INSTITUTO FEDERAL DE SANTA CATARINA

VITOR MANOEL DA SILVEIRA

# SISTEMA DE MANUTENÇÃO PREDITIVA UTILIZANDO REDES BLE E WI-FI COM APRENDIZADO DE MÁQUINA

São José - SC

dezembro/2018

# SISTEMA DE MANUTENÇÃO PREDITIVA UTILIZANDO REDES BLE E WI-FI COM APRENDIZADO DE MÁQUINA

Trabalho de conclusão de curso apresentado à Coordenadoria do Curso de Engenharia de Telecomunicações do campus São José do Instituto Federal de Santa Catarina para a obtenção do diploma de Engenheiro de Telecomunicações.

Orientador: Mario de Noronha Neto

São José - SC

dezembro/2018

# RESUMO

Atualmente, um dos principais problemas encontrados nas grandes indústrias são os defeitos apresentados em equipamentos cujo valor monetário é significativamente alto, resultando em grandes custos por interrupção na produção e reposição de peças. Para evitar tal situação e ter certa previsibilidade de quando os problemas ocorrerão, cada vez mais empresas estão procurando integrar seus equipamentos às redes *Internet of things* (IoT) devido aos diversos benefícios que esta traz quando aliada à técnicas de aprendizado de máquina e *Condition-based maintenance* (CBM). Neste documento consta uma introdução sobre o tema e a fundamentação teórica para sustentar a proposta de implementar um sistema que colete informações do motor elétrico de equipamentos industriais e, com base nisto, gere um alerta para que a manutenção seja efetuada previamente, evitando danos aos equipamentos, paradas desnecessárias na produção e aumentando a vida útil das máquinas.

**Palavras-chave**: Condition-based maintenance. Machine learning. Bluetooth low energy.

# LISTA DE ILUSTRAÇÕES

Figura 1 – Procedimento para a abordagem da CBM . . . . . 14
Figura 2 – Canais de comunicação utilizados pelo *Bluetooth low energy* (BLE). Os canais 37, 38 e 39 são os de anúncio, sendo que os demais são utilizados para a transferência de dados. . . . . 17
Figura 3 – Pilha de protocolos da tecnologia BLE. . . . . 18
Figura 4 – Exemplo de funcionamento do algoritmo *k-means*. . . . . 22
Figura 5 – Diagrama do sistema proposto . . . . . 23
Figura 6 – Componentes do *endpoint* . . . . . 24
Figura 7 – *Gateway* ESP32 . . . . . 25

# LISTA DE TABELAS

Tabela 1 – Cronograma das atividades previstas . . . . . 25

# LISTA DE ABREVIATURAS E SIGLAS

**AFH** *Adaptative frequency hopping* . . . . . 17

**ATT** *Attribute Protocol* . . . . . 19

**AWS** *Amazon web services* . . . . . 23

**BLE** *Bluetooth low energy* . . . . . 3

**CBM** *Condition-based maintenance* . . . . . 2

**CRC** *Cyclic redundancy code* . . . . . 17

**FFT** *Fast Fourier transform* . . . . . 14

**FHSS** *Frequency-hopping spread spectrum* . . . . . 17

**GAP** *Generic Access Profile* . . . . . 18

**GATT** *Generic Attribute Profile* . . . . . 19

**GFSK** *Gaussian frequency shift keying* . . . . . 17

**HCI** *Host Controller Interface* . . . . . 19

**IoT** *Internet of things* . . . . . 2

**L2CAP** *Logical Link Control and Adaptation Protocol* . . . . . 19

**LL** *Link Layer* . . . . . 19

**LPWAN** *Low power wide area network* . . . . . 15

**LTE** *Long term evolution* . . . . . 16

**M2M** *Machine-to-machine* . . . . . 15

**Msps** Mega símbolos por segundo . . . . . 16

**NFC** *Near Field Communication* . . . . . 15

**PDU** *Protocol data unit* . . . . . 17

**PHY** *Physical Layer* . . . . . 20

**PSK** *Phase shift keying* . . . . . 17

**SIG** *Special interest group* . . . . . 18

**SMP** *Security Manager Protocol* . . . . . 19

**UUID** *Universally Unique Identifier* . . . . . 19

# SUMÁRIO


**1 INTRODUÇÃO . . . . . 11**
**1.1 Objetivos . . . . . 12**

**2 FUNDAMENTAÇÃO TEÓRICA . . . . . 13**
**2.1 *Condition-based maintenance* (CBM) . . . . . 13**
**2.2 *Internet of things* (IoT) . . . . . 15**
2.2.1 Tecnologias . . . . . 15
**2.3 *Bluetooth low energy* (BLE) . . . . . 16**
2.3.1 Parâmetros da tecnologia . . . . . 16
2.3.2 Pilha de protocolos . . . . . 18
**2.4 Aprendizado de máquina . . . . . 20**
2.4.1 Regressão linear . . . . . 21
2.4.2 *K-means* . . . . . 21

**3 PROPOSTA . . . . . 23**

**REFERÊNCIAS . . . . . 27**

# 1 INTRODUÇÃO

O mercado está cada vez mais competitivo, o que resulta em uma busca contínua por parte das industrias em melhorar o seu processo produtivo e otimizar ao máximo os custos. Para tal, não basta desenvolver produtos de alta qualidade ou prestar um serviço com excelência, é preciso também reduzir os gastos, o que pode ser alcançado evitando paradas desnecessárias na linha de produção devido à problemas em equipamentos.

Tempos atrás, sem o auxílio da tecnologia, os equipamentos industriais eram substituídos apenas quando apresentavam algum defeito. Depois, passou-se a serem feitas manutenções agendadas, de acordo com um certo período de tempo, o que dá uma sobrevida às máquinas. Com o decorrer dos anos, o avanço tecnológico das comunicações sem fio e da Internet tornou possível prever, com certa antecedência, quando um equipamento apresentará algum problema. Isso faz com que o equipamento só seja parado quando realmente é necessário e permite que a manutenção seja efetuada logo nos primeiros indícios de defeitos.

Neste meio, a CBM visa possibilitar a identificação e o reparo de problemas em equipamentos, resultando em uma maior durabilidade e proporcionando uma grande redução de custos, tendo em vista que boa parte destes possuem alto valor agregado. De forma geral, a CBM pode ser tratada como um método usado para reduzir a incerteza das atividades de manutenção e é realizada de acordo com as condições dos equipamentos (SHIN; JUN, 2015).

Além disso, como os equipamentos industriais possuem certa complexidade e sua manutenção necessita de conhecimentos específicos, deseja-se que os dispositivos responsáveis pela coleta dos dados possam ser instalados de forma não intrusiva, ou seja, acoplados na parte externa dos equipamentos dos quais coletarão as informações, tornando assim muito mais simples a instalação, evitando paradas desnecessárias da produção e a desmontagem dos equipamentos.

