

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA DE CIÊNCIAS EXATAS E DA COMPUTAÇÃO
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



**MANIPULAÇÃO DE UMA MÃO MECÂNICA UTILIZANDO A
PLATAFORMA IBM WATSON IOT**

DÉBORA DA SILVA BARBOSA

GOIÂNIA

2020

DÉBORA DA SILVA BARBOSA

**MANIPULAÇÃO DE UMA MÃO MECÂNICA UTILIZANDO A
PLATAFORMA IBM WATSON IOT**

Trabalho de Conclusão de Curso apresentado à Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador(a):

Profa. Ma. Ludmilla Reis Pinheiro dos Santos

Banca examinadora:

Profa. Ma. Mírian Sandra Rosa Gusmão

Profa. Dra. Carmen Cecilia Centeno

GOIÂNIA
2020

DÉBORA DA SILVA BARBOSA

**MANIPULAÇÃO DE UMA MÃO MECÂNICA UTILIZANDO A
PLATAFORMA IBM WATSON IOT**

Trabalho de Conclusão de Curso aprovado em sua forma final pela Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, para obtenção do título de Bacharel em Engenharia de Computação, em ____/____/_____.

Orientadora: Ludmilla Reis Pinheiro dos Santos

Profa. Ma. Ludmilla Reis Pinheiro dos Santos
Coordenadora de Trabalho de Conclusão de Curso

GOIÂNIA
2020

AGRADECIMENTOS

Expresso minha eterna gratidão a minha mãe Maria José da Silva, que com seu esforço e trabalho me permitiu chegar até aqui, a qual me apoiou a alcançar esse sonho e me ensinou a nunca desistir diante das dificuldades.

Agradeço aos meus amigos Ismael Carlos e Vitoria Nunes pelo incentivo, companheirismo e memórias que sempre ficarão guardadas.

Deixo meu agradecimento a minha orientadora Profa. Ma. Ludmilla Reis Pinheiro dos Santos, pela confiança em mim depositada, mesmo em meio a uma pandemia me encorajou a não desistir. O seu esforço e amor pela profissão me inspiram a ser uma pessoa e uma profissional melhor.

Agradeço a todos aqueles que, por um lapso, esqueci de mencionar, mas que foram importantes para a realização deste trabalho. O apoio, incentivo, amor e carinho de todos foram extremamente necessários.

“O seu foco é a sua realidade.”

Mestre Yoda.

RESUMO

A mão mecânica é um dispositivo que possui a capacidade de imitar os movimentos e características de uma mão humana. Aliada com o conceito de Internet das Coisas transforma-se em uma ferramenta poderosa, a qual poderá ser manipulada de qualquer local do mundo que possua internet. Este trabalho tem como objetivo a implementação de um protótipo de uma mão mecânica controlado por uma página WEB, utilizando os conceitos de Internet das Coisas, a plataforma IBM Watson IoT e o microcontrolador ESP32, trazendo benefícios como, precisão e segurança para as operações que podem ser extremamente difíceis ou prejudiciais ao executante. As metodologias utilizadas foram pesquisas bibliográficas, pesquisas exploratórias e pesquisas experimentais, obtendo-se resultados satisfatórios, em decorrência do protótipo da mão mecânica ter realizado movimentos similares à de uma mão humana.

***Palavras-Chave:** Microcontrolador ESP32, IBM Watson, Internet das Coisas, Mão mecânica.*

ABSTRACT

The mechanical hand is a device that has the ability to imitate the movements and characteristics of a human hand. Allied with the Internet of Things concept, it becomes a powerful tool, which can be manipulated from anywhere in the world that has internet. This work aims to implement a prototype of a mechanical hand controlled by a WEB page, using the concepts of Internet of Things, the IBM Watson IoT platform and the ESP32 microcontroller, bringing benefits such as accuracy and safety to the operations that can be extremely difficult or harmful to the performer. The methodologies used were bibliographic research, exploratory research and experimental research, obtaining satisfactory results, due to the prototype of the mechanical hand having performed movements similar to that of a human hand.

Keywords: *ESP32 microcontroller, IBM Watson, Internet of Things, Mechanical hand.*

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 - Da Vinci Surgical System..... | 24 |
| Figura 2 - Modelo final da prótese. | 26 |
| Figura 3 – Exemplos de algumas aplicações IoT. | 27 |
| Figura 4 - Alimentação de gateways IoT com computação em nevoa e microrredes. | 29 |
| Figura 5 - Prótese impressa em 3D com os motores alocados. | 30 |
| Figura 6 - Arquitetura proposta. | 32 |
| Figura 7 - Fluxograma do projeto..... | 33 |
| Figura 8 - Editor de fluxo baseado em navegador do Node-RED..... | 34 |
| Figura 9 - NodeMCU e suas portas de entrada e saída..... | 35 |
| Figura 10 - Exemplo de servomotor..... | 37 |
| Figura 11 - Modelo 3D do corpo da mão mecânica..... | 38 |
| Figura 12 - Esquema de impressão 3D da mão mecânica..... | 39 |
| Figura 13 - Mão mecânica montada..... | 40 |
| Figura 14 - Servomotores fixados na base..... | 40 |
| Figura 15 - Diagrama do circuito de teste. | 42 |
| Figura 16 - Código de teste..... | 43 |
| Figura 17 - Diagrama do circuito de teste com os cinco servomotores..... | 44 |
| Figura 18 - Página da Plataforma IBM Watson IoT..... | 45 |
| Figura 19 - Status da conexão no monitor serial. | 45 |
| Figura 20 - Status da conexão na Plataforma IBM Watson IoT..... | 46 |
| Figura 21 - Chave API criada na Plataforma IBM Watson IoT. | 47 |
| Figura 22 - Visão geral do aplicativo criado no IBM Cloud. | 47 |
| Figura 23 - Conexão do Node-RED à Plataforma IBM Watson IoT estabelecida. | 48 |
| Figura 24 - Interface gráfica, modo manual. | 48 |
| Figura 25 - Interface gráfica, modo automático. | 49 |
| Figura 26 – Componentes de hardware utilizados..... | 50 |
| Figura 27 - Movimentos realizados no modo manual da página WEB..... | 51 |
| Figura 28 - Movimentos realizados no modo manual. | 52 |
| Figura 29 - Movimentos do modo automático da página WEB. | 52 |
| Figura 30 - Movimentos do modo automático. | 53 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Músculos e Funções..... | 17 |
| Tabela 2 - Valores gastos em componentes de hardware. | 54 |

LISTA DE SIGLAS E ABREVIATURAS

| | |
|-------|--|
| 3D | <i>Three-dimensional space</i> |
| ABS | <i>Acrilonitrila Butadieno Estireno</i> |
| API | <i>Application Programming Interface</i> |
| BOD | <i>Brown Out Detector</i> |
| CPU | <i>Central Processing Unit</i> |
| CSV | <i>Comma Separated Values</i> |
| D2D | <i>Device-to-Device</i> |
| EMG | <i>Electromyographic Sign</i> |
| GPIOs | <i>General Purpose Input/Output</i> |
| IBM | <i>International Business Machines</i> |
| IoT | <i>Internet of Things</i> |
| M2M | <i>Machine to Machine</i> |
| MIT | <i>Massachusetts Institute of Technology</i> |
| MQTT | <i>Menssage Queue Telemetry Transport</i> |
| P&G | <i>Procter & Gamble</i> |
| PLA | <i>Poliácido Láctico</i> |
| PWM | <i>Pulse Width Modulation</i> |
| RFID | <i>Radio-Frequency IDentification</i> |
| RMS | <i>Root Mean Square</i> |
| TLS | <i>Transport Layer Security</i> |
| USB | <i>Universal Serial Bus</i> |

SUMÁRIO

| | |
|---|----|
| 1. INTRODUÇÃO | 13 |
| 1.1 Objetivos | 14 |
| 1.1.1 Objetivos Gerais | 14 |
| 1.1.2 Objetivos Específicos | 14 |
| 1.2 Procedimentos metodológicos | 14 |
| 1.3 Justificativa | 15 |
| 1.4 Resultados esperados | 15 |
| 1.5 Organização do trabalho | 16 |
| 2. REFERENCIAL TEÓRICO | 17 |
| 2.1 A Mão | 17 |
| 2.1 Mão Mecânica | 18 |
| 2.2 Protocolo MQTT | 19 |
| 2.3 Internet das Coisas | 20 |
| 2.4 IBM Watson..... | 21 |
| 2.4.1 Plataforma IBM Watson IoT | 22 |
| 2.5 Trabalhos Relacionados..... | 23 |
| 2.5.1 Actualidad de la cirugía robótica | 23 |
| 2.5.2 Controle de Prótese Mecânica Aplicado à Robótica Assistiva através de Eletromiografia | 25 |
| 2.5.3 O uso da Internet das Coisas (IoT) a favor da saúde | 27 |
| 2.5.4 Interconnecting Fog Computing and Microgrids for Greening IoT | 28 |
| 2.5.5 Desenvolvimento de uma Prótese Robótica Microprocessada | 29 |
| 2.6 Considerações sobre os trabalhos relacionados | 30 |
| 3. PROPOSTA DE SOLUÇÃO..... | 32 |
| 3.1 Arquitetura proposta | 32 |

| | |
|--|----|
| 3.2 Componentes de software..... | 33 |
| 3.2.1 Node-RED | 33 |
| 3.3 Componentes de hardware..... | 34 |
| 3.3.1 Microcontrolador ESP32 | 35 |
| 3.3.2 Servomotores | 36 |
| 3.3.3 Mão mecânica impressa em 3D | 37 |
| 4. TESTES | 42 |
| 4.1 Angulatura dos servomotores | 42 |
| 4.2 Conexão dos cinco servomotores | 43 |
| 4.3 Conexão ESP32 à Plataforma IBM Watson IoT | 44 |
| 4.4 Conexão Node-RED à Plataforma IBM Watson IoT | 46 |
| 5. RESULTADOS | 50 |
| 5.1 Integração dos componentes..... | 50 |
| 6. CONSIDERAÇÕES FINAIS | 55 |
| 6.1 Trabalhos Futuros | 55 |
| REFERÊNCIAS | 57 |
| APÊNDICE A - Código de conexão dos cinco servomotores..... | 61 |
| APÊNDICE B - Código de conexão do ESP32 à Plataforma IBM Watson IoT..... | 62 |
| APÊNDICE C - Código final ESP32 para mão mecânica..... | 64 |
| ANEXO A – Termo de Autorização de Publicação de Produção Acadêmica | 70 |

1. INTRODUÇÃO

O elevado crescimento tecnológico, tem impulsionado um maior envolvimento da robótica nas tarefas rotineiras do ser humano, o que viabilizou o desenvolvimento da mão mecânica, a qual pode ser definida como, a aplicação de conhecimentos da biologia e robótica na resolução de problemas inerentes a engenharia e design (RIBEIRO, 2018).

Mãos mecânicas são elementos robóticos que imitam as características físicas e funcionais de uma mão humana. As mesmas associadas com o conceito de Internet das Coisas, se transformam em uma ferramenta poderosa, para o contexto da medicina e indústrias, a qual poderá ser controlada de qualquer local que possua internet (RIBEIRO, 2018).

Internet das Coisas (*Internet of Things*, IoT) é o conceito de integrar vários tipos de sensores e diversos dispositivos eletrônicos via internet, o que permite a comunicação e a troca de informações. Com isso, obtém-se uma rede que interconecta equipamentos e objetos pela internet, capaz de possuir funções de reconhecimento inteligente, localização, rastreamento e gerenciamento de dispositivos (AGRAWAL; VIEIRA, 2013).

Segundo Oliveira (2017), Internet das Coisas é mais do que apenas ligar uma lâmpada pelo *smartphone*, não é somente conectar as “coisas”, incluindo também as pessoas, mas também torná-las inteligentes, sendo capazes de coletar e processar informações do ambiente.

A IBM possui uma plataforma denominada Watson IoT, que permite conectar dispositivos a aplicativos através de conectividade e gerenciamento de dispositivos. IBM Watson IoT oferece recursos de gerenciamento de dados, gerenciamento de riscos e serviços de análise, permitindo a criação de painéis de alertas (NEDELTCHEVA; SHOIKOVA, 2017).

A robótica juntamente com a plataforma IBM Watson IoT, torna possível o controle de uma mão mecânica através de *smartphones*, *notebooks*, *tablets*, dentre outros. Alguns fatores de relevância são:

- Mobilidade: sistemas que utilizam a internet como base para o controle de equipamentos e periféricos, podem ser controlados de qualquer local do mundo;
- Precisão: a utilização de uma mão mecânica possibilita uma maior precisão e força na execução das atividades propostas;
- Segurança: a mão mecânica evita que o operador tenha contato direto com o elemento manipulado, o que impede dores, no caso de operações com cargas, ou doenças e degradações, no caso de exposição a elementos nocivos à saúde.

Diante deste contexto, a proposta deste trabalho é o controle de uma mão mecânica por meio de *smartphones* e *notebooks* utilizando os conceitos de IoT.

1.1 Objetivos

Esta seção apresenta os objetivos gerais e específicos para o desenvolvimento deste trabalho.

1.1.1 Objetivos Gerais

O objetivo deste trabalho é utilizar o conceito de Internet das Coisas, para controlar uma mão mecânica, através de uma página WEB, a qual será desenvolvida a partir da ferramenta de desenvolvimento visual Node-RED, podendo ser acessada através de dispositivos móveis.

Como estudo de caso, será implementado o controle de um protótipo de uma mão mecânica, utilizando a plataforma IBM Watson IoT, sendo que a comunicação usa o protocolo *Message Queue Telemetry Transport* (MQTT).

1.1.2 Objetivos Específicos

- Desenvolver uma mão mecânica, em uma impressora 3D, constituído de atuadores para repetir os movimentos enviados pelo *smartphone* ou *notebook*;
- Desenvolver um sistema de controle da mão mecânica;
- Desenvolver uma aplicação utilizando o IBM Watson IoT;
- Desenvolver uma página WEB utilizando Node-RED;
- Utilizar o protocolo MQTT para fazer a comunicação entre os dispositivos.

1.2 Procedimentos metodológicos

Segundo Wazlawick (2014), a pesquisa explicativa além de analisar os dados, busca suas causas e explicações. Desta forma, para este trabalho são feitas buscas e pesquisas por características do problema e das soluções já existentes que podem ser otimizadas para permitir a construção e controle da mão biônica.

Quanto aos procedimentos técnicos o trabalho inicialmente tem foco em uma pesquisa bibliográfica, pesquisa exploratória e posteriormente pesquisa experimental. A pesquisa bibliográfica não produz qualquer conhecimento novo, apenas supre o pesquisador de informações públicas que ele ainda não possuía (WAZLAWICK, 2014).

A pesquisa exploratória permite uma maior familiaridade entre o pesquisador e o tema pesquisado, visto que esse ainda é pouco conhecido e pouco explorado pelo pesquisador. Uma pesquisa experimental implica que o pesquisador sistematicamente provocará alterações no ambiente a ser pesquisado (GIL, 2002; GERHARDT, 2009; WAZLAWICK, 2014).

Conforme Gil (2002), as etapas a serem realizadas nas pesquisas são:

- Levantamento bibliográfico;
- Formulação do problema;
- Pesquisa de trabalhos relacionados;
- Construção das hipóteses;
- Definição do plano experimental;
- Análise e interpretação dos dados;
- Apresentação das conclusões.

De acordo com os estudos realizados, o desafio deste trabalho está na integração do microcontrolador ESP32 com a ferramenta de desenvolvimento visual Node-RED, utilizando a plataforma IBM Watson IoT. Essas ferramentas possibilitam o controle da mão mecânica através de dispositivos com conexão à internet.

1.3 Justificativa

Com o crescente envolvimento da robótica nas tarefas do ser humano, a possibilidade da manipulação de materiais, em longas distâncias, através de uma mão mecânica, se torna real e com benefícios de suma importância, principalmente em aplicações cirúrgicas, de robôs humanoides e manipulação de materiais.

Atualmente a IBM tem investido na computação em nuvem, tornando assim seus serviços mais baratos, ágeis, seguros e acessíveis. A plataforma IBM Watson IoT tem sido utilizada por diversas empresas, por causa da possibilidade de extração de valores em dispositivos IoT, o que gera crescimento e facilidade no controle de negócios, automação, dentre outros (MANZONI JUNIOR, 2019).

