



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Instituto de Ciência e Tecnologia
Câmpus de Sorocaba

GUSTAVO SANTOS DE SOUZA

INFRAESTRUTURA DE REDE ZIGBEE PARA O
MONITORAMENTO DE AMBIENTES NUM CAMPUS
INTELIGENTE

Sorocaba

2023

GUSTAVO SANTOS DE SOUZA

INFRAESTRUTURA DE REDE ZIGBEE PARA O
MONITORAMENTO DE AMBIENTES NUM CAMPUS
INTELIGENTE

Trabalho de Conclusão de Curso apresentado
ao Instituto de Ciência e Tecnologia de
Sorocaba, Universidade Estadual Paulista
(UNESP), como parte dos requisitos para
obtenção do grau de Bacharel em Engenharia
de Controle e Automação.

Orientador: Prof. Dr. Eduardo Paciência Godoy

Sorocaba

2023

S729i

Souza, Gustavo Santos de

Infraestrutura de rede zigbee para o monitoramento de ambientes em um campus inteligente / Gustavo Santos de Souza. -- Sorocaba, 2023

71 p. : tabs., fotos

Trabalho de conclusão de curso (Bacharelado – Engenharia de Controle e Automação) – Universidade Estadual Paulista (Unesp), Instituto de Ciência e Tecnologia, Sorocaba

Orientador: Eduardo Paciência Godoy

1. ZigBee. 2. Smart Campus. 3. Xbee. I. Título.

INFRAESTRUTURA DE REDE ZIGBEE PARA O MONITORAMENTO DE
AMBIENTES NUM CAMPUS INTELIGENTE

GUSTAVO SANTOS DE SOUZA

ESTE TRABALHO DE GRADUAÇÃO FOI JULGADO ADEQUADO
COMO PARTE DO REQUISITO PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Prof. Dr. Everson Martins
Coordenador

BANCA EXAMINADORA:

Prof. Dr. EDUARDO PACIENCIA GODOY
Orientador/UNESP-Campus de Sorocaba

PROF. DR. IVANDO SEVERINO DINIZ
UNESP-Campus de Sorocaba

Prof. Me. Vitor Mendes Caldana
IPSP – Campus de Sorocaba

2023

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a Deus por estar comigo nesse caminho longo e árduo, gostaria de agradecer aos meus pais, a senhora Sandra Cristina dos Santos minha mãe aquela que foi a pessoa responsável por me ajudar em toda essa trajetória e sempre me dar apoio nos momentos mais difíceis, ao meu pai senhor Marcelo de Souza por sempre ser um modelo de responsabilidade para minha pessoa, e gostaria de agradecer a minha avô materna senhora Deuzuíta um dos meus maiores exemplos de vida e parte do motivo na qual me fez entrar na faculdade, gostaria de agradecer ao Professor Doutor Eduardo Paciência Godoy por todo aprendizado, instrução, ajuda e companheirismo proporcionado ao longo de toda a criação desse projeto, e por fim gostaria de agradecer a toda instituição UNESP por estar fazendo parte desse capítulo da minha vida e por ter me dado a oportunidade de estar aprendendo valiosas informações que estarei levando pelo resto de toda a minha vida.

“Uma pessoa que nunca cometeu um erro,
nunca tentou nada novo.”

Albert Einstein

“O medo é o caminho para o lado
negro.”

Mestre Yoda

Souza, G. S. **Infraestrutura de rede zigBee para o monitoramento de ambientes em um campus inteligente**. 2023. Projeto Final de Curso (Bacharelado em Engenharia de Controle e Automação) – Instituto de Ciência e Tecnologia de Sorocaba, Universidade Estadual Paulista, Sorocaba, 2023.

RESUMO

No contexto acadêmico, a concepção de um campus inteligente, conhecido como “Smart Campus”, emerge como um princípio orientador nas universidades. No campus da UNESP Sorocaba esse conceito de Smart Campus tem sido desenvolvido em trabalhos que possibilitam o acesso às informações de interesse para uma gestão eficiente e sustentável dos recursos do campus. Diferentes aplicações estão disponíveis como a gestão do consumo de água, gestão de dados da estação meteorológica e a gestão do consumo de energia e da geração de energia solar do campus, sendo que todos os projetos utilizam conceitos educacionais que potencializam a experiência educacional dos estudantes através da integração de tecnologias interdisciplinares no processo de aprendizagem. A plataforma de Internet das Coisas (IoT) do campus inteligente suporta dois tipos de infraestrutura de comunicação sem fio para o desenvolvimento de aplicações, sendo o Wi-Fi com o protocolo MQTT e o LoRaWAN. Este projeto objetiva implementar uma nova infraestrutura de comunicação adicionando suporte para a tecnologia ZigBee, que representa um protocolo de comunicação sem fio em malha amplamente usado na automação predial e domótica. A infraestrutura ZigBee desenvolvida foi integrada à plataforma de IoT existente e contempla uma aplicação inicial de monitoramento de ambientes como salas de aula e laboratórios. Para essa aplicação, módulos Xbee integrados a sensores de movimento, luminosidade e temperatura realizam o monitoramento dessas variáveis e a comunicação via ZigBee com a plataforma de IoT do campus inteligente. Seguindo a padronização da plataforma IoT, a integração dos dados do ZigBee para o formato padrão do campus inteligente foi realizado através do Node-RED. O armazenamento desses dados estruturados foi realizado no banco de dados temporal InfluxDB. E a interação dos usuários com a aplicação foi disponibilizada através da criação de interfaces web ou dashboards na ferramenta Grafana. Resultados experimentais da aplicação de monitoramento de ambientes validaram o desenvolvimento da infraestrutura e sua compatibilidade com a plataforma de IoT do campus inteligente. Essa infraestrutura ZigBee é escalável para futuros trabalhos, tanto para o monitoramento de mais ambientes do campus quanto para o desenvolvimento de novas aplicações que utilizem esse tipo de tecnologia.

Palavras-chave: xbee; internet das coisas; smart campus; grafana.

Souza, G. S. ZigBee Network Infrastructure for Monitoring Environments on a Smart Campus, 2023. Graduation Work (Bachelor in Control and Automation Engineering) – Institute of Science and Technology of Sorocaba, São Paulo State University, Sorocaba, 2023.

ABSTRACT

In the academic context, the design of a Smart Campus emerges as a guiding principle in universities. In the campus of UNESP Sorocaba this concept of Smart Campus has been developed in projects that allow access to information of interest for an efficient and sustainable management of campus resources. Different applications are available such as water consumption management, weather station data management and the management of energy consumption and solar energy generation on campus, all of which use educational concepts that enhance the educational experience of students through the integration of interdisciplinary technologies in the learning process. The smart campus Internet of Things (IoT) platform supports two types of wireless communication infrastructure for application development, being Wi-Fi with the MQTT protocol and LoRaWAN. This project aims to implement a new communication infrastructure by adding support for ZigBee technology, which represents a mesh wireless communication protocol widely used in building automation and home automation. The developed ZigBee infrastructure was integrated into the existing IoT platform and includes an initial application for monitoring environments such as classrooms and laboratories. For this application, Xbee modules integrated with motion, luminosity and temperature sensors perform the monitoring of these variables and the communication via ZigBee with the IoT platform of the smart campus. Following the standardization of the IoT platform, the integration of ZigBee data into the standard smart campus format was accomplished through Node-RED. These structured data were stored in the InfluxDB temporal database. And the interaction of users with the application was made available through the creation of web interfaces or dashboards in the Grafana tool. Experimental results from the environment monitoring application validated the development of the infrastructure and its compatibility with the smart campus IoT platform. This ZigBee infrastructure is scalable for future work, both for monitoring more campus environments and for developing new applications that use this type of technology.

Keywords: xbee; internet of things; smart campus; grafana.

SUMÁRIO

1 INTRODUÇÃO	10
1.1 OBJETIVOS.....	12
2 REFERENCIAL TEÓRICO	13
2.1 SMART CAMPUS	13
2.2 REDE ZIGBEE: DEFINIÇÕES, CARACTERÍSTICAS E APLICAÇÕES	17
2.3 ARQUITETURA E TOPOLOGIA DE REDE	20
2.3.1 DISPOSITIVOS ZIGBEE	20
2.4 XBEE	23
2.4.1 ENDEREÇAMENTO	21
2.4.2 TRANSMISSÃO	21
2.4.3 FORMAÇÃO DE REDE E SEGURANÇA.....	22
2.4.4 ROTEAMENTO.....	22
3 MATERIAIS E MÉTODOS.....	24
3.1 PROPOSTA DE TRABALHO	24
3.1.1 NODE-RED	25
3.1.2 INFLUXDB.....	26
3.1.3 GRAFANA	26
3.2 MÓDULOS XBEE	27
3.2.1 COMUNICAÇÃO XBEE	28
3.3 SENSOR DE MOVIMENTAÇÃO	30
3.4 SENSOR DE TEMPERATURA.....	31
3.5 SENSOR DE LUMINOSIDADE.....	32
4. DESENVOLVIMENTO	34
4.1 DISPOSITIVOS ZIGBEE.....	34
4.1.1 CONFIGURAÇÃO DO FIRMWARE	34
4.1.2 PROJETO ELETRÔNICO	39
4.2 PROGRAMAÇÃO E INTEGRAÇÃO DE DADOS	41
4.2.1 NÓ XBEE RX	41
4.2.2 CÓDIGO DO SENSOR DE TEMPERATURA.....	42
4.2.3 CÓDIGO DO SENSOR DE LUMINOSIDADE	43
4.2.4 CÓDIGO DO SENSOR DE PRESENÇA	44
4.2.5 NÓ TO INFLUX.....	45
4.2.6 NÓ INFLUX TEST DB.....	46
4.3 ARMAZENAMENTO E VISUALIZAÇÃO DE DADOS	47
4.3.1 BANCO DE DADOS INFLUXDB	47
4.3.2 INTERFACE DE VISUALIZAÇÃO DO GRAFANA.....	51
5 RESULTADOS	60
6 CONCLUSÃO	68
REFERÊNCIAS.....	69

1 INTRODUÇÃO

A presença das aplicações de IoT (Internet das Coisas) tem se tornado cada vez mais comum no cotidiano, sendo utilizadas em smartphones, tablets e em diversos equipamentos eletrônicos conectados à internet que possam se comunicar entre si. Uma aplicação interessante da IoT é o Campus Inteligente, também conhecido como “Smart Campus”. Seu objetivo é criar uma universidade integrada e participativa, que busque soluções para problemas reais enfrentados no cotidiano universitário, visando a melhoria da qualidade de vida e sustentabilidade no campus.

Esse tipo de aplicação é uma realidade e vem sendo utilizada em algumas universidades, entre elas podemos citar a UNICAMP (UNICAMP, 2022) com o projeto de um dispositivo inteligente para fornecer a localização em tempo real de ônibus circulares (Figura 1) de circulação no campus.

Figura 1 – Projeto de Smart Campus da UNICAMP



Fonte: UNICAMP (2019)

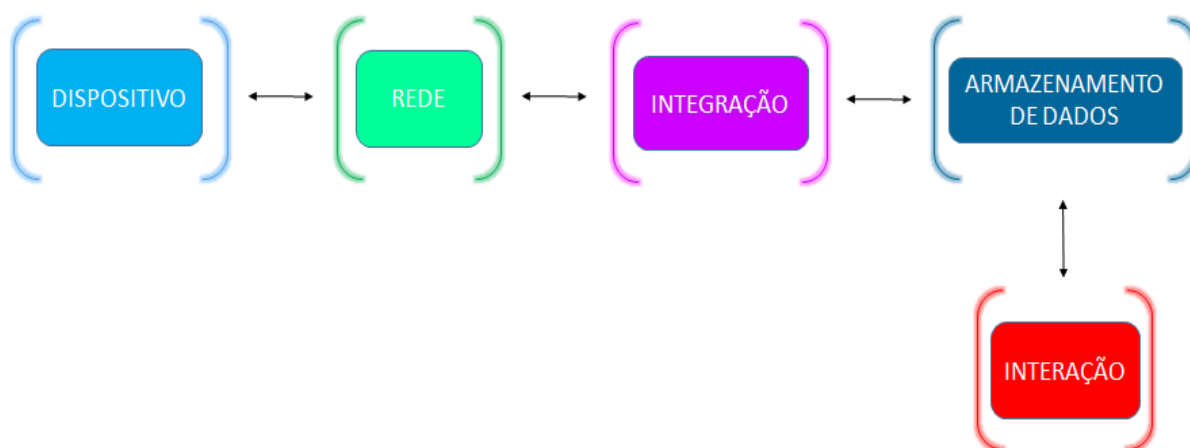
Uma arquitetura de IoT é o pilar de um campus inteligente. A comunicação utilizando soluções de IoT proporciona a aquisição de dados de diferentes objetos de interesse (sensores, ativos, pessoas, etc), o armazenamento dessas informações e disponibilização do acesso a elas de forma organizada e padronizada, além da sua análise e apresentação em painéis (dashboards) de dados obtidos em tempo real e históricos, fornecendo suporte às atividades técnicas e acadêmicas. Adicionalmente, essa infraestrutura inteligente pode fornecer dados que a serem usados para definir padrões de consumo energético e de hábitos dos usuários dos espaços, entre outras informações.

Na UNESP Sorocaba foi implementado o conceito de campus inteligente (SMART CAMPUS ICTS, 2023), na qual foi desenvolvido uma arquitetura IoT open source (código

aberto) para habilitar, padronizar e suportar o desenvolvimento de diferentes aplicações de interesse da comunidade do campus. Essa arquitetura IoT, mostrada na Figura 2, é composta resumidamente por cinco módulos principais (PROENÇA, 2023):

- 1) Dispositivo: componente gerador da informação, como um hardware, um computador, uma pessoa, um sensor ou atuador, entre outros;
- 2) Rede de comunicação: forma de transmissão da informação do seu local de obtenção para o seu painel de interação;
- 3) Integração de dados: formatação e padronização da informação recebida numa mesma estrutura definida para armazenamento;
- 4) Armazenamento de dados: sistema de armazenamento da informação em banco de dados temporal;
- 5) Painel de interação: ambiente web para desenvolvimento de telas de interação entre as aplicações e os usuários.

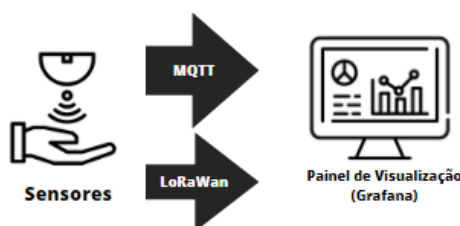
Figura 2 – Arquitetura IoT do Campus Inteligente da UNESP Sorocaba



Fonte: Proença (2022)

Até o momento, existem duas infraestruturas de comunicação sem fio para a transmissão de informação (Rede na Figura 2) na arquitetura do campus inteligente, sendo elas o padrão Wi-Fi usando MQTT e o padrão LoRaWAN conforme Figura 3. Essas duas infraestruturas foram implementadas através da criação de projetos anteriores no campus inteligente como a gestão do consumo de água, gestão de dados da estação meteorológica (UEMURA, 2023) e a gestão do consumo de energia e da geração de energia solar do campus (MORAES, 2023).

Figura 3 – Infraestruturas de Comunicação Existentes do Smart Campus



Fonte: Autoria Própria

O padrão ZigBee representa um protocolo de comunicação sem fio em malha amplamente usado na automação predial e domótica. No ZigBee, a comunicação sem fio é feita por vários nós ou dispositivos que são integrados a sensores, que coletam os dados de interesse e os enviam para um coordenador central. A partir do recebimento de dados deste coordenador de rede, as informações podem ser enviadas para tratamento, formatação e armazenamento. Esse formato tem suas vantagens, considerando que é possível ampliar a rede (com mais módulos) para obter um alcance maior, é possível instalar o módulo em praticamente qualquer lugar e dependendo do hardware do módulo é possível integrar uma grande quantidade de sensores para monitorar diversas variáveis.

Este trabalho concentra-se na implementação de uma nova infraestrutura que utiliza o padrão ZigBee no campus inteligente da Unesp Sorocaba.” A infraestrutura ZigBee desenvolvida foi integrada à plataforma de IoT existente e contempla uma aplicação inicial de monitoramento de ambientes como salas de aula e laboratórios que tem a princípio a pretensão de monitorar ambientes da universidade, por exemplo salas de aula ou laboratórios.

1.1 OBJETIVOS

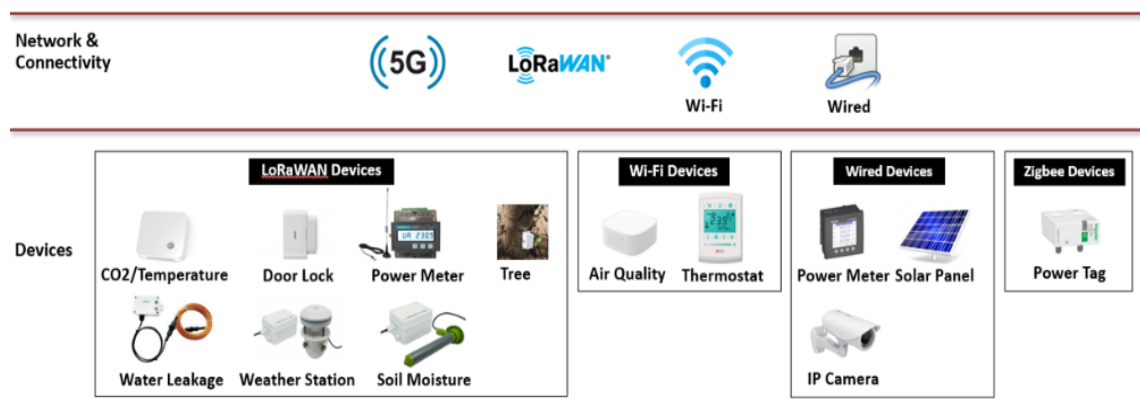
O objetivo deste trabalho é desenvolver uma infraestrutura de comunicação sem fio ZigBee integrada à plataforma de IoT do campus inteligente da UNESP Sorocaba. Essa infraestrutura deve permitir a adição de dispositivos compatíveis com o padrão de comunicação ZigBee para a criação de aplicações de interesse da comunidade, sendo inicialmente para o monitoramento de variáveis em ambientes do campus.