Contudo, levando em consideração o fato de que os sensores são os responsáveis pela obtenção de um determinado dado de interesse, é preciso fornecer algum meio de comunicação para que tais dados possam ser coletados e analisados remotamente. Com o avanço das tecnologias de comunicação sem fio, existem diversas opções que podem atender essa demanda por conectividade: ZigBee, Wi-Fi, LoRa, Sigfox, BLE, entre outras. Cada uma dessas tecnologias possui características e benefícios específicos, sendo necessária uma avaliação do cenário para que a melhor decisão seja efetuada.

Atualmente, existem no mercado empresas que se propõem a fornecer um sistema completo para a obtenção, envio, armazenamento e análise dos dados, como é o caso da Bluvision (2018), que oferece três linhas de produtos para o monitoramento das condições dos equipamentos: os *endpoints* (chamados de *beacons*) para a coleta dos dados, os *gateways* (chamados de *BluFi*) para o encaminhamento dos dados provenientes da rede BLE para a rede Wi-Fi e a plataforma *online Bluzone*, que armazena, gerencia e analisa os dados.

Contudo, é importante ressaltar que este projeto não visa competir com os sistemas

já comercializados atualmente, mas sim demonstrar a viabilidade da implementação de um sistema composto por componentes de baixo custo e *softwares* livres, possibilitando uma maior flexibilidade e customização, de acordo com o desejado em cada cenário.

Este documento está organizado da seguinte maneira: no Capítulo 2 é apresentada a fundamentação teórica do trabalho que serve de base para a proposta que, por sua vez, é detalhada no Capítulo 3.

## 1.1 Objetivos

Este trabalho tem como principal objetivo desenvolver um sistema de manutenção preditiva utilizando técnicas de aprendizado de máquina com base em dados coletados por sensores de vibração (acelerômetro) conectados à uma rede IoT composta pelas tecnologias BLE e Wi-Fi.

Para atingir tal meta, definiu-se os seguintes objetivos específicos:

- Desenvolvimento dos *endpoints* e *gateway*;
- Implementação e configuração das redes BLE e Wi-Fi;
- Desenvolvimento do servidor em nuvem para armazenamento e gerenciamento dos dados (*backend*);
- Aplicação de técnicas de aprendizado de máquina para predição de falhas em motores elétricos;
- Desenvolvimento de uma interface *web* para acompanhamento dos dados coletados e recebimento dos alertas em casos de detecção de anomalias.

# 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados os temas que servem como base teórica para que o objetivo proposto possa ser atingido. Na seção 2.1 é descrito o conceito de CBM e são citadas algumas aplicações em que ela pode ser inserida. Já na seção 2.2 é feita uma revisão sobre as redes IoT e as tecnologias que podem ser utilizadas para compô-la. Na seção 2.3 são detalhados os parâmetros e a pilha de protocolos de tecnologia BLE, que foi a escolhida para prover a conectividade para os dispositivos neste projeto. E, por fim, a seção 2.4 apresenta uma visão geral sobre aprendizado de máquina e alguns algoritmos das principais técnicas utilizadas para manutenção preditiva.

## 2.1 *Condition-based maintenance* (CBM)

No contexto das industrias, existem algumas categorias de manutenção de equipamentos, sendo que as três principais serão abordadas nesta sessão.

A primeira, mais simples e comum, é a manutenção corretiva, que ocorre apenas após o equipamento ter apresentado alguma falha, sendo que a ação tomada é o conserto ou a troca.

A segunda, também bastante utilizada, é conhecida como manutenção preventiva, que tem como premissa a checagem dos equipamentos a cada determinado período de tempo. Desta forma, é possível efetuar a troca e/ou reparo de peças desgastadas, garantindo um aumento na vida útil do equipamento.

A terceira e última, *Condition-based maintenance* (CBM), pode ser semelhante à manutenção preventiva, no sentido de que seu objetivo é prevenir anomalias no equipamento antes que elas ocorram. Contudo, a abordagem da CBM é diferente, pois o foco deste método é prever o processo de degradação de um produto, que baseia-se na hipótese de que a maioria dos comportamentos anômalos não ocorrem instantaneamente, mas sim de forma gradual. Sendo assim, diferentemente das categorias acima, a CBM não foca somente na detecção de falhas e diagnóstico dos componentes, mas também no monitoramento da degradação e predição de falhas do equipamento (SHIN; JUN, 2015).

O conceito de CBM foi criado pela empresa *Rio Grande Railway Company* no final da década de 40 e foi inicialmente chamado de "manutenção preditiva". A companhia ferroviária utilizava as técnicas de CBM para detectar vazamentos de fluídos refrigerantes, óleo e combustível do motor, analisando mudanças de temperatura e pressão. Tais técnicas tiveram grande sucesso em termos de reduzir os impactos de falhas não planejadas e identificar quando consertar um vazamento ou reabastecer um fluído ou o reservatório de óleo (PRAJAPATI; BECHTEL; GANESAN, 2012).

Desde então, diversas definições surgiram com o intuito de descrever o que é a CBM. Para os autores Shin e Jun (2015), que citam várias dessas definições em seu artigo, *Condition-based maintenance* (CBM) pode ser considerada uma política de manutenção que executa uma ação de

manutenção antes que as falhas no equipamento aconteçam, por meio da avaliação das condições do equipamento e do ambiente operacional, prevendo o risco de falhas em tempo real com base nos dados coletados do equipamento.

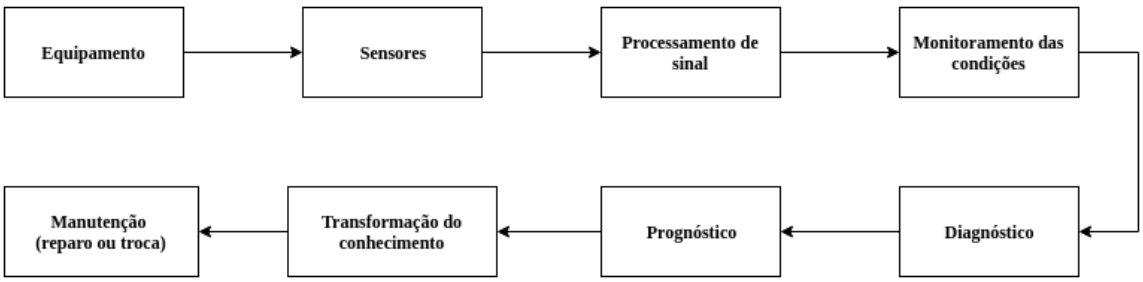
As técnicas de CBM possuem uma ampla área de aplicação, como em fábricas de manufatura, industrias, setores militar e naval, veículos da força aérea, infraestrutura de tecnologia e informação, veículos comerciais e aviação (PRAJAPATI; BECHTEL; GANESAN, 2012).

