

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

MATEUS LOPES DO NASCIMENTO

AUTOMATIZAÇÃO DE AMBIENTES UTILIZANDO TECNOLOGIA LORA

CAMPO MOURÃO

2023

MATEUS LOPES DO NASCIMENTO

AUTOMATIZAÇÃO DE AMBIENTES UTILIZANDO TECNOLOGIA LORA

Automating environments using lora technology

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Bacharel em Engenharia Eletrônica da
Universidade Tecnológica Federal do Paraná.

Orientador: Marcio Rodrigues da Cunha

CAMPO MOURÃO

2023



[4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/)

Esta licença permite download e compartilhamento do trabalho desde que sejam atribuídos créditos ao(s) autor(es), sem a possibilidade de alterá-lo ou utilizá-lo para fins comerciais. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

MATEUS LOPES DO NASCIMENTO

AUTOMATIZAÇÃO DE AMBIENTES UTILIZANDO TECNOLOGIA LORA

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Bacharel em Engenharia Eletrônica da
Universidade Tecnológica Federal do Paraná.

Data de aprovação: 27/novembro/2023

Marcio Rodrigues da Cunha
Doutorado em Engenharia Elétrica
Universidade Tecnológica Federal do Paraná

Roberto Wilhelm Krauss Martinez
Doutorado em Engenharia Elétrica e Informática Industrial
Universidade Tecnológica Federal do Paraná

Lucas Ricken Garcia
Doutorado em Engenharia Elétrica e Informática Industrial
Universidade Tecnológica Federal do Paraná

CAMPO MOURÃO
2023

RESUMO

A automação por meio de tecnologias de comunicação é uma ferramenta crucial na busca por resultados mais eficazes e precisos, no que tange velocidade de processamento, capacidade de resposta e a eficácia na execução de tarefas. Desta forma, o controle de sistemas de climatização contribui não apenas para o conforto, mas também para a saúde, produtividade e eficiência geral em diversos ambientes, desde residências até edifícios comerciais e instalações industriais. Diante disso, foi realizado o monitoramento da temperatura e umidade de um cômodo residencial com o intuito de solidificar a viabilização dos protocolos de comunicação, usando uma rede de sensores. Após a coleta de leituras pelos sensores, elas são enviadas para um microcontrolador por meio de um protocolo de comunicação denominado Esp-Now, para assim realizar o tratamento das informações, pois as leituras são utilizadas para acionar um ventilador, responsável por garantir o conforto térmico do ambiente. Em seguida, as informações tratadas são enviadas para outro microcontrolador, através do protocolo de comunicação LoRa. Por fim, envia-se os resultados para um site, através do protocolo Message Queuing Telemetry Transport (Transporte de Telemetria com Filas de Mensagens) (MQTT), que proporcionaram maior clareza na interpretação dos resultados e no acompanhamento do monitoramento do ambiente. Enfim, tem-se um ambiente automatizado, consolidando que a viabilidade e eficiência da automação, neste caso, utilizando multi protocolos de comunicação.

Palavras-chave: protocolos de comunicação; internet das coisas; controle; esp-now; rede de sensores.

ABSTRACT

Automation through communication technologies is a crucial tool in the pursuit of more effective and precise results, regarding processing speed, responsiveness, and task execution efficiency. In this way, the control of air conditioning systems contributes not only to comfort but also to health, productivity, and overall efficiency in various environments, from residences to commercial buildings and industrial facilities. In light of this, the monitoring of temperature and humidity in a residential room was carried out with the aim of solidifying the feasibility of communication protocols using a sensor network. After the sensors collect readings, they are sent to a microcontroller through a communication protocol called Esp-Now to process the information. The readings are used to activate a fan, responsible for ensuring thermal comfort in the environment. Subsequently, the processed information is sent to another microcontroller through the LoRa communication protocol. Finally, the results are sent to a website using the MQTT protocol, providing greater clarity in interpreting the results and monitoring the environment. In conclusion, there is an automated environment, consolidating the viability and efficiency of automation, in this case, using multiple communication protocols.

Keywords: communication protocols; internet of things; control; esp-now; sensor network.

LISTA DE FIGURAS

Figura 1 – Captura de imagem da fila dos restaurantes da UNICAMP	8
Figura 2 – Comparação do alcance e largura de banda de algumas redes.	9
Figura 3 – Exemplo de rede <i>LoRaWAN</i>	11
Figura 4 – Exemplo de <i>Smart Cities</i>	12
Figura 5 – Pilares da estrutura inteligente do <i>Campus</i>	13
Figura 6 – Topologia da rede <i>LoRaWAN</i>	16
Figura 7 – Diagrama de Blocos do Projeto	20
Figura 8 – Distribuição dos Sensores DHT11-ESP01 no comodo utilizado na simulação.	21
Figura 9 – Fluxograma da Captura e Envio da Medição da Temperatura e Umidade	22
Figura 10 – Fluxograma da Recepção e Armazenamento das Leituras de Temperatura e Umidade.	23
Figura 11 – Diagrama de Blocos da Função que Calcula a Media das Temperaturas e Umidades.	24
Figura 12 – Exemplo Saída Monitor Serial da IDE	25
Figura 13 – Fluxograma do algoritmo responsável pela avaliação do acionamento do ventilador.	26
Figura 14 – Diagrama de Blocos da Recepção e Tratamento dos Dados no ESP32 "Servidor".	28
Figura 15 – Exemplo do conteúdo do pacote de dados enviado via LoRa	29
Figura 16 – Exemplo Ventilador Desligado.	31
Figura 17 – Exemplo Ventilador Desligado.	31
Figura 18 – Exemplo Ventilador Ligado.	32
Figura 19 – Exemplo Ventilador Ligado.	32
Figura 20 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 1.	33
Figura 21 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 2.	34
Figura 22 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 3.	34

Figura 23 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 4.	35
Figura 24 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 5.	35
Figura 25 – Recebendo Média da Temperatura e Umidade do Sensor 1, calculada no algoritmo implantado no ESP32 "Local".	36
Figura 26 – Recebendo Média da Temperatura e Umidade do Sensor 3, calculada no algoritmo implantado no ESP32 "Local".	37
Figura 27 – Recebendo Média da Temperatura e Umidade do Sensor 1, calculada no algoritmo implantado no ESP32 "Local".	38
Figura 28 – Recebendo Média da Temperatura e Umidade do Sensor 2, calculada no algoritmo implantado no ESP32 "Local".	38
Figura 29 – Recebendo Média da Temperatura e Umidade do Sensor 4, calculada no algoritmo implantado no ESP32 "Local".	39
Figura 30 – Exemplo Funcionamento do Projeto	39
Figura 31 – Exemplo Funcionamento do Projeto	40
Figura 32 – Gráfico de Superfície Utilizando Leituras de Temperatura.	41

LISTA DE ABREVIATURAS E SIGLAS

Siglas

ANATEL	Agência Nacional de Telecomunicações
bps	bits por segundo
BW	<i>Band Width</i> (Largura de Banda)
CR	Código de Erro
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IoT	<i>Internet Of Things</i> (Internet das Coisas)
ISM	Industrial - Scientific - Medical (Industrial - Científica - Médica)
LMS	<i>Learnig Management Systems</i> (Sistema de Gestão da Aprendizagem)
LoRa	<i>Long Range</i> (Longo Alcance)
LPWAN	<i>Low Power Wide Area Network</i> - (Rede de Grande Alcance e Baixo Consumo)
MAC	<i>Media Access Control</i> (Controle de Acesso de Mídia)
MQTT	Message Queuing Telemetry Transport (Transporte de Telemetria com Filas de Mensagens)
P2P	<i>Peer-to-peer</i> (Ponto a ponto)
RFID	<i>Radio-frequency Identification</i> (Identificação por Rádio Frequência)
RSSI	<i>Received Signal Strength Indicator</i> (Indicador de intensidade do sinal recebido)
SF	<i>Spreanding Factor</i> (Fator de Espalhamento)
UNICAMP	Universidade Estadual de Campinas
WLAN	<i>Wireless Local Area Network</i> (Rede de Área Local Sem Fio)
WMAN	<i>Wireless Metropolitan Area Network</i> (Rede de Área Metropolitana Sem Fio)
WPAN	<i>Wireless Personal Area Network</i> (Rede de Área Pessoal Sem Fio)

SUMÁRIO

1	INTRODUÇÃO	8
1.1	Objetivos	10
1.1.1	Objetivos específicos	10
1.2	Justificativa	11
2	REFERENCIAL TEÓRICO	12
2.1	<i>Smart Campus</i>	12
2.2	Infra estrutura tecnológica	15
2.3	Protocolo LoRa	15
2.4	Protocolo esp-now	18
2.5	Protocolo mqtt	19
3	METODOLOGIA	20
4	RESULTADOS E DISCUSSÕES	30
5	CONCLUSÃO	42
	REFERÊNCIAS	43

1 INTRODUÇÃO

A tecnologia tornou-se parte do dia a dia das pessoas, auxiliando em diversas tarefas e proporcionando mais comodidade aos seus usuários, mas tal comodidade exige conhecimento e estrutura para que se torne verdadeiramente viável.

Dentro desse contexto, podemos introduzir os ambientes inteligentes (do inglês *smart spaces*), onde dispositivos estão interligados via comunicações sem fio e tomam decisões com base nas condições analisadas (Gilman *et al.*, 2020).

Tem-se neste modelo de ambiente: as cidades inteligentes (do inglês *smart cities*) e as casas inteligentes (do inglês *smart homes*). Nas *smart cities*, semáforos alteram o seu estado de acordo com o tráfego. Já nas *smart homes*, o alarme de um dispositivo, como um celular, pode servir de estímulo para o acionamento de um ar condicionado, adiantando, assim, o conforto térmico do ambiente antes que o proprietário chegue em sua residência (Evans, 2011).

Inserem-se também nesse exemplo de ambiente os denominados campus inteligentes (do inglês *smart campus*). O objetivo do *smart campus* é desenvolver, por meio da integração de tecnologias IoT, um ambiente sustentável, seguro e que ajude a gestão administrativa e o desenvolvimento acadêmico (Liu, 2017).

Diante disso, tarefas antes realizadas manualmente, hoje podem ser automatizadas, garantindo não só apenas comodidade, mas segurança e eficiência, no que tange velocidade e precisão na realização destas tarefas e melhores resultados. Além disso, podendo liberar tempo para outras atividades (Ramos; Santos, 2015).

Na Universidade Estadual de Campinas (UNICAMP), o projeto de *smart campus* deu-se início em 2016. Tem-se um exemplo de um projeto em operação, onde é possível com a captura das imagens das filas dos restaurantes, processa-las usando uma técnica de aprendizado de máquina e fazer a gestão mais eficiente do fluxo de pessoas (Avelar, 2023a). A seguir é ilustrado na Figura 1 a captura da imagem da fila do restaurante.

Figura 1 – Captura de imagem da fila dos restaurantes da UNICAMP



Fonte: Avelar (2023b).

Desse modo, diversas tecnologias e protocolos de comunicação foram desenvolvidos para fornecer auxílio na automação desses ambientes.

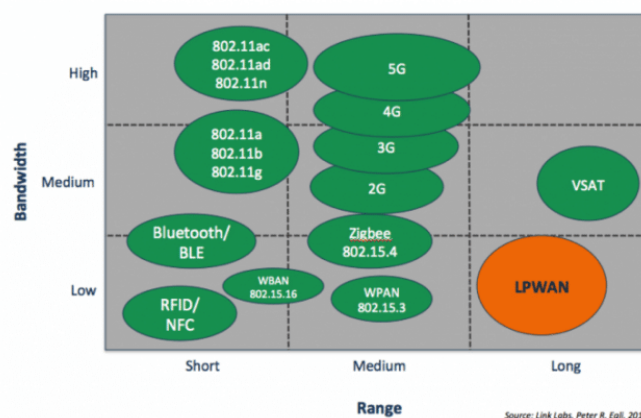
Dentre os protocolos de comunicação, tem-se o grupo 802, que é um segmento do Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) referente à rede e tecnologia de porte médio e local, elaborando os protocolos de comunicação em rede. Neste grupo despontam tecnologias como *Wireless Personal Area Network* (Rede de Área Pessoal Sem Fio) (WPAN), *Wireless Metropolitan Area Network* (Rede de Área Metropolitana Sem Fio) (WMAN) e *Wireless Local Area Network* (Rede de Área Local Sem Fio) (WLAN), que se diferenciam a partir dos seguintes parâmetros: contexto de uso, taxa de transmissão e alcance (Baratella; Guilherme Sobrinho, 2012), conforme descrito abaixo:

- WPAN - Menor alcance e baixa taxa de transmissão, definido pelo padrão 802.15. Exemplo: Bluetooth, ZigBee. Usado para interligar dispositivos pessoais próximos (Baratella; Guilherme Sobrinho, 2012).
- WLAN - Médio alcance e alta taxa de transmissão comparada com a WPAN. Exemplo: Wi-Fi (Kanika; Neha, 2014).
- WMAN - Longo alcance, são utilizadas em redes corporativas (Kanika; Neha, 2014).

Dentre os protocolos citados acima, o WLAN é o mais difundido para utilização em automações de ambientes. Isto se deve pela facilidade e baixo custo de implementação, já que grande parte dos ambientes a serem automatizados já possuem estrutura instalada (Kanika; Neha, 2014). Entretanto, a tecnologia WLAN tem um alcance limitado e pode ter um consumo de energia elevado em aplicações que necessitam manter uma conexão constante.

Diante do exposto, tem-se como alternativa de uso nas automações de ambientes a tecnologia *Low Power Wide Area Network* - (Rede de Grande Alcance e Baixo Consumo) (LPWAN), que tem como principais vantagens o baixo consumo de energia e longo alcance (Garcia; Kleinschmidt, 2017). Na Figura 2 é ilustrado uma comparação entre as tecnologias de comunicação citadas até aqui.

Figura 2 – Comparação do alcance e largura de banda de algumas redes.



Fonte: Egli (2015).

Apesar da presença de diversas alternativas em desenvolvimento em todo o mundo para redes LPWAN, no Brasil, ainda tem-se alguns empecilhos no que tange disponibilidade para uso. As mais avançadas, mas não com total acesso, se deparam com desafios como o processo de homologação na Agência Nacional de Telecomunicações (ANATEL) ou pouca cobertura. Entre essas tecnologias, pontuam-se as redes LoRa e SigFox. A diferença entre as duas redes é que a rede SigFox opera no Brasil através da WND Brasil, que é semelhante as operadoras de telefonia móvel, desta forma, torna-se a rede transparente para os usuários sem que tenha ação na gestão entre os dispositivos, enquanto isso a rede LoRa é uma rede aberta inserida no Brasil através de comunidades de usuários e seguidores (Garcia; Kleinschmidt, 2017).

Diante do exposto, será utilizado neste projeto multi protocolos de comunicação, como Esp Now, que é um protocolo do tipo WLAN, LoRa que é um protocolo do tipo LPWAN e será utilizado também o protocolo MQTT, que é um protocolo de mensagens leves e eficiente.

1.1 Objetivos

O escopo deste trabalho é implementar um sistema para automação de ambientes utilizando rede de sensores sem fio, internet das coisas e tecnologia Lora. Para alcançar esse objetivo, será implementado um caso de uso simples contendo o controle e a automação de um ventilador a partir do monitoramento da temperatura do ambiente, visando o conforto térmico do mesmo. Com essa abordagem almeja-se, futuramente, a aplicação de sistemas desse tipo em ambientes universitários, a partir da expansão de diversos casos de uso pertinentes a esses ambientes.

