



**PROPOSTA DE ESTAÇÃO
METEOROLÓGICA INTELIGENTE
BASEADA EM AMBIENTES IOT**

PEDRO HENRIQUE DA SILVA COELHO

**MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE REDES DE
COMUNICAÇÃO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**PROPOSTA DE ESTAÇÃO
METEOROLÓGICA INTELIGENTE
BASEADA EM AMBIENTES IOT**

PEDRO HENRIQUE DA SILVA COELHO

Orientador: RAFAEL TIMÓTEO DE SOUSA JÚNIOR, ENE/UNB

**MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE REDES DE
COMUNICAÇÃO**

BRASÍLIA-DF, 06 DE DEZEMBRO DE 2018.

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**PROPOSTA DE ESTAÇÃO
METEOROLÓGICA INTELIGENTE
BASEADA EM AMBIENTES IOT**

PEDRO HENRIQUE DA SILVA COELHO

MONOGRAFIA DE GRADUAÇÃO ACADÊMICA SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE GRADUANDO EM ENGENHARIA DE REDES DE COMUNICAÇÃO.

APROVADA POR:

Rafael Timóteo de Sousa Júnior, ENE/UnB
Orientador e examinador interno

Francisco Lopes de Caldas Filho, ENE/UnB
Coorientador e examinador interno

Robson de Oliveira Albuquerque, Gabinete de Segurança Institucional
Examinador externo

BRASÍLIA, 06 DE DEZEMBRO DE 2018.

FICHA CATALOGRÁFICA

PEDRO HENRIQUE DA SILVA COELHO

Proposta de Estação Meteorológica Inteligente Baseada em Ambientes IoT

2018xv, 70p., 201x297 mm

(ENE/FT/UnB, Graduando, Engenharia de Redes de Comunicação, 2018)

Monografia de Graduação - Universidade de Brasília

Faculdade de Tecnologia - Departamento de Engenharia Elétrica

REFERÊNCIA BIBLIOGRÁFICA

PEDRO HENRIQUE DA SILVA COELHO (2018) Proposta de Estação Meteorológica Inteligente Baseada em Ambientes IoT. Monografia de Graduação em Engenharia de Redes de Comunicação, Publicação xxx/AAAA, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 70p.

CESSÃO DE DIREITOS

AUTOR: PEDRO HENRIQUE DA SILVA COELHO

TÍTULO: Proposta de Estação Meteorológica Inteligente Baseada em Ambientes IoT.

GRAU: Graduando ANO: 2018

É concedida à Universidade de Brasília permissão para reproduzir cópias desta monografia de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor se reserva a outros direitos de publicação e nenhuma parte desta monografia de Graduação pode ser reproduzida sem a autorização por escrito do autor.

PEDRO HENRIQUE DA SILVA COELHO

Agradecimentos

Primeiramente, agradeço à minha família por todo o apoio e suporte dado nesse período de curso. Ao meu pai, por sempre ter incentivado e apoiado os meus estudos e a minha vida profissional. À minha mãe, por ser uma grande amiga, companheira, sempre ter me motivado e por ter me ajudado na revisão do texto. Aos meus avós, por serem a base de tudo o que é importante pra mim, principalmente ao meu avô Valder por ter me ajudado com o protótipo de madeira. E aos meus irmãos e irmãs, que são a razão para que eu queira ser alguém melhor e pra quem sempre digo a importância dos estudos.

Agradeço ao Francisco Lopes, meu coorientador, pela orientação e todas as ideias e discussões fomentadas ao longo do projeto. Ao Professor Rafael Timóteo, meu orientador, por ter me orientado e colaborado enormemente para a finalização do projeto. Ao Professor Robson de Oliveira, pelas dicas cruciais durante o projeto. Ao Lincoln, colega do laboratório UIoT, por toda a ajuda no projeto dos circuitos e nas noites viradas para resolvermos os problemas de alimentação energética. Agradeço também a todo o pessoal do laboratório que contribui para o funcionamento do mesmo.

Gostaria de agradecer também aos professores, que se dedicam a cada dia para que o conhecimento seja corretamente difundido no ambiente universitário.

O Laboratório UIoT conta com recursos de apoio das Agências brasileiras de pesquisa, desenvolvimento e inovação CAPES (Projeto FORTE 23038.007604/2014-69), CNPq (Projeto INCT em Segurança Cibernética 465741/2014-2) e Fundação de Apoio à Pesquisa do Distrito Federal FAPDF (Projetos UIoT 0193.001366/2016 e SSDDC 0193.001365/2016), bem como do Gabinete de Segurança Institucional da Presidência da República (TED 002/2017) e do Laboratório LATITUDE/UnB (Projeto SDN 23106.099441/2016-43).

Resumo

Este documento apresenta os resultados do projeto final de graduação do curso de Engenharia de Redes da Universidade de Brasília, projeto este cujo principal objetivo é elaborar um protótipo de estação meteorológica inteligente e autônoma, de forma a utilizar os conceitos de Internet das Coisas para aplicar uma possível solução para os problemas de conectividade e mobilidade na transmissão de dados meteorológicos.

O presente documento contém a fundamentação teórica do trabalho, a realização e implementação do projeto, bem como os dados experimentais obtidos com o protótipo construído. São abordados os principais conceitos que envolvem Internet das Coisas (IoT), as dificuldades e problemas encontrados na definição e aplicação desses conceitos, especificamente na implementação e aplicabilidade na prototipação do dispositivo, bem como na captura e comunicação de dados da estação meteorológica proposta.

O projeto, cuja metodologia reúne aspectos de análise, de prototipação e de validação, mostra a viabilidade da utilização de IoT em aplicação de meteorologia inteligente e apresenta indicadores de interesse no que se refere à autonomia e informações coletadas, tratadas e disponibilizadas.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	OBJETIVOS.....	1
1.2	MOTIVAÇÃO E JUSTIFICATIVA.....	2
1.3	ORGANIZAÇÃO DO TRABALHO	2
2	FUNDAMENTAÇÃO TEÓRICA	3
2.1	INTERNET DAS COISAS.....	3
2.1.1	VISÕES	4
2.1.2	O PARADIGMA DE INTERNET DAS COISAS	6
2.1.3	SEGURANÇA E PRIVACIDADE EM IOT.....	7
2.2	CIDADES INTELIGENTES.....	8
2.2.1	ESTAÇÕES METEOROLÓGICAS	9
2.3	COMUNICAÇÃO	9
2.3.1	RADIOFREQUÊNCIA	10
2.3.2	REDES SEM FIO.....	12
2.3.3	REDES MÓVEIS	14
2.3.4	PROTOCOLOS DE LONGA DISTÂNCIA	17
2.3.5	COMPARAÇÃO ENTRE LORA E OUTRAS TECNOLOGIAS	23
2.3.6	CARACTERÍSTICAS IOT GATEWAY	25
2.3.7	CONECTIVIDADE.....	26
2.3.8	MIDDLEWARE	26
2.4	UBIQUIDADE	27
2.4.1	SERVIÇOS SENSÍVEIS AO CONTEXTO.....	27
2.4.2	DISTRIBUIÇÃO DE DADOS EM AMBIENTES UBÍQUOS MÓVEIS	28
2.5	MOBILIDADE	29
2.6	P2P	30
2.6.1	ROTEAMENTO	30
2.7	ARQUITETURA DE IOT	33
2.7.1	CAMADA SENSORIAL	33
2.7.2	CAMADA DE REDE.....	33
2.7.3	CAMADA DE SERVIÇO.....	34
2.7.4	INTERFACE.....	35

2.8	CONSIDERAÇÕES SOBRE OS CONCEITOS ABORDADOS.....	35
3	ESTRUTURA E FUNCIONAMENTO DA ESTAÇÃO METEOROLÓGICA PRO- POSTA.....	36
3.1	INTRODUÇÃO	36
3.2	ARQUITETURA DA PROPOSTA	37
3.2.1	GATEWAY	37
3.2.2	MIDDLEWARE UIOT.....	37
3.2.3	ARQUITETURA DE TRANSMISSÃO	38
3.3	ANÁLISE DE DISPOSITIVOS DE BAIXO CUSTO	39
3.3.1	SENSORES	39
3.3.2	ARDUINO	50
3.3.3	RASPBERRY.....	51
3.3.4	ESP 32	51
3.3.5	MÓDULO WIFI LORA	52
3.4	ALIMENTAÇÃO	56
4	RESULTADOS EXPERIMENTAIS	59
4.1	INTRODUÇÃO	59
4.2	SITE SURVEY COM A DISTRIBUIÇÃO DA POTÊNCIA DO SINAL EM VOLTA DA FACULDADE DE TECNOLOGIA.....	59
4.3	PROTÓTIPO DA ESTAÇÃO	62
4.4	DADOS DISPONIBILIZADOS	64
4.4.1	TAMANHO E FREQUÊNCIA DOS DADOS TRANSMITIDOS	66
4.5	CAPTURA, ENVIO E RECEBIMENTO DE DADOS.....	66
4.6	AUTONOMIA	69
5	CONCLUSÃO E TRABALHOS FUTUROS.....	70
5.1	CONCLUSÃO	70
5.2	TRABALHOS FUTUROS	71
	REFERÊNCIAS BIBLIOGRÁFICAS.....	72

LISTA DE FIGURAS

2.1	Visões de IoT [Atzori et al. 2010].....	5
2.2	Forma de onda	10
2.3	Faixas de frequências.....	11
2.4	Tipos de redes sem fio ([Coulouris et al. 2007]).....	12
2.5	Minimum requirements 5G [ITU-R 2016]	16
2.6	Diagrama de rede de uma rede LoRa (http://www.semtech.com/wireless-rf/internet-of-things/what-is-lora/)	18
2.7	Cisco LoRaWAN Gateway.....	21
2.8	Raspberry LoRaWAN Gateway.....	22
2.9	Comparação entre ZigBee, Bluetooth e Wi-Fi (https://www.netguru.co/codestories/the-zigbee-protocol)	24
2.10	Comparativo entre LoRa e outras tecnologias (http://www.vertical-m2m.com/)	24
2.11	Arquitetura sensível ao contexto.....	29
3.1	Dados disponibilizados	38
3.2	Arquitetura proposta	39
3.3	Sensor DHT11	40
3.4	Módulo DHT11	41
3.5	Especificações técnicas [AOSONG]	42
3.6	Sensor LDR.....	43
3.7	Módulo YL-38	44
3.8	Cicruito LM393.....	45
3.9	Sensor de chuva YL-83	46
3.10	Sensor MQ-7	49
3.11	Sensor MQ-9	50
3.12	Kit de desenvolvimento Heltec LoRa 32	53
3.13	Especificações Kit de desenvolvimento Heltec LoRa 32	54
4.1	Mapa com a distância de transmissão de cada ponto	60
4.2	Antena utilizada na recepção dos dados	61
4.3	Protótipo da Estação	62
4.4	Circuito da Estação	63
4.5	Circuito de alimentação.....	63
4.6	Diagrama do Protótipo	64

4.7	Gateway LoRa mostrando MAC e CHIPSET da Estação 1	65
4.8	Dados recebidos pelo Middleware	65

LISTA DE TABELAS

3.1	Consumo estimado da Estação	56
3.2	Painel solar	57
3.3	Especificações necessárias da bateria.....	57

LISTA DE CÓDIGOS FONTE

3.1	Código do ATtiny85	57
4.1	Código da Estação LoRa	66
4.2	Código do Gateway LoRa	67

Capítulo 1

Introdução

Sabe-se da necessidade de que as pessoas se planejem para as atividades do dia, de forma que, ao saberem como estarão os ambientes que irão frequentar, se torna mais fácil a prevenção contra imprevistos, como chuva, por exemplo. Muitas vezes, o ambiente em que um indivíduo está localizado tem condições temporais completamente diferentes das do ambiente em que o indivíduo irá visitar. Além desse aspecto que tem vínculo com o conforto das pessoas, a meteorologia tem também importante papel na economia e na segurança da sociedade. A previsão meteorológica cumpre papel fundamental na produção agrícola que, em um país como o Brasil, é o principal motor econômico. A meteorologia é ainda um dos componentes principais de controle para o bom funcionamento da infraestrutura de transportes de insumos e de pessoas, no campo e nas cidades, além de contribuir para a segurança da sociedade, haja visto os diversos e conhecidos impactos da mudança climática.

Tendo em vista essa necessidade de disponibilizar informações sobre o clima no ambiente, este projeto visa construir uma estação meteorológica inteligente utilizando os conceitos e aplicações que permeiam o ambiente de Internet das Coisas.

1.1 Objetivos

Essa estação deverá ser um sistema completo de hardware, software e de rede de comunicação, capaz de capturar informações como temperatura, umidade, luminosidade do ambiente, precipitação de chuva, bem como medir a qualidade do ar de acordo com o nível de Monóxido de Carbono. Além da captura de dados, a estação deverá ser capaz de transmiti-los para um *middleware* alocado no laboratório UIoT (UnB Internet of Things), localizado na Faculdade de Tecnologia da Universidade de Brasília. A partir desses dados, deverá ser realizada uma inferência dos mesmos, de forma a disponibilizar essas informações à comunidade.

Para que essa transmissão seja feita de forma precisa, sincronizada e efetiva, propõe-se neste projeto uma arquitetura que permita a transmissão dos dados, com o propósito de

disponibilizá-los para a comunidade de forma clara, correta e simples.

1.2 Motivação e Justificativa

Dados meteorológicos são de enorme importância em vários aspectos da sociedade, sendo responsáveis por dar indicadores que permitam previsão e planejamento no que se refere à segurança, economia, transportes, qualidade de vida, desenvolvimento social, entre outros. De acordo com [WMO], esses dados são essenciais na prevenção de desastres naturais, como erupções, furacões e tsunamis.

Outro aspecto muito importante para a sociedade de hoje em dia é o conforto, de forma que cada vez mais têm sido desenvolvidas tecnologias que facilitem a vida do ser humano. Assim, dados do ambiente são bastante utilizados para que a sociedade tenha uma maior qualidade de vida.

Por fim, existe também a problemática do microclima [Romero 2011], muito forte em Brasília, o que torna o clima nas cidades bastante imprevisível. Assim, a proposta vem com o intuito de permitir a disponibilidade dos dados do ambiente onde ela estiver implementada.

1.3 Organização do Trabalho

O presente documento contém a fundamentação teórica do trabalho (Capítulo 2), a realização e implementação do projeto (Capítulo 3), bem como os dados experimentais obtidos com o protótipo realizado (Capítulo 4). São abordados os principais conceitos que envolvem Internet das Coisas (IoT), bem como dificuldades e problemas encontrados na definição e aplicação desses conceitos, especificamente na implementação e aplicabilidade na prototipação do dispositivo, e também na captura e comunicação de dados da estação meteorológica proposta.

O projeto, cuja metodologia reúne aspectos de análise, de prototipação e de validação, mostra a viabilidade da utilização de IoT em aplicação de meteorologia inteligente e apresenta indicadores de interesse no que se refere à autonomia e informações coletadas, tratadas e transmitidas.

Capítulo 2

Fundamentação Teórica

Nesta seção é apresentada a fundamentação teórica do trabalho, ou seja, as pesquisas bibliográficas encontradas e estudadas que serviram como base para a implementação da proposta. Os conceitos aqui apresentados são necessários para uma compreensão sobre o que é Internet das Coisas, seus paradigmas, suas aplicabilidades e as dificuldades em sua implementação, bem como para descrever e entender melhor a proposta do projeto. Também será falado sobre os mais diversos e modernos protocolos e meios de transmissão de dados, tendo em vista que são assuntos de extrema importância para a implementação da proposta.

2.1 Internet das coisas

O projeto proposto é baseado e ambientado no conceito de IoT, o qual tem sido cada vez mais utilizado na sociedade e, muito provavelmente, será altamente difundido em um futuro próximo. Esse conceito de IoT, bem como seus paradigmas e abordagens, são utilizados para estruturar e definir a proposta. É visível que a internet tem sido cada vez mais utilizada em dispositivos que não se imaginava que pudessem ter algum tipo de conexão décadas atrás [Atzori et al. 2010], como, por exemplo, relógios (smartwatches), celulares, televisores, utensílios eletrodomésticos e dispositivos que ainda não existiam, como drones e dispositivos capazes de monitorar e enviar dados de um ambiente. Além disso, tem-se acompanhado o desenvolvimento de dispositivos móveis das mais variadas formas. Esse conceito de permitir a conexão entre objetos com diferentes funcionalidades é conhecido como Internet das Coisas. Em suma, Internet das Coisas é uma forma de permitir conexão a todo e qualquer objeto que existe ou possa existir [Gubbi et al. 2013].

Existem muitas abordagens sobre o conceito de IoT, porém, num primeiro momento, ela pode ser dividida em duas partes principais: Internet e Coisas. A Internet, que é formada por computadores interconectados baseados em um protocolo de comunicação padrão, como o *Transmission Control Protocol* (TCP/IP) [Atzori et al. 2010], é muito difícil de ser identificada como um objeto. Além disso, o mundo está cada vez mais rodeado de objetos inteligentes e conectáveis, de forma que a convergência entre esses dispositivos, redes de sensores, RFID, e entre a própria Internet, dão espaço para o desenvolvimento do que se conhece hoje como IoT [Zhu et al. 2010].

2.1.1 Visões

De acordo com [Jun-Wei et al. 2011], o conceito de IoT pode ser dividido em basicamente três visões diferentes: orientação à coisas, orientação à internet e orientação à semântica. O termo "Internet das Coisas", por si só, já faz alusão entre uma perspectiva de orientação à internet ou de orientação à coisas, dependendo das finalidades e interesses específicos no uso da tecnologia. De acordo com [RFID INFSO G.2 2008], ao unir esses termos, cria-se uma "rede mundial de dispositivos interconectados, unicamente endereçados, e baseados em protocolos de comunicação padronizados".

Para [Jun-Wei et al. 2011], o fato de existir um objeto com endereço único, bem como o armazenamento de uma informação transmitida, trazem a necessidade de uma terceira perspectiva de IoT, que é a "orientação à semântica". A Figura 2.1, retirada de [Atzori et al. 2010], mostra a convergência entre essas três visões. Nela, é possível perceber a integração de diferentes tecnologias e conceitos para formar a Internet das Coisas.

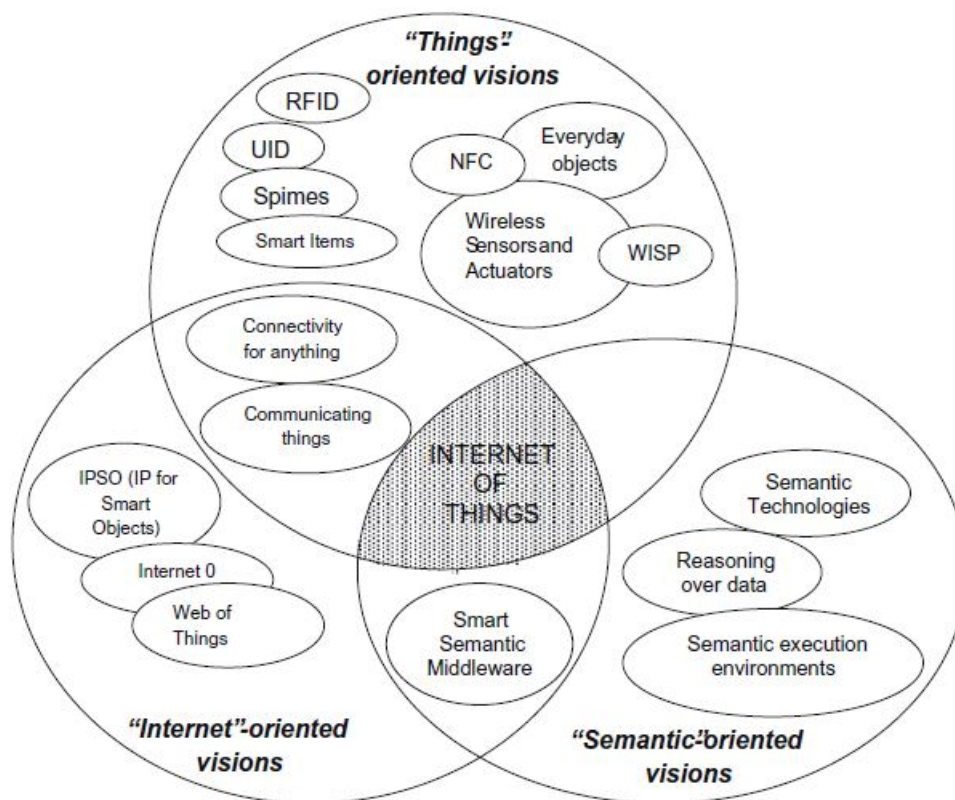


Figura 2.1: Visões de IoT [Atzori et al. 2010]

2.1.1.1 Orientação à Coisas

A orientação à coisas, resumidamente, se baseia no fato de que qualquer objeto pode ser rastreado utilizando sensores e RFID. Em suma, é o fato de identificar qualquer um desses objetos através de um Código de Produto Eletrônico (Electronic Product Code, em inglês), definido por [EPCglobal 2009]. Assim, o EPC permite o completo uso de RFID na rede mundial moderna, criando um padrão global para a EPCglobal Network. Esse padrão é necessário para permitir a visibilidade, rastreabilidade, localização e status de um objeto, sendo, componente chave no completo desenvolvimento da visão sobre Internet das Coisas.

Contudo, IoT não pode ser definida apenas em termos de sistemas EPC, onde os únicos objetos são RFID. O mesmo também vale para uma arquitetura alternativa, chamada de UUID (Universal Unique Identifier) [Sakamura 2006], que é um código de 128 bits, onde a principal ideia, baseada em middleware [Qilin and Mintian 2010], é o desenvolvimento de soluções para visibilidade global de objetos.

2.1.1.2 Orientação à Internet

A orientação à internet configura a necessidade de fazer objetos inteligentes e conectáveis, de forma que eles tenham características que funcionem através do protocolo IP, que atualmente é o mais utilizado. O objeto baseado em sensores poderá ser unicamente identificado e continuamente monitorado através de uma rede [Atzori et al. 2010].

2.1.1.3 Orientação à Semântica

A visão semântica entra no fato de que tecnologias semânticas serão necessárias para catalogar, processar e analisar os dados, ou seja, separar e interpretar os dados que tenham real utilidade para cada tipo de aplicação. Muito pode ser encontrado sobre essa perspectiva em [I. Toma and Hench 2009], [Katasonov et al. 2008], [Wahlster 2008] e [Vázquez 2009].

A principal ideia é haver uma preparação para o fato de que a quantidade de dispositivos que farão parte da internet será extremamente grande. Dessa forma, problemas como armazenar, interconectar, procurar (discovery), e organizar a informação gerada pela IoT serão bastante desafiadores. Nesse contexto, tecnologias semânticas podem modelar soluções para a descrição dos objetos e suas funcionalidades [I. Toma and Hench 2009].

2.1.2 O paradigma de Internet das Coisas

[Atzori et al. 2010] mostra que o paradigma da Internet das Coisas também pode ser analisado como o conjunto de três conceitos diferentes: middleware, sensores e o conhecimento base para reunir as três visões de Internet das Coisas. Individualmente, as visões não podem completar o conceito de conectividade, de forma que a interseção entre as três visões será responsável por conectar objetos à rede. Para qualquer rede, as coisas participarão ativamente em negócios, informações e processos sociais.

Logo, é necessário dar atenção às requisições do usuário aos sensores, habilitando a IoT para ambientes inteligentes com a interconexão de sensores capazes de processar informações. Isso só é possível com redes de sensores sem fio de larga escala (Wireless Sensor Network – WSN), processamento de dados e computação em nuvem. Os processos que permitem o monitoramento dessa Rede de Sensores Wireless são:

- WSN hardware: basicamente, uma WSN contém interfaces e sensores, unidades de processamento, unidades de transmissão e recepção, e fontes de energia. Como visto em [Atzori et al. 2010], nós de sensoriamento podem se comunicar em múltiplas frequências;
- WSN Communication Stack (WSNCS): os nós são desenvolvidos em ADHOC, de forma que a topologia da rede será um fator importante para a comunicação através dos nós WSN. De acordo com [Kotis and Katasonov 2012], terá um nó central capaz de interagir com o mundo conectado pela Internet que servirá como um gateway entre a subrede WSN e

a Internet;

- Middleware: o *middleware* será associado com a infraestrutura de rede e o conceito de SOA (Arquitetura Orientada a Serviço) de forma a acessar sensores heterogêneos, como descrito em [M. Yun 2010].