Dentre estes exemplos, os autores Prajapati, Bechtel e Ganesan (2012) destacam que, no contexto das industrias, a CBM é uma abordagem alternativa que reduz significativamente o tempo de inatividade, prevendo falhas de forma antecipada com base na análise de vibração, sendo capaz de, inclusive, determinar a vida útil restante do equipamento. Esta vibração é obtida por meio de sensores que podem ser montados em qualquer peça crítica, como motores, ventiladores, caixas de câmbio, bombas, rolamentos e outros meios rotativos. Com o decorrer do tempo, a deterioração do equipamento aumenta e o padrão de vibração muda, o que pode ser facilmente detectado por uma análise de *Fast Fourier transform* (FFT) da vibração de um determinado componente.

É importante ressaltar que as técnicas de CBM não são aplicadas apenas com a observação da vibração de um equipamento. Esta análise é apenas uma forma de fazer isto. Contudo, existem diversas outras, como por exemplo o monitoramento de temperatura, tensão e corrente elétrica, entre outras.

Segundo Shin e Jun (2015), a Figura 1 exibe um procedimento genérico para a implementação da CBM.

Figura 1 – Procedimento para a abordagem da CBM




Fonte: Adaptada de Shin e Jun (2015).

Com isso, esta técnica nos permite identificar e resolver possíveis defeitos nos produtos antes mesmos de ocorrerem. Tendo em vista que qualquer problema em equipamentos industriais podem resultar em grandes perdas, a CBM é um método bastante atrativo para as industrias em geral.

## 2.2 *Internet of things* (IoT)

Cada vez mais, um crescente número de dispositivos eletrônicos vêm se conectando à Internet, idealizando o conceito das redes IoT. Um exemplo muito comum são as casas inteligentes, que possuem diversos equipamentos conectados entre si, capazes de monitorar o ambiente e tomar decisões de forma autônoma, de acordo com os dados coletados.

Existem também outros ambientes em que as redes IoT podem ter um ótimo desempenho e facilitar as nossas vidas. Essas aplicações incluem o transporte, saúde, situações de emergência provocadas por desastres naturais e até mesmo pelo homem, além da automação industrial. Desta forma, espera-se que com o passar do tempo, as redes IoT tenham ainda mais impacto, aumentado a qualidade de vida das pessoas e fazendo crescer a economia mundial (AL-FUQAHA et al., 2015).

Ainda segundo Al-Fuqaha et al. (2015), estima-se que até o final de 2020 existam mais de 212 bilhões de dispositivos IoT espalhados pelo mundo. Mais além, em 2022, a previsão é de que 45% de todo o tráfego da Internet seja constituído por comunicações *Machine-to-machine* (M2M) que, para Xiao et al. (2016), é o termo utilizado para tecnologias que permitem as máquinas e os dispositivos móveis estabelecerem enlaces entre si de forma autônoma, visando a troca de informações.

Aplicação foco deste trabalho e citada por Samie, Bauer e Henkel (2016), as industrias inteligentes também contam com as redes IoT, que proporcionam soluções para automação, controle e monitoramento objetivando reduzir o custo de operação e manutenção, além de aumentar a qualidade do serviço oferecido. Um exemplo disso é o monitoramento remoto das máquinas para realizar a *Condition-based maintenance* (CBM).

### 2.2.1 Tecnologias

Por ser uma rede muito ampla e possuir tantas aplicações, é preciso levar em conta alguns parâmetros para definir a melhor tecnologia de comunicação a ser utilizada em um determinado cenário. Esse ponto também é abordado por Samie, Bauer e Henkel (2016), onde é feita uma breve descrição das seguintes tecnologias:

- ***Near Field Communication* (NFC)**: É uma tecnologia de comunicação sem fio de pequeno alcance, que permite a troca de informações entre dispositivo fisicamente próximos (aproximadamente 20 centímetros). Os dados são armazenados em *tags*, sendo que estas podem ser apenas para leitura ou regraváveis.

- ***Low power wide area network* (LPWAN)**: São tecnologias que focam no baixo consumo energético e transmissão em altas distância. Chegam a suportar mais de 10 quilômetros de distância entre o *gateway* e os *endpoints*. Contudo, as taxas de transmissão de dados são extremamente baixas, geralmente menores do que 1 Kbps. Além disso, operam em faixas de frequências próximas de GHz, cujo espectro é amplamente utilizado globalmente.

- **Redes celular**: Redes móveis muito difundidas, como o 3G e o *Long term evolution* (LTE), que provêm altas taxas de transmissão. Contudo, em geral, exigem alto consumo energético.
- **Wi-Fi**: As principais vantagens são a grande largura de banda e sua disponibilidade nas áreas urbanas. Também conta com alto consumo de potência.
- **ZigBee**: Rede de baixo custo, baixo consumo energético e fisicamente pequena capaz de suportar diferentes topologias, como por exemplo, *mesh*, árvore e estrela. Oferece amplo alcance de transmissão, dependendo da potência utilizada. Contudo, enfrenta algumas barreiras no mercado, principalmente relacionadas a uma tecnologia emergente, que conta com maior largura de banda e um menor custo de energia, conhecida como BLE.
- ***Bluetooth low energy* (BLE)**: A tecnologia *Bluetooth* já é bastante conhecida e muito utilizada em diversos dispositivos eletrônicas, para as mais variadas finalidades. Oferece taxas de transferência e largura de banda suficientes para o tráfego de *streams*, como o áudio. Porém, possui diversas limitações quanto ao número de nodos conectados à rede (um dispositivo mestre e até sete escravos). A tecnologia BLE, também conhecida como *Bluetooth Smart*, foi desenvolvida para alcançar curtas distâncias, utilizar uma pequena largura de banda e obter baixas latências para aplicações IoT. As vantagens do BLE sobre o *Bluetooth* convencional (também chamado de *Bluetooth* Clássico) incluem um menor consumo de potência, menor tempo de configuração dos dispositivos e o suporte às topologias estrela e *mesh* com um ilimitado número de nodos.

Tendo em vista que a aplicação deste trabalho não requer uma ampla cobertura da rede e nem altas taxas de transmissão, a tecnologia de comunicação escolhida foi o BLE, por ter um baixo consumo energético e permitir a conexão de um número ilimitado de nodos em uma mesma rede, além do fato de que tanto os *endpoints* quanto os *gateways* possuem baixo custo.

Na sessão a seguir, serão abordadas, de maneira mais detalhada, as características e propriedades da tecnologia BLE, escolhida para prover a conectividade neste projeto.

