<https://www.embarcados.com.br/coletor-analisador-portatil-de-vibracao/>

Sistema De Baixo Custo De Monitoramento De Falhas Críticas Utilizando Iot

CONCEITOS DE CIENCIAS DA COMPUTAÇÃO PARA DESENVOLVIMENTO DE SOFTWARE

ESCALABILIDADE

Core do tabalho: DESCREVER A FREQUENCIA DAS ANÁISES DE VIBRAÇÕES SÃO REALIZADAS , CAPADCIDADES do HARDWARE E POR ONDE CAMINHAM AS BOAS ANÁLISES

Ever in this page refs [2]

Node.js is an open-source language, developers use JavaScript for client-side functionality often. Node.js is the server component in the same language.

Representational State Transfer (REST) is an architecture style for building resources on the web. Examples of resources for website include HTML documents, images, and script files.

To retrieve or update a resource, perform an action through HTTP methods. To identify which resource to retrieve or update, REST uses a Uniform Resource Identifier (URI) to describe the network location of the resource.

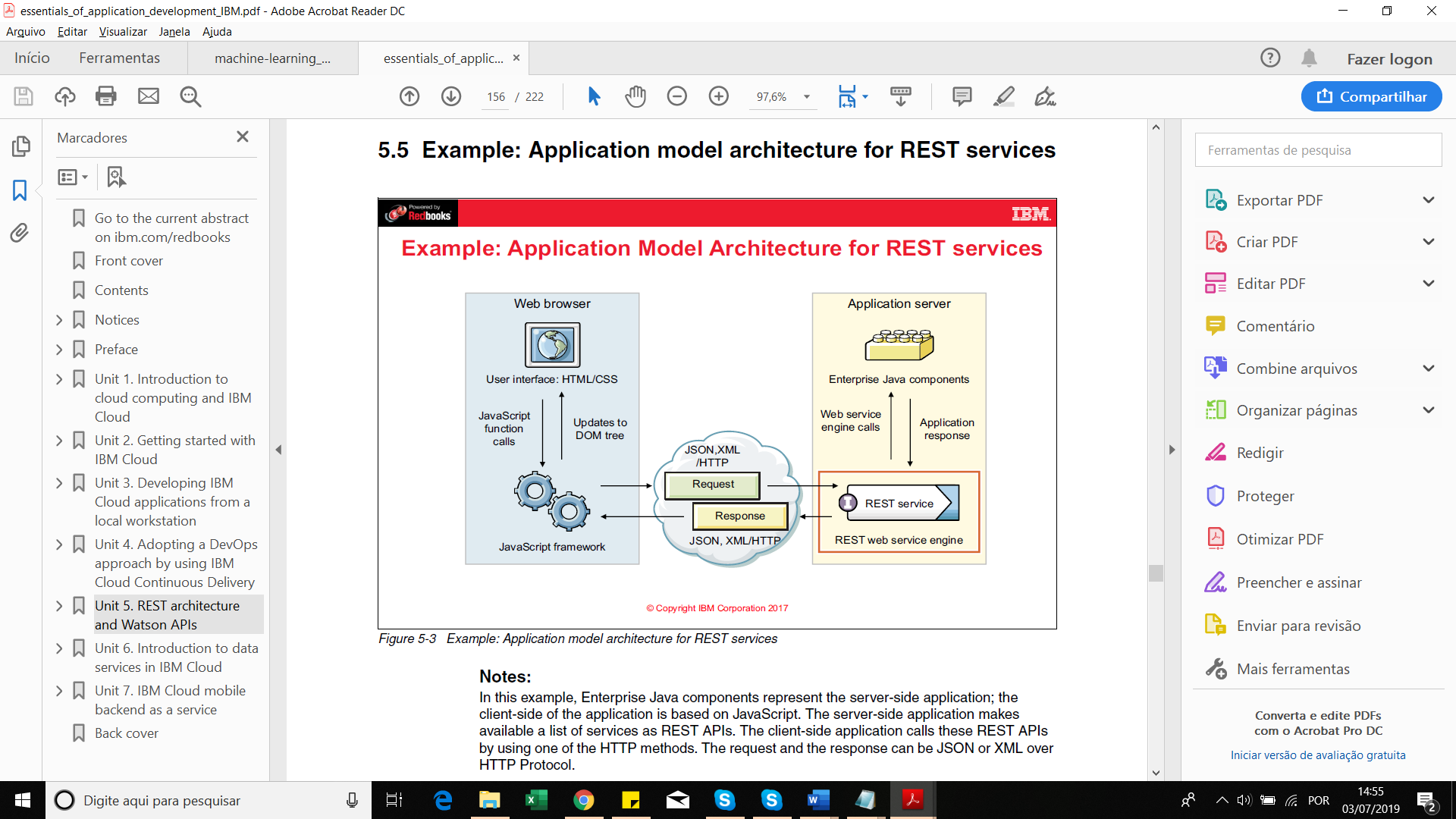
REST provides the following HTTP methods:

|  |  |  |  |
| --- | --- | --- | --- |
| GET | POST | DELETE | PUT |
| OPTIONS | HEAD | TRACE | CONNECT |

The GET method is used to retrieve information from the server. When you use your browser to navigate to any URI, you use the GET method to get the HTML of that website. The query string that contains the parameters that are needed for the request are sent in the URL by placing a question mark (?) at the end of the URI, and then, writing the parameters.

In a more general sense, web resources represent a source of information. For example, HTML documents define the structure of a web page. Cascading Style Sheet (CSS) documents define the presentation of a web page, and image files provide a visual representation of information. With REST services, you treat server applications as web resources.

A REST service is now an entry point to an application on the server. It provides information from the server application. To call a REST service, use HTTP method verbs, such as GET, PUT, and POST. To specify which REST service to call, use a URI to describe the location of the resource on the server.



* What is a RESTful web service?
* JavaScript Object Notation (JSON)?

1 INTRODUÇÃO ...................................................................................................

1.1 TEMA DA PESQUISA......................................................................................

1.1.1 Delimitação do tema ...................................................................................

1.2 PROBLEMA......................................................................................................

1.3 PREMISSA......................................................................................................

1.4 OBJETIVOS......................................................................................................

1.4.1 Objetivo Geral...............................................................................................

1.4.2 Objetivos Específicos....................................................................................

1.5 JUSTIFICATIVA................................................................................................

1.6 MÉTODO DA PESQUISA.................................................................................

2 DESENVOLVIMENTO..........................................................................................

2.1 ESTADO DA ARTE............................................................................

2..2 Tecnologias Da Transformação Digital......................................................

Iot

Tipos de Manutenção ...................................................................................

Concept and structure of the device........................................

Communication software.......................................

Vibration signal processing ..........................

Quantificação e Severidade de vibração ......................................................

Results of the device testing

Save eprom data and send

ABSTRACT

Nesta dissertação, vários métodos e diferentes abordagens foram estabelecidos e avaliados com sucesso, a fim de desenvolver um sistema on-line capaz de monitorar condição de motores de indução, através de análise de vibração.

Em geral, a manutenção preditiva de motores de indução é adequada de pequenas à grandes indústrias, a fim de reduzir o tempo de inatividade, aumentar a eficiência e confiabilidade. Nesse contexto, a vibração é analisada para reunir informações cruciais para mapear paradas de manutenção da máquina. O sinal de vibração é amplamente utilizado para determinações de condições do MIT. Para tanto, tratou-se de aplicar uma estrutura de hardware com o micro controlador ESP8266 e sensor ADXL345 junto a ferramentas como IoT e Cloud servives para criar um sistema para dar a condição em tempo real do monitoramento para motores de indução.

Palavras chave: XYZ

ESTRUTURA DO TRABALHO

De forma a estruturar o desenvolvimento, o trabalho será dividido em 5 partes: introdução, fundamentação teórica, desenvolvimento, discussão de resultados e conclusão.

Na introdução serão feitas as considerações iniciais através de *overview*, mostrando o tema proposto, bem como o problema que levou ao seu desenvolvimento. Também serão discutidos justificativa, objetivos e procedimentos metodológicos.

A fundamentação teórica apresentará o estado da arte necessário para o desenvolvimento do sistema. Fazendo ênfase a temática atual do processo refinado de tratamento de dados de **Sistemas embarcados** através de *cloud services*. Síntese de diferentes estudos sobre o desenvolvimento de aplicações com placa eletrônica ESP8266/32, sensor ADXL345, linguagens de programação, documentações e o projeto do protótipo a ser realizado.

Na terceira parte do trabalho consta o projeto e desenvolvimento do sistema supervisório de análise de vibração para “motores de indução”, apresentará como os dados coletados pelos sensores serão tratados e visualizados pelo usuário. Detalhamento do desenvolvimento do software e arquitetura de hardware utilizada. Discussão de domínio das análises(..)

Utilizando o protótipo desenvolvido, serão realizados testes e simulações do sistema, bem como a discussão dos resultados obtidos e dificuldades encontradas durante o seu desenvolvimento e comparações de resultados, para compor a quarta parte.