1.1.1 Objetivos específicos

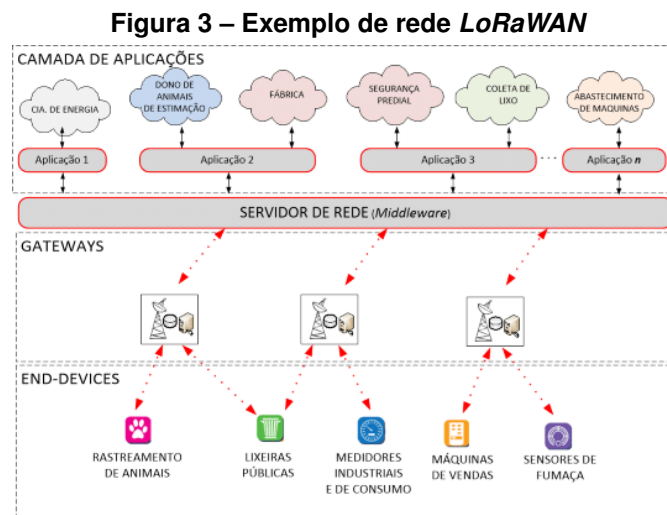
Para concretizar o escopo deste trabalho e melhor delimitá-lo, os objetivos específicos são:

1. Conceber a arquitetura do sistema;
2. Implementar uma rede de sensores sem fio com concentrador de dados local;
3. Implementar concentrador de dados global;
4. Implementar a persistência de dados em nuvem;
5. Implementar o controle e automação de um ventilador;
6. Testar o sistema implementado em um ambiente.

1.2 Justificativa

A automação residencial, predial ou ainda, de complexos multi prediais, como campus universitários, e os sistemas baseados em internet das coisas usam, na maioria das vezes, a tecnologia Wi-Fi (que está inserida no protocolo de comunicação WLAN) para a comunicação entre os dispositivos, mas no capítulo da introdução é notável que essa tecnologia possui limitações em relação a distância e infraestrutura. Dessa forma, automatizar ambientes que possuem dispositivos que estão distantes um dos outros ou que não disponham de infraestrutura apresentam um alto custo frente aos benefícios, se tratando da Tecnologia Wi-Fi.

Diante desse cenário, Silva (2017) destaca a tecnologia LoRa como uma alternativa viável. Essa tecnologia é empregada para conectar pontos distantes com baixo consumo de energia e de largura de banda (*bandwidth*) fornecendo uma conexão bidirecional. A seguir é ilustrado na Figura 3 a representação da topologia de rede LPWAN e LoRa.



Fonte: Garcia e Kleinschmidt (2017, p.2).

Na Figura 3 tem-se uma composição de camadas, divididas em coleta de informações pelos dispositivos (end-devices). Processamento e armazenamento através das antenas receptoras (gateways) e camada do Network Server (Middleware) e por fim a camada dos usuários, onde acontece a interação dos dados coletados.

2 REFERENCIAL TEÓRICO

A abordagem teórica deste trabalho acadêmico será realizada por de etapas que explicam o processo para proporcionar um ambiente automatizado que possa ser aplicado ao contexto de complexo multi-prediais, como universidades, utilizando alguns recursos como a Internet das Coisas (IoT). Para isso será necessário adquirir conhecimento sobre automação de ambientes universitários, focando nos requisitos das universidades, utilizando sensores, rede de sensores, protocolos de comunicação, protocolos de identificação e processamento.

2.1 *Smart Campus*

O conceito de *Smart Campus* pode ser comparado com o de *Smart Cities*, pois é possível afirmar que um ambiente de ensino como um *Campus* é uma "pequena" cidade, pois possui algumas características em comum, desta forma na Figura 4 é ilustrado um exemplo de *Smart Cities*.

Figura 4 – Exemplo de *Smart Cities*.



Fonte: Mitra (2018).

Na Figura 4 é ilustrada uma situação onde os carros utilizam de tecnologia para calcular a melhor rota no trânsito, e isso não acontece só entre os automóveis, mas com todos os objetos que estão próximos a eles. Da mesma forma é possível pensar em um ambiente universitário (*Campus*), usufruindo da mesma tecnologia, mas para interagir com os recursos oferecidos dentro do *Campus*.

Para Kwok (2015, p.2), um *campus* Inteligente é caracterizado “como um *campus* que tem a capacidade de responder a novas situações ocorridas no *campus* em suas tarefas diárias”. Esse mesmo autor identifica oportunidades de pesquisas envolvendo *Learnig Manage-*

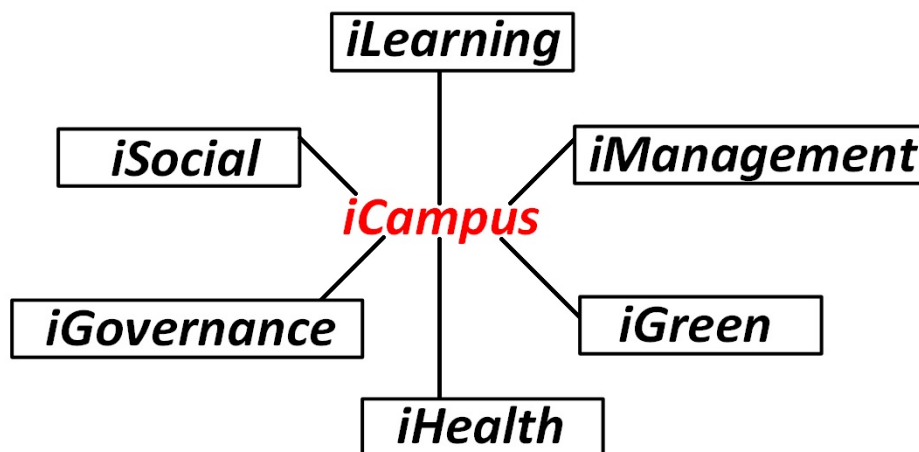
ment Systems (Sistema de Gestão da Aprendizagem) (LMS), aprendizagem personalizada, aprendizagem adaptativa, espaços de imersão educacional e ambiente de aprendizagem seguro. Além disso, também é diferenciado o *campus* digital do *campus* inteligente e identificado requisitos necessários para a construção de um *campus* inteligente: disponibilidade de dados, conhecimento de procedimentos para tomada de decisões, a integração de sistemas e o acompanhamento da evolução do *campus* (Kwok, 2015).

Para Liu (2017, p.919) "*smart campus* é um ambiente integrado de trabalho, estudo e convivência baseado em Internet das Coisas".

Em Giffinger *et al.* (2007) o autor considera cinco campos de ação para cidades inteligentes, sendo eles: Economia, Pessoas, Meio Ambiente, Energia e Mobilidade. No entanto, desconsidera aspectos relacionados à educação em aprendizado, que são inerentes a ambientes como universidades.

(Ng *et al.*, 2010) cita uma estrutura holística de *Campus* inteligente que é ilustrada na Figura 5.

Figura 5 – Pilares da estrutura inteligente do *Campus*



Fonte: Adaptado de Ng *et al.* (2010).

Vários pilares foram definidos na estrutura do *Campus* inteligente observando na Figura 5, como o *iLearning*, *iSocial*, *iGreen*, *iHealth*, *iManagement* e *iGovernance* como aspectos do *Campus*. Desta forma é interessante informar a função de cada um desses aspectos segundo (Hirsch, 2011):

- ***iLearning***: Ajuda os alunos e professores alcançarem o conhecimento em suas atividades. Providencia recursos para a preparação e entrega de material, mas dando mais importância em como os alunos possam adquirir o conhecimento individualmente ou em grupos de vários lugares e a qualquer hora. A educação em nuvem móvel tem uma enorme influência neste pilar.

- **iSocial:** Enfatiza redes sociais e grupos dentro da universidade, permitindo assim relações sociais sem a necessidade de ser só por algum motivo específico que tenham em comum.
- **iManagement:** Cuida da infra-estrutura do *Campus*, do acesso de alunos, segurança e monitoramento.
- **iGreen:** Tem como objetivo tornar o *Campus* mais auto-sustentável, através de energias renováveis e gerenciamento de recursos.
- **iGovernance:** Responsável por questões administrativas do *campus*, pela adaptabilidade e gerenciamento de processos.
- **iHealth:** Concede assistência médica preventiva, remota, sistemas de saúde e monitoramento e alerta de epidemias.

Como exemplo é possível descrever uma situação em que informações de localização enviadas por exemplo do dispositivo do aluno para a rede de dados do *campus* inteligente identifica o aluno e verifica a sua inscrição no sistema permitindo assim o seu acesso. Esse processo se enquadra no pilar *iManagement*. Desta forma permitindo o seu acesso, o aluno pode assim receber os seus horários de aulas ou lembretes de algum assunto específico que seja do âmbito do *campus*. É possível encaixar essa parte no pilar do *iGovernance* (Hirsch, 2011).

Dentro desse contexto, o *Campus* inteligente reúne várias informações emergentes como computação em nuvem, internet das coisas, internet móvel, *big data*, entre outras. Desta forma, o *Campus* inteligente pode ajudar professores e alunos a perceberem o ambiente físico do *Campus*, conhecer o seu estudo e situações de trabalho e características individuais com identificação inteligente. Assim muda-se a maneira como professores e alunos interagem com a universidade, recursos e meio ambiente, percebendo a inovação de serviços personalizados orientada para as pessoas (Liu, 2017).

Desta forma, a característica principal de *Campus* inteligentes é a rápida adaptação do ambiente em relação às demandas, que podem ter diversas origens e contextos diferentes (Ferreira, 2018).

Por exemplo, seria atrativo para um estudante do *Campus* que utiliza os serviços ofertados pela universidade, como o restaurante, saber qual o melhor horário para almoço de acordo com suas tarefas do dia a dia, assim como para o restaurante seria viável saber se existe alguma predominância entre restrições alimentar para que possa ser preparo refeições adequadas. Aos professores, ter acesso a ferramentas que proporcionam um maior aprendizado ao aluno, tanto dentro como fora de sala de aula também é uma preocupação aos *Campus* inteligentes (Ferreira, 2018).

2.2 Infra estrutura tecnológica

A infra estrutura tecnológica necessária para a automatização de ambientes de um campus universitário é constituída por vários fatores que contribuem para o bom funcionamento e segurança do ambiente, desta forma, é viável a utilização da tecnologia de sensores. A IoT surge para calçar esse argumento (Ferreira, 2018).

Sendo assim, para controlar o acesso aos ambientes, realizar a análise de comportamento e automatização de processos administrativos é possível utilizar a tecnologia *Radio-frequency Identification* (Identificação por Rádio Frequência) (RFID). Devido a grande quantidade de dados gerado por essa atividade de sensoriamento e também por essas informações terem que ser armazenadas e processadas em tempo real para retornar respostas rápidas para a tomada de decisões, o conceito de computação em nuvem e Big Data, tecnologia responsável por tratar, analisar e obter informações a partir de conjunto de dados, também são inseridas neste contexto (Ferreira, 2018).

Diante disso, a conectividade é um assunto muito abordado, devido ao fato do crescimento dos dispositivos móveis, a conectividade se torna fundamental para o acesso a serviços e informações. Além disso, várias abordagens teóricas enfatizam a tecnologia sem fio como principal forma de fornecer conectividade, como ferramenta de localização *indoor* e identificação de contexto dos usuários (Hirsch, 2011, p.718-723).

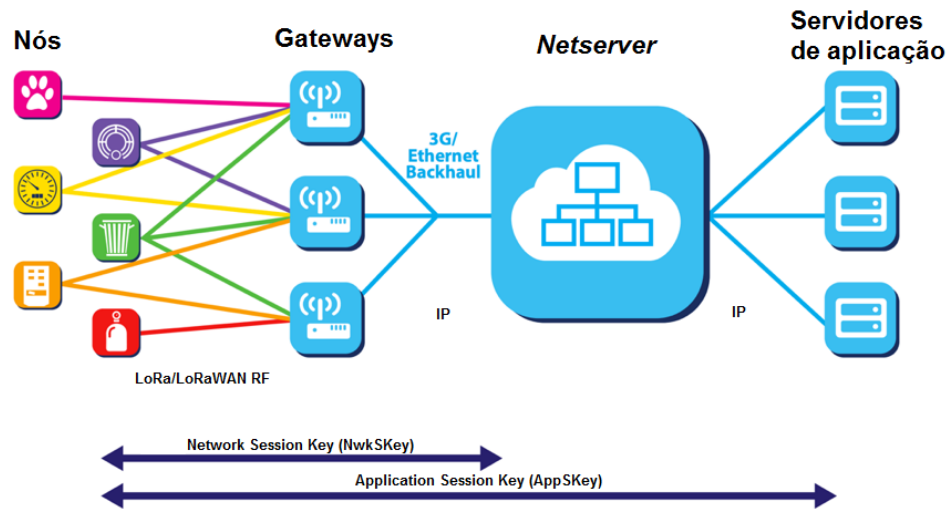
Portanto, conclui-se que construção de um *campus* inteligente com base em *Internet Of Things* (Internet das Coisas) (IoT) e informação em nuvem é uma tendencia inevitável. Mas ainda há algumas questões que devem ser discutidas, e por isso se destaca alguns alertas quanto a implementação de um *campus* inteligente como a motivação da comunidade para utilizar e colaborar com a plataforma, a hesitação do compartilhamento de informação pela comunidade acadêmica ou pelos gestores, a infraestrutura necessária para o desenvolvimento dos conceitos de *smart campus*, a falta de integração entre os setores da universidade, bem como a segurança dos dados (Neves *et al.*, 2017).

2.3 Protocolo LoRa

É uma tecnologia do tipo LPWAN, logo é possível através desta tecnologia conectar pontos distantes e com baixo consumo de energia e de largura de banda (do inglês *bandwidth*) fornecendo uma conexão bidirecional. A largura de banda é a faixa de frequência que o sinal irá ocupar durante a transmissão (Silva, 2017).

Diante disso, os dispositivos baseados na tecnologia *Long Range* (Longo Alcance) (LoRa) se comunicam com os *gateways* que por sua vez realiza a comunicação com servidores web, que recebem, armazenam, processam e transmitem mensagens e permitem uma interface através de aplicativos, como é ilustrado a seguir na Figura 6 (Alliance, 2017).

Figura 6 – Topologia da rede LoRaWAN



Fonte: Alliance (2017).

Na Figura 6, o *gateway* é entendido como uma ponte entre duas redes distintas, ou seja, o *gateway* age como intermediário entre um dispositivo e uma rede com ambos operando com diferentes protocolos, desde que possua um dispositivo que decifre esta comunicação. Isso ocorre em uma rede com topologia estrela onde os dispositivos se comunicam com o *gateway* que estabelece a comunicação entre redes distintas (Silva, 2017).

Para que se torne possível o funcionamento de uma aplicação em uma rede LoRaWAN, os passos a seguir precisam ser respeitados (Santana, 2017):

1. Os *endpoints* precisam estar se comunicando com pelo menos um *gateway* e o mesmo deve ser capaz de comunicar-se com o *Netserver*.
2. O *endpoint* emite periodicamente um ou mais dados em suas respectivas portas, cada porta representando uma aplicação.

Antes do próximo passo, é importante conceituar o termo *Netserver*, DevEUI e APPEUI. O *Netserver* descrito na Figura 6 é um centralizador na arquitetura LoRaWAN, ou seja, gerencia a comunicação entre os dispositivos finais e os aplicativos que desejam receber os dados.

Enquanto isso, o DevEUI é um identificador único globalmente dado a cada dispositivo LoRaWAN, ou seja, ele identifica estritamente um dispositivo na rede e é definido pelo fabricante. O APPEUI identifica uma aplicação na rede e é definido pelo usuário.

3. Após passar pelo *gateway* e *Netserver*, o servidor de aplicação armazena as mensagens e pode agrupá-las do modo que desejar: por dispositivo, utilizando o DevEUI, ou por proprietário do dispositivo, usando o AppEUI. O método de agrupamento varia de acordo com os objetivos de cada usuário da rede e dos dispositivos.