## 2.3 *Bluetooth low energy* (BLE)

Anunciada pela primeira vez em 2010, a tecnologia BLE tem como principal objetivo expandir o uso do Bluetooth para aplicações que possuem restrições de energia, como controles e sensores *wireless*. Também são características dessas aplicações a baixa quantidade de dados transmitidos e a comunicação que ocorre com pouca frequência. Essas são as maiores diferenças se comparadas às aplicações do Bluetooth convencional, como a transmissão de áudio e dados, que contam com grande quantidade de dados e frequente interação entre os dispositivos de comunicação.

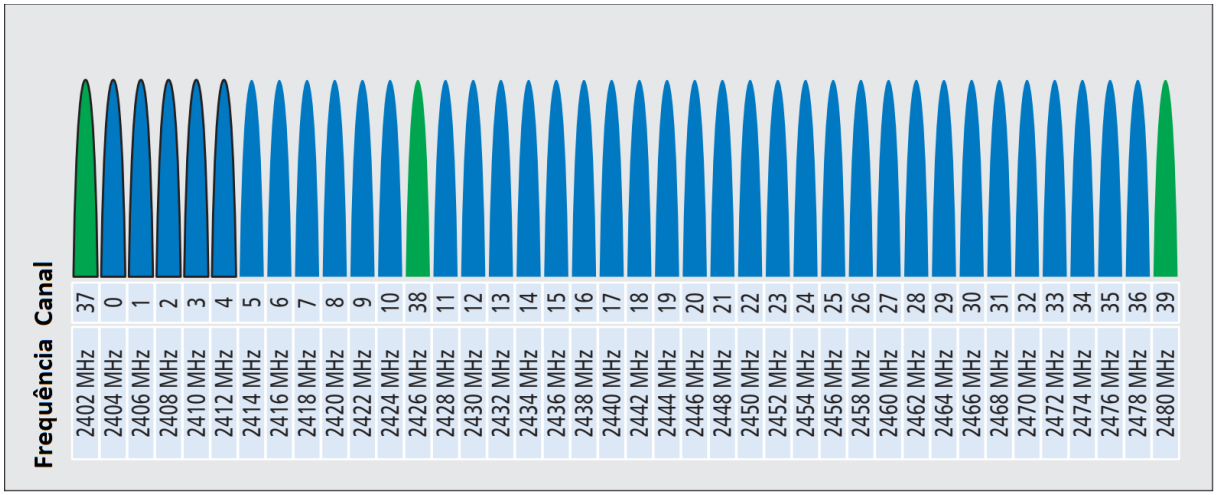
### 2.3.1 Parâmetros da tecnologia

O Bluetooth Clássico conta com 79 canais de comunicação, cuja largura de banda de cada um é de 1 MHz e a taxa de símbolos bruta é de 1 Mega símbolos por segundo (Msps),

esquema de modulação utilizado pode ser a *Gaussian frequency shift keying* (GFSK), 4-*Phase shift keying* (PSK) ou 8-PSK. Para o BLE, a modulação é a GFSK - o que, de acordo com Čabarkapa, Grujić e Pavlović (2015), reduz os picos de consumo de energia, a taxa bruta de símbolos também é de 1 Msps, porém, cada canal possui 2 MHz de largura de banda. Enquanto a técnica *Frequency-hopping spread spectrum* (FHSS) ainda é usada no BLE para mitigar os efeitos da interferência, o tempo de permanência em cada faixa de frequência é maior do que no Bluetooth Clássico, o que torna mais flexíveis os requisitos de tempo para o BLE (CHANG, 2014).

Devido ao aumento da largura de banda, no BLE existem 40 canais de comunicação, sendo três deles para anúncio e o restante para transferência de dados. A Figura 2 ilustra como são distribuídos os canais do BLE em função das faixas de frequência.

Figura 2 – Canais de comunicação utilizados pelo BLE. Os canais 37, 38 e 39 são os de anúncio, sendo que os demais são utilizados para a transferência de dados.




Fonte: Adaptada de Chang (2014).

Os canais de anúncio têm a função de facilitar a descoberta e o estabelecimento da comunicação entre dois dispositivos, o que inclui a negociação dos parâmetros a serem utilizados. Uma vez conectados, a transferência das informações é feita através dos canais de dados. Como a robustez dos canais de anúncio é de extrema importância para a inicialização da comunicação e, tendo em vista que o BLE ocupa algumas faixas de frequência utilizadas pelo Wi-Fi, tais canais são escolhidos para operarem nas bandas que sofrem menos interferências pelas redes Wi-Fi.

Além disso, para evitar a interferência e o desvanecimento com outras comunicações sem fio na mesma banda de frequência, como o Wi-Fi, o BLE implementa o *Adaptative frequency hopping* (AFH), que é responsável por definir, de maneira psuedo-aleatória, o canal de comunicação a ser utilizado pelos dispositivos (TOSI et al., 2017) .

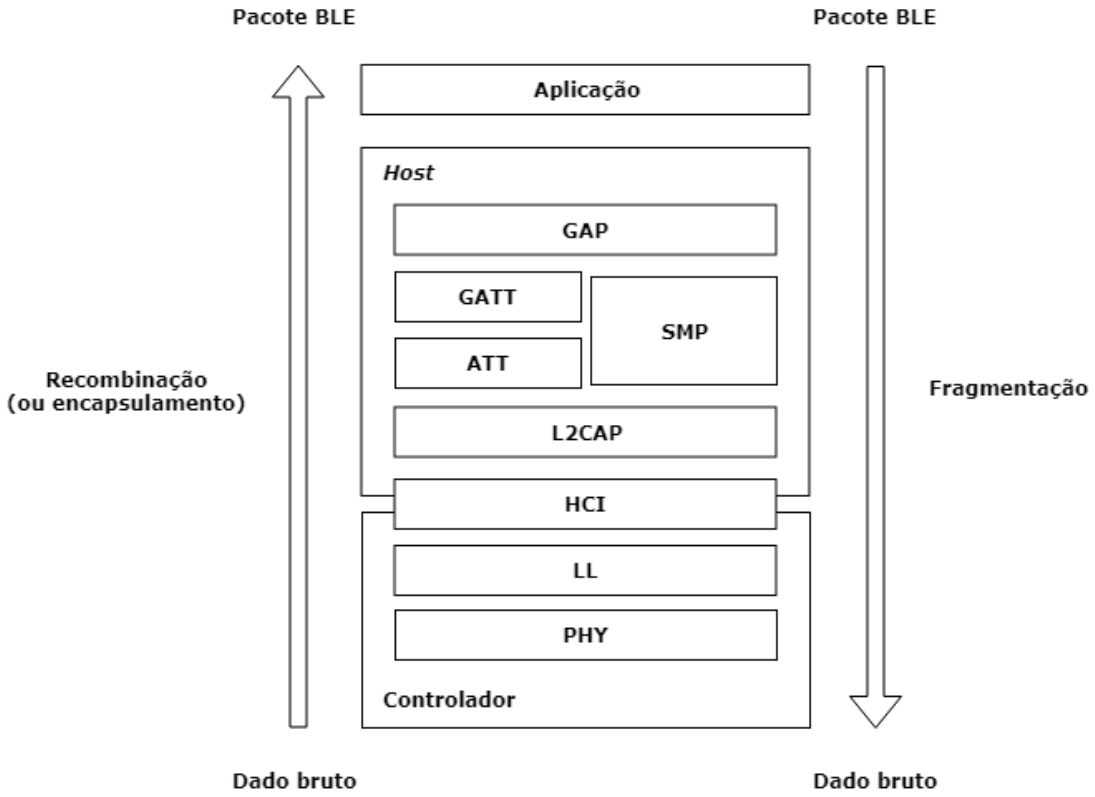
Segundo Chang (2014), o BLE foi projetado para que dois dispositivos possam se conectar rapidamente, já que os pacotes de anúncio transmitidos possuem apenas 1 byte de preâmbulo, 4 bytes de códigos de acesso, 3 bytes de *Cyclic redundancy code* (CRC) e um *Protocol data unit* (PDU) contendo de 2 a 39 bytes, sendo que, o menor pacote pode ser transmitido em 80 $\mu$s e o maior em 300 $\mu$s. Com um pequeno tamanho de pacote, o BLE atende aos requisitos de