Na conclusão deverão constar comentários pertinentes ao fechamento do ciclo do trabalho. Será avaliado se o objetivo foi atingido, sugestões e possíveis melhorias para trabalhos futuros.

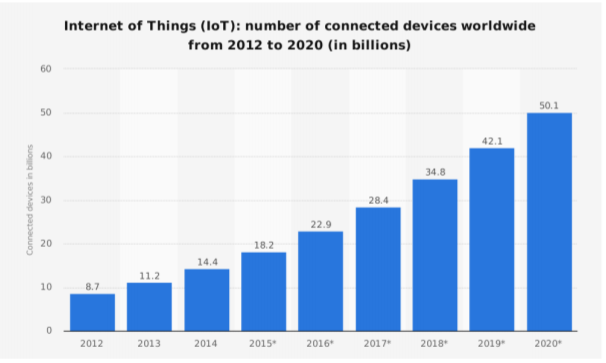
INTRODUÇÃO

Hoje em dia, as linhas de produção estão em processo constante de automatação, e a intervenção humana em um processo é menos frequente. Isto exige algoritmos de detecção automática de falhas a serem implementados, bem como dispositivos capazes de manter o monitoramento dos estados de um processo por um longo tempo em um modo autônomo, especialmente quando os locais onde os dispositivos são montados não são tão facilmente acessados. Isto requer, por sua vez, uma gestão eficiente do condicionamento do dispositivo, bem como capacidades de auto-diagnóstico.

Aplicações para monitorar sistemas em tempo real são de suma importância para determinações de confiabilidade. Adventos como IoT revolucionou aplicações de fácil construção e comunicação com a web, permitindo a comunicação de pacotes de dados de tamanho reduzidos com servidores de acordo com a programação incluida no hardware.

É observado ao aumento exponencial do volume de dados disponíveis e o aumento da capacidade computacional dos sistemas, a demanda pelo desenvolvimento de novas aplicações tem sido impulsionada ao redor do mundo, buscando melhorar a experiencia do usuário final, com alto nível de assertividade utilizando ferramentas e API’s diversas (*Application Programming Interface*) para conectar diferentes serviços.

O número de dispositivos conectados à web em todo o mundo está crescendo continuamente e de acordo com [5] irá atingir 50 bilhões de dispositivos até 2020.



Além disso, toda máquina em operação possui tendências a possíveis falhas. Tomando motores industriais como exemplo, fatores como como quantidade de lubrificação, considerações elétricas, ventilação do motor, alinhamentos e carga do motor são algumas possibilidades que podem ser motivo para o motor apresentar falhas. Esses fatores resultam em vibrações do motor ou aumento na temperatura do motor para níveis críticos [6].

OVERVIEW

JUSTIFICATIVA

O estudo DO DESENVOLVIMENTO DE sistema EMBARCADIS tem beneficiado o controle de manutenção de ativos como MIT, porém há de se notar que bons hardware. Sistemas disponíveis no mercado atualmente possuem um elevado custo de implementação, não sendo viáveis para uma difusão em larga escala. O estudo em questão propõe uma alternativa de baixo custo .....

OBJETIVOS GERAIS

O objetivo deste trabalho é apresentar o desenvolvimento de um sistema supervisório para monitoramento de vibração de máquinas rotativas. Além disso, trazer a importância de trabalhar com ferramentas da atualidade da ind. 4.0. Para isso, utiliza-se um microcontrolador ESP8266, o qual por meio de um software escrito através das livrarias disponíveis, (desenvolvido em linguagem C), possibilita comunicação com sensores e acesso à internet. O software do sistema embarcado registra os dados em um banco de dados noSQL, apresentando os resultados obtidos com fácil interação com o usuário.

Possibilitando implementações diversas, como métricas para determinação dos cenários aceitáveis de resultados para detecção de condições do MIT, além de aplicar ferramentas diversas de análise de banco de dados , ex IBM WATSON ou outras ferramentas em cloud services.

ESPECÍFICOS

* Ter sucesso na realização do desenvolvimento de um sistema supervisório de monitoração de vibração para máquinas rotativas.
* Implementar um software no Esp8266/adxl345 capaz de captar dados e se comunicar através de protocolos, para enviar os dados para uma data-base.
* Utilizar a infraestrutura disponível gratuitamente de análises estatísticas da Cloud.

METODOLOGIA

Este sistema foi projetado e desenvolvido do usando componentes comercialmente disponíveis no mercado e plataforma de software de código aberto para aquisições de dados, baixo consumo de energia e coleta de dados pelo sistema IoT. em tempo real. Os resultados experimentais revelam que o sistema é capaz de capturar e reportar parâmetros vitais do motor para o servidor em nuvem e uma notificação automática é enviada aos operadores quando a anormalidade do motor é detectada.