Por ser uma plataforma em constante crescimento e visibilidade, este trabalho tem o desafio de unir IBM Watson IoT com a robótica, com o objetivo de demonstrar a sua usabilidade computacional.

1.4 Resultados esperados

Espera-se que os resultados deste trabalho contribuam para pesquisas na área da robótica que envolvam IoT e o IBM Watson, plataforma de serviços cognitivos da IBM, pois a combinação dessas ferramentas trará maior acessibilidade ao controle de robôs humanoides, realização de cirurgias e manipulação de materiais de qualquer local com conexão à internet.

1.5 Organização do trabalho

Este trabalho está organizado em 6 capítulos. No capítulo 1 foi apresentado a introdução, os objetivos e as metodologias aplicadas na realização deste, assim como, as justificativas e os resultados esperados.

O capítulo 2 apresenta a fundamentação teórica e alguns trabalhos correlatos que auxiliam à uma melhor compreensão das técnicas e ferramentas utilizadas no desenvolvimento deste trabalho.

O capítulo 3 apresenta a proposta de solução para a construção e controle da mão mecânica, bem como os componentes de software e hardware que serão utilizados.

O capítulo 4 apresenta os testes realizados com os servomotores, a plataforma IBM Watson IoT, com conexão ao Node-RED.

O capítulo 5 apresenta os resultados obtidos e o capítulo 6 apresenta as considerações finais deste trabalho.

2. REFERENCIAL TEÓRICO

Este capítulo apresenta a fundamentação teórica para o desenvolvimento deste trabalho.

2.1 A Mão

Segundo Galindo e Pereira (2018), a mão é o principal órgão motor do corpo humano, estando presente na maior parte das atividades cotidianas, que em sua complexidade funciona como uma estrutura perfeitamente lógica, executando a manipulação de objetos de diferentes formas, utilizando de sua considerável força, precisão e eficácia.

A mão pode alcançar qualquer parte do corpo por conta da sua amplitude de espaço, derivada da mobilidade articular do ombro, do cotovelo e do punho. Ela possui duas características distintas, a primeira é a capacidade de movimentos e flexibilidade, e a segunda é a capacidade sensorial existente em toda a sua estrutura.

A sua importância vem sendo relatada em diversas pesquisas arqueológicas, onde se aponta o sucesso da evolução humana, a partir do funcionamento do polegar opositor sobre as primeiras ferramentas primitivas, possibilitando a sua sobrevivência sob os mais diversos desafios do cotidiano (XAVIER, 2016).

Analisando uma mão, observa-se que ela é bastante complexa pelos seus diversos ossos e músculos, que lhe dão vinte e três graus de liberdade, distribuídos da seguinte forma: (a) cinco graus de liberdade para o polegar; (b) quatro graus de liberdade para cada um dos demais dedos; e (c) dois graus de liberdade para a palma da mão. Sendo essa quantidade de graus de liberdade existente devido a uma estrutura de vinte e sete ossos, dezessete articulações, dezenove músculos e diversos tendões ligados aos músculos (GALINDO; PEREIRA, 2018).

Na Tabela 1 são apresentados os músculos inerentes ao movimento da mão humana e suas respectivas funções motoras.

Tabela 1 - Músculos e Funções.

| MÚSCULOS | FUNÇÕES |
|---|--|
| Lumbricais – Interósseos dorsais e palmares | Flexão de articulação do metacarpo |
| Extensor comum dos dedos; Extensor próprio do indicador; Extensor próprio do 5º dedo; | Extensão das articulações do metacarpo |

| MÚSCULOS | FUNÇÕES |
|---|--|
| Flexor superficial dos dedos (proximais) Flexor profundo dos dedos (distais) | Flexão e articulação interfalangianas proximais e dorsais |
| Interósseos dorsais – Abdutor do 5º dedo | Abdução dos dedos (abdutor do 5º dedo, apenas) |
| Interósseos palmares | Abdução dos dedos |
| Abdutor longo e breve do polegar Longo palmar | Abdução do polegar |
| Adutor do polegar – oblíquo transverso | Adução do polegar |
| Oponente do polegar Oponente do 5º dedo | Oposição do polegar e 5º dedo |
| Flexor breve e longo do polegar | Flexão das articulações Metacarpofalangianas e Interfalangianas do polegar |
| Extensor longo e breve do polegar | Extensão das articulações Metacarpofalangianas e Interfalangianas do polegar |

Fonte: XAVIER, 2016.

Sendo um membro tão importante e eficiente, o avanço da tecnologia juntamente com a robótica proporcionou a criação de mãos mecânicas, as quais tentam imitar as características físicas e funcionais de uma mão humana, podendo ser utilizadas em áreas médicas e industriais com a finalidade de obterem resultados mais eficientes e seguros, que apenas uma mão humana poderia proporcionar.

2.1 Mão Mecânica

Em 1980, tentativas de aplicações da robótica em benefício de deficientes foram realizadas, porém, um desenvolvimento mais expressivo, em termos de pesquisa, só foi alcançado a partir do final dos anos 90, com o desenvolvimento tecnológico possibilitando melhorias na velocidade de comunicação, processamento de sinais e algoritmos, dentre outros fatores (SILVA, 2016).

Em 1994, foram utilizados Sinais Eletromiográficos (EMG) para o controle de uma prótese com cinco dedos, uma mão mecânica, tendo em vista o seu controle da forma mais

natural possível. Para isso, os sinais obtidos foram amplificados e filtrados de forma a produzir um sinal de controle relativamente suave (SILVA, 2016).

As aplicações de uma mão mecânica vão desde o uso no contexto da tecnologia assistiva até a manipulação de objetos de maneira teleoperada. Também pode ser utilizada no desenvolvimento de humanoides e andróides. Pelos altos números de graus de liberdade de uma mão, a construção de uma mão mecânica é um problema particularmente interessante no contexto da engenharia, implicando assim, em um sistema de controle relativamente complexo e de alto custo (RIBEIRO, 2018).

Como o objetivo de diminuir o custo de uma mão mecânica, pesquisadores da Universidade de Ottawa no Canadá, em 2014, utilizaram tecnologias de baixo custo para analisar e desenvolver um algoritmo de manipulação para mãos mecânicas em tempo real e optaram por confeccionar o protótipo em uma impressora 3D (XAVIER, 2016).

Segundo Ribeiro (2018), o funcionamento de uma mão mecânica se dá em receber e interpretar os sinais de controle que são usados para movimentar os componentes físicos da mão, realizando com isso o movimento esperado.

Contudo, essa movimentação pode ser aperfeiçoada se for utilizado o conceito de Internet das Coisas, o qual dará uma maior mobilidade para o controle da mão mecânica e poderá ser feito de qualquer lugar do mundo que possua acesso à internet.

2.2 Protocolo MQTT

O MQTT é um protocolo de mensagens padrão OASIS desenvolvido pela IBM e Eurotech na década de 90 com sua primeira versão destinada para monitoramento de oleodutos (MQTT, 2020).

Extremamente leve, o MQTT é baseado em *publish/subscribe* (publicação/assinatura), onde seus princípios fundamentais são a minimização do uso da banda de rede e os recursos dos dispositivos, enquanto assegura confiabilidade e segurança de comunicação. Esses princípios o tornam ideal para comunicações *Machine to Machine* (M2M) e para aplicações IoT (MQTT, 2020).

O protocolo MQTT define dois tipos de indivíduos na rede, o *broker* e os clientes. O *broker* é um servidor que funciona como um intermediário entre os clientes, ou seja, recebe uma mensagem de um cliente e encaminha a mensagem ao cliente destinatário. Os clientes são dispositivos que podem interagir com o *broker* e receber ou enviar mensagens (PINTO, 2017).

O MQTT possui o seguinte funcionamento (BASÍLIO, 2020):

- O cliente conecta-se ao *broker* por meio de uma conexão TCP/IP simples, o servidor *broker* implementa estruturas de armazenamento de dados referenciadas como tópicos, com isso o cliente pode assinar qualquer tópico de mensagem no *broker*;
- O cliente publica as mensagens em um tópico, enviando a mensagem e o tópico ao *broker*;
- Em seguida, o broker encaminha a mensagem a todos os clientes que assinam esse tópico.

A conexão à internet é um requisito para dispositivos utilizarem o conceito de IoT. O protocolo fundamental da internet é o TCP/IP e o MQTT foi construído por meio deste protocolo, se tornando o protocolo padrão para comunicação da Internet das Coisas (PINTO, 2017).

2.3 Internet das Coisas

A discussão sobre objetos conectados está presente desde o princípio da tecnologia de informação. Bill Joy, cofundador da Sun Microsystems, na década de 1990 refletia sobre a conexão dispositivo para dispositivo (*Device-to-Device*, D2D), pensando em um tipo de conexão que englobava várias redes (MAGRANI, 2018).

Possibilidades reais de transformação se aproximaram com a evolução da comunicação proativa, através de computadores independentemente da interferência humana, ou seja, a comunicação direta entre computadores, conhecida como *Machine to Machine* (M2M) (MIRA, 2019).

Na década de 2000, M2M era considerado como a Internet das Coisas, visto que referia-se às tecnologias que permitiam que sistemas se comunicassem com outros sistemas, ou máquinas com outras máquinas, em tempo real com ou sem fio (PINOCHET, 2014). Porém, Internet das Coisas vai além da comunicação M2M, pois propõe que os objetos sejam conectados e comuniquem-se de forma inteligente a todo momento e de qualquer lugar.

O termo *Internet of Things* (IoT) ou Internet das Coisas, foi apresentado primeiro por Kevin Asthon do MIT (*Massachusetts Institute of Technology*) em uma apresentação sobre RFID (*Radio-Frequency IDentification*) e a cadeia de suprimentos da *Procter & Gamble* (P&G) em 1999. O conceito abrange a comunicação e processamento de diversos tipos de equipamentos, e se constitui de um paradigma tecnológico, no qual objetos físicos estão conectados na rede e são acessados através da internet, sem restrição de momento ou lugar (COLOMBO; LUCCA FILHO, 2018).

Atualmente, as aplicações de IoT são diversas e englobam desde o monitoramento da saúde de um indivíduo, o uso de dispositivos pessoais conectados, até o controle de um sistema de automação (PINTO, 2017).

Com o mundo globalizado, raramente encontra-se alguém que não possui um aparelho conectado à internet, sendo assim, a IoT tem gerado oportunidades para investimentos nas empresas e indústrias que trabalham com equipamentos e ferramentas que coletam dados da nuvem. Atualmente, as pessoas estão cada vez mais conectadas ao mundo digital, ao utilizar arquivos, sites, dados e informações disponibilizados na internet, tornando assim o uso de IoT uma ferramenta incrivelmente prática e essencial (LUCAS; SILVA, 2018).

A *International Business Machines* (IBM), empresa dos Estados Unidos voltada para a área de informática, é uma das empresas que tem investido na computação em nuvem e IoT, com o objetivo de oferecer soluções rápidas e criativas para problemas de gerenciamento de informação (IBM, 2019a).

2.4 IBM Watson

A IBM teve início no século XIX, com o desenvolvimento de máquinas elétricas para a contagem de dados do censo de 1890 nos Estados Unidos. Em 1924, Thomas J. Watson assumiu o comando da empresa, alavancando assim o seu potencial computacional (CANALTECH, 2019).

Atualmente, a IBM desenvolve, fabrica e vende hardware e software, incluindo sistemas de inteligência artificial, *deep learning*, armazenamento na nuvem e supercomputadores. Como exemplo de sistema de inteligência artificial tem-se o IBM Deep Blue, o primeiro computador a vencer um campeão mundial de xadrez em 1997, o russo Gary Kasparov. Em 2011, a IBM revelou ao mundo o Watson, um supercomputador que é um sistema completo de computação cognitiva (KLEINA, 2017).

A cognição é o processo que a mente humana utiliza para adquirir conhecimento a partir de informações recebidas. O Watson, possui aprendizado contínuo, entende linguagens diversas e gera hipóteses, se favorecendo da linguagem humana e da inteligência artificial para analisar quantidades gigantescas de dados e trazer respostas (CTI GLOBAL, 2019).

Essas respostas são o resultado da junção de vários serviços que o IBM Watson possui, sendo os mais conhecidos (IBM, 2019c):

- IBM Watson *OpenScale*: é a plataforma que rastreia e mede os resultados da inteligência artificial em todo seu ciclo de vida, adaptando e controlando a inteligência artificial para mudar as situações de negócios para modelos construídos e em execução em qualquer lugar;
- IBM Watson *Assistant*: é o serviço do Watson para a criação e desenvolvimento de interfaces conversacionais, ou *chatbots*;

- IBM Watson Discovery: serviço que facilita o desenvolvimento de aplicativos de exploração cognitivos baseados em *cloud* que ajudam a descobrir *insights* acionáveis ocultos em dados não estruturados;
- IBM Watson *Machine Learning*: é um conjunto de *Application Programming Interface* (API) para o desenvolvimento de aplicativos que tomem decisões mais inteligentes, resolvam problemas difíceis e melhorem os resultados do usuário;
- Plataforma IBM Watson IoT: é um serviço hospedado em nuvem gerenciado com recursos para registro de dispositivos, conectividade, controle, visualização rápida e armazenamento de dados;
- IBM *Language Translator*: é um serviço que traduz textos de um idioma para outro;
- IBM Watson *Speech to Text*: é um serviço de conversão de áudio e voz em texto;
- IBM Watson *Visual Recognition*: é um serviço que identifica e classifica o conteúdo visual usando o aprendizado de máquina;
- IBM Watson *Personality Insights*: serviço de previsão de características, necessidades e valores de personalidade por meio de texto.

Esses serviços na área cognitiva, permite que profissionais, desenvolvedores, *startups* e empresas construam sistemas cognitivos que possam melhorar processos, interações e ações. Um desses serviços é a plataforma IBM Watson IoT, o qual possui um serviço que permite a coleta e análise de dados relevantes para dispositivos habilitados para IoT.

2.4.1 Plataforma IBM Watson IoT

Sistemas cognitivos possuem um potencial específico para fazer sistemas e aplicativos, através da incorporação da capacidade cognitiva nos aplicativos e plataformas. A plataforma IBM Watson IoT possui esse potencial, pois ajuda a identificar tendências, anomalias, probabilidades e padrões que de outra forma poderiam ser invisíveis aos usuários (RAJ; RAMAN, 2017).

A plataforma IBM Watson IoT é um serviço integrado ao IBM Watson e hospedado em nuvem, o qual possui um serviço que permite a coleta e análise de dados relevantes para dispositivos habilitados para IoT, ferramentas de gerenciamento de dispositivos e comunicação segura usando os protocolos MQTT e *Transport Layer Security* (TLS), tornando prático, fácil e seguro sua implementação em diferentes projetos (IBM, 2019).

Conexão, captura, monitoramento, análise, armazenamento, gerenciamento e monitoração do uso de dados são os recursos principais da plataforma IBM Watson IoT, que visam um melhor aproveitamento do produto. A plataforma IoT pode ser expandida usando complementos como (IBM, 2019b):

- *IBM Watson IoT Platform on Blockchain*: que permite que seus recursos de IoT participem de redes de negócios de *blockchain*;
- *Watson IoT Platform Analytics*: que permite que os usuários da linha de negócios aprimorem e interajam facilmente com os dados brutos das métricas provenientes de entidades da IoT, usando funções analíticas, configuráveis e integradas.

Segundo Raj e Raman (2017), em breve o domínio da IoT proporcionará a única fonte de dados do planeta, mas quase 90% desses dados nunca serão utilizados. A plataforma IBM Watson IoT possui habilidades exclusivas para detectar, raciocinar e aprender. Com isso, governos e indivíduos poderão aproveitar esses dados, aprofundando conhecimentos e encontrando relações inesperadas que gerarão novas intuições para benefício dos negócios e da sociedade.

2.5 Trabalhos Relacionados

Esta seção apresenta cinco trabalhos relacionados ao tema IoT e robótica.