2 REFERENCIAL TEÓRICO

2.1 SMART CAMPUS

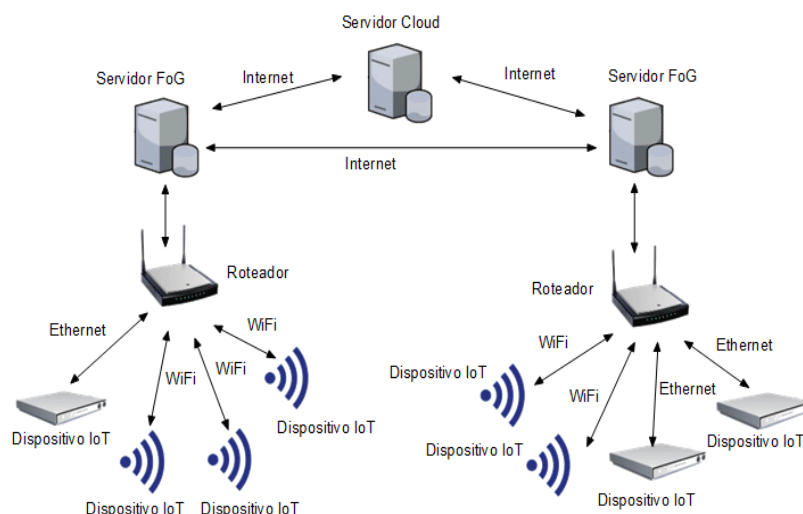
“O conceito de Tecnologia IoT é de que os objetos do nosso cotidiano podem ser embutidos com identificação, sensores, rede de comunicação e capacidade de processamento e deste modo, eles podem interagir e se conectar entre si, com outros dispositivos e serviços para alcançar um objetivo” (ATZORI *et al.*, 2010, p. 15), esse tipo de conectividade pode ser ilustrado como o exemplo da Figura 4, que mostra como seria uma rede estruturada com via comunicação Wi-fi. Instituições de ensino como o Instituto Nacional de Telecomunicações, Universidade de Campinas e Faculdade de Engenharia de Sorocaba (FACENS) têm se dedicado a enriquecer a infraestrutura de seu campus com sensores, a fim de coletar informações em tempo real e utilizá-las para melhorar a experiência dos estudantes e funcionários, aumentar a eficiência na prestação de serviços e promover o desenvolvimento de projetos da comunidade universitária. Tecnologias como 5G, LoRaWAN, WI-FI ou Zigbee, fazem parte do grupo de tecnologias sem fio que são amplamente utilizados em plataformas IoT, existem uma ampla gama de dispositivos, desde sensores simples até dispositivos mais complexos, como veículos autônomos e aparelhos domésticos inteligentes, como mostrado na Figura 5.

Figura 4 – Conectividades e Dispositivos



Fonte: ITSC HKUST (2021)

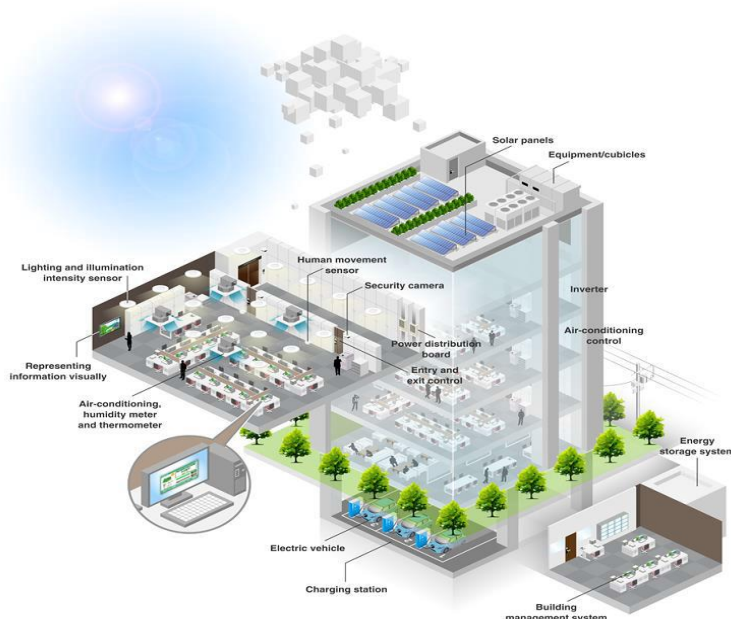
Figura 5 – Dispositivos Eletrônicos e objetos interconectados



Fonte: Tibco (2021).

Um Smart Campus (Figura 6) integra diversas tecnologias recentes, tais como armazenamento em nuvem, sistema de controle inteligente, big data, IoT e Mobile Internet, tendo como propósito a construção de um ambiente seguro, amigável, sustentável e que facilite tanto a gestão da universidade, quanto o desenvolvimento acadêmico (XIONG, 2016, p. 11).

Figura 6 – Exemplo Ilustrativo de um Smart Campus



Fonte: Introduction (2019)

São exemplos de aplicações práticas do Smart Campus:

- Salas inteligentes: nestas salas, há o controle da luz e ar condicionado de forma autônoma de acordo com a presença ou ausência de pessoas no interior da sala. É possível, ainda, verificar quais salas estão ocupadas e quais estão vazias.
- Estacionamentos inteligentes, em que é possível saber previamente quais vagas estão

o que a torna adequada para aplicações de IoT que operam com baterias de baixa potência.

- Baixa taxa de transferência de dados: a rede ZigBee não é projetada para transferir grandes volumes de dados, mas sim para fornecer comunicação confiável e de baixa latência para dispositivos IoT.
- Baixo custo: a tecnologia ZigBee é relativamente barata em comparação com outras tecnologias de rede sem fio, o que a torna uma opção atraente para a criação de redes de IoT de baixo custo.
- Segurança: a rede ZigBee utiliza criptografia de dados para garantir a segurança das informações transmitidas.
- Confiabilidade: a rede ZigBee usa uma topologia de rede em malha, o que significa que cada dispositivo pode se comunicar com outros dispositivos próximos. Isso torna a rede mais confiável e resistente a falhas.
- Flexibilidade: a rede ZigBee é flexível e pode ser configurada para diferentes aplicações, como automação residencial, cidades inteligentes, saúde digital, agricultura de precisão, entre outras.

De fato, a rede ZigBee pode ser amplamente utilizada em aplicações IoT, como automação residencial, cidades inteligentes, agricultura de precisão e muitas outras áreas, o que a torna uma das tecnologias mais promissoras para suportar a conectividade e a interoperabilidade de dispositivos inteligentes em diferentes contextos, alguns exemplos pode-se citar:

- Automação residencial e predial: a tecnologia ZigBee podendo ser usada em sistemas de iluminação, controle de temperatura, fechaduras eletrônicas, entre outras aplicações do campo da domótica.
- Monitoramento de saúde: dispositivos de monitoramento de saúde, como monitores de frequência cardíaca e glicose, podem usar a tecnologia ZigBee para enviar dados para um dispositivo central.
- Agricultura inteligente: a tecnologia ZigBee pode ser utilizada em sistemas de monitoramento e controle de irrigação, fertilização e outras condições agrícolas.
- Redes de sensores sem fio: a tecnologia ZigBee pode ser empregada em redes de sensores sem fio para coletar e transmitir dados em aplicações como monitoramento ambiental e detecção de incêndios.
- Controle de iluminação pública: a tecnologia ZigBee pode ser usada em sistemas de iluminação pública para permitir o controle remoto de lâmpadas individuais, bem como para coletar dados de uso e desempenho.

A figura 8 mostra alguns exemplos de aplicações Zigbee, como o uso em domótica, medidores de energia etc.

Figura 8 – Exemplos de Aplicações ZIGBEE

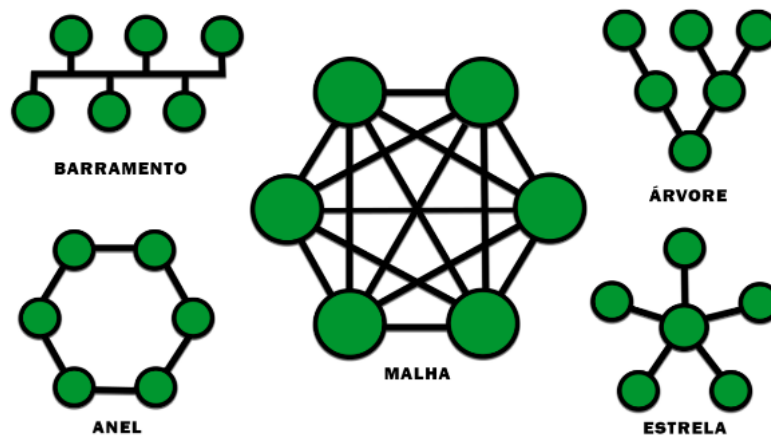


Fonte: CSA-IOT (2022)

2.3 ARQUITETURA E TOPOLOGIA DE REDE

“A rede ZigBee é baseada na norma IEEE 802.15.4, que define uma camada física e de controle de acesso ao meio (MAC) para redes sem fio de baixa taxa de transmissão e baixo consumo de energia” (AHMAD *et al.*, 2013, p. 2), essa norma é responsável pela definição dos pacotes de dados, frequências de operação, potência de transmissão etc. “O padrão IEEE 802.15.4 foi projetado para suportar várias topologias de rede, incluindo redes em malha e estrela, o que permite a construção de redes ZigBee escaláveis e flexíveis” (SHEHAB *et al.*, 2015, p. 3), a Figura 9 ilustra alguns exemplos de topologias de rede como Árvore, Malha, Estrela etc.

Figura 9 – Tipos de Topologias de Rede



Fonte: Wiki IFSC (2015)

No quesito topologia, a formação da rede é gerenciada pela camada de rede do ZigBee. A rede precisa estar em uma das duas topologias especificadas na IEEE 802.15.4: estrela ou peer-to-peer. Na topologia estrela, como pode ser visto na figura 10, cada dispositivo pode se comunicar apenas com o coordenador PAN.

Figura 10 – Topologia Peer To Peer



Fonte: Research Gate (2015)

Em uma topologia peer-to-peer cada dispositivo pode se comunicar diretamente com qualquer outro desde que estejam próximos o suficiente para estabelecer um link de comunicação. Uma topologia peer-to-peer pode assumir diferentes formatos bastando para isto definir restrições a respeito de quais dispositivos podem se comunicar. Se não houver nenhuma restrição, a rede peer-to-peer é conhecida com topologia mesh ou malha.

Em relação aos tipos de dispositivos que podem se conectar a uma rede ZigBee existem dois tipos principais de dispositivos ZigBee, de acordo com ZigBee Alliance (2021, p. 34):

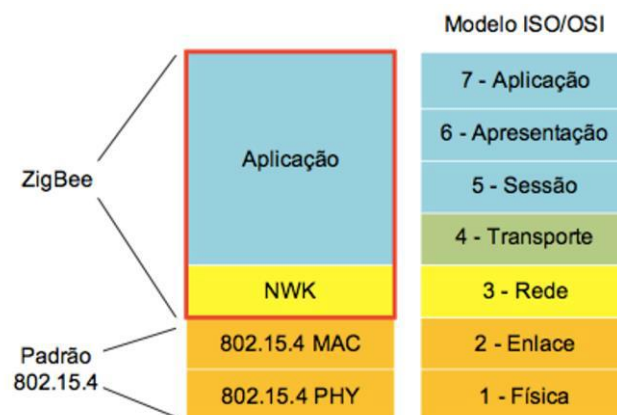
- O dispositivo FFD pode atuar como um coordenador ou um roteador dentro da rede ZigBee, possuindo características semelhantes. A diferença é que apenas um coordenador pode iniciar uma nova rede ou trocar dados entre diferentes redes.
- O dispositivo RFD é conhecido como um escravo, pois ele apenas se comunica com um coordenador ou roteador associado e não pode atuar como eles. Além disso, o dispositivo RFD é mais simples em sua construção em comparação com o FFD, sendo mais barato, uma vez que não precisa armazenar informações de roteamento, o que significa que ele necessita de menos memória. Além disso, esses dispositivos podem ser divididos em 3 diferentes classes como:
 - Coordenador (FFD), permite a formação de uma rede, atribuir endereços e guardar endereços de dispositivos vizinhos.
 - Roteador (FFD), efetua funções de controle e monitoramento e permite que mais

dispositivos se juntem à rede.

- End-device (FFD ou RFD), pode tanto atuar como controle e monitoramento do dispositivo que ele se encontra integrado.

Existem 4 tipos de camadas na pilha de protocolo ZigBee como mostra a Figura 11, sendo as mais importantes as camadas de rede/segurança e suporte e aplicação. A camada de suporte à aplicação (*Application Support Layer*) é a camada superior na arquitetura do protocolo ZigBee, e é responsável por fornecer suporte às aplicações que rodam na rede ZigBee. De acordo com a norma IEEE 802.15.4, a camada de suporte à aplicação é responsável por “fornece mecanismos para gerenciar os serviços de aplicação, e permitir que as aplicações se comuniquem com a camada de rede”, como evidenciado na Figura 11.

Figura 11 – Camada do Padrão ZigBee



Fonte: IPV6 (2023)

Algumas das principais funções da camada MAC do ZigBee incluem o controle de acesso ao canal (gerenciando os momentos em que os dispositivos podem transmitir e receber dados), o controle de fluxo (gerenciando a quantidade de dados que podem ser transmitidos de uma vez), o endereçamento de dispositivos (atribuindo endereços exclusivos a cada dispositivo da rede) e a segurança (fornecendo recursos de criptografia e autenticação para proteger a rede contra ataques).

Existe a camada intermediária PHY sendo a camada mais baixa na arquitetura do protocolo ZigBee e é responsável por transmitir e receber os sinais de rádio entre os dispositivos ZigBee. A camada PHY do ZigBee é baseada na tecnologia IEEE 802.15.4, que é uma especificação padrão para redes sem fio de baixa taxa de transmissão e baixo consumo de energia. A camada PHY do ZigBee define várias características importantes, incluindo a faixa de frequência utilizada (2,4 GHz, 868 MHz ou 915 MHz), a modulação do sinal (BPSK, QPSK ou

O-QPSK), a taxa de dados (250 kbps ou 20 kbps), o alcance da rede (até 100 metros ao ar livre) e a sensibilidade do receptor (até -100 dBm).

2.3.1 DISPOSITIVOS ZIGBEE

“A rede ZigBee se enquadra em uma classe de redes sem fio denominada de WPAN (Wireless Personal Area Network), ou redes sem fio de área pessoal, sendo um padrão caracterizado pelos baixos custos, consumo de energia e complexidade, mas também baixa transmissão de dados” (KUMAR, 2017, p.3), ainda, apresenta alcance longo, podendo chegar a 100 metros.

No entanto, hoje já se encontram módulos que podem alcançar mais de 1 km, dependendo do modelo do rádio e da antena. Dentre suas principais características, a rede ZigBee é capaz de suportar muitos dispositivos conectados simultaneamente, oferecendo assim uma maior flexibilidade em termos de configuração e gerenciamento da rede. Além disso, ela é uma tecnologia de baixo consumo de energia, o que a torna adequada para dispositivos alimentados por bateria, como sensores e dispositivos móveis.

2.4 XBEE

“O Xbee é um dispositivo desenvolvido pela empresa Digi International, reconhecida na área de dispositivos de comunicação sem fio” (ESCHNER, 2011, p.18). Trata-se de um módulo que possui o protocolo ZigBee embarcado no próprio microcontrolador, sendo disponibilizado em dois modelos, Xbee e Xbee-PRO, com as mesmas funcionalidades, diferindo-se apenas em alcance e potência de transmissão. Há diversos tipos de módulos de rádios Xbee que utilizam diferentes protocolos de comunicação, como Bluetooth, WiFi, ZigBee entre outros.

De acordo com Eschner (2011), uma das vantagens do módulo Xbee é a sua capacidade de operar em diferentes topologias, além da implementação da tecnologia de sleep mode, que proporciona maior autonomia às baterias ou pilhas utilizadas no dispositivo, podendo chegar a alguns anos em alguns casos sem necessidade de substituição. Os módulos Xbee também podem ser configurados para atuar como roteadores, formando uma topologia de rede mesh ou malha, possibilitando a expansão da rede conforme a necessidade. O firmware do Xbee é responsável pelos tratamentos e controles para garantir a transmissão bem-sucedida das mensagens entre os módulos, seguindo as regras e normas especificadas pelo padrão ZigBee. “O módulo Xbee, que já possui o protocolo ZigBee embarcado em seu microcontrolador, é um dos principais componentes utilizados no projeto, sendo responsável pela comunicação entre cada nó sensor e a central” (ESCHNER, 2011, p. 18).

2.4.1 ENDEREÇAMENTO

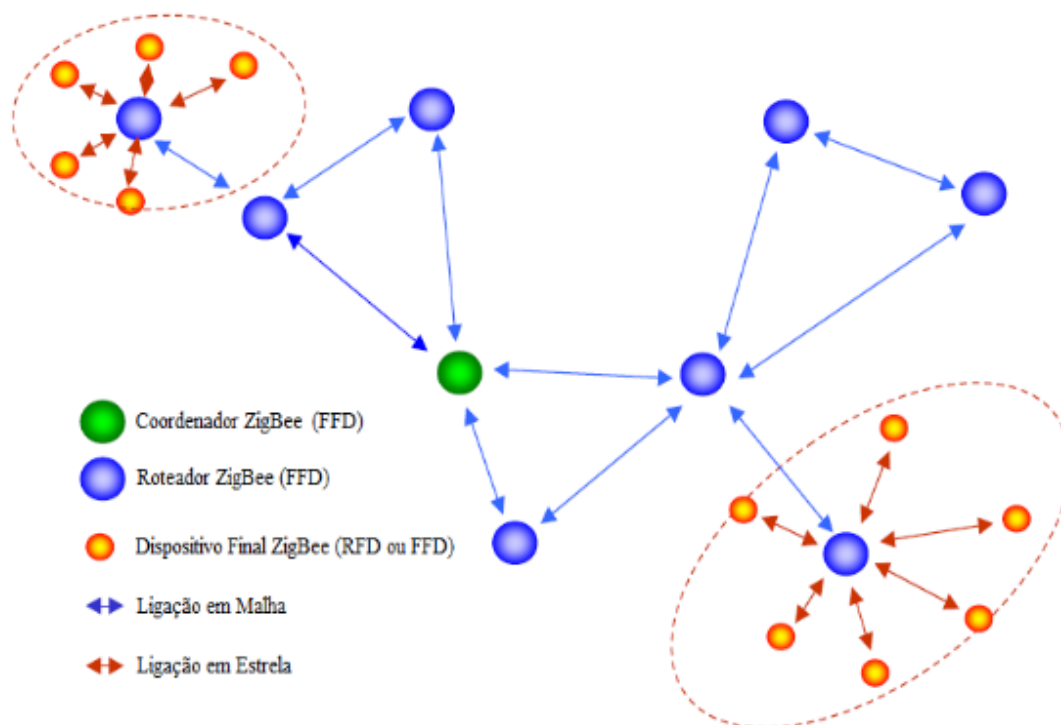
O esquema de endereçamento na rede ZigBee é crucial para garantir a eficiente comunicação entre dispositivos em uma rede de baixa potência e baixa taxa de transmissão de dados. Conforme descrito por Zigbee Alliance (2023, p. 15) e Li *et al.* (2011, p. 5), cada dispositivo ZigBee possui um endereço de 64 bits exclusivo, composto por um identificador de 16 bits definido pela ZigBee Alliance e um identificador de 48 bits exclusivo para cada dispositivo. O identificador de 16 bits é utilizado para identificar o dispositivo em sua rede local, enquanto o identificador de 48 bits é usado para identificá-lo de forma única em toda a rede ZigBee.

