

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ESPECIALIZAÇÃO EM INTERNET DAS COISAS

RODRIGO CAMARGOS BERNARDES

**TECNOLOGIAS PARA APOIAR O DESENVOLVIMENTO DE
APLICAÇÕES DE INTERNET DAS COISAS**

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA

2018

RODRIGO CAMARGOS BERNARDES

**TECNOLOGIAS PARA APOIAR O DESENVOLVIMENTO DE
APLICAÇÕES DE INTERNET DAS COISAS**

Monografia de Especialização, apresentada ao Curso de Especialização em Internet das Coisas, do Departamento Acadêmico de Eletrônica – DAELN, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Especialista.

Orientador: Prof. M.Sc. Omero Francisco Bertol

CURITIBA

2018



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Curitiba

Diretoria de Pesquisa e Pós-Graduação
Departamento Acadêmico de Eletrônica
Curso de Especialização em Internet das Coisas



TERMO DE APROVAÇÃO

TECNOLOGIAS PARA APOIAR O DESENVOLVIMENTO DE APLICAÇÕES DE INTERNET DAS COISAS

por

RODRIGO CAMARGOS BERNARDES

Esta monografia foi apresentada em 26 de Novembro de 2018 como requisito parcial para a obtenção do título de Especialista em Internet das Coisas. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. M.Sc. Omero Francisco Bertol
Orientador

Prof. M.Sc. Danillo Leal Belmonte
Membro titular

Prof. Dr. Kleber Kendy Horikawa Nabas
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

RESUMO

BERNARDES, Rodrigo Camargos. **Tecnologias para apoiar o desenvolvimento de aplicações de Internet das Coisas**. 2018. 41 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

O intuito desta monografia é apresentar um estudo sobre algumas das tecnologias usadas no desenvolvimento de aplicações em Internet das Coisas (*Internet of Things* - IoT), tendo em voga a parte de comunicação sem fio com uso do IEEE 802.15.4 e do 6LoWPAN que é um adaptador do protocolo (*Internet Protocol version 6* - IPv6) para esse tipo de rede sem fio pessoal (*Wireless Personal Area Network* - WPAN). Bem como abordar algumas das técnicas que definem os protocolos usados em transmissões sem fio no IoT e demonstrar um exemplo, com um estudo de caso que exhibe como é a arquitetura básica para uma rede de sensores sem fio com hardwares didáticos, a exemplo, Raspberry PI 3 model B, microcontrolador Texas Instruments, CC2650 SimpleLink® e aplicações em software como Java.

Palavras-chave: IEEE 80215.4. 6LoWPAN. IPv6. Rede de sensores. Internet das Coisas (IoT).

ABSTRACT

BERNARDES, Rodrigo Camargos. **Technologies to support the development of applications of Internet of Things**. 2018. 41 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

The aim of this paper is to present a study on some of the technologies used in the development of Internet of Things (IoT) applications, having in vogue the part of wireless communication using IEEE 802.15.4 and 6LoWPAN what comes to be the adapter to Internet Protocol Version 6 (IPv6) for this type of Wireless Personal Area Network (WPAN). As well to present some of the techniques that define the protocols used in wireless transmissions in IoT and demonstrate an example, with a case study that shows how is the basic architecture for a network of wireless sensors with didactic hardware, for example, Raspberry PI 3 model B, Texas Instruments microcontroller, CC2650 SimpleLink®, and software applications such as Java.

Keywords: IEEE 80215.4. 6LoWPAN. IPv6. Sensor Network. Internet of Things (IoT).

LISTA DE FIGURAS

Figura 1 - Atribuição dos canais (<i>Channel</i>) e bandas (<i>Band</i>) do IEEE 802.15.4.....	21
Figura 2 - Comparação entre o padrão 802.15.4 e o modelo ISO/OSI (parte A) e topologias possíveis numa rede 802.15.4 (parte B)	22
Figura 3 - Comunicação Multi-hop	23
Figura 4 - Frequências e bandas de operação IEEE 802.15.4.....	24
Figura 5 - Cabeçalhos (<i>header</i>) dos protocolos de internet IPv4 e IPv6	26
Figura 6 - 6LoWPAN + Gateway 6LoWPAN + IPv6	27
Figura 7 - Exemplo de encadeamento de cabeçalhos no 6LoWPAN.....	28
Figura 8 - Cabeçalho IPv6 comprimido HC1	28
Figura 9 - As camadas da RSSF e os protocolos da solução proposta	31
Figura 10 - Diagrama em blocos da solução proposta.....	32

LISTA DE SIGLAS

6LoWPAN	<i>IPv6 over Low power Wireless Personal Area Networks</i>
AMQP	<i>Advanced Message Queuing Protocol</i>
API	<i>Application Programming Interface</i>
BPSK	<i>Binary Phase Shift Keying</i>
CLP	<i>Controladores Lógicos Programáveis</i>
CoAP	<i>Constrained Application Protocol</i>
CSMA-CA	<i>Carrier-Sense Multiple Access with Collision Avoidance</i>
FFD	<i>Full Function Device</i>
HC	<i>Header Compression</i>
HTTP	<i>HyperText Transfer Protocol</i>
IAB	<i>Internet Architecture Board</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IETF	<i>Internet Engineering Task Force</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
IPv4	<i>Internet Protocol version 4</i>
IPv6	<i>Internet Protocol version 6</i>
ISO/OSI	<i>International Organization for Standardization/ Open System Interconnection</i>
LTE-A	<i>Long-Term Evolution Advanced</i>
M2M	<i>Machine to Machine</i>
MAC	<i>Media Access Control</i>
MIT	<i>Massachusetts Institute of Technology</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
MQTT-SN	<i>MQTT for Sensor Network</i>

MSA	<i>MicroService Architecture</i>
MTU	<i>Maximum Transmission Unit</i>
OS	<i>Operating System</i>
OQPSK	<i>Offset Quadrature Phase Shift Keying</i>
PAN	<i>Personal Area Network</i>
PCP	Programação e Controle de Produção
PLC	<i>Power Line Communication</i>
RFC	<i>Requests For Comments</i>
RFID	<i>Radio-Frequency Identification</i>
RPL	<i>IPv6 Routing Protocol for Low Power and Lossy Network</i>
RSSF	Rede de Sensores Sem Fio
RTD	<i>Resistance Temperature Detectors</i>
RTOS	<i>Real Time Operating System</i>
SLIP	<i>Serial Line Internet Protocol</i>
SO	Sistema Operacional
TCP	<i>Transmission Control Protocol</i>
UDP	<i>User Datagram Protocol</i>
WPAN	<i>Wireless Personal Area Network</i> , ou rede pessoal sem fio
XMPP	<i>eXtensible Messaging and Presence Protocol</i>

LISTA DE UNIDADES DE MEDIDA

Bit	<i>binary digit</i>	é a menor unidade de informação que pode ser armazenada ou transmitida, podendo assumir somente dois valores: 0 (zero) ou 1 (um).
dBm	decibel-milliwatt	é uma unidade de energia elétrica em decibéis (dB), referenciada a 1 miliwatt (mW).
GHz	GigaHertz	é uma unidade de frequência, equivalente a um bilhão de Hertz.
Hz	Hertz	é uma unidade de medida de frequência.
Kbps	Kilobit por segundo	significa ou envio ou recepção de quilobit por segundo. 1 Kbps, corresponde a 1000 bits por segundo.
MHz	MegaHertz	é uma unidade de medida de frequência, equivalente a um milhão de Hertz.
mW	miliWatt	é uma unidade de medida de potência, correspondente a um milésimo (1/1000) de Watt.
W	Watt	é a unidade básica de potência (P). 1 Watt significa 1 Ampere (A) de corrente em 1 Volt (V) de tensão. Em termos matemáticos tem-se: $P = VA$.

SUMÁRIO

1 INTRODUÇÃO	11
1.1 PROBLEMA	11
1.2 OBJETIVOS.....	12
1.2.1 Objetivo Geral.....	12
1.2.2 Objetivos Específicos	12
1.3 JUSTIFICATIVA.....	13
1.4 METODOLOGIA	13
1.5 ESTRUTURA DO TRABALHO	14
2 INTERNET DAS COISAS (INTERNET OF THINGS - IOT)	15
2.1 CARACTERÍSTICAS	15
2.2 COMUNICAÇÃO.....	15
2.3 PROTOCOLOS E PADRÕES.....	17
2.4 SISTEMAS OPERACIONAIS.....	18
3 PADRÃO IEEE 802.15.4.....	20
3.1 CARACTERÍSTICAS	20
4 PROTOCOLO DE INTERNET IPV6 OVER LOW POWER WIRELESS PERSONAL AREA NETWORKS (6LOWPAN)	25
4.1 APRESENTAÇÃO	25
4.2 CARACTERÍSTICAS	26
4.3 SISTEMA OPERACIONAL	29
5 ESTUDO DE CASO: REDE DE SENSORES SEM FIO PARA MONITORAMENTO DE ESTAÇÕES EMBALADORAS EM LINHAS DE PRODUÇÃO DE MATERIAL MÉDICO E CIRÚRGICO	31
5.1 ESTAÇÃO.....	32
5.2 NÓ SENSOR	33
5.2.1 CC2650 MCU	33
5.2.2 Sensores de Temperatura, Tempo de Aplicação e Pressão de Ar	33
5.3 ROTEADOR DE BORDA (BORDER ROUTER)	34
5.3.1 SLIP Radio	34
5.3.2 Raspberry PI 3 Model B.....	34
5.3.3 6LBR Smart Bridge.....	35
5.4 MQTT BROKER.....	35
5.5 APLICAÇÃO DE CONFIGURAÇÃO, PROCESSAMENTO E INTEGRAÇÃO (APPLICATION).....	36
5.5.1 MQTT Client	36
5.5.2 RESTful WS	36
5.6 BANCOS DE DADOS CORPORATIVOS	37

5.7 APLICAÇÃO ANDROID (APP)	37
5.8 SISTEMA DE CONTROLE DA QUALIDADE	38
6 CONCLUSÃO	39
REFERÊNCIAS.....	40

1 INTRODUÇÃO

1.1 PROBLEMA

Uma empresa de fabricação de material médico e cirúrgico, como seringas, agulhas, dosadores orais, tubos coletores de sangue, entre outros, precisa registrar junto a cada lote produzido informações dos equipamentos da linha de produção que têm contato direto com o produto, como as estações embaladoras. Os registros dessas informações, como temperatura, pressão e tempo de aplicação, serão posteriormente enviados ao cliente junto ao lote produzido, por exigência da agência governamental reguladora da área da saúde.