baixo *duty-cycle*, o que é muito importante para aplicações de redes de sensores sem fio.

### 2.3.2 Pilha de protocolos

A pilha de protocolos da tecnologia BLE é composta por três blocos principais, conforme mostra a Figura 3.

Figura 3 – Pilha de protocolos da tecnologia BLE.




Fonte: Adaptada de Tosi et al. (2017).

De acordo com Tosi et al. (2017), estes blocos e suas camadas podem ser descritos da seguinte forma:

- A **Aplicação** é o bloco mais alto da pilha e representa a interface direta com o usuário. É ela quem define alguns perfis que permitem a interoperabilidade entre diferentes aplicações, os quais são definidos pelo *Bluetooth Special interest group* (SIG). A especificação *Bluetooth* também permite estabelecer perfis específicos de fornecedores para casos de uso não cobertos pelos padrões definidos pelo SIG.

- O ***Host*** inclui as seguintes camadas:

  – *Generic Access Profile* (GAP): Faz interface diretamente com a camada de aplicação e, portanto, com o usuário, que pode definir todos os parâmetros que a rede necessita. É o protocolo responsável por implementar e controlar todas as camadas inferiores da pilha. Além disso, especifica papéis, modos e procedimentos do dispositivo, bem como o de gerenciar o estabelecimento da conexão e a segurança.

- *Generic Attribute Profile* (GATT): encapsula a camada *Attribute Protocol* (ATT) e sua função principal é estabelecer como trocar informações e dados de todos os perfis em um enlace BLE. Os perfis são definições de possíveis aplicações e especificam comportamentos gerais que os dispositivos *BLE* usam para se comunicar com outros dispositivos *BLE*, além de definir mais claramente que tipo de dados um módulo *BLE* está transmitindo. Esses dados são organizados em uma estrutura hierárquica composta de seções chamadas *serviços*, que, por sua vez, agrupam os dados em contêineres chamados de *características*.

  O GATT define as duas funções, cliente e servidor, em uma conexão. O *Bluetooth* SIG define alguns *serviços* e *características* padrão, representados em um formato de endereço de 16 bits, mas o BLE permite que os fabricantes definam seus próprios serviços usando um *Universally Unique Identifier* (UUID) de 128 bits para adaptar a tecnologia à novas e originais aplicações. Durante o estabelecimento da conexão, o servidor informa ao cliente quais são os seus *serviços* e *características* com o objetivo de definir como a conexão será estruturada.

  Um *serviço* é basicamente um contêiner que conceitualmente agrupa os atributos relacionados, enquanto as *características* são os atributos incluídos em um serviço, e cada um deles é usado para comunicar um tipo específico de dados, como por exemplo, temperatura, umidade, corrente elétrica, vibração, etc...

- *Logical Link Control and Adaptation Protocol* (L2CAP): é um protocolo que atua como um multiplexador; ele manipula os dados das camadas inferiores e os encapsula no formato de pacote padrão, de acordo com as camadas superiores e vice-versa. Estes processos são chamados respectivamente de recombinação (ou encapsulamento) e fragmentação,

- *Attribute Protocol* (ATT): define as funções de uma arquitetura cliente-servidor, onde o cliente é aquele que solicita dados do servidor, que, por sua vez, envia dados para os clientes. Normalmente, essas funções correspondem respectivamente ao mestre e ao escravo definidos na camada de elance. O ATT também executa a organização de dados em atributos, aos quais é atribuído um índice, um UUID, um conjunto de permissões e um valor. Esse protocolo é encapsulado no GATT, que usa as funções definidas no ATT para realizar conexões.

- *Security Manager Protocol* (SMP): cuida do emparelhamento de dispositivos, da distribuição das chaves de segurança e é projetado para minimizar os requisitos de recursos para os dispositivos escravos, deslocando a carga de trabalho para dispositivos mestres que possuem maior capacidade de processamento.

- *Host Controller Interface* (HCI): responsável por padronizar a comunicação entre o controlador e o *host*.

- O **Controlador** é implementado em *hardware* e composto pelas camadas:

  - *Link Layer* (LL): a camada de enlace provê o acesso ao meio, controle de erro e de fluxo e o estabelecimento da conexão entre os dispositivos.

– *Physical Layer* (PHY): a camada física é responsável pela transmissão e recepção dos bits.

No contexto deste trabalho, os módulos BLE (servidores) serão utilizados para transferir os dados coletados a partir do sensor de vibração para o *gateway* BLE (cliente) que, por sua vez, encaminhará os dados para a nuvem, onde serão inseridos em um modelo de aprendizado de máquina, afim de analisá-los e alertar os responsáveis, caso os dados representem um comportamento anômalo.

## 2.4 Aprendizado de máquina

Uma definição formal de aprendizado de máquina proposta pelo cientista da computação Tom M. Mitchell afirma que uma máquina aprende sempre que é capaz de utilizar sua experiência de tal forma que seu desempenho melhore em experiências semelhantes no futuro. Apesar de esta definição ser intuitiva, ela ignora completamente o processo de como exatamente uma experiência pode ser traduzida em ação futura. Embora o cérebro humano seja naturalmente capaz de aprender desde o nascimento, as condições necessárias para que os computadores aprendam devem ser explicitadas (LANTZ, 2015).