Além disso, a rede ZigBee também suporta endereços de grupo, que são usados para endereçar um conjunto específico de dispositivos que precisam receber a mesma mensagem. Os endereços de grupo são atribuídos dinamicamente pela rede e permitem que um dispositivo envie uma mensagem para um grupo específico de dispositivos. Outros aspectos relevantes do esquema de endereçamento na rede ZigBee incluem a interoperabilidade entre diferentes dispositivos ZigBee, garantida pela estrutura definida pela ZigBee Alliance para os endereços de 64 bits, e a possibilidade de encaminhamento de mensagens entre dispositivos, permitindo maior confiabilidade e resiliência da rede.

2.4.2 TRANSMISSÃO

Pacotes de dados ZigBee podem ser enviados em transmissões tanto unicast como broadcast. De acordo com Andrade (2012, p. 55), a transmissão unicast é usada em situações em que é necessário enviar informações específicas para um único destinatário. Por exemplo, ao enviar um e-mail para um único destinatário ou ao acessar um site específico na internet, o envio e recebimento das informações ocorrem por meio da transmissão unicast, já a transmissão Broadcast segundo Tanenbaum & Wetherall (2011, p. 15) é utilizada em situações em que é necessário enviar informações para um grande número de destinatários simultaneamente. Por exemplo, ao transmitir um programa de TV ou rádio pela internet, as informações são enviadas por meio da transmissão broadcast. A figura 12 nos mostra um exemplo de esquema de Transmissão Zigbee em Broadcast.

Figura 12 – Esquema de Transmissão



Fonte: ELETRONET (2013)

2.4.3 FORMAÇÃO DE REDE E SEGURANÇA

De acordo com Bezerra (2009, p. 33) cada rede formada pelos dispositivos possui um endereço específico chamado PAN ID que é o meio para diferenciar as redes por diversos dispositivos coordenadores. Esse endereço deve ser armazenado em cada dispositivo ZigBee ao se juntar a rede especificada. Somente um coordenador pode iniciar uma PAN, os roteadores e dispositivos finais (END-DEVICE) devem se juntar a PAN criada mais próxima para formar uma rede de trabalho.

Os roteadores fazem o papel de “agente” que permite o acesso dos dispositivos a rede, o controle de acesso depende basicamente da quantidade de dispositivos filhos que está integrada à rede. “A segurança é feita após o coordenador iniciar a rede ZigBee, é usado uma chave de criptografia de 128 bits e somente dispositivos que possuem a mesma chave podem ter acesso a rede formada, sendo que essa chave pode ser adquirida por pré-configuração ou na aquisição da conexão da rede existente” (ZIGBEE ALLIANCE, 2009, p. 55).

2.4.4 ROTEAMENTO

O protocolo ZigBee inclui vários conceitos diferentes para roteamento de pacotes, conforme descrito no quadro 1 abaixo.

Quadro 1 – Tipos de Roteamentos

Roteamento	Descrição	Quando Usar
Ad hoc On-demand Distance Vector (AODV) Mesh Routing	Caminhos de roteamento são criados entre origem e destino, possivelmente atravessando vários nós (“saltos”). Cada dispositivo sabe para onde enviar os dados a seguir para finalmente alcançar o destino.	Usar em redes que não terão mais que 40 dispositivos de destino
Many-to-One Routing	Uma única transmissão de broadcast configurar rotas reversas em todos os dispositivos, para o dispositivo que envia a transmissão.	Útil quando muitos dispositivos remotos devem enviar dados para um único gateway ou dispositivo coletor.
Source Routing	Pacote de dados incluem a rota inteira que o pacote deve percorrer para ir da origem ao destino.	Melhora a qualidade do roteamento em redes grandes (com mais de 40 dispositivos)

Fonte: Autoria Própria (2023)

3 MATERIAIS E MÉTODOS

3.1 PROPOSTA DE TRABALHO

O presente trabalho apresenta o desenvolvimento de uma Infraestrutura ZigBee complementar para a arquitetura de IoT do Smart Campus da Unesp Sorocaba. Essa infraestrutura será inicialmente aplicada para o monitoramento de ambientes, sendo 3 salas selecionadas no GASI do campus da UNESP Sorocaba. Os dispositivos ZigBee utilizados para o projeto são os módulos Xbee e as variáveis a serem monitoradas de cada ambiente são a temperatura, luminosidade e detecção de movimento de pessoas. A Figura 13 representa o esquemático para o desenvolvimento da infraestrutura de rede ZigBee integrada ao campus inteligente.

Figura 13 – Topologia da Infraestrutura de rede ZigBee



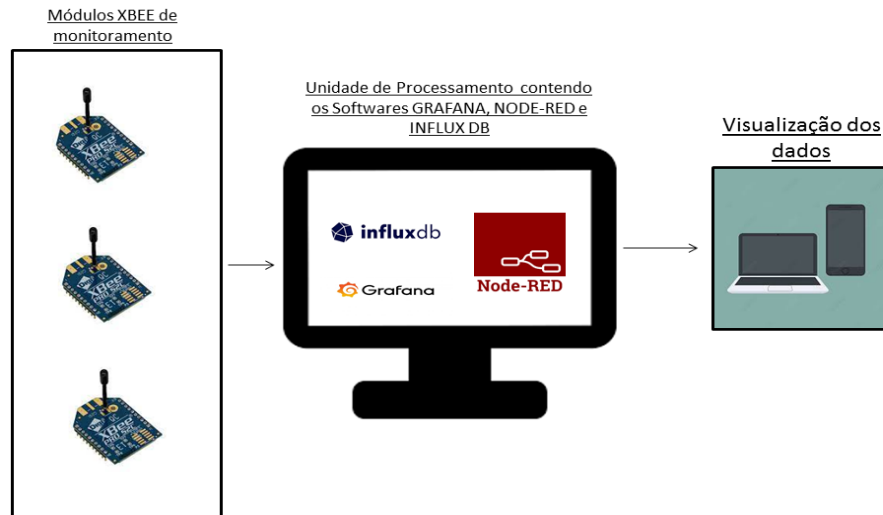
Fonte – Autoria Própria

Para desenvolvimento do projeto é aproveitado uma parte da infraestrutura existente do campus inteligente, sendo o servidor e painei da Figura 13. Nesse servidor está conectado o USB-ZigBee (Xstick), através de uma porta COM serial, que será o coordenador da rede ZigBee na qual os módulos de monitoramento Xbee (Sistema Embarcado) estão conectados. O monitoramento de ambiente é feito através dos módulos alocadas nos locais de interesse, os quais fazem a coleta dos dados do ambiente. Vale lembrar que esses módulos funcionam na rede ZigBee como nós terminais (*End Devices*). O sensoramento do ambiente é feito pelos módulos Xbee S2 (*End Device – ZigBee*). Em todos os módulos, existe um sensor de presença para indicar o movimento de pessoas nos ambientes, também foi acoplado um LDR para medir a luminosidade do local e foi inserido também a presença de um sensor de temperatura em um dos módulos para monitoramento da temperatura local.

A Figura 14 resume o esquemático de desenvolvimento. As informações de monitoramento são enviadas pelos dispositivos Xbee são recebidas e tratadas pelo Node-RED. Essas informações são formatadas e estruturadas no formato padrão do campus inteligente. Após esse tratamento, as informações são armazenadas no servidor local num banco de dados

InfluxDB. A visualização dos dados será realizada através de acesso Web numa interface criada usando a ferramenta Grafana.

Figura 14 – Unidade Central de Processamento

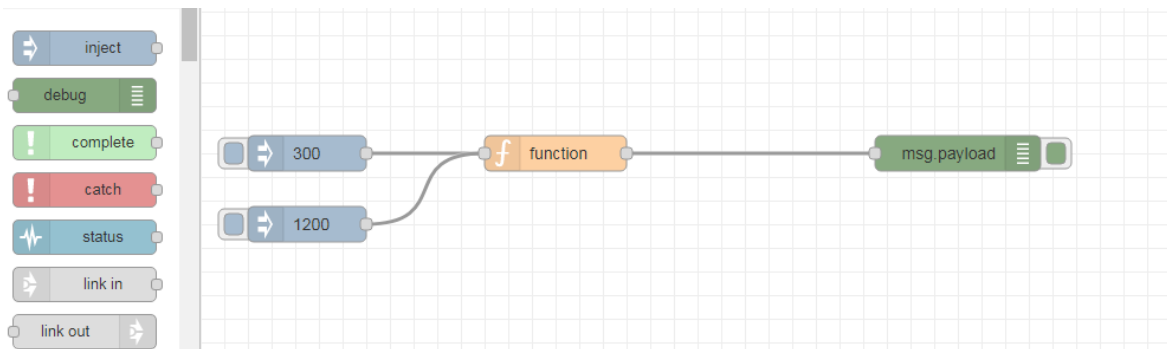


Fonte: Autoria Própria

3.1.1 NODE-RED

Node-RED é uma plataforma de programação visual que permite a criação de aplicações de Internet das Coisas (IoT) e automação de fluxo de trabalho. Ele fornece um ambiente de desenvolvimento baseado em browser que utiliza uma abordagem de arrastar e soltar para a criação de fluxos de trabalho, onde os usuários podem conectar diferentes nós para construir lógica de programação (NODERED,2022). A figura 15 nos mostra um exemplo de aplicação de Node que utiliza uma função para criar alguma expressão matemática que é mostrado no Payload.

Figura 15 – Exemplo de Node



Fonte: Autoria Própria

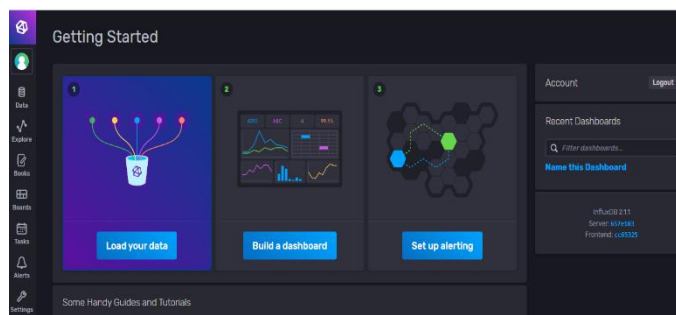
O Node-RED é baseado no mecanismo Node.js e utiliza a linguagem JavaScript. Ele fornece uma ampla gama de nós pré-construídos, permitindo que os usuários integrem facilmente diferentes serviços e dispositivos em seus fluxos de trabalho. Além disso, é possível criar nós personalizados para estender a funcionalidade do Node-RED.

3.1.2 InfluxDB

O software InfluxDB é desenvolvido pela empresa InfluxData, sendo um banco de dados de séries temporais de código aberto, otimizado para armazenar e consultar grandes volumes de dados baseados em tempo real. Esse software foi projetado para lidar com dados de séries temporais, como dados de sensores, logs de eventos, métricas de monitoramento e dados de IoT.

Ele fornece uma linguagem de consulta poderosa (InfluxQL ou Flux) e APIs para coleta, armazenamento e consulta eficientes desses dados (INFLUXDATA, 2021), além de possuir integrações com outros softwares como Node-RED e o Grafana. A Figura 16 nos revela como é a interface de Usuário do InfluxDB.

Figura 16 – Interface de Usuário do InfluxDB

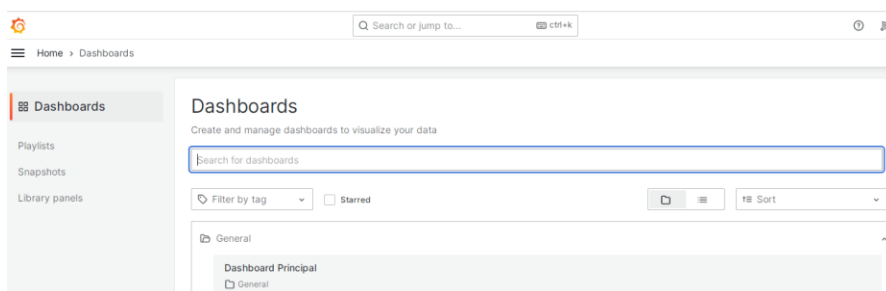


Fonte: Autoria Própria

3.1.3 GRAFANA

O Grafana é uma plataforma de análise e visualização de dados de código aberto. Ele permite a criação de painéis interativos e personalizados, oferecendo uma variedade de opções de visualização, gráficos e recursos de monitoramento (GRAFANA, 2021). A Figura 17 nos mostra como é o portal inicial do Grafana.

Figura 17 – Portal do Grafana



Fonte: Autoria Própria

É um recurso muito utilizado no conceito de IoT (Internet das coisas), e é compatível com muitos sistemas operacionais e outros programas como o Node-RED e o InfluxDB que são utilizados na proposta do projeto.

3.2 MÓDULOS XBEE

Os módulos de comunicação sem fio Xbee PRO S2 da Digi International são dispositivos de baixo consumo de corrente e de baixo custo que utilizam o protocolo ZigBee. Esses módulos incorporam diversos I/Os e oferecem flexibilidade na configuração dos sinais de cada pino, seja como entrada analógica ou entrada/saída digital.

Figura 18 – Módulo XBEE Pro S2



Fonte: DIGI (2013)

Existe também outro tipo de módulo que foi implementado na arquitetura, que no caso seria o USB-ZIGBEE XBEE, esse módulo faz o papel de coordenador da rede, ou seja, é o responsável pela criação, configuração e monitoramento da rede. A *Figura 19* nos mostra um modulo adaptador Xstick.

Figura 19 – Adaptador USB-ZigBee Xstick



Fonte: DIGI (2013)

Existe também uma tabulação com alguns dos principais parâmetros de um módulo XBEE Pro S2, como mostrado na *Tabela 1*:

Tabela 1 – Especificações do Módulo XBEE

Especificação Módulo XBEE Pro S2	
Alcance	de 60 até 1200 m
Potência de Transmissão	6.3 Mw / 3.1 Mw
Taxa de dados RF	250 Kbps
Tensão de Operação (placa)	2.1 – 3.3 V
Tensão de Operação (Pinos)	1.2 V
Corrente de transmissão (Vcc = 3.3)	45Ma / 33 Ma
Canais	11 ao 26
Banda de Frequência	ISM 2.4 – 2.5 G Hz

Fonte: Autoria Própria

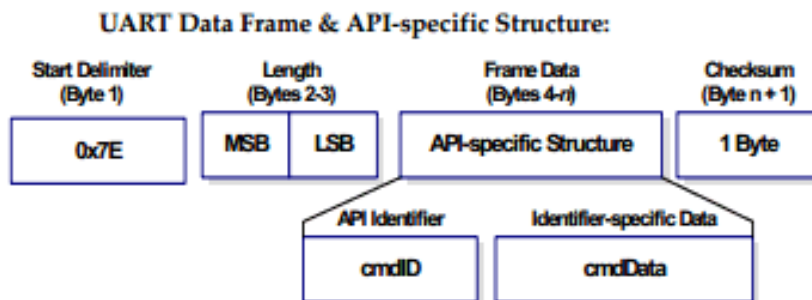
3.2.1 COMUNICAÇÃO XBEE

A comunicação entre os módulos XBEE é feita por 2 modos: o modo de operação AT e o API. Esse modo de operação pode ser configurado alterando o firmware do dispositivo. O modo AT ou modo de comunicação transparente, a comunicação é feita apenas através de 2 dispositivos, nesse modo é fácil realizar a configuração de parâmetros dos módulos, além da ampla variação de comandos AT para configuração de parâmetros, porém, esse modo tem muitas limitações, como exemplo, não é possível saber a origem ou a rota dos dados devido a estrutura do pacote dos dados, devido a simplicidade da comunicação, não é possível criação de uma rede ampla com várias módulos conectados.

A comunicação usada no caso da proposta será a API, que é a mais recomendada devido à complexidade da rede, esse modo especifica como os módulos de uma rede XBEE podem

interagir com comandos, respostas a comandos e status dos módulos na rede pelos quais são enviados e recebidos utilizando a comunicação serial UART. A figura abaixo delimita a estrutura dos dados API.

Figura 20 – Estrutura de Dados



Fonte: IFSC (2014)

- **START DELIMITER**, esse byte informa aos módulos XBEE que iniciou uma comunicação em modo API. Quando os módulos receberem o byte 0x7E perceberão que se iniciou uma comunicação API e passarão a decifrar os bytes seguintes como um pacote dado.
- **LENGHT**, esse byte representa o comprimento da data frame aos módulos remotos.
- **DATA FRAME**, é dividido em 2 partes cmdID e cmdDATA, na qual o cmdID informa aos módulos XBEE qual o comando a ser executado e o cmdDATA contém o endereço de 16 a 64 bits do dispositivo remoto.
- **CHECKSUM**, é o verificador de dados, para garantir a integridade do pacote é feita a soma de todos os bytes do frame data com o pacote CheckSum, se a soma der 0Xff o pacote foi recebido sem perda de informação.