Tecnologias de Redes de Sensores Wireless avançam no domínio do hardware provendo circuitos e comunicação sem fio robustos, bem como dispositivos cada vez mais baratos no uso de aplicações sensíveis. Isso possibilita utilizar sensores em dispositivos de comunicação sem fio em diversos tipos de ambientes, como descrito por [Atzori et al. 2010]. Os dados dos sensores são coletados e enviados para módulos de processamento centralizados, distribuídos ou até mesmo híbridos. Porém, como será visto mais à frente, alguns desafios são encontrados.

2.1.3 Segurança e Privacidade em IoT

Redes IoT ainda são bastante vulneráveis contra ataques de segurança. Primeiramente, dispositivos IoT passam a maior parte do tempo sem serem vigiados, possibilitando ataques físicos. Em segundo lugar, a maioria das conexões são sem fio, o que facilita a espionagem. Por último, e mais importante, dispositivos IoT costumam ter baixa capacidade energética e computacional, o que pode dificultar a implementação de mecanismos de segurança complexos.

Outros problemas em relação à segurança são autenticação e integridade de dados. Mecanismos de autenticação costumam requerer infraestrutura e servidores que permitam trocar mensagens de forma segura com outros nós. Contudo, assim como nos problemas com conexões TCP, etiquetas RFID não podem trocar muitas mensagens de autenticação com servidores, e essa restrição também se aplica a redes de sensores sem fio. Para essas redes, várias soluções já foram propostas, como visto em [Eschenauer and Gligor 2002]. Nos últimos anos, surgiram algumas soluções propostas para sistemas RFID, porém, como descrito em [Jules 2006], ainda existem vários problemas a serem resolvidos.

O problema de integridade de dados tem sido estudado em todos os tipos de sistemas de computação e comunicação tradicionais, mas existem apenas alguns estudos preliminares quando se trata de redes de sensores, como apresentado em [R. Acharya 2008]. De uma forma ou de outra, soluções de segurança acabam utilizando algum tipo de criptografia, o que demanda energia, processamento e até mesmo largura de banda, elementos que costumam ser relativamente limitados em ambientes IoT.

Uma outra questão relativa à segurança em ambientes IoT é a privacidade. Em termos de Internet das Coisas, a proteção da privacidade pode significar uma barreira para a difusão desse tipo de tecnologia, já que a forma como dados serão coletados e fornecidos é completamente diferente do que se está acostumado, sem contar que muitos dados pessoais podem ser expostos. Proteção da privacidade significa que os usuários podem definir quais dados

podem ser coletados, por quem, quando e por qual motivo, ou seja, os dados pessoais coletados só devem ser utilizados por serviços autorizados e, ainda, só devem ser armazenados enquanto estiverem em uso. Assim, de forma a assegurar que os dados pessoais coletados sejam utilizados apenas por serviços autorizados, [Lioudakis et al. 2007] propõe uma solução baseada em um sistema chamado "privacy broker". A ideia principal é de que o proxy interaja com o usuário de um lado e com os serviços do outro, com o intuito de garantir que o provedor do serviço tenha acesso restrito às informações do usuário, ou seja, este pode configurar as informações necessárias no proxy. Com relação à redes de sensores e sistemas RFID, o proxy vai funcionar entre eles e os serviços. Porém, neste caso, o usuário não conseguirá configurar e controlar as políticas utilizadas pelo "privacy broker".

2.2 Cidades Inteligentes

O conceito de Cidades Inteligentes (*Smart Cities*) surgiu com o intuito de ajudar a resolver problemas socioeconômicos e aumentar a qualidade de vida da sociedade. Para [Schaffers et al. 2011], *Smart Cities* são ambientes de inovação abertos e orientados para o usuário, de forma a validar serviços habilitados para a Internet do Futuro. O autor ainda cita que maiores preocupações no que se refere à sustentabilidade, bem como o desenvolvimento de novas tecnologias (smartphones, web semântica, computação em nuvem e IoT), têm fomentado um maior interesse no conceito de Cidades Inteligentes.

Hoje em dia, podemos encontrar vários exemplos de cidades inteligentes em desenvolvimento. A cidade de Masdar, em Abu Dhabi (Emirados Árabes), também conhecida como "oásis do futuro", pretende se tornar o primeiro centro tecnológico renovável e sustentável do mundo [Kingsley]. Helsinque e Intelligent Thessaloniki, na Grécia, são tidos como exemplos de cidades que incentivam o desenvolvimento de novas aplicações móveis utilizando dados abertos e Tecnologia da Informação para aumentar a competitividade e sustentabilidade [Komninos et al. 2013]. Na Europa, pode-se citar também Barcelona e Amsterdam. A primeira é bastante conhecida pelo seu modelo Smart City e, em Novembro de 2014, sediou a quarta "Smart City Expo World Congress" (<http://www.smartcityexpo.com/>), enquanto a iniciativa Amsterdam Smart City é um grande exemplo de como renovar uma cidade para melhorar as condições de vida e econômicas, bem como reduzir emissões de carbono [Hollands] [Kirby]. No Brasil, tem sido desenvolvida a Smart City Laguna, localizada em São Gonçalo, Ceará (<https://smartcitylaguna.com.br/>),

Um importante ponto de partida desta lista de cidades é a natureza de suas abordagens. O objetivo não é apenas utilizar tecnologias, mas também fomentar a prática do desenvolvimento econômico. [Mohanty et al.] cita que, para o desenvolvimento de *Smart Cities*, também é necessário utilizar conceitos importantes como, *Smart Agriculture*, *Smart Home*, *Smart Grid/Smart Energy*, *Smart Government*, *Smart Health Care* e *Smart Transportation*.

O autor ainda apresenta que esses conceitos têm IoT como principal base.

2.2.1 Estações Meteorológicas

Algo muito comum em ambientes inteligentes é a captura de dados meteorológicos. Variáveis climáticas urbanas locais como calor, umidade, falta de luz do dia, acesso solar e ventilação são de grande interesse para planejadores urbanos e governos [Ng].

Dados Meteorológicos são coletados rotineiramente por estações em todo o mundo [WMO] para monitorar condições climáticas em escala mundial e local. Dados de estações meteorológicas com longos períodos de observação são essenciais para que planejadores e arquitetos entendam as condições climáticas predominantes e críticas do ambiente. Além disso, dados que descrevem as condições do clima são frequentemente utilizados em estudos relacionados ao planejamento urbano detalhado, bem como de avaliações de impacto ambiental [Ng].

Com isso, existem vários exemplos de capturas de dados meteorológicos. A Agência Espacial Brasileira entregou, no primeiro semestre de 2017, 11 estações meteorológicas didáticas para escolas do Distrito Federal (<http://portal-antigo.aeb.gov.br/aeb-entrega-estacoes-meteorologicas-a-escolas-publicas-e-particulares-do-df/>). Esses equipamentos fazem parte do projeto Rede de Estações Meteorológicas Didáticas (Remade), foram desenvolvidos com material alternativo e possuem um sistema automático que permite funcionamento durante todo o dia. Os dados podem ser visualizados em (<http://www.climaescola.com.br/remade/mapa.php>).

Existe uma iniciativa chamada Open Weather Map (<https://openweathermap.org/>) que já consegue captar dados Meteorológicos de várias cidades em vários países, sendo capaz de disponibilizar até dois bilhões de previsões por dia.

2.3 Comunicação

A visão de Internet das Coisas traz o questionamento de como todos esses objetos e dispositivos estarão conectados à internet. Isso mostra a necessidade de gateways na camada de aplicação, tanto em software quanto em hardware, de forma a prover aplicações específicas para conectividade em IoT [Chen et al. 2011]. O problema de IoT Gateway existe, em grande parte, porque, hoje em dia, os gateways confundem conectividade de rede, processamento em rede e interface do usuário [Zhu et al. 2010]. Com isso, o desenvolvimento de um gateway IoT poderia explorar de forma mais precisa as aplicações de conectividade [Zhu et al. 2010]. Para a proposta do projeto, foi identificada a necessidade de um gateway de comunicação entre a estação meteorológica e o laboratório do UIoT. Esse gateway é utilizado para receber diversos tipos de dados dos mais variados dispositivos.

Atualmente, as redes de comunicação móveis tradicionais são principalmente usadas na troca de informações entre usuários, enquanto uma WSN (Wireless Sensor Network) permite comunicação de curta distância entre dispositivos, construindo redes sem fio em ad-hoc [Xu et al. 2014]. Porém, é difícil conectar redes de sensores sem fio, redes móveis e a própria Internet entre si, visto que é necessária uma padronização uniforme nos protocolos de comunicação, sem contar que tecnologias sensíveis e Redes de Sensores Wireless não podem transmitir em longa distância, devido às limitações dos protocolos de transmissão de WSN's [Xu et al. 2014].

Contudo, com o crescimento da Internet das Coisas, um novo tipo de conceito, chamado IoT Gateway, foi estabelecido com o objetivo de unir a heterogeneidade de dispositivos, redes móveis e a Internet, de forma a facilitar o gerenciamento de WSN's [Chen et al. 2011], e unir redes de comunicação com sensores de rede, tornando mais fáceis a comunicação e gerenciamento de sensores de rede [Xu et al. 2014]. Entretanto, o principal problema em implementar um IoT Gateway está em endereçar os diferentes dispositivos heterogêneos e integrar a diversidade de protocolos em WSN com as redes de telecomunicações tradicionais [Sakamura 2006].

2.3.1 Radiofrequência

Radiofrequência são correntes alternadas de alta frequência que, ao chegarem nas antenas, são convertidas em ondas eletromagnéticas. O receptor recebe essas ondas e as converte em sinais elétricos, de forma que os rádios possam compreender a informação transmitida.

Uma onda eletromagnética possui as seguintes características, que podem ser visualizadas na figura 2.2:

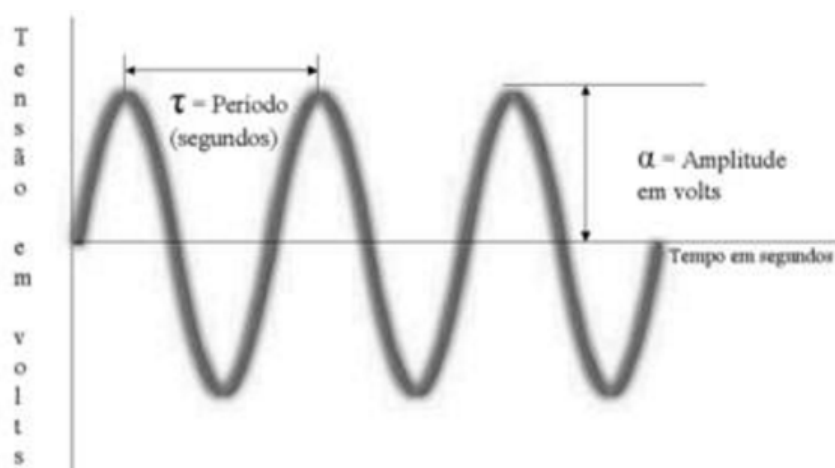


Figura 2.2: Forma de onda

1. Amplitude: que é a tensão máxima, em Volts, que a onda atinge;
2. Período: é o tempo, em segundos, que uma onda completa um ciclo;

3. Comprimento de onda: espaço em metros que a onda percorre entre dois sinais a partir do qual ela se repetiu. Ou seja, o espaço que essa onda percorre em um período;

4. Frequência: é o número de ciclos que a onda percorre no tempo, no caso em um segundo. A frequência é o inverso do período em segundos, e é dada em Hertz (HZ);

Para que não ocorram sobreposições na utilização de ondas de rádio, foram disponibilizadas faixas de frequência. Como pode ser visto na Figura 2.3, as faixas em que estão alocadas as WLAN's são a Ultra Alta cuja frequência destinada a essa faixa é 300MHz – 3GHz; e a faixa Super Alta que varia de 3GHz – 30GHz.

FAIXA DE FREQUÊNCIA (Hz)	DENOMINAÇÃO TÉCNICA	DENOMINAÇÃO POPULAR	EXEMPLOS DE UTILIZAÇÃO
300 a 3000	E.L.F (<i>Extremely Low Frequency</i>)	Ondas Extremamente Longas	Comunicações para submarinos, escavações de minas etc.
3k a 30k	V.L.F (<i>Very Low Frequency</i>)	Ondas Muito Longas	Comunicações para submarinos, escavações de minas etc.
30k a 300k	L.F (<i>Low Frequency</i>)	Ondas Longas	Auxílio a navegação aérea, serviços marítimos e radiodifusão local.
300k a 3M	M.F (<i>Médium Frequency</i>)	Ondas médias	Auxílio a navegação aérea, serviços marítimos e radiodifusão local.
3M a 30M	H.F (<i>High Frequency</i>)	Ondas tropicais/ ondas curtas	Radiodifusão local e distante, sistemas marítimos (estações costeiras)
30M a 300M	V.H.F (<i>Very High Frequency</i>)	Micro-ondas	Transmissão de TV, sistemas comerciais e particulares de comunicação, serviços de segurança pública (Polícia, Bombeiros etc.)
300M a 2G	U.H.F (<i>Ultra High Frequency</i>)	Micro-ondas	Transmissão de TV, sistemas comerciais e particulares de comunicação, serviços de segurança pública (Polícia, Bombeiros etc.)
2G a 3G	U.H.F (<i>Ultra High Frequency</i>)	Micro-ondas	Comunicação publica a longa distância: sistemas interurbanos e internacionais em radiovisibilidade, tropodifusão e satélite.
3G a 30G	S.H.F (<i>Super High Frequency</i>)	Micro-ondas	Comunicação publica a longa distância: sistemas interurbanos e internacionais em radiovisibilidade, tropodifusão e satélite.
30G a 300G	E.H.F (<i>Extremely High Frequency</i>)	Micro-ondas	Comunicação publica a longa distância: sistemas interurbanos e internacionais em radiovisibilidade, tropodifusão e satélite.

Figura 2.3: Faixas de frequências

2.3.2 Redes sem fio

O principal objetivo das redes sem fio é permitir o desenvolvimento de comunicações móveis, possibilitando ao usuário comunicação com outras infraestruturas de comunicação. [Coulouris et al. 2007] mostra, de acordo com a Figura 2.4, que existem basicamente quatro tipos diferentes de redes sem fio e, em cada uma delas, a tecnologia mais utilizada.

Rede	Exemplo	Alcance	Largura de banda (Mbps)	Latência (ms)
WPAN	Bluetooth (IEEE 802.15.1)	10-30m	0,5-2	5-20
WLAN	Wifi (IEEE 802.11)	0,15-1,5 Km	2-54	5-20
WMAN	WiMax (IEEE 802.16)	5-50 Km	1,5-20	5-20
WWAN	Redes telefônicas GSM, 3G	Mundial	0,010-2	100-500

Figura 2.4: Tipos de redes sem fio ([Coulouris et al. 2007])

WPAN (Wireless Personal Area Network): são redes pessoais sem fio responsáveis por fornecer conectividade de curta distância para dispositivos móveis, onde estes estão fisicamente próximos. São bastante utilizadas para interligar periféricos de computadores ou celulares (teclados, mouses, impressoras, fones de ouvido), telefones móveis entre si, entre outros dispositivos, geralmente pequenos e com baixo consumo e processamento. Para esse tipo de rede, as tecnologias mais utilizadas são o Bluetooth e o Infravermelho.

WLAN (Wireless Local Area Networks): redes locais sem fio, assim como as LAN's, são redes que fornecem conectividade para dispositivos móveis dentro de uma área (casa, prédio) e prover acesso à Internet ou outra rede fixa (LAN, WAN). Possui variantes de padrões, oferecendo largura de banda de 10 e 100 Mbps com abrangência de até 1,5 quilômetros.

WMAN (Wireless Metropolitan Area Network): são redes metropolitanas sem fio. Costumam possuir maior abrangência que as WLAN, por exemplo. Também conhecidas como Anel Local Rádio (BLR), fornecem velocidades entre 1 e 10 Mbps e um alcance entre 4 a 10 quilômetros.

WWAN (Wireless Wide Area Networks): são redes de longa distância sem fio amplamente utilizadas para telefonia móvel. Elas têm acesso com ampla cobertura, seja por antenas ou sistemas de satélite. É importante salientar que estas redes oferecem taxas de transmissão de dados relativamente baixas quando comparadas com as demais redes.

2.3.2.1 Wi-Fi

A Wi-Fi Alliance [WiFiAlliance], dona da patente do protocolo Wi-Fi, que significa "*wireless fidelity in*", define essa tecnologia como "qualquer produto/serviço/dispositivo WLAN (Wireless Local Access Network) que esteja baseado no padrão IEEE 802.11" [IEEE b]. Wi-Fi utiliza radiofrequência, uma frequência do espectro eletromagnético associada à propagação de ondas eletromagnéticas [Sandhu et al. 2013]. Essa tecnologia funciona de forma bastante parecida com outras tecnologias de rádio, como walkie-talkies e telefones celulares. Ou seja, ambos podem transmitir e receber ondas de rádio. Porém, o Wi-Fi trabalha nas frequências 2.4 GHz e 5 GHz, que são consideradas altas, se comparadas com as de outras tecnologias. [Sandhu et al. 2013] ainda cita que quanto maior a frequência, maior a quantidade de dados que podem ser transmitidos.

Dentre os padrões contidos no IEEE 802.11, pode-se citar o 802.11b e o 802.11a/g, que atingem, respectivamente, taxas de dados de 11 e 54 Mbps [802.11 1999] [Xiao and Rosdahl 2002] e o 802.11n, com taxas ainda maiores [Kadir et al. 2016].

Com o intuito de suportar, de forma eficiente, aplicações IoT e M2M, foi adicionado o 802.11ah [IEEE 2015], também conhecido como Wi-Fi HaLow. Muitas funcionalidades foram adicionadas à camada MAC de forma a prover escalabilidade para várias estações, aumentar o alcance de operação e, ao mesmo tempo, reduzir o consumo de energia, comparado aos outros padrões 802.11 [Akeela and Elziq 2017]. De acordo com [Adame et al. 2014], o 802.11ah possui funcionalidades únicas, tais como:

- Até 8,191 dispositivos associados a um único ponto de acesso (AP – Access Point)
- Frequências portadoras abaixo de 1 GHz
- Alcance de transmissão de até 1 Km em áreas externas
- Taxas de dados de no mínimo 100 Kbps

Outra característica bastante importante do Wi-Fi HaLow é o baixo consumo de energia. A redução do consumo de energia foi conseguida adotando protocolos da camada MAC que utilizam *frames* de formatos menores [IEEE 2015].

Para a nossa proposta, a utilização dessa tecnologia é muito atraente. Porém, não é muito popular e ainda existe dificuldade de encontrar dispositivos e pontos de acesso que trabalhem com esse padrão no Brasil.

2.3.3 Redes Móveis

Sabe-se que uma grande variedade de tecnologias de redes sem fio vem sendo padronizadas e comercializadas, porém, devido às várias limitações de cobertura e largura de banda de cada uma, não se pode escolher uma tecnologia como a melhor. Além disso, para sistemas ubíquos de alta performance, é imprescindível a interconexão entre os diferentes e heterogêneos sistemas de redes sem fio [Shin et al. 2006].

Redes celulares como CDMA [3GPP2 Technical Specifications v1.0 2002] e UMTS [3GPP2 Technical Specifications v6.4.0 2004] cobrem um grande espectro de redes sem fio pessoais (WPANs). O 3G, por exemplo, que ainda possui a maior cobertura no Brasil, em relação ao 4G, provê uma grande cobertura com largura de banda limitada (2Mbps) [Shin et al. 2006], enquanto o padrão 802.11 consegue entregar velocidades de 54Mbps, mesmo que com uma cobertura bem menor [Buddhikot et al. 2003].

Nas últimas décadas, redes móveis têm experienciado enormes mudanças. De acordo com [Vora], quando se fala da Geração de redes sem fio (G), refere-se às evoluções de parâmetros como velocidade, tecnologia, frequência, capacidade de dados, latência, entre outros. Cada uma dessas gerações pode ser dividida em padrões, técnicas, funcionalidades e tecnologias diferentes das gerações anteriores. A primeira geração (1G), por exemplo, era uma rede de comunicação sem fio completamente analógica e utilizada apenas para chamadas de voz. Já a segunda geração (2G) é uma tecnologia digital e permite o envio de mensagens de texto, através do protocolo SMS (Short Message Service). A terceira geração (3G), sem dúvidas, foi a que trouxe as mudanças mais significativas, permitindo maiores taxas de transmissão de dados e suporte ao envio de multimídias como imagens e vídeos. A quarta geração (4G), uma tecnologia de certa forma integrada ao 3G, foi uma evolução na tecnologia móvel e superou as limitações do seu antecessor, trazendo também aumento na largura de banda e redução de custos [Gawas 2015]. Já a quinta geração, bastante conhecida como 5G, vem com o intuito de revolucionar a experiência do usuário no que diz respeito ao nível de conectividade e cobertura.

[Gawas 2015] cita que o 1G, desenvolvido nos anos 80, era baseado em uma tecnologia conhecida como AMPS, que, em inglês, significa Advanced Mobile Phone System. Essa tecnologia era modulada na frequência e utilizava FDMA (Frequency Division Multiple Access) [Introduction to single carrier FDMA 2007], suportando velocidades de até 2.4Kbps. O 2G, introduzido no final dos anos 80, utilizava tecnologia digital e esquemas de modulação como TDMA (Time Division Multiple Acces) e CDMA (Code Division Multiple Access), onde o primeiro permite a divisão dos sinais em *slots* de tempo e o segundo permite com que cada usuário tenha um código especial pra se comunicar via um canal multiplexado. Dentre as tecnologias TDMA utilizadas no 2G, pode-se citar o GSM (Global System for Mobile Communication), que é o mais utilizado no padrão 2G. [Gawas 2015] diz ainda que o GSM foi a primeira tecnologia a permitir *roaming* internacional. Algumas melhorias feitas no GSM introduziram o 2.5G, onde foram alcançadas taxas de 144Kbps. Também foram introduzidas

tecnologias como GPRS (General Packet Radio Service) e EDGE (Enhanced Data rates in GSM Environment), onde a primeira provia protocolos de trocas de pacotes, rápida conexão com o provedor de Internet e a possibilidade de cobrar o usuário de acordo com a quantidade de dados trafegados, não apenas pelo tempo de conexão. Além disso, o GPRS provê transmissão flexível de dados e conexão contínua com a rede, se tornando assim um importante passo para a evolução para o 3G [Gawas 2015].

Com a introdução do 3G no Japão, no ano de 2001, as taxas de dados subiram de 144 para 348Kbps, chegando a até 2Mbps em áreas de maior cobertura [Gawas 2015]. Com o 3G, foi possível o acesso a serviços como TV, acesso web, e-mail, videoconferências e até acesso a mapas, com uma largura de banda entre 15 e 20MHz [3GPP TR 45.820 2015].

[IEEE 2006] cita que, dentre os principais objetivos do 4G, podem ser elencados os seguintes: prover aumento significativo na taxas de dados, quando comparado ao 3G, em números teóricos de 100Mbps e 1Gps para download, e metade desses valores para upload de dados; reduzir a latência, alcançando performance parecida com a de protocolos como TCP, bem como reduzir atrasos; e aumentar a capacidade do sistema, em pelo menos o triplo, com relação ao 3G.

Com a crescente demanda de aplicações em IoT, [Shafi et al. 2017] mostra que os requisitos da quinta geração de telefonia móvel (5G) vêm com o intuito de já atender essas necessidades. Trabalhando no espectro de micro-ondas (3.3-4.2GHz) e utilizando largas bandas disponíveis em *millimeter wave*, visa-se prover escalabilidade e flexibilidade. Diferentemente do 4G, o 5G deve ser flexível e adaptável de forma a suportar diversos casos de uso, inclusive em aplicações IoT. Uma das demandas, de acordo com [Wien 2015], é que, com o aumento da resolução de vídeo para 4K, será necessária uma taxa de dados de no mínimo 15.4Mbps por usuário. O tempo de visualização de vídeos também tem aumentado, se tornando cada vez mais comum a utilização de *streaming* de vídeo. Com a crescente demanda por conteúdo, estima-se que o crescimento de tráfego anual aumente para 291,8 exabytes em 2018 [Rappaport et al. 2015].

Contudo, o uso de 5G (IMT2020) não é limitado apenas à banda larga móvel. [Shafi et al. 2017] diz que o 5G deve suportar a diversa variedade de uso de dados em três categorias: rede móvel melhorada (Enhanced Mobile Broadband – eMBB), comunicações confiáveis e de baixa latência (Ultra-reliable and low latency communications – URLLC) e comunicações massivas para máquinas (Massive machine type communications – mMTC).