Apesar de as linhas de produção serem automatizadas, a captura e registro dessas informações são realizadas manualmente, entre o início e o término de cada lote. Esse cenário proporciona problemas como os enumerados na lista (não exaustiva) abaixo:

- Operadores precisam deixar o posto para fazer a leitura;
- Imprecisão em relação ao período de cada lote;
- Incompatibilidade entre os Controladores Lógicos Programáveis (CLP) e o sistema responsável pelo controle da produção para capturar e registrar os dados;
- Indisponibilidade de portas e barramentos dos equipamentos de controle da automação para mais sensores;
- Falha humana na leitura.

Não obstante os problemas citados já serem suficientes para motivar uma solução baseada em Rede de Sensores Sem Fio (RSSF), o ambiente de produção tem características, listadas a seguir, que motivam e justificam a aplicação de uma rede de sensores sem fio:

- Não há restrição para alimentação dos nós sensores;
- Apesar do chão de fábrica ser um ambiente ruidoso, não há outras redes sem fio, somente nas áreas administrativas da empresa;
- Não há espaço para uma nova rede cabeada, uma vez que a infraestrutura é utilizada pelos cabos da rede de automação das linhas de produção.

Há ainda outras características e restrições deste cenário, importantes para a definição de uma solução usando RSSF. As medidas que devem ser tomadas para o relatório a ser enviado junto ao lote produzido são:

- Temperatura de “formagem”;
- Temperatura de selagem;
- Tempo de selagem;
- Pressão de aplicação;
- Temperatura da água de refrigeração da embaladora;
- Pressão da água de refrigeração da embaladora;
- Pressão de ar comprimido da rede.

O intervalo de tomada das medidas ocorre somente no início do lote e a cada alteração no comando das máquinas, como quando o operador muda a temperatura de formagem. Há um sistema de Programação e Controle da Produção (PCP) que inicializa e finaliza o lote com leitura de código de barras feita pelo operador. Idealmente, deve-se tomar as medidas em intervalos regulares dentro do tempo de produção de cada lote. Uma RSSF pode realizar as leituras em intervalo mais adequado, para posterior extração de dados do sistema responsável por emitir o relatório.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

A solução proposta é um sistema composto por uma rede de sensores sem fio, uma aplicação de servidor com banco de dados, uma interface desta aplicação para integração com o sistema de controle de qualidade corporativo e um aplicativo para dispositivos móveis Android.

1.2.2 Objetivos Específicos

Para atender ao objetivo geral neste trabalho de conclusão de curso os seguintes objetivos específicos serão abordados:

- Apresentar a Internet das Coisas (ou *Internet of Things* - IoT);
- Analisar o sistema sem fio IEEE 802.15.4 para rede de sensores;

- Explicar o funcionamento do *Internet Protocol version 6* (IPv6) no IEEE 802.15.4;
- Apresentar o *IPv6 over Low power Wireless Personal Area Networks* (6LoWPAN) para Internet das Coisas (ou *Internet of Things* - IoT).

1.3 JUSTIFICATIVA

O estudo do tipo de rádio empregado em projetos de Internet das Coisas, bem como os protocolos com que ele trabalha e as próprias características físicas das transmissões se torna extremamente importante para a viabilidade de um projeto com muitos nós.

Em algumas situações a energia será o limitante, em outras a distância de transmissão bem como o ruído eletromagnético presente no local. Tendo ainda a possibilidade e haver combinações em um ambiente tão hostil para ondas de rádio que se deva pensar em um provável cabeamento.

Para o projeto exemplo desse estudo que foi a proposta de projeto: “Rede de sensores sem fio para monitoramento de estações embaladoras em linhas de produção de material médico e cirúrgico”, 2018. Trabalho que fez parte da Especialização em Internet das Coisas, disciplina “INC10: Projeto de Redes de Sensores sem Fio”.

A parte de distribuição de energia não se torna um limitante em um primeiro momento, devido a planta fabril ter disponibilidade de rede elétrica distribuída. Outro benefício da planta em questão é que os sinais de rádio presentes são provenientes de redes Wi-Fi administrativas, o que minimiza as chances de fontes de interferência.

1.4 METODOLOGIA

Para este trabalho de conclusão de curso (monografia), a metodologia aplicada no desenvolvimento está no estudo de bibliografias, simuladores virtuais, pesquisas na internet, vídeo aulas e apresentações.

1.5 ESTRUTURA DO TRABALHO

Esta monografia de especialização está dividida em 6 (seis) seções. Nesta primeira seção, foi introduzido o assunto tema do trabalho e também foram abordados a motivação e os objetivos geral e específicos da pesquisa, a justificativa e a estrutura geral do trabalho.

Já na segunda seção: “Internet das Coisas (Internet of Things - IoT)”, será apresentada as características da Internet das Coisas (ou *Internet of Things* - IoT) os modelos de comunicação, protocolos e padrões e os sistemas operacionais utilizados na implementação da IoT.

A seguir, na terceira seção: “Padrão IEEE 802.15.4”, será abordado o protocolo de comunicação para redes pessoais sem fio.

Na quarta seção: “Protocolo de Internet IPv6 over Low power Wireless Personal Area Networks (6LoWPAN)”, será apresentado o protocolo para internet pessoal sem fio 6LoWPAN.

Já quinta seção: “Estudo de Caso: Rede de Sensores Sem Fio para Monitoramento de Estações Embaladoras em Linhas de Produção de Material Médico e Cirúrgico”, será demonstrada a utilização das tecnologias no desenvolvimento de uma aplicação para a Internet das Coisas.

Por último, na sexta seção: “Conclusão”, serão retomados a pergunta de pesquisa e os seus objetivos e apontado como foram solucionados, respondidos e atingidos, por meio do trabalho realizado.

2 INTERNET DAS COISAS (INTERNET OF THINGS - IOT)

2.1 CARACTERÍSTICAS

O termo “Internet das Coisas”, *Internet of Things* (IoT) foi cunhado por Kevin Ashton em uma apresentação à Proctor & Gamble em 1999. Ashton é um dos fundadores do Laboratório de Auto-ID do *Massachusetts Institute of Technology* (MIT). Ele foi pioneiro no uso de *Radio-Frequency IDentification* (RFID) no gerenciamento da cadeia de suprimentos (ASHTON, 2015).

De uma forma mais técnica a Internet das Coisas surgiu com os avanços em várias áreas como os sistemas embarcados, a microeletrônica, a comunicação sem fio, a evolução do sensoriamento e, principalmente, a maior facilidade de se ter rede e dispositivos com acesso à internet. Seja esse acesso direta ou indiretamente através de redes criadas para IoT com o intuito de otimizar as comunicações de dispositivos mais simples, sem grande capacidade de processamento e ou energia. Esse conjunto de fatores viabilizou se conectar qualquer coisa à internet e ter interação com ela. Essa é a grande chave do IoT, ter o canal de ida e volta com o que se deseja interagir. E que as coisas possam interagir também entre elas, com o humano passando a ser um observador do sistema (ASHTON, 2015).

2.2 COMUNICAÇÃO

Em março de 2015 foi publicado pelo *Internet Engineering Task Force* (IETF) os quatro modelos básicos para comunicação em IoT: a) *Device-to-Device* (dispositivo para dispositivo); b) *Device-to-Cloud* (dispositivo para nuvem); c) *Device-to-Gateway* (dispositivo para o gateway); e d) *Back-End Data Sharing* (compartilhamento de dados do Back-End). Esse documento é uma publicação da *Requests For Comments* (RFC), número 7452, com o título “Architectural Considerations in Smart Object Networking” que foi submetido pelo *Internet Architecture Board* (IAB) (TSCHOFENIG et al., 2015).

No primeiro modelo: *Device-to-Device* (dispositivo para dispositivo), um dispositivo se comunica diretamente com outro dispositivo mesmo que não haja conexão direta com a internet (TSCHOFENIG et al., 2015). A tecnologia de comunicação *bluetooth*, seria um exemplo dessa forma de comunicação.

Uma rede de sensores sem fio, seria um outro exemplo, desse modelo de comunicação direta em dispositivos. Os dispositivos que também podem ganhar o nome de Nó em IoT, usam protocolos mais simples, pois muitas vezes as informações enviadas são também simples, como um sensor de temperatura, que irá apenas informar de tempo em tempo o valor medido.

Esses protocolos também têm atenção para outros limites do dispositivo, como capacidade de processamento, memória e energia que são limitados em sua maioria. Para padronizar tais protocolos alguns grupos como IEEE, ITU, *Open Connectivity Foundation*, *AllSeen Alliance*, entre outros, criam e incentivam o uso de seus protocolos cada um oferecendo vantagens e desvantagens em cada situação (MOMOTE, 2016).

No caso do segundo modelo: *Device-to-Cloud* (dispositivo para nuvem), o dispositivo está conectado diretamente à internet (TSCHOFENIG et al., 2015). Um exemplo, seria o monitoramento GPS de um veículo, na qual o dispositivo estaria ligado à rede móvel celular coletando dados de posicionamento e enviado a um servidor o qual pode ser acessado de outro dispositivo ou plataforma.

O terceiro modelo relacionado: *Device-to-Gateway* (dispositivo para o gateway), o dispositivo se conecta a um *gateway* para ter acesso à internet (TSCHOFENIG et al., 2015). Um exemplo, deste tipo de modelo seria um dispositivo *bluetooth* que ao se comunicar com o telefone consegue reportar informações a um servidor.

No quarto e último modelo: *Back-End Data Sharing* (compartilhamento de dados do Back-End), a organização e processamento dos dados coletados pelos dispositivos e a aplicação de uma inteligência computacional para tomar decisões sobre ações futuras ou imediatas (TSCHOFENIG et al., 2015).

Com análises feitas a partir de informações de vários sensores, como por exemplo, em uma fazenda, é possível determinar que área está pronta para colheita ou que área precisa de mais atenção em adubação tudo por um computador e já exportar tais informações para outros sistemas como de colheita ou irrigação (MOMOTE, 2016; TSCHOFENIG, 2015).