Graças à tecnologia IoT, a manutenção preventiva do motor pode ser planejada de maneira eficaz e remota, com coleta e análise de dados valiosas. Com técnicas avançadas de redução de consumo de energia, um nó de sensor consome uma quantidade extremamente baixa de energia de bateria, ideal para aplicações móveis.

I. INTRODUÇÃO

O MIT deve ser eficiente e confiável, pois é necessário fornecer velocidade e torque em ampla faixa de operação, mantendo o controle preciso do acionamento do motor com segurança [1]. Para evitar anormalidades do motor de tração, é desejável a confiabilidade melhorada e a operação efetiva com um aviso antecipado com notificação instantânea. Vibração, corrente e temperatura são três parâmetros que são bem estudados e amplamente aceitos na detecção de falhas do motor devido ao fato de que falhas elétricas e mecânicas estão sendo monitoradas [2].

De acordo com a pesquisa feita pelo Instituto de Engenheiro Elétrico e Eletrônico (IEEE) [3], 44% das falhas do motor são de rolamentos e 24% são de estator. A maioria das falhas mecânicas no motor são desequilíbrio mecânico, rolamento e mancais, porque um estresse contínuo sobre eles pode resultar na falha maior. Fatores como lubrificação inadequada, instalação inadequada, contaminação e corrosão frequentemente contribuíram para falhas de laminação e rolamentos. Um sensor de vibração e um sensor de corrente são capazes de detectar o funcionamento irregular do rolamento do motor aumentando a vibração e desequilibrando a corrente do eixo devido ao distúrbio do fluxo causado pelas excentricidades do rotor.

A falha do rolamento também faz com que a temperatura exceda a temperatura de carga predeterminada do motor [4] [5]. Nosso nó de sensor proposto para Internet das Coisas (IoT) é capaz de integrar esses sensores ativos e transmitir dados para a Internet em tempo real com o mínimo de energia necessária. Comparado com o sistema com fio, o sistema IoT oferece muitas vantagens, como custo relativamente baixo, facilidade de instalação, software atualizável remotamente e automatizar a análise de dados em tempo real e as notificações de alerta para os operadores. Além disso, a manutenção preventiva do motor de tração pode ser planejada de maneira eficaz e remota, com coleta e análise de dados valiosas. Este trabalho apresenta um desenvolvimento e implementação de um sistema de monitoramento de condições de acionamento de motor de tração sem fio IoT baseado em ESP8266