Para configuração dos módulos XBEE existem uma gama variada de pacotes, relativos à comunicação ZIGBEE, que podem ser enviados pela UART do XBEE a tabela abaixo exemplifica alguns, para que se possa integrar sensores e outros dispositivos eletrônicos, apenas 2 pacotes podem ser usados no modo API, o *ZigBee IO Data Sample RX Indicator* e o *XBEE Sensor Read Indicator*, porém esse último pode ser usado apenas em conjunto com uma placa específica para acoplagem de sensores, o pacote *ZigBee IO Data Sample RX Indicator* entretanto pode ser usado sem necessariamente precisar de alguma placa específica acoplada ao módulo desde que os componentes eletrônicos respeitem as especificações técnicas de tensão e corrente das suas entradas e saídas analógicas/digital. O quadro 2 mostra alguns pacotes Zigees:

Quadro 2 – Nomes e Valores dos Pacotes ZIGBEE no modo API

ZigBee Receiver Packet	Recebe um pacote pelas portas UART do módulo	0x90
ZigBee IO Data Sample RX Indicator	São enviados pela UART os status das portas digitais e os valores da conversão analógica	0x92
XBEE Sensor Read Indicator	Recebe leituras dos 33sensors	0x94
Nome do dado API ID	Função	Valor em Hexadecimal
ZigBee Transmit Request	Comando ZIGBEE que solicita uma transmissão para um determinado endereço	0x10
ZigBee Transmit Status	Indica se a transmissão do último pacote foi transmitida com sucesso	0x8B

Fonte: Autoria Própria

3.3 SENSOR DE MOVIMENTAÇÃO

Para a execução deste projeto, utilizou-se o sensor de movimento passivo MH-SR602. Trata-se de um sensor infravermelho de baixo custo, alimentado com uma tensão de 3.3 VDC e com saída de 3,3VDC (compatível com a porta de entrada digital do módulo Xbee). Esse sensor possui sensibilidade de detecção ajustável e cobertura de até 3 – 5 metros. A *Figura 21* mostra um sensor de movimento.

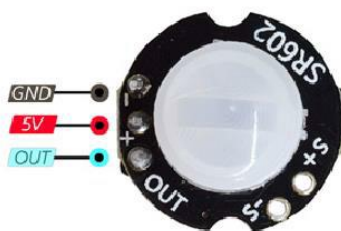
Figura 21 – Sensor de Movimento MH – SR602



Fonte: Autoria Própria

O princípio de funcionamento é simples o sensor detecta movimentos medindo as variações de calor no ambiente. Quando uma pessoa ou objeto se move dentro do campo de visão do sensor, ocorre uma mudança no padrão de calor, que é detectada pelo sensor. O sensor geralmente fornece uma saída digital que é ativada quando o movimento é detectado. Isso pode ser usado para acionar dispositivos ou sistemas em resposta a um movimento. A figura 22 mostra como é a conexão do MH-SR602.

Figura 22 – Conexões do MH-SR602



Fonte: Autoria Própria

3.4 SENSOR DE TEMPERATURA

O sensor MCP9700 é um dispositivo de sensoriamento de temperatura analógico fabricado pela Microchip Technology Inc. Ele é projetado para medir a temperatura ambiente ou a temperatura de um objeto próximo a ele. Aqui estão algumas informações sobre o sensor MCP9700, o princípio de funcionamento é simples o MCP9700 utiliza o princípio de junção de transistor de efeito de campo de metal-óxido-semicondutor (MOSFET) para medir a temperatura.

Ele possui uma junção semicondutora que muda sua resistência com base na temperatura ambiente, ele possui uma faixa de medição de -40°C a $+125^{\circ}\text{C}$. Ele fornece uma saída analógica proporcional à temperatura, com uma sensibilidade de $10\text{ mV}/^{\circ}\text{C}$, ou seja, para cada grau Celsius de variação de temperatura, a saída do sensor muda em 10 mV sua alimentação é de $2,3\text{V}$ a $5,5\text{V}$, tornando-o compatível com uma ampla variedade de circuitos eletrônicos. A tensão de saída V_{OUT} varia linearmente com a temperatura. A Figura 23 mostra como é um sensor de temperatura MCP 9700.

Figura 23 – MCP 9700

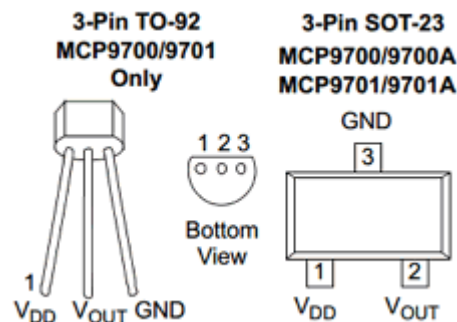


Fonte: Microchip Technology (2015)

Ele possui aplicações em projetos eletrônicos onde a medição de temperatura é necessária. Pode ser usado em sistemas de controle de temperatura, sistemas de monitoramento ambiental, termômetros digitais, entre outros, o MCP9700 possui três pinos, ele possui a conexão: V_{DD} (alimentação), V_{OUT} (saída analógica) e GND (terra). A Figura 24 é revelado o

datasheet do MCP 9700.

Figura 24 – Datasheet do MCP9700



Fonte: Datasheet-PDF (2015)

3.5 SENSOR DE LUMINOSIDADE

O sensor LDR (*Light-Dependent Resistor*), também conhecido como foto resistor ou célula fotocondutiva, é um dispositivo que varia sua resistência com base na intensidade da luz incidente sobre ele. Ele possui um princípio de funcionamento composto por um material semicondutor que possui alta resistência quando está no escuro e baixa resistência quando exposto à luz.

Quando a luz incide sobre o sensor, a quantidade de elétrons livres no material aumenta, reduzindo assim a resistência do componente, ele tem uma variação de resistência de forma inversa à intensidade da luz. Em ambientes escuros, a resistência é alta, podendo alcançar valores de megaohm ou mais. Já em ambientes bem iluminados, a resistência pode cair para alguns poucos ohms. Os sensores LDR são mais sensíveis a determinados comprimentos de onda da luz, dependendo do material semicondutor utilizado. Alguns sensores são mais sensíveis à luz visível, enquanto outros podem ser mais sensíveis a infravermelho ou ultravioleta. A figura 25 nos mostra uma imagem de um LDR.

Figura 25 – Imagem do LDR



Fonte: Eletrogate (2010)

O sensor LDR é frequentemente utilizado em aplicações de detecção e controle de luz. Alguns exemplos incluem:

- Luminárias inteligentes: O sensor LDR pode ser usado para ajustar automaticamente a intensidade da luz em uma sala de acordo com a iluminação ambiente.
- Sistemas de segurança: Pode ser empregado em sistemas de segurança para detecção de movimento com base em mudanças na luz ambiente.
- Fotocélulas: Os sensores LDR são utilizados em fotocélulas para detectar a presença de objetos ou pessoas em dispositivos como portas automáticas ou sistemas de iluminação pública.
- Controle de exposição em câmeras: Em câmeras fotográficas, o sensor LDR pode ser utilizado para medir a quantidade de luz disponível no ambiente e ajustar automaticamente as configurações de exposição

Os sensores LDR têm algumas limitações. Eles podem apresentar uma resposta relativamente lenta às mudanças de luz e não são adequados para aplicações que requerem alta precisão na medição da intensidade luminosa. Além disso, a resposta do sensor pode ser influenciada por outros fatores, como temperatura e interferência de luz ambiente não desejada.

4. DESENVOLVIMENTO

4.1 DISPOSITIVOS ZIGBEE

O desenvolvimento do XBEE foi dividido em 2 partes:

- Configuração do Firmware
- Projeto Eletrônico

A configuração do Firmware corresponde a definição da rede, segurança, entradas e saídas e o tempo de envio de cada módulo, e o projeto eletrônico corresponde a integração dos sensores aos XBEE e suas particularidades.

4.1.1 CONFIGURAÇÃO DO FIRMWARE

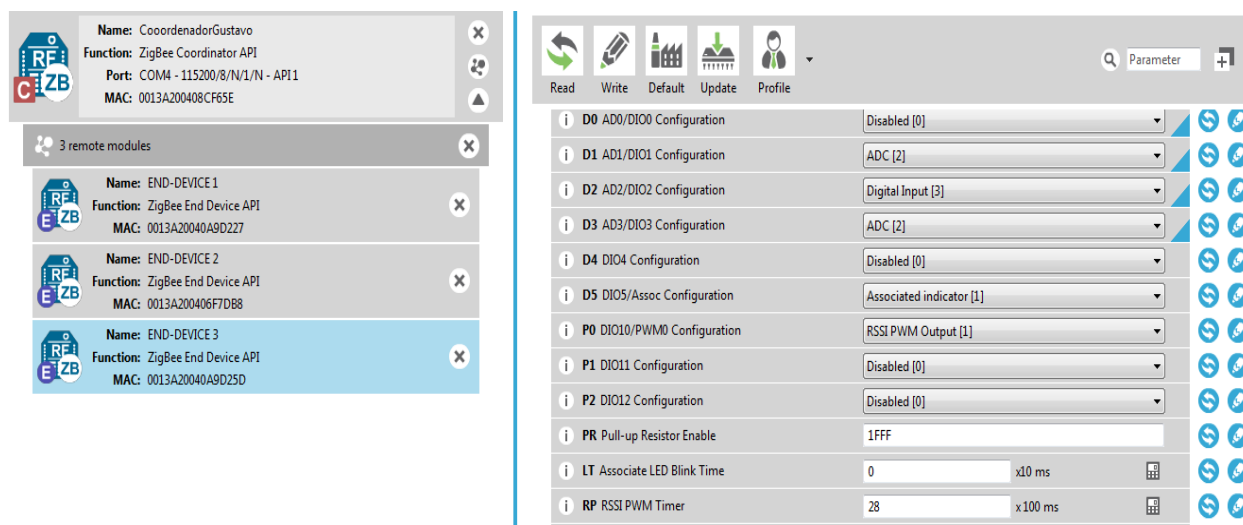
Foram utilizados 3 módulos Xbee e um módulo Xstick, para atuar como coordenador da rede, os módulos XBEE foram configurados como dispositivos finais ou “END-DEVICES”, e sua relação de endereço com sua denominação foi registrado na tabela. A *Tabela 2* mostra a relação de dispositivos usados, e a figura 26 é revelado uma visão geral dos módulos XBEE do projeto.

Tabela 2 -Relação de Dispositivos / Endereços

Nome de Módulo	Endereço de 16 Bits
Coordenador	13A200408CF65E
END-DEVICE 1	13A20040A9D227
END-DEVICE 2	13A200406F7DB8
END-DEVICE 3	13A20040A9D25D

Fonte: Autoria Própria

Figura 26 – Visualização dos Módulos XBEE



Fonte: Autoria Própria

Vale ressaltar que os End-Devices nada mais são que os dispositivos que são responsáveis pela coleta e transmissão de dados, sempre será o dispositivo que está conectado ao sensor. As devidas configurações foram desenvolvidas usando o software XCTU, desenvolvido pela empresa DIGI, e cada módulo foi configurado os seguintes parâmetros:

- rede
- segurança
- sleep
- entradas e saídas I/O

4.1.1.1 PARÂMETROS DE REDE

Para que os módulos XBEE possam realizar a transmissão de dados e criar uma rede, é necessário a definição de um módulo coordenador para que seja possível que os Xbee possam operar. No caso foi configurado o módulo Xstick como “CoordenadorGustavo” e através dele foi criada a Rede “12” que está alocada pelo parâmetro “PAN ID”, logo o coordenador é o ponto focal da transmissão os End – Devices sempre transmitem as informações a ele. A figura 27 mostra a configuração dos Parâmetros de segurança da rede Zigbee.

Figura 27 – Visualização dos Parâmetros Configurados

▼ Networking
Change networking settings

i ID PAN ID	12	
i SC Scan Channels	3FFF	Bitfield
i SD Scan Duration	3	exponent
i ZS ZigBee Stack Profile	2	
i NJ Node Join Time	FF	x 1 sec
i OP Operating PAN ID	12	
i OI Operating 16-bit PAN ID	B368	
i CH Operating Channel	16	
i NC Number of Remaining Children	9	

Fonte: Autoria Própria

Através dessa rede que os módulos irão se comunicar, enviando e recebendo dados entre eles e o coordenador. Os parâmetros “SC” (*Scan Channels*), “SD” (*Scan Duration*) e “NJ” (*Node Join Time*) é padrão do firmware. Vale lembrar que para que outros módulos possam ser inseridos na rede ZigBee criada é necessário usar a rede com o PAN ID criado pelo módulo Coordenador Xstick. O ZigBee Stack Profile é o que define qual protocolo será usado pelo módulo, sendo 2 o corresponde ao ZigBee PRO. A Tabela 3 apresenta os parâmetros de rede da Zigbee.

Tabela 3 – Parâmetros da Rede ZigBee

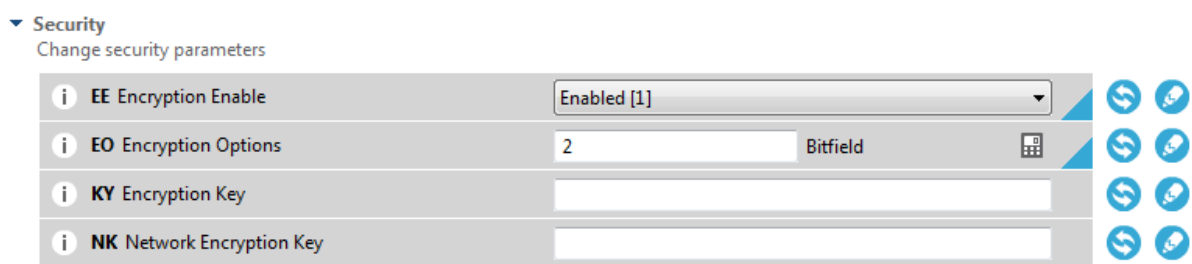
PAN ID	12
ZigBee Stack Profile	2

Fonte: Autoria Própria

4.1.1.2 PARÂMETROS DE SEGURANÇA

Para garantir que dispositivos não autorizados não se conectem à rede, é necessário ativar o sistema de criptografia e autenticação, o parâmetro “EE” (*Encryption Enable*) precisa estar habilitado para que seja possível escolher o tipo de encriptação, o parâmetro “EO” (*Encryption Options*) configura o tipo de criptografia, no caso do projeto a melhor opção é “2” (ou “010” no *Bitfield*) que utiliza o “Center True” que é um conceito na qual os módulos enviam uma chave hexadecimal para o módulo coordenador da rede, e essa chave será comparada com a chave pré configurada no coordenador e caso sejam iguais o “Center True” retorna 1 e a mensagem é autenticada e recebida, caso seja devolvido 0 a mensagem não é autenticada. Vale ressaltar que nesse caso, a pessoa que desenvolve a rede precisa configurar todos os módulos com a mesma chave de criptografia (*NK Network Encryption Key*) que a do módulo coordenador, para que seja possível estabelecer a comunicação entre COORDENADOR - END DEVICES. A Figura 28 mostra a configuração dos parâmetros de Criptografia do projeto.

Figura 28 - Habilitação da Criptografia



▼ Security
Change security parameters

EE Encryption Enable	Enabled [1]	
EO Encryption Options	2	Bitfield
KY Encryption Key		
NK Network Encryption Key		

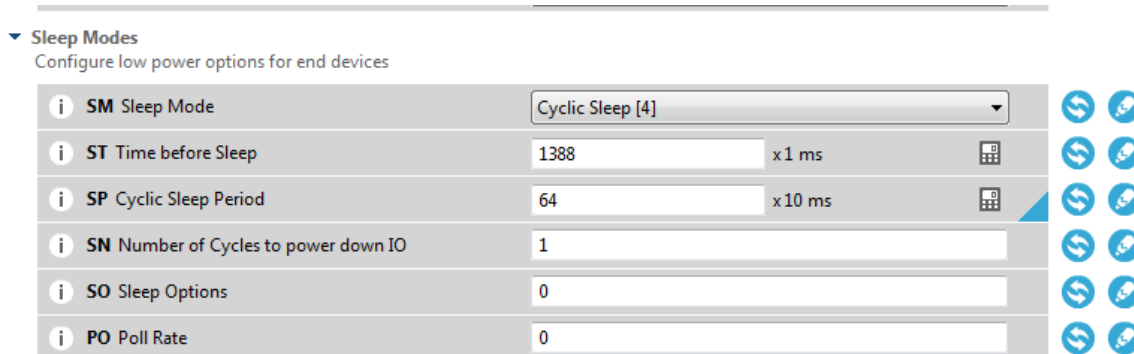
Fonte: Autoria Própria

4.1.1.3 MODO DE OPERAÇÃO – BAIXO CONSUMO DE ENERGIA

O modo de operação em baixo consumo de energia (sleep) do XBee é uma funcionalidade que permite economizar energia ao colocar o módulo em um estado de baixo consumo quando não está transmitindo ou recebendo dados. Quando o XBee está no modo sleep, ele desliga várias partes do circuito para reduzir o consumo de energia.

No entanto, o módulo ainda mantém a capacidade de ser acordado por algum evento específico, como um temporizador expirado ou um sinal externo. Para configurar os módulos foi usado Sleep Cíclico (4- *Cyclic Sleep*), nesse modo, o XBee alterna entre períodos de atividade e sono. Durante os períodos de sono, o módulo desliga certos circuitos para economizar energia, mas permanece parcialmente ativo para verificar se há eventos importantes. Ele acorda periodicamente para transmitir ou receber dados, conforme necessário. Na figura abaixo mostra os parâmetros configurados para os XBee.