2.3 PROTOCOLOS E PADRÕES

Como a IoT ainda é uma tecnologia relativamente nova e com uma abrangência em praticamente todas as áreas em que se deseja ou possa usar um sensor ou atuador remoto, várias iniciativas existem para tentar firmar um padrão de protocolos a ser usado. Tais protocolos estão em constante atualização e alguns exemplos são mostrados a seguir.

Para a camada de enlace onde tem-se a parte física e MAC, os quatro principais protocolos são: a) Bluetooth low energy; b) Zigbee Smart Energy; c) Long-Term Evolution Advanced (LTE-A); e d) IEEE 802.15.4.

O primeiro protocolo: *Bluetooth low energy*, ou também chamado de *Bluetooth Smart*, é destinado a comunicações curtas, pode ser até dez vezes mais econômico energeticamente e com uma latência até quinze vezes maior do que o *bluetooth* convencional, também oferece maior tempo de *sleep*, dormir enquanto não está transmitindo.

No caso do segundo protocolo: Zigbee Smart Energy, ele é destinado a comunicações de médio alcance, como casas e hospitais. Tem suporte a topologias estrela, P2P e *cluster tree*, com a presença de um nó coordenador por topologia, também conferindo segurança na camada física.

Já o terceiro protocolo: *Long-Term Evolution Advanced*, usa a rede celular para aplicações máquina-máquina (*Machine to Machine - M2M*), é um protocolo de menor custo para a rede se comparado com os protocolos padrões de telefonia, pode ser usado em várias bandas. A arquitetura do LTE-A consiste no coordenador de rede, no rádio de acesso e nos nós moveis, o coordenador de rede que é o responsável por controlar e manter os IPs para cada Nó.

Por último, o quarto protocolo: IEEE 802.15.4, é o mais comum para aplicações IoT, ele define o tipo de *frame*, de cabeçalhos, incluindo endereços de origem e destino, bem como os nós iram se comunicar entre si. O IEEE 802.15.4 será melhor detalhado no próximo capítulo (JAIN, 2015).

Para a camada de rede onde tem-se os encapsuladores/ adaptadores, que vão implementar, por exemplo, o IPv6 em uma rede bluetooth ou IEEE 802.15.4.

O *IPv6 over Bluetooth Low Energy (6LoWPAN)* reusa a maior parte das técnicas de compressão do 6LoWPAN, outra característica significativa é que o

Bluetooth Low Energy não suporta atualmente a formação de redes *multi-hop* na camada de enlace.

O 6LoWPAN é o mais comum nesta categoria. Ele encapsula eficientemente cabeçalhos longos de IPv6 em pequenos pacotes IEEE802.15.4. As especificações suportam diferentes comprimentos de endereços, baixa largura de banda, diferentes topologias incluindo a topologia em malha, longo tempo de *Sleep*, economia de energia dentre outras características que serão abordadas em outro capítulo (JAIN, 2015).

Para a camada de aplicação onde se tem as mensagens com a real informação, a maioria das aplicações fazem uso de protocolos TCP ou *User Datagram Protocol* (UDP) para transporte, alguns exemplos dessas aplicações o *Message Queue Telemetry Transport* (MQTT) e o *Constrained Application Protocol* (CoAP).

O MQTT foi introduzido pela IBM em 1999 e padronizado pelo OASIS em 2013 ele foi projetado para fornecer conectividade incorporada entre aplicativos e que estão de um lado da rede e os que estão do outro lado. Por exemplo o nó sensor de temperatura e o banco de dados que vai trabalhar com essas informações, ele faz isso com uma técnica chamada de publicação e assinatura (*publish/subscribe*), onde o nó publica em um tópico a informação e o banco de dados está inscrito naquele tópico.

O CoAP é um protocolo de transferência web adaptado para uso em nós restritos em redes IoT. O modelo de interação do CoAP é semelhante ao modelo cliente/ servidor do protocolo HTTP. Um pedido CoAP é equivalente ao do HTTP onde um cliente solicita uma ação para um servidor. E o servidor envia uma resposta. Mas ao contrário do HTTP, o CoAP lida com esses intercâmbios de forma assíncrona ao longo de um transporte, orientado a datagramas, como o UDP. Isso é feito logicamente usando uma camada de mensagens que suporta uma confiabilidade opcional (JAIN, 2015).

2.4 SISTEMAS OPERACIONAIS

Para os sistemas operacionais em IoT a disputa em determinar um padrão é tão acirrada quanto os protocolos, com empresas grandes ao nível de Google e Microsoft desenvolvendo suas soluções. Para exemplo, em sistemas *open source*,

tem-se, principalmente, três sistemas operacionais (SO, ou *Operating System* - OS): a) RIOT-OS; b) TinyOS; e c) Contiki.

O primeiro sistema operacional: RIOT-OS, que teve seu início em 2008, como um SO para redes de sensores *wireless* atualmente se auto-denomina o “Sistema Operacional Amigável para a Internet das Coisas”. Tem como ponto de destaque o suporte aos padrões e pilhas de protocolos IPv6, 6LoWPAN, *IPv6 Routing Protocol for Low Power and Lossy Network* (RPL), UDP e *Constrained Application Protocol* (CoAP). E a possibilidade de desenvolvimento de aplicações em C/C++ (RIOT, 2018).

No caso do segundo sistema operacional: TinyOS, que teve início em 2000, na Universidade da Califórnia, em Berkley com o apoio da Intel e é um sistema operacional de código aberto e licença BSD projetado para dispositivos sem fio de baixa potência, como aqueles usados em redes de sensores sem fios. O TinyOS utiliza como base uma linguagem de programação chamada NesC, que é uma extensão do C contendo certas estruturas para o gerenciamento de rede e processamentos com foco em redes de sensores sem fio (RAYMOND, 2007).

Por último, o terceiro sistema operacional: Contiki, foi criado em 2002, por Adam Dunkels mas hoje é mantido por uma equipe de desenvolvedores ao redor do mundo. O projeto se auto-denomina o “Sistema Operacional de Código Aberto para a Internet das Coisas”. Possui suporte aos protocolos TCP/IP, além dos protocolos IPv6 focados em baixo consumo como 6LoWPAN, RPL e CoAP (CONTIKI, 2018).

3 PADRÃO IEEE 802.15.4

3.1 CARACTERÍSTICAS

O IEEE 802.15.4 é um padrão (protocolo, conjunto de regras) para *Wireless Personal Area Networks* (WPAN) que em tradução direta seria “rede pessoal sem fio”. Essas redes têm como principal característica as baixas taxas empregadas bem como o baixo custo dos dispositivos e o curto alcance, variando entre dez a cem metros de distância, podendo haver também configurações de maior alcance. Os dispositivos que geralmente são alimentados por baterias têm baixo processamento e restrição de memória disponível visando aumentar o tempo de duração da bateria.




O padrão IEEE 802.15.4 tem baixa de complexidade se comparado a outros padrões como o Bluetooth, por exemplo. Estão disponíveis três bandas de frequência: a) 868 MHz^{1,2}, b) 915 MHz, e c) 2,4 GHz³; sendo essa última a mais comum, como mostrado na Figura 1, com 16 canais de 5 MHz. É utilizado o *Carrier Sense Multiple Access with Collision Avoidance* (CSMA-CA) ou o CSMA-CA com uso de *slots*, para o acesso ao meio físico (MOREIRAS, 2013).

¹ “Hertz” é uma unidade de medida de frequência, ela deriva do Sistema Internacional de Medidas. O Hertz é equivalente a ciclos por segundo e pode ser usado para medir a frequência de ondas sonoras como o rádio, as batidas do coração humano, a frequência de um processador de computador como medida de desempenho do mesmo. Fonte: **Diferença entre GHz (gigahertz) e MHz (megahertz)**. Disponível em: <<https://www.educacao.cc/educacao/diferenca-entre-ghz-gigahertz-e-mhz-megahertz-uhf-e-vhf/>>. Acesso em: 17 set. 2018.

² “MHz (MegaHertz)” é uma unidade de frequência, equivalente a um milhão de Hertz ou mil kilohertz (kHz). Se algo acontece com uma frequência de uma vez por segundo, este fenômeno tem a frequência de 1 Hertz. Sendo assim, uma frequência de 1 MHz significa que algo acontece um milhão de vezes por segundo. Fonte: **Glossário (MHz - Megahertz)**. Disponível em: <<https://www.decea.gov.br/sirius/index.php/2011/06/14/mhz-megahertz/>>. Acesso em: 17 set. 2018.

³ “GHz (GigaHertz)” é uma unidade de frequência, equivalente a um bilhão de Hertz. Fonte: **Diferença entre GHz (gigahertz) e MHz (megahertz)**. Disponível em: <<https://www.educacao.cc/educacao/diferenca-entre-ghz-gigahertz-e-mhz-megahertz-uhf-e-vhf/>>. Acesso em: 17 set. 2018.

Figura 1 - Atribuição dos canais (*Channel*) e bandas (*Band*) do IEEE 802.15.4

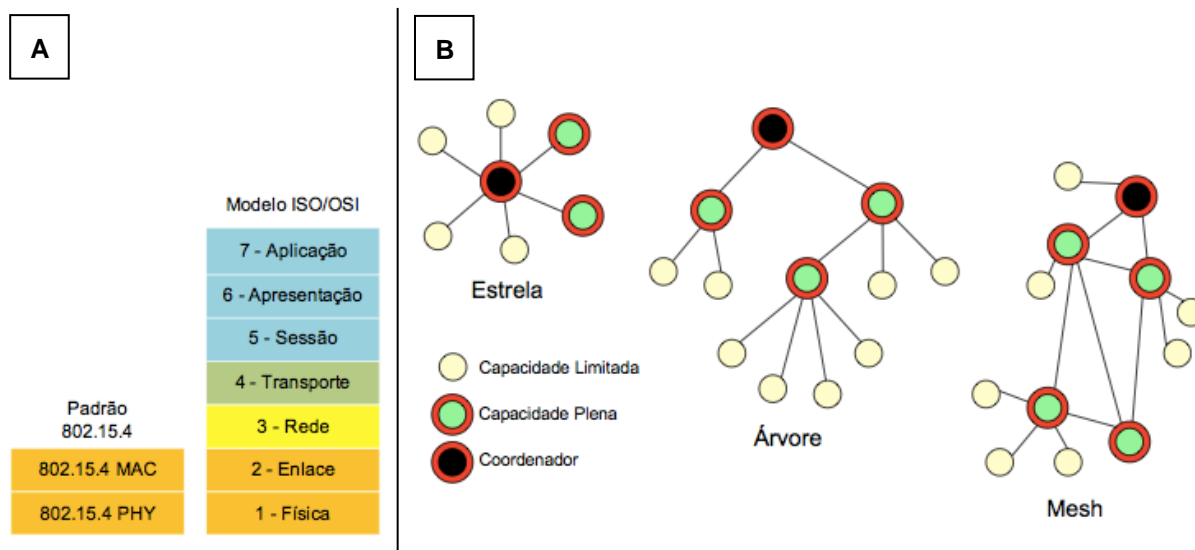
	Channel	Center Frequency (MHz)	Availability
868 MHz Band	0	868.3	 Europe
915 MHz Band	1	906	 Americas
	2	908	
	3	910	
	4	912	
	5	914	
	6	916	
	7	918	
	8	920	
	9	922	
	10	924	
2.4 GHz Band	11	2405	 World Wide
	12	2410	
	13	2415	
	14	2420	
	15	2425	
	16	2430	
	17	2435	
	18	2440	
	19	2445	
	20	2450	
	21	2455	
	22	2460	
	23	2465	
	24	2470	
	25	2475	
	26	2480	

Fonte: Gutierrez (2005).