De acordo com esta definição, podemos concluir que as técnicas de aprendizado de máquina podem ser utilizadas para ensinar um computador a tomar decisões futuras baseadas em experiências semelhantes passadas e em um determinado conjunto de dados. Para Amruthnath e Gupta (2018), tais técnicas podem ser classificadas como supervisionadas, não supervisionadas, semi-supervisionadas e por reforço, sendo que as mais comumente utilizadas para manutenção preditiva são as supervisionadas e não-supervisionadas.

As técnicas de aprendizado supervisionado são utilizadas para treinar um modelo preditivo que, por sua vez, tem como objetivo prever um valor usando outros valores de um mesmo conjunto de dados que possui uma ou mais variáveis independentes (entradas) e uma ou mais variáveis dependentes (saídas), sendo que tanto as entradas quanto as saídas são conhecidas. Estas técnicas podem ser aplicadas em problemas de regressão, onde se deseja prever dados numéricos, ou em problemas de classificação, para prever categorias (LANTZ, 2015). O termo ”supervisionado” se deve ao fato de que o treinamento de um algoritmo com um determinado conjunto de dados pode ser considerado como um professor que supervisiona o processo de aprendizado. Afinal, como as respostas corretas são conhecidas, o algoritmo faz iterativamente previsões sobre os dados de treinamento e é corrigido pelo professor. O ciclo só é encerrado quando o algoritmo obtém um desempenho aceitável. Geralmente, os modelos de classificação de aprendizagem supervisionada devem executar as seguintes etapas: coleta de dados, extração de características, seleção de um algoritmo de aprendizado de máquina, construção do modelo usando o algoritmo selecionado e, por fim, avaliação da precisão do algoritmo (ALFIAN et al., 2018).

Por outro lado, as técnicas de aprendizado não-supervisionado são aplicadas no treinamento de modelos descritivos, cujo conjunto de dados é caracterizado por não fornecer variáveis de saída. Portanto, faz parte do aprendizado realizar inferências a partir das variáveis de entradas para determinar padrões (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Tais técnicas podem

ser aplicadas em problemas de clusterização, onde se deseja descobrir agrupamentos inerentes aos dados, ou em problemas de associação, buscando o descobrimento de regras que descrevam grande parte de um conjunto de dados. Analogamente ao aprendizado anterior, o termo "não-supervisionado" se refere ao fato de que não há um professor e nem respostas corretas.

Neste projeto, não pretende-se testar várias técnicas de aprendizado de máquina e nem definir a melhor técnica para o conjunto de dados a ser utilizado. O objetivo é testar uma ou duas técnicas de aprendizado de máquina para que seja possível validar o sistema como um todo. Desta forma, optou-se por estudar e avaliar um algoritmo supervisionado e um não-supervisionado: regressão linear para problemas de regressão (aprendizado supervisionado) e *k-means* (aprendizado não-supervisionado) para problemas de clusterização.

### 2.4.1 Regressão linear

Os modelos de regressão para manutenção preditiva são usados para computar a vida útil restante de um equipamento. Este parâmetro é definido como a quantidade de tempo que um equipamento está funcionando antes que ocorra a próxima falha. Entretanto, o cálculo não é possível sem saber quanto tempo o equipamento sobreviveu antes de uma falha. Portanto, em contraste com outras técnicas de aprendizado de máquina, equipamentos sem um histórico de falhas não podem ser utilizados para essa modelagem (MICROSOFT AZURE, 2018).

Um modelo de regressão linear simples estabelece a relação entre uma variável dependente (saída) e uma única variável independente (entrada) usando uma reta definida pela Equação 2.1 (LANTZ, 2015)t:

$$y = \alpha + \beta x \tag{2.1}$$

### 2.4.2 *K-means*

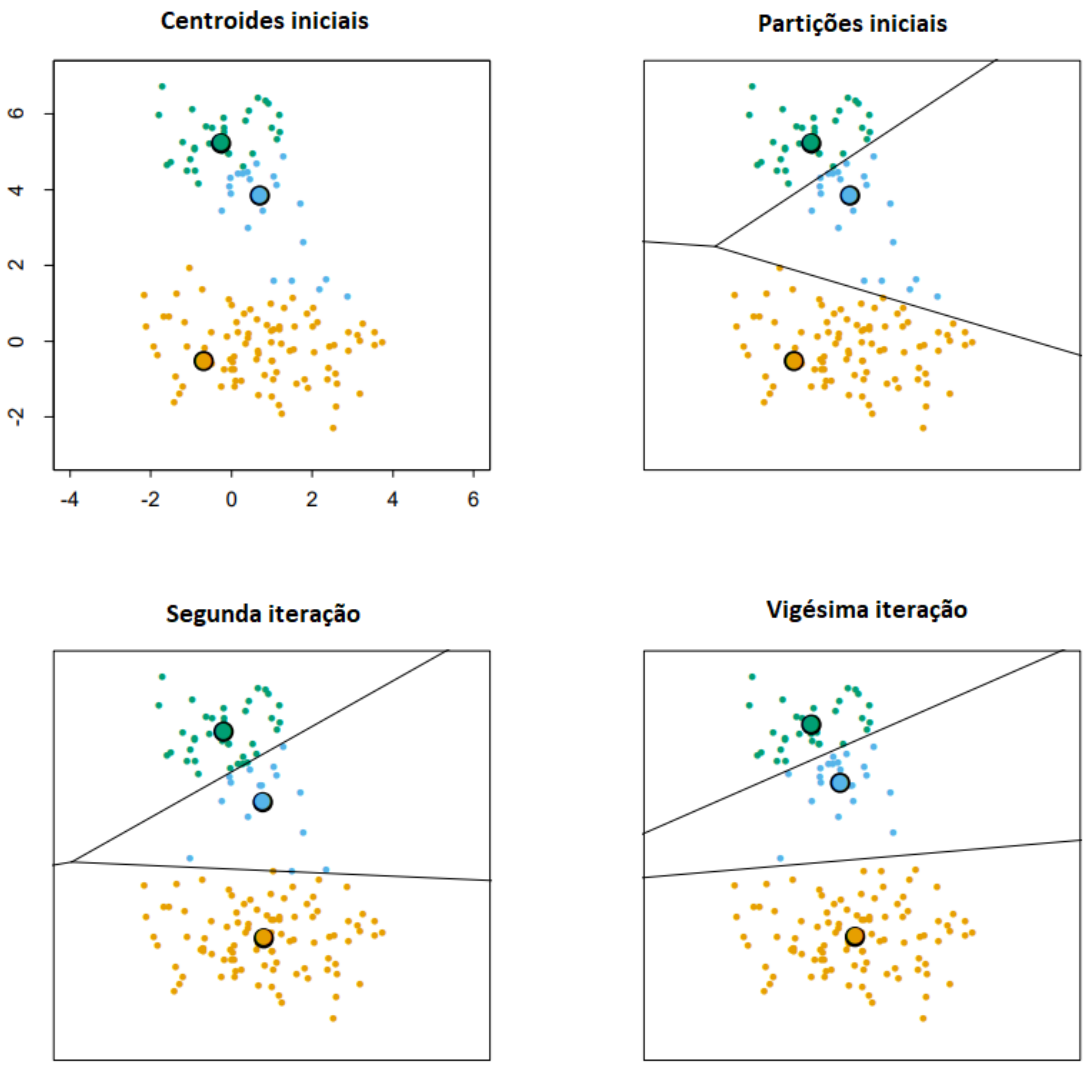
O algoritmo *k-means* é o mais comum dentre os métodos de clusterização e bastante utilizado na detecção de falhas e predição das classes de falhas em técnicas de aprendizado não-supervisionado. Esse algoritmo usa a distância euclidiana entre as amostras para formar os *clusters* (AMRUTHNATH; GUPTA, 2018).