Figura 29 - Habilitando o Sleep Mode



▼ Sleep Modes
Configure low power options for end devices

SM Sleep Mode	Cyclic Sleep [4]	
ST Time before Sleep	1388	x 1 ms
SP Cyclic Sleep Period	64	x 10 ms
SN Number of Cycles to power down IO	1	
SO Sleep Options	0	
PO Poll Rate	0	

Fonte: Autoria Própria

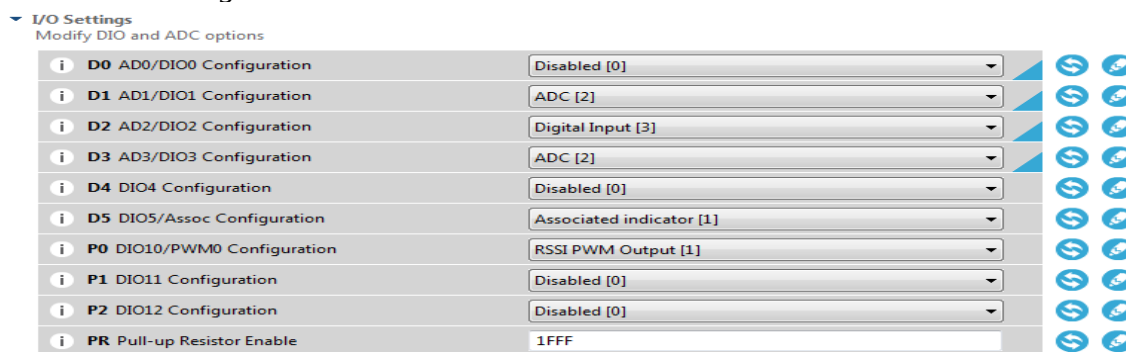
O parâmetro Sleep Time (ST), determina o tempo que o XBEE permanecerá no modo sleep antes de “acordar” para verificar se há eventos ou transmitir/receber dados. O valor do ST foi definido em 5000 milissegundos ou “1388” em Hexadecimal.

O parâmetro Sleep Period (SP) é o intervalo total entre os ciclos de sono do XBEE. Ele define o tempo total que o módulo passará dormindo e em atividade antes de repetir o ciclo. O valor de SP foi definido em 10 segundos. Ou seja, os módulos funcionarão em um tempo de 10 segundos sendo que vai dormir 5 segundos.

4.1.1.4 PARÂMETROS DE I/O – ENTRADAS E SAÍDAS

O XBee tem 4 entradas analógicas e 12 entradas digitais em sua estrutura, porém foram consideradas apenas 3 entradas nos módulos, sendo 2 entradas analógicas AD3 (conectado ao sensor de Temperatura) e AD1 (conectado ao sensor de luminosidade), a terceira entrada é digital DIO2 que foi integrada ao sensor de presença. A *Figura 30* mostra como foi habilitado as E/S dos módulos XBEE.

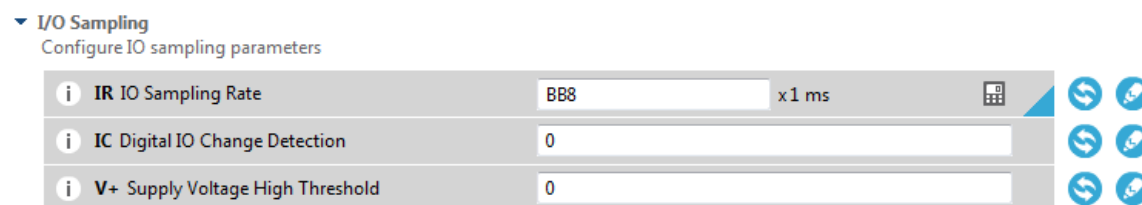
Figura 30 - Habilitando as Entradas e Saídas dos Módulos



Fonte: Autoria Própria

Para realizar o monitoramento de cada entrada habilitada no módulo XBEE, é necessário habilitar no XCTU o parâmetro “IR” (*IO Sampling Rate*) da aba “I/O Sampling Rate”, nesta aba foi configurado o valor de BB8 ou 3 milissegundos em hexadecimal, ou seja, a cada 3 segundos o XBEE vai transmitir o estado atual de cada entrada habilitada para o coordenador. A figura abaixo mostra a configuração dos Parâmetros da aba I/O Sampling Rate.

Figura 31 - Configuração do Parâmetro I/O Sampling Rate



Fonte: Autoria Própria

4.1.2 PROJETO ELETRÔNICO

Para a integração dos sensores aos módulos XBEE, foi necessário seguir alguns critérios

relacionados a algumas limitações técnicas dos módulos, como:

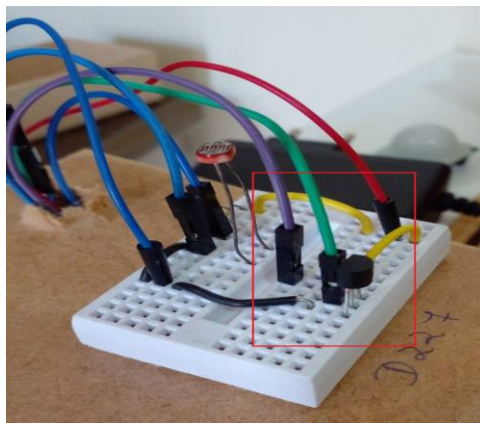
1. O Módulo XBEE suporta apenas 1.2 V em suas entradas analógicas.
2. A placa INT 700 consegue fornecer apenas 3.3 V para alimentar os dispositivos.

Seguindo esses critérios então foi possível escolher os sensores certos para a aplicação, os sensores de luminosidade e temperatura precisam ter uma resposta de saída que siga o range de 0 a 1.2 V.

4.1.2.1 SENSOR DE TEMPERATURA

O sensor de temperatura MCP9700 é um sensor que opera na faixa de 2.7 à 5 V de alimentação, e pode enviar sinais de saída de 0 a 1.2V, porém para temperaturas acima de 50 °C ele ultrapassa o valor de 1.2 V, o que faz dele um sensor para uso doméstico, esse sensor não é necessário de nenhum circuito externo ou de qualquer outro dispositivo eletrônico para seu pleno funcionamento, porém foi necessário usar um capacitor de 1 μ F para filtrar ruídos. A *Figura 32* mostra como foi montado o circuito de sensor de temperatura.

Figura 32 - Circuito com Sensor de Temperatura

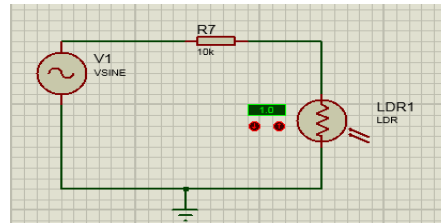


Fonte: Autoria Própria

4.1.2.2 SENSOR DE LUMINOSIDADE

O sensor de luminosidade LDR pode ser alimentado pela fonte de 3.3 V da placa INT 700, porém os valores de saída excedem e muito o limite de 1.2 V suportados pelas entradas analógicas do módulo. Logo foi -se necessário implantar um divisor de tensão como circuito externo para adequar as respostas de saídas, no caso foi usado resistores de 1 K Ω para integrar o sistema externo (resistores acima de 1 k também produzem sinal de saída adequados). A *Figura 33* foi produzida através do software Protheus 8.0 para representar o circuito divisor de tensão.

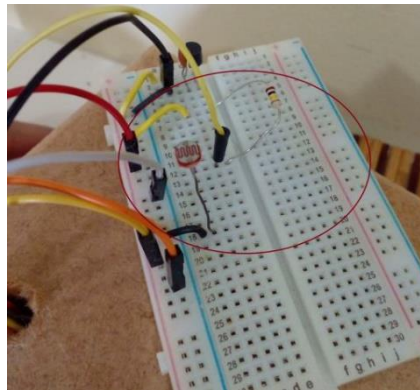
Figura 33 - Representação do Circuito Divisor de Tensão



Fonte: Autoria Própria

A figura 34 evidencia a construção do circuito LDR:

Figura 34 - Circuito com o LDR

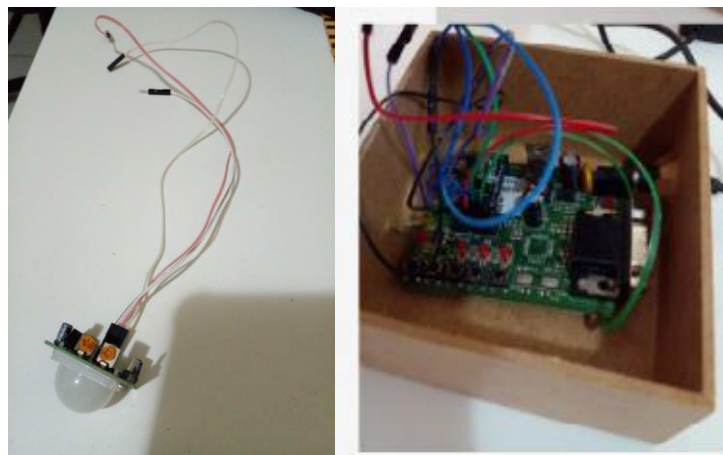


Fonte: Autoria Própria

4.1.2.3 SENSORES DE PRESENÇA

Os sensores de presença não precisavam de nenhum tipo de ajuste de algum circuito externo, devido à sua simplicidade o seu pino de saída envia um sinal de 3.3 V, que é representado como um sinal “ON”, diretamente a entrada analógica da placa INT 700 e seus cabos de alimentação são conectados a placa de protoboard. A figura 35 mostra a montagem e instalação do sensor de movimento.

Figura 35 - (a) Cabeamento do Sensor, (b) Montagem do Sensor no Módulo



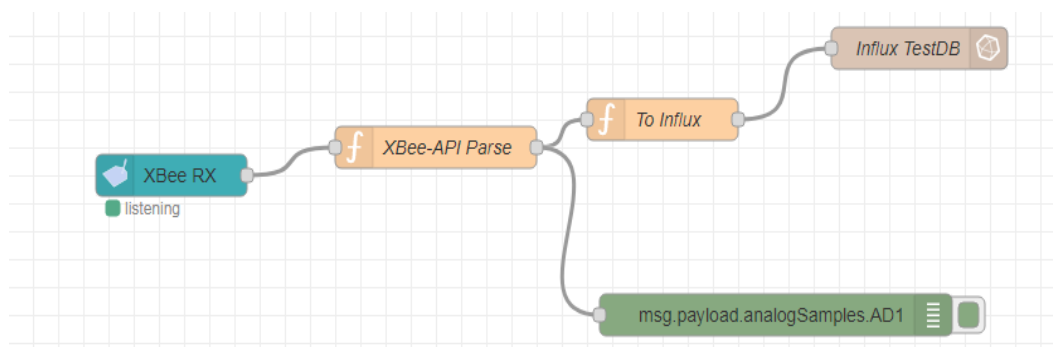
(a) (b)

Fonte: Autoria Própria

4.2 PROGRAMAÇÃO E INTEGRAÇÃO DE DADOS

A integração Node-RED é basicamente dividida entre 4 Nós (ou fluxos), como mostra a imagem abaixo.

Figura 36 - Desenvolvimento Node-RED



Fonte: Autoria Própria

Os 4 nós são:

- XBEE RX
- XBee API Parse
- To influx
- Influx TestDB

4.2.1 NÓ XBEE RX

Esse nó é uma biblioteca própria do Node-RED que tem como função receber dados do módulo coordenador XBEE através da porta serial (COM4) ao qual ele está conectado, e onde será enviado as informações para o node-red. É importante salientar que é necessário configurar o “Baud Rate” ou Taxa de Transmissão de dados para que seja igual ao dos módulos dispositivos XBee usados, para esse projeto a taxa foi de 115200. Este nó foi configurado conforme a Figura 37. É importante salientar que o parâmetro API Mode, representa o modo na qual os XBee estão configurados, que no caso seria o modo Nativo que permite o uso de envio e recebimento de dados através de sensores externos.

Figura 37 - Configuração do Nó XBee RX

Fonte: Autoria Própria

Na configuração desse nó foram editados os seguintes campos:

- **Convert ADC:** realizar a conversão dos dados para valores em mV
- **Serial Port:** Porta serial na qual o coordenador foi instalado
- **Baud Rate:** A taxa de transmissão de dados do coordenador

Os campos **API MODE** e **Value REF ADC** é padrão, ou seja, é configurado automaticamente com os valores API 1 e 1200 (valor em milivolt) respectivamente.

Nó XBEE API Parse

Esse node é uma “function” (ou função) responsável pelo tratamento e separação das variáveis, no caso os dados recebidos pelo node XBEE RX, passaram pelo processo de conversão para uma representação mais fidedigna do resultado nos painéis do dashboard do Grafana. Os sensores de Temperatura e Luminosidade foi-se necessário criar funções específicas para realizar a conversão do valor de mV para °C e Lumens respectivamente o sensor de movimento não foi necessário nenhuma função em específico.

4.2.2 CÓDIGO DO SENSOR DE TEMPERATURA

Para realizar a conversão dos dados enviados em mV para °C foi-se necessário desenvolver um algoritmo na function XBEE API PARSE. A criação da escala de temperatura foi criada através do datasheets do próprio fabricante do material (MICROCHIP), esse sensor de

temperatura tem um comportamento característico de uma função de 1º grau, ou seja, respeitando a seguinte função matemática:

$$Y = Ax + B; \quad (1)$$

O sensor MCP9700 tem a tensão de saída diretamente proporcional a temperatura medida com variação de 10 mV/°C, porém este sensor tem um de leitura erro de ± 4 °C e tem um offset de 500 mV, logo a equação característica desse sensor seria:

$$y = \frac{x - 500}{10}; \quad (2)$$

Para realizar a conversão dessa função para algoritmo no Node-RED, desenvolveu-se, o código da *Figura 38*:

Figura 38 - Algoritmo do Sensor de Temperatura

```

1 var remote64 = msg.payload.remote64;
2
3 if (msg.payload.remote64 != null)
4 {
5     if (msg.payload.analogSamples.AD3 > 0) {
6         var temp = msg.payload.analogSamples.AD3;
7         msg.payload.valor = ((temp - 500) / 10);
8         msg.payload.id = msg.payload.remote64;
9         msg.payload.local = 'Gasi';
10        msg.payload.tipo = "Sensor";
11        msg.payload.variavel = "Temperatura";
12        msg.payload.unidade = "°C";
13        msg.measurement = "Paciencia";
14        node.send(msg);
15    }

```

Fonte: Autoria Própria

No caso do sensor de temperatura ele está conectado à entrada analógica AD3, então foi necessário introduzir uma função condicional “IF” para filtrar as informações provenientes de outras entradas.

4.2.3 CÓDIGO DO SENSOR DE LUMINOSIDADE

Para realizar a conversão dos dados enviados em mV para correspondente em Lumens, foi-se necessário criar uma escala própria, uma vez que não foi possível adquirir um sensor de luminosidade para corroborar o resultado.

A escala numérica foi desenvolvida através do teste prático, com o auxílio de um multímetro e medindo o valor da saída do circuito com o LDR, foi possível obter alguns valores

representados pelos extremos “totalmente acesso” e “totalmente apagado”, vale ressaltar que 1º extremo significa um ambiente totalmente iluminado e o 2º extremo um ambiente totalmente escuro.

Para o código também precisou incluir uma função condicional “IF”, para filtrar as informações enviadas por outras entradas, no caso o sensor de temperatura está conectado a entrada analógica AD1, então sempre que alguma informação for enviada por essa entrada a função desenvolvida para a conversão de luminosidade será acionada. A *Tabela 4* mostra os valores de referência usados.

Tabela 4 – Valores de Referência

Valor em ambiente “totalmente acesso”	0 v
Valor em ambiente “totalmente escuro”	1200 mV

Fonte: Autoria Própria

A partir dessas informações, e partindo do suposto que o sensor LDR possui uma característica linear, foi possível realizar uma análise criando duas equações de retas:

$$100 = 0a + b \quad .(1)$$

$$0 = 1200a + b \quad .(2)$$

A partir desse sistema de equações é possível criar uma equação geral para a função:

$$Y = -1/12x + 100; \quad .(3)$$

Figura 39 - Algoritmo do Sensor de Luminosidade

```
if (msg.payload.analogSamples.AD1 > 0) {
  var y = msg.payload.analogSamples.AD1;
  var tot = -0.09 * y + 108;
  if(tot <= 100){
    msg.payload.valor = parseFloat(tot.toFixed(1));
  }
  msg.payload.id = msg.payload.remote64;
  msg.payload.local = 'Gasi';
  msg.payload.tipo = "Sensor";
  msg.payload.variavel = "Luminosidade";
  msg.payload.unidade = "%";
  msg.measurement = "Paciencia";
  node.send(msg);
}
```

Fonte: Autoria Própria

4.2.4 CÓDIGO DO SENSOR DE PRESENÇA

Para esse sensor não foi preciso de nenhum ajuste em específico para o sinal de saída.

Como se trata de uma entrada digital então só é possível 2 valores possíveis: “0” ou “1”. A figura seguir mostra o desenvolvimento do algoritmo do sensor.

Figura 40 - Algoritmo do Sensor de Movimento

```

if (msg.payload.digitalSamples.DIO2 != null) {
  msg.payload.valor = msg.payload.digitalSamples.DIO2;
  msg.payload.id = msg.payload.remote64;
  msg.payload.local = 'Gasi';
  msg.payload.tipo = 'Atuador';
  msg.payload.variavel = 'Presença';
  msg.payload.unidade = 'Binario';
  msg.measurement = "Paciencia";
  node.send(msg);
}

```

Fonte: Autoria Própria

4.2.5 NÓ TO INFLUX

O node “To Influx” é uma “function”, é o responsável por levar as informações para o banco de dados InfluxDB, para que ocorra a transferência dos valores que foram tratados pelo node XBEE API Parse faz se necessário que os valores sejam transferidos para as variáveis **valor, id, local, tipo, variável, unidade e professor**, como está representado pela imagem.

Figura 41 - Código do Node "To Influx"

```

1
2 var valor = msg.payload.valor
3 var id = msg.payload.id
4 var local = msg.payload.local
5 var tipo = msg.payload.tipo
6 var variavel = msg.payload.variavel
7 var unidade = msg.payload.unidade
8 var professor = msg.payload.professor
9

```

Fonte: Autoria Própria

Vale Ressaltar que a criação das variáveis nesse formato, é resultado da padronização de informações da própria plataforma IoT, como o trabalho se utiliza de uma infraestrutura existente foi necessário adequar os nomes das variáveis criadas aos respectivos nomes:

- **valor:** respectivo valor coletado pelo sensor
- **id:** identificação do dispositivo, ou seja, a identificação de 64 bits do XBEE
- **local:** local onde o modulo está instalado
- **tipo:** se um sensor, atuador etc.
- **variável:** o tipo do sensor ou atuador, sé presença, temperatura etc.

- **unidade:** unidade de medida, lumens, binário etc.
- **professor:** Professor responsável do projeto

Para que ocorra a escrita correta no InfluxDB é acrescentado mais uma lógica em que é feita as devidas separações entre as variáveis do campo “Field Value” e as variáveis “Tag Value” que seriam as variáveis de valor e string respectivamente. como apresentado na imagem. Nesse código é necessário acrescentar duas informações que no caso seria o tipo de rede (que no caso do projeto seria ZigBee) e a aplicação (identidade do projeto), essas informações são necessárias para correta identificação da plataforma. A figura abaixo mostra o código completo para a transmissão de dados para o Influx.