O IEEE 802.15.4 define apenas as camadas física e de enlace, ou seja, camadas 1 e 2, respectivamente (Figura 2, parte A), do modelo de referência *International Organization for Standardization/ Open Systems Interconnection* (ISO/OSI), que em 802.15.4 tem o nome de física para 1 e *Media Access Control* (MAC) para 2, conforme ilustrado na Figura 2.

Na Figura 2 (parte B), também mostra algumas das topologias mais comuns que são implementadas na camada MAC e os tipos de dispositivos normalmente presentes em uma rede 802.15.4.

Figura 2 - Comparação entre o padrão 802.15.4 e o modelo ISO/OSI (parte A) e topologias possíveis numa rede 802.15.4 (parte B)



Fonte: Moreiras (2013).

Há duas classes básicas de dispositivos: a) os de Capacidade Plena ou *Full Function Devices* (FFD); e b) os de Capacidade Limitada ou *Reduced Function Devices* (RFD) (GUTIERREZ, 2005).

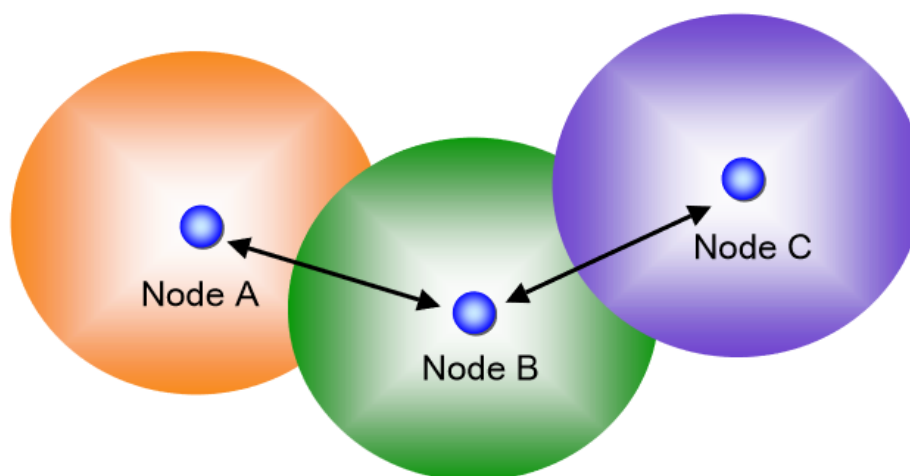
Os dispositivos “Capacidade Plena”, geralmente são alimentados via rede elétrica e tem maior capacidade de processamento. Estão presentes em todas as modalidades de rede. Eles executam uma implementação completa dos protocolos de rede, podendo assumir a função de Coordenador da Rede (ou *Personal Area Network* - PAN Coordinator) e sendo capazes de executar roteamento (JAIN, 2016).

Os dispositivos de “Capacidade Limitada”, geralmente são alimentados por pilhas ou baterias, têm baixa capacidade de processamento e memória, não sendo capazes de realizar roteamento, ficando na borda da rede, ou seja, dispositivos RFD se comunicação apenas com dispositivos FFD sendo uma rede estrela essa relação. Geralmente os RFD são equipados com os sensores e atuadores utilizados na rede, contudo os dispositivos FFD também podem assumir essa função antagonicamente um dispositivo RFD não pode se tornar um coordenador da rede (JAIN, 2016).

Algumas características da rede no IEEE 802.15.4, um coordenador pode perguntar e atribuir a outro dispositivo FFD a capacidade de coordenador, para otimizar o gerenciamento de rede e redução do consumo de energia. Bem como a troca de funções entre dispositivos já mencionada. Isso faz parte da adaptação automática da topologia que também é usada quando nos dispositivos são removidos ou adicionados, fazendo o próximo nó se tornar um roteador do seu

vizinho imediato, fazendo uma auto-organização da rede. Essa característica de se auto adaptar também é possível graças a característica de múltiplos saltos (*multi-hop*), onde um nó A para chegar ao um nó C, usa um nó intermediário B, como mostra a Figura 3 (GUTIERREZ, 2005).

Figura 3 - Comunicação Multi-hop



Fonte: Gutierrez (2005).

Outra característica do IEEE 802.15.4 é a capacidade de dormir entre uma transmissão e outra, em inglês, *beacon enabled csma-ca*, também visando otimizar o consumo de energia, o que gera uma adaptação de rede, vale ressaltar que cada nó se conecta ao seu vizinho de maneira automática existem estudos em que algumas aplicações seria viável modificar essa característica para o radio nunca dormir, mas isso não será abordado nesse material (GUTIERREZ, 2005).

Com relação as taxas, o IEEE 802.15.4, trabalha na camada física com 250 Kbps⁴ na banda de 2,4 GHz com modulação *Offset Quadrature Phase Shift Keying* (OQPSK), nas outras bandas é menor entre 20 e 40 Kbps em 868 e 915 MHz respectivamente e 50 Kbps, com modulação *Binary Phase Shift Keying* (BPSK) na camada de aplicação. O frame da camada MAC com 127 bytes e *payload* de 77 bytes. E é justamente o tamanho desse frame que motiva o aparecimento do

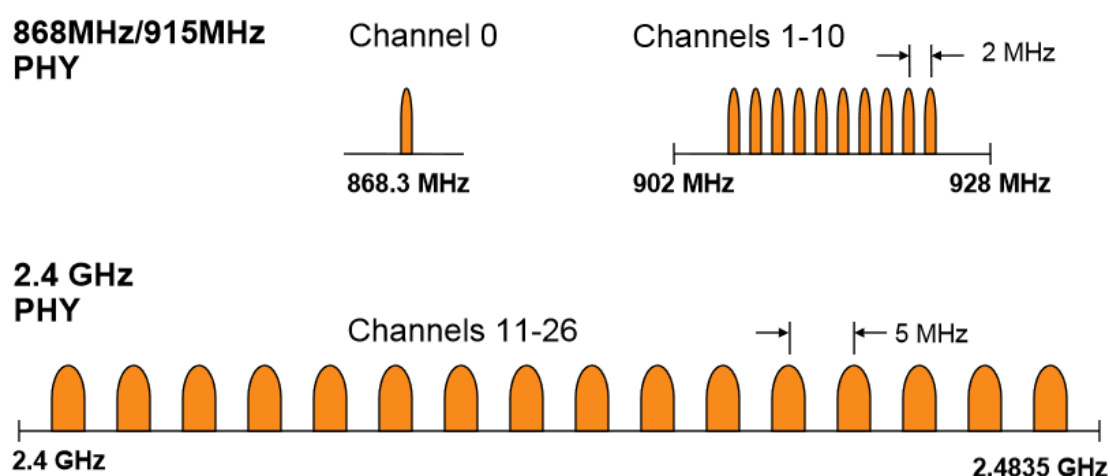
⁴ "Kbps" significa ou envio ou recepção de kilobit por segundo. Sendo assim: um kilobit por segundo (1 Kbps) corresponde ao envio ou recepção de 1000 bits por segundo. A palavra "bit" é uma contração do termo inglês "*binary digit*" que significa "dígito binário". O "bit" é a menor unidade de informação que pode ser armazenada ou transmitida, podendo assumir somente dois valores: **0** (zero) ou **1** (um). Fonte: **Significado de Kbps**. Disponível em: <<https://www.significados.com.br/kbps/>>. Acesso em: 17 set. 2018.

6LoWPAN, como solução para adaptar o IPv6 ao IEEE 802.15.4 o qual será mostrado no próximo capítulo (GUTIERREZ, 2005).

Um detalhamento das bandas de frequência no padrão IEEE 802.15.4 operado pode ser observado na Figura 4.

Figura 4 - Frequências e bandas de operação IEEE 802.15.4

Operating Frequency Bands



Fonte: Gutierrez (2005).

Como características de transmissão e recepção da camada de ar do 802.15.4 tem-se a potência de transmissão que é pelo menos 1 mW^{5,6} e a sensibilidade de recepção como -85 dBm⁷ para a banda de 2,4 GHz e -92 dBm para a banda de 868/915 MHz (GUTIERREZ, 2005).

⁵ "W (Watt)" é a unidade básica de potência (P). 1 Watt significa 1 Ampere (A) de corrente em 1 Volt (V) de tensão. Em termos matemáticos tem-se: $P = VA$. Fonte: **Watt (W) e miliWatt (mW)**. Disponível em: <<http://www.telecomhall.com/br/o-que-e-0-dbm.aspx>>. Acesso em: 23 set. 2018.

⁶ "mW (miliWatt)" é uma unidade de medida de potência, correspondente a um milésimo (1/1000) de Watt. Fonte: **Significado de Miliwatt**. Disponível em: <<https://www.dicio.com.br/miliwatt/>>. Acesso em: 23 set. 2018.

⁷ "dBm (decibel-miliwatt)" é uma unidade de energia elétrica em decibéis (dB), referenciada a 1 miliwatt (mW). Fonte: **dBm definition**. Disponível em: <https://www.rapidtables.com/electric/dBm.html#dBm_Definition>. Acesso em: 23 set. 2018.