Segundo Lantz (2015), o algoritmo *k-means* começa escolhendo, de forma aleatória, $k$ pontos no espaço de recursos para serem os centros dos *clusters* que, por sua vez, são os catalisadores que estimulam as amostras restantes a se agruparem. Existem diferentes métodos para determinar quais serão os centros iniciais, que propõem a diminuição do efeito randômico sobre o resultado final. Após escolher os centros iniciais dos *clusters*, as outras amostras são aproximadas destes centros de acordo com a função de distância, que tradicionalmente é a euclidiana, mas também podem ser utilizadas as distâncias de *Manhattan* e *Minkowski*. Se considerarmos que $n$ é o número de amostras, é possível deteminar a distância euclidiana entre uma amostra $x$ e uma amostra $y$ através da Equação 2.2:

$$dist(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2.2}$$

Ainda de acordo com Lantz (2015), após a fase de atribuição inicial ser concluída, o algoritmo *k-means* prossegue para a fase de atualização. A primeira etapa da atualização dos *clusters* envolve a mudança dos centros iniciais para um novo local, conhecido como centroide, que é calculado a partir da posição média dos pontos atualmente atribuídos a esse *cluster*. Esse processo é repetido ciclicamente, afim de aumentar a precisão com relação às amostras e ao *cluster* em que estão inseridas, conforme ilustra a Figura 4.

Figura 4 – Exemplo de funcionamento do algoritmo *k-means*.




Fonte: Adaptada de Hastie, Tibshirani e Friedman (2009).

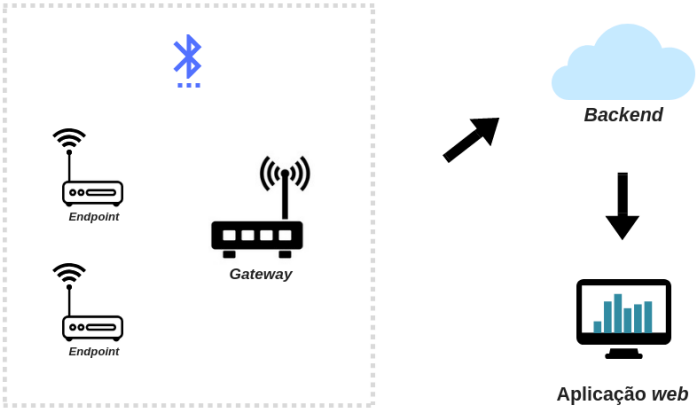
# 3 PROPOSTA

A proposta deste trabalho é desenvolver um sistema capaz de implementar os conceitos de CBM para a predição de falhas em motores elétricos utilizados em equipamentos industriais. Para tal, são acoplados *endpoints* na parte externa destes equipamentos afim de coletar as informações referentes à vibração do motor e enviá-las, através de uma rede BLE, para um *gateway* que tem como principal função encaminhar esses dados para um servidor remoto, hospedado na *Amazon web services* (AWS). O servidor, por sua vez, armazena os dados obtidos em um banco de dados *MySQL* e executa um *script* desenvolvido em *Python* responsável pela aplicação dos algoritmos de aprendizado de máquina citados nas subseções 2.4.1 e 2.4.2, os quais são treinados para que sejam capazes de detectar um comportamento anômalo, baseando-se na vibração do motor. Por fim, os dados armazenados são consumidos por uma aplicação *web* que exibe os valores em forma de gráficos e alerta o usuário nos casos em que o algoritmo de aprendizado de máquina observa algo incomum.

Além das atribuições mencionadas ao *backend*, nele também são definidas as *APIs RESTful* para viabilizar sua comunicação tanto com o *gateway* quanto com a aplicação.

A Figura 5 ilustra a interação entre os módulos do sistema proposto.

Figura 5 – Diagrama do sistema proposto

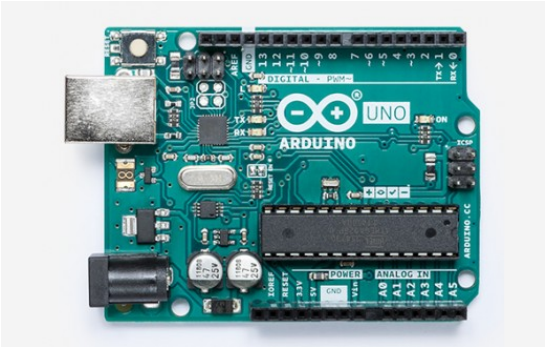



Fonte: Criada pelo autor.

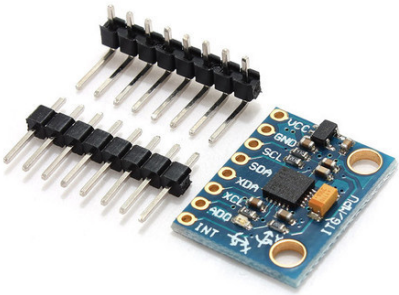
Buscando alcançar um sistema de baixo custo e que utilizasse os recursos disponíveis na instituição, optou-se por utilizar *softwares* livres e *hardwares* acessíveis na composição dos módulos do subsistema de coletas (*endpoint* e *gateway*), que são detalhados a seguir:

- *Endpoint*: é constituido por um Arduino Uno (Figura 6a), um sensor de vibração MPU-6050 (Figura 6b) e um módulo BLE HM-10 (Figura 6c). Estes dois últimos são conectados ao Arduino que obtém os dados do sensor de vibração e utiliza o módulo BLE para enviar as informações ao *gateway*, cuja conexão é inicializada pelos pacotes de anúncio, frequentemente enviados à rede pelo módulo BLE. Após a conexão BLE ser estabelecida, o *endpoint* assume o papel de servidor e envia os dados, inclusive os obtidos a partir do sensor de vibração, para *gateway*.