2.5.1 Actualidad de la cirugía robótica

Actualidad de la cirugía robótica buscou apresentar aspectos gerais e atuais da robótica, aplicada à medicina e principalmente à cirurgia. Para isso, foi realizada uma revisão sistemática em artigos científicos mostrando as etapas da robótica até chegar ao início da robótica médica, a qual começou a partir de 1980, com experimentos cirúrgicos em neurocirurgia e ortopedia (FRAGA, 2017).

Foi suscitado que em 1997, o protótipo chamado Mona, um precursor do atual Da Vinci, que usava um sistema mestre-escravo com um console de mãos e braços independentes foi concluído. Em 1999, Da Vinci, surgiu, sendo que o mesmo, utiliza do sistema de telepresença. A Figura 1 mostra como o Da Vinci é utilizado (FRAGA, 2017).

Figura 1 - Da Vinci Surgical System.



Fonte: FRAGA, 2017.

O autor evidenciou que a cirurgia robótica possui aplicabilidade em procedimentos complexos, com baixo índice de conversões e poucas complicações, sendo que as falhas do robô e seus componentes são extremamente escassas. Foi evidenciado tanto que as desvantagens da cirurgia robótica estão relacionadas a falta de sensação tátil, tamanho do equipamento e o custo. Foi apresentado também, que uma das perspectivas para o futuro da cirurgia robótica, é que o cirurgião possa ter o controle de vários robôs em diferentes salas de operações, ou realize cirurgias em que ele esteja em outra cidade, estado ou país (FRAGA, 2017).

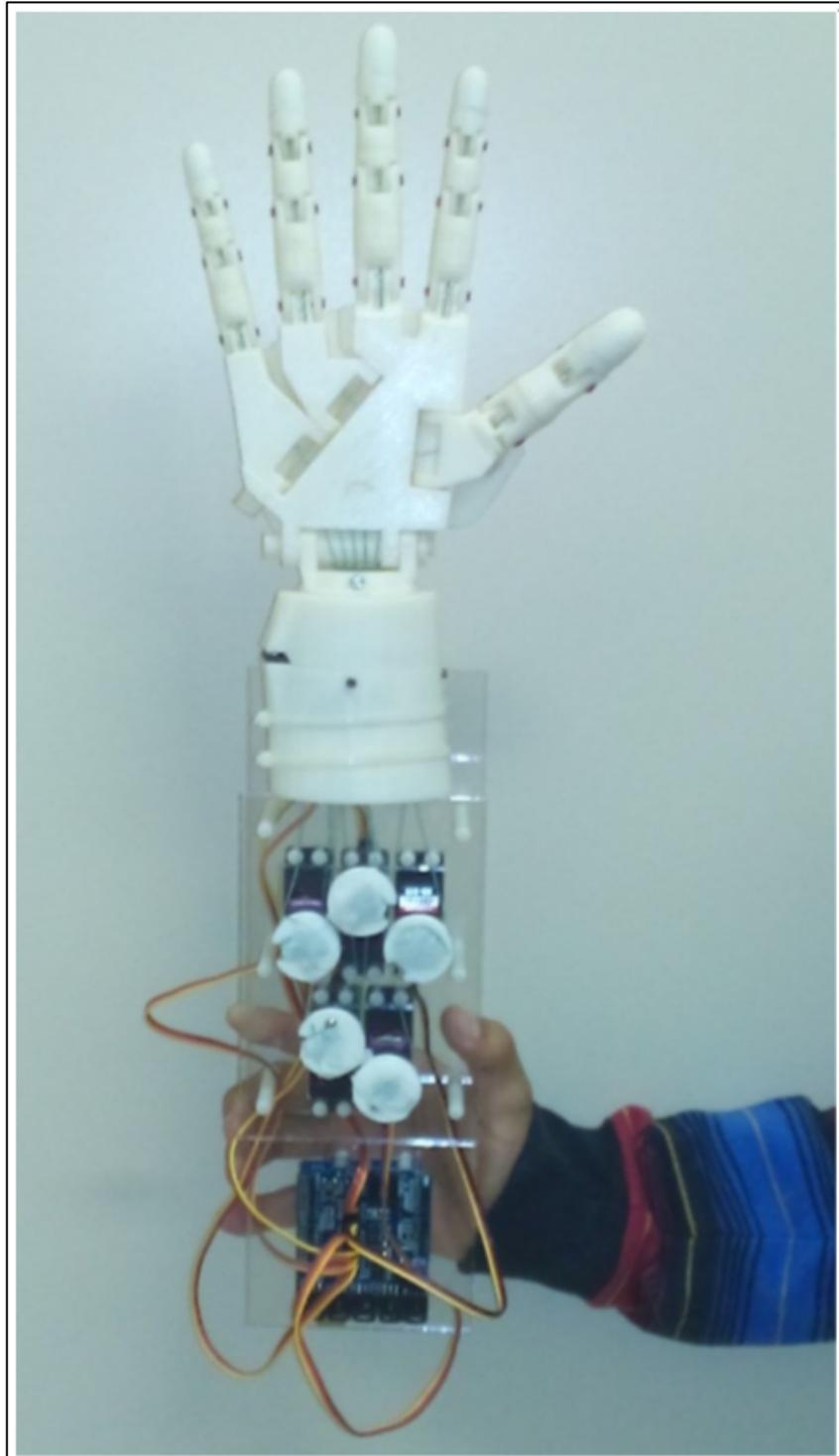
Como resultado obteve-se que, a cirurgia robótica oferece um tratamento adequado e resultados satisfatórios para os pacientes, tais como: (a) menor risco de intervenção e uma recuperação ideal no menor tempo possível; e (b) impacto nas especialidades médicas, uma vez que sua intervenção é mínima e atinge pequenos lugares que o braço humano não poderia alcançar. No entanto, exigirá uma nova geração de médicos treinados para lidar com sistemas complexos (FRAGA, 2017).

2.5.2 Controle de Prótese Mecânica Aplicado à Robótica Assistiva através de Eletromiografia

O trabalho de conclusão de curso ***Controle de Prótese Mecânica Aplicado à Robótica Assistiva através de Eletromiografia*** buscou reproduzir uma prótese robótica de uma mão utilizando uma impressora 3D para a fabricação de suas peças, servomotores e fios de polietileno para o acionamento dos movimentos dos dedos da prótese (SILVA, 2016).

O material utilizado para a impressão foi o polímero Acrilonitrila Butadieno Estireno (ABS) e o modelo da prótese foi obtido de uma comunidade *open source* denominada InMoov. Foi utilizado no trabalho a plataforma Arduino Uno, que é uma plataforma de prototipagem eletrônica de código aberto, baseado no microcontrolador ATmega328P. A Figura 2 mostra a prótese produzida juntamente com os servomotores e o Arduino Uno (SILVA, 2016).

Figura 2 - Modelo final da prótese.



Fonte: SILVA, 2016.

O trabalho realizou um estudo preliminar sobre a possibilidade da utilização do sinal biológico de Eletromiografia para o controle da prótese. Os sinais EMG utilizados foram

oriundos de um banco de dados público sendo utilizados como parâmetro de análise desses sinais o valor quadrático médio (*Root Mean Square, RMS*) (SILVA, 2016).

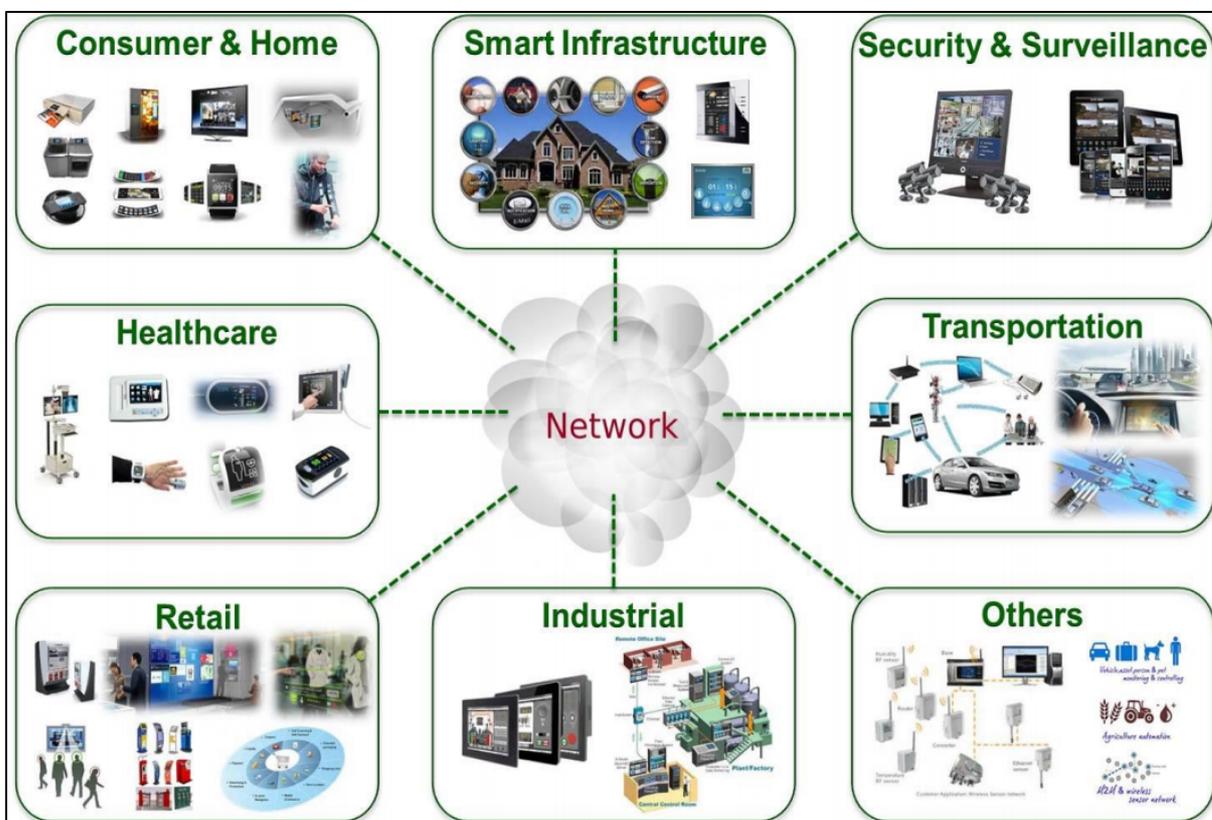
Os resultados obtidos com a interpretação do sinal EMG mostram que a escolha do valor quadrático médio como parâmetro de análise não foi suficiente para realizar a distinção entre o estado de relaxamento e o estado de contração do músculo. Entretanto, a confecção da prótese foi realizada com êxito, bem como o acionamento em ambiente simulado (SILVA, 2016).

2.5.3 O uso da Internet das Coisas (IoT) a favor da saúde

O artigo *O uso da Internet das Coisas (IoT) a favor da saúde* teve como objetivo a exploração do conceito de IoT em aplicações no campo da medicina e nos cuidados da saúde, sendo que seu foco foi evidenciar de forma conceitual, a importância da utilização da IoT na área da saúde (MASSOLA; PINTO, 2018).

Foi descrito uma fundamentação teórica sobre Internet das Coisas, o seu surgimento e algumas de suas aplicações como, residências, sistemas de energia, segurança, transporte, agricultura, saúde, indústria, entre outras, como mostrado na Figura 3.

Figura 3 – Exemplos de algumas aplicações IoT.



Fonte: MASSOLA; PINTO, 2018.

Através de pesquisas científicas em livros, artigos, revistas, jornais, foram encontradas inúmeras aplicações da IoT na área da saúde, sendo as principais e com maior foco (MASSOLA; PINTO, 2018):

- Uso da IoT em um hospital de Boston para acompanhar os recém-nascidos na Unidade de Terapia Intensiva Neonatal (UTIN);
- IoT para gerenciamento de estoque em hospital;
- IoT para monitoramento de um paciente com insuficiência cardíaca crônica.

Os resultados obtidos com a exploração do conceito de IoT na área da saúde foram que os benefícios são muitos, mas que os desafios não são poucos também, pois os executivos da área ficam preocupados com a segurança dos dados e com o gerenciamento dos dispositivos IoT (MASSOLA; PINTO, 2018).

2.5.4 Interconnecting Fog Computing and Microgrids for Greening IoT

O artigo *Interconnecting Fog Computing and Microgrids for Greening IoT* explorou opções diferentes para a implantação de aplicativos IoT com eficiência energética, sendo esta área não muito explorada por pesquisadores, ao invés das áreas de segurança e privacidade, sendo que para essa exploração foi utilizado microrredes e a computação em névoa (JALALI et al., 2016).

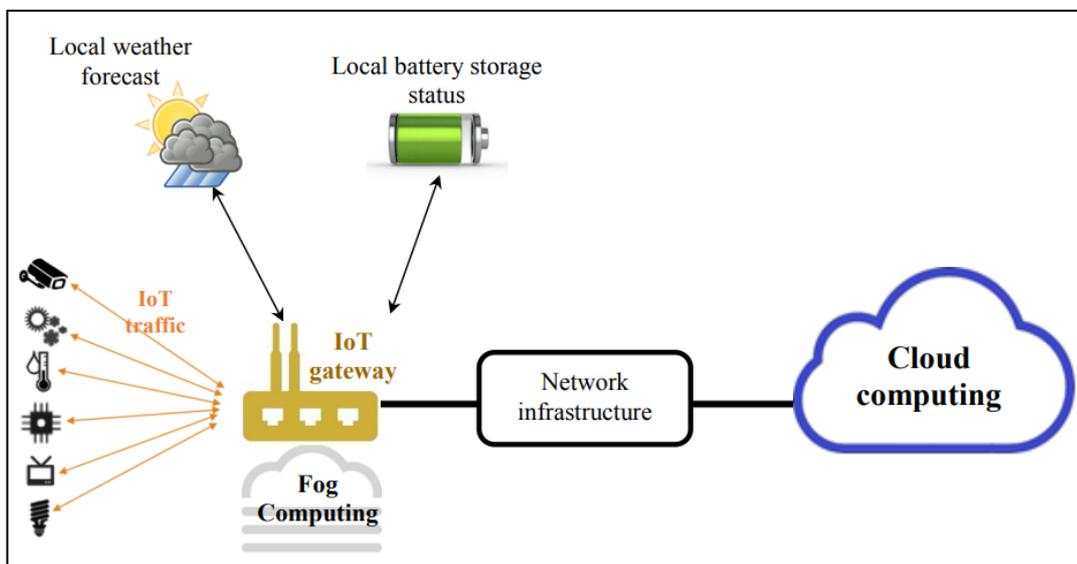
A computação em névoa realiza parte do processamento de dados, armazenamento e filtragem, ao invés de enviar todas as informações para a nuvem. Porém, a computação em névoa não está totalmente independente da nuvem, pois a mesma envia atualizações importantes para sincronização (JALALI et al., 2016).

Microgrids, ou microrredes, são modernas versões em pequena escala da rede elétrica centralizada tradicional, que compreende os geradores, armazenamento de energia e cargas. Contudo, microrredes diferem-se da rede tradicional por sua proximidade com a demanda, resultando no aumento da eficiência e redução no custo de transmissão (JALALI et al., 2016).

A Figura 4 mostra a conexão dos dispositivos e sensores a um *gateway*, que funciona tanto na nuvem como em névoa. O *gateway* obtém os status da bateria local, que foi carregada por uma microrrede residencial, e também possuía acesso a previsão do tempo para estimar a disponibilidade de energia renovável, sendo possível a mudança de uma tarefa da névoa para a

nuvem, ou vice-versa, de acordo com o status da bateria local para maximizar a eficiência energética (JALALI et al., 2016).

Figura 4 - Alimentação de gateways IoT com computação em nevoa e microrredes.



Fonte: JALALI et al., 2016.