4 PROTOCOLO DE INTERNET IPV6 OVER LOW POWER WIRELESS PERSONAL AREA NETWORKS (6LOWPAN)

4.1 APRESENTAÇÃO

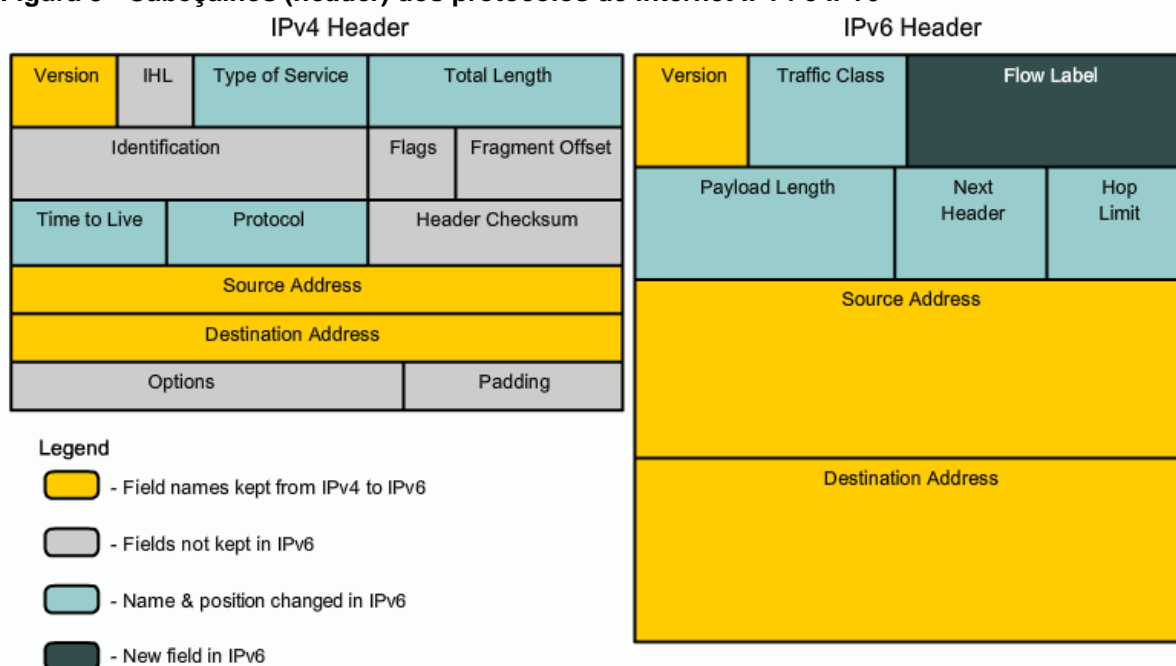
O Protocolo de Internet versão 4 (ou *Internet Protocol version 4* - IPv4), que é o protocolo utilizado pela maioria das redes hoje em dia, com seu cabeçalho de 32 bits, ou seja, 2^{32} o que possibilita a criação de 4.294.967.296 (quatro bilhões, duzentos e noventa e quatro milhões, novecentos e sessenta e sete mil e duzentos e noventa e seis) de endereços IP. Um valor que para os dias atuais se torna insuficiente mesmo com esforços para prolongar a vida útil do IPv4 (MOREIRAS, 2013).

Além da crescente rede de computadores de forma global outros fatores como crescimento de usuários moveis (telefonia celular) e equipamentos como carros, casas, sistemas de segurança, redes elétricas (*smart grid*) interagindo diretamente na internet a demanda por IP cresceu (MOREIRAS, 2013).

Como solução veio o IPv6, usando um cabeçalho de 128 bits, ou seja, 2^{128} o que possibilita a criação de 340.282.366.920.938.463.463.374.607.431.768.211.456 de endereços IP. Um protocolo de internet que entrega aproximadamente 79 octilhões ($7,9 \times 10^{28}$) de vezes o número de IP que o IPv4 disponibiliza, ou como alguns especialistas dizem “um IP para cada grão de areia” (MOREIRAS, 2013).

Outras características que são melhorias no IPv6 em relação ao IPv4 são o endereçamento IP aprimorado, o cabeçalho simples em relação ao predecessor com mostra a Figura 5, a mobilidade e segurança.

Figura 5 - Cabeçalhos (*header*) dos protocolos de internet IPv4 e IPv6



Fonte: Cisco (2018).

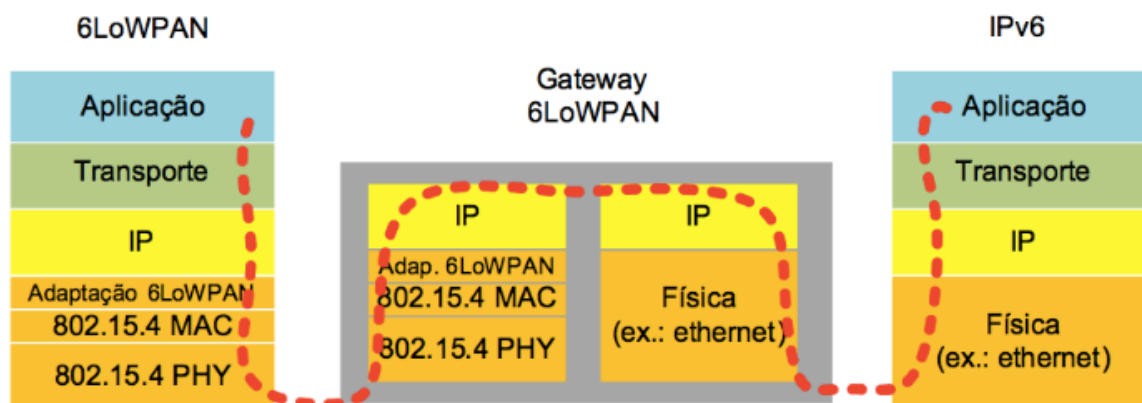
4.2 CARACTERÍSTICAS

O *IPv6 over Low power Wireless Personal Area Networks (6LoWPAN)* IEEE 802.15.4 é um padrão (protocolo, conjunto de regras) para *Wireless Personal Area Networks (WPAN)*, foi criado e nomeado por Geoff Mulligan, considerado o "Pai da internet embarcada" e detentor de mais de quinze patentes em segurança computacional (MULLIGAN, 2018). Geoff Mulligan implementou o IPv6 em pequenas redes de sensores para mostrar seu potencial em 2001. Posteriormente veio propor a formação de um grupo de trabalho do *Internet Engineering Task Force (IETF)* com intenção de padronizar a ideia. Ele nomeou o grupo de trabalho de 6LoWPAN para facilitar a localização na lista de ativos do grupo de trabalho (MULLIGAN, 2018).

6LoWPAN é uma adaptação, uma implementação para que as redes IEEE 802.15.4 possam suportar o IPv6 (Figura 6). Essa adaptação em uma maneira simplista consiste na compressão dos cabeçalhos IPv6 para sua posterior transmissão na rede IEEE 802.15.4 e o componente de rede responsável por essa tarefa é o gateway. Essa tecnologia trabalha nas camadas física, enlace e rede, ou seja 1, 2 e 3 do modelo OSI para transmissão dos dados, agilizando assim o processo e sem complexidades. Também aproveitando protocolos que funcionam

sobre *Internet Protocol (IP)*, *Transmission Control Protocol (TCP)* ou *User Datagram Protocol (UDP)* (MOREIRAS, 2013).

Figura 6 - 6LoWPAN + Gateway 6LoWPAN + IPv6



Fonte: Moreiras (2013).

A primeira *Request For Comments (RFC)* do grupo de trabalho 6LoWPAN foi a RFC 4919 de 2007 e nela foram descritos alguns desafios e metas, como exemplo, a necessidade do protocolo funcionar em dispositivos com restrição de processamento e memória; com um *Maximum Transmission Unit (MTU)* pequeno na camada de rede a necessidade da compressão dos pacotes e fragmentação na camada 2 OSI; a previsão de redes mesh com protocolos de roteamento simples devido a limitação dos dispositivos/nó dentre outros.

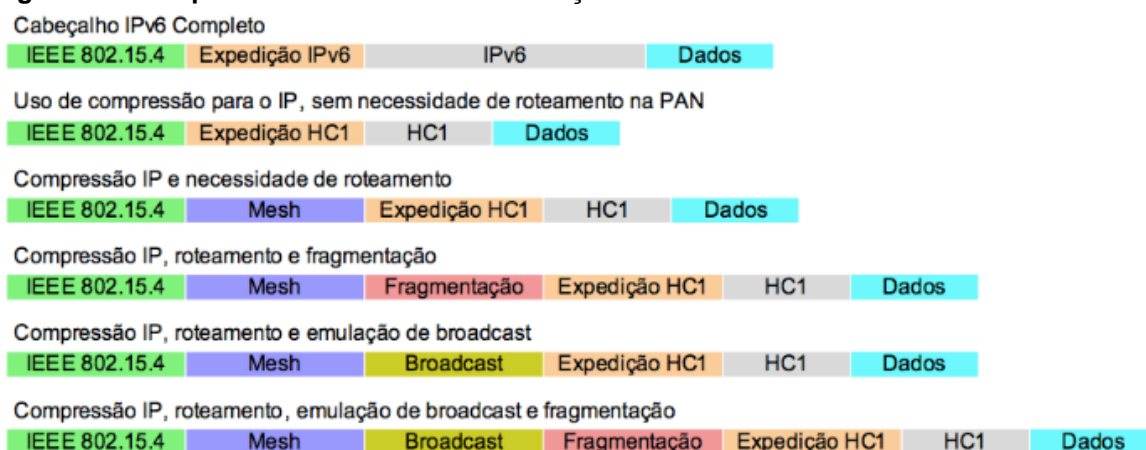
A RFC 4944 especifica a camada de adaptação, onde é feita a compressão dos cabeçalhos e a fragmentação e desfragmentação dos pacotes, isso é necessário já que a MTU da camada física é menor do que o mínimo especificado pelo IPv6, também é onde está a permissão para se emular o broadcast e suporte a protocolos de roteamento da camada 2 para redes mesh (MONTENEGRO, 2007).

Como protocolos de aplicação IoT, tem-se: *Message Queuing Telemetry Transport (MQTT)*, *Advanced Message Queuing Protocol (AMQP)*, *eXtensible Messaging and Presence Protocol (XMPP)* que são usados ao invés de por exemplo o *HyperText Transfer Protocol (HTTP)* (NARAYANA; SHARMA; VEETURI, 2018).

