANAL, CHAVE_ESCRITA);

```

```

    if (x == 200) {
        Serial.println("Atualização do canal bem sucedida.");
    }
    else {
        Serial.println("Problema ao atualizar canal. Código de erro HTTP " + String(x));
    }
    tempo_atual_envio = millis();
}
}

void setup()
{
    Serial.begin(115200);

    /* Inicializa a placa */
    Heltec.begin(true /*Habilita o Display*/, true /*Heltec.Heltec.Heltec.LoRa Disable*/, true /*Habilita debug Serial*/, true /*Habilita o PABOOST*/, BAND /*Frequência BAND*/);

    LoRa.setSyncWord(0xF33); /* Palavra de sincronização - Garante a comunicação entre os dois dispositivos LoRa de interesse */

    Heltec.display->init();
    Heltec.display->flipScreenVertically();
    Heltec.display->setFont(ArialMT_Plain_10);
    Heltec.display->clear();
    Heltec.display->drawString(10, 5, "Iniciado");
    Heltec.display->drawString(10, 30, "Aguardando os dados...");
    Heltec.display->display();
    Serial.println("Iniciado");
    Serial.println("Aguardando os dados...");
    delay(1000);

    LoRa.receive(); /* Função para habilitar o rádio LoRa para receber dados */
    WiFi.mode(WIFI_STA);
    ThingSpeak.begin(client); /* Inicializa ThingSpeak */
}

void loop(){
    contador = contador + 1;
    Serial.print("Recebendo pacote ");
    Serial.println(contador); /* Verifica quantos pacotes foram recebidos */
    Serial.print(" RSSI: ");
    Serial.println(LoRa.packetRssi()); /* Verifica a intensidade do sinal LoRa em dBm */

    int packetSize = LoRa.parsePacket(); /* Verifica se um novo pacote foi recebido */
    if (packetSize) { /* Verifica se há dados chegando via LoRa */
        recebe_dados(packetSize);
        Serial.print("Recebendo dados: ");
        Serial.println(packet); /* Imprime no monitor serial os dados recebidos */

        enviar_para_nuvem(); /* Envia os dados para o ThingSpeak */
    }
}
}

```

## ANEXOS C – CÓDIGO *MATLAB ANALYSIS*

```
% Identificação do canal a ser lido
readChannelID = [1884389];

% Chave API de leitura do canal a ser lido
readAPIKey = '*****';

% Identificação do canal para escrita de dados
writeChannelID = [1884389];

% Chave API de escrita do canal para escrita de dados
writeAPIKey = '*****';

%% Leitura de dados %%
% Leitura do Campo 3 - Consumo

preco_kg = 100; % Valor do insumo

pause(2);
pause(2);
pause(2);
pause(2);

consumo_atual = thingSpeakRead(readChannelID, 'ReadKey', readAPIKey, 'Field', 3);

pause(2);
pause(2);
pause(2);
pause(2);

custo = consumo_atual/100 * preco_kg

thingSpeakWrite(writeChannelID, custo, 'WriteKey', writeAPIKey, 'Field', 4);
```