Figura 42 - Código de Transmissão de Informação para o Influx

```
{
  Valor: valor
},
{
  ID: id,
  Aplicacao: "PFC_GUSTAVO",
  Local: local,
  Tipo: tipo,
  Variavel: variavel,
  Unidade: unidade,
  Rede: "ZigBee",
  Professor: "Paciencia"
}];
```

Fonte: Autoria Própria

Esse método permite que os dados possam ser escritos em 9 colunas no Influx DB.

4.2.6 NÓ INFLUX TEST DB

Esse nó conecta o Node - Red com o Influx DB, ele é o responsável pelo envio das informações para o banco de dados designado. A imagem 43 mostra a configuração desse nó.

Figura 43 - Configuração do Node InfluxDB

The screenshot shows the 'Properties' tab of an InfluxDB node configuration. The fields are as follows:

- Name:** Influx TestDB
- Server:** [v2.0] http://186.217.146.134:8086
- Organization:** UNESP
- Bucket:** Test
- Measurement:** (empty)
- Time Precision:** Milliseconds (ms)

A tip box at the bottom states: "Tip: If no measurement is specified, ensure `msg.measurement` contains the measurement name."

Fonte: Autoria Própria

Os seguintes campos desse node foram editados:

- **Name:** nome do node;
- **Organization:** organização responsável (deve ser a mesma escolhida no InfluxDB);
- **Bucket:** nome da base de dados criada no InfluxDB

É importante ressaltar que é importante configurar o campo **Server** que é responsável pela conectividade ao servidor do InfluxDB, nele é necessário inserir o token e a versão do Influx utilizado. A *Figura 44* mostra a configuração do Server.

Figura 44 - Configuração do Parâmetro Server

The screenshot shows the 'Properties' tab of an InfluxDB node configuration. The fields are as follows:

- Name:** Name
- Version:** 2.0
- URL:** http://186.217.146.134:8086
- Token:** (masked with dots)
- Verify server certificate:** ☒

Fonte: Autoria Própria

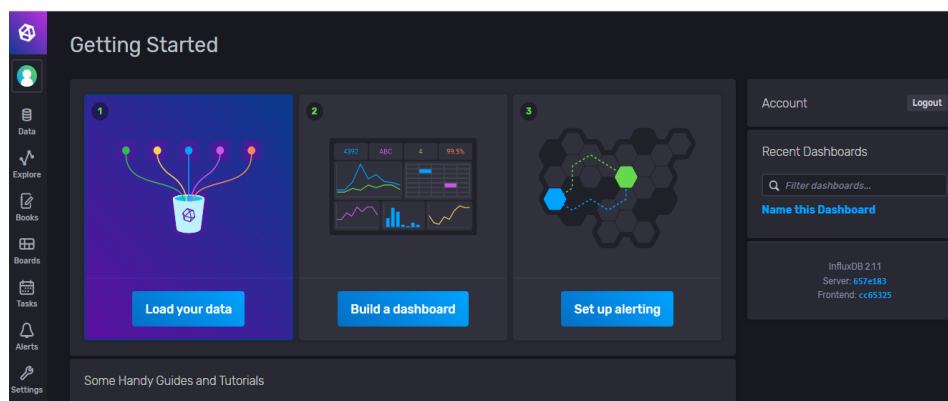
4.3 ARMAZENAMENTO E VISUALIZAÇÃO DE DADOS

4.3.1 BANCO DE DADOS INFLUXDB

Para o desenvolvimento do banco de dados foi criado uma base de dados denominada “Test”, cada módulo XBee vai enviar uma informação de cada sensor acoplado nele, essa

informação será captada pelo coordenador que coletará cada dado e enviará para o tratamento via Node-RED na qual cada informação será separada e readequadas para ser armazenada no InfluxDB, vale ressaltar que o InfluxDB usa a linguagem Flux para consultar e armazenar dados. A figura 45 mostra como é a interface do usuário do InfluxDB.

Figura 45 – Interface de Usuário do InfluxDB



Fonte: Autoria Própria

Para acesso às informações dos dispositivos XBee, é criada uma query. Segundo Markus Winand (2018) uma query (ou consulta, em português) é uma instrução ou comando utilizado em linguagens de programação de bancos de dados para recuperar, manipular e gerenciar informações armazenadas em um banco de dados. Essas instruções permitem que os usuários interajam com os dados de maneira estruturada e eficiente.

Criação das QUERYs

Para criação da query é necessário que alguma informação seja enviada para o InfluxDB, nesse caso quem será o intermediário responsável pelo envio é o Node-RED, o InfluxDB apenas capta a informação proveniente do Node-RED e realiza o devido filtro de cada uma das variáveis, a função “filter” é a responsável pela ação, a seguir é apresentado um exemplo de query gerada para o sensor de Temperatura.

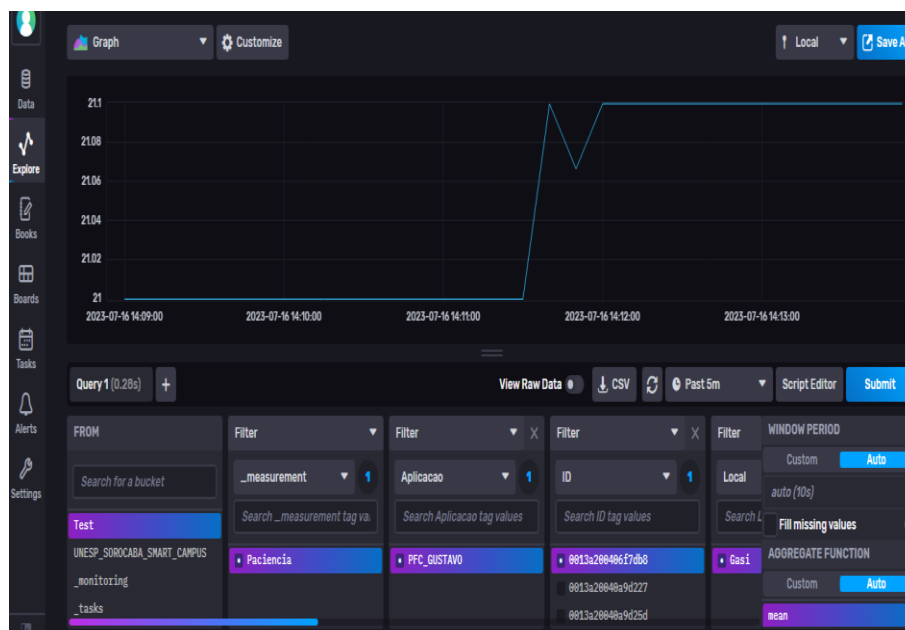
Query do Sensor de Temperatura

```
> filter(fn: (r) => r["_measurement"] == "Paciencia")
> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
> filter(fn: (r) => r["ID"] == "0013a200406f7db8")
> filter(fn: (r) => r["Local"] == "Gasi")
> filter(fn: (r) => r["Professor"] == "Paciencia")
> filter(fn: (r) => r["Rede"] == "ZigBee")
```

```
|> filter(fn: (r) => r["Tipo"] == "Sensor")
|> filter(fn: (r) => r["Unidade"] == "°C")
|> filter(fn: (r) => r["Variavel"] == "Temperatura")
|> filter(fn: (r) => r["_field"] == "Valor")
```

No caso essa query foi criada por uma informação vinda do modulo “13A200406f7db8” que no caso é o modulo instalado na sala 1. É importante ressaltar que cada XBee estará enviando uma informação referente a cada sensor, porém a estrutura da query será a mesma, o que difere será o campo “ID” contendo cada endereço do XBee, após a criação da query é criado um gráfico correspondente pela coleta das queries em um determinado espaço de tempo, que nesse caso é de 5 minutos, o gráfico gerado é apresentado pela *Figura 46* abaixo.

Figura 46 - Visualização de uma coleta de Variáveis do sensor de temperatura



Fonte: Autoria Própria

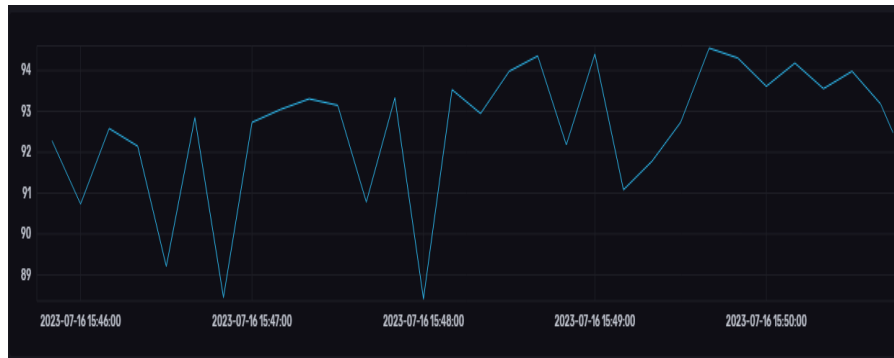
O código abaixo é referente à query criada para uma informação que provém de um sensor de Luminosidade da Sala 1, a *Figura 47* representa a coleta das queries em um range de 5 minutos.

Query do Sensor de Luminosidade

```
|> filter(fn: (r) => r["_measurement"] == "Paciencia")
|> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
|> filter(fn: (r) => r["ID"] == "0013a200406f7db8")
|> filter(fn: (r) => r["Local"] == "Gasi")
|> filter(fn: (r) => r["Professor"] == "Paciencia")
```

```
|> filter(fn: (r) => r["Rede"] == "ZigBee")
|> filter(fn: (r) => r["Tipo"] == "Sensor")
|> filter(fn: (r) => r["Unidade"] == "%")
|> filter(fn: (r) => r["Variavel"] == "Luminosidade")
|> filter(fn: (r) => r["_field"] == "Valor")
```

Figura 47 - Gráfico do Sensor de Luminosidade



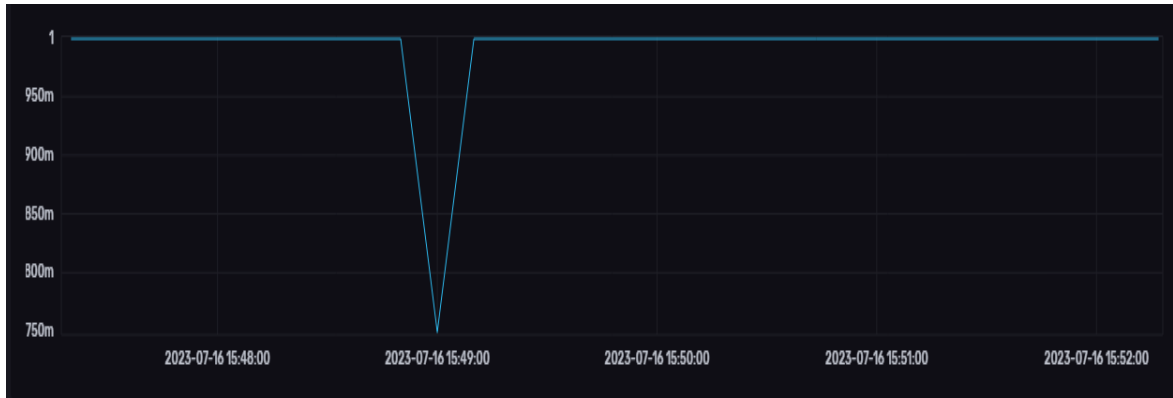
Fonte: Autoria Própria

A query referente ao sensor de movimento se encontra abaixo, no caso essa query é proveniente do sensor de movimento da sala 1 e a figura 48 mostra um gráfico de variáveis para o sensor de movimento.

Query do Sensor de Movimento

```
|> filter(fn: (r) => r["_measurement"] == "Paciencia")
|> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
|> filter(fn: (r) => r["ID"] == "0013a200406f7db8")
|> filter(fn: (r) => r["Local"] == "Gasi")
|> filter(fn: (r) => r["Professor"] == "Paciencia")
|> filter(fn: (r) => r["Rede"] == "ZigBee")
|> filter(fn: (r) => r["Tipo"] == "Atuador")
|> filter(fn: (r) => r["Unidade"] == "Binario")
|> filter(fn: (r) => r["Variavel"] == "Presença")
|> filter(fn: (r) => r["_field"] == "Valor")
```

Figura 48 - Gráfico de variáveis coletadas do Sensor de Movimento

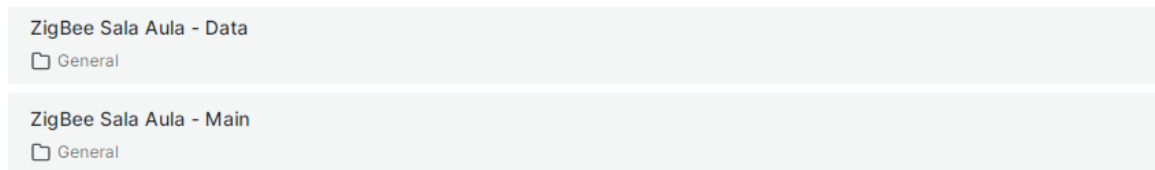


Fonte: Autoria Própria

4.3.2 INTERFACE DE VISUALIZAÇÃO DO GRAFANA

Para o desenvolvimento do Grafana foram criados dois dashboards, conforme Figura 49, denominados “ZIGBEE SALA AULA (MAIN)” e “ZIGBEE SALA AULA (DATA)”, sendo o primeiro o dashboard de apresentação do projeto e o segundo apresenta os dados históricos de coleta dos módulos, para a melhor visualização desses dados de projeto foi criado os gráficos para os sensores de temperatura, luminosidade e movimento, para 3 salas de aula. A Tabela 5 mostra a relação de dispositivos por sala.

Figura 49 - Dashboards do Projeto no Grafana



Fonte: Autoria Própria.

Tabela 5 - Relação de Endereço / Local

Endereço XBEE	Local
0013a200406f7db8	Sala 1
0013a20040a9d227	Sala 2
0013a20040a9d25d	Sala 3

Fonte: Autoria Própria

Criação do Dashboard ZIGBEE SALA AULA MAIN

Na Figura 50 é apresentado o visual do dashboard “ZigBee Sala Aula - Main”:

Figura 50 - Imagem do Dashboard Principal



Fonte: Autoria Própria

No painel é necessário colocar a query que é criada no INFLUX DB para que o gráfico possa assimilar os dados enviados do modulo correspondente. Na figura 51 é apresentado o mostrador de temperatura das salas monitoradas e na Figura 52 a aba para escrever a query.

Na Figura 53 é apresentado o painel de Monitoramento de Luminosidade de cada Sala, sendo que a unidade de medida está em % (porcentagem), esse painel é do tipo “Stat”, ou seja, apresenta o valor configurado, que no caso dessa aplicação é o próprio valor do dado recebido de um dos módulos. Os Parâmetros configurados foram:

- **Calculation:** Last (escolhe o último número enviado)
- **Fields:** Numeric Fields (apenas dados numéricos são lidos)

Figura 51 - Configuração do Painel de Luminosidade

Fonte: Autoria Própria

Figura 52 - Query do Sensor de Luminosidade

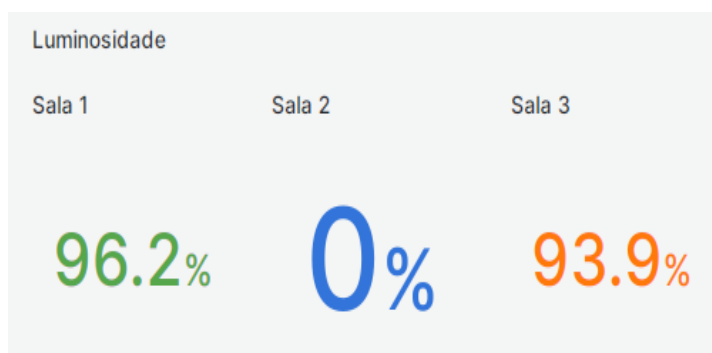
```

from(bucket: "Test")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["_measurement"] == "Paciencia")
  |> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
  |> filter(fn: (r) => r["ID"] == "0013a200406f7db8")
  |> filter(fn: (r) => r["Local"] == "Gasi")
  |> filter(fn: (r) => r["Professor"] == "Paciencia")
  |> filter(fn: (r) => r["Rede"] == "ZigBee")
  |> filter(fn: (r) => r["Tipo"] == "Sensor")
  |> filter(fn: (r) => r["Unidade"] == "%")

```

Fonte: Autoria Própria

Figura 53 - Painel de Monitoramento de Luminosidade



Fonte: Autoria Própria

Para a construção do painel que mostra o movimento das pessoas em cada sala foi usado o painel do tipo “Stat”, a figura 57 é apresentado o painel de monitoramento de Movimento de cada Sala completo, para esse painel foi necessário definir algumas regras:

- Caso seja captado movimento, será enviado o valor 1 pelo sensor, e no painel do Grafana será apresentado a palavra “SIM” em cor de fundo verde.
- Caso não seja captado movimento, será enviado o valor 0 pelo sensor, e no painel do Grafana será apresentado a palavra “NÃO” em cor de fundo vermelha.

Para configurar esse painel, na aba “Overrides” foi instaurado as regras como mostrado nas figuras 54 e 55. A aba Override 1 é referente ao envio de informação indicando que não há movimento, ou seja, o sensor irá transmitir nível logico 0, quando o sensor transmitir a informação de que foi detectado movimento, será transmitido nível logico 1 que foi configurado no Override 2.

Figura 54 - Configuração do Override 1

▼ Override 1

Fields with values

Last *	==	0
--------	----	---

▼ Value mappings

0	→	NÃO	●
---	---	-----	---

Edit value mappings

Fonte: Autoria Própria

No caso é importante ressaltar que a validação é feita pela condição de igualdade representada pelo símbolo “==” (igualdade), no caso da figura Y o último valor enviado pelo sensor é comparado com a condição instaurada, que no caso é 1, e se o valor corresponder (ou seja acusar movimento) então a resposta é a palavra “SIM” com fundo verde.

Figura 55 - Configuração Override 2

▼ Override 2

Fields with values

Last *	==	1
--------	----	---

▼ Value mappings

1	→	SIM	●
---	---	-----	---

Fonte: Autoria Própria

Na Figura 56 é apresentada a query do painel de movimento.