A compressão de cabeçalho definida pela RFC 4944 define o mecanismo para adaptar os 1280 bytes de MTU do IPv6 para o 802.15.4 que tem um tamanho de quadro (*frame*) máximo de 102 octetos, que é derivado da remoção de 25 octetos (*frame overhead*) de um tamanho máximo de pacote de 127 octetos (NARAYANA; SHARMA; VEETURI, 2018).

Como exemplo, apresentado na Figura 7, onde o *Header Compression* (HC), pode ser HC1 quando o cabeçalho é IPv6, HC2 quando é um cabeçalho UPD. O mesh é utilizado para roteamento na camada 2, e o de fragmentação e *broadcast* para fragmentação e serviços em *broadcast*, respectivamente.

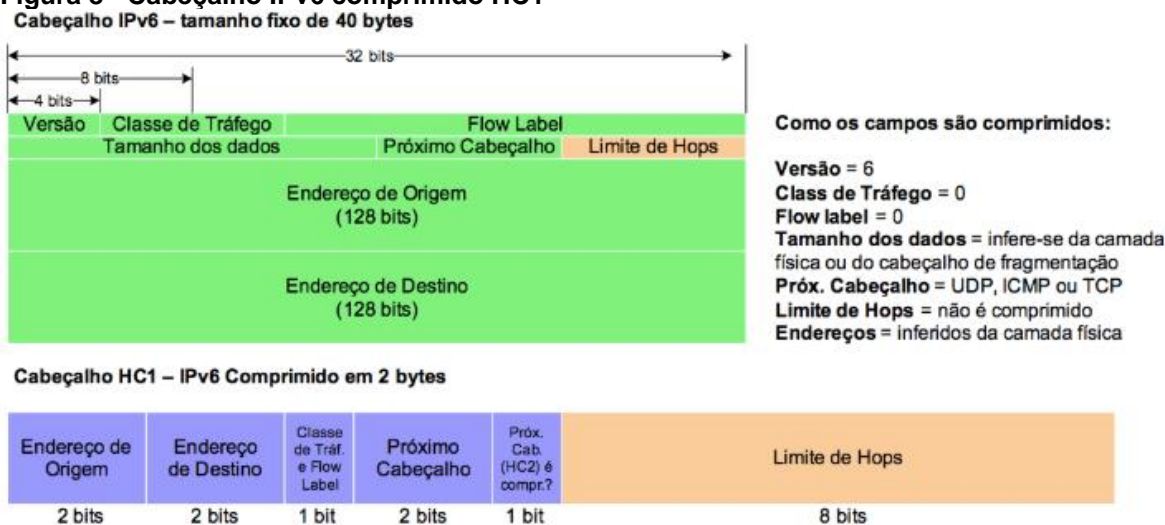
Figura 7 - Exemplo de encadeamento de cabeçalhos no 6LoWPAN



Fonte: Moreiras (2013).

Na Figura 8, mostra a técnica de compressão *stateless* que é usando no 6LoWPAN, onde os 40 bytes do cabeçalho IPv6 são comprimidos em apenas 2 bytes. Essa técnica é possível devido ao formato do IPv6 ser proveniente dos endereços físicos de cada dispositivo 802.15.4 bem como outras simplificações. Todos os campos em verde serão comprimidos ou excluídos mas o limite de *hops* será enviado de forma integral (MOREIRAS, 2013).

Figura 8 - Cabeçalho IPv6 comprimido HC1



Fonte: Moreiras (2013).

Apesar do 6LoWPAN ter sido desenvolvido para rede 802.15.4, existem estudos para ser usado em outras redes como Wi-Fi de baixa potência, Bluetooth, *Power Line Communication* (PLC), com o objetivo não só de compressão de cabeçalho mas também visando melhor utilizar a energia envolvida.

4.3 SISTEMA OPERACIONAL

Para os trabalhos em estudo foi usado o Contiki como software de implementação, software esse que é *Open Source*, outro exemplo de software *Open Source* seria o Arduino.

De acordo com o site oficial do software Contiki (CONTIKI, 2018), Contiki é um sistema operacional de código aberto para a Internet das Coisas. O Contiki conecta pequenos microcontroladores de baixo custo e baixo consumo de energia à internet. O Contiki é uma ferramenta poderosa para a construção de sistemas sem fio complexos.

Ainda conforme o site do Contiki (CONTIKI, 2018), o sistema operacional Contiki fornece uma poderosa comunicação de baixa potência na internet. O Contiki suporta os protocolos IPv6 e IPv4 totalmente padrão, junto com os recentes padrões sem fio de baixo consumo de energia: 6LoWPAN, *IPv6 Routing Protocol for Low Power and Lossy Network* (RPL) e *Constrained Application Protocol* (CoAP). Com ContikiMAC e *sleepy routers* (roteadores que dormem enquanto não estão trabalhando) da Contiki, até mesmo roteadores sem fio podem ser operados por bateria.

O Contiki suporta os protocolos IETF recentemente padronizados para redes IPv6 de baixa potência, incluindo a camada de adaptação do 6LoWPAN, o protocolo de roteamento *multi-hop* RPL IPv6 e o protocolo da camada de aplicação CoAP RESTful (CONTIKI, 2018).

De acordo com o site oficial da plataforma Arduino (ARDUINO, 2018), Arduino é uma plataforma *open source* de computação física baseada em uma placa de microcontrolador simples e um ambiente de desenvolvimento para desenvolver softwares para a placa.

Placas Arduino são relativamente baratas em comparação com outras plataformas de microcontroladores. E possuem módulos pré-montados. O software Arduino roda em Windows, Macintosh OSX e Linux. Com um ambiente de

programação simples, limpo e fácil de usar para iniciantes, mas suficientemente flexível para usuários avançados para um bom aproveitamento (ARDUINO, 2018).

Outra característica do Arduino, é que vários códigos já estão disponíveis no modelo *open source* por programadores experientes, e essas bibliotecas podem ser expandidas no formato C++ caso o usuário queira programar o código (ARDUINO, 2018).

Devido ao tipo de hardware do Arduino com vários fornecedores hoje em dia seguindo o mesmo padrão e aos grupos de desenvolvedores de software, o Arduino se torna modular tanto em hardware quanto em software, o que transforma essa plataforma extremamente atraente para entusiastas e para didática básica. Além de estudos elaborados ou mesmo aplicações comerciais.

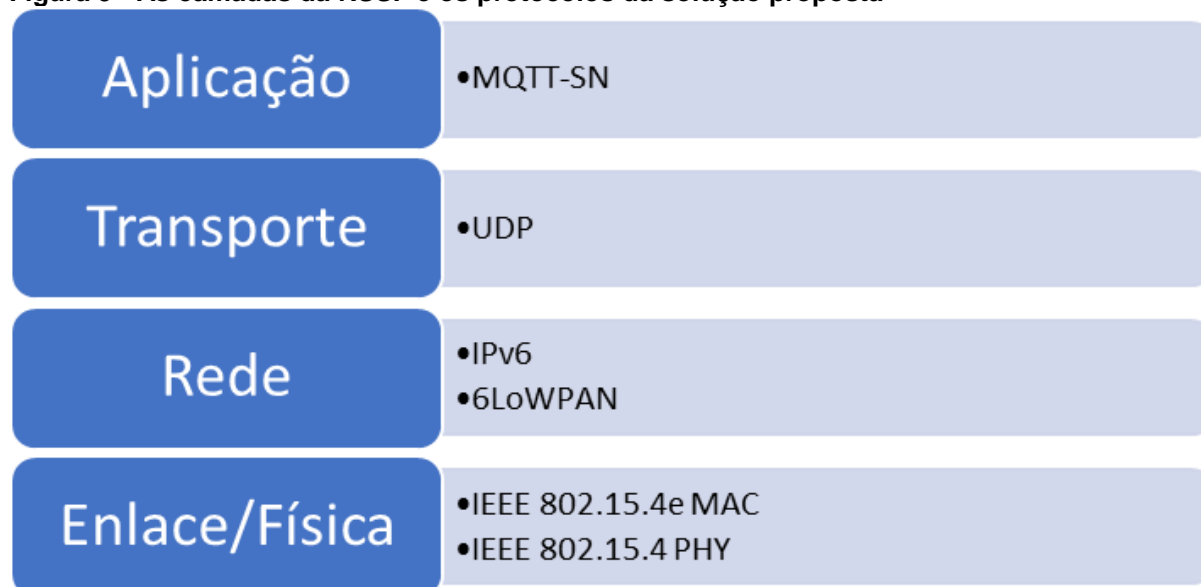
5 ESTUDO DE CASO: REDE DE SENSORES SEM FIO PARA MONITORAMENTO DE ESTAÇÕES EMBALADORAS EM LINHAS DE PRODUÇÃO DE MATERIAL MÉDICO E CIRÚRGICO

O projeto exemplo desse estudo foi uma proposta de projeto: “Rede de sensores sem fio para monitoramento de estações embaladoras em linhas de produção de material médico e cirúrgico”, 2018. Trabalho que fez parte da Especialização em Internet das Coisas, disciplina “INC10: Projeto de Redes de Sensores sem Fio”.

Os sensores da RSSF serão instalados aos equipamentos que devem ser monitorados, as estações embaladoras, para captura e envio dos dados de temperatura, pressão e tempo utilizando o protocolo *MQTT for Sensor Network* (MQTT-SN). A aplicação de servidor irá coletar os dados para armazenamento no banco de dados e fornecerá interface para o sistema de controle de qualidade, já existente, para criação do relatório para o lote. O aplicativo poderá ser usado para capturar as medições em tempo real, para inspeção e conferência dos operadores.

A Figura 9, ilustra as camadas da RSSF (aplicação, transporte, rede e enlace/física) propostas com os seus respectivos protocolos.