De acordo com [Shafi et al. 2017], os requisitos mínimos para o 5G são os descritos na Figura 2.5

KPI	Key Use Case	Values
Peak Data Rate	eMBB	DL: 20 Gbps, UL: 10 Gbps
Peak Spectral Efficiency	eMBB	DL: 30 bps/Hz, UL: 15 bps/Hz
User Experienced Data Rate	eMBB	DL: 100 Mbps, UL: 50 Mbps (Dense Urban)
5% User Spectral Efficiency	eMBB	DL: 0.3 bps/Hz, UL: 0.21 bps/Hz (Indoor Hotspot); DL: 0.225 bps/Hz, UL: 0.15 bps/Hz (Dense Urban); DL: 0.12 bps/Hz, UL: 0.045 bps/Hz (Rural)
Average Spectral Efficiency	eMBB	DL: 9 bps/Hz/TRxP, UL: 6.75 bps/Hz/TRxP (Indoor Hotspot); DL: 7.8 bps/Hz/TRxP, UL: 5.4 bps/Hz/TRxP (Dense Urban); DL: 3.3 bps/Hz/TRxP, UL: 1.6 bps/Hz/TRxP (Rural)
Area Traffic Capacity	eMBB	DL: 10 Mbps/m ² (Indoor Hotspot)
User Plane Latency	eMBB, URLLC	4 ms for eMBB and 1 ms for URLLC
Control Plane Latency	eMBB, URLLC	20 ms for eMBB and URLLC
Connection Density	mMTC	1,000,000 devices/km ²
Energy Efficiency	eMBB	Capability to support high sleep ratio and long sleep duration
Reliability	URLLC	$1 - 10^{-5}$ success probability of transmitting a layer 2 protocol data unit of 32 bytes within 1 ms in channel quality of coverage edge
Mobility	eMBB	Up to 500 km/h
Mobility Interruption Time	eMBB, URLLC	0 ms
Bandwidth	eMBB	At least 100 MHz; Up to 1 GHz for operation in higher frequency bands (e.g., above 6 GHz)

Figura 2.5: Minimum requirements 5G [ITU-R 2016]

2.3.3.1 NarrowBand - IoT

Na versão 13 do padrão 3GPP, foi introduzido um sistema chamado de Narrow-band Internet of Things (NB-IoT), com o intuito de prover baixo custo, baixa potência e conexão móvel (celular) para a IoT em grandes áreas (WAN) [3GPP a] [3GPP 2017] [3GPP TR 45.820 2015] [Ratasuk et al. 2017]. Na versão seguinte, Release 14, foram feitas melhorias e implementações. Pode-se citar como exemplos: posicionamento, *multicast*, e maiores taxas de dados – que acabam por aumentar o consumo de energia [3GPP b].

A NB-IoT tem sido padronizada de forma a permitir alta compatibilidade com plataformas móveis como o LTE (Long Term Evolution) [Ratasuk et al. 2016], que foi utilizado como base para seu design. Adicionalmente aos objetivos de melhora em performance, um objetivo de compatibilidade também requer que NB-IoT evite impactos negativos ao legado de sistemas celulares do 3GPP, que trabalham na mesma banda de frequência e atendem aos mesmos requisitos regulatórios.

De acordo com [Ratasuk et al. 2016], sistemas NB-IoT podem ser utilizados em três modos de operação diferentes: sinalização in-band, sinalização guard-band e sinalização stand-alone. No primeiro modo, um ou mais blocos LTE são reservados para a NB-IoT, de forma que a potência total de uma estação base seja compartilhada entre LTE e NB-IoT, com a possibilidade de aumentar momentaneamente a Densidade Espectral de Potência do sinal para NB-IoT, fazendo com que o compartilhamento de estações base entre LTE e NB-IoT permita uma maior eficiência de uso do espectro de potência [Ratasuk et al. 2016]. No segundo modo, NB-IoT pode ser utilizada na mesma frequência da portadora do LTE. Já no terceiro, NB-IoT pode ser utilizada como substituta para uma ou mais portadoras GSM.

Porém, um dos maiores problemas na utilização de NB-IoT é a segurança. Apesar de utilizar a mesma banda de GSM e LTE, as funcionalidades de segurança dessas duas tecnologias não se aplicam à NB-IoT, de acordo com [Anand and Routray 2017]. O autor cita ainda que existe outro problema, que é a falta de um robusto suprimento de serviço em tempo real. Os serviços disponibilizados pela NB-IoT não comportam aplicações em tempo real, onde [Anand and Routray 2017] mostram que a principal razão por trás disso é a falta de controle de atrasos, já que, até o momento, métodos tolerantes a atrasos não foram implementados em NB-IoT. Para que esses problemas sejam superados, é necessário que aplicações e dispositivos IoT gastem mais processamento na recepção de sinais, o que dificulta o uso dessa tecnologia.

2.3.4 Protocolos de longa distância

Protocolos sem fio de longa distância são protocolos utilizados em redes do tipo WWAN (Wireless Wide Area Network) e WMAN (Wireless Metropolitan Area Network). Pelas comparações com outras tecnologias que serão vistas mais à frente, o protocolo utilizado para a proposta foi o LoRa.

2.3.4.1 LoRa

Com o desenvolvimento das tecnologias IoT, novos tipos de protocolos têm surgido de forma a suprir essas novas demandas de comunicação entre pequenos dispositivos. O conceito de cidades inteligentes (smart cities) e comunidades inteligentes (smart communities) começaram a se tornar realidade nessa era da Internet das Coisas [Islam et al. 2017]. Com o intuito de prover comunicações IoT sem fio de baixa potência, diferentes tecnologias de comunicações têm ganhado espaço nesse cenário, tanto para pequenas quanto para grandes áreas [The LoRa Alliance] [Raza et al. 2017]. Pode-se dividir essas tecnologias em basicamente duas categorias:

1. LPLAN (Low Power Local Area Network), onde os alcances são menores que 1000 metros. Essa categoria inclui os padrões IEEE 802.15.4 (ZigBee, ISA100.11a, WirelessHART, MiWi e Thread), Bluetooth/LE, entre outros, que são aplicados diretamente em áreas pessoais de curto alcance, body area networks e, se organizadas em uma topologia mesh (todos os dispositivos da rede interconectados), também podem ser aplicadas em grandes áreas;

2. LPWAN (Low-Power Wide Area Network), com alcances maiores de 1000 metros, são essencialmente versões de baixa potência de redes celulares, de forma que cada célula cobre milhares de dispositivos. Essa categoria inclui LoRaWAN e também protocolos como Sigfox e DASH7. Resumidamente, o LPWAN é para as redes IoT o que o WiFi é para as redes tradicionais.

A rede LoRaWAN é relativamente nova e se tornou o foco de vários centros de pesquisa ao redor do mundo [Lavric and Popa 2017]. Sendo um padrão de rede para operadoras de telecomunicações baseado na tecnologia LoRa (Long Range) [The LoRa Alliance], consiste em prover serviços de rede que permitam com que dispositivos IoT transfiram dados, sem fio e a longa distância, para gateways e servidores remotos [Ke et al. 2017]. Um simples LoRa gateway, de acordo com [Islam et al. 2017] é capaz de cobrir uma pequena cidade. Os Lora gateways se conectam à servidores de rede, via tecnologias como 4G ou Ethernet. Esses servidores, por sua vez, encaminham os dados a servidores de aplicações, via protocolo TCP/IP. Por fim, os servidores de aplicação se responsabilizam por disponibilizar os dados recebidos às respectivas aplicações.

LoRaWAN (long range ou Low-Power Radio Alliance WAN) é um sistema de telecomunicações sem fio de longo alcance, baixa potência e baixa taxa de bits, proposto como uma solução de infraestrutura para a Internet das Coisas [K. Mikhaylov 2016], de forma que os dispositivos usem um nó de rede sem fio para se comunicar com gateways que estejam conectados à Internet, onde aqueles agem como pontes transparentes e retransmitem as informações entre os dispositivos e um servidor central, como visto na figura 2.6.

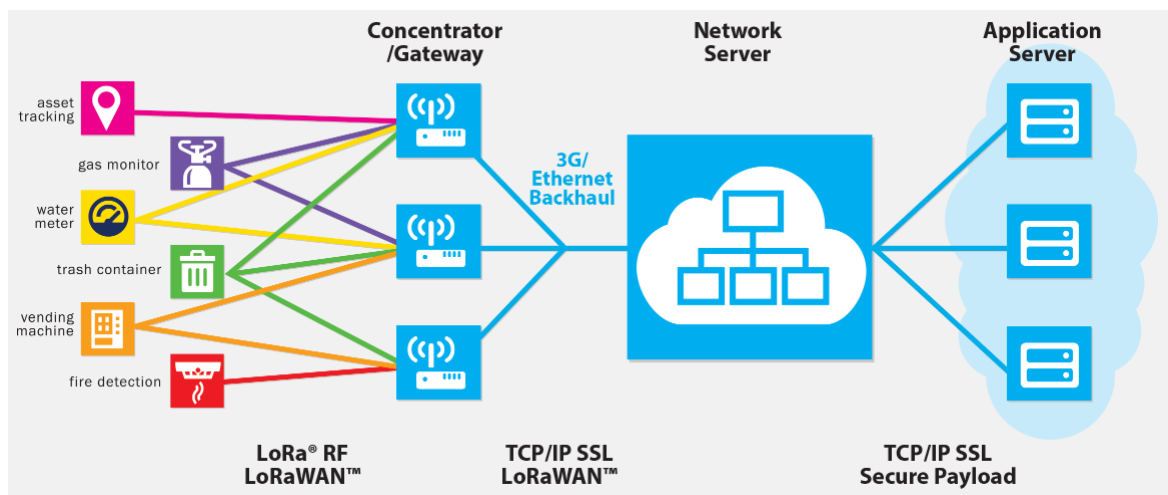


Figura 2.6: Diagrama de rede de uma rede LoRa (<http://www.semtech.com/wireless-rf/internet-of-things/what-is-lora/>)

De acordo com [Ke et al. 2017], é utilizada uma topologia do tipo estrela entre os dispositivos e os gateways, de forma que só exista um salto entre um dispositivo e um gateway. Os dispositivos-alvo da LoRa são aparelhos que costumam ter limitações energéticas e computacionais, de forma que a natureza de alto alcance e baixa potência da LoRa a torna uma interessante solução para tecnologias sensíveis inteligentes [Lavric and Popa 2017].

LoRa pode se referir a duas camadas de comunicações distintas: uma camada física que usa CSS (Chirp Spread Spectrum) e uma camada MAC (Media Access Control) [IEEE a], que é o protocolo LoRaWAN. A camada física, desenvolvida pela Semtech, permite comunicações de longo alcance, baixa taxa de dados e baixa potência, além de ser a camada respon-

sável por permitir a comunicação de dispositivos com gateways LoRa [Islam et al. 2017]. A LoRaWAN provê um mecanismo de controle de acesso que permite a comunicação de vários dispositivos com um gateway usando uma técnica de modulação desenvolvida pela própria Semtech (LoRa modulation). Porém, mesmo essa modulação sendo proprietária, a LoRaWAN é um padrão aberto desenvolvido pela LoRa Alliance.

A LoRaWan [LoRa Alliance 2015b], nos refere-se à camada MAC (Media Access Control), a qual, no caso do LoRa, determina como os múltiplos dispositivos-fim se comunicarão com gateways LoRa [Sinha et al. 2017]. Essa rede possui as seguintes propriedades:

- Sub-bandas e canais: LoRa opera em um limite específico de frequências, onde cada banda é dividida em sub-bandas, e cada sub-banda ainda é separada em um número de canais. Uma das bandas do Lora é a de 915MHz, que engloba as frequências entre 902 e 928MHz. Essa banda é repartida em oito sub-bandas, e cada sub-banda conotem 10 canais, dos quais 8 operam em downlinks de 125KHz, 1 em downlink de 500KHz, e um em uplink de 500KHz [Bor et al. 2016].

- Interferência: Cada gateway LoRa trabalha em uma sub-banda em particular. Quando dois dispositivos se comunicam com o mesmo gateway LoRa, ao mesmo tempo, utilizando o mesmo fator de espalhamento e no mesmo canal, eles irão sofrer interferência um do outro, o que pode fazer com que seus pacotes colidam e o gateway não receba os dados [Bor et al. 2016].

- Classes dos dispositivos: o protocolo define três classes para os dispositivos, de acordo com as demandas dos diferentes tipos de aplicações. A classe A engloba os dispositivos que utilizam o protocolo ALOHA [Abramson 2017] – o protocolo ALOHA é um protocolo da camada MAC que permite um nó enviar data sempre que estiver pronto – para postar pacotes de transmissão, bem como dispositivos que sejam alimentados por baterias. A classe B é destinada aos dispositivos que precisem de download adicional de pacotes. Por fim, a classe C é para dispositivos que sempre aguardam algum pacote ou transmissão, antes de transmitir alguma coisa [Bor et al. 2016].

A rede LoRaWAN, responsável por definir o protocolo de comunicação e a arquitetura do sistema [LoRa Alliance 2015a], possibilita a utilização de dispositivos mais robustos chamados de Gateway LoRa, capazes de entregar mensagens entre dispositivos-fim, ou nós, e a rede. Numa rede LoRaWAN, os nós não estão associados a um gateway específico, de forma que os dados transmitidos são recebidos por múltiplos gateways LoRa [Sinha et al. 2017]. Cada gateway se compromete de enviar os dados que foram recebidos via LoRa. A comunicação entre o gateway e a nuvem pode ocorrer através de várias tecnologias, como Ethernet, redes celulares, Wi-Fi ou até mesmo via satélite. Os servidores de uma rede LoRaWAN devem ter a inteligência de filtrar os dados duplicados recebidos [Sinha et al. 2017]. Mais à frente será falado sobre os variados tipos de Gateways LoRa.

Já a LoRa física é uma técnica de modulação que permite transferência de dados com baixas taxas e longas distâncias [Lavric and Popa 2017]. Patentada pela Semtech, foi criada com o propósito de oferecer soluções eficientes, flexíveis e baratas para problemas do mundo real, tanto em áreas rurais quanto urbanas, de forma a cobrir falhas de tecnologias já existentes, como Wi-Fi [Semtech]. A modulação LoRa utiliza o Chirp Spread Spectrum (CSS), que consiste em utilizar um sinal que varie constantemente com a frequência [Lavric and Popa 2017]. A vantagem de se utilizar esse método, de acordo com [Semtech], é que o deslocamento no tempo e a frequência são os mesmos para o transmissor e o receptor, de forma a reduzir a complexidade do receptor. De acordo com [Bor et al. 2016], o uso de CSS ameniza os efeitos de interferência, efeito Doppler e multicaminhos. A camada física do LoRa opera nas faixas ISM (The industrial, scientific and medical) de 433MHz, 868MHz ou 915MHz [Bor et al. 2016].

A LoRa Alliance é uma iniciativa sem fins lucrativos, formada por líderes de indústrias, composta por uma associação de membros que acreditam que a era da Internet das Coisas já começou. Tem como missão padronizar redes LPWAN com o intuito de facilitar a difusão da Internet das Coisas [The LoRa Alliance].

Muitas operadoras de telecomunicações ao redor do mundo estão desenvolvendo Gateways LoRa baseados na LoRaWAN. Recentemente, na Coreia do Sul [Tomás 2016], uma implementação nacional da LoRaWAN foi concluída, cobrindo aproximadamente 99% da população. Também, países como os Países Baixos e a Alemanha estão ativamente desenvolvendo redes LoRa. Um estudo recente mostra que cerca de 1,5 milhões de dispositivos LoRa estão utilizando a infraestrutura de redes dos Países Baixos [Tomás 2016].

A LoRaWAN garante segurança para dispositivos LoRa através de uma criptografia de chaves simétricas [LoRa Alliance 2015b]. Apesar disso, esses dispositivos são bastante suscetíveis à ataques de segurança. A modulação LoRa, por exemplo, demora entre 900 milissegundos e 1,2 segundos para cada transmissão, de forma que essa janela facilite a ação de ataques de segurança. Em trabalhos futuros são propostas possíveis soluções para este problema.

Dessa forma, visando transmitir dados utilizando o mínimo de processamento computacional possível, fez-se necessário avaliar as opções disponíveis atualmente para captura, transmissão e recepção de dados de um ambiente.

2.3.4.2 Gateways LoRa

Gateways Comerciais

Alguns fabricantes, pensando em se posicionar no mercado diante da crescente utilização de dispositivos baseados em LoRaWAN, desenvolveram seus próprios gateways LoRa. No site The Things Network (<https://www.thethingsnetwork.org/docs/gateways/start/>), pode-se ver alguns desses produtos.

Dentre eles, é interessante citar o Cisco LoRaWAN Gateway, mostrado na Figura 2.7. Esse aparelho é uma solução de nível de operadora adequada para usuários com experiência em software da Cisco e que estejam procurando um gateway LoRaWAN de nível industrial. Suporta 16 canais e geolocalização.



Figura 2.7: Cisco LoRaWAN Gateway

Porém, por se tratarem de dispositivos desenvolvidos por empresas conhecidas no mercado, com bastante robustez e por serem completos, acabam se tornando soluções relativamente caras, se comparadas à proposta. Além disso, neste primeiro momento, não será utilizada a rede LoRaWAN, apenas a modulação LoRa.

Gateways baseados no Raspberry Pi

Módulos embarcados, como o Raspberry Pi, por serem de fácil acesso e terem custo relativamente baixo, também têm sido utilizados como soluções para Gateways LoRa. Um projeto bastante interessante pode ser visto em (<https://www.rs-online.com/designspark/building-a-raspberry-pi-powered-lorawan-gateway>), onde foi utilizado um concentrador LoRaWAN. Para isso, foi utilizada a placa IMST iC880A, que integra dois chips *transceivers* SX1257 (<https://www.semtech>).

com/uploads/documents/DS_SX1276-7-8-9_W_APP_V5.pdf) e um chip de banda base digital SX1301 (<https://www.semtech.com/uploads/documents/sx1301.pdf>).

Um Raspberry Pi 2 Modelo B foi utilizado para executar o software de gateway, interagindo com o concentrador via SPI. Com essa combinação, é possível emular 49 demoduladores LoRa, de forma que seja possível receber 8 pacotes LoRa simultaneamente, desde que enviados com fatores de espalhamento e em canais diferentes. As placas foram alojadas em um invólucro de alumínio fundido para fornecer a blindagem de RF. A Figura 2.8 mostra a composição final do gateway.



Figura 2.8: Raspberry LoRaWAN Gateway

Apesar do baixo custo, quando comparado aos módulos LoRaWAN comerciais, a utilização de módulos Raspberry Pi ainda pode significar custo elevado na solução, até mesmo devido à necessidade de módulos adicionais, dependendo da arquitetura do gateway.

Mais à frente será falado sobre o ESP32, um microcontrolador de baixíssimo custo e consumo de energia. Muitas soluções têm sido desenvolvidas com base nesse módulo, pois o mesmo, além de ser pequeno menor que um Raspberry Pi ou Arduino, permite conexões Wi-Fi e Bluetooth, bem como uma ampla gama de integração com outras placas. Dessa forma, é possível fazer pequenos gateways LoRa com poucos recursos e sem a necessidade de utilizar a LoRaWAN.

2.3.5 Comparação entre LoRa e outras tecnologias

De acordo com [Devalal and Karthikeyan 2018], existem inúmeras tecnologias que são utilizadas em aplicações IoT, sendo que cada uma delas tem suas próprias funcionalidades, vantagens e desvantagens. Dessa forma, uma única tecnologia não atende todos os requisitos de aplicações para IoT [The LoRa Alliance]. A escolha da tecnologia utilizada só pode ser baseada nos requisitos que a proposta visa atender, em comparação com as outras tecnologias.

Sabe-se que o Wi-Fi é a tecnologia mais popular no uso de comunicações sem fio de distância relativamente longa. Existem também tecnologias como Bluetooth e Zig-Bee, que se suportam bem aplicações IoT de curta distância. Porém, nas três tecnologias citadas, a utilização de dispositivos alimentados por bateria se torna um problema [Devalal and Karthikeyan 2018]. LoRa, por outro lado, permite comunicação móvel bidirecional e de baixo custo para dispositivos IoT, M2M (machine to machine) e aplicações industriais. Além disso, a LoRa tem ganhado crescente popularidade e tem sido muito utilizada em aplicações IoT de sistemas embarcados, devido ao seu longo alcance, alta capacidade e densidade de nós na rede, consumo de bateria otimizado, comunicação bidirecional eficiente e baixa interferência [Devalal and Karthikeyan 2018].

Quando se compara LoRa com Zigbee, percebe-se que, para esta segunda, a necessidade de se utilizar vários dispositivos em *mesh* para transmitir dados entre os dispositivos até chegar ao seu destino [Kaushal et al. 2014], faz com que os nós consumam maior energia, devido ao fato de estarem interconectados. LoRa, por outro lado, ao utilizar uma topologia estrela entre os nós e o gateway, elimina a transmissão de dados para dispositivos intermediários, diminuindo o gasto de energia. Além disso, redes *mesh* não atingem a capacidade de longo alcance de redes LoRa.

Tecnologias como o Wi-Fi são do tipo LAN (Local Access Network), de forma que seu sinal costuma ficar confinado em pequenas áreas, como salas e construções médias. Mesmo em campo aberto, por maior que seja o alcance de uma rede Wi-Fi, dispositivos com essa tecnologia tendem a ser mais robustos no quesito consumo de energia.


Além de LoRa, também existe como tecnologia emergente a chamada Narrowband-IoT. Enquanto LoRa é baseada no protocolo ALOHA [Abramson 2017] e é assíncrona, a NB-IoT é baseada em FDMA, o que leva a um maior consumo de energia devido a sincronização [Sinha et al. 2017]. Em contrapartida, em desfavorecimento do LoRa, a latência e a taxa de dados suportadas pela NB-IoT são maiores [Sinha et al. 2017].

Porém, deve-se levar em consideração que nossa proposta demanda aplicações IoT que costumam ter baixas taxas de transmissão. Pode-se ver na figura 2.9 a comparação de desempenho entre algumas das tecnologias citadas acima.

Standard	ZigBee/IEEE 802.15.4	Bluetooth	UWB	IEEE 802.11 b/g
Working frequency	868/915 MHz, 2.4GHz	2.4 GHz	3.1 - 10.6 GHz	2.4 GHz
Range (m)	30 – 75+	10 – 30	~10	30 – 100 +
Data rate	20/40/250 kbps	1 Mbps	100+ Mbps	2 – 54 Mbps
Devices	255 – 65k	8		50 – 200
Power consumption	~1 mW	~40 – 100 mW	~80 – 300 mW	~160 mW – 600W
Cost (\$US)	~2 – 5	~4 – 5	~5 – 10	~20 – 50

Figura 2.9: Comparação entre ZigBee, Bluetooth e Wi-Fi (<https://www.netguru.co/codestories/the-zigbee-protocol>)

Existem também outras tecnologias emergentes voltadas para aplicações IoT, tais como Sigfox [Sigfox], LTE Cat1 e LTE M [3GPP c], contudo, pode-se ver na 2.10 que o LoRa supera as tecnologias citadas em praticamente todos os parâmetros, além de ser a que mais se aplica à ideia da nossa proposta.