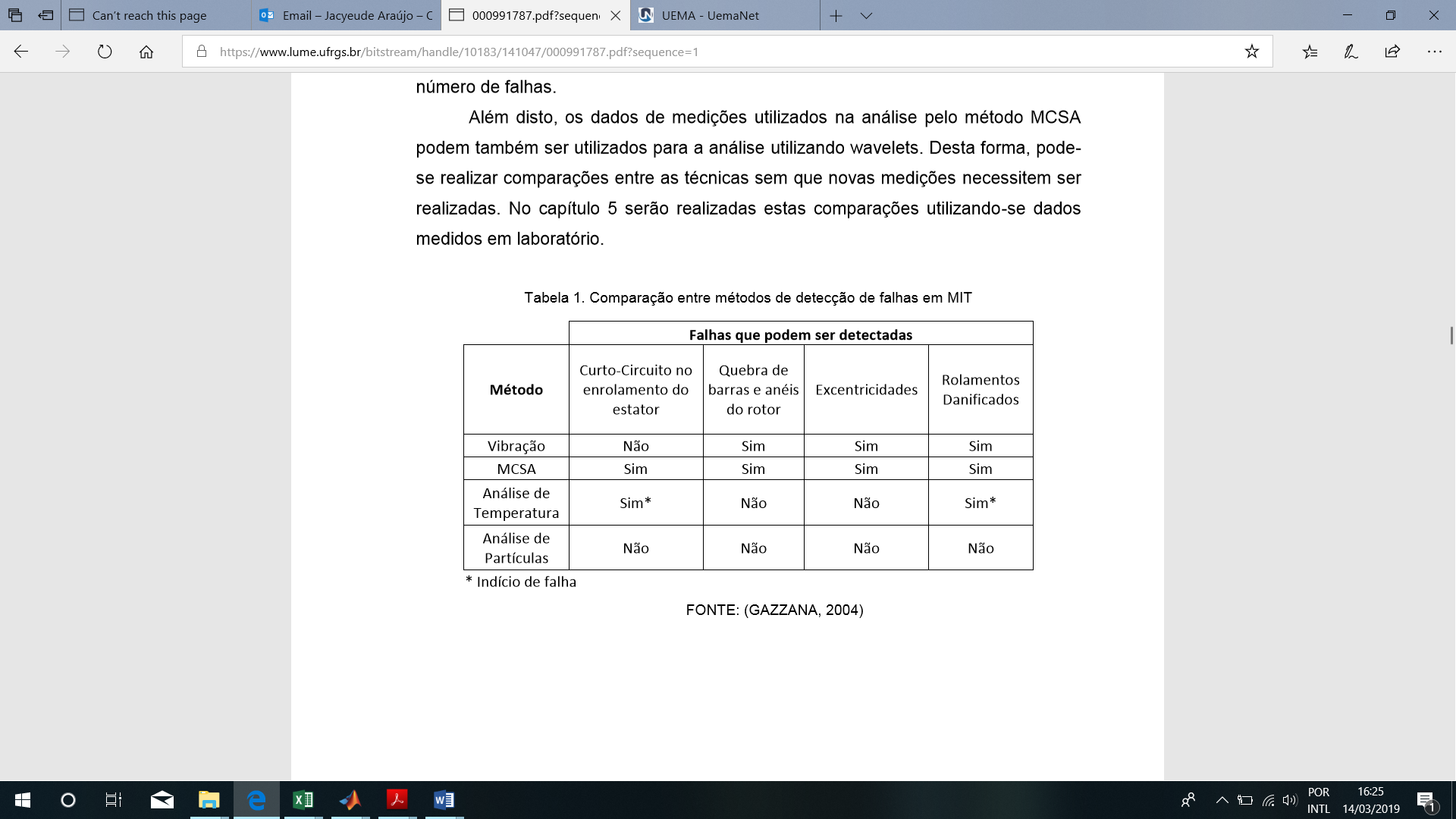
FUNDAMENTAÇÃO TEÓRICA

Tal concentração de fontes de processamento permite capturar, processar e analisar muitas vezes em tempo real, dados provenientes de bilhões de dispositivos conectados que servem a diferentes aplicações, desde monitoramento de ambientes, aplicações industriais e comerciais além de aplicações voltadas para melhorar a vida dos usuários.

How is Vibration Measured?

In this section we will answer these questions.After reading this section you will be able to:

* Recognize which machines should be monitored
* Understand how vibration sensors should be mounted
* Determine how measurement parameters should be set
* Take measurements in a systematic way



1 - Tecnologias da transformação digital, a utilização das novas ferramentas que estão possibilitando desenvolvimento aprimorado e acelerado de aplicações em N áreas.

\_\_

Nas últimas duas décadas, vem crescendo exponencialmente a disponibilidade de dados digitais, além de maior eficiência e velocidade de computação. Isso tudo caminha por um interesse renovado nas tecnologias da transformação digital que vem inovando e agregando valor à diferentes tipos de portifólios de negócios. [1]

Diversos desses fatores, com origem principalmente na tecnologia, que vem impulsionam o uso destas técnicas de *Data Science* em um conjunto amplo de setores. Estes fatores podem ser agrupados em quatro eixos: [1]

(1) o aumento sem precedentes do volume e da tipologia de dados disponíveis;

(2) a conectividade e o acesso aos dados;

(3) a melhoria dos algoritmos utilizados;

(4) o aumento da capacidade computacional dos sistemas;