Figura 56 - Query do Painel de Movimento

```
from(bucket: "Test")
|> range(start: v.timeRangeStart, stop: v.timeRangeStop)
|> filter(fn: (r) => r["_measurement"] == "Paciencia")
|> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
|> filter(fn: (r) => r["ID"] == "0013a20040a9d25d")
|> filter(fn: (r) => r["Local"] == "Gasi")
|> filter(fn: (r) => r["Professor"] == "Paciencia")
|> filter(fn: (r) => r["Rede"] == "ZigBee")
|> filter(fn: (r) => r["Tipo"] == "Atuador")
|> filter(fn: (r) => r["Unidade"] == "Binario")
```

Fonte: Autoria Própria

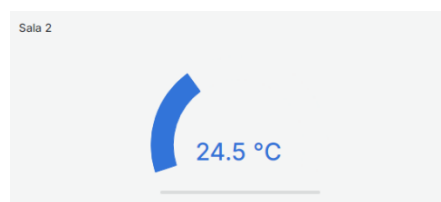
Figura 57 - Painel de Monitoramento de Salas



Fonte: Autoria Própria

Para a criação do painel de temperatura foi escolhido o gráfico do tipo “GAUGE” como mostra na figura 58, a configuração desse painel é mostrada na figura 59.

Figura 58 - Painel tipo "Gauge" de Temperatura



Fonte: Autoria Própria

Figura 59 - Configuração do Painel de Temperatura

Fonte Autoria Própria

Na figura 60 é apresentada a query para o sensor de temperatura e a Figura 61 mostra o painel de monitoramento de temperatura completo.

Figura 60 - Aba da Query

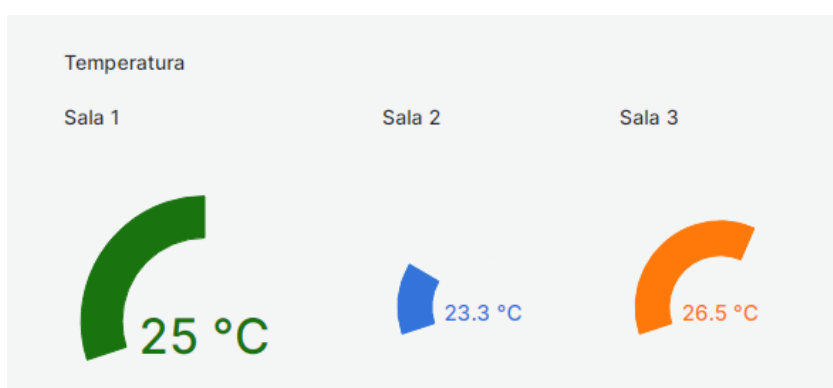
```

1  from(bucket: "Test")
2  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3  |> filter(fn: (r) => r["_measurement"] == "Paciencia")
4  |> filter(fn: (r) => r["Aplicacao"] == "PFC_GUSTAVO")
5  |> filter(fn: (r) => r["ID"] == "0013a200406f7db8")
6  |> filter(fn: (r) => r["Local"] == "Gasi")
7  |> filter(fn: (r) => r["Professor"] == "Paciencia")
8  |> filter(fn: (r) => r["Rede"] == "ZigBee")
9  |> filter(fn: (r) => r["Tipo"] == "Sensor")
10 |> filter(fn: (r) => r["Unidade"] == "°C")

```

Fonte: Autoria Própria

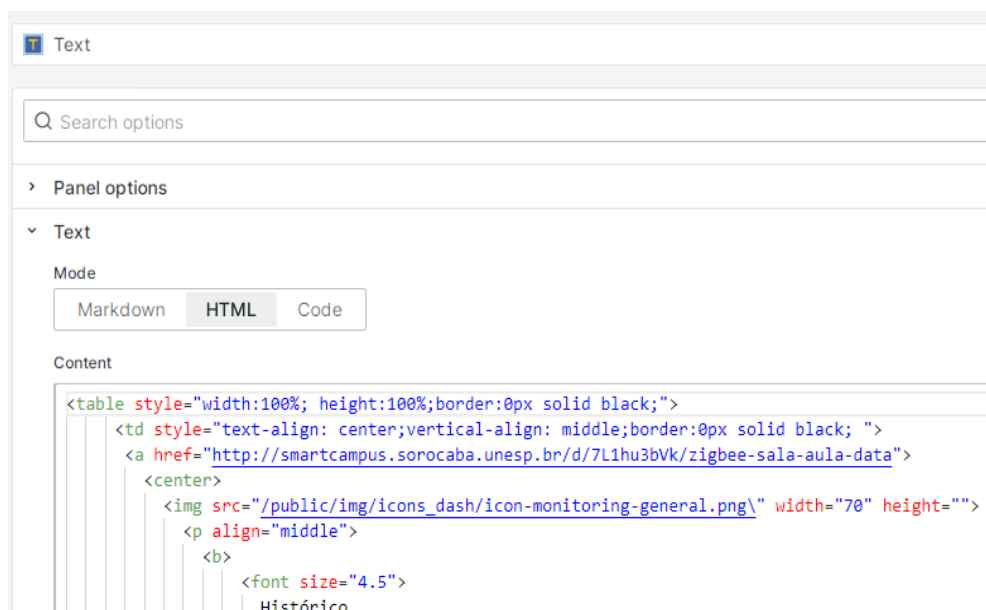
Figura 61 - Painel de Monitoramento de Temperatura



Fonte: Autoria Própria

Na Figura 64 é apresentado os botões “Histórico” que redireciona o usuário aos dados históricos coletados pelos sensores e os botões “Voltar” que retorna ao Dashboard Principal do Smart Campus. Esses plug-ins foram construídos por aplicações anteriores então eles não foram criados no caso apenas configurados para irem para os devidos locais. O Plug-In “Histórico” foi necessário colocar o link para o dashboard “Data”, como se encontra na figura 62. O Plug-In “Voltar” é configurado com o link da página principal do Smart Campus da UNESP Sorocaba, como apresentado na figura 63.

Figura 62 - Configuração do Plug-In "Histórico"



Fonte: Autoria Própria

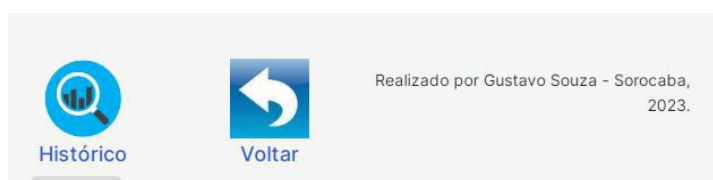
Figura 63 - Configuração do Plug-In "Voltar"



Fonte: Autoria Própria

Na Figura 64 é apresentado o painel completo:

Figura 64 - Painel dos botões Histórico e Voltar

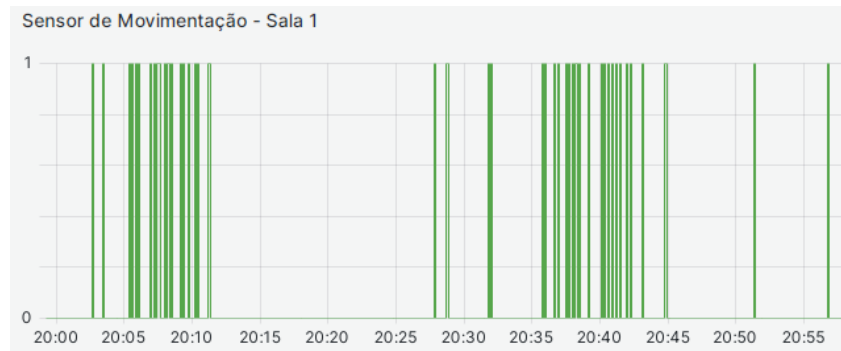


Fonte: Autoria Própria

Criação do Dashboard ZIGBEE SALA AULA DATA

Neste Dashboard é apresentado os dados históricos coletados por cada sensor, todos os gráficos são do tipo “Time Series”, ou seja, apresenta os valores ao longo de um determinado período estabelecido. A Figura 65 mostra o gráfico do sensor de movimento após a coleta de variáveis em um período de 1 hora.

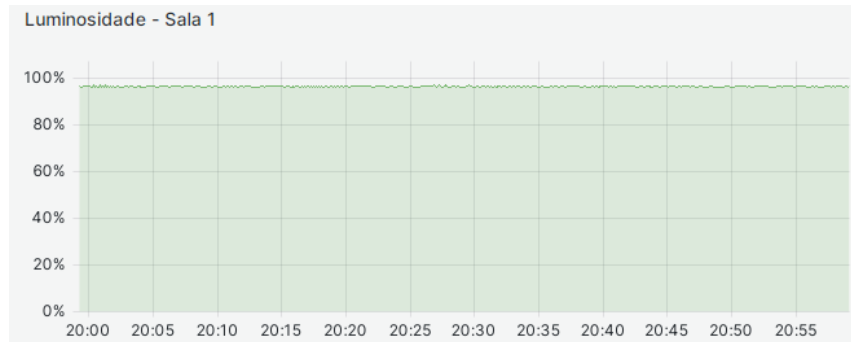
Figura 65 – Gráfico do Sensor De Movimento



Fonte: Autoria Própria

A Figura 66 mostra o gráfico do sensor de luminosidade da Sala 1 após o período de coleta de 1 hora.

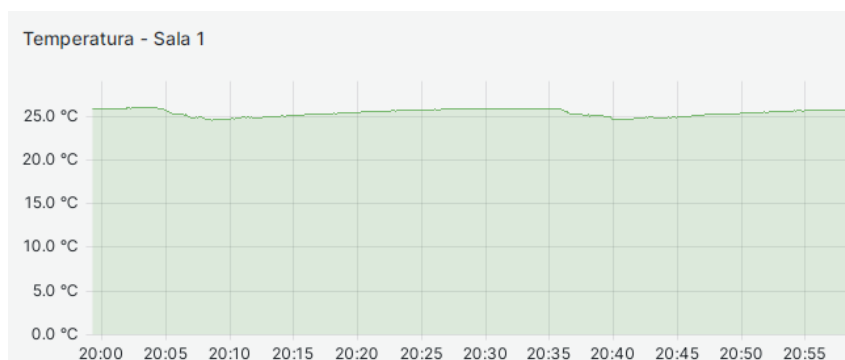
Figura 66 - Gráfico Do Sensor de Luminosidade



Fonte: Autoria Própria

A Figura 67 mostra o gráfico do Sensor de Temperatura após 1 hora de coleta de variáveis.

Figura 67 - Gráfico do Sensor de Temperatura



Fonte: Autoria Própria

É importante ressaltar que no Grafana existe uma condição de que à medida que é determinado um período de tempo muito grande para monitoramento, o GRAFANA não consegue criar os pontos para representação do gráfico, logo se faz necessário o uso da função “aggregateWindow” que no caso agrega todos os pontos em períodos de 1 minuto após o período de 6 horas, isso com o intuito de não estabelecer erro na criação dos gráficos por excesso de pontos, o código para agregação de pontos se encontra na *Figura 69*, vale lembrar que todas as queries tem o mesmo formato do código acima, as únicas informações diferentes são o tipo de sensor e o endereço do XBee. A *Figura 68* apresenta o código da query de filtragem do tempo.

Figura 68 - Código da query para filtrar período de tempo

```
if int(v:duration) <= int(v:6h) then
```

Fonte: Autoria Própria

Figura 69 - Código da query para agregar pontos

```
36 |> aggregateWindow(every: if int(v:duration) <= int(v:1d) then 1m
37 else if int(v:duration) <= int(v:7d) then 30m else 1h, fn: mean)
```

Fonte: Autoria Própria

5 RESULTADOS

Após o desenvolvimento do projeto, cada dispositivo contendo o módulo ZigBee foi posicionada em lugares estratégicos do GASI, de forma que respeite a distância de envio de dados entre o módulo coordenador. A Figura 70 mostra o posicionamento do módulo denominado “Sala 1” que foi colocado de forma a monitorar a entrada da porta do laboratório do GASI.

Figura 70 - Local do dispositivo - Sala 1



Fonte: Autoria Própria

O módulo da Sala 1 será responsável pelo monitoramento das condições do Local. A Figura 71 mostra o gráfico de Luminosidade no dia de quinta-feira (10/08/2023) do período das 07:00 às 22:00.

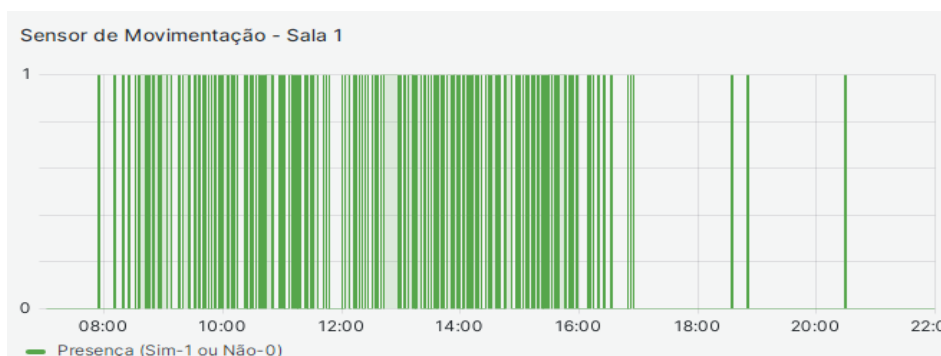
Figura 71 - Gráfico da Luminosidade - Sala 1



Fonte: Autoria Própria

No gráfico é possível visualizar que entre os períodos das 07:00 às 09:00 da manhã existe um aumento gradual da luminosidade até atingir o seu pico, no período das 10:00 às 16:00 a iluminação do ambiente começa a decrescer, no período das 16:00 às 22:00 as luzes foram apagadas. E importante salientar que a fonte emissora de luz são as lâmpadas de LED do laboratório e não a luz solar na qual é baseado a escala e por isso a porcentagem luminosa é baixa em torno dos 60 a 70 %. A Figura 72 apresenta o gráfico de monitoramento da movimentação/presença na sala 1.

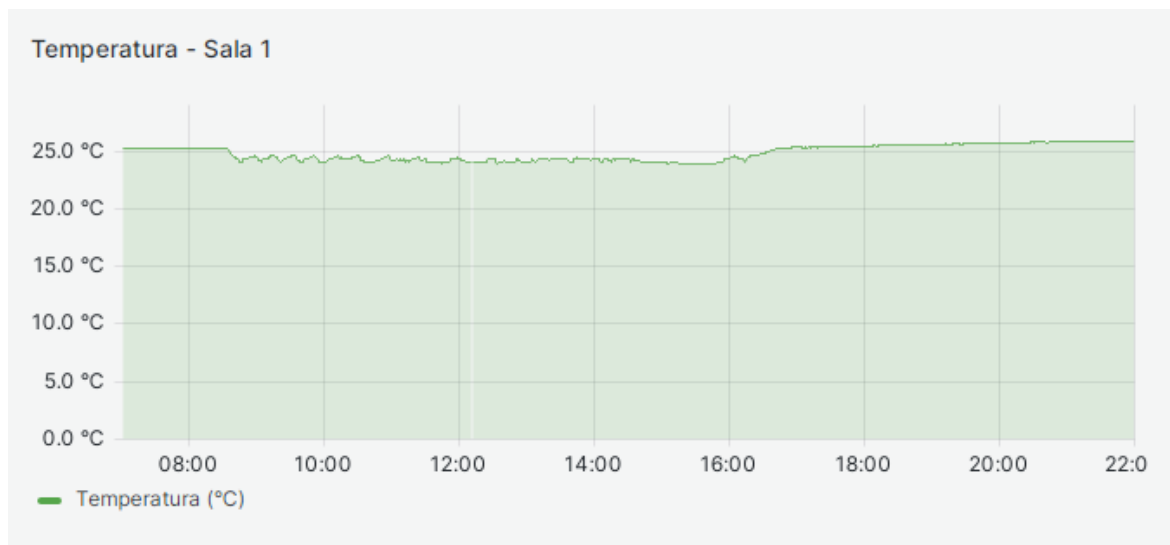
Figura 72 - Gráfico de Movimentação - Sala 1



Fonte: Autoria Própria

A Figura 72 apresenta intensa atividade de movimento principalmente no período das 08:00 às 17:00 e apresenta quase inatividade das 18:00 às 22:00. O gráfico representa a temperatura do local que praticamente se manteve constante o dia inteiro com um leve aumento no período das 18:00 às 22:00. A Figura 73 apresenta o gráfico de monitoramento da temperatura na sala 1.

Figura 73 - Gráfico da Temperatura - Sala 1



Fonte: Autoria Própria

O módulo 2 foi instalado diretamente em uma sala próxima ao GASI como mostra na figura , os gráficos de Luminosidade, movimento e temperatura são diferentes dos outros módulos, graças a algumas particularidades do próprio local, o módulo foi instalado próxima a uma janela então ao contrário dos outros módulos a fonte emissora de luz será a solar, isso quer dizer que o pico máximo de luminosidade tende a ser maior que a dos outros 2 módulos, também existe um ar-condicionado bem próximo ao módulo o que afeta diretamente o sensor de temperatura e o sensor de movimento não monitora mais pessoas que os outros 2 módulos, no caso é restringido ao professor que se encontra na sala, logo o sensor de movimento não é mais atuante do que os outros.

Figura 74 - Local do Dispositivo - Sala 2



Fonte: Autoria Própria

O gráfico de Luminosidade tem seu valor de pico na casa dos 90 % pois diferente dos outros módulos ele se encontra próximo a uma janela, logo sua fonte emissora é a luz do sol. E assim como os outros módulos, após o período das 17:00 ele tende a ter uma queda gradual, chegando a 0, devido ao fato de que vai anoitecendo.