COMPARISON – main LPWAN technologies

Feature	LORAWAN	SIGFOX	LTE Cat 1	LTE M	NB - LTE
Modulation	SS chip	UNB / GFSK / BPSK	OFDMA	OFDMA	OFDMA
Rx Bandwidth	500 – 125 KHz	100 Hz	20 MHz	20 – 1.4 MHz	200 KHz
Data Rate	290bps – 50Kbps	100 bit / sec 12 / 8 bytes Max	10 Mbit /sec	200 kbps – 1 Mbps	Average 20K bit / sec
Max. # Msgs/day	Unlimited	UL: 140 msg / day	Unlimited	Unlimited	Unlimited
Max Output Power	20 dBm	20 dBm	23 – 46 dBm	23/30 dBm	20 dBm
Link Budget	154 dB	151 dB	130 dB+	146 dB	150 dB
Battery lifetime – 2000 mAh	105 months	90 months		18 months	
Power Efficiency	Very High	Very High	Low	Medium	Med high
Interference immunity	Very High	Low	Medium	Medium	Low
Coexistence	Yes	No	Yes	Yes	No
Security	Yes	No	Yes Oui	Yes	Yes
Mobility / localization	Yes	Limited mobility, No localization	Mobility	Mobility	Limited mobility, No localization

Figura 2.10: Comparativo entre LoRa e outras tecnologias (<http://www.vertical-m2m.com/>)

2.3.6 Características IoT Gateway

Em Internet das Coisas, um gateway IoT é um tipo de *middleware* utilizado para resolver problemas relacionados à grande variedade de dispositivos, devido ao fato de existir uma enorme homogeneidade entre os protocolos de comunicação. Como uma ponte que conecta redes de sensores e redes de comunicações tradicionais, IoT Gateway pode prover funcionalidades para protocolos de conversação e gerenciamento de dispositivos. De acordo com [Zhu et al. 2010], divide-se IoT Gateway em três características:

1. Grande capacidade de acesso: atualmente, padrões de tecnologia de curto alcance são diversos em tecnologias como Zigbee, Z-Wave, Rubees, WirelessHART, entre outras. Entretanto, essas tecnologias são pobres em compatibilidade de protocolos;

2. Gerenciabilidade: para controlar e perceber o mundo físico, aplicações IoT enxergam sensores, antenas e aplicações rodando como se fossem o "cérebro". Porém, é necessário gerenciar milhões de nós sensoriais, de forma que o gerenciamento de IoT Gateway não significa apenas gerenciar sensores na rede, mas também gerenciar o gateway do dispositivo.

3. Protocolo de interconexão: redes de sensores e redes tradicionais precisam trocar informações entre si baseando-se nos cenários IoT. O IoT Gateway deve suportar interfuncionamento de protocolos entre redes tradicionais e WSN's (Redes de Sensores Sem fio) perfeitamente.

O IoT Gateway deve suportar agregação de dados em WSN's e transmissão de dados entre redes 2G/3G/4G, DSL e interfaces de rede [Zhu et al. 2010]. Para resolver o problema de conectividade em IoT Gateway, devem ser implementados os seguintes requisitos:

1. Encaminhamento de dados: a função básica de IoT Gateway é receber dados de uma rede de sensores e transferir para outras redes de forma correta e transparente;

2. Conversão de protocolo: o protocolo de comunicação IEEE 802.15.4, ou Zigbee, é o proposto para ser utilizado em redes WSN, enquanto as redes de Internet padrão são baseadas no protocolo TCP/IP. Isso Significa que IoT Gateway deve utilizar protocolos de comunicação sem fio de curta distância para capturar pacotes dos nós sensoriais, e usar redes móveis como 3G, além de DSL e outras interfaces de rede, para enviar esse pacotes para redes de telecomunicações ou para a Internet. Entretanto, IoT Gateway deve analisar e reempacotar os dados dos sensores baseando-se nos protocolos da WSN, e então encapsular e enviar os dados baseados nos protocolos de comunicações;

3. Gerenciamento e controle: além de enviar e receber dados, IoT Gateway deve suportar gerenciamento e controle dos nós sensoriais. Por exemplo, quando o gateway recebe comandos de um servidor remoto, ele deve processar os comandos e enviar para os sensores, de forma que servidor remoto possa gerenciar e controlar a rede de sensores através do IoT Gateway.

2.3.7 Conectividade

Hoje em dia, tecnologias Machine-to-Machine (M2M) são uma importante parte da conectividade de um sistema IoT [Strategic Research Agenda] e serve como um facilitador para o desenvolvimento da Internet das Coisas. Ao longo do rápido crescimento das tecnologias de comunicação, o cenário de conectividade tem aderido cada vez mais tecnologias como RFID e Wireless Sensor Network (WSN) [Stojmenovic 2014], como visto anteriormente. Ultimamente, tem sido visto o desenvolvimento e padronização de soluções M2M, tais como os padrões ISA100.11a, WirelessHART, e Z-Wave. Tecnologias de conectividade mais genéricas também têm sido desenvolvidas, assim como soluções de baixa potência e curto alcance, que é o caso do Bluetooth (Bluetooth SIG) e 802.15.4 (Zigbee) [Sachs et al. 2014]. Com o intuito de prover cobertura global de comunicação, o 3GPP desenvolveu módulos de conectividade para mercados com mobilidade inerente [Astely et al. 2013]. Como consequência, o campo de Internet das Coisas agora é um ponto de convergência entre promissoras tecnologias de rádio emergentes baseadas no modelo de conectividade M2M: Low-Power WiFi, redes Low-Power Wide Area (LPWA), e várias melhorias para sistemas celulares baseados em M2M.

Essas soluções, se devidamente combinadas, podem ser bastante significativas para o desenvolvimento do paradigma de Internet das Coisas, tanto do ponto de vista de disponibilidade quanto de confiabilidade. [Andreevy et al.] acredita que essas soluções podem permitir uma transformação decisiva na indústria Machine-to-Machine global, permitindo, assim, um ecossistema IoT altamente dinâmico e sustentável, perfeitamente conectado com a Internet.

2.3.8 Middleware

De acordo com [Fan et al. 2011], a arquitetura de sistemas IoT contém, basicamente, três camadas: a camada de sensores, a camada de rede e a camada de aplicação, onde a primeira é responsável por coletar e transmitir dados de objetos. Sabe-se que existem vários tipos de sensores e diferentes tecnologias utilizadas por eles, de forma que os protocolos de transmissão são bastante heterogêneos [Yun and Yuxin 2010]. De forma a resolver esse problema, indústrias têm focado no desenvolvimento de *middlewares* IoT.

O conceito de Middleware não é restrito à IoT, e pode ser visto como um software de suporte entre sistemas e aplicações de uma rede [Yun and Yuxin 2010]. Para IoT, o middleware é uma camada que intermedia a transmissão de dados entre as diferentes tecnologias utilizadas por sensores, e as várias aplicações que consomem esses dados. De acordo com [Razzaque et al. 2016], um middleware pode facilitar o processo de desenvolvimento ao integrar os vários dispositivos, bem como integrar as diversas aplicações e serviços. O autor ressalta que a maioria das propostas de middlewares para IoT estão voltadas para WSN, mas não consideram tecnologias como RFID, Machine-to-Machine e Sistemas de Supervi-

são e Aquisição de Dados (Supervisory Control and Data Acquisition – SCADA), que são elementos importantes para IoT.

Um middleware IoT deve também atender requisitos como escalabilidade, confiabilidade, disponibilidade e segurança, além de conter funcionalidades arquiteturais que permitam abstração de programação, interoperabilidade, adaptabilidade, consciência de contexto e autonomia [Palade et al. 2016].

Como exemplos de Implementações de *middlewares* IoT pode-se citar OpenIoT [Kim and Lee 2014], CHOReOS [Hamida et al.], UBIWARE [Katsonov and Cochez 2012] e LinkSmart [Kostelnik et al. 2011], os quais, de acordo com [Razzaque et al. 2016] são os mais bem-conceituados em termos de funcionalidades suportadas.

2.4 Ubiquidade

Sistemas baseados em ambientes IoT necessitam de ubiquidade, ou seja, estarem disponíveis em qualquer lugar a qualquer hora. Juntamente com o crescimento de dispositivos sem fio e a difusão de suas diversas infraestruturas, esse contexto tem permitido novos cenários para o fornecimento de serviços, de forma que os usuários consigam descobrir novos dispositivos, conectar, trocar informações e se desconectar dos mesmos, mesmo estando em movimento [Gershenfeld et al. 2004]. Para melhor explorar o potencial de conectividade desses objetos e construir sistemas mobile cada vez mais modernos e ubíquos, existem conceitos considerados essenciais.

2.4.1 Serviços sensíveis ao contexto

As definições de Contexto ainda são muito amplas e vagas para identificar os aspectos que o designer deve considerar úteis para modelar e descrever onde um dado serviço será desenvolvido e executado [Bellavista et al. 2013], de forma que muitos autores acabam estabelecendo suas próprias definições de Contexto. Pode-se citar como exemplos [N. et al.], que define contexto como "onde estiver, com quem estiver e recursos disponíveis"; [Dey and Abowd 2000], onde contexto é "qualquer informação que pode ser usada para caracterizar a situação de uma entidade"; e [Zimmermann et al. 2007], que divide os elementos da descrição de contexto em 5 categorias: individualidade, atividade, localidade, tempo e relações". Resumidamente, em um senso comum, e de acordo com [Bolchini et al. 2009], "Contexto é um conjunto de variáveis que podem ser de interesse de um agente, e que influenciam em suas ações".

2.4.2 Distribuição de dados em ambientes ubíquos móveis

O conceito de Serviços sensíveis ao contexto permite a interconectividade entre dispositivos em várias situações diferentes, como mudança de local físico e de informações, de forma a adaptar as variáveis de desenvolvimento do ambiente mobile para um melhor desempenho da interação entre os dispositivos, dependendo do contexto em que o ambiente está inserido [Bellavista et al. 2013].

Para que um sistema ubíquo mobile seja capaz de compreender o mundo real através de um serviço sensível ao contexto, são necessárias tarefas complexas que demandam um alto grau de conhecimento em várias tecnologias e que incluam operações nada triviais, englobando diferentes camadas e plataformas. [Chen and Kotz 2000] mostra que a maior causa para a lentidão na implementação de serviços sensíveis ao contexto, em sistemas ubíquos móveis, é a dificuldade em gerenciar esses aspectos tecnológicos. Dessa forma, para facilitar a difusão de serviços sensíveis ao contexto, faz-se necessário o uso de soluções middleware sensíveis ao contexto, com o intuito de tornar mais transparente as fases envolvidas no processo de contextualização para camadas de serviços como, por exemplo, representação, memorização, agregação e distribuição de dados [Baldauf et al. 2007]. Assim, enquanto um *middleware* de contexto é responsável por executar operações de forma a distribuir dados de contexto, os serviços sensíveis a contexto deverão produzir e enviar esses dados, declarando quais serão recebidos pelo *middleware*. [Bellavista et al. 2013] divide a distribuição desse tipo de dados em duas classificações:

- Distribuição não informada: se caracteriza por encaminhar dados de contexto de acordo com as necessidades dos dispositivos, de forma que os dados sejam entregues sem que haja uma inspeção do conteúdo.
- Distribuição informada: nesse caso, as informações do dado de contexto são utilizadas para se adaptar automaticamente e autogerir o processo de distribuição da informação.

Assim, [Bellavista et al. 2013], com o intuito de definir os principais processos realizados desde a produção até o consumo dos dados de contexto, sugeriu uma arquitetura, apresentada na figura 2.11. Tal arquitetura é descrita em três componentes, que são: fontes de contexto, mediador de dados de contexto e consumidores de contexto.

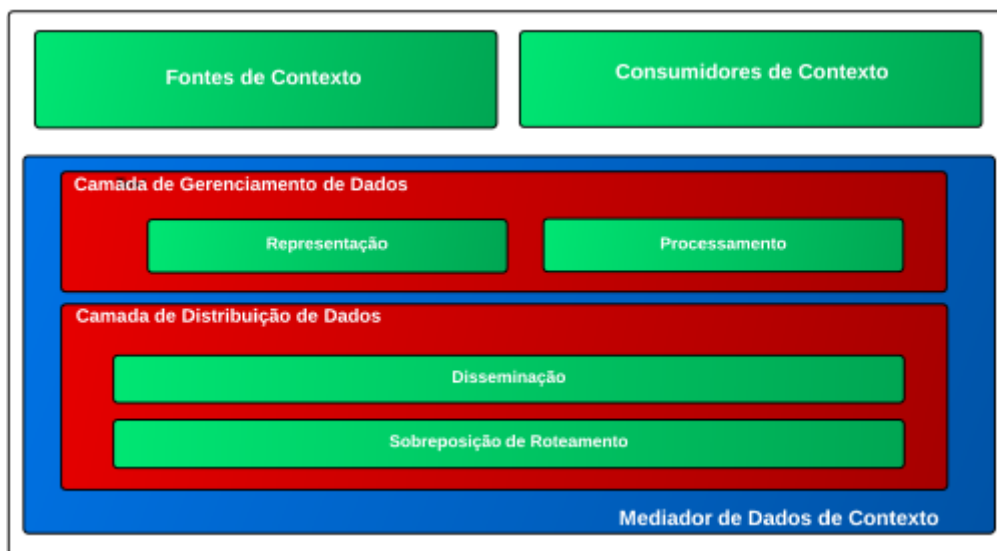


Figura 2.11: Arquitetura sensível ao contexto

Pode-se definir fontes de contexto como responsável por abranger todos os serviços de acesso aos sensores, bem como habilitador da distribuição de dados de contexto. Os consumidores de contexto são as requisições geradas pelos serviços e aplicações. Já o mediador de dados de contexto permite a interação entre fontes e consumidores, de forma que a representação, processamento e distribuição de dados é descrita por ele.

2.5 Mobilidade

A visão do conceito de Internet das Coisas fortalece a necessidade de que dispositivos devam processar e trocar informações em movimento e em qualquer lugar. Nesse contexto, tecnologias wireless tem um papel essencial, por permitirem a comunicação móvel de curto, médio e longo alcance. Ademais, a crescente diminuição de tamanho, peso, energia, consumo e custo desses equipamentos promove um avanço cada vez maior do uso dos mesmos, permitindo uma interação wireless com quase todos os tipos de objetos, fazendo referência ao conceito de "qualquer coisa", no contexto de IoT. Ainda, como visto em [Srivastava 2006], a proporção entre dispositivos wireless e humanos está se aproximando de 1:1, tornando a problemática de conectividade um grande desafio científico.

Uma importante tecnologia aplicada ao conceito de IoT são as chamadas redes sensoriais que, juntamente com RFID, são capazes de rastrear status, localização, temperatura, movimentos, entre outros, de objetos ligados à rede, bem como servir como uma ponte entre o mundo físico e o digital.

Assim, dentre os principais componentes da Internet das Coisas estão os sistemas RFID, que são, basicamente, compostos de um ou mais leitores e várias etiquetas RFID. Essas etiquetas são definidas como um identificador único responsável por caracterizar cada objeto. Do ponto de vista físico, etiquetas RFID são pequenos microchips ligados a uma antena, de dimensões também pequenas, que é usada tanto para receber o sinal do leitor quanto para transmitir o identificador do objeto ou da etiqueta.

Outro sistema que também pode ter uma grande influência no avanço de IoT são as redes sensoriais, que podem contribuir com sistemas RFID para rastreamento de status, tais como localização e temperatura, dos objetos. Redes de sensores, ou redes sensoriais, são compostas de um grande número de nós conectados via wireless, em um modelo multi-hop, que é uma arquitetura baseada em múltiplos saltos. Esse nós são responsáveis por transmitir informações para nós principais, chamados de "sink". Uma das principais vantagens dessa arquitetura é a redução no consumo de energia de transmissão, dado que as informações não são enviadas diretamente ao sink, pois primeiramente são distribuídas entre os nós vizinhos.

2.6 P2P

O conceito de comunicação par a par (P2P) não é recente. P2P consiste em um conjunto de nós, com certa capacidade computacional, vastamente distribuídos geograficamente, onde cada nó (ou par) troca informação entre si, de forma que a escolha dos pares é completamente autônoma. Sistemas como Napster e Gnutella foram responsáveis por popularizar o P2P, e hoje em dia os torrents são os mais conhecidos por usar esse tipo de arquitetura.

A principal ideia é que discos e largura de banda, para armazenamento e transferência de arquivos, respectivamente, sejam compartilhados entre os pares na rede, resultando em um serviço de baixo custo, pois não necessitam de servidores caros e com alto poder computacional. O sistema de compartilhamento Kaaza, por exemplo, reportou que no dia 30 de abril de 2003, um total de 5,5 milhões de usuários compartilharam cerca de 7 petabytes de dados, o que mostra uma dimensão do que o P2P é capaz.

2.6.1 Roteamento

Uma importante parte da elaboração da proposta é definir como será a estrutura e os protocolos de comunicação e roteamento. Devido aos conceitos que englobam Internet das Coisas, a comunicação sem fio é, logicamente, a única capaz de permitir mobilidade, e é a principal base para uma comunicação ubíqua e móvel. Assim, redes de dispositivos ubíquos serão responsáveis por unir mobilidade e onipresença, de forma que distância geográfica deixe de ser um fator limitante para a projeção de sistemas e serviços.

2.6.1.1 Redes de Sensores Sem fio

Redes de sensores sem fio são, geralmente, redes compostas por pequenos sensores de baixo custo, responsáveis por coletar e distribuir dados do ambiente em que se inserem. Essas redes permitem o monitoramento e controle de ambientes físicos remotamente. Sabe-se que os sensores têm capacidade computacional limitada, porém, se a interação entre os sensores for bem coordenada e as tarefas devidamente distribuídas, o ambiente físico pode ser caracterizado de maneira mais detalhada.

Devido ao fato de combinar tecnologias de comunicação e computação com a tecnologia de uma rede de sensores, é considerada uma das dez tecnologias emergentes mais promissoras, de forma que irá afetar significativamente o futuro da civilização humana. Essas redes são compostas por inúmeros e ubíquos nós de micro-sensores que têm a capacidade de se comunicar e, em alguns casos, fazer pequenos cálculos. Esses nós podem monitorar, sensoriar e coletar informações de diferentes ambientes em cooperação com vários outros objetos de monitoramento [Zhang et al. 2008].

De acordo com [Bharathidasan 2002], redes de sensores sem fio dependem de uma implementação densa e coordenada para realizar um conjunto de tarefas, de forma que se necessita de diferentes abordagens para trabalhar com sensores. [Akildiza 2002] diz que, apesar dos diferentes algoritmos e protocolos para redes ad-hoc sem fio, as características de redes de sensores demandam outras implementações. Como micro sensores têm uma natureza bastante pervasiva, ou seja, são imperceptíveis ao usuário final, redes de sensores têm o potencial de revolucionar a própria maneira de entender e construir sistemas físicos complexos [Estrin 1999].

Redes de sensores são sistemas capazes de capturar informações sobre o meio, processá-las e divulgá-las para as aplicações necessárias, porém não se pode deixar de lado as limitações computacionais e energéticas desses dispositivos. Dessa forma, as aplicações podem tomar decisões baseadas nas informações do ambiente, tornando redes de sensores sem fio altamente importante para aplicações sensíveis ao contexto, que é o caso da Internet das Coisas.

2.6.1.2 Principal dificuldade: problemas de endereçamento

Sabe-se que a IoT incluirá uma grande quantidade de dispositivos, onde cada um deles será considerado como um nó, em se tratando de endereçamento, o que requer políticas efetivas de distribuição de endereços. Atualmente, o protocolo IPv4 utiliza um endereço de 4 Bytes para identificar cada nó, o que reforça que o número de endereços IPv4 já está próximo de zero. Uma das propostas para o endereçamento de nós de comunicação sem fio de baixa potência é o 6LOWPAN, baseado no protocolo IPv6, que é caracterizado por uma sequência de 128 bits, possibilitando, assim, o uso de 10^{28} endereços.

Além disso, o uso do protocolo TCP para redes IoT acaba sendo um tanto quanto inadequado, devido aos seguintes motivos:

1. Configurações de conexão: o protocolo TCP é orientado a conexão, e cada uma de suas sessões começa com um procedimento padrão, bastante conhecido como "three way handshake". Como uma conexão baseada em ambientes IoT envolve uma pequena quantidade de dados, o TCP se torna desnecessário pois seu padrão acaba tomando parte considerável do tempo da sessão. Além disso, o "three way handshake" seria processado por dispositivos que devem ser bastante limitados em termos de energia, que é o caso do RFID.

2. Controle de congestionamento: O TCP também é responsável pelo controle de fluxo fim-a-fim, o que pode causar problemas de performance em IoT, visto que ele utilizará sinais sem fio de baixa potência. Sem contar com o fato de que uma sessão IoT troca baixas quantidades de dados, de forma que quase não ocorram congestionamentos, transformando, assim, o controle de fluxo de dados em uma funcionalidade redundante.

3. Bufferização de dados: Como o TCP faz o máximo esforço para garantir a entrega de dados, ele utiliza buffers, que são responsáveis por armazenar dados a serem retransmitidos em caso de perda de conexão. Como dispositivos IoT são limitados em termos de energia e processamento, essa bufferização se torna altamente inviável.

Em vista do que foi apresentado acima, vê-se que o protocolo TCP é ineficiente para o controle de fluxo de dados fim-a-fim em redes IoT. Porém, é conhecido que as características do tráfego em redes wireless dependem fortemente do cenário de aplicação, como visto em [D. Singh 2004]. Assim, uma definição para a característica de tráfego no cenário IoT é muito importante, para que seja planejada a expansão dessa infraestrutura. Muito já foi feito para a melhora da qualidade de serviço (QoS) em redes wireless, como visto em [Chen and Varshney 2004]. Porém, no que se refere a sistemas RFID, essa questão ainda não é tão explorada.

2.7 Arquitetura de IoT

Além das visões e conceitos que temos de Internet das Coisas, deve-se também olhar para um lado mais técnico. Baseando-se em uma perspectiva de tecnologia, a arquitetura de IoT deve ser proposta considerando aspectos como abrangência, escalabilidade, modularidade, e interoperabilidade entre diferentes tipos de dispositivos. Como os dispositivos são, em maior parte, móveis, e interagem em tempo real com o ambiente IoT, é necessária uma arquitetura que seja adaptável, de forma a permitir que os dispositivos troquem informações dinamicamente. Como a natureza de um ambiente IoT é descentralizada e bastante heterogênea, pode-se utilizar os conceitos de Orientação à Serviço com o objetivo de alcançar essa interoperabilidade entre dispositivos heterogêneos [Atzori et al. 2010], [Miorandi et al. 2012], [Xu 2011], de forma resume-se a arquitetura IoT em quatro camadas: camada sensorial, camada de rede, camada de serviços e, por final, interface.

2.7.1 Camada Sensorial

Como os dispositivos em IoT podem ser interligados em uma rede de abrangência global e controlados remotamente, quanto mais dispositivos equipados com RFID ou sensores inteligentes, maior será a facilidade em conectar esses dispositivos [Wu et al. 2013]. Assim, a ideia é que, através de tags e sensores, dispositivos móveis e inteligentes possam pairar, se conectar e trocar informações automaticamente, aumentando a capacidade de um ambiente IoT identificar objetos. Em algumas indústrias, sistemas de serviços inteligentes e um Identificador Universal Único (Universal Unique Identifier – UUID) estarão assimilados com cada serviço ou dispositivo necessário. Um UUID pode ser facilmente identificado, de forma que esses identificadores sejam essenciais para o desenvolvimento de serviços em grandes redes, como em Internet das Coisas [Wu et al. 2013], [Ilie-Zudor et al. 2011].

2.7.2 Camada de Rede

O papel da camada de rede é conectar todas as coisas e permitir que elas troquem informações entre si. Além disso, essa camada é capaz de agregar dados de infraestruturas de TI já existentes. Com o conceito de SOA, serviços fornecidos por objetos são tipicamente desenvolvidos em uma rede heterogênea [Atzori et al. 2010], [Han et al. 2013]. Esse processo envolve gerenciamento e controle de Qualidade de Serviço (Quality of Service – QoS), de acordo com as requisições dos usuários/aplicações. Por outro lado, é essencial que uma rede que muda dinamicamente possa automaticamente descobrir e mapear objetos na rede. Para desenvolver a camada de rede em IoT, projetistas precisam distribuir problemas como tecnologias de gerenciamento de rede para redes heterogêneas, eficiência de energia em redes, requisitos de QoS, descoberta de serviço, processamento de dados e sinais, e segurança [Guinard et al. 2010].

2.7.3 Camada de Serviço

A camada de serviço se baseia em middlewares que promovem funcionalidades para integrar serviços e aplicações em IoT. O *middleware* permite que IoT tenha eficiência de custo, onde o hardware e o software podem ser reutilizados. A principal atividade da camada de serviços envolve especificações de serviço para *middlewares*, que podem ser desenvolvidos por diversas organizações. Uma camada de serviço bem definida pode ser capaz de identificar requisitos comuns entre aplicações e prover API's e protocolos que suportam os serviços requisitados, aplicações e necessidades do usuário. Essa camada também processa todos os problemas orientados a serviço (visão orientada à semântica em IoT), incluindo troca e armazenamento de informações, gerenciamento de dados, ferramentas de busca, e comunicações [Atzori et al. 2010], [Miorandi et al. 2012], [Guinard et al. 2010]. Essa camada inclui os seguintes componentes:

1. Descoberta de serviço: encontrar objetos que podem oferecer os serviços e informações necessários de forma eficiente [Atzori et al. 2010];
2. Composição de serviço: permitir interação e comunicação entre objetos conectados. A descoberta procura em diferentes dispositivos o serviço desejado, ao passo que a composição agenda ou recria serviços de forma a pegar os serviços mais confiáveis para atender o requisito [Atzori et al. 2010], [Miorandi et al. 2012];
3. Gerenciamento de confiança: tem o objetivo de determinar mecanismos de confiança e reputação, usando a informação provida por outros serviços para criar um sistema confiável [Atzori et al. 2010], [Han et al. 2013], [Guinard et al. 2010].
4. API's de serviço: auxilia as interações entre os serviços requisitados em IoT [Zhang and Zhu 2011], [Guinard et al. 2010].