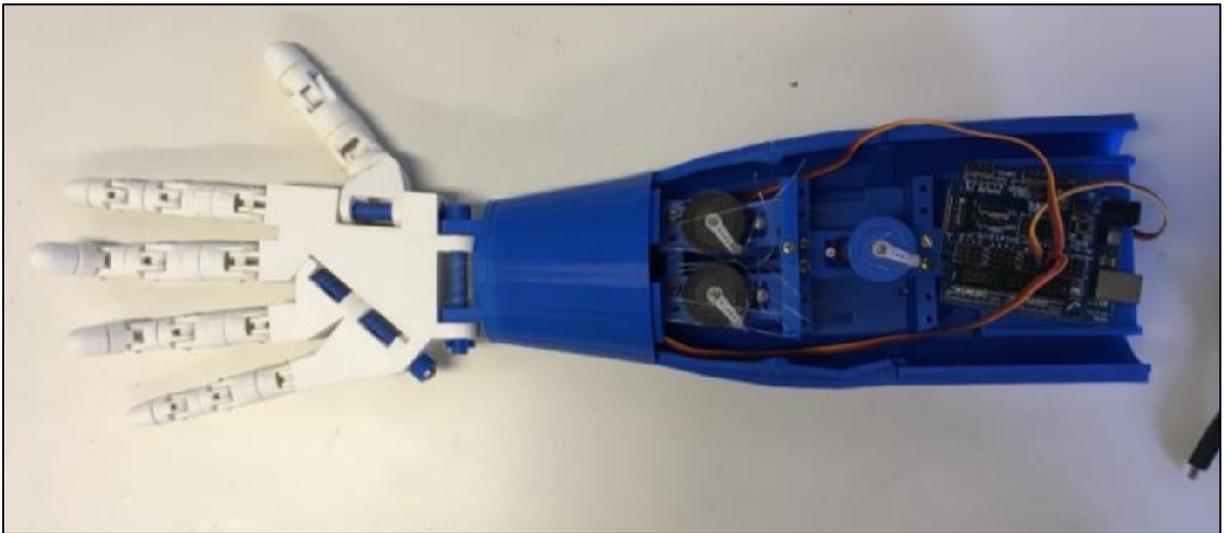
Os resultados obtidos no trabalho indicaram que o uso de microrredes e computação em nevoa pode permitir que aplicativos de IoT consumam menos energia, sendo também, uma estratégia de gerenciamento de energia dinâmica e em tempo real para o uso mais eficiente de energia em aplicações IoT (JALALI et al., 2016).

2.5.5 Desenvolvimento de uma Prótese Robótica Microprocessada

O artigo *Desenvolvimento de uma Prótese Robótica Microprocessada* refere-se à aplicação dos sinais da EMG e da impressão 3D para desenvolvimento de uma prótese robótica acessível à população, pois o custo de uma prótese é entre R\$ 130 mil e R\$ 150 mil, dependendo da extensão da amputação (GUATELLI; MENDES; LIMA; LIMA, 2018).

As peças da prótese foram desenvolvidas e impressas em 3D utilizando o material ABS, com a ajuda do software de edição em três dimensões FreeCAD, e o sinal elétrico do músculo captado pela EMG, convertido de analógico para digital com o uso de um microcomputador *Raspberry Pi 3*, uma plataforma de prototipagem eletrônica que possui microcontrolador, processador e *Bluetooth*. A Figura 5 mostra a prótese impressa na impressora 3D totalmente montada e os motores alocados (GUATELLI; MENDES; LIMA; LIMA, 2018).

Figura 5 - Prótese impressa em 3D com os motores alocados.



Fonte: GUATELLI; MENDES; LIMA; LIMA, 2018.

Os motores utilizados no projeto foram três micro servomotores SG90 TowerPro alimentados por uma bateria de 5V e alocados dentro da prótese, que recebiam o sinal de controle digitalizado realizando movimento de abrir e fechar os dedos da mão biônica. Cada micro servomotor era responsável por um ou dois dedos da prótese e possuía um movimento angular específico determinado no algoritmo, sendo um movimento angular de 60 graus para os dedos polegar e mínimo, 80 graus para o dedo médio e 70 graus para os dedos anelar e indicador (GUATELLI; MENDES; LIMA; LIMA, 2018).

Por meio dos sinais da eletromiografia e da impressão 3D com o material ABS, a prótese de mão biônica que foi desenvolvida possuiu baixo custo, comparado com a tecnologia de próteses robóticas no mercado que custam entre R\$ 130 mil a R\$ 150 mil, e foi capaz de realizar os movimentos de abrir e fechar os dedos da mão (GUATELLI; MENDES; LIMA; LIMA, 2018).

2.6 Considerações sobre os trabalhos relacionados

Após as pesquisas realizadas, pode-se destacar que impressão 3D é uma solução adequada para se obter um baixo custo na construção da mão mecânica. Um ponto em destaque é a usabilidade do protocolo MQTT, pois o mesmo possibilita uma melhor integração e comunicação com sistemas que utilizam o conceito de IoT e computação em nuvem.

Diante do contexto apresentado nos trabalhos relacionados, optou-se pela utilização do microcontrolador ESP32 neste trabalho, pois diferentemente do ESP8266 e o ATmega328P, microcontroladores usados nos trabalhos pesquisados, o ESP32 possui dois núcleos de CPU,

Wi-Fi, maior velocidade de processamento, maior quantidade de *General Purpose Input/Output* (GPIOs) e suporte ao *Bluetooth* 4.2.

3. PROPOSTA DE SOLUÇÃO

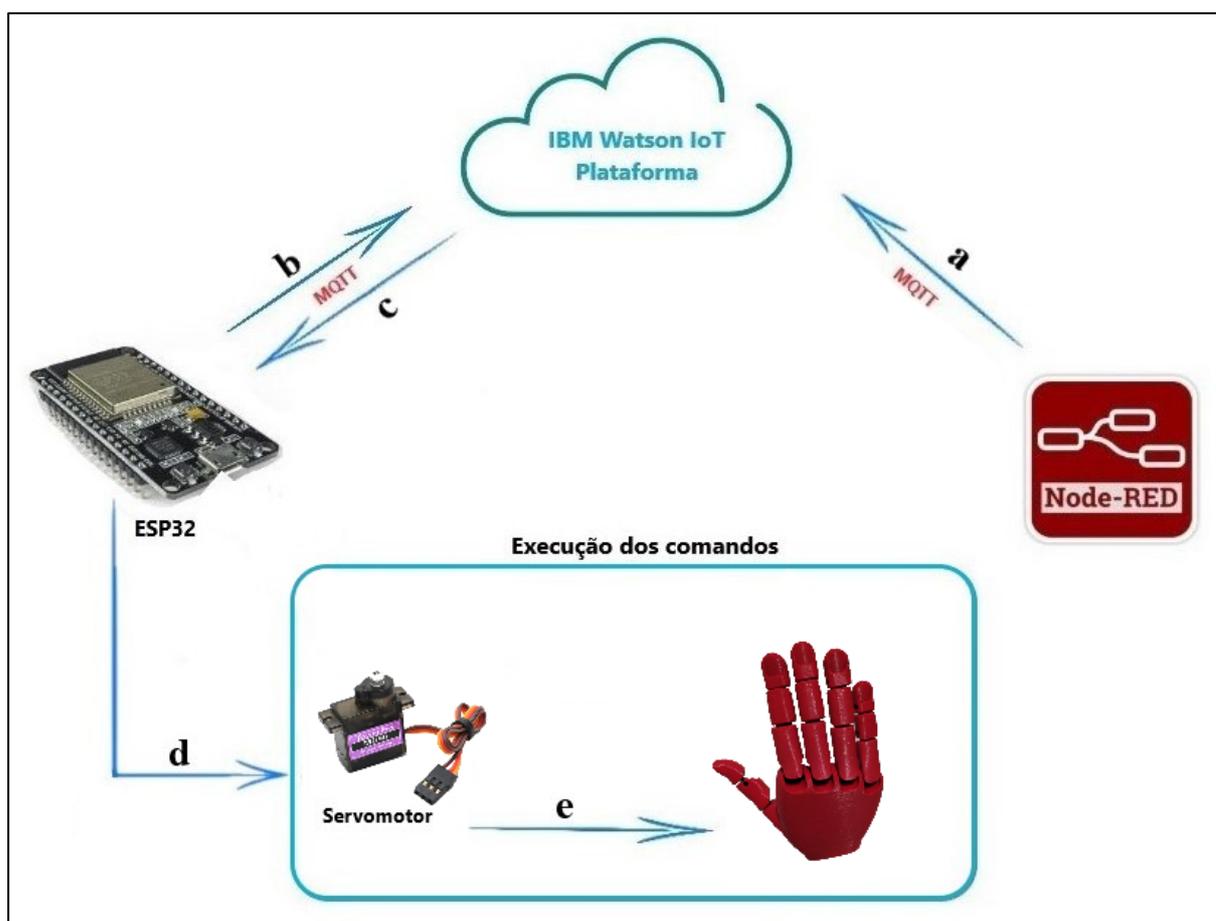
Este capítulo apresenta a arquitetura proposta para a aplicação dos conceitos de robótica e IoT deste trabalho, utilizando ferramentas, plataformas e protocolos de comunicação atuais.

3.1 Arquitetura proposta

A Figura 6 ilustra a arquitetura proposta para este trabalho, onde a plataforma IBM Watson IoT funciona como um *broker*, sendo que os fluxos de informações são:

- (a) Node-RED publica as informações para a plataforma IBM Watson IoT;
- (b) ESP32 faz requisição de informações para a plataforma IBM Watson IoT;
- (c) ESP32 recebe as informações solicitadas;
- (d) ESP32 envia comando para o servomotor;
- (e) Servomotor executa o comando para a movimentação da mão mecânica.

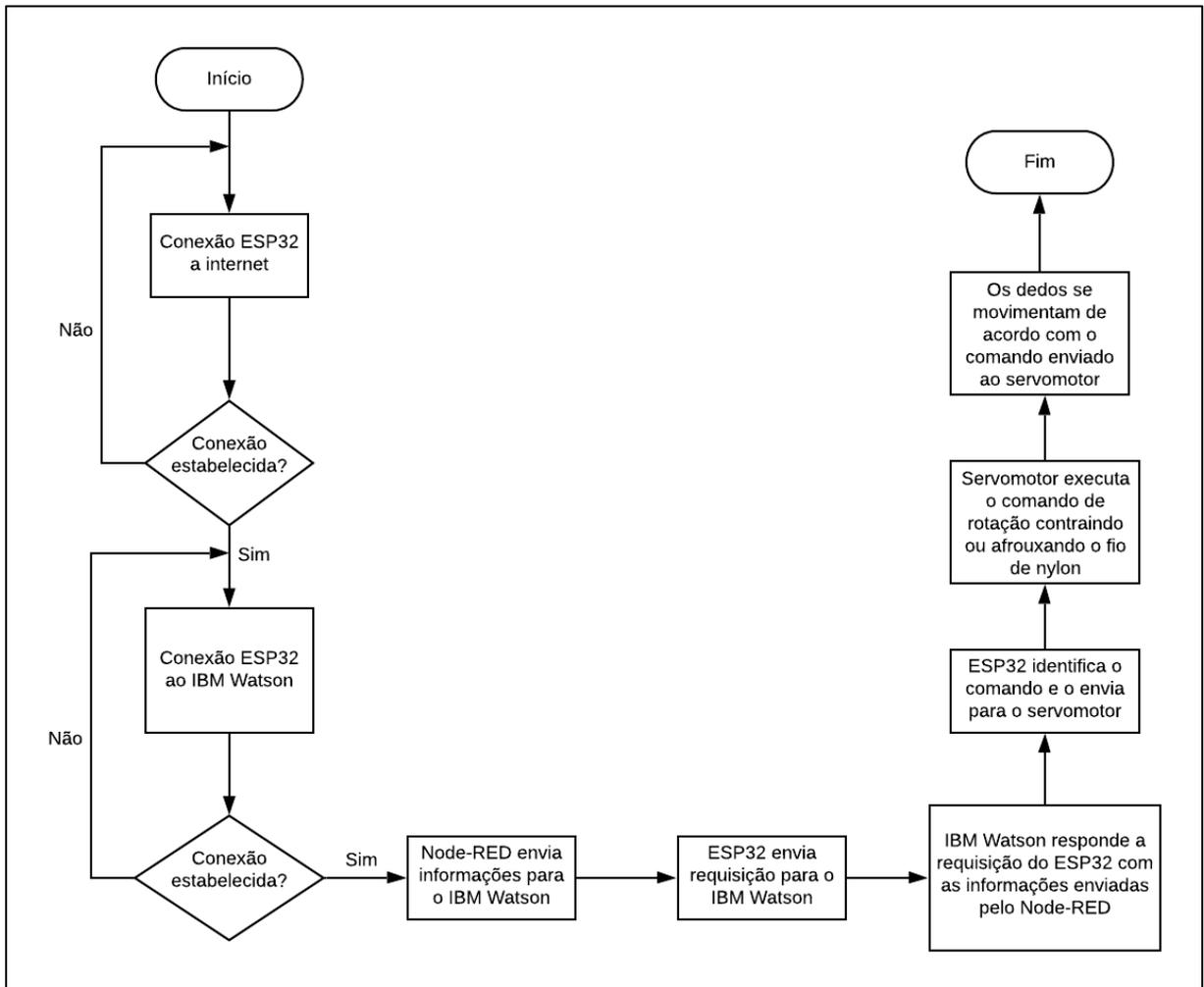
Figura 6 - Arquitetura proposta.



Fonte: Autoria própria.

A Figura 7 apresenta um fluxograma que ilustra de maneira simplificada a manipulação da mão mecânica deste trabalho.

Figura 7 - Fluxograma do projeto.



Fonte: Autoria própria.

3.2 Componentes de software

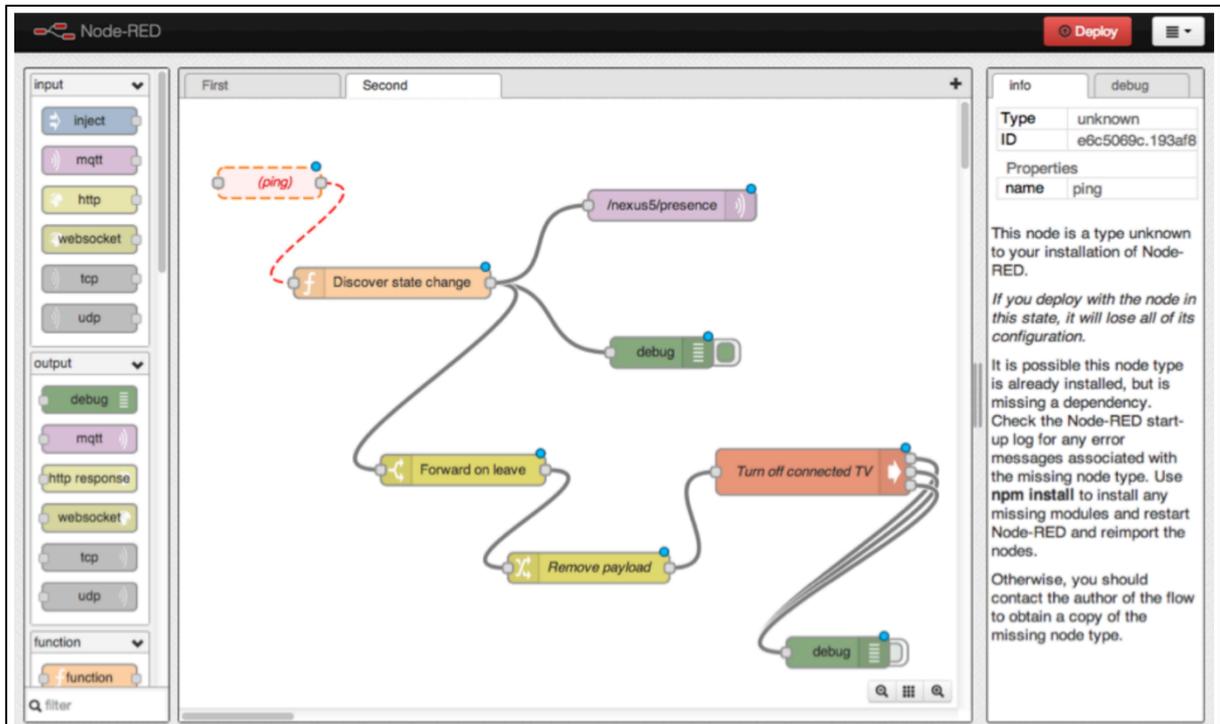
Esta seção apresenta os componentes de software utilizados para a elaboração deste trabalho, sendo que a ferramenta Node-RED fará comunicação com a plataforma IBM Watson IoT.