4. Nesta etapa, os conjuntos de mensagens são separados considerando a porta utilizada pelo dispositivo, que identifica a qual aplicação pertence a informação contida na mensagem.
5. Por fim, os dados são processados e compartilhados com os usuários interessados.

De acordo com a tecnologia LoRa, teoricamente é possível que um *gateway* possa receber requisições de vários dispositivos. É importante realçar que se a comunicação acontecer entre dois dispositivos não é necessário o *gateway* que é utilizado apenas para a criação da rede LoRaWAN ou para a comunicação com um servidor da internet (Silva, 2017).

É comum os dispositivos IoT utilizarem a faixa de frequência denominada como Industrial - Scientific - Medical (Industrial - Científica - Médica) (ISM), cada país por sua vez adota suas próprias faixas, no caso do Brasil a faixa de frequência utilizada é a mesma dos Estados Unidos, que fica entre 902,0 MHz a 928,0 MHz, mas com algumas restrições (Silva Junior, 2017):

- 902,0 MHz a 907,5 MHz (ISM - Radiação Restrita);
- 907,5 MHz a 915,0 MHz (Telefonia SMP/STFC) - (Faixa reservada – não disponível);
- 915,0 MHz a 928,0 MHz (ISM - Radiação Restrita);

Segundo Silva Junior (2017) algumas pessoas veem a faixa ISM como livre, mas é importante lembrar que existe a necessidade de homologação dos equipamentos junto a ANATEL, desta forma, Equipamentos LPWAN que operam nestas faixas de frequência devem obedecer a RESOLUÇÃO ANATEL Nº 680, de 27 de Junho de 2017, publicada no DOU em 29 de junho de 2017.

A modulação da tecnologia LoRa é uma opção de esquema de modulação do tipo *Spread Spectrum* (Espalhamento do Espectro). Essa forma de modulação é utilizada quando mais de um usuário necessita compartilhar a mesma banda de frequência, por exemplo, telefones celulares, sistema de alarmes em edifícios e comunicações de controle de tráfego aéreo. A modulação concede que os usuários transmitam ao mesmo tempo e paralelo a isso que a transmissão ocupe todo o espectro. A técnica de Espalhamento Espectral é realizada através dos seguintes métodos: Salto de Frequência (*Frequency Hopping*) e Sequência Direta (*Direct Sequence*) ou então uma junção dos dois métodos chamada de sistema híbrido (Haykin, 2008).

Nesse contexto, tem-se que a tecnologia LoRa utiliza a técnica de sequência direta como parte de seu esquema de espalhamento espectral, o que contribui para sua capacidade de comunicação de longo alcance e resistência a interferências (Haykin, 2008).

A modulação pode ser determinada por três fatores: o *Spreading Factor* (Fator de Espalhamento) (SF) pode ser de 7 a 12. Quanto maior o fator de espalhamento mais informação por bit é transmitida. A largura de banda *Band Width* (Largura de Banda) (BW) é ajustável nos valores: 125, 250 e 500 kHz. Quanto menor a BW, mais sensibilidade a interferências e mais

tempo para transmissão. O parâmetro Código de Erro (CR) que vai de 0 a 4 e determina a taxa do código de erro (Santana, 2017).

Quanto maior o alcance, menor será o *payload* que é a informação útil a ser transmitida, ou seja, sem as informações de endereçamento. Considerando um *payload* de 242 bytes, a taxa de transmissão é de 12500 bits por segundo (bps) e a BW é 500 kHz com um SF igual a 8. Já se a rede for configurada para ter um maior alcance, o SF é de 10 e a BW é de 125 kHz (Silva Junior, 2017).

Com essas informações, conclui-se que a tecnologia LoRa não é compatível com aplicações que requerem transmissão de grandes volumes de dados, como imagens, áudio, dentre outras.

2.4 Protocolo esp-now

É um protocolo de comunicação definido pela Espressif, não faz uso do protocolo TCP e é muito utilizado em projetos que envolve IoT, sem fio e sem a necessidade de se conectar a um roteador, sendo desnecessário o acesso por meio de um SSID e senha. Pode ser implementado juntamente com o microcontrolador que será utilizado no projeto (ESP32) (Cia, 2022).

O protocolo é análogo à conectividade sem fio de baixa potência de 2,4 GHz, amplamente utilizada em mouses sem fio (O.Muniz. *et al.*, 2019).

A comunicação utilizando o protocolo Esp-Now pode ser realizada entre vários módulos, por exemplo, pode-se ter um microcontrolador enviando ou recebendo dados de vários outros módulos ao mesmo tempo (Cia, 2022).

A seguir, são descritos alguns dos recursos deste protocolo, de acordo com (Ludwig, 2020):

- Comunicação Criptografada e não criptografada.
- Conexão limitada a 20 pares não criptografados.
- Função de *callback* para envio da confirmação ou não do recebimento do pacote de dados.
- Limitação no que tange o tamanho da mensagem de até 250 bytes.

Para possibilitar a comunicação entre os dispositivos utilizando o protocolo Esp-Now é necessário conhecer o endereço *Media Access Control* (Controle de Acesso de Mídia) (MAC) dos dispositivos. Após o primeiro emparelhamento, a conexão é persistente, ou seja, mesmo que o dispositivo que esta recebendo as informações seja desligado/reiniciado, a comunicação não é desfeita (Ludwig, 2020).

Para usufruir dos recursos ofertados pelo protocolo Esp-Now, foi utilizado o módulo ESP-01S equipado com o chip ESP8266, também desenvolvido pela Espressif.

2.5 Protocolo mqtt

O protocolo de comunicação MQTT é baseado na arquitetura *publish/subscribe* e possui um modelo cliente/servidor, onde cada sensor é um cliente e se conecta a um servidor, conhecido como *broker*, via TCP (Toby, 2020).

As mensagens são publicadas em endereços, conhecidos como tópicos. Um cliente pode participar de vários tópicos e recebe todas as mensagens publicadas neste tópico. O protocolo MQTT oferece três graus de qualidade de serviço, conforme descrito por (Toby, 2020):

- Entrega no máximo uma vez (entrega não garantida).
- Entrega pelo menos uma vez (entrega garantida, mas pode haver duplicatas).
- Entrega exatamente uma vez (entrega garantida e sem repetir as informações).

O *broker* utilizado no projeto será a plataforma Ubidots, que simplifica a comunicação de dispositivos IoT com a plataforma, facilitando a coleta e análise de dados em tempo real (Koyanagi, 2019).

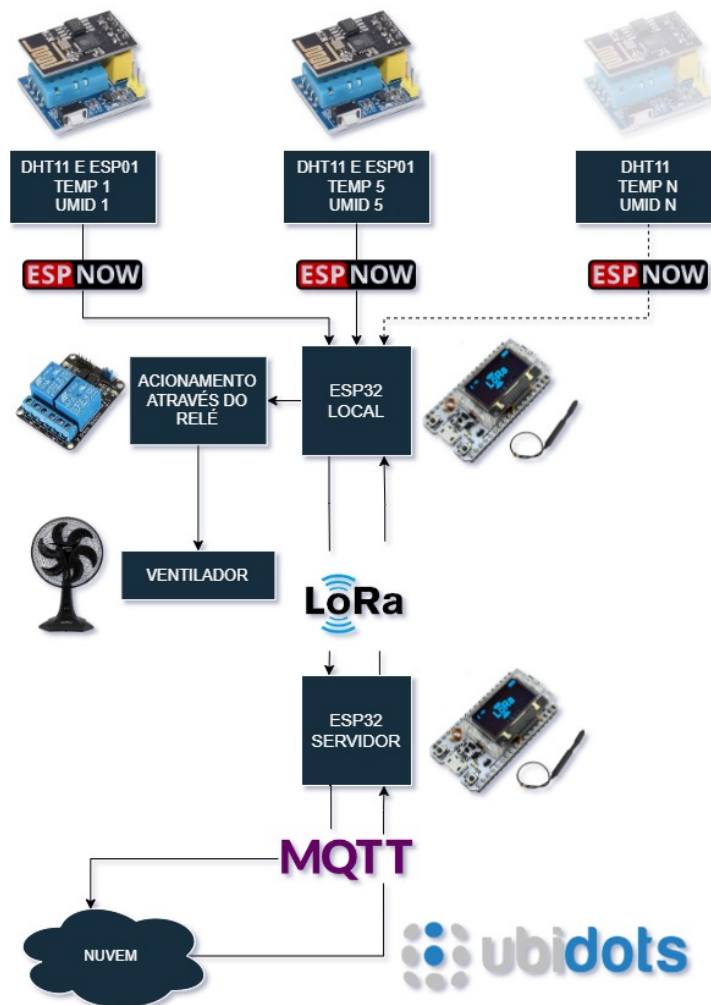
3 METODOLOGIA

De acordo com o embasamento teórico adquirido na pesquisa inicial é possível definir quais os requisitos mínimos necessários para se obter um ambiente automatizado, sendo eles: dispositivos compatíveis, conectividade, plataforma de controle, protocolos de comunicação, fonte de energia e entre outros.

Diante disso, tem-se que o projeto se enquadra em uma automatização, visto que utiliza-se multi protocolos de comunicação entre rede de sensores e microcontroladores, além do mais que, tem-se o acionamento automático de um ventilador com o intuito de atingir o conforto térmico do ambiente.

O trabalho desenvolvido pode ser descrito pelo diagrama de blocos ilustrado na Figura 7, sendo estes baseados em: monitoramento, tratamento de dados, controle e visualização dos resultados.

Figura 7 – Diagrama de Blocos do Projeto

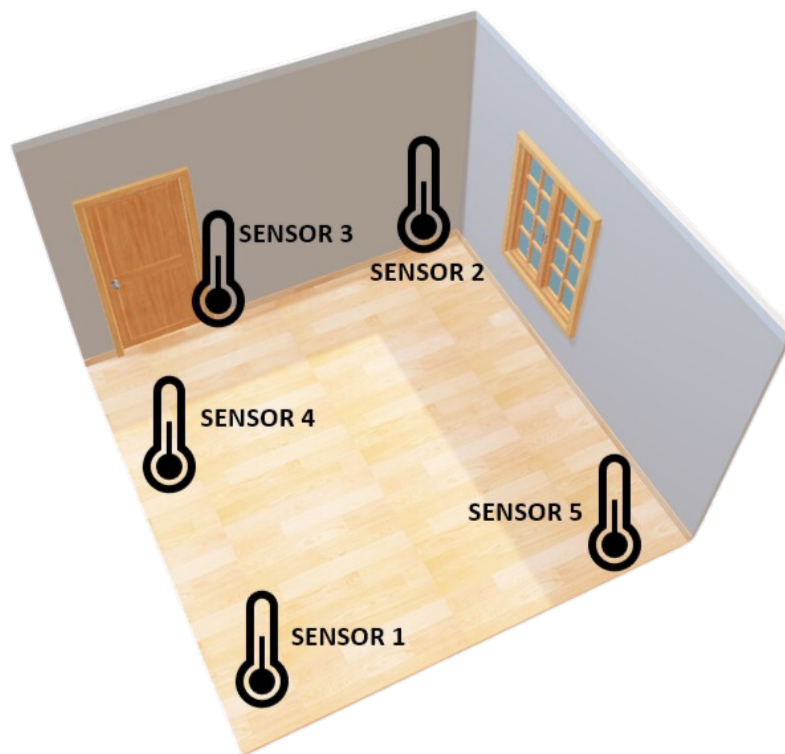


Fonte: Autoria Própria (2023).

Através da Figura 7 é possível descrever os componentes utilizados no projeto, sendo eles: Cinco conjuntos de Sensor DHT11 com ESP-01, dois microcontroladores ESP32 LoRa, um módulo com dois relés, um ventilador e um computador pessoal.

Diante disso, o projeto ilustrado na Figura 7 pode ser implementado em qualquer ambiente, viabilizando várias aplicações. Neste projeto, o ambiente utilizado como teste foi um comodo residencial, onde foram distribuídos os 5 conjuntos de sensores DHT11 com ESP-01 em pontos distintos deste comodo. No dia do teste havia um fluxo de no máximo quatro pessoas na residência, e dentre as quatro pessoas, duas permaneceram no comodo por mais tempo. O ESP32 "Local" estava cerca de 4 metros de distância dos sensores e a 2 metros de distância do ESP32 "Servidor". É importante ressaltar que entre os dispositivos havia uma parede de concreto. Essa simulação é ilustrada na Figura 8

Figura 8 – Distribuição dos Sensores DHT11-ESP01 no comodo utilizado na simulação.

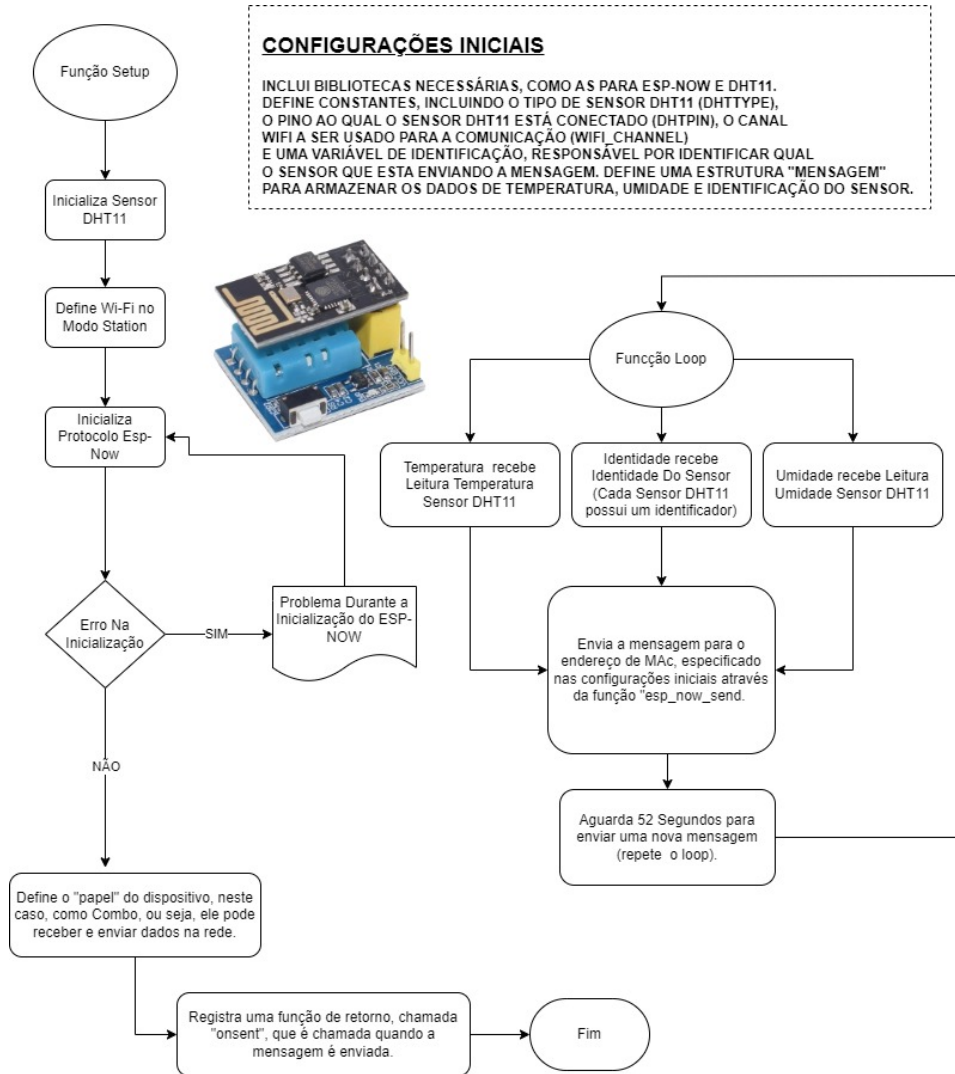


Fonte: Aatoria Própria (2023).