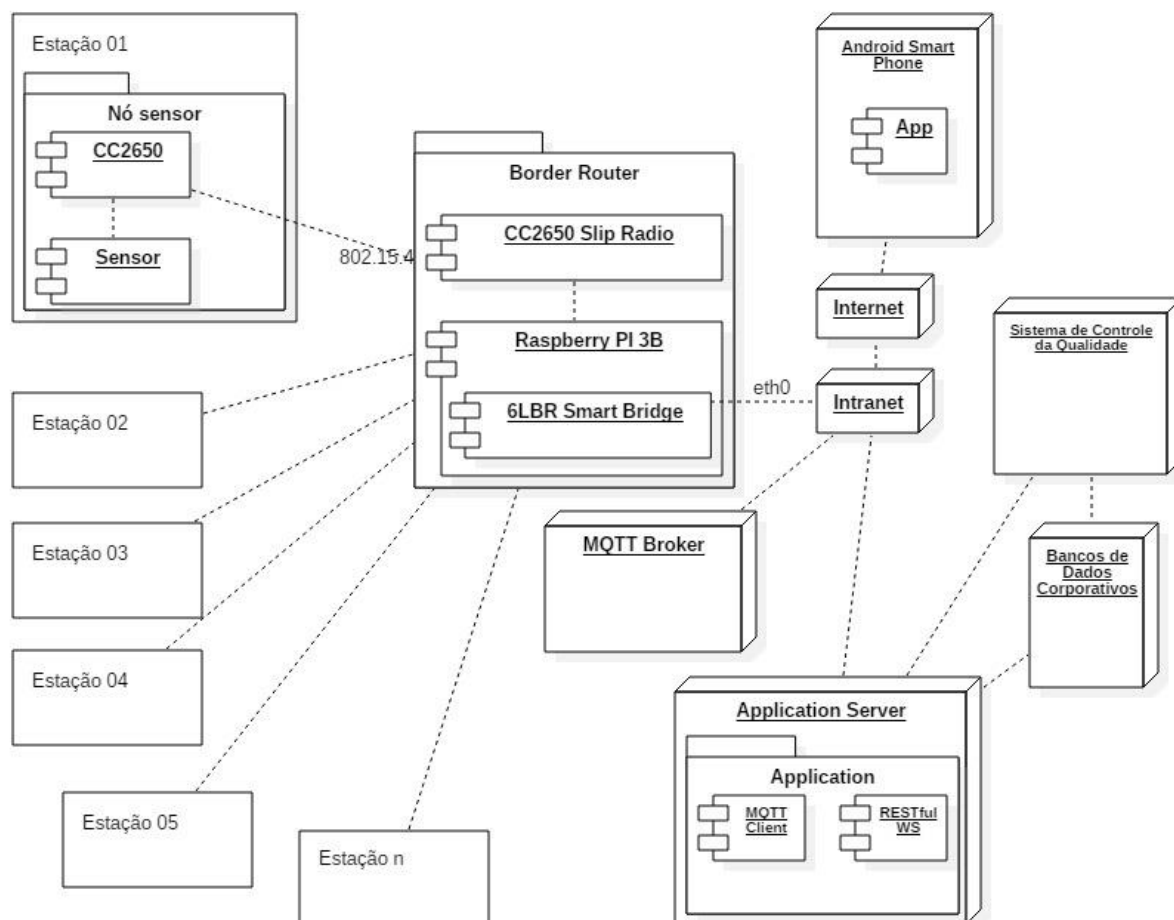
Figura 9 - As camadas da RSSF e os protocolos da solução proposta



Fonte: Bernardes, Bernardino e Chiesorin (2018).

A visão geral da solução está apresentada na Figura 10, cujos elementos estão descritos a seguir.

Figura 10 - Diagrama em blocos da solução proposta



Fonte: Bernardes, Bernardino e Chiesorin (2018).

5.1 ESTAÇÃO

Representa uma máquina embaladora ou outro elemento da linha de produção a ser monitorado. Cada estação pode ter um ou mais nós sensores instalados. A estação fornecerá a alimentação de energia para o hardware do nó.

A estação monitorada terá um tópico MQTT específico para cada tipo de medida, por exemplo, um tópico para a temperatura de selagem e outro tópico para tempo de aplicação.

5.2 NÓ SENSOR

É o principal elemento da RSSF. O microcontrolador (MCU) deverá ter apenas um sensor acoplado; portanto, estão previstos três tipos de nós sensores:

- Nó sensor de temperatura;
- Nó sensor de pressão de ar;
- Nó sensor de tempo.

A restrição de um sensor por MCU tem como objetivo simplificar a implantação da RSSF, de modo que cada nó publicará em apenas um tópico; outras justificativas são a complexidade do código de *firmware*⁸, que pode ser diminuída ao ser projetada para monitorar eventos de apenas uma interface de comunicação, e a utilização mais eficiente da transmissão de pacotes na rede.

5.2.1 CC2650 MCU

O *hardware* do nó sensor será implementado usando o MCU da Texas Instruments, CC2650 SimpleLink. Este hardware possui o módulo de rádio no padrão IEEE 802.15.4 e é compatível com o *firmware* construído por meio do Contiki (sistema operacional de tempo real, ou *Real Time Operating System* - RTOS). Cada MCU será responsável por ler os sinais de um sensor instalado, em intervalo de tempo pré-definido, e publicar as medidas por meio do protocolo MQTT-SN no respectivo tópico. Por exemplo, um nó sensor instalado na “Embaladora 01”, da linha de produção “Aguilhas 01”, que fará leitura de temperatura de selagem, irá publicar as mensagens no tópico “/agu01/emb01/temp/sel”.

5.2.2 Sensores de Temperatura, Tempo de Aplicação e Pressão de Ar

Os sensores a serem instalados nos nós são específicos para o tipo de medida, como temperatura e para a faixa de operação da estação. Por exemplo,

⁸ **Firmware** é um software, ou um conjunto de softwares, que vêm armazenados na memória do equipamento desde a fábrica, e que contém instruções e comandos para controlar determinado aparelho ou equipamento, como por exemplo, uma câmera fotográfica. Fonte: ARAUJO, Ana Paula de. **Firmware de Máquinas Fotográficas**. Disponível em: <<https://www.infoescola.com/fotografia/firmware-de-maquinas-fotograficas/>>. Acesso em: 22 out. 2018.

poderão ser utilizados sensores de temperatura (ou *Resistance Temperature Detectors* - RTD), termoresistores ou termopares. Para pressão, poderá ser utilizado o sensor piezoelétrico, e para os de tempo de aplicação, sensores *reed switch*. Os sensores podem ser conectados ao MCU usando as portas analógicas (para o caso de sensores analógicos), ou usando as interfaces seriais I2C ou SPI. O software de cada MCU será programado para o sensor especificado para aquele nó, como já citado na subseção 2.2: Comunicação.

5.3 ROTEADOR DE BORDA (BORDER ROUTER)

O *border router* é um *gateway* de rede que desempenha o papel de conectar os nós sensores à rede corporativa, tratando o tráfego de dados de e para as interfaces IEEE 802.15.4 por meio dos protocolos 6LoWPAN e IPv6. É o elemento que permitirá, portanto, que os nós sensores publiquem as medidas no *broker* MQTT implementado na rede corporativa. O roteador de borda é composto pelos blocos: a) SLIP Radio; b) Raspberry PI 3 Model B; e c) 6LBR Smart Bridge.

5.3.1 SLIP Radio

SLIP é uma sigla em inglês para *Serial Line Internet Protocol*, uma implementação do protocolo IP para trafegar os pacotes de dados em portas seriais.

Nesta proposta de solução, o SLIP Radio é implementado em um MCU CC2650; cumpre o papel de coordenador na topologia de RSSF do protocolo IEEE 802.15.4 e também o papel de encaminhar o tráfego de dados da interface IEEE 802.15.4 para uma porta UART conectada ao Raspberry PI 3 model B. Mais especificamente, desempenha o protocolo de comunicação entre o processo 6LBR (descrito a seguir) no sistema operacional do Raspberry e o rádio do MCU CC2650.

5.3.2 Raspberry PI 3 Model B

O outro elemento de hardware que compõe o *border router*, além do CC2650 do SLIP *Radio*, é o Raspberry PI 3 model B. É um computador *single-board* que oferece as interfaces necessárias para o desempenho do roteador de borda: uma porta UART (USB), conectada ao SLIP Radio, e uma porta Ethernet, conectada à

rede corporativa da fábrica. O Raspberry tem como sistema operacional o Raspbian, uma distribuição Linux dedicada.

O principal serviço deste bloco é o 6LBR, responsável por processar os pacotes recebidos pela porta USB e encaminhar para a porta Ethernet. O serviço está configurado para ser executado junto à inicialização do hardware, para garantir o gerenciamento dos nós sensores na rede.

5.3.3 6LBR Smart Bridge

Junto ao *firmware* do SLIP Radio, o 6LBR é o outro módulo de software que compõe o roteador de borda. É uma sigla para 6LoWPAN *Border Router*, isto é, um roteador de borda que utiliza os protocolos 6LoWPAN e RPL para conectar sensores à internet em uma RSSF.

Pode ser executado de modo embarcado em um hardware dedicado ou, como no caso da solução proposta, executado num sistema operacional Linux.

Neste cenário, o modo de operação do 6LBR escolhido é o *Smart Bridge*, que permite interconectar uma mesma rede IPv6 a uma RSSF de topologia *mesh* baseada no protocolo RPL, de modo que cada nó sensor terá seu IPv6 público fora da RSSF.

5.4 MQTT BROKER

A solução utiliza como protocolo de camada de aplicação o MQTT-SN. Cada nó sensor irá publicar as medidas realizadas em um *topic*; estes tópicos do protocolo MQTT-SN serão endereçados em um MQTT *broker*, isto é, o elemento que permite que o modelo *publish-subscribe* do protocolo MQTT seja implementado. O *broker* é componente do lado servidor, e, portanto, será implantado em um servidor de aplicação disponível na infraestrutura da empresa, ou, se necessário, implantado em uma infraestrutura de nuvem.

Nesta proposta, será utilizado o Really Small MQTT Broker, uma solução de código aberto que implementa o serviço de broker MQTT e que atende às características de desempenho e escalabilidade presentes neste cenário.

5.5 APLICAÇÃO DE CONFIGURAÇÃO, PROCESSAMENTO E INTEGRAÇÃO (APPLICATION)

Este é o módulo da arquitetura de ponta a ponta proposta que permitirá que os dados lidos e publicados pelos sensores possam ser utilizados pelos sistemas corporativos, como o Sistema de Controle da Qualidade. O módulo também fará a configuração dos nós sensores no nível da implantação - em qual estação o nó sensor está instalado.