3.2.1 Node-RED

O Node-RED é uma ferramenta de programação visual, para conectar dispositivos de *hardware*, *Application Programming Interface* (API) e serviços *online*, ele também fornece um editor de fluxo baseado em navegador, sendo um conjunto de nós conectados por fios, como

mostrado na Figura 8. Inventada por J. Paul Morrison na década de 1970, a programação baseada em fluxo é uma maneira de descrever o comportamento de um aplicativo como uma rede de caixas-pretas, ou "nós", como são chamados no Node-RED (BLACKSTOCK; LEA, 2014).

Figura 8 - Editor de fluxo baseado em navegador do Node-RED.



Fonte: BLACKSTOCK; LEA, 2014.

A ferramenta Node-RED possui ambiente de código aberto e é implementada no JavaScript usando a estrutura Node.js, aproveitando o modelo de evento interno do nó e o suporte nativo ao JavaScript no editor do cliente e no servidor. Com isso, por meio dos nós é possível ler arquivos *Comma Separated Values* (CSV), perceber eventos e outros (BASÍLIO, 2019).

O Node-RED foi escolhido como ferramenta para a implementação deste trabalho por ser de uso simples, ter a possibilidade de agregação de ferramentas e possuir a vantagem da IBM disponibilizá-lo com o IBM Watson em seus próprios servidores, tornando vantajoso a utilização do Node-RED em implementações IoT (OLIVEIRA, 2018).

3.3 Componentes de hardware

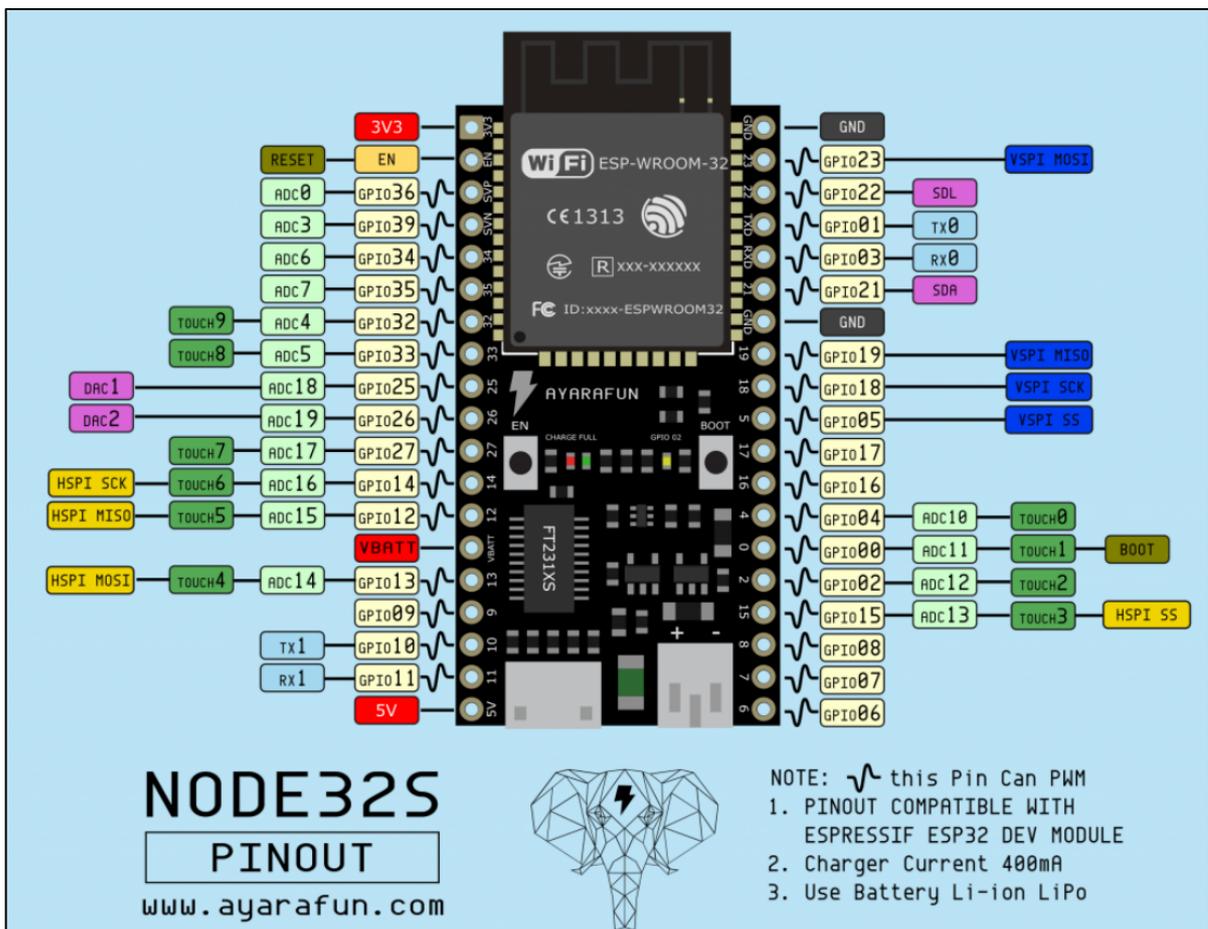
Esta seção apresenta os componentes de hardware utilizados neste trabalho.

3.3.1 Microcontrolador ESP32

O microcontrolador ESP32 foi criado pela *Espressif Systems* para ser um sistema de baixo custo e consumo de energia, possuindo recursos de comunicação *wireless*, *Bluetooth* e um microprocessador Tensilica Xtensa 32-bit LX6 de núcleo duplo com taxa de *clock* de até 240 MHz. O ESP32 é comercializado tanto separado quanto em placas com componentes, por exemplo, regulador de tensão e diferentes números de portas de entrada e saída (ESPRESSIF, 2018).

O NodeMCU é um dos mais completos módulos que aceitam o microcontrolador ESP32, tendo como característica a possibilidade de programar a *chip* através de um ambiente de desenvolvimento integrado, nas linguagens Lua e C/C++. Este módulo possui regulador de tensão que estabiliza a tensão em 3.3V e possui 36 pinos de entrada e saída que permite transmitir dados por um cabo micro *Universal Serial Bus* (USB), como mostrado na Figura 9 (BORGES, 2019).

Figura 9 - NodeMCU e suas portas de entrada e saída.



Fonte: BORGES, 2019.

Comparando o ESP32 a outros microcontroladores, como o ATmega328P, microcontrolador utilizado no Arduino Uno, tem-se como vantagem o ESP32 possuir 2 núcleos, possibilitando assim execuções paralelas, possuir uma arquitetura de 32bits, conexões *wireless* e *Bluetooth*, as quais o Arduino Uno não possui. O ESP32 possui também o modo de hibernação, que permite desligá-lo em momentos em que não houver uso, o que resulta em economia de energia, pois a demanda da mesma é reduzida para mantê-lo em atividade (BORGES, 2019).

3.3.2 Servomotores

Servomotores são dispositivos eletromecânicos utilizados para realizar movimentos de forma precisa e controlada. Estes dispositivos são compostos de um potenciômetro, um motor de corrente contínua, um circuito de controle e um conjunto de engrenagens (SILVA, 2016).

O servomotor possui uma característica de acionamento que é o controle por Modulação de Largura de Pulso (PWM), sendo que, a largura de pulso determina o ângulo de rotação do eixo do servomotor (XAVIER, 2016).

O circuito de controle do servomotor monitora este sinal em períodos de 20 milissegundos e possui nível lógico alto de 5V. Se neste período, o controle detectar uma alteração do sinal, ele altera a posição do eixo para que a sua posição coincida com o sinal recebido (XAVIER, 2016).

O sinal enviado ao servomotor é originado de um circuito de comando, microcontrolado, programado para realizar as rotações necessárias no servomotor. Essas rotações são responsáveis pelo posicionamento do eixo do motor (XAVIER, 2016). Na Figura 10 é mostrado um exemplo de servomotor.

Figura 10 - Exemplo de servomotor.



Fonte: Autoria própria.

3.3.3 Mão mecânica impressa em 3D

A impressão 3D tem recebido atualmente um enfoque especial do mercado, e passou a ser uma realidade tecnológica importante, que possui aplicabilidades diversas e a cada ano apresenta novos potenciais de aplicação, sendo estes desde a objetos decorativos até tecnologias biomédicas (XAVIER, 2016).

Os materiais geralmente utilizados para a impressão 3D são: (a) materiais rígidos e leves, como Poliacido Láctico (PLA) e *Acrylonitrile Butadiene Styrene* (ABS); e (b) materiais flexíveis, como Nylon e Filaflex (GYROBOT LIMITED, 2020).

O projeto 3D da mão mecânica utilizado neste trabalho está disponível em Thingiverse, comunidade online da MakerBot para projetos de impressão 3D de código aberto: <https://www.thingiverse.com/thing:242639/remixes>. A Draw.Up foi a empresa responsável pela impressão 3D e dos ajustes do projeto, dando todo suporte e atenção necessários para uma excelente impressão da mão mecânica.

A escolha do projeto da mão mecânica foi feita pelo realismo do protótipo em relação a mão humana, e também por ele possuir partes flexíveis, tirando a necessidade de adicionar

molas as juntas dos dedos. A Figura 11 apresenta o corpo da mão mecânica escolhido para a realização deste trabalho.

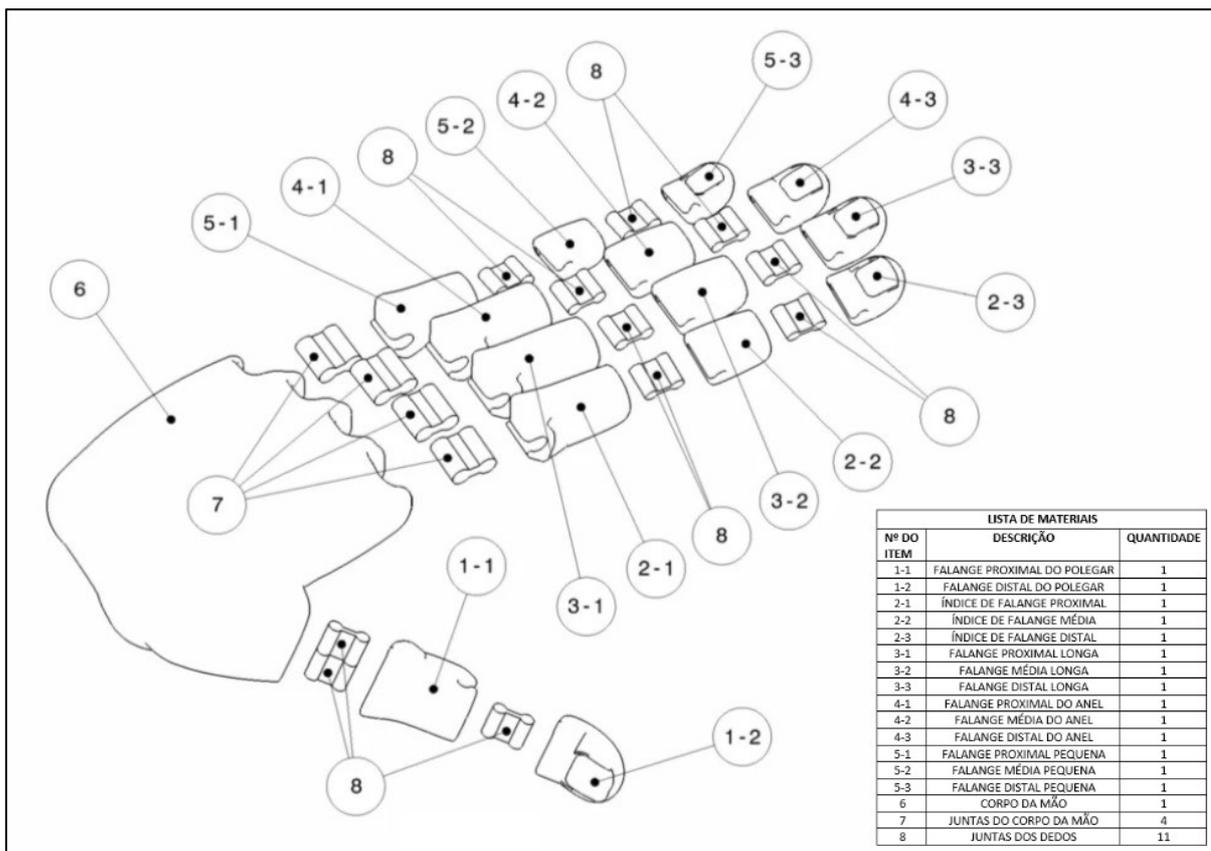
Figura 11 - Modelo 3D do corpo da mão mecânica.



Fonte: GYROBOT, 2014.

A Figura 12 apresenta o esquema de impressão 3D da mão mecânica, onde é possível observar com precisão todas as peças impressas juntamente com o local de encaixe de cada uma delas.

Figura 12 - Esquema de impressão 3D da mão mecânica.



Fonte: Editada de GYROBOT, 2014.

Após a conclusão da impressão as peças foram limpas, retirando o excesso de impressão, encaixadas e ajustadas manualmente até chegar na versão final, Figura 13.

Figura 13 - Mão mecânica montada.



Fonte: Autoria própria.

A Figura 14 mostra a base com os servomotores fixados e prontos para serem conectados aos fios de nylon, os quais funcionam como tendões na mão mecânica.

Figura 14 - Servomotores fixados na base.



Fonte: Autoria própria.

4. TESTES

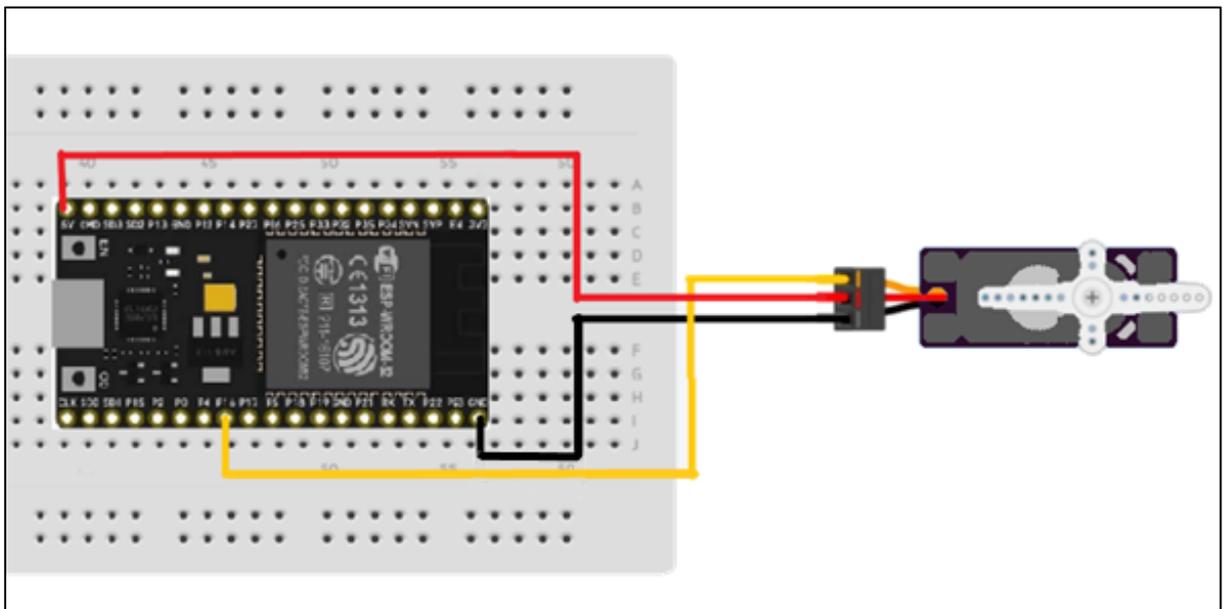
Este capítulo apresenta os testes realizados para a execução deste trabalho.

4.1 Angulatura dos servomotores

Este teste teve como objetivo verificar a real angulatura (0° à 180°) dos servomotores utilizados no protótipo.

Na Figura 15 é apresentado o diagrama de ligação do circuito utilizado para o teste, onde o servo motor está conectado ao microcontrolador ESP32. A Figura 16 apresenta o código fonte utilizado para o teste, o qual foi escrito e executado utilizando o software Arduino IDE.