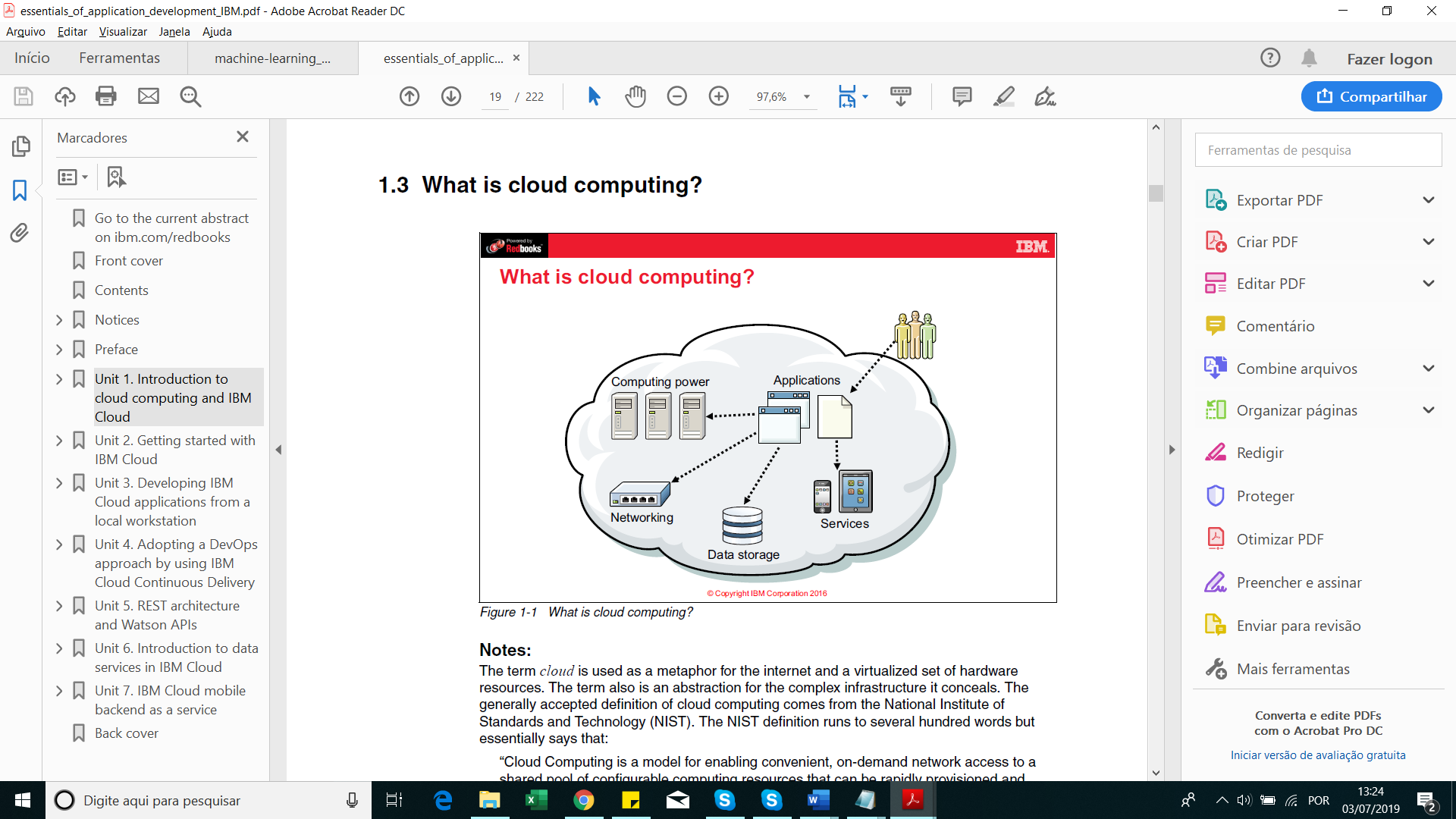


É necessário pontuar sobre os seguintes tópicos

**Acesso móvel e internet:** devido à generalização da telecomunicação e ao acesso a uma ampla diversidade de serviços on-line, interfaces de programação de aplicativos (APIs) estão sendo desenvolvidas pelas empresas, o que lhes permite ter novos canais de interlocução com os clientes e contar com um amplo conjunto de dados. Isto leva à aparição de novas formas relevantes de informação, como a geolocalização exemplo.

**Internet of Things:** O conceito de iot refere-se à interconexão através de redes abertas de dispositivos com capacidade computacional que enviam e recebem dados sem intervenção humana, o que permite a compilação massiva e direta de dados, bem como a operação remota em tempo real dos dispositivos conectados à internet. Visando principalmente aprimorar a experiência do usuário final.

**Cloud computing:** O uso de aplicativos na nuvem permite o acesso à informação em qualquer lugar, facilita operações comerciais e permite maior rapidez, segurança e custos mais baixos. Isto leva a um novo modelo de negócio em que esses recursos são oferecidos como utilities e cobrados como um serviço.



The term *cloud* is used as a metaphor for the internet and a virtualized set of hardware resources. The term also is an abstraction for the complex infrastructure it conceals. The generally accepted definition of cloud computing comes from the National Institute of Standards and Technology (NIST). The NIST definition runs to several hundred words but essentially says that: [2]

“Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

Examples of computing resources include:

\_ Networks

\_ Servers

\_ Storage

\_ Applications

\_ Services

**Factors contributing to growth of cloud**

One factor contributing to the growth of cloud computing is that today’s applications must be delivered quickly. Developers are pressured to get their product to market as soon as possible. They want to get feedback quickly, and then iterate on the idea to make the product better and faster. [2]

Cloud makes hardware resources readily available and quick to configure, which shortens the time required for developers to show a working version of their products. Also, cloud allows the reuse of the same resources for multiple successive projects, which is more cost-efficient. [2]