2.7.4 Interface

Em redes IoT, os dispositivos envolvidos são produzidos por fabricantes diferentes, sendo que nem sempre eles seguem os mesmos padrões e protocolos. Como resultado de toda essa heterogeneidade, existem muitos problemas na troca de informações e comunicações entre objetos. Além disso, o crescente aumento de dispositivos que compõem a IoT dificulta o processo de, dinamicamente, conectar, comunicar, operar e desconectar esses dispositivos. Também existe a necessidade de uma camada para simplificar o gerenciamento e a interconexão entre os objetos. Uma interface pode ser vista como um padrão de serviços que permitem interação entre aplicações na rede. Uma boa interface está relacionada com a implementação do Universal Plug and Play (UPnP), que define um protocolo facilitador da interação entre serviços fornecidos por vários objetos [Guinard et al. 2010], [Gama et al. 2012]. A interface é usada para descrever especificações entre aplicações e serviços, onde os serviços da camada de serviços rodam diretamente em infraestruturas de rede limitadas, de forma a encontrar novos serviços para uma aplicação, assim que se conectam à rede. Baseado no contexto da aplicação, e na QoS, todos os serviços são ranqueados, facilitando a identificação do serviço que mais combina com a requisição da aplicação e, no final, esse processo é avaliado [Zhou 2012], [Atzori et al. 2012].

2.8 Considerações sobre os conceitos abordados

Tendo em vista os conceitos abordados, será elaborado um projeto que visa utilizar as possíveis aplicabilidades de Internet das Coisas para solucionar problemas de conectividade em ambientes móveis, bem como garantir a disponibilidade de dados em ambientes IoT. A principal ideia é criar uma estação meteorológica móvel e autônoma, capaz de captar dados do ambiente e enviá-los a um *middleware* de interesse. Para isso, serão utilizados os protocolos e conceitos abordados, fazendo com que os dados captados por um dispositivo sejam devidamente coletados, tratados, transmitidos e disponibilizados. Ao final da transmissão, esses dados deverão servir como base para gerar informações aos usuários finais, de forma que estes possam saber de forma clara e precisa a situação do clima no ambiente.

Capítulo 3

Estrutura e Funcionamento da Estação Meteorológica Proposta

3.1 Introdução

O principal intuito da proposta foi elaborar o projeto de uma estação meteorológica inteligente e autônoma, utilizando-se dos conceitos abordados em Internet das Coisas. Alguns dos atributos são que a estação seja pequena, possa ser colocada em diferentes locais do campus e seja autossuficiente em termos de consumo de energia. Os dados capturados pela estação serão enviados via LoRa para um dispositivo chamado de Gateway LoRa. O segundo passo é enviar os dados para um outro gateway via Wi-Fi, utilizando uma rede que permita acesso ao Gateway UIoT. Os dados são enviados de forma que o gateway saiba identificar os tipos de dados, bem como a Estação transmissora.

Os dados recebidos são enviados para o Middleware UIoT, onde esse é responsável por tratar e disponibilizar dados, com o intuito de informar usuários sobre a situação do clima no ambiente onde estará instalada a estação. Neste capítulo serão descritos os componentes envolvidos, bem como a arquitetura da solução proposta, os protocolos utilizados e o ambiente em que será implementada.

Assim, propõe-se uma solução onde os dados sejam capturados por sensores e transmitidos via conexão LoRa. Para isso, serão usados dois dispositivos, onde o primeiro é o transmissor e o segundo é o receptor. Este segundo módulo, deverá tratar os dados de forma que eles sejam enviados ao gateway nos parâmetros corretos, seguindo seus protocolos e bibliotecas, com o objetivo final de disponibilizar dados coletados pela estação.

3.2 Arquitetura da Proposta

Para que os dados capturados pelo dispositivo LoRa sejam corretamente enviados e disponibilizados, será utilizada uma arquitetura DeviceLoRa-GatewayLoRa-GatewayUIoT-MiddlewareUIoT. Dessa forma, o primeiro módulo fará a captura dos dados dos sensores ligados à estação e enviará esses dados ao segundo. Este deverá tratar os dados recebidos de forma que eles possam ser corretamente enviados ao Gateway UIoT.

Como visto na seção sobre distribuição de contexto, em sistemas ubíquos móveis, a camada de entrega de dados de contexto é responsável pelo roteamento dos dados. Nessa proposta, essa camada é o Gateway LoRa, o qual será responsável por enviar ao Gateway UIoT os dados recebidos de outros dispositivos.

3.2.1 Gateway

Em Internet das Coisas, um gateway IoT deve ser capaz de receber dados dos mais variados tipos de serviços, através dos diferentes protocolos existentes. No laboratório UIoT da Faculdade de Tecnologia, foi desenvolvido um gateway semântico [de Caldas Filho et al. 2017], chamado de Gateway Semântico UIoT.

Além de ser capaz de oferecer uma interface REST/JSON que pode ser utilizada pelas várias aplicações de IoT, diz-se que o gateway é semântico devido ao fato de possibilitar o recebimento de dados de dispositivos que se comunicam utilizando diferentes tecnologias. O passo seguinte é encaminhar esses dados para um *middleware*, após o trabalho de interpretação semântica e transformação de dados, com o intuito de permitir a disponibilidade dos dados capturados pela estação.

3.2.2 Middleware UIoT

O Middleware UIoT é um software capaz de concentrar serviços e interfaces dentro de uma infraestrutura integrada e homogênea [Martins et al. 2017], e foi desenvolvido de forma a dar suporte aos objetos inteligentes.

Com o intuito de permitir modularização, foram projetadas duas aplicações distintas para compor o *middleware* em questão. A primeira delas é o RAISe, que significa REST *Abstract Service Layer* e contém uma API REST que pode ser consumida por qualquer aplicação que integre a rede do UIoT. A segunda aplicação é o UIMS (User-oriented IoT Management System), um software responsável pela consulta amigável dos dados gerados pelos elementos da rede [Martins et al. 2017].

Com isso, é possível criar aplicações capazes de disponibilizar os dados recebidos, como mostrado na Figura 3.1

Lora 1



ENDPOINT: 172.16.9.69 NAME: lora-1

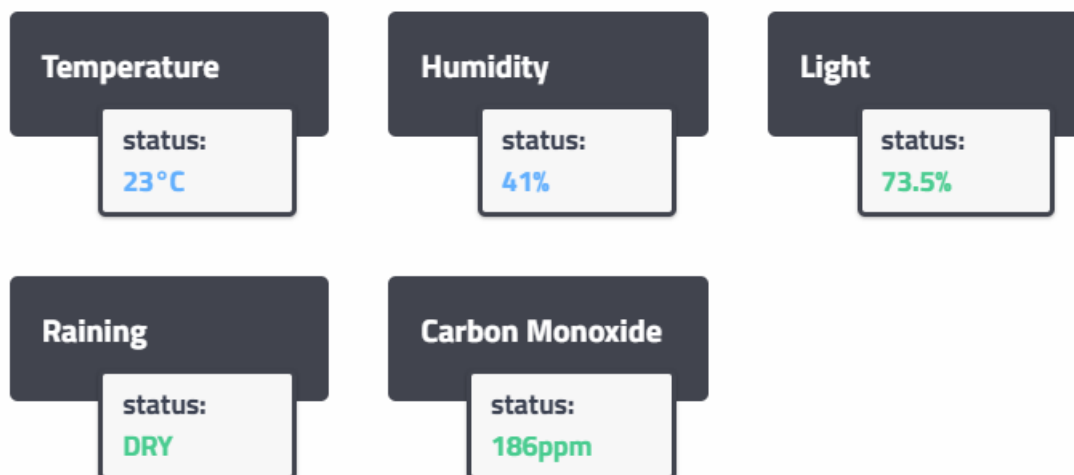


Figura 3.1: Dados disponibilizados

3.2.3 Arquitetura de transmissão

A Figura 3.2, nos mostra uma abstração para a arquitetura da proposta. Podemos ver que os dados são capturados e transmitidos via LoRa. Esse primeiro Gateway se comunica diretamente com o Gateway UIoT, o qual, através do protocolo REST, envia os dados dos dispositivos para o Middleware UIoT, após serem devidamente tratados e interpretados. Essa arquitetura permite que várias estações transmitam dados para um mesmo concentrador, centralizando o tratamento dos dados.

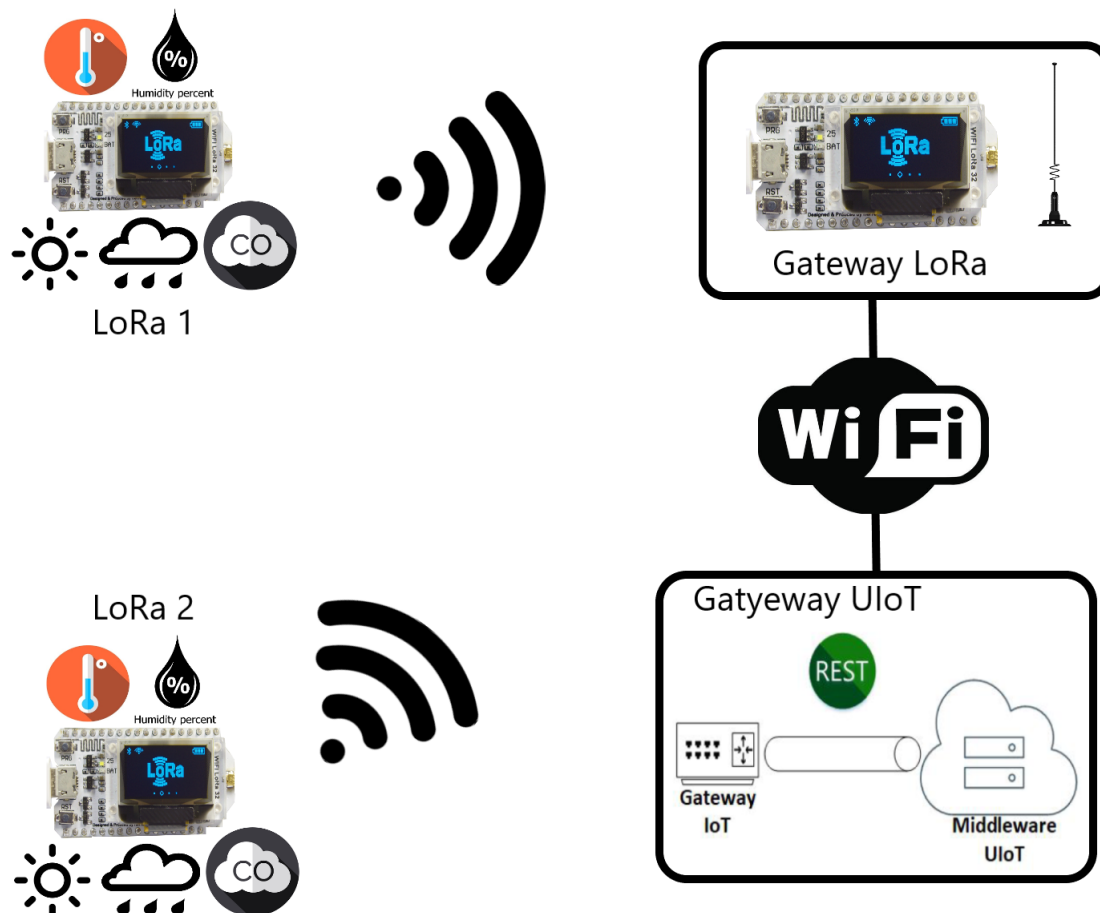


Figura 3.2: Arquitetura proposta

3.3 Análise de dispositivos de baixo custo

3.3.1 Sensores

Para fazer a captura dos dados de temperatura, umidade, luminosidade, de chuva e de qualidade do ar, foram necessários sensores específicos para essas funcionalidades. Cada um deles possui esquemáticos e pinagem próprias, de forma a utilizá-los em conjunto da forma mais apropriada possível.

Temperatura é a medida da energia cinética média das partículas em uma amostra, expressada em termos de graus designados em uma escala padrão, enquanto a umidade é uma medida da composição de água no ar [Kopuklu and Ertas 2013]. Ambas têm impacto em todos os organismos, seja ele única célula ou até mesmo o organismo mais complexo [Tang 2009]. Esses dados são facilmente medidos manualmente com dispositivos de baixo custo como termômetros digitais ou higrômetros [Sookchaiya et al. 2008]. Porém, esse tipo de medição acaba consumindo tempo desnecessário, além de ser suscetível a erros de usuários. Muitos desses problemas de medição são solucionados utilizando medidores precisos e automatizados, de forma a automaticamente coletar, armazenar e disponibilizar dados me-

teorológicos [Kopuklu and Ertas 2013]. Métodos avançados e automatizados de medições empregam dispositivos eletrônicos com sensores dedicados para a captura dos respectivos dados [FasLab 2013, Zheng et al. 2010]

3.3.1.1 Sensor de temperatura e Umidade

Para medir-se esses dois dados extremamente importantes para o monitoramento do clima de um ambiente, foram pesquisados sensores com essas funcionalidades. Como os únicos que mediam simultaneamente eram os DHT11, DHT22 e o AM2302, os outros foram descartados. Isso se deve ao fato de que sensores a mais podem ocasionar um maior consumo de corrente, sendo que o foco é obter maior economia de energia. Os sensores citados são bastante parecidos. Ambos são pequenos, possuem alimentação entre 3 e 5V, consomem 2,5mA de corrente máxima e possuem precisão parecida. Foi utilizado o sensor DHT11 (Figura 3.3), do fabricante AOSONG, devido ao fato de possuir uma maior taxa de amostragem em relação aos outros dois. O DHT11 possui 1 Hz de taxa de amostragem, enquanto o DHT22 e o AM2302 possuem 0.5 Hz. Esse sensor é bastante utilizado em monitoramento de ambientes com microcontroladores, além de ter um baixíssimo custo. Estes foram os principais fatores para a escolha do mesmo, além do fato de ser capaz de medir dois dados de bastante significância ao mesmo tempo, com uma única saída digital.

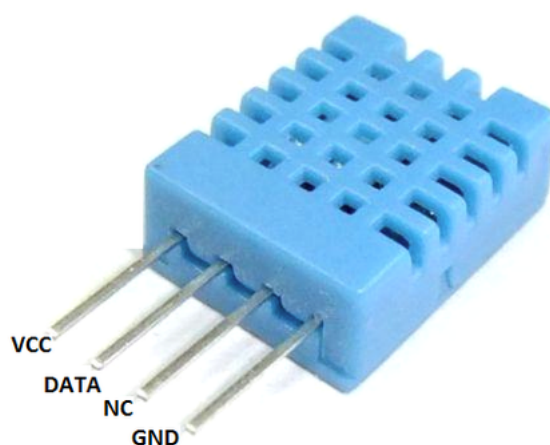


Figura 3.3: Sensor DHT11

O DHT11 pode ser utilizado em várias aplicações, como HVAC (Heating, ventilation, and air conditioning), estações meteorológicas, reguladores de ambiente, entre outras. Como principais características, possui baixo custo, alta durabilidade, qualidade das medições, resposta rápida, forte proteção contra interferência, transmissão de sinal de longa distância, saída de sinal digital e calibração precisa [AOSONG].

O mais comum é encontrá-lo na forma de módulo, o qual possui apenas 3 pinos (Vcc, dados e GND), já que o pino NC não é utilizado.

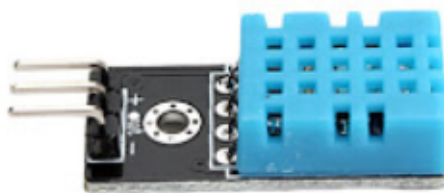


Figura 3.4: Módulo DHT11

A vantagem de se ter utilizado o módulo é que o mesmo já possui um capacitor e um resistor embutido, de forma que não foi necessário adicionar esses componentes no circuito [DHT11].

No manual do fabricante, pode-se conferir as seguintes especificações técnicas do produto:

5、Product parameters

Relative humidity

Resolution: 16Bit

Repeatability: $\pm 1\%$ RH

Accuracy: At 25°C $\pm 5\%$ RH

Interchangeability: fully interchangeable

Response time: 1 / e (63%) of 25°C 6s

1m / s air 6s

Hysteresis: $< \pm 0.3\%$ RH

Long-term stability: $< \pm 0.5\%$ RH / yr in

Temperature

Resolution: 16Bit

Repeatability: $\pm 0.2^\circ\text{C}$

Range: At 25°C $\pm 2^\circ\text{C}$

Response time: 1 / e (63%) 10S

Electrical Characteristics

Power supply: DC 3.5 ~ 5.5V

Supply Current: measurement 0.3mA standby 60 μ A

Sampling period: more than 2 seconds

Pin Description

1, the VDD power supply 3.5 ~ 5.5V DC

2 DATA serial data, a single bus

3, NC, empty pin

4, GND ground, the negative power

Figura 3.5: Especificações técnicas [AOSONG]

3.3.1.2 Sensor de Luminosidade

O sensor de luminosidade LDR (Light Dependent Resistor) é um sensor o qual sua resistência interna varia de acordo com a intensidade da luz identificada por ele. Quanto maior a intensidade da luz, menor o valor da resistência. Em uma temperatura de 25 graus Celsius,

o valor da resistência é entre 8 e 20K ohm, para uma medida de 10Lux. No ambiente mais escuro, de 0Lux, o valor da resistência é de 1M ohm. Lux (lx) é a unidade Sistema Internacional de medidas para luminosidade, que mede a incidência perpendicular de 1 lúmen em uma superfície de 1 metro quadrado. Uma foto do sensor LDR pode ser vista na Figura 3.6

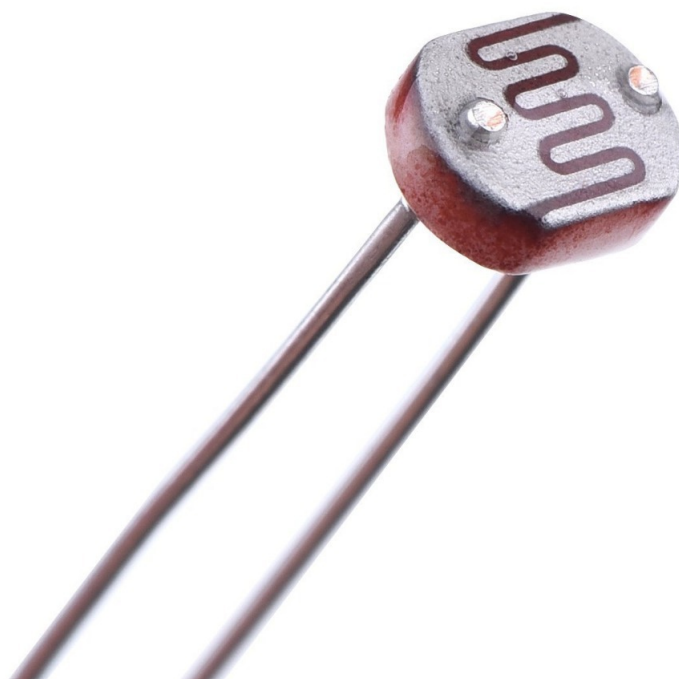


Figura 3.6: Sensor LDR

Especificações:

- Modelo: GL5528
- Diâmetro: 5mm
- Tensão máxima: 150VDC
- Potência máxima: 100mW
- Tensão de operação: -30 a 70 graus Celsius
- Espectro: 540nm
- Comprimento com terminais: 32mm
- Resistência no escuro: 1 M ohm (Lux 0)
- Resistência na luz: 8-20 K ohm (Lux 10)

Para capturar os valores medidos pelo sensor de luz, foi utilizado um módulo controlador/comparador YL-38, mostrado na Figura 3.7. O módulo contém 1 pino para a alimentação do VCC de 3,3 a 5V e GND (terra) de 0V; e 2 pinos para dados de saída, sendo: D0 a interface de saída digital (H = 1 / L = 0) e A0 a interface de saída analógica. Este módulo permite ajustar a medição do sensor através de um potenciômetro.

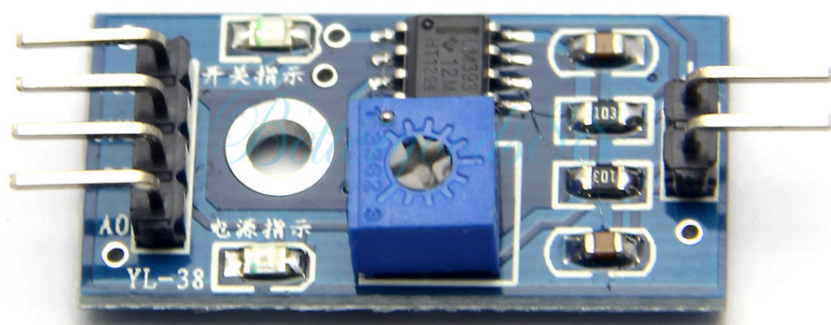


Figura 3.7: Módulo YL-38

Além do potenciômetro, o módulo compara a voltagem de entrada utilizando um CI LM393 [ON Semiconductor]. A Figura 3.8 mostra o Circuito LM393.

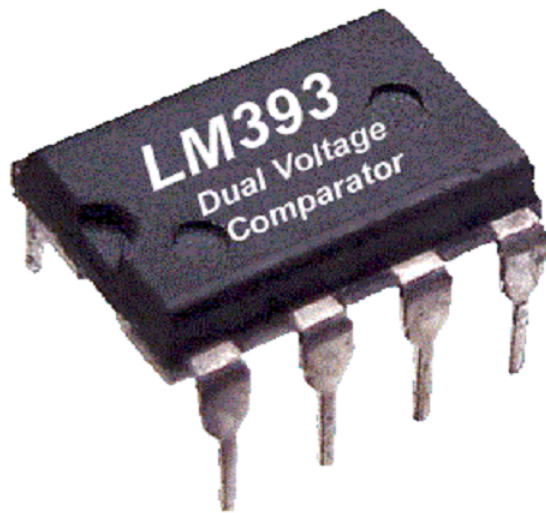


Figura 3.8: Circuito LM393

Funcionalidades LM393:

- Alimentação simples: 2,0 Vdc até 36 Vdc
- Alimentação dupla: 1,0 Vdc até 18 Vdc
- Corrente de drenagem baixa, independente da tensão de alimentação: 0,4 mA
- Baixa corrente de polarização de entrada: 25 mA
- Corrente de entrada de baixo deslocamento: 5.0 mA
- Voltagem de entrada de baixo deslocamento: 5.0 mV (max) LM293/393
- Entrada comum para o terra
- Faixa de tensão de entrada diferencial igual à tensão de alimentação
- Voltagem de saída compatível com os níveis lógicos DTL, ECL, TTL, MOS, e CMOS
- Grampos de ESD nas entradas aumentam a robustez do dispositivo sem afetar o desempenho
- Esses dispositivos são livres de chumbo, livres de halogêneo e atendem aos requisitos RoHS

3.3.1.3 Sensor de Chuva

O módulo sensor de chuva é uma ferramenta simples capaz de identificar precipitação de água, medindo também a intensidade da mesma. A resistência do sensor aumenta de acordo com a quantidade de água depositada nele, de forma que os valores diminuem proporcionalmente à intensidade da chuva. Possui uma saída analógica e uma saída digital, responsáveis por disponibilizar os valores lidos. Este sensor, assim como o de luminosidade, utiliza o módulo YL-38 para fazer a comparação dos dados de entrada. Porém, a placa responsável por capturar esses valores é a YL-83, mostrada na Figura 3.9.



Figura 3.9: Sensor de chuva YL-83

Especificações:

- Adota alta qualidade de material de dupla face RF-04
- Dimensões: placa de níquel de 5cm x 4cm
- Anti-oxidação, anti-condutividade, com longo tempo de uso
- Boa forma de onda de sinal de saída do comparador limpo
- Capacidade de condução de mais de 15mA
- Potenciômetro para ajustar a sensibilidade
- Voltagem de operação de 5V

- Formato de saída: Saída de comutação digital D0 (0 e 1) e saída de tensão analógica AO

- Com furos para instalação fácil
- Usa um comparador LM393 de tensão ampla

3.3.1.4 Sensores de Gás

Para saber a qualidade do ar, é necessário medir a quantidade de Monóxido de Carbono (CO) no ambiente. Dentre os sensores de gases disponíveis no mercado para sistemas embarcados, os mais conhecidos, baratos e utilizados são os da família MQ (<https://playground.arduino.cc/Main/MQGasSensors>). Segue a lista de sensores:

- MQ-2 Sensível para metano, butano, GLP, fumaça. Este sensor é sensível a gases inflamáveis combustíveis. O aquecedor usa 5V.