Figura 15 - Diagrama do circuito de teste.



Fonte: Autoria própria.

Figura 16 - Código de teste.

```
01-SimpleServo $
#include <Servo.h>

static const int servoPin = 27;

Servo servol;

void setup() {
  Serial.begin(115200);
  servol.attach(servoPin);
}

void loop() {
  for(int posDegrees = 0; posDegrees <= 180; posDegrees++) {
    servol.write(posDegrees);
    delay(20);
  }
}
```

Fonte: Autoria própria.

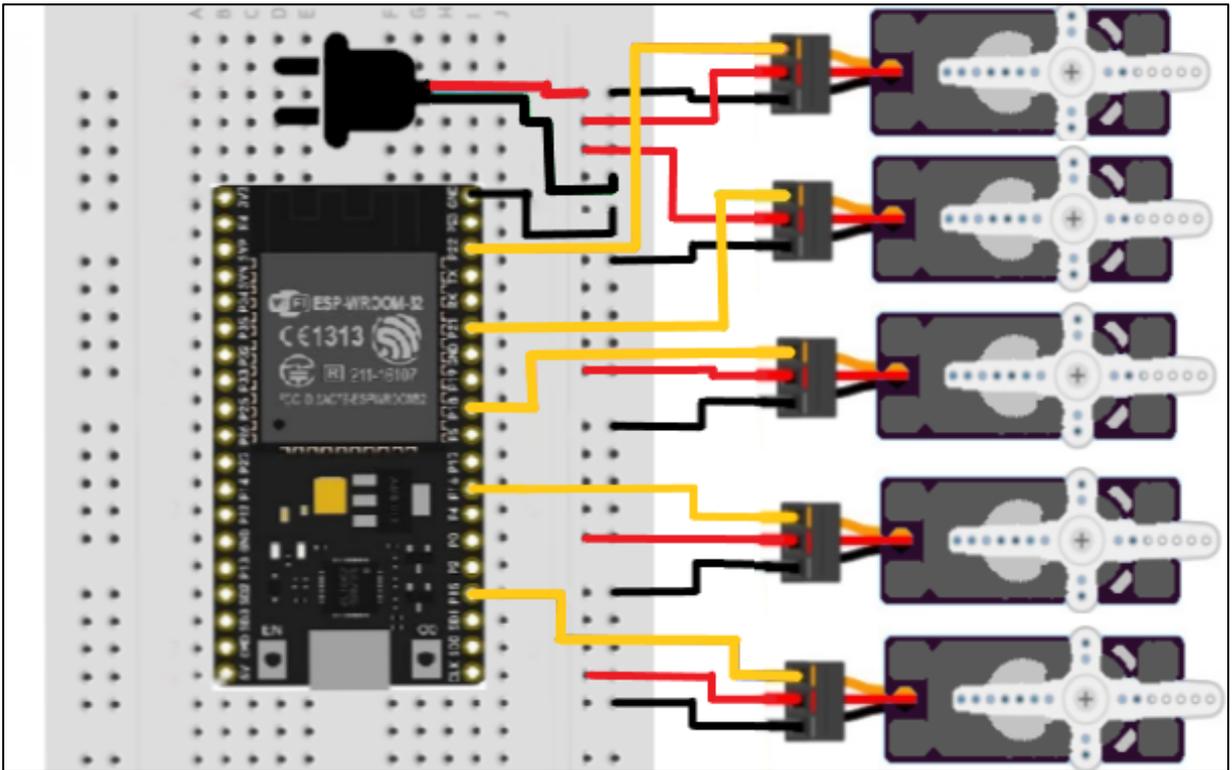
Os resultados obtidos foram que os cinco servomotores estavam operantes, sendo três com angulatura de 5° à 180° e dois com angulatura de 0° à 180°.

4.2 Conexão dos cinco servomotores

O objetivo desse teste foi fazer a integração dos cinco servomotores e verificar o seu funcionamento. A princípio todos os servomotores foram alimentados diretamente pelo microcontrolador ESP32, o que ocasionou trepidações nos servomotores e uma queda de alimentação no ESP32, acionando assim o *Brown Out Detector* (BOD), que é um hardware interno ou externo que detecta problemas na alimentação de um sistema, gerando assim uma interrupção na *Central Processing Unit* (CPU) (MORAIS, 2020).

A solução adotada foi a adição de uma fonte externa de 5V, 750mA ao circuito, resultando na eliminação das trepidações e a queda de alimentação no ESP32. A Figura 17 mostra o diagrama de ligação do circuito utilizado para o teste com a fonte externa. O código fonte utilizado se encontra no Apêndice A - Código de conexão dos cinco servomotores.

Figura 17 - Diagrama do circuito de teste com os cinco servomotores.



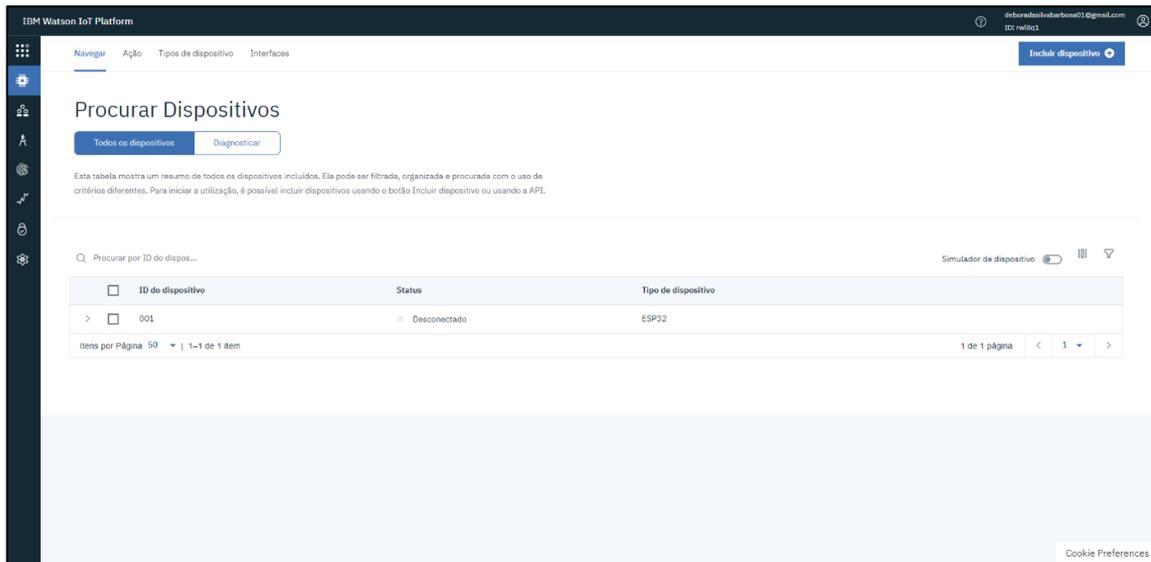
Fonte: Autoria própria.

4.3 Conexão ESP32 à Plataforma IBM Watson IoT

O objetivo desse teste foi a conexão do microcontrolador ESP32 à Plataforma IBM Watson IoT através do protocolo de comunicação MQTT, sendo que a princípio foi necessário a criação de um usuário no IBM Cloud. Após a criação do usuário, foi feito a contratação do serviço gratuito Plataforma IBM Watson IoT.

Na plataforma foi criado um tipo de dispositivo e seu identificador, denominados respectivamente, ESP32 e 001. A Figura 18 apresenta a página da Plataforma IBM Watson IoT com o dispositivo criado e desconectado.

Figura 18 - Página da Plataforma IBM Watson IoT.



Fonte: Autoria própria.

Com a criação de um dispositivo, a plataforma gera as suas credenciais, que são os dados utilizados na conexão do microcontrolador com a plataforma. O código com as credenciais e a função de conexão do ESP32 com a plataforma através do protocolo MQTT se encontram no Apêndice B - Código de conexão do ESP32 à Plataforma IBM Watson IoT.

A Figura 19 apresenta o status da conexão no monitor serial do Arduino IDE, o qual informa que a conexão foi estabelecida e a Figura 20 apresenta o status da conexão do ESP32 à Plataforma IBM Watson IoT a qual mostra que a conexão foi estabelecida e o status foi alterado para conectado.

Figura 19 - Status da conexão no monitor serial.



Fonte: Autoria própria.

Figura 20 - Status da conexão na Plataforma IBM Watson IoT.

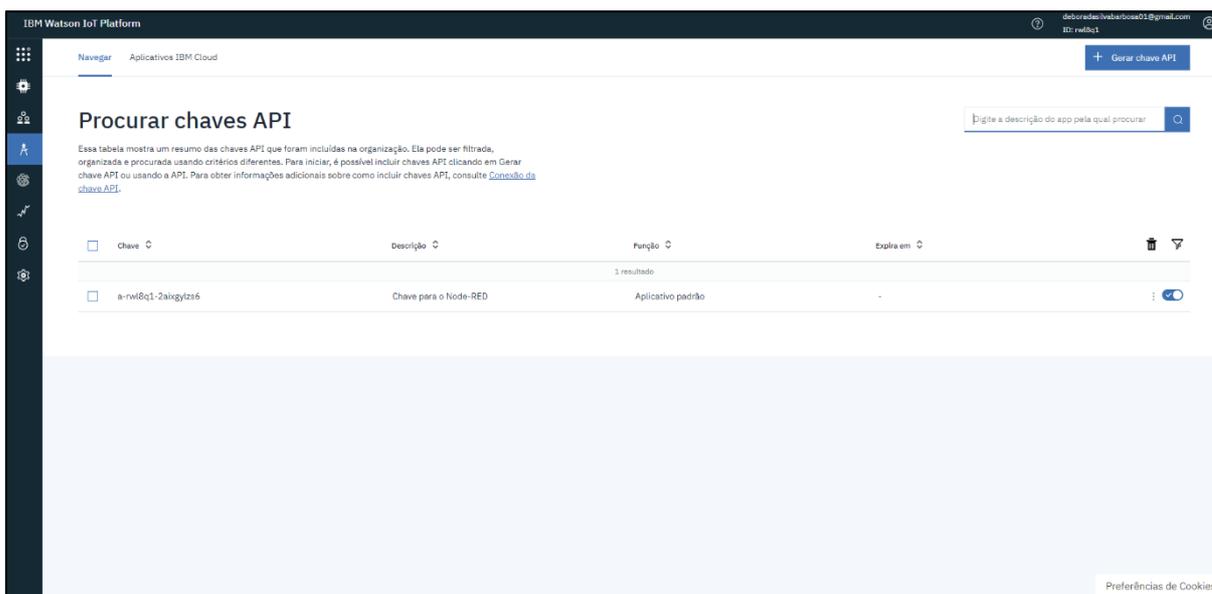


Fonte: Autoria própria.

4.4 Conexão Node-RED à Plataforma IBM Watson IoT

Este teste teve o objetivo de conectar a ferramenta de programação visual Node-RED à Plataforma IBM Watson IoT utilizando o protocolo de comunicação MQTT. Para a execução desse teste foi criada uma chave API na Plataforma IBM Watson IoT, sendo esta única e não reutilizável para outras APIs. A Figura 21 mostra a chave API criada e pronta para uso.

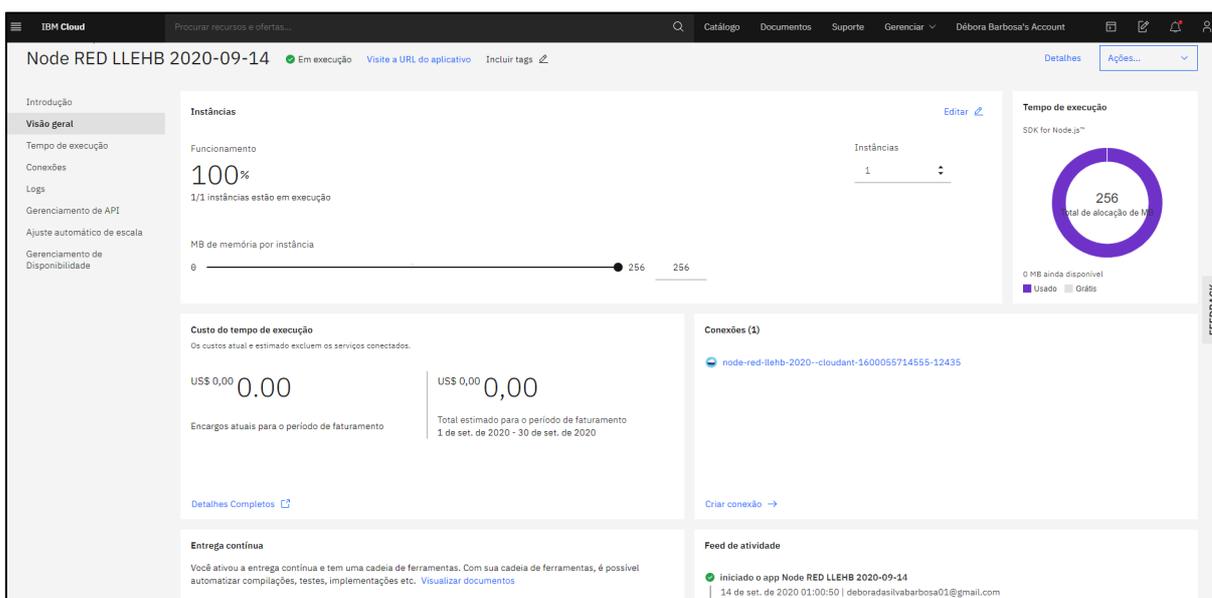
Figura 21 - Chave API criada na Plataforma IBM Watson IoT.



Fonte: Autoria própria.

Após a criação da chave API, foi criado um aplicativo Node-RED no IBM Cloud, o qual faz conexão ao serviço Node-RED, possibilitando o acesso para a criação da página WEB na ferramenta através de uma URL gerada pelo aplicativo. A Figura 22 mostra a tela de visão geral do aplicativo criado, onde observa-se o status “Em execução” e ao lado a URL de acesso do aplicativo.

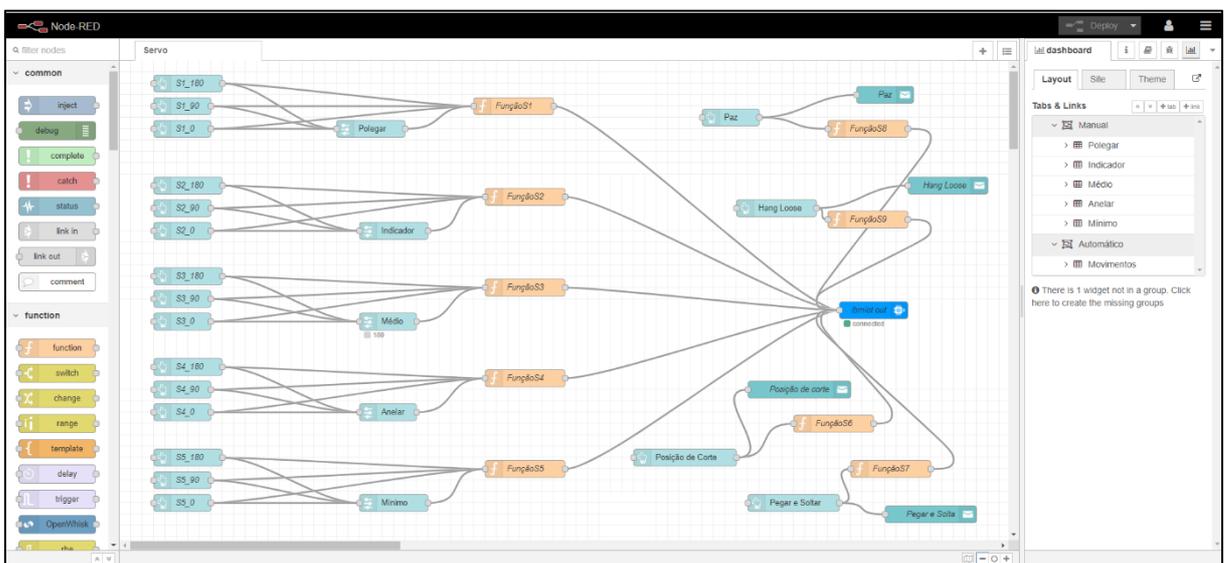
Figura 22 - Visão geral do aplicativo criado no IBM Cloud.