De acordo com a Figura 7, inicialmente tem-se o sensor DHT11 obtendo as medições de temperatura e umidade, que são enviadas via protocolo Esp-Now, através do módulo Wi-Fi acoplado no sensor (ESP-01), para o ESP32 "Local", que fará o tratamento das leituras de temperatura e umidade e avaliação referente ao acionamento do ventilador através do relé. Em seguida, os dados tratados no ESP32 "Local" serão enviados para o ESP32 "Servidor", através do protocolo LoRa e por fim, os dados serão enviados para nuvem com o intuito de facilitar a visualização dos resultados através de uma plataforma denominada Ubidots.

A seguir é possível observar o fluxograma referente à lógica de programação do processo referente a leitura da temperatura e umidade e do envio destas informações para o ESP32 "Local".

Figura 9 – Fluxograma da Captura e Envio da Medição da Temperatura e Umidade

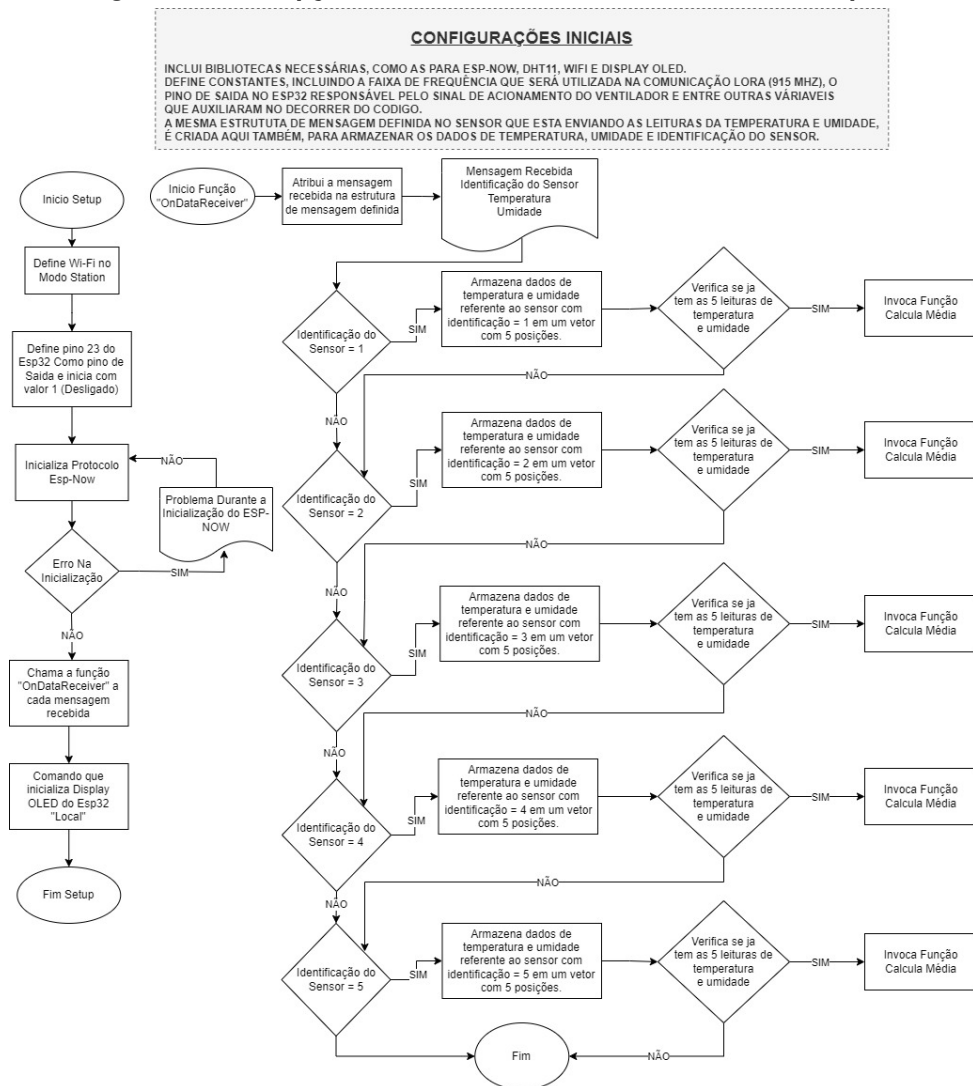


Fonte: Autoria Própria (2023).

O fluxograma ilustrado na Figura 9 é referente ao algoritmo implementado no ESP-01, através de um conversor USB Serial RS232. Após a gravação do código no ESP-01, o mesmo é acoplado ao sensor DHT11, desta forma, o algoritmo é executado, inicializando o sensor DHT11 e o protocolo de comunicação Esp-Now. É criado dentro do código uma estrutura de mensagem (*Struct*), que possui três variáveis, sendo elas, temperatura, umidade e identificação. A variável identificação é fixa para cada sensor, ou seja, tem-se que o valor da variável de identificação será entre 1 e 5, pois neste caso, tem-se 5 sensores DHT11. Diante disso, o sensor DHT11 inicia a operação capturando as leituras de temperatura e umidade, após isso, o protocolo Esp-Now entra em ação, enviando as leituras e identificação do sensor para o ESP32 "Local", através do protocolo Esp-Now.

Como o protocolo Esp-Now é uma comunicação *Peer-to-peer* (Ponto a ponto) (P2P) é repassado na programação o endereço do dispositivo ESP32 "Local". A leitura da temperatura e umidade é realizada de período em período (No sensor, no qual a variável identificação é igual a 1 as leituras são realizadas a cada 52 segundos, no sensor 2 a cada 54 segundos, no sensor 3 a cada 56 segundos, no sensor 4 a cada 58 segundos e no sensor 5 a cada 60 segundos), evitando o envio de informações no mesmo instante para o ESP32 "Local". A recepção dos dados no ESP32 "Local" é descrito no fluxograma ilustrado na Figura 10.

Figura 10 – Fluxograma da Recepção e Armazenamento das Leituras de Temperatura e Umidade.



Fonte: Autoria Própria (2023).

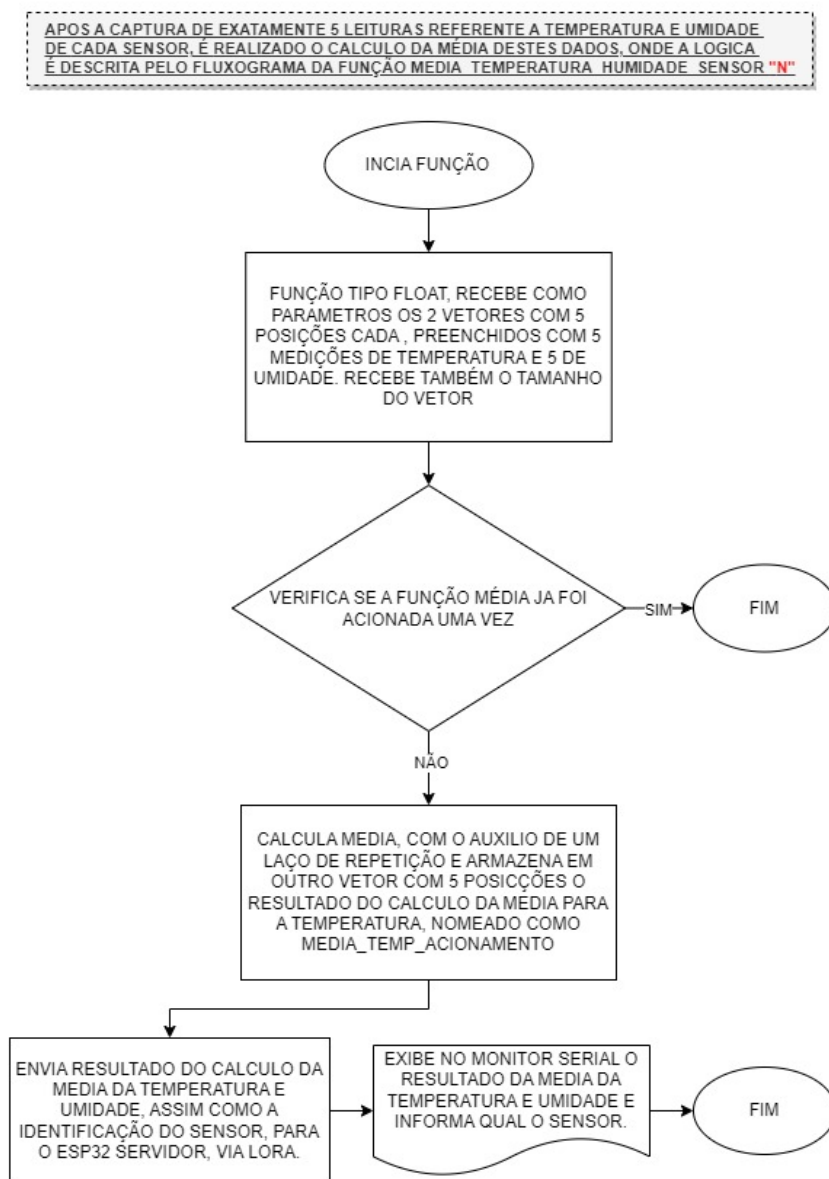
Na etapa ilustrada na Figura 10, o código descreve a seguinte operação: as leituras de temperatura e umidade e identificação do sensor são recebidas em uma estrutura de mensagem idêntica à que foi criada no algoritmo do ESP-01. Recebendo essas informações, é verificado de qual sensor se trata as leituras recebidas. Sabendo qual o sensor que enviou as informações, é atribuído o valor das leituras de umidade e temperatura em novas variáveis, com um identificador no nome destas variáveis que indicam de qual sensor foram recebidas estas informações. Essa

variável é um vetor, com 5 posições. Após o armazenamento das leituras de temperatura e umidade é verificado se o vetor com 5 posições já recebeu as 5 leituras de temperatura e umidade, se sim, é invocada a função responsável pelo cálculo das médias, caso contrário, o código continua verificando de qual sensor é a leitura de temperatura e umidade e armazenando na sua respectiva variável.

Antes de ilustrar a figura com o algoritmo da função responsável por realizar o cálculo da média é importante ressaltar que esse cálculo é realizado com o intuito de obter uma maior gama de dados para a tomada de decisão no que tange o acionamento do ventilador.

Desta forma, o cálculo da média é ilustrado na Figura 11:

Figura 11 – Diagrama de Blocos da Função que Calcula a Media das Temperaturas e Umidades.



Fonte: Autoria Própria (2023).

Conforme ilustrado na Figura 11, a função calcula média, recebe como parâmetro o vetor de 5 posições contendo as 5 leituras de temperatura, o vetor contendo as 5 leituras de

umidade e o tamanho do vetor. Com estas informações a função realiza o cálculo da média, com o auxílio de um laço de repetição e armazena o resultado em outro vetor que também possui 5 posições. Entretanto, se um sensor parasse de enviar as informações por algum motivo, algum dos demais sensores teria o cálculo da média realizado mais de uma vez, o que não pode acontecer, pois o intuito do projeto é monitorar um ambiente em pontos distintos. Para tratar este problema, é incrementada uma variável a cada vez que a função média é executada, sendo assim, garante-se que a cada rodada, todos os sensores terão enviado apenas uma média das temperaturas e umidades que foram lidas. Após o cálculo da média da temperatura e umidade a comunicação LoRa é requisitada, para enviar estas informações ao ESP32 "Servidor" e é exibido na tela do monitor serial da *Integrated Development Environment* (Ambiente de Desenvolvimento Integrado) (IDE) o valor da média da temperatura, umidade e a identificação do sensor. Na Figura 12 é ilustrado as informações que são apresentadas no Monitor Serial do ESP32 "Local".

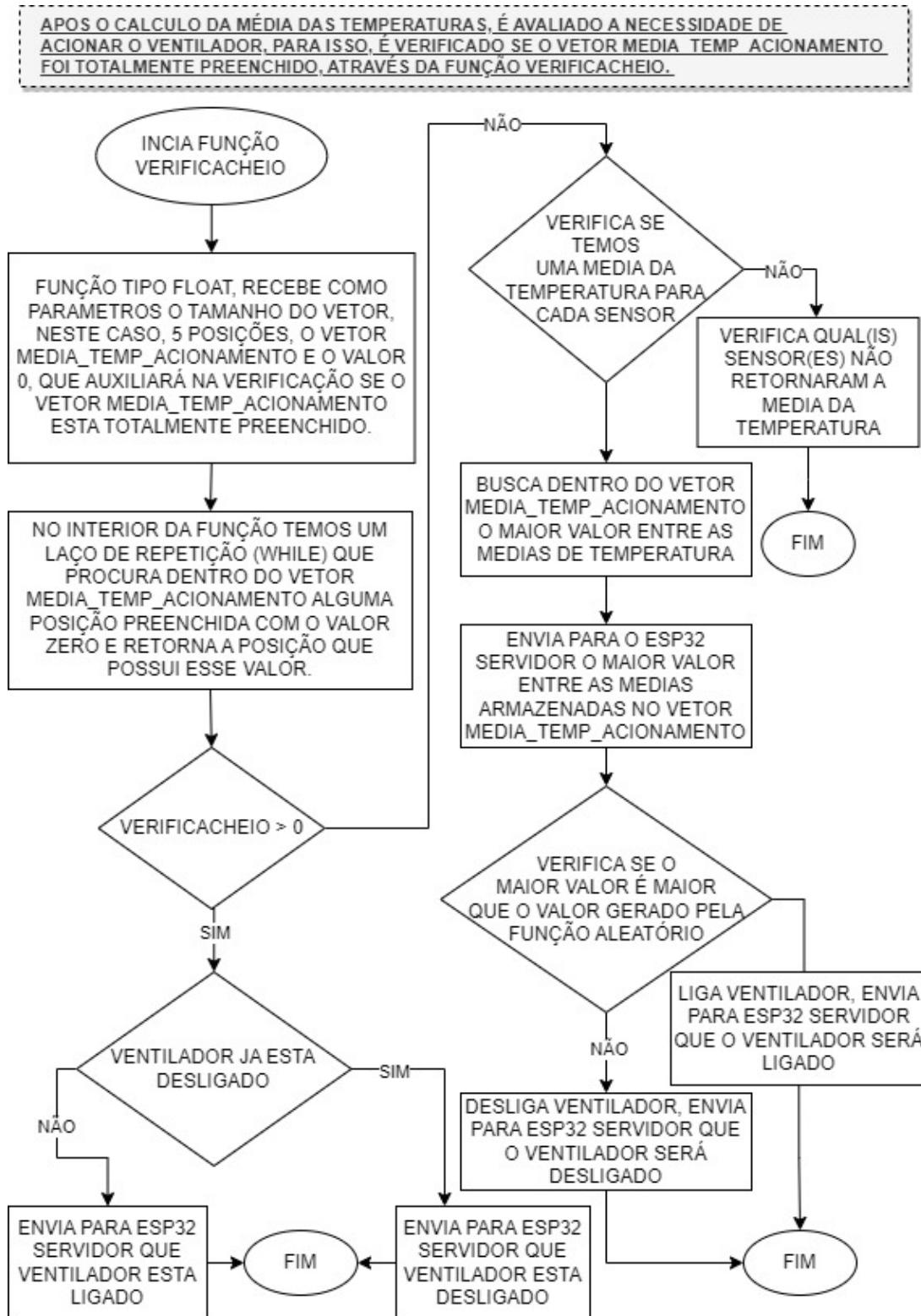
Figura 12 – Exemplo Saída Monitor Serial da IDE

Saída	Monitor Serial x	Saída	Monitor Serial x	Saída	Monitor Serial x
Mensagem (WiFi LoRa 32(V2) + Enter		Mensagem (WiFi LoRa 32(V2) + Enter		Mensagem (WiFi LoRa 32(V2) + Enter para env	
Identificação do Sensor: 1 Temperatura: 24.50 Umidade: 62.20		mensagem recebida Identificação do Sensor: 1 Temperatura: 29.40 Umidade: 46.80		Identificação do Sensor: 1 Temperatura: 29.50 Umidade: 47.00	
Posição Vetor: 1 Temperatura_Sensor1: 24.50 Humidade_Sensor1: 62.20		Posição Vetor: 3 Temperatura_Sensor1: 29.40 Humidade_Sensor1: 46.80		Posição Vetor: 5 Temperatura_Sensor1: 29.50 Humidade_Sensor1: 47.00	
Saída	Monitor Serial x	Saída	Monitor Serial x		
Mensagem (WiFi LoRa 32(V2) + Enter		Mensagem (WiFi LoRa 32(V2) + Enter		Media Temperatura_Sensor_1: 29.42 Media Humidade_Sensor_1: 49.94	
Identificação do Sensor: 1 Temperatura: 29.20 Umidade: 46.90		Identificação do Sensor: 1 Temperatura: 29.50 Umidade: 46.80			
Posição Vetor: 2 Temperatura_Sensor1: 29.20 Humidade_Sensor1: 46.90		Posição Vetor: 4 Temperatura_Sensor1: 29.50 Humidade_Sensor1: 46.80			

Fonte: Autoria Própria (2023).