- MQ-3 Sensível para álcool, etanol, fumaça O aquecedor usa 5V O blog do Arduino sobre o "bafômetro" usando um MQ-3: <http://arduino.cc/blog/2010/09/23/arduino-breathalyzer-calibrating-the-mq-3-alcohol-sensor/>

O MQ303A, especificado mais abaixo, é basicamente como o MQ-3, mas usa uma tensão de aquecimento mais baixa.

- MQ-4 Sensível para metano e GNV O aquecedor usa 5V.

- MQ-5 Sensível para GLP O aquecedor usa 5V.

- MQ-6 Sensível para GLP e gás butano O aquecedor usa 5V. O MQ306A (também nesta página) é como este sensor, mas usa uma tensão de aquecedor menor.

- MQ-7 Sensível para monóxido de carbono O aquecedor usa uma tensão alternada de 5V e 1,4V. Uma biblioteca para o MQ-7: <http://thesis.jmsaavedra.com/prototypes/software/mq-7-breakout-arduino-library/> O MQ307A (também nesta página) é como este sensor, mas usa uma tensão de aquecimento mais baixa.

- MQ-8 Sensível para o gás de hidrogênio O aquecedor usa 5V.

- MQ-9 Sensível para monóxido de carbono e gases inflamáveis. O aquecedor usa uma tensão alternada de 5V e 1,5V, dependendo de como os gases usam essa tensão alternada. Se apenas o monóxido de carbono for testado, o aquecedor pode ser ajustado em 1,5V. O MQ309A (também nesta página) é como este sensor, mas usa uma tensão de aquecedor mais baixa.

- MQ131 Sensível para ozônio O aquecedor usa 6V.

O resistor de carga é de 100k a 200k, o que é muito mais alto do que para outros sensores. Este sensor também é muito sensível. Mede em ppb (partes por bilhão), sendo que outros sensores medem em ppm (partes por milhão).

- MQ135 Para a qualidade do ar. Sensível para benzeno, álcool, fumaça. O aquecedor usa 5V. Um exemplo de como usá-lo: <http://wiring.org.co/learning/basics/airqualitymq135.html>

Um exemplo com cálculo do valor de CO2: <http://davidegironi.blogspot.it/2014/01/cheap-co2-meter-using-mq135-sensor-with.html>

- MQ136 Sensível para o gás de sulfeto de hidrogênio. O aquecedor usa 5V.
- MQ137 Sensível para amônia. O aquecedor usa 5V.
- MQ138 Sensível para Benzeno, Tolueno, Álcool, Acetona, Propano, Formaldeído gasoso e Hidrogênio gasoso. O aquecedor usa 5V.
- MQ214 Sensível para metano e gás natural. O aquecedor usa 6V.
- MQ216 Sensível para gás natural e gás de carvão.
- MQ303A Sensível para álcool, etanol e fumaça (assim como o MQ-3) O aquecedor usa 0,9V. Um exemplo para o MQ302A <http://insapio.com/2011/12/27/mq3-or-mq303a-alcohol-sensor/>. Ele detecta os mesmos gases que o MQ-3, mas usa uma tensão mais baixa.
- MQ306A Sensível para GLP e gás butano. O aquecedor usa 0,9V. Ele detecta os mesmos gases que o MQ-6, mas usa uma tensão mais baixa.
- MQ307A Sensível para monóxido de carbono. O aquecedor usa uma tensão alternada de 0,2V e 0,9. Ele detecta os mesmos gases que o MQ-7, mas usa uma tensão mais baixa.
- MQ309A Sensível para monóxido de carbono e gases inflamáveis. O aquecedor usa uma tensão alternada de 0,2V e 0,9V. Depende dos gases como usar essa tensão alternada. Ele detecta os mesmos gases que o MQ-9, mas usa uma tensão mais baixa do aquecedor.
- MG811 Sensível ao dióxido de carbono (CO2). O aquecedor usa 6V. O sinal deste sensor de gás pode ser conectado ao Arduino, mas é melhor amplificar o sinal com um OpAmp (amplificador operacional). Um exemplo deste sensor (sem um OpAmp): <http://itp.nyu.edu/physcomp/sensors/Reports/MG811>
- AQ-104 Mede a qualidade do ar.
- AQ-2 Sensível para gases inflamáveis e fumaças referentes a fumos.
- AQ-3 Sensível para Álcool e Benzina.
- AQ-7 Sensível para monóxido de carbono.

De todos esses sensores, os que melhores se adequam à proposta são: MQ-7, MQ-9, MQ307A, MQ309A e MG811. Todos os outros sensores medem gases que não se aplicam à estação, por serem gases não comuns em ambientes normais. Deu-se preferência ao MQ-7 e ao MQ-9, pois, além de mais baratos, são mais fáceis de serem encontrados. Ambos os sensores têm um funcionamento de captura de gases bastante parecido, possuem uma saída analógica para leitura dos dados e possuem alimentação entre 1.4 e 5 V. Os sensores MQ-7 e MQ-9 podem ser vistos, respectivamente, nas Figuras 3.10 e 3.11.



Figura 3.10: Sensor MQ-7

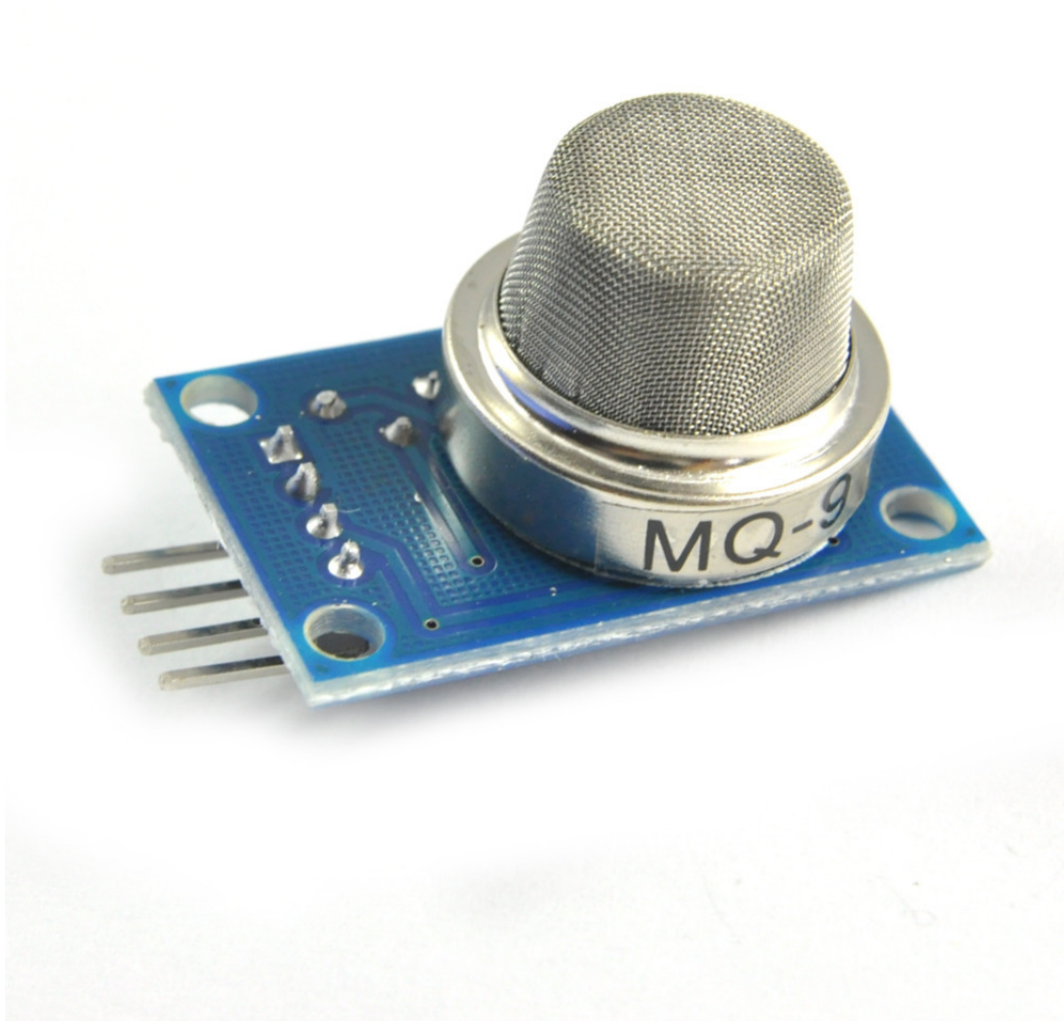


Figura 3.11: Sensor MQ-9

O sensor MQ-7 é um sensor de bastante abrangência no mercado, enquanto o MQ-9, além de ser mais caro, é encontrado com mais dificuldade. Levando esses fatores em consideração, o primeiro foi o sensor utilizado na proposta.

3.3.2 Arduino

Arduino é uma plataforma open-source simples, porém bastante completa, que engloba hardware e software. Essa plataforma é facilmente extensível pelo acoplamento de outras placas de desenvolvimento, como, por exemplo, para ZigBee (XBee shield). Um Arduino pode receber e enviar sinais analógicos e digitais usando PWM, pinos digitais e conversores A/D embutidos.

Em um dos exemplos de uso do embarcado, pode-se citar [Anbya et al. 2012], o qual, baseado na utilização de ZigBee para intercomunicação entre placas Arduino, propõe leitura e troca de informação mútua entre as placas, via uma rede *mesh*, de forma a alcançar o objetivo comum de monitoramento do uso de energia elétrica em uma casa.

Para a implementação da proposta, os primeiros testes começaram com a utilização de um Arduino UNO para a captura e transmissão dos dados. Dessa forma, os sensores eram conectados no Arduino. O Arduino, por si só, não possui conexão com a Internet. Para que se pudesse conectar o Arduino à internet de forma que a estação tivesse conexão móvel, foram feitos estudos e testes com os módulos GPRS SIM800L e A6. Para que ambos os módulos funcionassem de forma correta, era necessária uma fonte de alimentação constante, o que dificultaria a medição de uma bateria adequada para o protótipo, bem como acarretaria em um maior consumo da mesma.

Dessa forma, foram feitas pesquisas referentes a outros meios de comunicação, bem como outros dispositivos embarcados capazes de capturar e transmitir dados.

3.3.3 Raspberry

Assim como o Arduino, o Raspberry PI é um embarcado bastante utilizado em aplicações e sistemas IoT. Como diferencial, o Raspberry possui conexão Ethernet e mais processamento. Para essa proposta, a utilização de um Raspberry PI se mostrou inviável por vários motivos. Primeiramente, além de ocupar espaço, assim como o Arduino, o foco da proposta é que seja economicamente viável. Dispositivos embarcados como estes acabariam por tornar a solução mais cara. Outro ponto, é que devido ao fato de só possuir conexão Ethernet, o Raspberry também precisaria de um módulo a mais para permitir conexão sem fio, o que acarretaria em mais um aumento no consumo de Energia. Por fim, sabe-se que os dados capturados pela estação são pequenos, e não se necessita de muito processamento para a captura e transmissão dos mesmos.

Com a continuidade das pesquisas sobre outros meios de captura e transmissão dos dados dos sensores, foram avaliadas outras possibilidades que se tornaram mais viáveis, o que será descrito nos tópicos seguintes.

3.3.4 ESP 32

O ESP32 é um microcontrolador construído pela Espressif System, introduzido como o sucessor do ESP8266 [Zhangjiang High-Tech Park Pudong 2018]. Além de ser de baixo custo, possui a funcionalidade de economizar energia e é composto de um microcontrolador integrado com Wi-Fi e Bluetooth, o que o torna bastante versátil em aplicações móveis e IoT. Por funcionar em grandes variações de temperatura de operação, têm se tornado uma opção bastante confiável no ambiente industrial. O microcontrolador pode funcionar tanto como um sistema completo *standalone*, quanto como um dispositivo escravo de outro microcontrolador [Biswas and Iqbal 2018]. O fato de já ter integrado Wi-Fi e Bluetooth facilita bastante o desenvolvimento do módulo em várias aplicações, porém limitadas à essas tecnologias. Dessa forma, para integrações mais completas, é necessária a utilização de outros componentes com o ESP32. Tendo isso em vista, alguns fabricantes têm utilizado esse mi-

crocontrolador como base para desenvolver outros kits de desenvolvimento, possibilitando a agregação de novas tecnologias e aplicações. Esse é o caso do módulo WiFi LoRa 32, da Heltec, o qual será explicado mais à frente.

3.3.5 Módulo WiFi LoRa

Para que os dados da estação pudessem ser capturados e enviados, foi escolhido a utilização de um módulo de desenvolvimento WiFi LoRa 32, produzido pela empresa Heltech. Este módulo utiliza como CPU um ESP32 integrado, capaz de se conectar com outros dispositivos via protocolo LoRa, bem como se conectar à rede WiFi.

A escolha desse módulo deu-se ao fato de que, além de pequena, a estação deve ter o menor consumo de energia possível. Além disso, apenas para a captura e envio de dados, não é necessário muito processamento, o que descartou a utilização de microcontroladores como Arduino e Raspberry Pi, tendo em vista que esses dispositivos têm maior capacidade computacional, consomem mais energia, são maiores e ainda necessitariam de módulos a mais de conectividade.

Com dois módulos LoRa, é possível criar uma conexão LoRa entre a estação, que utiliza um dos módulos, e o segundo módulo, que seria utilizado como um Gateway LoRa de baixo custo.

O módulo LoRa escolhido foi o Heltec LoRa WiFi 32 de frequências 868-915 MHz. Esse módulo possui um alcance teórico de aproximadamente 3 quilômetros, dependendo das antenas utilizadas e das interferências do ambiente. A Figura 3.12 mostra kit de desenvolvimento escolhido e a Figura 3.13 as especificações do mesmo.



Figura 3.12: Kit de desenvolvimento Heltec LoRa 32

Specification:

- CPU: ESP32
 - 240 MHz dual core
 - WiFi up to 150Mbps 802.11 b/g/n/e/i
 - Bluetooth 4.2 (BLE)
- Flash: 4MB (32Mbit)
- USB-Serial Converter: CP2102
- Radio: Semtech SX1276
- Antenna connector: IPX (U.FL)
- OLED Screen:
 - Size: 0.96"
 - Driver: SSD1306
 - Resolution: 128x64 px
- Li-Ion/Li-Po charging circuit
- Battery socket: 2pin raster 1.25 mm
- Size: 52 x 25.4 x 10.3 mm

Figura 3.13: Especificações Kit de desenvolvimento Heltec LoRa 32

O módulo ainda dispõe de um display OLED de 0,96 polegadas, com uma resolução de 128x64 pixels. O fato de ser OLED implica em um baixo consumo de energia, porém, para o módulo que fica na estação, o display pode ficar desabilitado.

Componentes internos do Kit de Desenvolvimento Heltec LoRa 32 868-915MHz (<https://www.curtocircuito.com.br/placa-wifi-lora-32-esp32-lora-display-oled.html>):

ESP32

- Chip Base: ESP32-D0WDQ6;
- Processador: Xtensa 32-Bit LX6 Dual Core;
- Clock: 80 à 240 MHz (Ajustável);
- Memória ROM: 448KB;
- Memória SRAM: 520Kb;
- Memória Flash Externa: 32-Bit de acesso e 4Mb;
- Tensão de Alimentação: 2,7 à 3,6 VDC;
- Tensão de nível lógico: 3,3VDC (não tolera 5V);
- Corrente de consumo: 80mA (típica);
- Corrente de consumo: 500mA (máxima);
- Interfaces: Cartão SD, UART(3 canais), SPI (3 canais), SDIO, I2C (2 canais), I2S (2 canais), IR, PWM LED (2 canais) e PWM motor (3 canais);

- GPIO: Digital IO (36), ADC 12-Bits (18 canais), DAC 8-Bits (2 canais), Sensor Capacitivo (10 canais); LNA pré-amplificador;

- Wi-Fi 802.11 b/g/n: 2.4 à 2.5 GHz;
- Segurança do Wi-Fi: WPA / WPA2 / WPA2-Enterprise / WPS;
- Criptografia Wi-Fi: AES / RSA / ECC / SHA;
- Bluetooth 4.2 BR / EDR e BLE (Bluetooth Low Energy);
- RTC Integrado de 8Kb (Slow / Fast);
- Sensor integrado: Temperatura e Hall;
- Temperatura de trabalho: -40 à +85 graus Celsius;

LoRa SX1276

- Chip Base: SX1276;
- Frequência de operação: 868-915MHz;
- Sensibilidade: -139dBm (máximo);
- Tensão de Alimentação: 1,8 à 3,7 VDC;
- Corrente de consumo: 10,8mA (recepção – RX) e 120mA (transmissão – TX);
- Potência máxima: 20dBm;
- Modo de modulação: LoRa, FSK, GFSK e OOK;
- Taxa de transferência: 0,018 à 37,5 Kbps (LoRa);
- Interface de comunicação: SPI;
- Temperatura de trabalho: -40 à +85 graus Celsius;
- Dimensões: 17,0 x 17,3 x 2,6 mm (sem antena);

Display OLED

- Tensão de trabalho: 3,0 à 5,0V;
- Tensão lógica: 3,3 - 5,0V;
- Potência Máxima: 80 mW;
- Resolução: 128x64 pixels;
- Ângulo de Visão: > 160 Graus;
- Interface: I2C;
- Escrita Azul;

- Driver: SSD1306;
- Dimensão do visor: 1,90 x 2,73 cm (C x L);
- Dimensão total: 2,78 x 2,73 x 0,41 cm (C x L x A);

Dois módulos desse foram utilizados, de forma a criar uma conexão LoRa. Como a configuração de transmissor e/ou receptor é feita em código, via programação do chip CP2102 da Silicon LAB (responsável por fazer a conversão USB/Serial, logo, a conexão com o PC), o módulo com os sensores foi codificado para ser o transmissor. O segundo módulo foi codificado para funcionar como o receptor dos dados. Os dados recebidos pela estação são tratados de forma que o módulo identifique os dados referentes à cada medição. Após esse tratamento, o módulo se conecta com o Wi-Fi do laboratório UIoT e envia os dados nos padrões que o gateway do UIoT espera receber.

3.4 Alimentação

Para que a Estação seja autônoma em termos de energia, é necessário que ela seja capaz de receber alimentação via placa solar e armazenar essa energia em uma bateria. A bateria deve ser capaz de suprir a estação durante um tempo considerável sem sol. Para isso, foram medidos dados de consumo de energia da estação, e mensurados capacidade do painel solar e da bateria.

A Tabela 3.1 mostra o consumo de energia do circuito da estação, circuito este composto por quatro sensores e um kit desenvolvimento LoRa. Este consumo foi medido durante todo o processo de captura e envio dos dados da Estação, o que dura menos de um minuto. O circuito de alimentação foi projetado de forma que, a cada 10 minutos, a estação ligue e passe 2 minutos transmitindo dados, para depois desligar novamente.

Consumo da Estação		
Tempo ligado por hora	12	minutos
Média de consumo por hora	0,15	Wh
Consumo diário máximo	3,6	Wh
Fator de segurança	25	%
Consumo diário considerado	4,8	Wh

Tabela 3.1: Consumo estimado da Estação

Nas Tabelas 3.2 e 3.3, respectivamente, vemos a potência gerada pela placa solar e as características da bateria para uma autonomia de 2 dias.

Painel Solar		
Geração por dia	4,8	Wh
Tempo de geração máxima	4	horas
Potência do painel	1,5	Wp
Módulos	1	
Eficiência do módulo	80	%

Tabela 3.2: Painel solar

Bateria		
Autonomia	2	dias
% de descarga	85	%
Tensão	3,7	V
Corrente média	0,040541	A
Capacidade do banco	2,289348	Ah

Tabela 3.3: Especificações necessárias da bateria

Para a alimentação da bateria, foi necessário projetar um circuito capaz de receber a energia da placa solar e converter para os parâmetros da bateria. Além disso, esse circuito projetado também é capaz de alimentar o circuito da estação apenas no momento de fazer a captura e envio dos dados, de forma a economizar energia.

Para isso, foram utilizados um relé ATXD601 (https://www3.panasonic.biz/ac/e/search_num/index.jsp?c=detail&part_no=ATXD601), um regulador de voltagem L7805CV (<https://www.st.com/resource/en/datasheet/l78.pdf>), um dispositivo *step-down* para abaixar a tensão fornecida pela placa solar, um dispositivo *step-up* para aumentar a tensão fornecida pela bateria, e um microcontrolador ATtiny85 (http://ww1.microchip.com/downloads/en/devicedoc/atmel-2586-avr-8-bit-microcontroller-attiny25-attiny45-attiny85_datasheet.pdf), que é o responsável por ligar e desligar a estação no tempo desejado. Este último, além de ser bastante pequeno, quase não consome energia. O dispositivo foi programado de forma a alimentar a Estação durante 30 segundos, cortar a alimentação nos próximos 10 minutos, alimentar novamente por mais 30 segundos, e assim por diante.

Listagem 3.1: Código do ATtiny85

```

1 void setup() {
2   // Inicializa a porta 4 como SAIDA.
3   pinMode(4, OUTPUT);
4 }
5
6 // a funcao LOOP fica sempre rodando
7 void loop() {
8   digitalWrite(4, HIGH); // Ativa o circuito

```

```
9   delay(30000);           // espera 30 segundos
10  digitalWrite(4, LOW);   // Desativa o circuito
11  delay(600000);          // Espera 10 minutos
12 }
```

Capítulo 4

Resultados Experimentais

4.1 Introdução

Feita a implementação da proposta, foram obtidos os resultados experimentais subsequentes. Esses resultados têm como base experimentos feitos em laboratório e em campo. Neste capítulo, são descritos os resultados de potência de sinal obtidos em pontos específicos ao redor da Faculdade de Tecnologia, bem como resultados de transmissões de dados e de autonomia da bateria.

4.2 Site Survey com a distribuição da potência do sinal em volta da Faculdade de Tecnologia

Como o Gateway LoRa se comunica diretamente com o Gateway UIoT através da rede UIoT, foi necessário manter o primeiro dentro do Laboratório, até mesmo por questões de segurança e manutenção. Dessa forma, foram analisados os pontos em volta do laboratório onde existisse uma boa relação entre distância, potência de transmissão, bem como densidade espacial de pessoas. Percebeu-se que, quanto menores as interferências, como árvores e prédios, melhores eram os alcances conseguidos. Além disso, verificou-se que a potência de transmissão máxima, sem perda de sinal, entre os dois dispositivos LoRa (Estação e Gateway LoRa) é de 121dB.

A Figura 4.1 mostra, em um mapa geográfico, a distribuição do sinal em alguns pontos. Esses pontos foram escolhidos visando maior alcance entre a Estação e o Gateway LoRa, de forma que, em linha reta, houvesse a menor interferência do ambiente. Deve ser levado em consideração que, além de o módulo receptor estar alocado dentro do laboratório, a Estação estava em uma altura relativamente baixa durante os testes, o que significa uma maior interferência. Em pontos mais altos, com menos interferência, os alcances teóricos são maiores. Em cada um desses pontos, a potência alcançada antes de perder conexão foi

de -121dB.



Figura 4.1: Mapa com a distância de transmissão de cada ponto

Dentro da Faculdade de Tecnologia, a potência média do sinal variou entre -70 e -90dB. Isso se deve ao fato de que estruturas metálicas são bastante influentes em efeitos como refração do sinal. Além disso, a Faculdade de Tecnologia é um ambiente repleto de antenas e equipamentos, o que influencia em interferências de sinal.

Esses valores foram obtidos utilizando as antenas próprias do Kit de Desenvolvimento LoRa 32 nas estações, e uma antena mais apropriada para recepção de sinal (<https://www.steelbras.com.br/wp-content/uploads/2015/02/AP3900.pdf>), instalada no Gateway LoRa. Com essa antena, o ganho foi de aproximadamente 5dB, bem como mais estabilidade da conexão. A Figura 4.2 mostra as especificações da antena utilizada na recepção dos dados.