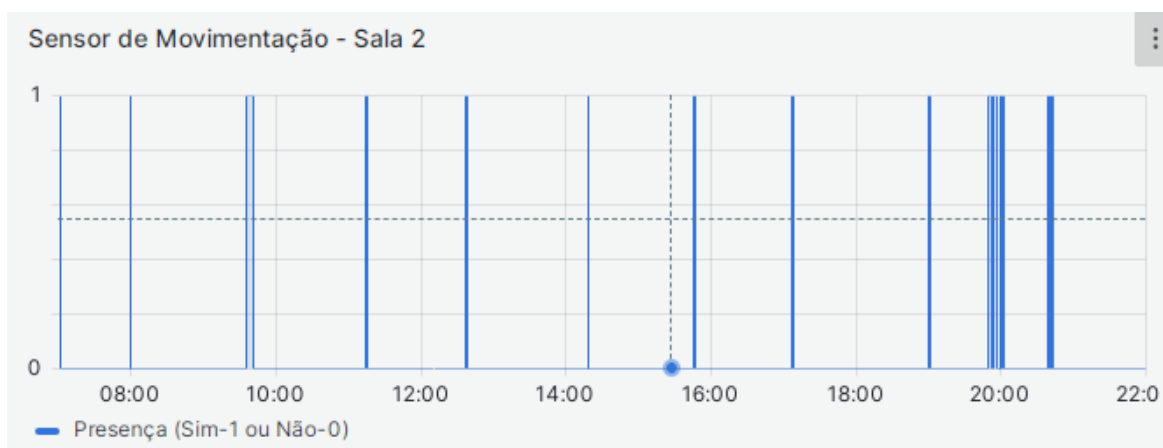
Figura 75 - Gráfico da Luminosidade - Sala 2



Fonte: Autoria Própria

De acordo com a figura 76, o sensor de movimento não foi muito atuante durante o período da manhã e tarde, entretanto no período da noite entre as 19:00 e 20:00 foi captado presença maior que o normal ao longo do dia.

Figura 76 - Gráfico do Sensor de Movimento - Sala 2

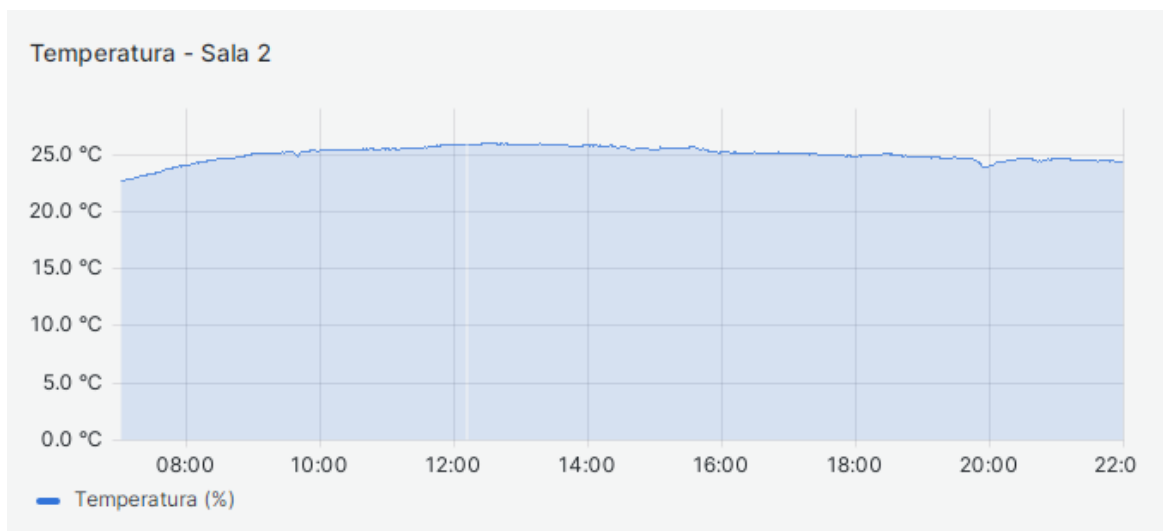


Fonte: Autoria Própria

De acordo com a figura 77 o sensor de temperatura captou temperaturas ligeiramente menores do que os outros módulos, muito por decorrência da localização do módulo que foi instalado próximo ao ar-condicionado da sala, a temperatura no final da tarde estabilizou-se

nos 24°C.

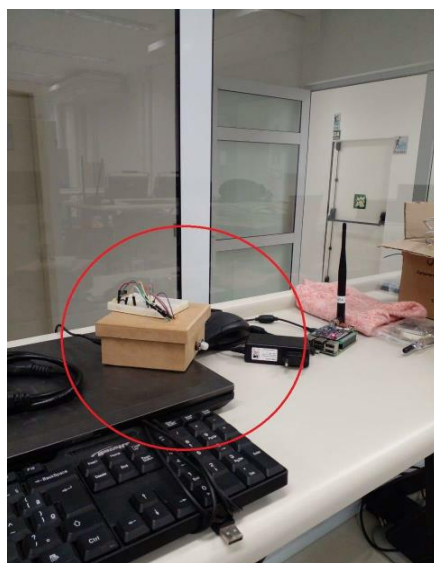
Figura 77 - Gráfico do Sensor de Temperatura - Sala 2



Fonte: Autoria Própria

O módulo da Sala 3 foi instalado no laboratório do GASI, como mostrado na figura 78, próximo à porta do laboratório, como mostrado na figura abaixo, porém de frente a bancada do setor esquerdo para monitorar esse setor.

Figura 78 - Local do Dispositivo - Sala 3



Fonte: Autoria Própria

O gráfico da Luminosidade da Sala 3, mostrado na figura 79, apresenta a mesma característica dos outros, um aumento gradual no período da manhã e decrescendo ao longo da tarde até chegar no 0 que seria quando as luzes se apagam do laboratório.

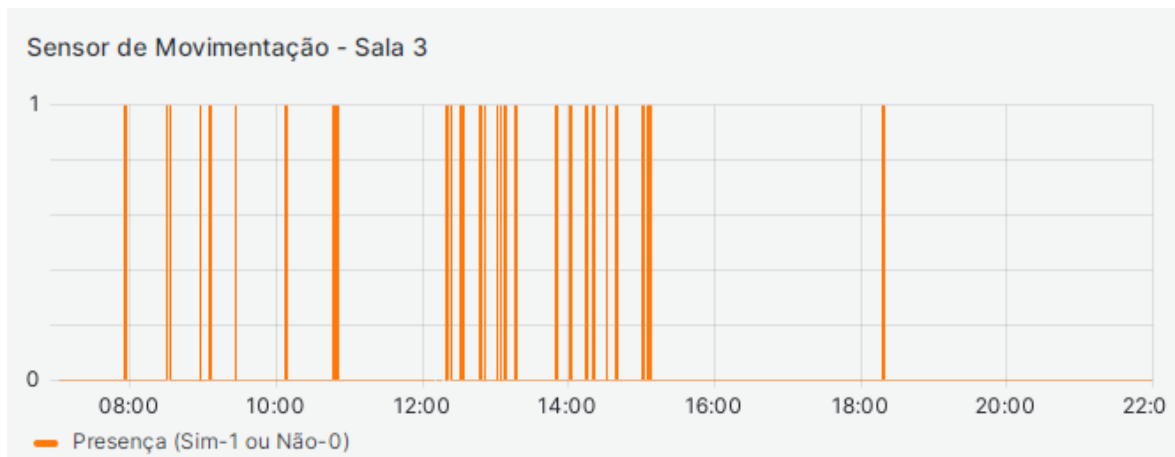
Figura 79 - Gráfico de Luminosidade - Sala 3



Fonte: Autoria Própria

O gráfico do sensor de movimentação da sala 3, figura 80, foi ativado diversas vezes ao dia, principalmente no período das 12:00 até as 15:00, após isso o fluxo da bancada diminuiu ao longo da noite.

Figura 80 - Gráfico de Movimentação - Sala 3



Fonte: Autoria Própria

O gráfico da temperatura se manteve constante na casa dos 26°C com um leve aumento para 27°C no período da noite.

Figura 81 - Gráfico do Sensor de Temperatura - Sala 3



Fonte: Autoria Própria

Em suma a infraestrutura de rede conseguiu se integrar perfeitamente a plataforma IoT do Smart Campus, os sensores conseguiram se integrar perfeitamente e pode-se coletar uma quantidade razoável de dados, todos os dados não sofreram nenhum tipo de atraso quando os módulos se integraram a rede (com exceção daqueles que foram programados com o modo Sleep para ter atrasos), é importante também pontuar algumas funcionalidades interessantes do uso do XBEE no projeto, como o modo Sleep uma importante ferramenta de temporização e para otimização do consumo de energia dos módulos, outro ponto importante a ressaltar é o uso das porta E/S analógicos e digitais, não teve problemas quanto a integração aos sensores, apesar da limitação das entradas analógicas de que apenas dispositivos de saída de 1.2 V fossem usados, foi um recurso de fácil habilitação e configuração, outro ponto importante para se ressaltar é a facilidade da integração de novos módulos XBEE a rede, mesmo o usuário adicionando um router (roteador), para expandir a rede, ou um dispositivo final (End-Device), para integrar a novos sensores, a configuração dos dispositivos XBEE em uma dessas aplicações, é bem intuitiva e de fácil manipulação, com o uso do software XCTU, o que torna esse dispositivo em uma ferramenta bem simples para se configurar.

A rede ZigBee de um modo geral não teve muitos problemas de uso, os dados foram transmitidos em sua totalidade, não houve registro de perda de informação ou qualquer outro problema semelhante, um ponto importante do uso dessa tecnologia para se ressaltar foi uso da rede em malha (ou mesh) com os módulos, essa aplicação de rede foi importante pois os módulos podem se comunicar entre si a fim de encontrar a melhor rota possível para se enviar alguma informação, foi feito testes aplicando novos módulos XBEE configurados como “Routers” ou roteadores na rede criada para o projeto e o resultado direto foi a total integração desses novos módulos a rede e por consequente o aumento do alcance da rede e novos caminhos

para direcionamento de dados, isso é um fator importante para futuros novos trabalhos que quiserem usar a tecnologia ZigBee junto com os módulos XBEE, os usuários irão poder aproveitar a infraestrutura já criada, para poder diversificar ou melhorar aplicação do Smart Campus. Outro fator vantajoso do uso da rede será o fator de Auto-organização e recuperação da rede, ou seja, a rede consegue criar caminhos alternativos caso algum dos módulos se desconecte a rede, com base em alguns testes realizados mesmo se os dispositivos forem desconectados, reconectados ou se estiverem em movimento isso não compromete a integridade da rede nem dos outros módulos conectados, assim sendo a infraestrutura Zigbee instalada é mais uma opção que com certeza vai estabelecer uma nova dinâmica para a aprendizagem e gestão eficiente no campus da universidade.

6 CONCLUSÃO

O objetivo principal foi alcançado com êxito, permitindo a integração perfeita de todos os sensores e a coleta eficaz de dados. A integração dos módulos na rede ocorreu sem atrasos significativos. Existem alguns pontos a se ressaltar referente ao uso do XBEE no projeto, esses módulos contêm algumas configurações interessantes de uso, especialmente o modo Sleep, que é uma boa opção de otimização de consumo de energia dos módulos, também é importante ressaltar que esses módulos não precisam ser programáveis o que deixa sua configuração mais fácil e intuitiva ao usuário.

A utilização da rede ZigBee revelou um desempenho sólido e confiável. A transmissão de dados ocorreu sem perdas ou interrupções significativas. A implementação da rede em malha (mesh) com os módulos foi particularmente proveitosa, permitindo uma comunicação eficaz entre eles para determinar as rotas mais eficientes para a transmissão de informações. A adição de novos módulos XBEE, foi possível expandir o alcance da rede, e proporcionar novas trajetórias para os dados.

A capacidade de recuperação da rede foi outro ponto positivo, ou seja, é uma rede que possui a capacidade de reorganizar o caminho da transmissão que a mensagem é enviada, criando alternativas para o funcionamento da rede. A desconexão, reconexão ou movimentação dos dispositivos não afetou a integridade da rede ou dos módulos. Com perspectivas futuras, considerar a incorporação de sensores adicionais, como os de umidade, poderia expandir as possibilidades do projeto. Da mesma forma, a expansão da rede para abranger mais áreas proporcionaria um entendimento mais abrangente de sua interação em cenários mais amplos e diversificados.

REFERÊNCIAS

AHMAD, I.; ASLAM, M.; SAEED, U. **ZigBee based wireless remote patient monitoring system using dynamic thresholding**. In: 2013 IEEE Conference on Open Systems (ICOS), 2013, Langkawi, Malaysia. Proceedings [...]. IEEE, 2013. p. 1-6. Disponível em: https://www.researchgate.net/publication/224262428_A_Reliable_Transmission_Protocol_for_ZigBeebased_Wireless_Patient_Monitoring. Acesso em: 08 mar. 2023.

ALIANÇA ZIGBEE. **ZigBee Specifications Overview**. Disponível em: <https://ZigBeealliance.org/ZigBee-for-developers/ZigBee-specifications/>. Acesso em: 30 abr. 2023.

CSA-IoT. **Todas as Soluções ZigBee**. Disponível em: <https://csa-iot.org/pt/todas-as-solu%C3%A7%C3%B5es/ZigBee/>. Acesso em: 15 ago. 2023.

ESCHNER, G. **ZigBee Wireless Networks and Transceivers**. Berlin: Springer, 2011. Disponível em: https://www.researchgate.net/publication/316698329_ZigBee_Wireless_Networks_and_Transceivers. Acesso em: 15 abr. 2023.

ELETROGATE. **LDR Sensor fotoresistor de 5mm**. Disponível em: <https://www.eletrogate.com/sensor-fotoreistor-ldr-de-5mm>. Acesso em: 15 mai. 2023.

GRAFANA. **The analytics platform for all your metrics**. Disponível em: <https://grafana.com/grafana/>. Acesso em: 08 mar. 2023.

GUNTUPALLI, L. et al. **ZigBee based wearable sensor network for health monitoring applications**. In: 2018 IEEE International Conference on Communications (ICC), 2018, Kansas City, MO, USA. Proceedings [...]. IEEE, 2018. p. 1-6. Disponível em: <https://acadpubl.eu/jsi/2017-115-6-7/articles/8/95.pdf>. Acesso em: 07 mar. 2023.

HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY. **Smart Campus Infrastructure**. Disponível em: <https://itsc.hkust.edu.hk/services/it-infrastructure/smart-campus-infrastructure>. Acesso em: 17 de agosto de 2023.

IEEE Standard for Low-Rate Wireless Personal Area Networks (LR-WPANs). ZigBee Alliance. ZigBee Specification v1.2. 2012. Disponível em: <https://ieeexplore.ieee.org/document/7460875>. Acesso em: 08 mar. 2023.

INFLUXDATA. **InfluxDB is a time series platform**. 2021. Disponível em: <https://www.influxdata.com/>. Acesso em: 08 mar. 2023.

INTRODUCTION. [S.1]: **Vives Smart Campus**, 2019. Disponível em: <http://projectwerk.gitbook.io>. Acesso em: 18 ago. 2023.

IPv6.br. **ZigBee usa agora 6LoWPAN: sua próxima lâmpada terá IPv6.** Disponível em: <https://ipv6.br/post/ZigBee-usa-agora-6lowpan-sua-proxima-lampada-tera-ipv6/>. Acesso em: 15 de ago. 2023.

KOBLE, Equipe. **4 Reasons Why ZigBee is a Preferred Choice for Home Automation.** Koble, 2023. Disponível em: <https://koble.sg/4-reasons-ZigBee-home-automation/>. Acesso em: 15 de agosto de 2023.

MORAES, V. O. **Sistema Embarcado para a Coleta de Dados de Geração Fotovoltaica para Campus Inteligente.** Trabalho de Graduação (Engenharia de Controle e Automação) — Instituto de Ciência e Tecnologia de Sorocaba, UNESP - Universidade Estadual Paulista, Sorocaba, 2023.

NAIR, M.G. (2019). **ZigBee Technology for IoT: A Review.** *Journal of Advanced Research in Dynamical and Control Systems*, v. 11, n. 1, p. 190-199. Disponível em: <https://ijcrt.org/papers/IJCRT2105372.pdf>. Acesso em: 15 jul. 2023.

OPENJSFOUNDATION (org.). **Node-RED: low-code programming for event-driven applications. Low-code programming for event-driven applications.** 2022. Disponível em: <https://nodered.org/>. Acesso em: 10 mar. 2023.

PROENÇA, M. H. **Arquitetura IoT para Aplicação em Smart Campus.** Trabalho de Graduação (Engenharia de Controle e Automação) – Instituto de Ciência e Tecnologia de Sorocaba, UNESP - Universidade Estadual Paulista, Sorocaba, 2022.

SHEHAB, A. et al. **Design and implementation of a ZigBee-based wireless sensor network for patient monitoring.** *Journal of Medical Systems*, v. 39, n. 6, p. 61, 2015. DOI: 10.1007/s10916-015-0262-1. IEEE Std 802.15.4-2015. Disponível em: <https://repositorio.unesp.br/handle/11449/129451?locale-attribute=es>. Acesso em: 07 mar. 2023.

SMART CAMPUS ICTS. **Tela Inicial do Smart Campus.** Disponível em: <http://smartcampus.sorocaba.unesp.br/>. Acesso em: 17 de agosto de 2023.

TANWAR, S. et al. **Internet of things (IoT): A review of enabling technologies, challenges, and open research issues.** *Journal of Network and Computer Applications*, v. 97, p. 27-54, 2017. Disponível em: https://www.researchgate.net/publication/326539025_Internet_of_Things_IoT_A_Review_of_Enabling_Technologies_Challenges_and_Open_Research_Issues. Acesso em: 15 jul. 2023.

UEMURA, L. **Sistema Embarcado para a coleta de dados da Estação Meteorológica para Campus Inteligente.** Trabalho de Graduação (Engenharia de Controle e Automação) — Instituto de Ciência e Tecnologia de Sorocaba, UNESP - Universidade Estadual Paulista, Sorocaba, 2023.

UNICAMP. **Smart Campus - UNICAMP**. Disponível em:<<https://smartcampus.prefeitura.unicamp.br/>>. Acesso em: 27 dez. 2022.

XUE, Y. et al. **Wireless sensor networks for IoT: a review**. **Journal of Industrial Information Integration**, v. 15, p. 43-53, 2019. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S2214785321036439>. Acesso em: 08 mar. 2023.

USB to XBee Wireless Adapter. **XStick USB Adapters**, 2013. Disponível em: <<http://www.digi.com/products/xbee-rf-solutions/modems/xstick>>. Acesso em: 18 ago. 2023. DIGI.

XBee™ / XBee-Pro™ ZB RF Modules. Manual do Produto. Disponível em: http://ftp1.digi.com/support/documentation/90000976_C.pdf, Acesso em: 18 ago. 2023.