Após o preenchimento do vetor, responsável por armazenar o resultado do cálculo da média da temperatura de cada sensor, é possível avaliar a necessidade de acionar o resfriador, neste caso, o ventilador. Essa avaliação é ilustrado na Figura 13.

Figura 13 – Fluxograma do algoritmo responsável pela avaliação do acionamento do ventilador.



Fonte: Autoria Própria (2023).

O fluxograma ilustrado acima é o processo final do algoritmo implementado no ESP32 "Local". Este processo, consiste na avaliação, de acordo com os dados recebidos dos sensores DHT11 e do cálculo da média das temperaturas, da necessidade do acionamento do ventilador. Antes desta avaliação, é verificado se o vetor responsável por armazenar o resultado do cálculo

das médias das temperaturas está totalmente preenchido, através de uma função denominada "VerificaCheio". Esta função recebe como parâmetro o vetor com o resultado das médias das temperaturas, o valor 0, que será o procurado dentro do vetor, com o intuito de verificar se alguma posição ainda não foi preenchida e o tamanho do vetor. Com estas informações e com o auxílio de um laço de repetição é verificado se possui dentro do vetor, passado como parâmetro, alguma posição preenchida com o valor 0, se sim, a função retornará qual a posição que possui esse valor. Caso o retorno da função "VerificaCheio" seja maior que 0, é verificado se o ventilador já está desligado, se sim é enviado ao ESP32 "Servidor", que o ventilador permanecerá desligado ou caso contrário, é enviado que ele permanecerá ligado.

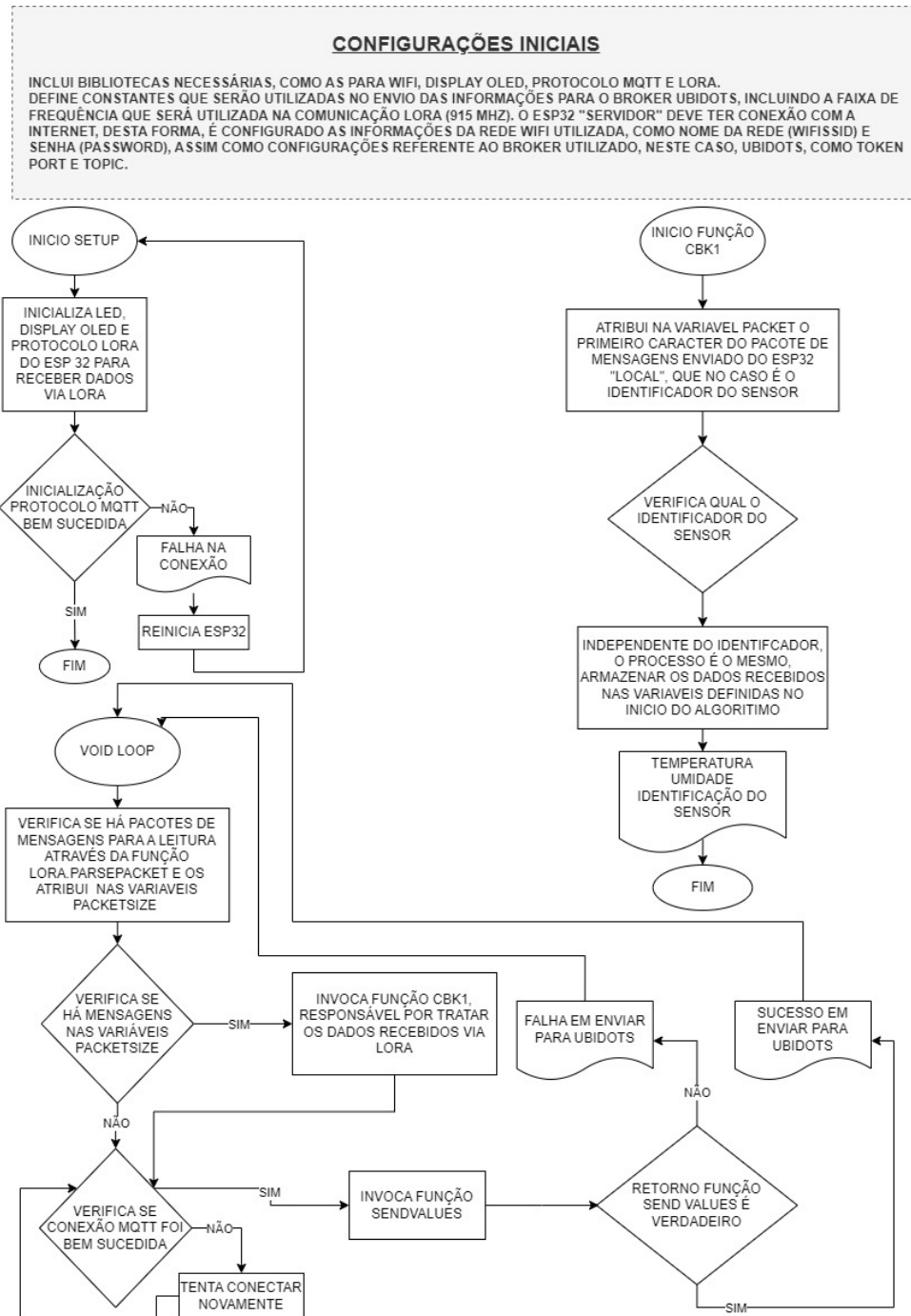
Caso o retorno da função "VerificaCheio" for 0, é verificado se há uma média da temperatura para cada sensor, se sim, inicia-se a busca pelo maior valor entre as médias das temperaturas e após encontrar o maior valor é verificado se esse valor é maior do que o valor gerado pela função "calaleatório", que é a função responsável por gerar um valor de temperatura entre 25°C e 40°C, com o intuito de simular a temperatura do ambiente. Essa função foi implementada para facilitar os testes, não sendo necessário gravar o código novamente no ESP32 "Local". Diante disso, se o maior valor entre as médias das temperaturas for maior que o valor gerado pela função "calaleatorio", o ventilador é acionado e é enviada uma informação para o ESP32 "Servidor" de que o ventilador será acionado, caso contrário, o ventilado é desligado e essa informação é enviada ao ESP32 "Servidor", por meio da variável "StatusVentilador".

Enfim, todas as informações foram coletadas e enviadas do ESP32 "Local" para o ESP32 "Servidor", por meio da comunicação LoRa.

No ESP32 "Servidor" também há alguns algoritmos de programação, responsáveis por Receber os dados do ESP32 "Local" via LoRa, tratar as informações recebidas e enviá-las para o Broker, neste caso, Ubidots, via MQTT.

Na Figura 14, é ilustrado o fluxograma do recebimento das informações enviadas pelo ESP32 "local", assim como o tratamento destas informações no ESP32 "Servidor".

Figura 14 – Diagrama de Blocos da Recepção e Tratamento dos Dados no ESP32 "Servidor".



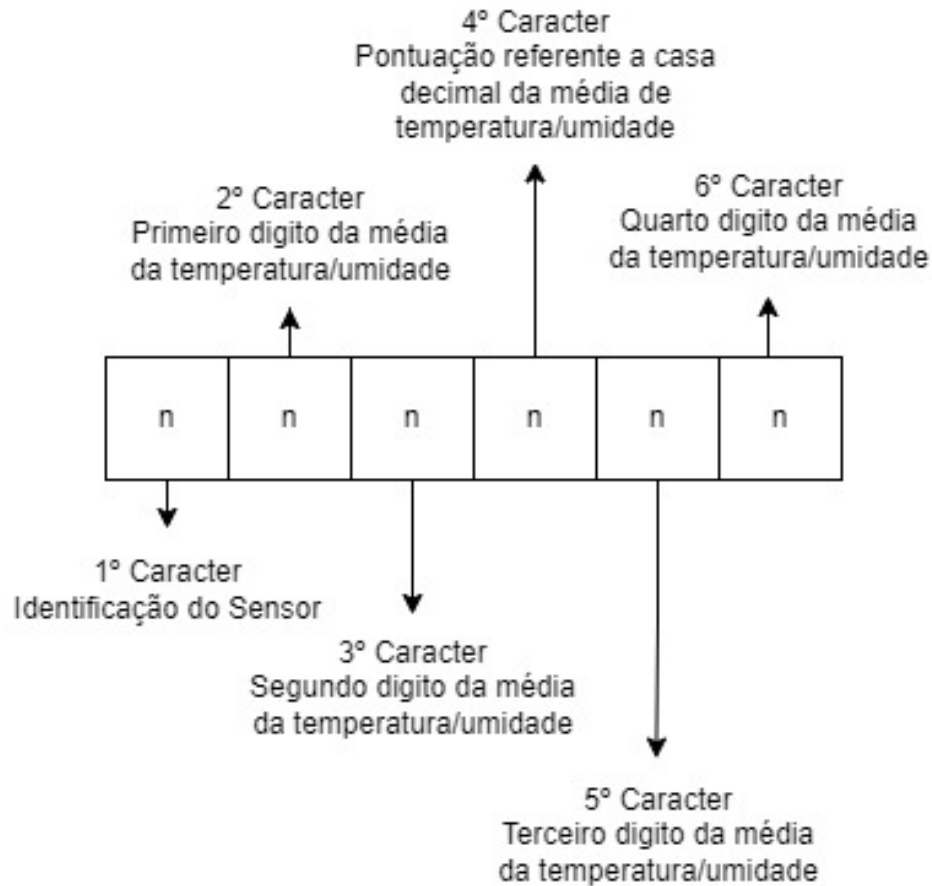
Fonte: Autoria Própria (2023).

No fluxograma ilustrado acima, tem-se o algoritmo de recebimento e tratamento das informações. O primeiro passo é habilitar o ESP32 "Servidor" para receber informações via LoRa e depois inicializar o protocolo MQTT, que será responsável pela comunicação entre o ESP32 "Servidor" e a nuvem.

Na função "Loop", verifica-se se existem pacotes de mensagens para a leitura, recebidos através do protocolo LoRa. Caso seja encontrado pacotes de mensagens, a função "cbk" é invocada. Esta função é responsável por atribuir nas variáveis criadas no algoritmo as infor-

mações recebidas via LoRa. O conteúdo do pacote de dados enviado do ESP32 "Local" para o ESP32 "Servidor" é ilustrado na Figura 15

Figura 15 – Exemplo do conteúdo do pacote de dados enviado via LoRa



Fonte: Autoria Própria (2023).

Dado isso, cada pacote de dados enviado é recepcionado e tratado conforme ilustrado na Figura 14, ressaltando que a única variável que receberá informações referente a todos os sensores é a variável "packet", pois ora vai receber o valor 1, que se refere aos dados do sensor denominado como 1 e ora vai receber 2, que se refere aos dados do sensor denominado como 2 e assim para os demais sensores.

Após esse tratamento de dados, é verificado se a conexão MQTT foi bem sucedida, caso não, a tentativa de se conectar persiste até que tenha sucesso. Caso contrário, é invocada a função "SendValues", responsável por enviar as informações ao Ubidots.

Por fim, chegou-se na etapa final do projeto, onde tem-se os resultados ilustrados no site do Ubidots.

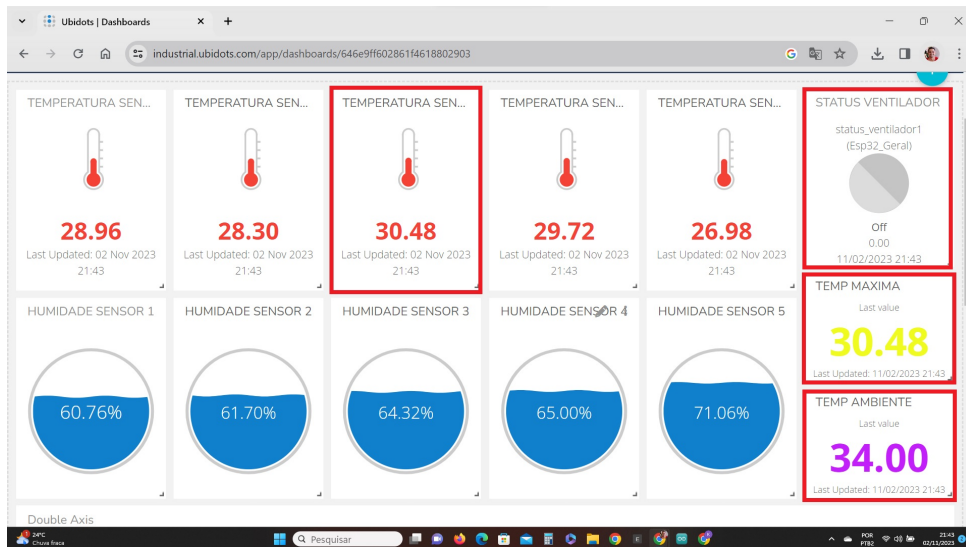
4 RESULTADOS E DISCUSSÕES

O propósito inicial do projeto foi atendido, através do êxito no envio das informações pelo protocolo LoRa, pela rede de sensores sem fio que obtêm as medições das grandezas temperatura e umidade e o envio dos resultados para a nuvem, através do protocolo MQTT.