ESPECIFICAÇÕES:

Frequência.....: 900 MHz
Potência..... : 100 Watts
Impedância.....: 50 Ohms
R.O.E.....: Menor que 1,5:1
Ganho..... : 5,15 dBi
Altura..... : 320mm
Peso.....: 62g

EXEMPLO: Gráfico da uma Antena AP3900.

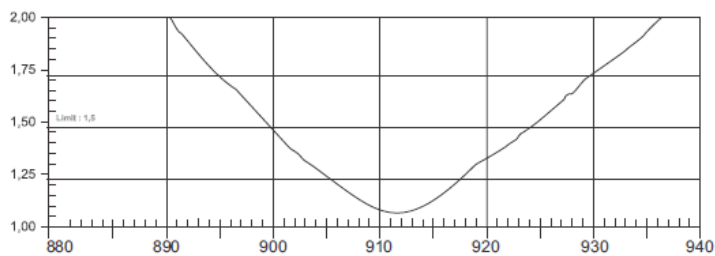


Figura 4.2: Antena utilizada na recepção dos dados

Com base nas especificações e no gráfico de frequências da antena, resolveu-se trabalhar com os módulos na frequência de 915MHz, ao invés de 868MHz.

4.3 Protótipo da Estação

Foi desenvolvido um protótipo da Estação, que será utilizado para a apresentação do projeto final. Na Figura 4.3, vê-se o circuito da Estação acoplado ao de alimentação, bem como a bateria e os sensores utilizados.



Figura 4.3: Protótipo da Estação

O circuito da Estação, visto na Figura 4.4, foi projetado de forma que os pinos JP2 são conectados nos pinos CIRCUIT do circuito de alimentação, o qual é mostrado na Figura 4.5. Nela também pode ser visto o microcontrolador ATtiny85, responsável por ligar a Estação no momento da captura de dados, e desligar para economizar energia.

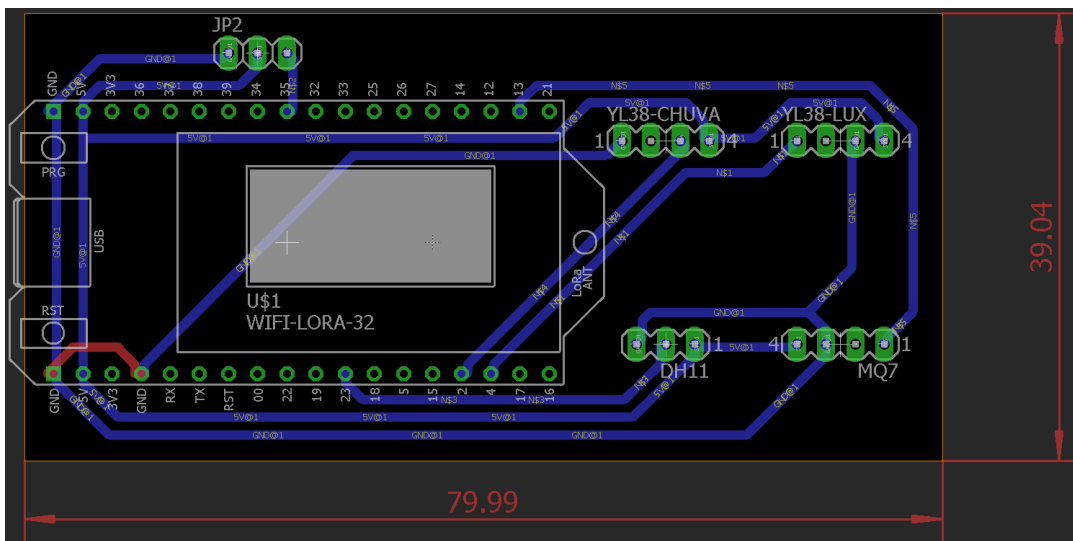


Figura 4.4: Circuito da Estação

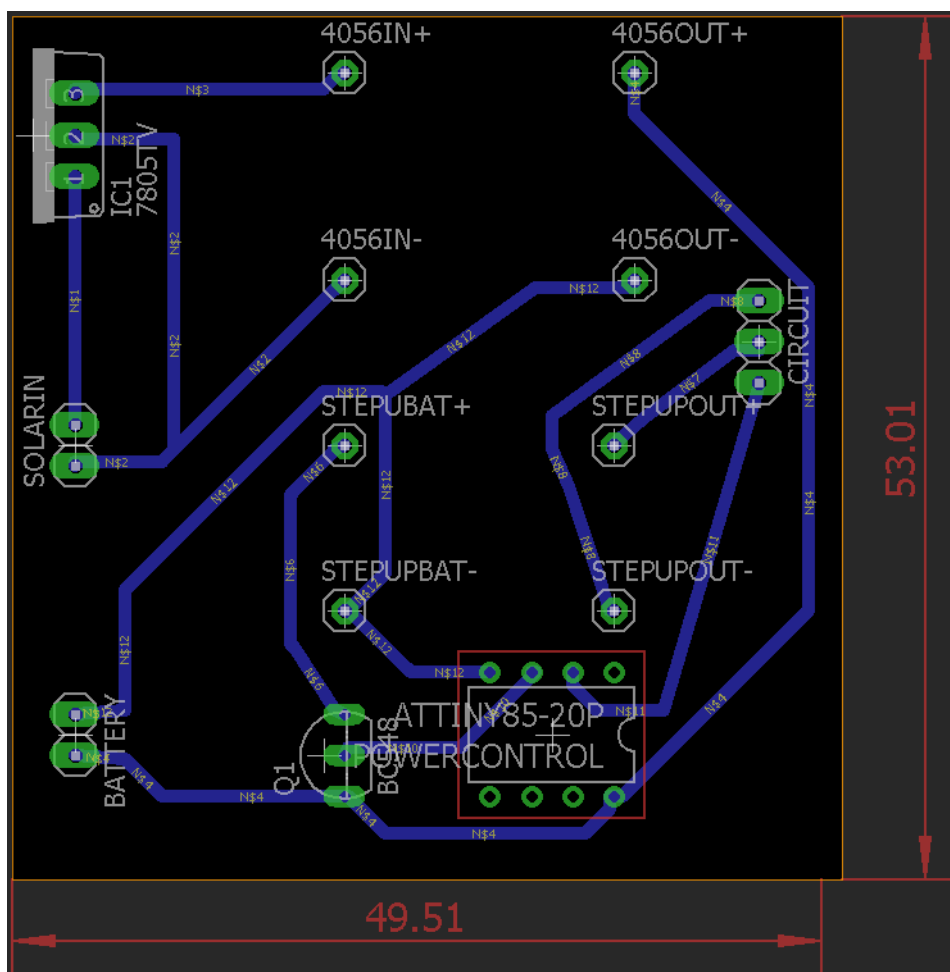


Figura 4.5: Circuito de alimentação

De forma a abstrair os circuitos utilizados, a Figura 4.6 mostra um diagrama com os principais componentes do projeto da Estação.

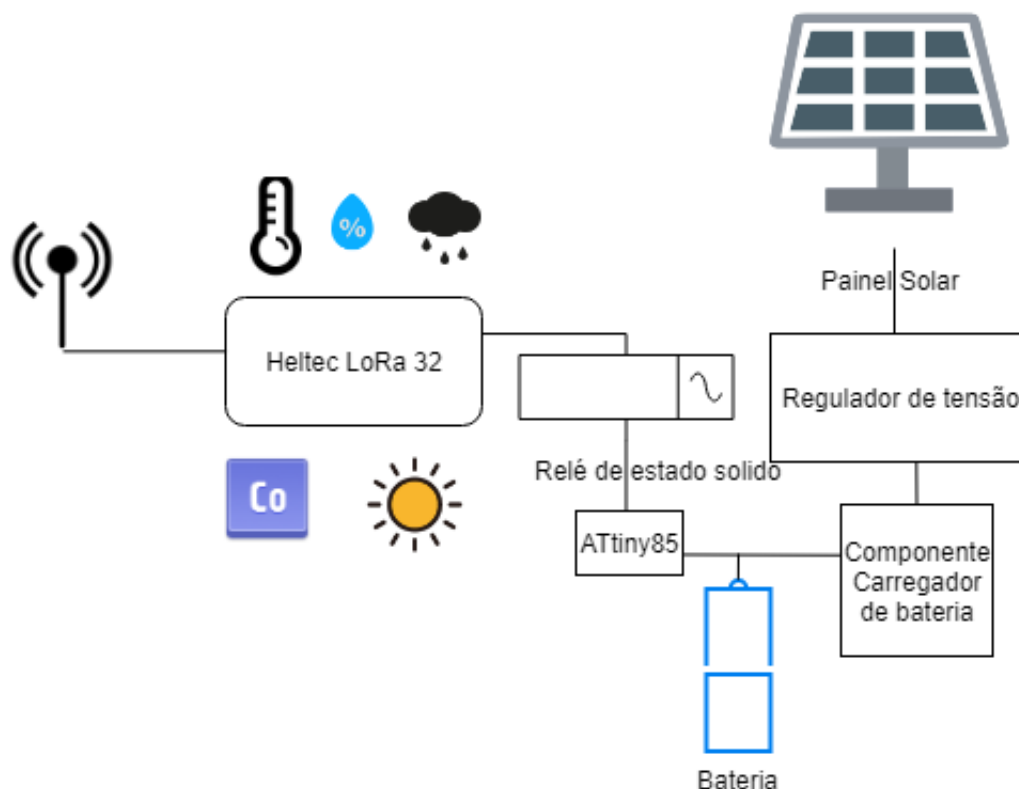


Figura 4.6: Diagrama do Protótipo

4.4 Dados disponibilizados

Após serem enviados ao Gateway LoRa, os dados são tratados e registrados no Gateway UIoT. Ao identificar, através de um *Index*, de qual estação recebeu os dados, o Gateway LoRa cria um "cliente" no Gateway UIoT referente à estação, onde esse cliente recebe um MAC e um CHIPSET. Com o cliente criado, é possível registrar até cinco serviços no Gateway, onde cada serviço é referente a um tipo de dado. Devido ao fato de possuir um display OLED, o dispositivo LoRa utilizado como o primeiro gateway permite visualizar em tempo real o Index da Estação, os dados, o MAC e o CHIPSET recebidos pelo cliente da Estação, bem como a potência do sinal da transmissão.

A Figura 4.7 mostra o Gateway LoRa recebendo os dados da Estação 1, de forma que o "cliente" criado pelo gateway para esta estação receba um MAC e um CHIPSET, com o intuito de se identificar o dispositivo no Gateway UIoT, bem como a potência do sinal de transmissão.



Figura 4.7: Gateway LoRa mostrando MAC e CHIPSET da Estação 1

Os dados do Gateway LoRa, após passarem pelo Gateway UIoT, chegam no Middleware através de uma conexão REST/JSON. Esses dados podem ser visualizados através de um link que contém o CHIPSET do dispositivo, por exemplo: <http://172.16.9.69:5000/list/data?chipset=CH:IP:SE:T1:1E:1F>. A Figura mostra o padrão em que esses dados são disponibilizados, de forma que outras aplicações possam acessá-los.

```

← → ↻ ⓘ Não seguro | 172.16.9.69:5000/list/data?chipset=CH:IP:SE:T1:1E:1F
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 4, "time": "2018-12-06T17:09:30.528889", "values": [224.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 3, "time": "2018-12-06T17:09:30.423601", "values": [4095.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 2, "time": "2018-12-06T17:09:30.312075", "values": [72.99]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 1, "time": "2018-12-06T17:09:25.996934", "values": [41.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 0, "time": "2018-12-06T17:09:25.676760", "values": [23.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 4, "time": "2018-12-06T17:09:22.345905", "values": [224.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 3, "time": "2018-12-06T17:09:22.242098", "values": [4095.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 2, "time": "2018-12-06T17:09:22.133960", "values": [72.89]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 1, "time": "2018-12-06T17:09:21.854736", "values": [41.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 0, "time": "2018-12-06T17:09:21.518610", "values": [23.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 4, "time": "2018-12-06T17:09:18.385573", "values": [235.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 3, "time": "2018-12-06T17:09:18.280939", "values": [4095.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 2, "time": "2018-12-06T17:09:18.126897", "values": [72.75]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 1, "time": "2018-12-06T17:09:18.027524", "values": [41.0]},
{"chipset": "CH:IP:SE:T1:1E:1F", "mac": "9A:49:4F:54:1E:1F", "sensitive": 0, "serviceNumber": 0, "time": "2018-12-06T17:09:17.796038", "values": [23.0]},

```

Figura 4.8: Dados recebidos pelo Middleware

4.4.1 Tamanho e frequência dos dados transmitidos

Os dados são enviados no seguinte formato:

Index-Temperatura;Umidade;Luminosidade;Chuva;CO

Index é o número da estação, e os parâmetros seguintes são os dados, do tipo *float*, referentes à cada tipo de dado meteorológico medido. Para o envio dos dados, a estação empacota em uma *string* seu próprio Index e os dados capturados. Um exemplo de dados enviados, é a seguinte *string*:

1-25.00;40.00;3125.00;4095.00;2925.00

Essa sequência possui 37 caracteres, sendo que cada um deles tem o tamanho de 1 Byte. Isso significa que, após cada captura, a estação envia 296 bits de dados.

As Estações foram programadas para capturar os dados, empacotá-los e fazer o envio, a cada 1 segundo. Feito esse processo, é enviado um sinal para o circuito alimentador, de forma que a Estação perca a alimentação. Com esse processo, é possível economizar energia e enviar uma menor quantidade de dados para o gateway.

4.5 Captura, envio e recebimento de dados

Para o processo de captura e envio de dados da Estação, bem como para o recebimento e tratamento desses dados pelo o Gateway LoRa, foram necessárias programações diferentes para os dispositivos. Além disso, para que o Gateway LoRa enviasse esses dados para o Gateway UIoT nos parâmetros corretos, foi necessária a utilização de uma biblioteca que permitisse essa comunicação.

No dispositivo LoRa da Estação, a programação básica utilizada para a captura e envio dos dados foi a seguinte:

Listagem 4.1: Código da Estação LoRa

```
1
2 void setup () {
3
4     // configuracoes da modulacao LoRa. Deve ser a mesma do receptor ,
5     // caso contrario nao se comunicam
6     LoRa.enableCrc ();
7     LoRa.setCodingRate4 (5);
8     LoRa.setSpreadingFactor (12);
9     LoRa.setSignalBandwidth (125000);
10    LoRa.setPreambleLength (8);
11    LoRa.setSyncWord (0x34);
12 }
13 void loop () {
```

```

14
15     ldrValue = analogRead(ldrPin);
16     rainValue = analogRead(rainPin);
17     mqValue = analogRead(mqAnalog);
18
19     ldrValueAdjusted = 100-ldrValue*0.0244;
20
21     String mq = String(mqValue);
22
23     String ldr = String(ldrValueAdjusted);
24     String rain = String(rainValue);
25
26     String dht = String(dht.readTemperature());
27     String dhth = String(dht.readHumidity());
28
29     delay(500);
30
31     // formato em que os pacotes sao enviados
32     measures = "1-" + dht + ";" + dhth
33             + ";" + ldr + ";" + rain + ";" + mq;
34
35     LoRa.beginPacket();           // inicia o pacote
36     LoRa.println(measures);      // envia o pacote
37     LoRa.endPacket();           // encerra o pacote
38
39 }

```

Já no Gateway LoRa, os dados recebidos são desempacotados e enviados para o Gateway UIoT via Wi-Fi. Para isso, foi necessária a importação de uma biblioteca de comunicação com o Gateway UIoT. Dessa forma, foi necessário executar os seguintes passos para o registro do dispositivo, dos serviços e envio dos dados.

Listagem 4.2: Código do Gateway LoRa

```

1
2 #include "UHttp_esp8266.h"
3 UHttp_esp8266 cliente;
4
5 // servicos referentes aos dados
6 int service0 = 0;
7 int service1 = 0;
8 int service2 = 0;
9 int service3 = 0;
10 int service4 = 0;
11 int service5 = 0;
12
13 void setup () {
14
15     // configuracoes da modulacao LoRa. Deve ser a mesma do
16     // transmissor, caso contrario nao se comunicam

```

```

16     LoRa.enableCrc();
17     LoRa.setCodingRate4(5);
18     LoRa.setSpreadingFactor(12);
19     LoRa.setSignalBandwidth(125000);
20     LoRa.setPreambleLength(8);
21     LoRa.setSyncWord(0x34);
22
23 // mac e chipset atribuidos em código para o dispositivo, de forma a
    identifica-lo
24     String mac1 = "9A:49:4F:54:1E:1F";
25     String chipset1 = "CH:IP:SE:T1:1E:1F";
26     cliente.init(mac1, chipset1); // conecta na rede Wi-Fi do UIoT e
    registra o dispositivo (cliente) no gateway UIoT
27
28 // cria os serviços do cliente
29     service0 = cliente.create_service(0, "getTemp50", "*C", true, "
    Temperatura");
30     service1 = cliente.create_service(1, "getHum50", "%", true, "
    Umidade");
31     service2 = cliente.create_service(2, "getLight50", "Lux", true, "
    Luminosidade");
32     service3 = cliente.create_service(3, "getRain50", "mm", true, "
    Chuva");
33     service4 = cliente.create_service(4, "getMQ50", "CO2", true, "CO2
    ");
34
35 }
36
37 void loop () {
38     int packetSize = LoRa.parsePacket(); // procura pacote
39     // caso pacote exista
40     if (packetSize) {
41         String data = "";
42         while (LoRa.available()) {
43             data = LoRa.readString(); // lê pacote
44
45             // descompacta pacote
46             int firstComma = data.indexOf(",");
47             int secondComma = data.indexOf(",", firstComma + 1);
48             int thirdComma = data.indexOf(",", secondComma + 1);
49             int fourthComma = data.indexOf(",", thirdComma + 1);
50
51             String firstValue = data.substring(2, firstComma);
52             String secondValue = data.substring(firstComma + 1,
    secondComma);
53             String thirdValue = data.substring(secondComma + 1,
    thirdComma);
54             String fourthValue = data.substring(thirdComma + 1,
    fourthComma);

```

```

55     String fifthValue = data.substring(fourthComma + 1);
56
57     float tempValue = firstValue.toFloat();
58     float humValue = secondValue.toFloat();
59     float ldrValue = thirdValue.toFloat();
60     float rainValue = fourthValue.toFloat();
61     float mqValue = fifthValue.toFloat();
62
63     char result[10];
64
65     // passa os valores dos dados pro tipo que o Gateway UIoT
        recebe
66
67     char *dTemp = dtostrf(tempValue, 6, 2, result);
68     char *dHum = dtostrf(humValue, 6, 2, result);
69     char *dLdr = dtostrf(ldrValue, 6, 2, result);
70     char *dRain = dtostrf(rainValue, 6, 2, result);
71     char *dMq = dtostrf(mqValue, 6, 2, result);
72
73     // registra os dados no gateway UIoT
74     cliente.send_data(service0, dTemp, 0);
75     cliente.send_data(service1, dHum, 0);
76     cliente.send_data(service2, dLdr, 0);
77     cliente.send_data(service3, dRain, 0);
78     cliente.send_data(service4, dMq, 0);
79 }
80 }
81 }

```

4.6 Autonomia

Para o armazenamento da energia da estação, foi utilizada uma bateria de 2400mAh, o que atende bem os aproximadamente 2300mAh calculados. É sabido que a estação consome aproximadamente 240mA em seu funcionamento normal, ou seja, capturando e enviando os dados para o Gateway LoRa. Com o intuito de medir na prática a duração da bateria, foi feito um teste de *stress*, onde a Estação, em trabalho total, ficou alimentada apenas pela bateria. Nesse cenário, a bateria conseguiu alimentar a Estação por aproximadamente 20 horas seguidas. O circuito de alimentação, que é responsável por fornecer a energia da placa solar para o circuito da estação e para a bateria, também é capaz de desligar a alimentação da Estação após a transmissão dos dados para o Gateway LoRa. Com essa configuração de desligamento possibilitada pelo ATtiny85, conseguiu-se uma duração de aproximadamente 40 horas sem alimentação, enviando dados durante 30 segundos, a cada 10 minutos de desligamento.

Capítulo 5

Conclusão e Trabalhos Futuros

5.1 Conclusão

Neste trabalho foi apresentada uma solução para capturar dados meteorológicos através de uma estação meteorológica autônoma, explorando ambientes, conceitos e dispositivos de Internet das Coisas. É sabido que, hoje em dia, é de suma importância para várias aplicações a existência de informações que descrevam o ambiente.

Essas informações podem ser usadas para diversos fins, de forma que o ambiente possa ser monitorado remotamente. No projeto da estação, foi possível medir dados como temperatura, umidade, precipitação de chuva, luminosidade e qualidade do ar, através da medição dos níveis de Monóxido de Carbono. Para isso, foram necessárias pesquisas em dispositivos IoT capazes de capturar, processar e transmitir esses dados, bem como elaborar uma arquitetura de comunicação entre a estação e o laboratório UIoT, o qual está preparado para receber dados e informações de dispositivos externos.

5.2 Trabalhos futuros

Para trabalhos futuros, pretende-se implementar uma camada de segurança na transmissão dos dados capturados, bem como conseguir maiores alcances entre dispositivos LoRa.

Como foi utilizada apenas a modulação LoRa, métodos de segurança devem ser implementados na camada física, na transmissão entre a Estação e o Gateway LoRa. Propõe-se uma criptografia de chave simétrica AES 128, de forma que, com a mesma chave, a Estação e o Gateway LoRa podem trocar dados de forma segura. Também propõe-se utilizar o método de criptografia HMAC (Hash-based Message Authentication Code), o qual utiliza uma função hash criptográfica em combinação com uma chave secreta, para calcular o código de autenticação de mensagem (Message Authentication Code – MAC).

Uma outra melhoria, seria utilizar a rede LoRaWAN para fazer a transmissão de dados, pois a mesma já possui criptografia e funcionalidades de segurança [LoRa Alliance 2015b]. Essa segunda solução requererá um Gateway LoRaWAN, que é um dispositivo mais caro e robusto, o que tornará a solução ainda mais completa e capaz de receber dados de uma quantidade maior de dispositivos.

Com o intuito de permitir maior conectividade, propõe-se também implementações da Estação utilizando Redes móveis para o envio de dados meteorológicos. Com redes celulares, consegue-se medir dados de locais mais distantes e remotos, sem a necessidade de outros dispositivos no intermédio da transmissão.

Além disso, pretende-se também criar as aplicações responsáveis por fazer e disponibilizar inferências sobre os dados coletados. Uma delas, seria um aplicativo Android, que não apenas mostre os dados do local onde dada Estação está instalada, mas também dê avisos sobre a situação do tempo no ambiente.

Referências Bibliográficas

- [3GPP a] 3GPP. Release 13. <http://www.3gpp.org/release-13>.
- [3GPP b] 3GPP. release 14. <http://www.3gpp.org/release-14>.
- [3GPP c] 3GPP. Standards for the iot. http://www.3gpp.org/news-events/3gpp-news/1805-iot_r14.
- [3GPP 2017] 3GPP (2017). *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description V13.7.0, TS 36.300*.
- [3GPP TR 45.820 2015] 3GPP TR 45.820 (2015). *Cellular System Support for Ultra Low Complexity and Low Throughput Internet of Things, V2.1.0*.
- [3GPP2 Technical Specifications v1.0 2002] 3GPP2 Technical Specifications v1.0 (2002). 3gpp2 technical specifications, wireless ip network standard. *Third Generation Partnership Project 2 (3GPP2)*.
- [3GPP2 Technical Specifications v6.4.0 2004] 3GPP2 Technical Specifications v6.4.0 (2004). 3gpp2 technical specifications, general packet radio service (gprs). *Third Generation Partnership Project*.
- [802.11 1999] 802.11, I. (1999). Wireless lan medium access control (mac) and physical layer (phy) specification. *IEEE, New York*.
- [Abramson 2017] Abramson, N. (2017). The aloha system: another alternative for computer communications. in *Proceedings of the November 17-19, 1970, fall joint computer conference*. ACM, pages 281—285.
- [Adame et al. 2014] Adame, T., Bel, A., Bellalta, B., Barcelo, J., and Oliver, M. (2014). Ieee 802.11ah: the wifi approach for m2m communications. *Wireless Communications, IEEE, vol. 21, no. 6*, pages 144—152.
- [Akeela and Elziq 2017] Akeela, R. and Elziq, Y. (2017). Design and verification of ieee 802.11ah for iot and m2m applications. *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*.