A aplicação será implementada em uma arquitetura de micro serviços (ou *MicroService Architecture* - MSA), que permite características de escalabilidade e tolerância a falhas; será desenvolvida utilizando a linguagem de programação Java e será composta, principalmente, pelos módulos MQTT *client* e RESTful WS, descritos a seguir.

Os serviços da aplicação serão implantados em um servidor de aplicação Java, por exemplo, Tomcat ou WildFly, a ser disponibilizado pela fábrica ou contrato como serviço na nuvem.

5.5.1 MQTT Client

Este é o serviço que se conectará ao MQTT Broker para receber as publicações dos tópicos utilizados pelos nós sensores. Será implementado utilizando o Eclipse Paho Java Client e processará as medidas de temperatura, pressão e tempo para serem persistidas no banco de dados.

O serviço pode ser replicado em outros servidores para prever o aumento de escala, ou seja, o aumento de nó sensores numa mesma linha de produção ou ainda no caso de implantar outras RSSF nas demais linhas de produção da fábrica.

Cada instância do serviço será configurada para se inscrever nos tópicos MQTT de determinada linha de produção.

5.5.2 RESTful WS

É o módulo da aplicação que disponibilizará serviços web (*web services*) de arquitetura RESTful para dois propósitos principais: a) expor uma *Application Programming Interface* (API) de integração com outros sistemas corporativos; e b)

expor outra API para acesso à configuração da RSSF para a aplicação Android, descrita a seguir na subseção 5.7: Aplicação Android (App).

A API de integração permitirá que o Sistema de Controle da Qualidade realize consultas ao banco de dados onde estão armazenadas as medidas de temperatura, pressão e tempo, para construir os relatórios que acompanham os lotes produzidos. A arquitetura RESTful facilita a integração entre sistemas por utilizar os métodos e mensagens HTTP para acessar recursos; por exemplo, um recurso da API é a lista de uma determinada medida em determinado período, para determinada estação e linha de produção. Estes parâmetros são informados junto à URL de acesso ao recurso, e o serviço irá retornar no formato JSON o conjunto das medidas realizadas.

O outro serviço, que é a API de acesso à configuração dos nós na RSSF, permitirá que a aplicação Android monitore em tempo real um determinado nó sensor ou um conjunto deles. Ficarão disponíveis, entre outros recursos, a lista de nós sensores configurados e ativos, os dados de configuração de cada um deles (estação, tópico de publicação) e o acesso ao tópico MQTT diretamente.

Do mesmo modo que os demais serviços da aplicação, os *web services* ficarão implantados em um servidor de aplicação Java.

5.6 BANCOS DE DADOS CORPORATIVOS

Os dados publicados pelos nós sensores são persistidos em um banco de dados já existente na empresa; a aplicação descrita anteriormente na subseção 5.5: Aplicação de Configuração, Processamento e Integração (Application), irá processar os dados e realizar a persistência por meio de uma conexão com o servidor de bancos de dados corporativos.

5.7 APLICAÇÃO ANDROID (APP)

É a aplicação para smartphones Android que poderá ser utilizada por funcionários da fábrica no monitoramento direto dos nós sensores. O aplicativo é cliente da aplicação descrita anteriormente na subseção 5.5: Aplicação de Configuração, Processamento e Integração (Application), à qual terá acesso por

meio da internet utilizando a API disponível e descrita anteriormente na subseção 5.5.2: RESTful WS.

Ao ser configurado para acessar a API implantada, o aplicativo exibirá a lista de nós sensores já configurados e ativos, com seus dados de configuração e instalação. O usuário poderá selecionar um nó para ser monitorado, ação que fará com que o aplicativo se inscreva ao tópico MQTT daquele sensor no *broker*.

Além de uma interface cliente RESTful, o aplicativo também utilizará o componente Paho Android Service, que é um cliente MQTT, além de outros componentes do Android SDK 27, como Activity, ContentProvider, SQLite, RecyclerView, Adapter e Notifications.

5.8 SISTEMA DE CONTROLE DA QUALIDADE

É o sistema responsável pela emissão do relatório exigido pela agência reguladora, e que será encaminhado ao cliente junto ao lote produzido. O sistema irá acessar os dados capturados pelos sensores por meio da API RESTful da aplicação.

6 CONCLUSÃO

O estudo e trabalho com Internet das Coisas mesmo tendo referência a quase vinte anos atrás ainda é algo novo no âmbito globalizado, são muitas tecnologias e muitos padrões na disputa para se tornar o oficial e essa disputada está em todos os níveis que envolvem essa área sejam eles software, hardware ou transmissão.

A interoperabilidade, ou seja, a competência de um sistema trabalhar em conjunto com um ou mais sistemas diferentes, é a chave, ter sistemas e tecnologias que se comunicam de forma efetiva e que sejam simples ao usuário garantindo a expansão da tecnologia.

Padrões como os apresentados ao longo deste trabalho, tais como o IEEE 802.15.4 e o 6LoWPAN são os exemplos de esforços para fazer algo com eficiência e que atenda diversas demandas e situações, mas muitas vezes um *lobby* ou uma penetração de mercado de uma marca ou modelo acaba por determinar um “vencedor”.

Independente da forma que ocorrer essa padronização e evolução o IoT já chegou e está cada vez mais presente na vida das pessoas e na indústria, mesmo que de forma indireta. É um campo sem volta que apenas irá progredir e que deixará a vida de todos nós mais dinâmica e interativa. É uma área de estudos até o momento sem limites para alcançar e as futuras pesquisas na área podem se dedicar as diversas esferas dessa tecnologia, como por exemplo o aprimoramento de softwares que funcionem tendo em vista a interoperabilidade.

REFERÊNCIAS

ARDUINO (2018). **Arduino: Introduction**. Copyright© 2018 Arduino. Disponível em: <<http://arduino.cc/en/Guide/Introduction>>. Acesso em: 2 out. 2018.

ASHTON, Kevin. **Kevin Ashton: entrevista exclusiva com o criador do termo “Internet das Coisas”**. Financiadora de Inovação e Pesquisa (FINEP), entrevista publicada em: 13 jan. 2015. Disponível em: <<http://finep.gov.br/noticias/todas-noticias/4446-kevin-ashton-entrevista-exclusiva-com-o-criador-do-termo-internet-das-coisas>>. Acesso em: 30 out. 2018.

BERNARDES, Rodrigo Camargos; BERNARDINO, Marco Antônio Gomes; CHIESORIN, Samuel. Proposta de projeto de disciplina: **Rede de sensores sem fio para monitoramento de estações embaladoras em linhas de produção de material médico e cirúrgico**. Universidade Tecnológica Federal do Paraná, Curso de Especialização em Internet das Coisas, Disciplina: INC10 - Projeto de Redes de Sensores Sem Fio. Curitiba, 2018.

CISCO (2006). **IPv6 extension headers review and considerations**. Copyright© 1992-2006 Cisco Systems, Inc. Disponível em: <https://www.cisco.com/en/US/technologies/tk648/tk872/technologies_white_paper0900aecd8054d37d.html>. Acesso em: 4 nov. 2018.

CONTIKI (2018). **Contiki: The open source OS for the Internet of Things**. Disponível em: <www.contiki-os.org/>. Acesso em: 2 out. 2018.

GUTIERREZ, José A. **IEEE std. 802.15.4 enabling pervasive wireless sensor networks**. Copyright© 2005 Eaton Corporation, publicado em: 04 dez. 2005. Disponível em: <<https://people.eecs.berkeley.edu/~prabal/teaching/cs294-11-f05/slides/day21.pdf>>. Acesso em: 07 set. 2018.

JAIN, Raj. **CSE570S: Internet of Things protocols and standards**. Washington University. Saint Lois. Professor of Computer Science and Engineering. 2015. Disponível em: <https://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot/>. Acesso em: 07 set. 2018.

JAIN, Raj. **CSE574S: Wireless and mobile networking**. Washington University. Saint Lois. Professor of Computer Science and Engineering. 2016. Disponível em: <<https://www.cse.wustl.edu/~jain/cse574-16/>>. Acesso em: 07 set. 2018.

MOMOTE, Victor. **Modelos de comunicação para IoT**. Copyright© Embarcados, publicado em: 19 mai. 2016. Disponível em: <<https://www.embarcados.com.br/modelos-de-comunicacao-para-iot/>>. Acesso em: 25 out. 2018.

MONTENEGRO, Gabriel; et al. **Transmission of IPv6 packets over IEEE 802.15.4 networks**. Network Working Group, publicado em: set. 2007. Disponível em: <<http://tools.ietf.org/html/rfc4944>>. Acesso em: 20 set. 2018.

MOREIRAS, Antônio M. **ZigBee usa agora 6LoWPAN! Sua próxima lâmpada terá IPv6?** Blog IPv6.br, publicado em: 30 mar. 2013. Disponível em: <<http://ipv6.br/post/zigbee-usa-agora-6lowpan-sua-proxima-lampada-tera-ipv6/>>. Acesso em: 20 set. 2018.

MULLIGAN, Geoff. **IEEE talks IoT**. Entrevista com Geoff Mulligan. Copyright© 2018 IEEE Internet of Things. 2018. Disponível em: <<https://iot.ieee.org/articles-publications/ieee-talks-iot/166-ieee-talks-iot-geoff-mulligan.html>>. Acesso em: 10 set. 2018.

NARAYANA, Anireekshith; SHARMA, Tarun; VEETURI, Ramesh Chandra. **A comprehensive analysis of 6LoWPAN for IoT**. International Journal for Research Trends and Innovation (IJRTI), v. 3, n. 8, 2018. p 152-155. Disponível em: <<http://www.ijrti.org/papers/IJRTI1808024.pdf>>. Acesso em: 30 out. 2018.

RAYMOND, Gerson. **O que é TinyOS?** Viva o Linux, publicado em: 2 dez. 2007. Disponível em: <<https://www.vivaolinux.com.br/artigo/TinyOS>>. Acesso em: 20 out. 2018.

RIOT (2018). **RIOT: The friendly operating aystem for the Internet of Things**. Página oficial do Sistema Operacional RIOT. Disponível em: <<https://riot-os.org/>>. Acesso em: 17 out. 2018.

TSCHOFENIG, Hannes; et al. **Architectural considerations in smart object networking**. Copyright© 2015 IETF Trust and the persons identified as the document authors, publicado em: mar. 2015. Disponível em: <<https://www.rfc-editor.org/rfc/pdf/rfc/rfc7452.txt.pdf>>. Acesso em: 30 out. 2018.