Na figura Figura 16 e Figura 17 tem-se os resultados dos algoritmos implementados nos dispositivos, descritos nas etapas anteriores, sendo eles:

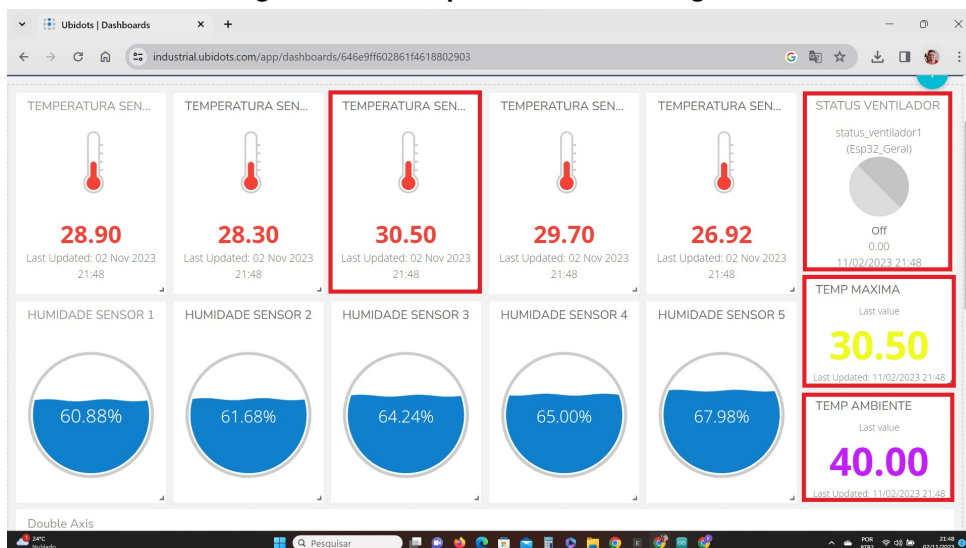
- Média das 5 leituras de temperatura, obtidas através do sensor DHT11, referente ao sensor 1;
- Média das 5 leituras de temperatura, obtidas através do sensor DHT11, referente ao sensor 2;
- Média das 5 leituras de temperatura, obtidas através do sensor DHT11, referente ao sensor 3;
- Média das 5 leituras de temperatura, obtidas através do sensor DHT11, referente ao sensor 4;
- Média das 5 leituras de temperatura, obtidas através do sensor DHT11, referente ao sensor 5;
- Média das 5 leituras de umidade, obtidas através do sensor DHT11, referente ao sensor 1;
- Média das 5 leituras de umidade, obtidas através do sensor DHT11, referente ao sensor 2;
- Média das 5 leituras de umidade, obtidas através do sensor DHT11, referente ao sensor 3;
- Média das 5 leituras de umidade, obtidas através do sensor DHT11, referente ao sensor 4;
- Média das 5 leituras de umidade, obtidas através do sensor DHT11, referente ao sensor 5;
- Variável "Status do Ventilador", ou seja, se está ligado (1) ou desligado (0);
- Variável "Maior", representando o maior valor entre as médias de temperatura;
- Variável "aleatório", representando a temperatura ambiente;

Figura 16 – Exemplo Ventilador Desligado.



Fonte: Autoria Própria (2023).

Figura 17 – Exemplo Ventilador Desligado.



Fonte: Autoria Própria (2023).

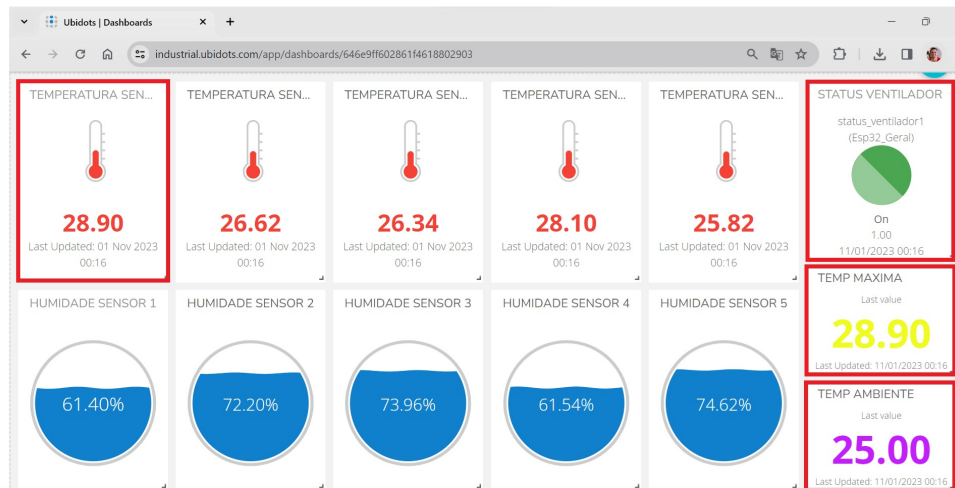
Diante disso, nota-se que nestes casos ilustrados nas Figura 18 e Figura 19 o projeto esta operando conforme o esperado, ou seja, as leituras de temperatura e umidade estão sendo coletadas pela rede de sensores, compartilhadas entre os dispositivos através dos protocolos de comunicação e disponibilizadas em um site. A visualização no site é a prova do sucesso no que tange o funcionamento do projeto através dos algoritmos de programação.

Usando como exemplo a Figura 16, tem-se o valor de 30,48°C, que é o maior valor entre as médias da temperatura não é maior do que o valor de 34,00°C, que representa a temperatura ambiente, lembrando que esse valor de 34,00°C é gerado pela função "calaleatorio". Diante disso, é verificado se o valor de 30,48°C (Ponto mais quente no ambiente) é maior do que o valor de 34,00°C (temperatura ambiente) e como isso não é verdade, o ventilado permanece desligado, pois entende-se que o ambiente não precisa de resfriamento.

Isso ocorre também no caso ilustrado da Figura 17. o Ventilador somente será ligado, se na próxima interação o valor representado pela variável "TEMP MÁXIMA" for maior que o valor representado pela variável "TEMP AMBIENTE".

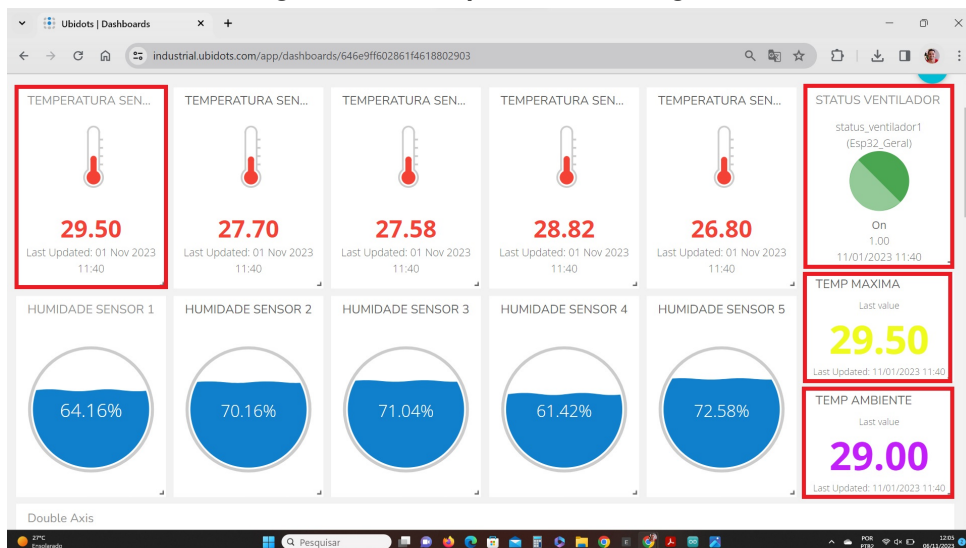
Na Figura 18 e Figura 19, acontece o inverso do ilustrado e descrito anteriormente, ou seja, o ventilador é acionado.

Figura 18 – Exemplo Ventilador Ligado.



Fonte: Autoria Própria (2023).

Figura 19 – Exemplo Ventilador Ligado.



Fonte: Autoria Própria (2023).

Neste caso, como pode-se observar que o valor de "TEMP MAXIMA" é maior do que o valor de "TEMP AMBIENTE", sendo assim, o ventilador é acionado, pois entende-se que esta muito calor no ambiente que esta sendo monitorado.

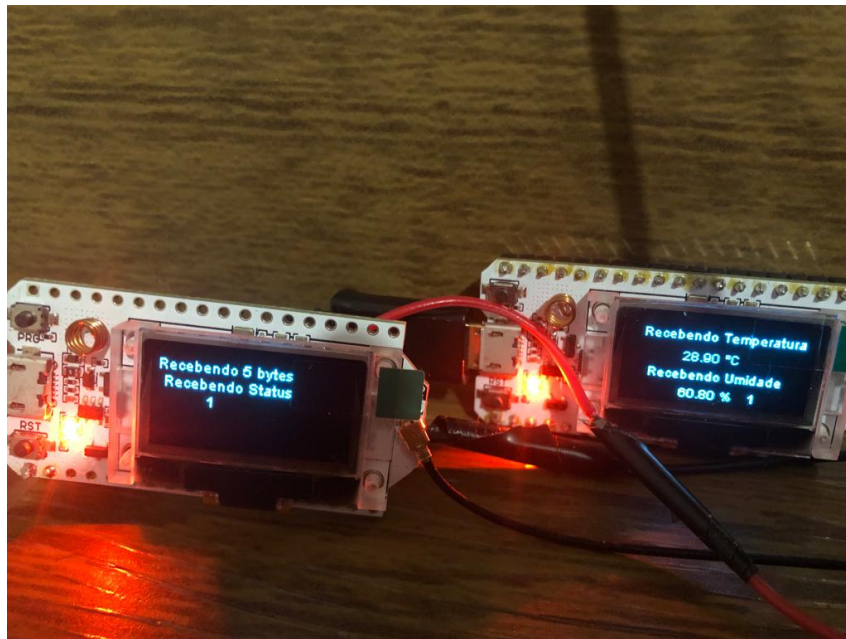
Na Figura 19, a diferença entre as variáveis "TEMP MAXIMA" e "TEMP AMBIENTE" é de apenas 0,5°C, e mesmo assim a comparação nos retorna o resultado correto, neste caso,

o acionamento do ventilador. Conclui-se com essa situação, que a tomada de decisão será assertiva.

A verificação do acionamento do ventilador é realizado quando se tem as 5 médias das temperaturas calculadas, só assim tem-se o valor maior entre as médias e também a invocação da função "calaleatorio", que gera o valor que é representado pela variável "TEMP AMBIENTE". Entretanto, no projeto tem-se a opção de ligar ou desligar o ventilador "manualmente", alterando o valor da variável "STATUS VENTILADOR", no site do Ubidots, desta forma, o ESP32 "Servidor" verifica se alguma variável sofreu alteração, através de uma função pertencente a biblioteca do protocolo MQTT, denominada "callback" e atualiza o valor da variável "StatusVentilador", que será transmitida via LoRa para o ESP32 "Local" e por fim, o ventilador terá sido ligado ou desligado "manualmente".

Na Figura 20, Figura 21, Figura 22, Figura 23 e Figura 24 é ilustrado o momento em que o ESP32 "Local" esta recebendo as leituras de temperatura e umidade do sensor DHT11 e enviando para o ESP32 "Servidor" o "Status Ventilador", ou seja, se o ventilador está ligado ou desligado, além disso, é exibido no Display do ESP32 "Servidor" o tamanho do pacote que é recebido.

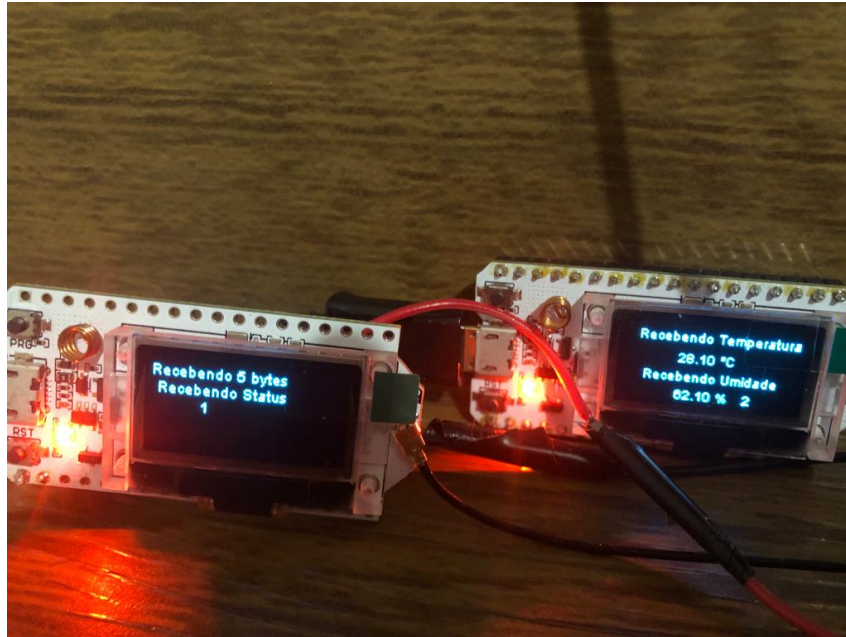
Figura 20 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 1.



Fonte: Autoria Própria (2023).

Na Figura 20 é ilustrado o momento em que o ESP32 "Local" (dispositivo da direita na ilustração) recebe as leituras de temperatura e umidade do sensor 2. O dispositivo a esquerda na ilustração é o ESP32 "Servidor", que por sua vez, informa através de seu *display* que esta recebendo Status 1, ou seja, o ventilador está ligado.

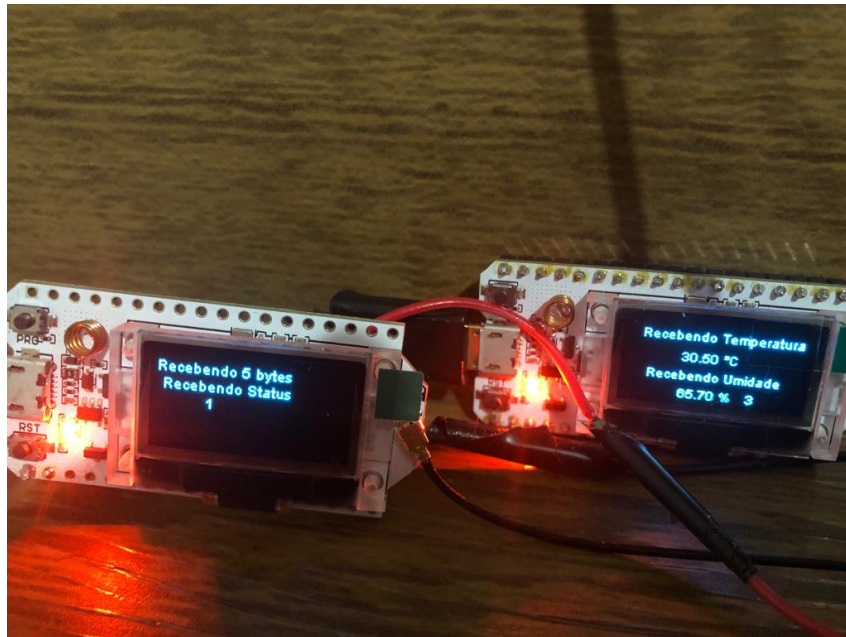
Figura 21 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 2.



Fonte: Aatoria Própria (2023).

Na Figura 21 é ilustrado o momento em que o ESP32 "Local" (dispositivo da direita na ilustração) recebe as leituras de temperatura e umidade do sensor 3. O dispositivo a esquerda na ilustração é o ESP32 "Servidor", que por sua vez, informa através de seu *display* que esta recebendo Status 1, ou seja, o ventilador está ligado.

Figura 22 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 3.

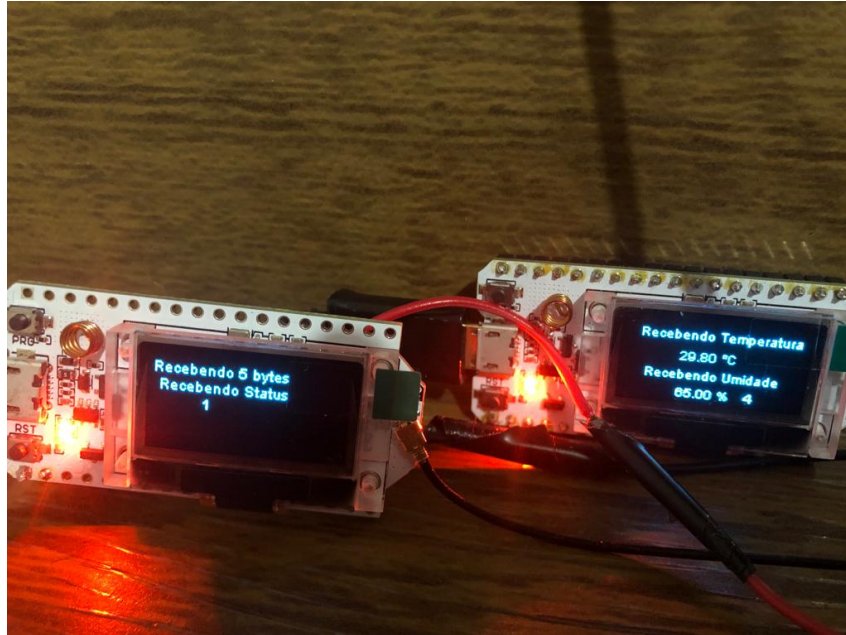


Fonte: Aatoria Própria (2023).