2 - Real-Time Embedded Systems

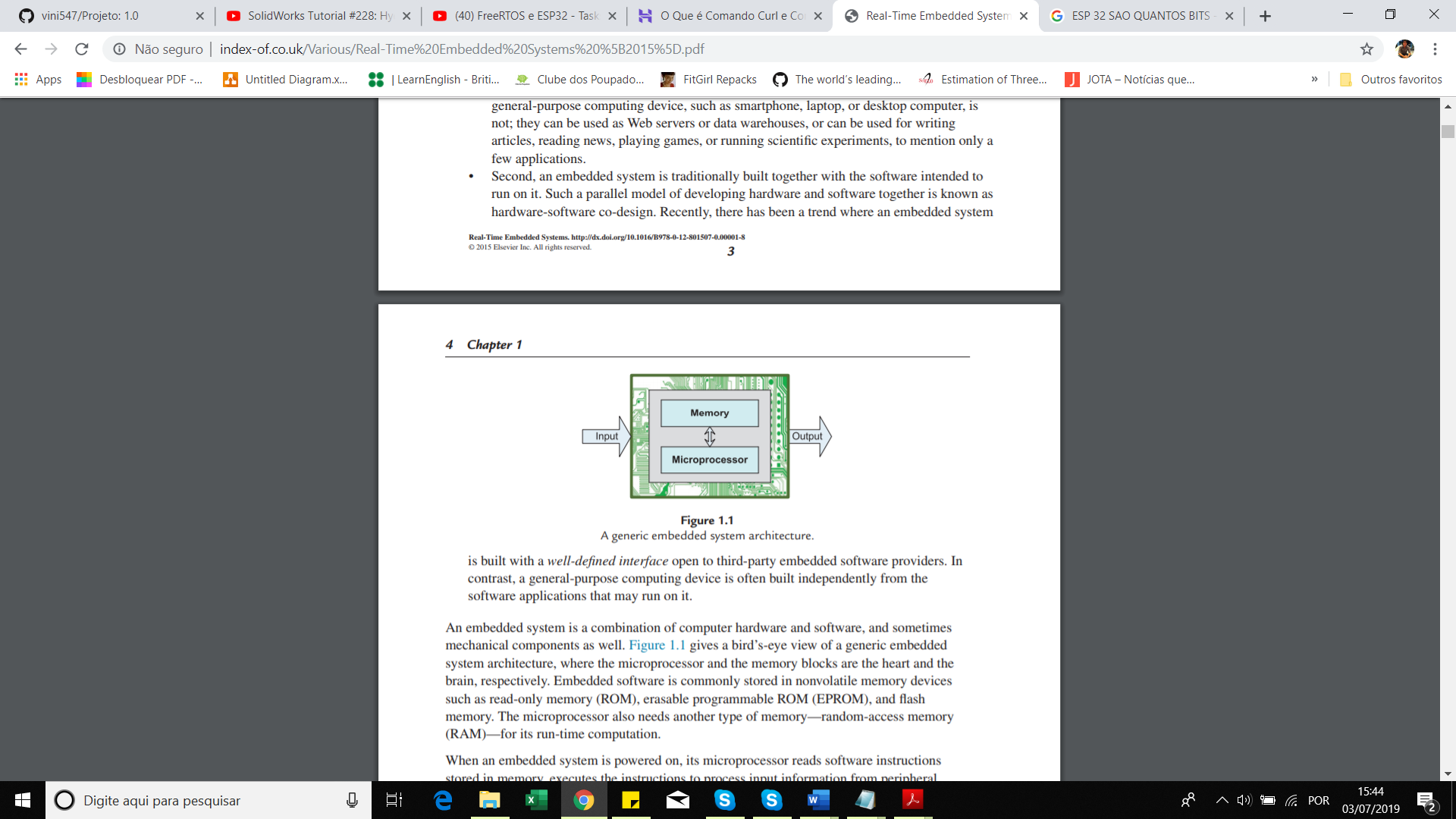
\_\_

Segundo [3] existem três classes de sistemas digitais: emulação e sistemas de prototipação, sistemas de computação de propósito geral e sistemas embarcados (embedded systems). Sistemas de emulação e prototipação são baseados em tecnologias de hardware reprogramáveis, onde o hardware pode ser reconfigurado pela utilização de ferramentas de síntese. Tais sistemas requerem usuários especialistas e são utilizados para a validação de sistemas digitais. Sistemas de computação de propósito geral incluem computadores tradicionais abrangendo desde laptops até supercomputadores, estes sistemas são caracterizados pelo fato de que usuários finais podem programar o sistema. Diferentes aplicações são suportadas dependendo do tipo do software utilizado pelo usuário.