Figura 6 – Componentes do *endpoint*




(a) Arduino Uno




(b) Sensor de vibração

(c) Módulo BLE

- *Gateway*: por ser compatível nativamente tanto com a tecnologia Wi-Fi quanto com o BLE, o módulo de desenvolvimento ESP32 (Figura 7) foi o escolhido para realizar o papel de *gateway* do sistema. É ele quem se conecta aos *endpoints* via BLE para obter os dados coletados pelos sensores e os envia para o *backend*, utilizando a rede Wi-Fi para conectar-se à Internet. Com relação à conexão BLE, o *gateway* faz o *scan* da rede afim de procurar pacotes de anúncio, que são enviados pelos módulos BLE. Após receber um pacote deste, o *gateway* se conecta ao *endpoint* e assume a função de cliente, que coleta as informações enviadas pelo servidor e então as encaminha para o *backend*, através de uma *API RESTful* fornecido por este último.

É importante ressaltar que, por serem utilizados componentes de baixo custo na construção do subsistema de coletas, possivelmente os dados não serão realísticos o suficiente para que se obtenha resultados satisfatórios a partir da aplicação de algoritmos de aprendizado de máquina. Neste caso, o *endpoint* e o *gateway* servirão para validar apenas a coleta da vibração do motor e enviar estes dados ao *backend*.

Figura 7 – *Gateway* ESP32

Contudo, para que seja possível validar o subsistema de gerenciamento e análise dos dados, é proposto que sejam utilizados os produtos da empresa Bluvision (2018), mencionados na seção introdutória deste documento. Por possuírem *beacons* e *gateways* de mais alta qualidade, os dados coletados são de mais alta fidelidade, sendo possível aplicar e verificar a assertividade dos algoritmos de aprendizado de máquina.

Foram definidas sete tarefas para atingir o objetivo proposto neste trabalho, conforme descrito no cronograma apresentado na Tabela 1:

Tabela 1 – Cronograma das atividades previstas

| Tarefa | Mês | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Dez | Jan | Fev | Mar | Abr | Mai | Jun | Jul |
| T1 | √ | | | | | | | |
| T2 | | √ | √ | | | | | |
| T3 | | | | √ | √ | | | |
| T4 | | | | | √ | | | |
| T5 | | | | | | √ | √ | |
| T6 | | | | | | | √ | |
| T7 | | | | | | | √ | √ |

- T1: Correção e ajustes do documento de acordo com o avaliado pela banca;
- T2: Configuração e conexão do *endpoint* e *gateway*;
- T3: Configuração do servidor, banco de dados e definição da *API RESTful* na AWS;
- T4: Encaminhamento das informações do *gateway* para o *backend* e armazenamento no banco de dados;
- T5: Desenvolvimento do *script* em *Python* para a aplicação do algoritmo de aprendizado de máquina;
- T6: Implementação da aplicação *web*;
- T7: Elaboração de um documento final.

# REFERÊNCIAS


AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *Communications Surveys & Tutorials, IEEE*, IEEE, USA, v. 17, n. 4, p. 2347–2376, 2015. ISSN 1553-877X. Citado na página 15.

ALFIAN, G. et al. A personalized healthcare monitoring system for diabetic patients by utilizing ble-based sensors and real-time data processing. *Sensors*, v. 18, n. 7, 2018. ISSN 1424-8220. Disponível em: <http://www.mdpi.com/1424-8220/18/7/2183>. Citado na página 20.

AMRUTHNATH, N.; GUPTA, T. Fault class prediction in unsupervised learning using model-based clustering approach. In: *2018 International Conference on Information and Computer Technologies (ICICT)*. [S.l.: s.n.], 2018. p. 5–12. Citado 2 vezes nas páginas 20 e 21.

BLUVISION. *Site*. Florida, 33334, 2018. Disponível em: <http://bluvision.com/>. Acesso em: 18 nov. 2018. Citado 2 vezes nas páginas 11 e 25.

CHANG, K.-H. Bluetooth: A viable solution for iot? *Wireless Communications*, IEEE, USA, p. 6–7, 2014. Citado na página 17.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The elements of statistical learning: data mining, inference and prediction*. 2nd. ed. Springer, 2009. Disponível em: <http://www-stat.stanford.edu/~tibs/ElemStatLearn/>. Citado 2 vezes nas páginas 20 e 22.

LANTZ, B. *Machine Learning with R*. 2nd. ed. [S.l.]: Packt Publishing, 2015. ISBN 1784393908, 9781784393908. Citado 3 vezes nas páginas 20, 21 e 22.

MICROSOFT AZURE. *Guia de IA do Azure para soluções de manutenção preditiva*. Redmond, WA 98052-7329, 2018. Disponível em: <https://docs.microsoft.com/pt-br/azure/machine-learning/team-data-science-process/cortana-analytics-playbook-predictive-maintenance>. Acesso em: 16 nov. 2018. Citado na página 21.

PRAJAPATI, A.; BECHTEL, J.; GANESAN, S. Condition based maintenance: a survey. *Journal of Quality in Maintenance Engineering*, v. 18, n. 4, p. 384–400, 2012. Citado 2 vezes nas páginas 13 e 14.

SAMIE, F.; BAUER, L.; HENKEL, J. Iot technologies for embedded computing: A survey. In: ACM. *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis*. [S.l.], 2016. p. 8. Citado na página 15.

SHIN, J.-H.; JUN, H.-B. On condition based maintenance policy. *Journal of Computational Design and Engineering*, v. 2, n. 2, p. 119 – 127, 2015. ISSN 2288-4300. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2288430014000141>. Citado 3 vezes nas páginas 11, 13 e 14.

TOSI, J. et al. Performance evaluation of bluetooth low energy: A systematic review. *Sensors*, MDPI, USA, p. 1–34, 2017. Citado 2 vezes nas páginas 17 e 18.

XIAO, Y. et al. Full-duplex machine-to-machine communication for wireless-powered internet-of-things. In: . [S.l.]: IEEE, 2016. p. 1–6. ISBN 9781479966646. ISSN 1938-1883. Citado na página 15.

ČABARKAPA, D.; GRUJIć, I.; PAVLOVIć, P. Comparative analysis of the bluetooth low-energy indoor positioning systems. *Telsiks*, IEEE, USA, p. 76–79, 2015. Citado na página 17.