Na Figura 22 é ilustrado o momento em que o ESP32 "Local" (dispositivo da direita na ilustração) recebe as leituras de temperatura e umidade do sensor 4. O dispositivo a esquerda

na ilustração é o ESP32 "Servidor", que por sua vez, informa através de seu *display* que esta recebendo Status 1, ou seja, o ventilador está ligado.

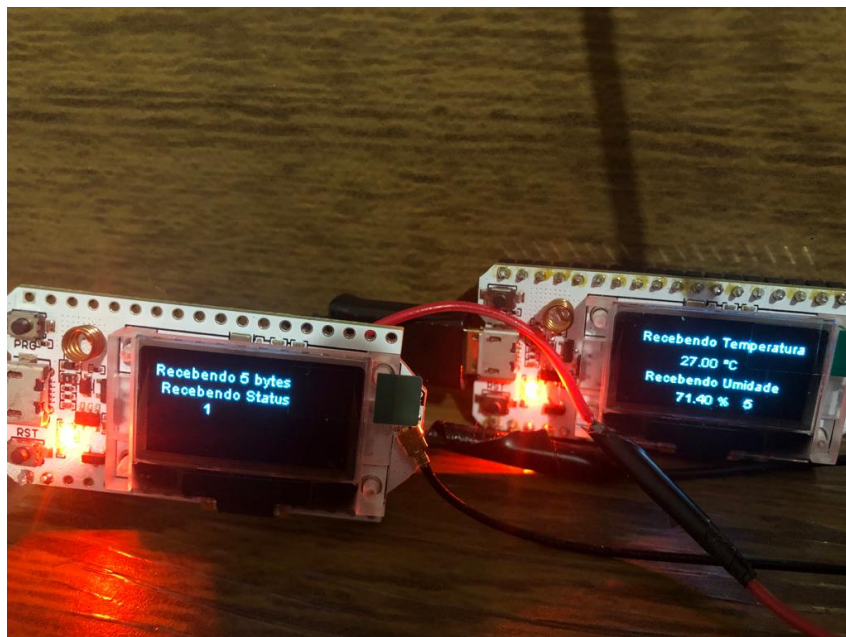
Figura 23 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 4.



Fonte: Aatoria Própria (2023).

Na Figura 23 é ilustrado o momento em que o ESP32 "Local" (dispositivo da direita na ilustração) recebe as leituras de temperatura e umidade do sensor 4. O dispositivo a esquerda na ilustração é o ESP32 "Servidor", que por sua vez, informa através de seu *display* que esta recebendo Status 1, ou seja, o ventilador está ligado.

Figura 24 – Recebendo Temperatura e Umidade do Sensor DHT11, nomeado como Sensor 5.

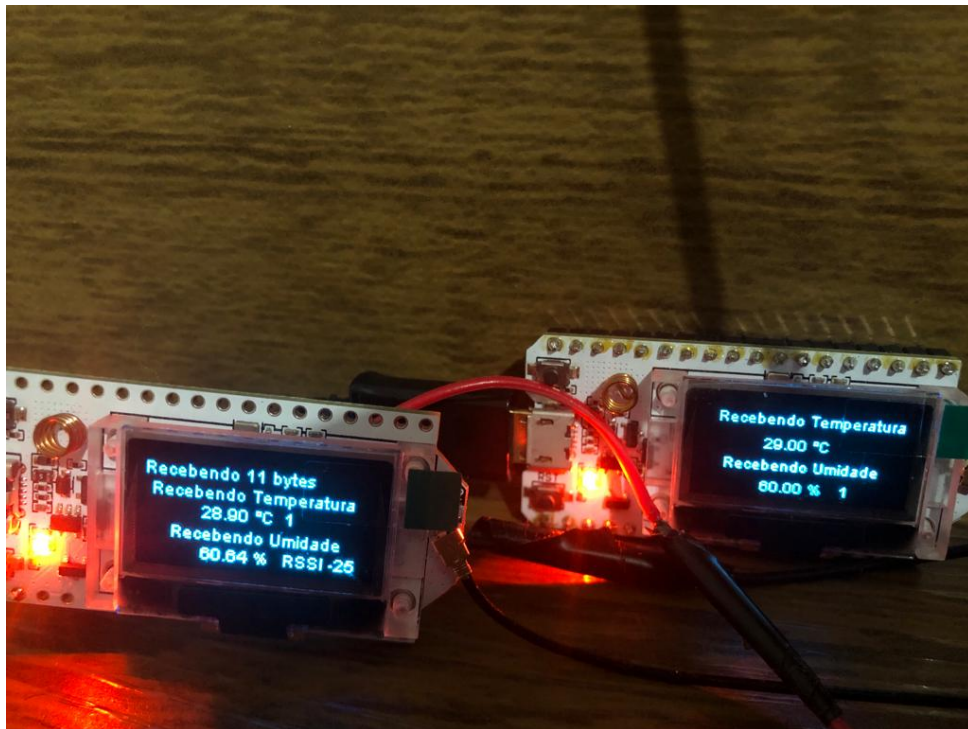


Fonte: Aatoria Própria (2023).

Na Figura 24 é ilustrado o momento em que o ESP32 "Local" (dispositivo da direita na ilustração) recebe as leituras de temperatura e umidade do sensor 5. O dispositivo a esquerda na ilustração é o ESP32 "Servidor", que por sua vez, informa através de seu *display* que esta recebendo Status 1, ou seja, o ventilador está ligado.

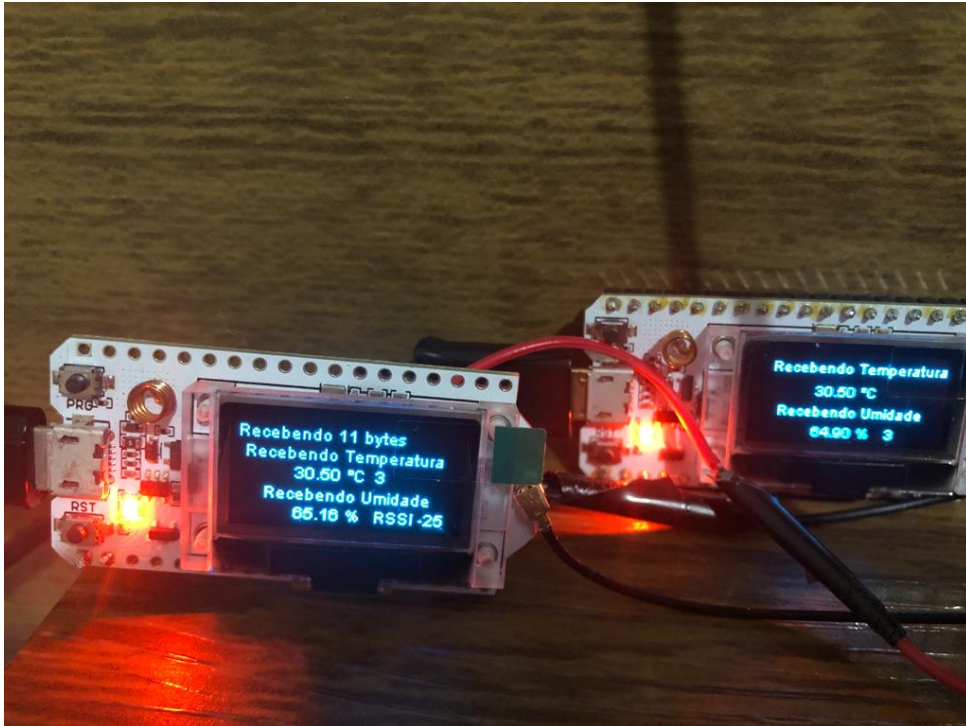
O display do ESP32 "Servidor", na maior parte do tempo, apresenta a mensagem ilustrada nas Figura 20, Figura 21, Figura 22, Figura 23 e Figura 24, pois o algoritmo implementado no ESP32 "Local", verifica a cada repetição qual o estado do ventilador, mas na Figura 25 e Figura 26, é ilustrado o momento em que o ESP32 "Servidor" recebe a média da temperatura e da umidade do sensor 1, o tamanho do pacote da mensagem (1 bit referente a identificação do sensor + 5 bits referente a média da temperatura + 5 bits referente a média da umidade = 11 bits) e o valor do *Received Signal Strength Indicator* (Indicador de intensidade do sinal recebido) (RSSI), como a definição indica é a intensidade do sinal recebido relacionado à potência do sinal transmitido, neste caso, -25 dBm.

Figura 25 – Recebendo Média da Temperatura e Umidade do Sensor 1, calculada no algoritmo implantado no ESP32 "Local".



Fonte: Autoria Própria (2023).

Figura 26 – Recebendo Média da Temperatura e Umidade do Sensor 3, calculada no algoritmo implantado no ESP32 "Local".



Fonte: Autoria Própria (2023).

Nas Figura 25 e Figura 26, os dados de temperatura e umidade que são exibidos no *display* do ESP32 "Local" (dispositivo à direita), são as leituras do sensor DHT11, enquanto os dados exibidos no *display* do ESP32 "Servidor" (dispositivo à esquerda), são os resultados referente ao cálculo da média da temperatura e da umidade, ou seja, o valor exibido no ESP32 "Servidor" já é o dado tratado. É importante ressaltar, pois pode-se confundir que o valor exibido no ESP32 "Local" dever ser o mesmo do ESP32 "Servidor", o que não é verdade, pois cada um exibe uma etapa do algoritmo.

Como é possível observar na Figura 25 e Figura 26, os dois microcontroladores estão bem próximos. Foi realizado também o teste com os dois componentes mais afastados (aproximadamente 3 metros) e obteve-se os resultados ilustrado na Figura 27, Figura 28 e Figura 29.

Figura 27 – Recebendo Média da Temperatura e Umidade do Sensor 1, calculada no algoritmo implantado no ESP32 "Local".



Fonte: Autoria Própria (2023).

Figura 28 – Recebendo Média da Temperatura e Umidade do Sensor 2, calculada no algoritmo implantado no ESP32 "Local".



Fonte: Autoria Própria (2023).

Figura 29 – Recebendo Média da Temperatura e Umidade do Sensor 4, calculada no algoritmo implantado no ESP32 "Local".

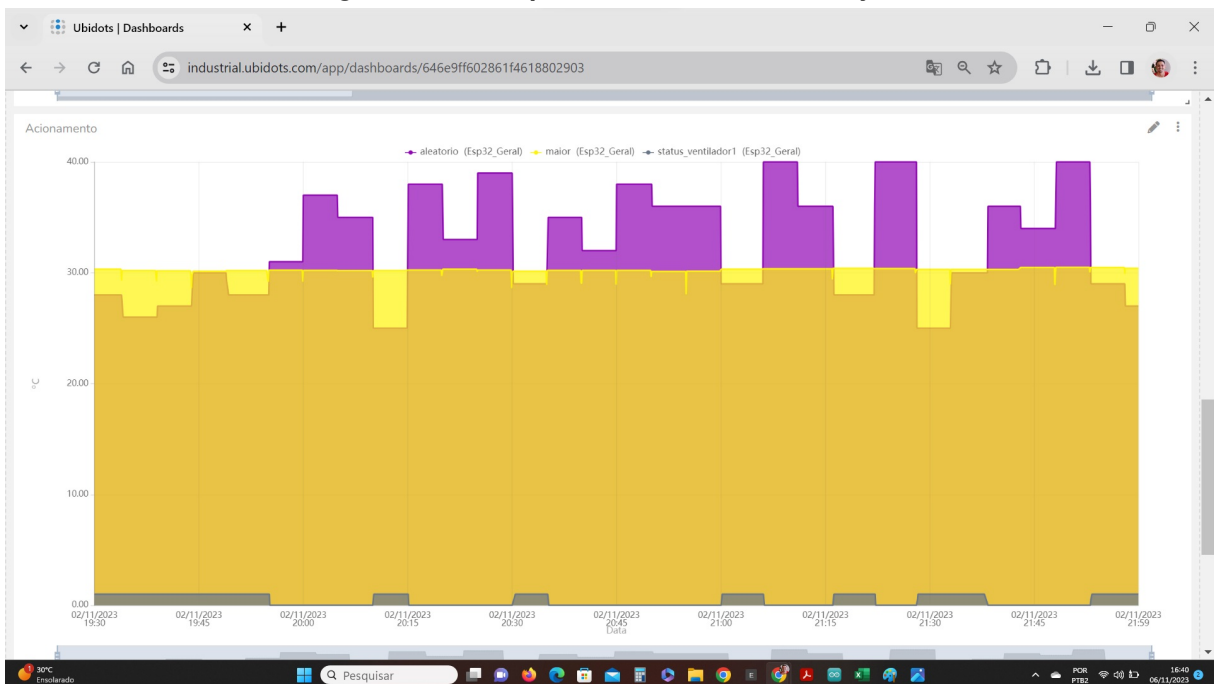


Fonte: Autoria Própria (2023).

As ilustrações acima exibem no *display* do dispositivo do ESP32 "Servidor" o resultado do cálculo das médias de temperatura e umidade, dos sensores 1, 2 e 4, mas desta vez nota-se que o valor do RSSI é maior, desconsiderando o sinal, isto acontece pois os dispositivos estão distantes um do outro, mas mesmo assim os dispositivos continuam interagindo entre si e enviando as informações com qualidade.

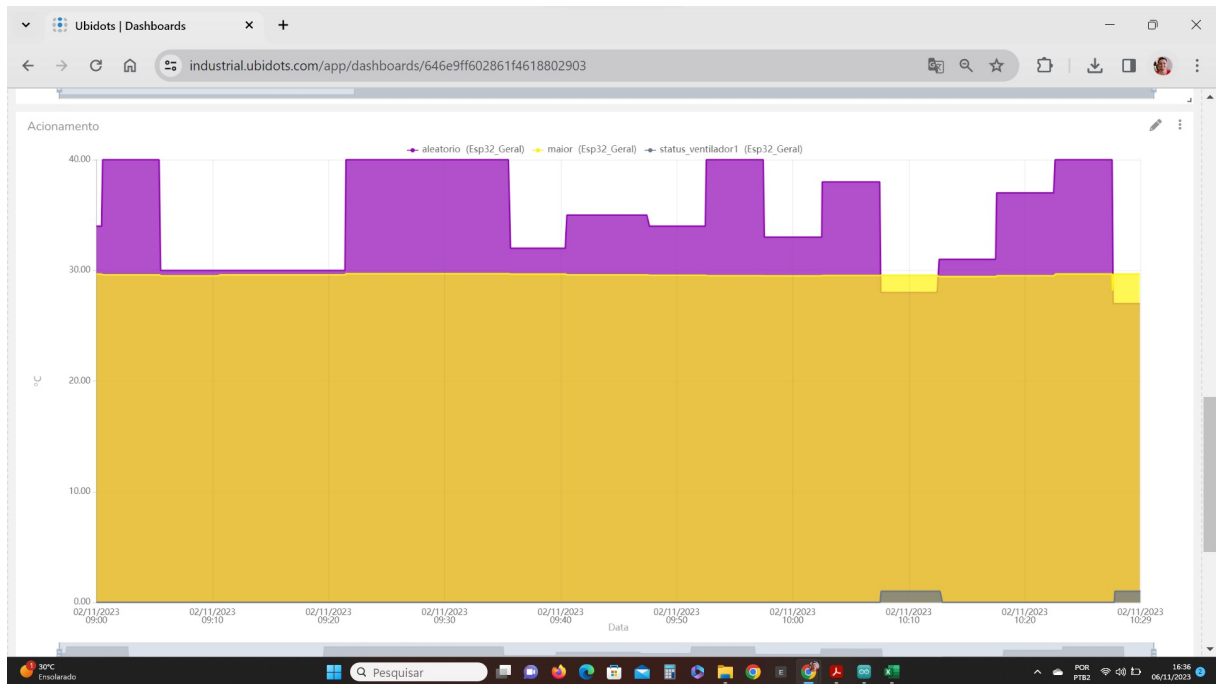
É possível extrair do Ubidots, alguns gráficos que demonstram o funcionamento do projeto, conforme ilustrado na Figura 30.