Fonte: Autoria própria.

Ao acessar a URL gerada, foi necessário a criação de um usuário para o aplicativo Node-RED, após a definição do usuário e senha obteve-se acesso ao editor do Node-RED integrado ao IBM Watson, possibilitando a construção da interface da página WEB através de nós. Para a conexão do Node-RED com a Plataforma IBM Watson IoT ser finalizada, foi necessário a inserção da chave API no nó “ibmiot out” e o status “connected” aparecer abaixo do mesmo, como mostrado na Figura 23.

Figura 23 - Conexão do Node-RED à Plataforma IBM Watson IoT estabelecida.



Fonte: Autoria própria.

As Figuras 24 e 25 mostram a interface gráfica da página WEB em funcionamento, a qual pode ser acessada através do link: <https://node-red-llehb-2020-09-14.mybluemix.net/ui/#!/0?socketid=tP2YJXdLWmTOWOV9AAAJ>

Figura 24 - Interface gráfica, modo manual.



Fonte: Autoria própria.

Figura 25 - Interface gráfica, modo automático.



Fonte: Autoria própria.

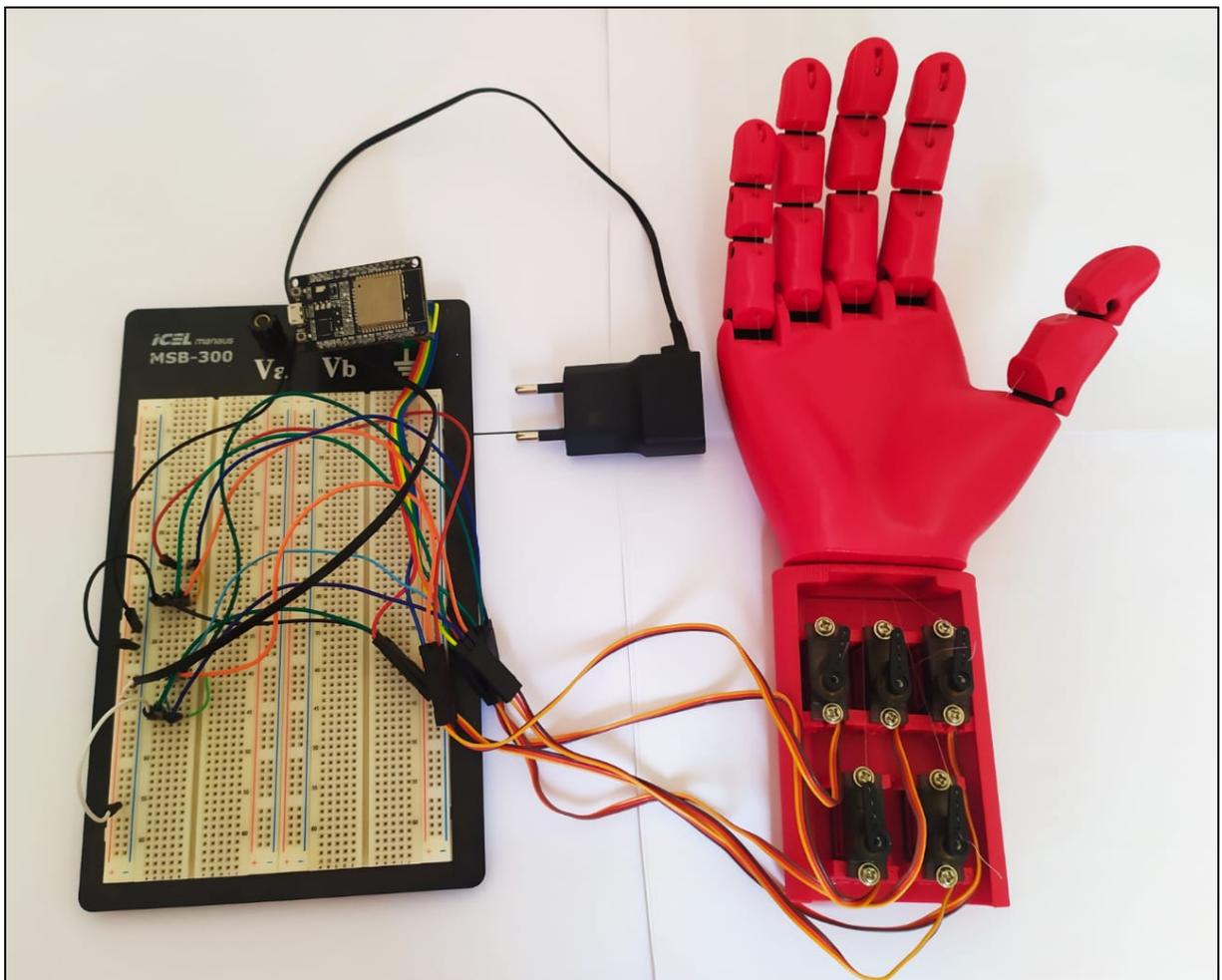
5. RESULTADOS

Este capítulo apresenta os resultados deste trabalho.

5.1 Integração dos componentes

Foi realizado a união da mão mecânica com a base e a conexão dos dedos aos servomotores através dos fios de nylon. Essa conexão é o que permite a movimentação dos dedos em resposta aos comandos enviados pelo IBM Watson IoT ao microcontrolador ESP32. A Figura 26 apresenta todos os componentes de hardware necessários para a montagem do protótipo.

Figura 26 – Componentes de hardware do protótipo.

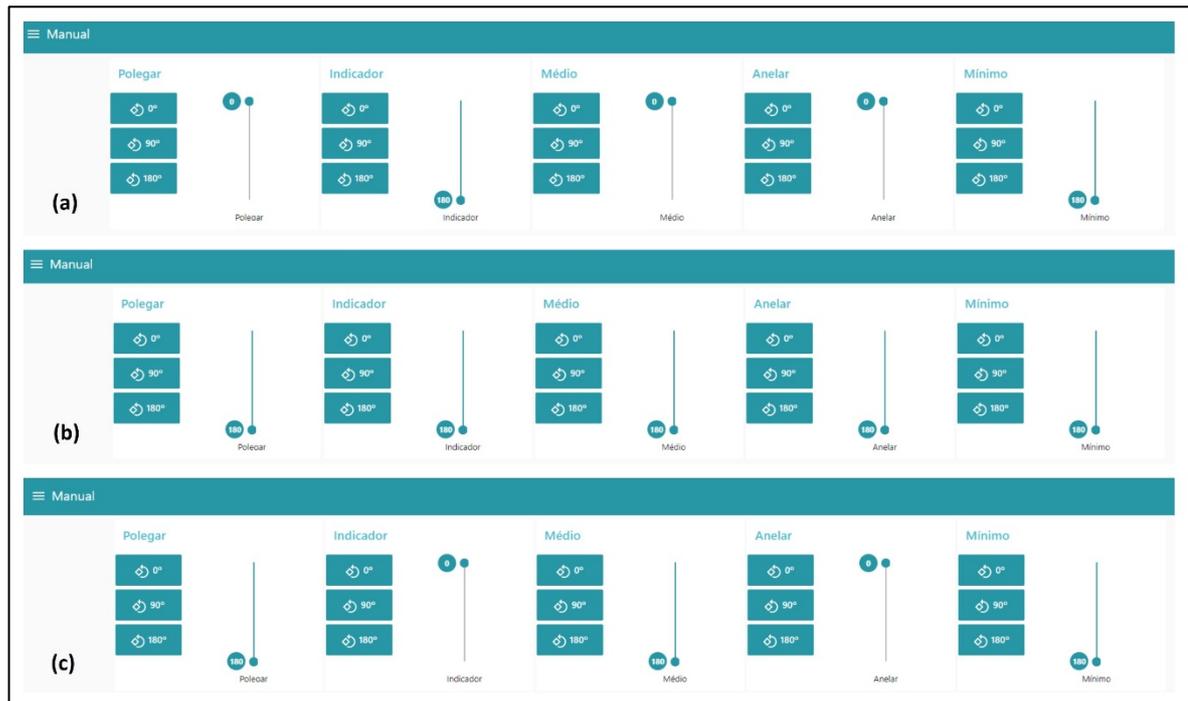


Fonte: Autoria própria.

Os primeiros testes realizados se mostraram satisfatórios, sendo os mesmos realizados utilizando o modo manual disponibilizado na página WEB. A mão mecânica apresentou uma

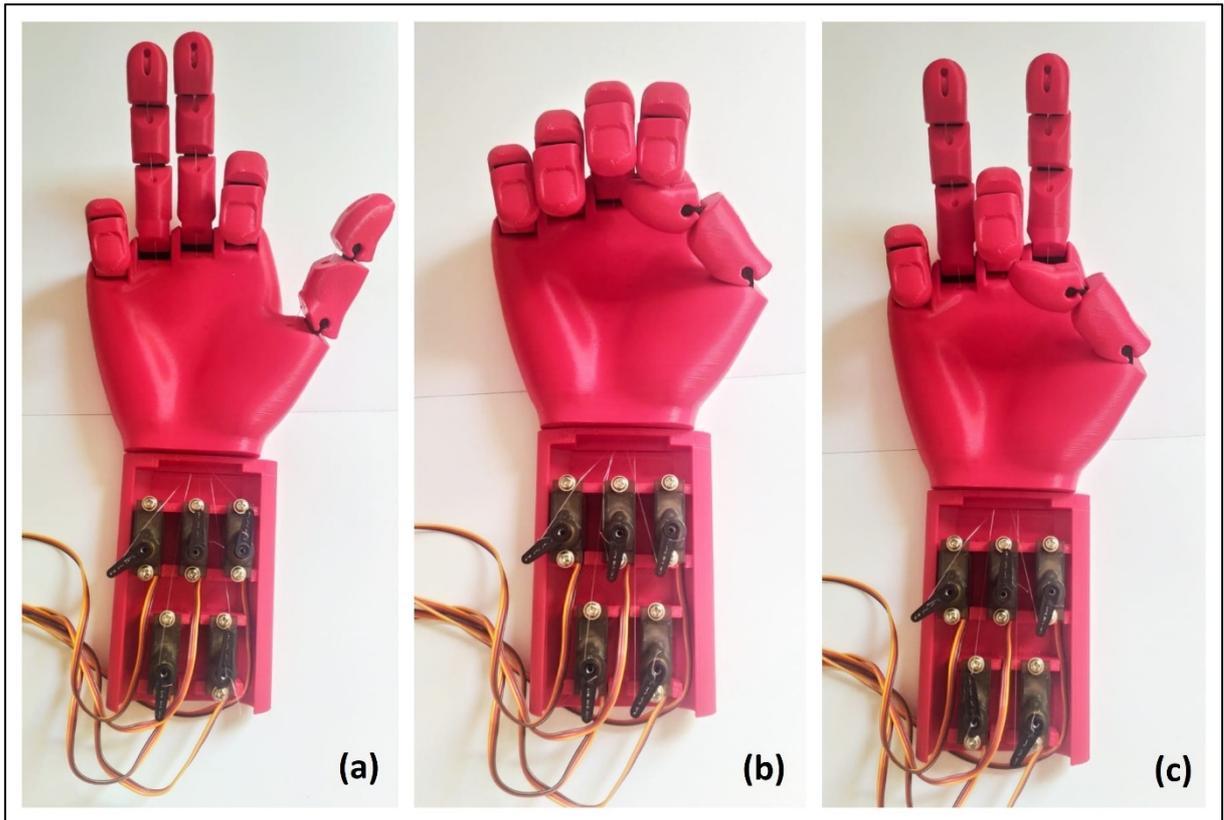
resposta rápida, de 2 a 5 segundos e apresentou-se eficiente, pois cada movimento desejado foi executado. As Figuras 27 e 28 apresentam as configurações dos movimentos: (a), (b) e (c) realizados no modo manual da página WEB, e a representação dos movimentos na mão mecânica, respectivamente.

Figura 27- Configurações dos movimentos realizados no modo manual da página WEB.



Fonte: Autoria própria.

Figura 28- Movimentos realizados no modo manual.



Fonte: Autoria própria.

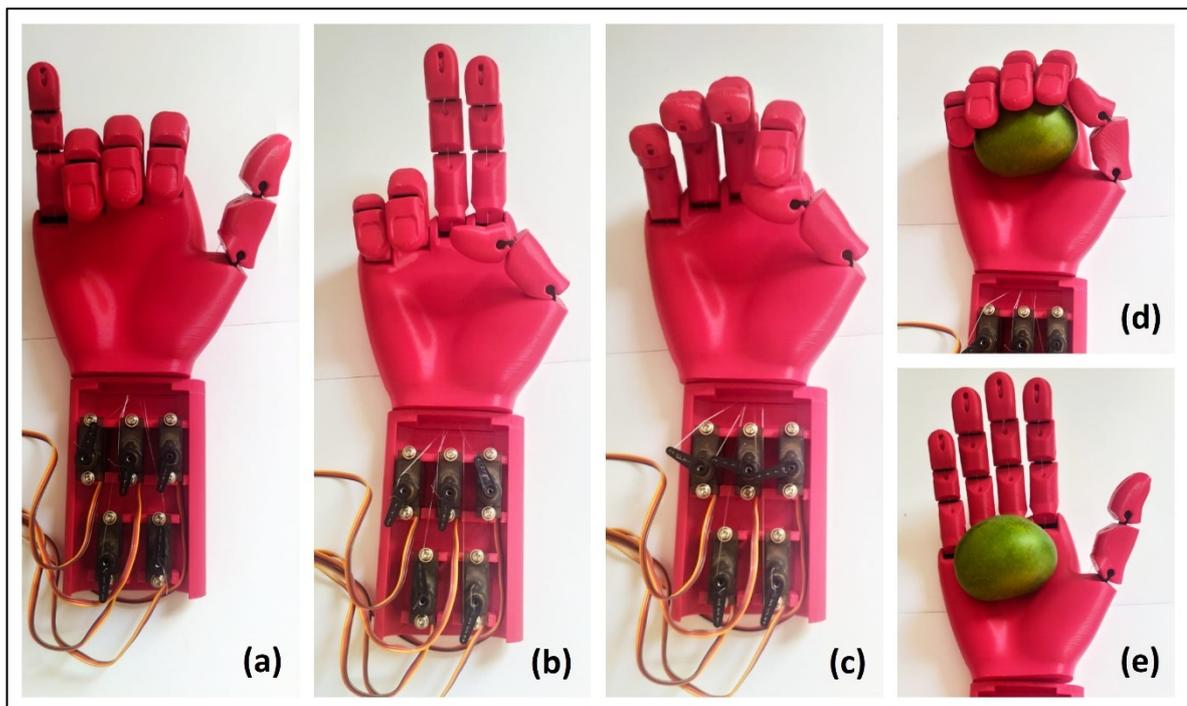
Os testes realizados no modo automático obtiveram trepidações em consequência ao torque oferecido pelos servomotores MG90s. Na movimentação pegar e soltar os testes se mostraram insatisfatórios em decorrência de trepidações, nas demais movimentações as trepidações foram poucas e não afetaram os resultados esperados. As Figuras 29 e 30 apresentam as configurações dos movimentos: (a) *hang loose*; (b) paz; (c) posição de corte; (d) e (e) pegar e soltar, realizados no modo automático da página WEB, e a representação dos movimentos na mão mecânica, respectivamente.

Figura 29 - Movimentos do modo automático da página WEB.



Fonte: Autoria própria.

Figura 30 - Movimentos do modo automático.



Fonte: Autoria própria.

A Tabela 2 apresenta os valores gastos em componentes de hardware para a fabricação da mão mecânica. Os componentes de software são plataformas e ferramentas gratuitas.

Tabela 2 - Valores gastos em componentes de hardware.

| COMPONENTES | VALOR UNITÁRIO |
|--------------------------------|-----------------------|
| Fio de nylon | R\$ 4,00 |
| Mão mecânica com a base | R\$ 560,00 |
| Servo motor MG90s (5 unidades) | R\$ 24,21 |
| Microcontrolador ES32 | R\$ 55,00 |
| VATOR TOTAL | R\$ 740,05 |

Fonte: Autoria própria.