- [Akildiza 2002] Akildiza, I. F. (2002). A survey on sensor networks. *Communications magazine, IEEE*, v. 40, n. 8, pages 102—114.
- [Anand and Routray 2017] Anand, S. and Routray, S. K. (2017). Issues and challenges in healthcare narrowband IoT. *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*.
- [Anbya et al. 2012] Anbya, M., Salehuddin, M., Hadisupadmo, S., and Leksono, E. (2012). Wireless sensor network for single phase electricity monitoring system via Zigbee protocol. *2012 IEEE Conference on Control, Systems and Industrial Informatics (ICCSII)*, 23–26 September, pages 261—266.
- [Andreevy et al.] Andreevy, S., Galinina, O., Pyattaev, A., Gerasimenko, M., Tirronen, T., Torsner, J., Sachs, J., Dohler, M., and Koucheryavy, Y. Understanding the IoT Connectivity Landscape – A Contemporary M2M Radio Technology Roadmap.
- [AOSONG] AOSONG. *Temperature and humidity module DHT11 Product Manual*, <https://akizukidenshi.com/download/ds/aosong/DHT11.pdf>, pages 1–8.
- [Astely et al. 2013] Astely, D., Dahlman, E., Fodor, G., Parkvall, S., and Sachs, J. (2013). “LTE Release 12 and beyond”. *IEEE Communications Magazine*, vol. 51, no. 7, pages 154—160.
- [Atzori et al. 2010] Atzori, L., Iera, A., and Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, www.elsevier.com/locate/comnet.
- [Atzori et al. 2012] Atzori, L., Iera, A., Morabito, G., and Nitti, M. (2012). “The social internet of things (SIoT)-when social networks meet the internet of things: Concept, architecture and network characterization”. *Comput. Netw.*, vol. 56, no. 16, pages 3594—3608.
- [Baldauf et al. 2007] Baldauf, M., Dustdar, S., and Rosenberg, F. (2007). A Survey on Context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing*.
- [Bellavista et al. 2013] Bellavista, P., Corradi, A., Fanelli, M., and Foschini, L. (2013). A Survey of Context Data Distribution for Mobile Ubiquitous Systems. *ACM Computing Surveys*.
- [Bharathidasan 2002] Bharathidasan, A.; PONDURU, V. A. S. (2002). Sensor networks: An overview. paper. *University of California, Davis, CA*.
- [Biswas and Iqbal 2018] Biswas, S. B. and Iqbal, M. T. (2018). "solar water pumping system control using a low cost esp32 microcontroller". *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*.

- [Bolchini et al. 2009] Bolchini, C., Curino, C. A., Orsi, G., Quintarelli, E., Rossato, R., Schreiber, F. A., and Tanca, L. (2009). And What Can Context do for Data? *Communications of the ACM*, pages 136–140.
- [Bor et al. 2016] Bor, M., Vidler, J., and Roedig, U. (2016). LoRa for the Internet of Things. *Department of Computing and Communications Lancaster University*, pages 361—366.
- [Buddhikot et al. 2003] Buddhikot, M., Chandranmenon, G., Han, S., Lee, Y. W., Miller, S., and Salgarelli, L. (2003). Integration of 802.11 and third generation wireless data networks. *IEEE INFOCOM 2003 vol. 1*, pages 503—512.
- [Chen and Varshney 2004] Chen, D. and Varshney, P. (2004). QoS support in wireless sensor networks: a survey. *Proceedings of International Conference on Wireless Networks*.
- [Chen and Kotz 2000] Chen, G. and Kotz, D. (2000). A survey of context-aware mobile computing research.
- [Chen et al. 2011] Chen, H., Jia, X., and Li, H. (2011). “a brief introduction to iot gateway”. *IET International Conference on Communication Technology and Application (ICCTA 2011)*.
- [Coulouris et al. 2007] Coulouris, G., Dollimore, J., and Kindberg, T. (2007). Sistemas distribuídos-conceitos e projeto.
- [D. Singh 2004] D. Singh, G. Tripathi, A. J. J. (2004). A survey of Internet-of-Things: Future Vision, Architecture, Challenges and Services. *IEEE World Forum on Internet of Things (WF-IoT)*.
- [de Caldas Filho et al. 2017] de Caldas Filho, F. L., e Martins, L. M. C., Araújo, I. P., de Mendonça, F. L. L., da Costa, J. P. C. L., and de Sousa Júnior, R. T. (2017). “design and evaluation of a semantic gateway prototype for iot networks”. *UCC Companion, UCIoT 2017 Workshop Presentation, Austin, Texas, USA*.
- [Devalal and Karthikeyan 2018] Devalal, S. and Karthikeyan, A. (2018). LoRa technology-an overview. *Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018) IEEE Conference Record*, pages 1—7.
- [Dey and Abowd 2000] Dey, A. K. and Abowd, G. D. (2000). Towards a Better Understanding of Context and Context-Awareness. *In Proceedings of the Workshop on The What, Who, Where, When, and How of Context-Awareness within*, pages 1–12.
- [DHT11] DHT11. *Temperature and Humidity Sensor*, <https://components101.com/dht11-temperature-sensor>.
- [EPCglobal 2009] EPCglobal (2009). The EPCglobal Architecture Framework, EPCglobal Final Version 1.3, Approved 19 March 2009, <www.epcglobalinc.org> . .

- [Eschenauer and Gligor 2002] Eschenauer, L. and Gligor, V. (2002). A key-management scheme for distributed sensor networks. *Proceedings of the Ninth ACM Conference on Computer and Communications Security*.
- [Estrin 1999] Estrin, D. (1999). Next century challenges: Scalable coordination in sensor networks. *ACM. Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking. [S.l.]*, pages 263—270.
- [Fan et al. 2011] Fan, C., Wen, Z., Wang, F., and Wu, Y. (2011). A middleware of Internet of Things (IoT) based on ZigBee and RFID. *IET International Conference on Communication Technology and Application (ICCTA 2011)*.
- [FasLab 2013] FasLab (2013). Temperature & Humidity Data: Traditional Monitoring & Data Collection Methods. <http://www.faslab.it/pdf/networked-data-loggers-viewlinwhitepaper.pdf>.
- [Gama et al. 2012] Gama, K., Touseau, L., and Donsez, D. (2012). “Combining heterogeneous service technologies for building an internet of things middleware”. *Comput. Commun., vol. 35, no. 4*, pages 405—417.
- [Gawas 2015] Gawas, M. A. U. (2015). An overview on evolution of mobile wireless communication networks: 1g-6g. *JRITCC, VOLUME 3 ISSUE 5*.
- [Gershenfeld et al. 2004] Gershenfeld, N., Krikorian, R., and Cohen, D. (2004). The Internet of Things. *Scientific American Magazine*, pages 76–81.
- [Gubbi et al. 2013] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems Volume 29, Issue 7*, pages 1645–1660.
- [Guinard et al. 2010] Guinard, D., Trifa, V., Karnouskos, S., Spiess, P., and Savio, D. (2010). “Interacting with the soa-based internet of things: Discovery, query, selection, and ondemand provisioning of web services”. *IEEE Trans. Serv. Comput., vol. 3, no. 3*, pages 223—235.
- [Hamida et al.] Hamida, A., Kon, F., Lago, N., Zarras, A., Athanasopoulos, D., Piliou, D., Vassiliadis, P., Georgantas, N., Issarny, V., Mathioudakis, G., Bouloukakis, G., Jarma, Y., Hachem, S., and Pathak, A. Integrated CHOReOS middleware - Enabling large-scale, QoS-aware adaptive choreographies (D3.3).
- [Han et al. 2013] Han, C., Jornet, J. M., Fadel, E., and Akyildiz, I. F. (2013). “A cross-layer communication module for the internet of things”. *Comput. Netw., vol. 57, no. 3*, pages 622—633.
- [Hollands] Hollands, R. Critical interventions into the corporate smart city. *Cambridge Journal of Regions, Economy and Society*, 8, pages 61–77.

- [I. Toma and Hench 2009] I. Toma, E. S. and Hench, G. (2009). A joint roadmap for semantic technologies and the internet of things. *in: Proceedings of the Third STI Roadmapping Workshop, Crete, Greece.*
- [IEEE a] IEEE. 802.1ac-2016/cor 1-2018 - ieee standard for local and metropolitan area networks—media access control (mac) service definition - corrigendum 1: Logical link control (llc) encapsulation ethertype. *Standard for MAC, IEEE Computer Society.*
- [IEEE b] IEEE. A study of ieee 802.11 standard for use in vehicle to infrastructure communication. <https://ieeexplore.ieee.org/document/6825174>.
- [IEEE 2006] IEEE (2006). 4g, solution for convergence? *IEEE MTT-S International Microwave Symposium Digest*, <https://ieeexplore.ieee.org/document/4015040>.
- [IEEE 2015] IEEE (2015). Draft 3.0, draft standard for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Sub 1 ghz license exempt operation. *IEEE 802.11ah Std. 1516-2000.*
- [Ilie-Zudor et al. 2011] Ilie-Zudor, E., Kemeny, Z., van Blommestein, F., Monostori, L., and van der Meulen, A. (2011). “A survey of applications and requirements of unique identification systems and RFID techniques”. *Comput. Ind., vol. 62, no. 3*, pages 227—252.
- [Introduction to single carrier FDMA 2007] Introduction to single carrier FDMA (2007). Introduction to single carrier fdma. *2007 15th European Signal Processing Conference*, <https://ieeexplore.ieee.org/document/7099187>.
- [Islam et al. 2017] Islam, B., Islam, M. T., and Nirjon, S. (2017). Feasibility of LoRa for Indoor Localization.
- [ITU-R 2016] ITU-R (2016). Minimum requirements related to technical performance for imt-2020 radio interface(s). *Technical Report M.[IMT-2020.TECH PERF REQ]*.
- [Jules 2006] Jules, A. (2006). RFID security and privacy: a research survey. *IEEE Journal & on Selected Areas in Communications*, 24(2):381–394.
- [Jun-Wei et al. 2011] Jun-Wei, H., Shouyi, Y., Leibo, L., Zhen, Z., and Shaojun, W. (2011). A Crop Monitoring System Based on Wireless Sensor Network, *Procedia Environmental Sciences*. pages 558–565.
- [K. Mikhaylov 2016] K. Mikhaylov, J. Petäjäjärvi, T. H. (2016). Analysis of the capacity and scalability of the lora wide area network technology. *Centre for Wireless Communications, Faculty of Information Technology and Electrical Engineering, University of Oulu, Finland.*

- [Kadir et al. 2016] Kadir, E. A., Siswanto, A., and Syukur, A. (2016). Performance analysis of wireless lan 802.11n standard for e-learning. *2016 4th International Conference on Information and Communication Technology (ICoICT)*.
- [Katasonov and Cochez 2012] Katasonov, A. and Cochez, M. (2012). UBIWARE Platform - RAB overview. http://users.jyu.fi/miselico/papers/ubiware/RAB_overview.pdf.
- [Katasonov et al. 2008] Katasonov, A., Kaykova, O., Khriyenko, O., Nikitin, S., and Terziyan, V. (2008). Smart semantic middleware for the internet of things. in: *Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics, Funchal, Madeira, Portugal*.
- [Kaushal et al. 2014] Kaushal, K., Kaur, T., and Kaur, J. (2014). ZigBee based Wireless Sensor Networks. *International Journal of Computer Science and Information Technologies, CTIEMT Shahpur, India*.
- [Ke et al. 2017] Ke, K., Liang, Q., Zeng, G., Lin, J., and Lee, H. (2017). A LoRa Wireless Mesh Networking Module for Campus-Scale Monitoring. *2017 16th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*.
- [Kim and Lee 2014] Kim, J. and Lee, J.-W. (2014). Openiot: An open service framework for the internet of things. in *Internet of Things (WF-IoT), 2014 IEEE World Forum on. IEEE*, pages 89–93.
- [Kingsley] Kingsley, P. Masdar: the shifting goalposts of Abu Dhabi's ambitious eco-city, Wired UK, Technology. <http://www.wired.co.uk/magazine/archive/2013/12/features/reality-hits-masdar>.
- [Kirby] Kirby, T. City design: transforming tomorrow. <http://www.guardian.co.uk/smarter-cities/transformingtomorrow>.
- [Komninos et al. 2013] Komninos, N., Pallot, M., and Schaffers, H. (2013). Special issue on smart cities and the future internet in Europe. *Journal of the Knowledge Economy*, 4, pages 119–134.
- [Kopuklu and Ertas 2013] Kopuklu, I. and Ertas, G. (2013). Development of a multichannel temperature and humidity monitoring device. *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)*.
- [Kostelnik et al. 2011] Kostelnik, P., Sarnovsk, M., and Furdik, K. (2011). The semantic middleware for networked embedded systems applied in the Internet of Things and Services domain. *Scalable Computing: Practice and Experience*, vol. 12, no. 3.
- [Kotis and Katasonov 2012] Kotis, K. and Katasonov, A. (2012). Semantic Interoperability on the Web of Things: The Smart Gateway Framework. *Sixth International Conference on Complex, Intelligent, and Software Intensive Systems*.

- [Lavric and Popa 2017] Lavric, A. and Popa, V. (2017). Internet of Things and LoRa™ Low-Power Wide- Area Networks: A Survey. *Computers, Electronics and Automation Department, Stefan cel Mare University of Suceava, Romania*, pages 1—7.
- [Lioudakis et al. 2007] Lioudakis, G., Koutsoloukas, E., N. Dellas, S. K., Prezerakos, G., Kaklamani, D., and Venieris, I. (2007). A proxy for privacy: the discreet box. *EUROCON*.
- [LoRa Alliance 2015a] LoRa Alliance (2015a). Lorawan what is it. *Technical Marketing Workgroup 1.0*, Available: https://www.loraalliance.org/portals/0/documents/whitepapers/LoRaWAN10_1.pdf.
- [LoRa Alliance 2015b] LoRa Alliance (2015b). “lorawan specification”. *LoRa Alliance, 2015, rev. 1.0*.
- [M. Yun 2010] M. Yun, B. Y. (2010). Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid. *Advances in Energy Engineering (ICAEE)*, pages 69–72.
- [Martins et al. 2017] Martins, L. M. C., de Caldas Filho, F. L., de Sousa Júnior, R. T., Gi-ozza, W. F., and da Costa, J. P. C. L. (2017). Proposta de Adoção de Microserviços em IoT. *IINCT em Segurança Cibernética - Núcleo 6, Departamento de Engenharia Elétrica, Universidade de Brasília UnB - Brasil*.
- [Miorandi et al. 2012] Miorandi, D., Sicari, S., Pellegrini, F. D., and Chlamtac, I. (2012). “Internet of things: Vision, applications and research challenges”. *Ad Hoc Netw.*, vol. 10, no. 7, pages 1497–1516.
- [Mohanty et al.] Mohanty, S. P., Choppali, U., and Koungianos, E. Everything you wanted to know about smart cities: The Internet of things is the backbone. *IEEE Consumer Electronics Magazine*, 5(3), pages 60–70.
- [N. et al.] N., S. B., Adams, N., and Want, R. Context-Aware Computing Applications. *In Proceedings of the Workshop on Mobile Computing Systems and Applications (WMCSA'94)*.
- [Ng] Ng, E. Towards planning and practical understanding of the need for meteorological and climatic information in the design of high-density cities: A case-based study of Hong Kong. *International Journal of Climatology*, 32(4).
- [ON Semiconductor] ON Semiconductor. Low offset voltage dual comparators. <https://www.onsemi.com/pub/Collateral/LM393-D.PDF>.
- [Palade et al. 2016] Palade, A., Cabrera, C., White, G., Razzaque, M. A., and Clarke, S. (2016). Middleware for Internet of Things: A quantitative evaluation in small scale. *Distributed Systems Group, SCSS, Trinity College Dublin, Dublin, Ireland*.

- [Qilin and Mintian 2010] Qilin, L. and Mintian, Z. (2010). The State of the Art in Middleware. *2010 International Forum on Information Technology and Applications*.
- [R. Acharya 2008] R. Acharya, K. A. (2008). Data integrity and intrusion detection in wireless sensor networks. *Proceedings of IEEE ICON 2008*.
- [Rappaport et al. 2015] Rappaport, T. S., Maccartney, G. R., Samimi, M. K., and Sun, S. (2015). Wideband millimeter-wave propagation measurements and channel models for future wireless communication system design. *IEEE Trans. Commun., vol. 63, no. 9*, pages 3029–3056.
- [Ratasuk et al. 2016] Ratasuk, R., Mangalvedhe, N., Xiong, Z., Robert, M., and Bhatoolaul, D. (2016). Overview of narrowband IoT in LTE Rel-13. *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*.
- [Ratasuk et al. 2017] Ratasuk, R., Mangalvedhe, N., Xiong, Z., Robert, M., and Bhatoolaul, D. (2017). Enhancements of narrowband IoT in 3GPP Rel-14 and Rel-15. *2017 IEEE Conference on Standards for Communications and Networking (CSCN)*.
- [Raza et al. 2017] Raza, U., Kulkarni, P., and Sooriyabandara, M. (2017). Low power wide area networks: An overview. *IEEE Communications Surveys Tutorials, vol. 19, no. 2*, pages 855—873.
- [Razzaque et al. 2016] Razzaque, M. A., Milojevic-Jevric, M., Palade, A., and Clarke, S. (2016). Middleware for Internet of Things: A Survey. *IEEE Internet of Things Journal (Volume: 3, Issue: 1, 2016)*, pages 70–95.
- [RFID INFSO G.2 2008] RFID INFSO G.2 (2008). Co-operation with the Working Group RFID of the ETP EPOSS, Networked Enterprise, RFID INFSO G.2 Micro and Nanosystems. *Internet of Things in 2020, Roadmap for the Future*.
- [Romero 2011] Romero, M. A. B. (2011). Correlação entre o microclima urbano e a configuração do espaço residencial de Brasília. *Fórum Patrimônio, Ambiente Construído e Patrimônio Sustentável*.
- [Sachs et al. 2014] Sachs, J., Beijar, N., Elmdahl, P., Melen, J., Militano, F., and Salmela, P. (2014). “Capillary networks – a smart way to get things connected”. *Ericsson Review, vol. 8*, pages 1–8.
- [Sakamura 2006] Sakamura, K. (2006). Challenges in the age of ubiquitous computing: a case study of T-engine – an open development platform for embedded systems. *in: Proceedings of ICSE’06, Shanghai, China*.
- [Sandhu et al. 2013] Sandhu, G. K., Mann, G. S., and Kaur, R. (2013). Benefit and security issues in wireless technologies: Wi-fi and WiMax. *International Journal of Innovative Research in Computer and Communication Engineering, Vol. 1, Issue 4*, pages 976—982.

- [Schaffers et al. 2011] Schaffers, H., Komninos, N., Pallot, M., Trousse, B., Nilsson, M., and Oliveira, A. (2011). Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation. *Domingue J. et al. (eds) The Future Internet. FIA 2011. Lecture Notes in Computer Science, vol 6656. Springer, Berlin, Heidelberg.*
- [Semtech] Semtech. <https://www.semtech.com/uploads/documents/an1200.22.pdf>.
- [Shafi et al. 2017] Shafi, M., Molisch, A. F., Smith, P. J., Haustein, T., and Zhu, P. (2017). 5g: A tutorial overview of standards, trials, challenges, deployment, and practice. *IEEE Journal on Selected Areas in Communications (Volume: 35, Issue: 6, June 2017).*
- [Shin et al. 2006] Shin, M., Ma, J., Mishra, A., and Arbaugh, W. A. (2006). Wireless network security and interworking. *Proceedings of the IEEE, 94(2)*, pages 455—466.
- [Sigfox] Sigfox. <https://www.sigfox.com/en>.
- [Sinha et al. 2017] Sinha, R. S., Wei, Y., and Hwang, S. H. (2017). A survey on LPWA technology: LoRa and NB-IoT. *ICT Express, 3(1)*, pages 14–21.
- [Sookchaiya et al. 2008] Sookchaiya, T., Monyakul, V., and Thepa, S. (2008). A Study and Development of Temperature and Relative Humidity Control System in Hospital Buildings in Thailand. <https://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1039&context=ceducom>.
- [Srivastava 2006] Srivastava, L. (2006). Pervasive, ambient, ubiquitous: the magic of radio. *European Commission Conference “From RFID to the Internet of Things”.*
- [Stojmenovic 2014] Stojmenovic, I. (2014). “Machine-to-machine communications with in-network data aggregation, processing, and actuation for large-scale cyber-physical systems”. *IEEE Internet of Things Journal, vol. 1, no. 2*, pages 122–128.
- [Strategic Research Agenda] Strategic Research Agenda. Finnish Strategic Centre for Science, Technology, and Innovation, Internet of Things Strategic Research Agenda (IoT-SRA), September 2011.
- [Tang 2009] Tang, J. (2009). The effect of environmental parameters on the survival of airborne infectious agents. *J R Soc Interface, vol. 6*, pages 737–746.
- [The LoRa Alliance] The LoRa Alliance. <https://www.lora-alliance.org/>.
- [Tomás 2016] Tomás, J. P. (2016). “operators in korea, netherlands deploy lora networks for iot”. <http://www.rcrwireless.com/20160704/carriers/operators-korea-netherlands-deploy-lora-networks-iot-tag23>.
- [Vora] Vora, L. J. Evolution of mobile generation technology: 1g to 5g and review of upcoming wireless technology 5g. *EC Department, Government Polytechnic, Ahmedabad.*

- [Vázquez 2009] Vázquez, I. (2009). Social Devices: Semantic Technology for the Internet of Things. *Week@ESI, Zamudio, Spain*.
- [Wahlster 2008] Wahlster, W. (2008). Web 3.0: Semantic Technologies for the Internet of Services and of Things. *Lecture at the 2008 Dresden Future Forum*.
- [Wien 2015] Wien, M. (2015). High efficiency video coding.
- [WiFiAlliance] WiFiAlliance. <https://www.wi-fi.org/>.
- [WMO] WMO. World Meteorological Organization. *Manual on the Global Observing System, Volume 1*.
- [Wu et al. 2013] Wu, Y., Sheng, Q. Z., and Zeadally, S. (2013). “RFID: Opportunities and challenges”. in *Next-Generation Wireless Technologies*, N. Chilamkurti, Ed. New York, NY, USA: Springer, ch. 7, pages 105—129.
- [Xiao and Rosdahl 2002] Xiao, Y. and Rosdahl, J. (2002). Throughput and delay limits of ieee 802.11. *IEEE Commun. Lett.* 6(8).
- [Xu 2011] Xu, L. (2011). “Enterprise Systems: State-of-the-art and future trends”. *IEEE Trans. Ind. Informat.*, vol. 7, no. 4, pages 630–640.
- [Xu et al. 2014] Xu, L. D., He, W., and Li, S. (2014). “internet of things in industries: A survey”. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, VOL. 10, NO. 4.
- [Yun and Yuxin 2010] Yun, M. and Yuxin, B. (2010). Research on the architecture and key technology of Internet of Things (IOT) applied on smart grid. *Computer Theory and Engineering 2010, Beijing*.
- [Zhang and Zhu 2011] Zhang, H. and Zhu, L. (2011). “Internet of things: Key technology, architecture and challenging problems”. in *Proc. 2011 IEEE Int. Conf. Comput. Sci. Autom. Eng. (CSAE), Shanghai, China, Jun. 10–12*, pages 507—512.
- [Zhang et al. 2008] Zhang, J., Wenbin, L., Ning, H., and Jiangming, K. (2008). Forest Fire Detection System Based on ZigBee Wireless Sensor Network. *Journal of Beijing Forestry University*, 2008.10, pages 122–124.
- [Zhangjiang High-Tech Park Pudong 2018] Zhangjiang High-Tech Park Pudong (2018). "esp32 technical reference manual version 2.8 espressif system #101 block 2 690 bibo road".
- [Zheng et al. 2010] Zheng, L., Ying, Y., and Wang, W. (2010). Automatic monitoring system of temperature-humidity for lab. in *Control and Decision Conference (CCDC)*, pages 4258–4261.
- [Zhou 2012] Zhou, H. (2012). The Internet of Things in the Cloud: A Middleware Perspective. *Boca Raton, FL, USA: CRC Press*.

- [Zhu et al. 2010] Zhu, Q., Wang, R., Chen, Q., Liu, Y., and Qin, W. (2010). IOT Gateway: Bridging Wireless Sensor Networks into Internet of Things. *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, pages 1—6.
- [Zimmermann et al. 2007] Zimmermann, A., Lorenz, A., and Oppermann, R. (2007). An operational definition of context. *In Proceedings of the 6th International and Interdisciplinary Conference on Modeling and using Context (CONTEXT07)*, Springer Press, pages 558–571.