Figura 30 – Exemplo Funcionamento do Projeto



Fonte: Autoria Própria (2023).

Figura 31 – Exemplo Funcionamento do Projeto

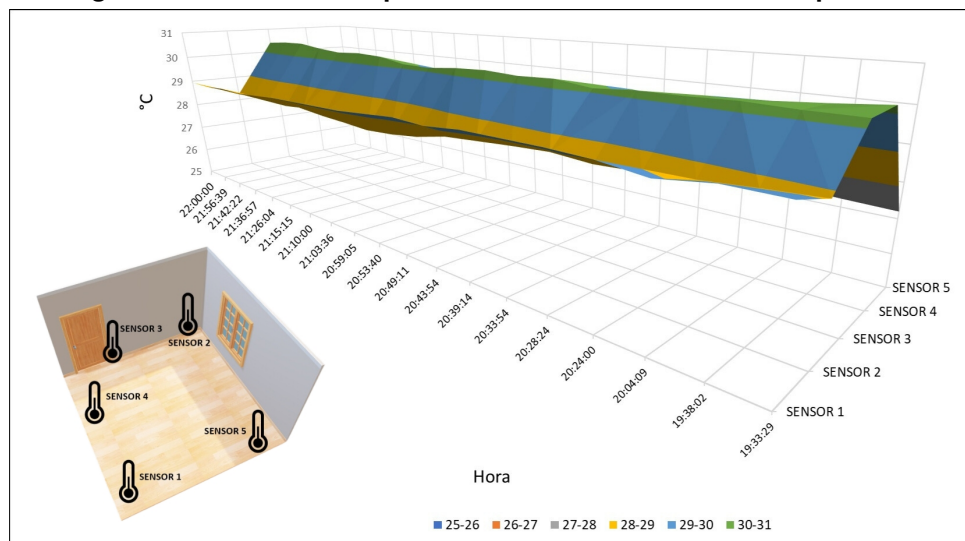


Fonte: Autoria Própria (2023).

Na Figura 30 e Figura 31, é ilustrado em amarelo o maior valor das médias das temperaturas, em roxo é ilustrado o valor da temperatura "ambiente" e em cinza é possível acompanhar o acionamento do ventilador. Nos momentos em que a área em amarelo sobrepõe a área na cor roxa por completo, o ventilador é acionado, caso contrario, é desligado. O intervalo do tempo (eixo x) é diferente para as duas Figuras, sendo que na Figura 30 o eixo x inicia as 19:30 Hrs e finaliza as 21:59 Hrs, enquanto que na Figura 31 o eixo x inicia as 09:00 Hrs e finaliza as 10:29 Hrs..

Com os dados obtidos através do broker da distribuição dos sensores no comodo da residência, conforme a Figura 8, é possível construir um gráfico, demonstrando a distribuição da temperatura no ambiente. Essa informação auxilia na tomada de decisão de qual o melhor lugar para instalação do ventilador, ou até mesmo de outro equipamento que seja responsável pela climatização do ambiente. Na Figura 32 é ilustrado essa simulação.

Figura 32 – Gráfico de Superfície Utilizando Leituras de Temperatura.



Fonte: Autoria Própria (2023).

Conclui-se que a melhor opção para a instalação do ventilador no ambiente ilustrado acima é perto da janela, desta forma, tem-se uma corrente de ar frio contra uma corrente de ar quente.

5 CONCLUSÃO

A partir das pesquisas iniciais e após a conclusão do projeto, é possível reafirmar que existem várias abordagens eficientes e viáveis no contexto de comunicação sem fio.

Neste projeto, pode-se afirmar que os resultados esperados foram alcançados de acordo com as ilustrações do funcionamento do projeto, onde tem-se um monitoramento de um ambiente, no que tange temperatura e umidade por uma rede de sensores sem fio, que se comunica, com outro dispositivo, que também está nesse ambiente, no caso, o ESP32 "Local", que por sua vez, é responsável pelo tratamento das informações e acionamento do ventilador.

A interação entre os sensores, que capturam a leitura da temperatura e umidade e o ESP32 "Local" é implementada através de tecnologias sem fio, como ESP-NOW e LoRa.

Através destas comunicações foi possível capturar as leituras de Temperatura e Umidade, envia-las de um microprocessador para o outro e por último, disponibilizar os resultados na nuvem, para que seja possível visualizar os resultados.

Os protocolos de comunicação utilizados neste projeto demonstraram que são eficientes, no que tange, velocidade em enviar os dados, precisão na entrega da informação e baixo consumo de energia.

Atualmente, a automatização, seja em qualquer ambiente, tem se tornado cada vez mais "indispensável", pois a necessidade de ser detentor de informações, seja quais forem, tem se tornado inevitável. Informações que muitas das vezes podem auxiliar nas tomadas de decisão.

Diante disso, a procura por recursos que entregam bons resultados se torna mais frequente, desta forma, pode-se aplicar esse projeto em diversos tipos de automações, como por exemplo, na área industrial, onde muitas vezes a saída para o aumento da produtividade é automatizar o processo.

Como sugestão para trabalhos futuros, pode-se inserir neste projeto o processo de acionamento de um umidificador, e obter a eficiência energética do circuito, através da medição da corrente e tensão das cargas que serão acionadas.

Desafios também surgiram no momento da alimentação dos sensores e microprocessadores, pois como o intuito é que o projeto fosse todo automatizado, se torna inviável o uso de cabamentos para realizar a alimentação, desta forma, foi utilizado baterias, entretanto, ainda se torna necessário recarregar as baterias entre períodos não muito alongados.

REFERÊNCIAS

- ALLIANCE, L. Technical Marketing Workgroup 1.0, **LoRaWAN™ - What is it?:** A technical overview of lora and lorawan™. 2017. Disponível em: <https://lora-alliance.org/resource-hub/what-lorawantm>. Acesso em: 15 set. 2019.
- AVELAR, J. N. Monitoramento inteligente de filas: Uma abordagem com interceptação de pacotes de rede wifi. **Nome da Revista ou Conferência**, Maio 2023. Artigo submetido ou publicado em uma revista ou conferência.
- AVELAR, J. N. **Tecnologias Emergentes de Conectividade na IoT:** Uma abordagem utilizando interceptacao de pacotes de rede wifi. 2023. Disponível em: https://tiberiusferreira.github.io/blog/posts/machine_learning_rust_simd_i/. Acesso em: 05 dez. 2023.
- BARATELLA, L. A.; GUILHERME SOBRINHO, D. Tecnologias wireless para automação industrial: Wireless_hart, bluetooth, wisa, wi-fi, zigbee e sp-100. **Instituto Nacional de Telecomunicações Inatel**, 2012. Disponível em: https://biblioteca.inatel.br/cict/acervo%20publico/sumarios/Artigos%20de%20TCC/TCC_Pos%20Gradua%C3%A7%C3%A3o/SAISEE%20-%20Engenharia%20de%20Sistemas%20%20Eletr%C3%B4nicos,%20Automa%C3%A7%C3%A3o%20e%20Controle%20Industrial/2012/TCC_Tecnologias%20Wireless%20para%20Automa%C3%A7%C3%A3o%20Industrial.pdf. Acesso em: 10 jan. 2020.
- CIA, A. e. Technical Marketing Workgroup 1.0, **ESP-NOW: Comunicação direta entre módulos ESP8266 e ESP32**. 2022. Disponível em: <https://www.arduinoecia.com.br/esp-now-comunicacao-direta-entre-modulos-esp8266-e-esp32/#comments>. Acesso em: 4 mai. 2022.
- EGLI, P. R. **LPWAN LOW POWER WIDE AREA NETWORK:** Overview of emerging technologies for low power wide area networks in internet of things and m2m scenarios lpwanlow power wide area network. 2015. Disponível em: <https://fr.slideshare.net/PeterREgli/lpwan>. Acesso em: 15 set. 2019.
- EVANS, D. A internet das coisas: Como a próxima evolução da internet está mudando tudo. **Cisco IBSG**, 2011. © 2011 Cisco e/ou suas afiliadas. Todos os direitos reservados.
- FERREIRA, R. M. A. F. H. C. **Campus Inteligentes: Conceitos, aplicações, tecnologias e desafios**. 2018. 19 p. Relatórios Técnicos do DIA - Pós Graduação em Informática (Grupo de Pesquisa e Inovação em Ciberdemocracia) — Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2018. Disponível em: https://www.researchgate.net/publication/322992882_Campus_Inteligentes_Conceitos_aplicacoes_tecnologias_e_desafios. Acesso em: 6 set. 2019.
- GARCIA, P.; KLEINSCHMIDT, J. Tecnologias emergentes de conectividade na iot: Estudo de redes lpwan. In: **XXXV SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS**. [s.n.], 2017. p. 5. Disponível em: <https://ieeexplore.ieee.org/document/7555573>. Acesso em: 6 set. 2019.
- GIFFINGER, R. *et al.* **Smart cities - Ranking of European medium-sized cities**. Viena: [s.n.], 2007. - p. Disponível em: http://www.smartcity-ranking.eu/download/city_ranking_final.pdf. Acesso em: 6 set. 2019.

- GILMAN, E. *et al.* Internet of things for smart spaces: A university campus case study. **Sensors**, v. 20, n. 13, 2020. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/20/13/3716>.
- HAYKIN, M. M. S. **Sistemas Modernos de Comunicações Wireless**. 1ª. ed. Porto Alegre: Bookman, 2008. 580 p. Acesso em: 05 jan. 2020.
- HIRSCH, J. W. B. Education beyond the cloud: Anytime-anywhere learning in a smart campus environment. In: **2011 International Conference for Internet Technology and Secured Transactions, ICITST 2011**. [s.n.], 2011. p. 718–723. ISBN 978-1-4577-0884-8. Disponível em: https://www.researchgate.net/publication/261195696_Education_beyond_the_cloud_Anytime-anywhere_learning_in_a_smart_campus_environment/link/57f5ebb308ae91deaa5c8ccb/download. Acesso em: 15 set. 2019.
- KANIKA, S.; NEHA, D. Um estudo de redes sem fio: Wlans, wpans, wmans e wwans com comparação. **Jornal Internacional de Ciência da Computação e Tecnologias da Informação**, v. 5, p. 7810–7813, 2014. Disponível em: http://www.academia.edu/download/45426150/Comparison_design.pdf. Acesso em: 30 Ago. 2020.
- KOYANAGI, F. **Introdução ao protocolo MQTT com Ubidots - ESP32**. 2019. Disponível em: <https://www.instructables.com/Introduction-to-the-MQTT-Protocol-With-Ubidots-ESP/>. Acesso em: 25 out. 2023.
- KWOK, L. A vision for the development of i-campus. **Smart Learning Environments**, v. 2, p. 1–12, 2015. Disponível em: <https://repositorio.ufsc.br/xmlui/handle/123456789/83015>. Acesso em: 6 set. 2019.
- LIU, X. A study on smart campus model in the era of big data. In: **2016 2nd International Conference on Economics, Management Engineering and Education Technology (ICEMEET 2016)**. Atlantis Press, 2017. ISBN 978-94-6252-288-6. ISSN 2352-5398. Disponível em: <https://doi.org/10.2991/icemeet-16.2017.191>. Acesso em: 6 set. 2019.
- LUDWIG, N. S. **Análise Ergonômica pelo método owas fazendo uso de sensores microcontrolados**. 2020. 84 p. Trabalho de Conclusão de Curso — Universidade de Caxias do Sul, Caxias do Sul - RS, 2020. Disponível em: <https://repositorio.ucs.br/xmlui/bitstream/handle/11338/9667/TCC%20Nicolas%20Savaris%20Ludwig.pdf?sequence=1&isAllowed=y>. Acesso em: 04 dez. 2020.
- MITRA. **New Engineering Education Transformation (NEET)**. Mai 2018. Executive Director NEET — Technische Universiteit Eindhoven, Países Baixos, Mai 2018.
- NEVES, A. R. de M. *et al.* Iniciativa smart campus: um estudo de caso em progresso na universidade federal do Pará. In: **Anais do I Workshop de Computação Urbana**. Porto Alegre, RS, Brasil: SBC, 2017. ISSN 2595-2706. Disponível em: <https://sol.sbc.org.br/index.php/courb/article/view/2576>. Acesso em: 15 set. 2019.
- NG, J. W. P. *et al.* The intelligent campus (icampus): End-to-end learning lifecycle of a knowledge ecosystem. In: **2010 Sixth International Conference on Intelligent Environments**. [s.n.], 2010. p. 332–337. Disponível em: http://naion.weebly.com/uploads/8/8/5/9/8859234/iseee_cps_nora_11-26pc.ppt. Acesso em: 16 set. 2019.
- O.MUNIZ., V. A. D. *et al.* Protocolos de comunicação para internet of things (iot). **Intercursos Revista Científica**, v. 17, 06 2019. Disponível em: <https://revista.uemg.br/index.php/intercursosrevistacientifica/article/view/3712>. Acesso em: 06 dez. 2023.

RAMOS, A. L. C.; SANTOS, J. E. L. dos. **Sistema integrado de automação residencial com comunicação sem fio**. 2015. 66 p. Trabalho de Conclusão de Curso do Curso de Engenharia de Controle e Automação — Universidade Tecnológica Federal do Paraná, Curitiba, 2015. Disponível em: <https://www.docsity.com/pt/sistema-integrado-de-automacao-residencial-com-comunicacao-sem-fio/5197484/>. Acesso em: 10 jan. 2020.

SANTANA, J. M. de S. **Redes LoRaWAN: Implantação e desenvolvimento de aplicações**. 2017. 67 p. Trabalho de Conclusão de Curso do Bacharelado em Engenharia de Telecomunicações — Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina, São José, 2017. Disponível em: https://wiki.sj.ifsc.edu.br/wiki/images/a/ac/TCC290_Jean_Michel_de_Souza_Sant%27Ana.pdf. Acesso em: 17 set. 2019.

SILVA, J. C. da. **Aplicação da tecnologia lora na automação de ambientes para controle de acesso**. 2017. 33 p. Projeto de pesquisa apresentado à disciplina de Trabalho de Conclusão de Curso — Universidade Tecnológica Federal do Paraná, Campo Mourão, 2017.

SILVA JUNIOR, V. Pereira da. **Lora em Ação - Praticando a teoria**. 2017. Engenheiro Eletrônico — ABINC - Associação Brasileira de Internet das Coisas, 2017. Disponível em: https://cec39f55-2b8d-4bcc-9312-30503c6eff53.filesusr.com/ugd/384e24_48c588b446474ce48e605aa1a50080ad.pdf. Acesso em: 20 ago. 2020.

TOBY, J. **MQTT and CoAP, IoT Protocols**. 2020. Disponível em: https://www.eclipse.org/community/eclipse_newsletter/2014/february/article2.php. Acesso em: 25 out. 2023.