[4]Embedded systems differ from general-purpose computing devices mainly in two aspects:

• First, an embedded system is designed simply for a specific function, whereas a general-purpose computing device, such as smartphone, laptop, or desktop computer, is not; they can be used as Web servers or data warehouses, or can be used for writing articles, reading news, playing games, or running scientific experiments, to mention only a few applications.

• Second, an embedded system is traditionally built together with the software intended to run on it. Such a parallel model of developing hardware and software together is known as hardware-software co-design. Recently, there has been a trend where an embedded system is built with a well-defined interface open to third-party embedded software providers. In contrast, a general-purpose computing device is often built independently from the software applications that may run on it.[4]



When an embedded system is powered on, its microprocessor reads software instructions stored in memory, executes the instructions to process input information from peripheral components (through sensors, signals, buttons, etc.), and produces output to meet the needs of the external embedding system.[4]

3 - Monitor de Vibrações em tempo real, usando o ESP32+Acelerômetro, como fonte de dados que alimenta um servidor.

Machine condition monitoring is the process of monitoring the condition of a machine with the intent to predict mechanical wear and failure. Vibration, noise, and temperature measurements are often used as key indicators of the state of the machine. Trends in the data provide health information about the machine and help detect machine faults early, which prevent unexpected failure and costly repair.

4 - Aplicação da firmware desenvolvida pela espressif , permitindo aplicar linguagem nativa (C, C++) usando esp-idf.

5 - Protocolos de comunicação usados para transitar informações, dentre eles Curl.

\_\_

**HTTP Request**

Rest call - Habilidade de invocar um serviço através de protocolos TPC/IP ou HTTP. São basicamente a tecnologia de serviço ao cliente que está gerenciando web servers. Isso faz com que o ESP32 seja capaz de fazer uma "Rest Call" à um provedor de serviço que ira receber essa call, trabalhar internamente e retornar uma reposta. É, em resumo, uma request envelopada em um protocolo HTTP com uma URL e uma Payload que quando enviada a um servidor, que provavelmente está portando um código que fará alguma lógica, recebe uma resposta de volta.

**CURL**

O curl é usado em linhas de comando para transferir dados.

Ele é usado como abreviação para “Client URL”. Comandos Curl são destinados para funcionar como uma forma de verificar a conectividade da URL, além de ser uma ótima ferramenta de transferência de dados. O Comando Curl suporta diversos tipos de protocolos, dentre eles o HHTP.

6 - Utilização IDF e FreeRTOS, sistema op. Task, priorização, stack reservado para executar tarefas através do core esprcificado.

Normalmente chamada apenas de IDF, podemos dizer que esta engloba todos os conjuntos de API’s e configurações do ESP32. Já a Toolchain **(mingw32)** consiste em uma ferramenta que compila e constrói o código que escrevemos juntamente com as configurações escolhidas no **“menuconfig”**. Por fim, ESP IDF é o conjunto de bibliotecas feitas pro ESP32.

Aqui é o “paraíso” do ESP32, onde conseguimos configurar praticamente tudo desse microcontrolador, desde clock até alocação de memórias dinâmicas e Watchdog.

Os principais pontos a serem levados em consideração para você trocar a Arduino IDE ou outra, pela IDF são:

* Suporte a todas features, incluindo **Bluetooth**, **Flash Encryption e Secure Boot** (itens importantes para quem pretende criar e vender produtos).
* Configuração total do sistema, como por exemplo **eFuses**, **clock**, **watchdog’s**, **timer’s**, **memória dinâmica para WiFi** e até **tempo de Wake-UP após Deep sleep.**

7- Especificar os end points que se conectarão a estruturação da data base

8 -

9 -

10-

REFERÊNCIAS

[1] revista **Machine Learning, uma peça-chave na transformação dos modelos de negócio**. Editora PeD 2018

[2] Ahmed Azraq,Hala A. Aziz, Uzma Siddiqui - **Essentials of Application Development on IBM Cloud**. Presentations Guide

[3] Borriello, G.; Chou, P.; and Ortega, R. "**Embedded System Co-design: Towards Portability and Rapid Integration**". M. Sami, G. DeMicheli . Kluwer Academic Publishers, 1996,

[4] Xiaocong Fan. **Real-Time Embedded Systems. Design Principles and Engineering Practices** 2015 **NEWSNE** ELSEVIER

[5] www.statista.com

[6] Dr. M. Muller, **Motor Maintenance – A Survey of Techniques and Results**, ACEEE organization New Jersey, 1997.

[7]

[8]

[9]