O código utilizado na programação do microcontrolador ESP32 se encontra no Apêndice C – Código final ESP32 para mão mecânica, e o código JSON da ferramenta Node-RED está disponível em: <https://github.com/DeboraBarbosa01/C-digo-Node-RED-M-o-Mec-nica.git>.

6. CONSIDERAÇÕES FINAIS

A internet das coisas aliada com a robótica proporciona um universo vasto a ser estudado e implementado. Mãos mecânicas podem auxiliar nos processos de cirurgias, indústrias, fábricas, dentre outras áreas, com a intenção de facilitar o manuseio de ferramentas e proteger o ser humano de materiais que sejam prejudiciais à saúde.

A IoT pode permitir a realização de movimentos de uma mão mecânica, de qualquer lugar do mundo que possua conexão à internet, facilitando a realização de procedimentos a longa distância.

A proposta deste trabalho foi alcançada, com a construção do protótipo de uma mão mecânica, confeccionada em impressora 3D e controlada através de uma página WEB, por meio de *smartphones* e *notebooks* utilizando os conceitos de IoT.

Para a criação da página WEB, a ferramenta Node-RED se mostrou satisfatória, contudo, conclui-se que esta ferramenta geralmente é mais útil para realizar a visualização de dados enviados de algum dispositivo de controle.

Os resultados obtidos pelo microcontrolador ESP32, em relação ao controle dos servomotores, também se mostraram satisfatórios em decorrência da disponibilidade da conexão *wireless* e velocidade de execução dos comandos.

Em geral, os resultados obtidos mostraram que a implementação de uma mão mecânica, utilizando o conceito de IoT e a plataforma em nuvem da IBM é realizável, uma vez que o protótipo da mão mecânica conseguiu replicar alguns movimentos de uma mão humana.

6.1 Trabalhos Futuros

Sugere-se para trabalhos futuros, as seguintes melhorias:

- Substituição dos servomotores MG90s para servomotores MG995, com o objetivo de eliminar as trepidações e fornecer força para os dedos da mão mecânica;
- Utilização das execuções paralelas e tratamento de interrupções em tempo real, disponibilizados pelo microcontrolador ESP32, possibilitando uma maior aproximação da movimentação de uma mão humana;
- Inserção da inteligência artificial ao controle automático da mão mecânica, tornando possível a tomada de decisões assertivas em situações específicas, como cirurgias e manuseio de toxinas;

- Implementação de uma luva com sensores e a introdução de uma camada de silicone nas pontas dos dedos da mão mecânica, visando facilitar a manipulação de objetos;
- Integração de mais serviços disponibilizados pelo IBM Watson;
- Acoplamento da mão mecânica em um manipulador robótico.

REFERÊNCIAS

AGRAWAL, S.; VIEIRA, D. A survey on internet of things. **Abakós**, v. 1, n. 2, p. 78–95, 2013.

BASÍLIO, Shirley. **Conhecendo o protocolo MQTT**. Disponível em: <<https://blogmasterwalkershop.com.br/outros/conhecendo-o-protocolo-mqtt/#:~:text=O%20princ%C3%ADpio%20de%20funcionamento%20do,para%20clientes%20de%20destino%20relevante>>. Acesso em: 07 nov. 2020.

BASÍLIO, Shirley. **O que é Node-RED? Conhecendo e instalando**. Disponível em: <<https://blogmasterwalkershop.com.br/outros/o-que-e-node-red-conhecendo-e-instalando/>>. Acesso em: 07 nov. 2019.

BORGES, Eduardo Augusto de Oliveira. **SISTEMA DE AUTOMAÇÃO RESIDENCIAL UTILIZANDO A TECNOLOGIA ESP32 COMO ALTERNATIVA DE BAIXO CUSTO**. 2019. 59 f. TCC (Graduação) - Curso de Engenharia de Computação, O Instituto Federal de Minas Gerais, Bambuí, 2019. Disponível em: <<https://repositorio.bambui.ifmg.edu.br/index.php/ec/article/view/77>>. Acesso em: 08 nov. 2019.

BLACKSTOCK, Michael; LEA, Rodger. Toward a Distributed Data Flow Platform for the Web of Things. In: IEEE IOT CONFERENCE, 5., 2014, Cambridge. **Anais....** Cambridge: Acm, 2014. p. 1 - 7. Disponível em: <<https://dl.acm.org/citation.cfm?id=2684432>>. Acesso em: 07 nov. 2019.

CANALTECH. **IBM**. Disponível em: <<https://canaltech.com.br/empresa/ibm/>>. Acesso em: 22 set. 2019.

COLOMBO, Jamires Fátima; LUCCA FILHO, João de. INTERNET DAS COISAS (IOT) E INDÚSTRIA 4.0. **Revista Interface Tecnológica**, [s.l.], v. 15, n. 2, p.72-85, 30 dez. 2018. *Interface Tecnológica*. <<http://dx.doi.org/10.31510/infra.v15i2.496>>. Acesso em: 05 out. 2019.

CTI GLOBAL. **Watson**. Disponível em: <<http://www.ctiglobal.com/watson/>>. Acesso em: 14 set. 2019.

ESPRESSIF. **ESP-WROOM-32 Series: Datasheet**. 3. ed. Xangai: Espressif System, 2018. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Acesso em: 07 nov. 2019.

FRAGA, Jorge Gerardo Pereira. Actualidad de la cirugía robótica. **Revista Cubana de Cirugía**, Habana, v. 56, n. 1, p. 1-1, mar. 2017. Disponível em: <http://scielo.sld.cu/scielo.php?pid=S0034-74932017000100006&script=sci_arttext&tlng=pt>. Acesso em: 06 set. 2020.

GALINDO, Johanne Monteiro Correia; PEREIRA, Diego de Camargo Minelli. **Mão Biônica**. 2018. Disponível em: <https://www.researchgate.net/publication/322582081_Mao_Bionica>. Acesso em: 27 mar. 2020.

GERHARDT, Tatiana Engel; SILVEIRA, Denise Tolfo (Org.). **Métodos de Pesquisa**. Rio Grande do Sul: UFRGS, 2009. 113 p.

GIL, Antonio Carlos. **Como Elaborar Projetos de Pesquisa**. 4. ed. São Paulo: Atlas S.A., 2002.

GUATELLI, Giovanna Prado; MENDES, Alessandro Corrêa; LIMA, Fernanda Pupio Silva; LIMA, Mário Oliveira. DESENVOLVIMENTO DE UMA PRÓTESE ROBÓTICA MICROPROCESSADA. **Inicepg**, São José dos Campos, v. 1, n. 1, p.1-6, ago. 2018. Disponível em:<http://www.inicepg.univap.br/cd/INIC_2018/anais/arquivos/RE_0297_0362_01.pdf>. Acesso em: 28 mar. 2020.

GYROBOT. **Flexy-Hand**. 2014. Disponível em:<<https://www.thingiverse.com/thing:242639>>. Acesso em: 24 jun. 2020.

GYROBOT LIMITED (Reino Unido). **Impressão 3D**. Disponível em:<<http://www.gyrobot.co.uk/3d-printing.html>>. Acesso em: 27 set. 2020.

IBM. **Produtos IBM Cloud**. Disponível em:<<https://www.ibm.com/cloud/products#936881>>. Acesso em: 22 set. 2019a.

Product overview. Disponível em:<<https://www.ibm.com/support/knowledgecenter/SSQP8H/iot/overview/overview.html>>. Acesso em: 29 set. 2019b.

Serviço IBM Watson IoT Platform. Disponível em:<https://www.ibm.com/support/knowledgecenter/SSQP8H/iot/developing/iot_platform.html>. Acesso em: 14 set. 2019c.

JALALI, Fatemeh et al. **Interconnecting Fog Computing and Microgrids for Greening IoT**. Melbourne, Australia: Ibm Research, 2016. 6 p. Disponível em:<<https://ieeexplore.ieee.org/abstract/document/7796469/>>. Acesso em: 28 out. 2019.

KLEINA, Nilton. **A história da IBM, uma empresa centenária e revolucionária**. 2017. Disponível em:<<https://www.tecmundo.com.br/mercado/121588-historia-ibm-empresa-centenaria-revolucionaria-video.htm>>. Acesso em: 20 set. 2019.

LANGEVIN, Gael. **InMoov Open Source 3D Painted Life-Size Robot**. Disponível em:<<https://inmoov.fr/gallery-v2/>>. Acesso em: 27 set. 2020.

LUCAS, Iuri Damasceno; SILVA, José Rodolfo Clementino da. **A UTILIZAÇÃO DA TECNOLOGIA DA INFORMAÇÃO NAS ORGANIZAÇÕES: O uso da Internet das Coisas (IoT)**. 2018. 35 f. Tese (Doutorado) - Curso de Administração, Faculdade Cesmac do Sertão, Palmeira dos Índios, 2018. Disponível em:<<https://ri.cesmac.edu.br/bitstream/tede/323/1/A%20utiliza%C3%A7%C3%A3o%20da%20tecnologia%20da%20informa%C3%A7%C3%A3o%20nas%20organiza%C3%A7%C3%B5es%20o%20uso%20da%20internet%20das%20coisas%20%28iot%29.pdf>>. Acesso em: 05 set. 2019.

MASSOLA, Silze Cristina; PINTO, Giuliano Scombatti. O USO DA INTERNET DAS COISAS (IOT) A FAVOR DA SAÚDE. **Revista Interface Tecnológica**, Taquaritinga, v. 15, n. 2, p. 124-137, 30 dez. 2018. Interface Tecnológica. Disponível em:<<http://dx.doi.org/10.31510/inf.v15i2.515/>>. Acesso em: 01 set. 2020.

MAGRANI, Eduardo. **A internet das coisas**. Rio de Janeiro: FGV, 2018. 192 p.

MANZONI JUNIOR, Ralphe (ed.). **O Watson, da IBM, fica mais brasileiro**. 2019. Disponível em:<<https://neofeed.com.br/blog/home/o-watson-da-ibm-fica-mais-brasileiro/>>. Acesso em: 25 set. 2020.

MIRA, José Eugênio de. **DESENVOLVIMENTO DE UM MODELO DE NEURÔNIO ARTIFICIAL COM MICROCONTROLADOR WI-FI AUTÔNOMO COM COMUNICAÇÃO M2M PARA IOT**. 2019. 72 f. Monografia (Especialização) – Programa de Pós-Graduação em Mídia e Tecnologia, Faculdade de Arquitetura, Artes e Comunicação – FAAC, Bauru, 2019. Disponível em:<https://repositorio.unesp.br/bitstream/handle/11449/181226/mira_je_me_bauru.pdf?sequence=5>. Acesso em: 30 ago. 2019.

MORAIS, José. **ESP32 – Utilizando o Brown Out Detector (BOD) para quedas de alimentação**. Disponível em:<<https://portal.vidadesilicio.com.br/esp32-utilizando-o-brown-out-detector-bod/>>. Acesso em: 26 set. 2020.

MQTT. **MQTT: o padrão para mensagens IoT**. Disponível em:<<https://mqtt.org/>>. Acesso em: 07 nov. 2020.

NEDELTCHEVA, Galia Novakova; SHOIKOVA, Elena. **Models for Innovative IoT Ecosystems**. Bulgaria: Sofia University, 2017. Disponível em:<https://www.researchgate.net/profile/Elena_Shoikova2/publication/320623589_Models_for_Innovative_IoT_Ecosystems/links/5a58d3cc0f7e9b5fb383f443/Models-for-Innovative-IoT-Ecosystems.pdf>. Acesso em: 20 ago. 2019.

OLIVEIRA, Neto. **Por que Node-RED para IOT?** 2018. Disponível em:<<https://medium.com/@netoolii/por-que-node-red-para-iot-41a4ab170c56>>. Acesso em: 07 nov. 2019.

OLIVEIRA, Por Sérgio de. **Internet das Coisas com ESP8266, Arduino e Raspberry Pi**. 1. ed. São Paulo: Novatec, 2017. 235 p.

PINOCHET, Luis. **Tecnologia da informação e comunicação**. Rio de Janeiro: Elsevier, 2014.

PINTO, Otávio Henrique Gotardo. **AUTOMAÇÃO RESIDENCIAL UTILIZANDO A PLATAFORMA EM NUVEM IBM BLUEMIX**. 2017. 65 f. TCC (Graduação) - Curso de Engenharia Elétrica, Universidade de São Paulo, São Carlos, 2017.

RAJ, Pethuru; RAMAN, Anupama C.. **The Internet of Things: Enabling Technologies, Platforms, and Use Cases**. New York: Crc Press, 2017. 393 p.

RIBEIRO, Deivyson Bruno Silva. **Mão biônica controlada por luva de sensores**. 2018. 39 f. TCC (Graduação) - Curso de Engenharia da Computação, Universidade Federal de Ouro Preto, João Monlevade, 2018.

SILVA, Luiz Augusto Zillmann da. **Controle de Prótese Mecânica Aplicado à Robótica Assistiva através de Eletromiografia**. 2016. 41 f. TCC (Graduação) - Curso de Engenharia Elétrica, Universidade Federal de Juiz de Fora, Juiz de Fora, 2016.

XAVIER, Ricardo Taoni. **IMPLEMENTAÇÃO DE UMA PRÓTESE ATIVA PARA MEMBRO SUPERIOR DE BAIXO CUSTO**. 2016. 118 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Unesp, Ilha Solteira, 2016. Disponível em:<https://repositorio.unesp.br/bitstream/handle/11449/144525/xavier_rt_me_ilha.pdf?sequence=3>. Acesso em: 28 mar. 2020

WAZLAWICK, Raul Sidnei. **Metodologia de Pesquisa para Ciência da Computação**. 1. ed. Rio de Janeiro: Elsevier, 2014.

APÊNDICE A - Código de conexão dos cinco servomotores

```
#include <Servo.h>

static const int servosPins[5] = {14, 27, 23, 19, 22};

Servo servos[5];

void setServos(int degrees) {
    for(int i = 0; i < 5; ++i) {
        servos[i].write((degrees + (35 * i)) % 180);
    }
}

void setup() {
    Serial.begin(115200);

    for(int i = 0; i < 5; ++i) {
        if(!servos[i].attach(servosPins[i])) {

        }
    }
}

void loop() {
    for(int posDegrees = 0; posDegrees <= 170; posDegrees++) {
        setServos(posDegrees);
        Serial.println(posDegrees);
        delay(20);
    }

    for(int posDegrees = 170; posDegrees >= 0; posDegrees--) {
        setServos(posDegrees);
        Serial.println(posDegrees);
        delay(20);
    }
}
```

APÊNDICE B - Código de conexão do ESP32 à Plataforma IBM Watson IoT

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <DNSServer.h>
#include <WebServer.h>
#include <WiFiManager.h>
#include <Servo.h>

const char* ssid = "20ever";
const char* password = "17752010";
const char* host = "data.sparkfun.com";

//Credenciais Plataforma IBM Watson IoT
const String ORG = "rwl8q1";
const String DEVICE_TYPE = "ESP32";
const String DEVICE_ID = "001";
#define DEVICE_TOKEN "mkDD1WHESeyqXQPFNy"

//ID que usada para conectar
const String CLIENT_ID = "d:"+ORG+": "+DEVICE_TYPE+": "+DEVICE_ID;

//Server MQTT utilizada
const String MQTT_SERVER = ORG + ".messaging.internetofthings.ibmcloud.com";

WiFiClient wifiClient;

PubSubClient client(MQTT_SERVER.c_str(), 80, wifiClient);

void setup()
{
  Serial.begin(115200);

  Serial.println();
  Serial.println();
  Serial.print("Conectando a ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi conectado!");
  Serial.println(WiFi.localIP());
```

```

    connectMQTTServer();
}
void loop()
{
    client.loop();

    WiFiClient client;
    const int httpPort = 80;
    if (!client.connect(host, httpPort)) {
        Serial.println("conexão falhou");
        return;
    }
    Serial.println();
}

//Função responsável por conectar ao server MQTT
void connectMQTTServer() {
    Serial.println("Conectando ao MQTT Server...");
    //Se conecta com as credenciais obtidas na Plataforma
    if (client.connect(CLIENT_ID.c_str(), "use-token-auth", DEVICE_TOKEN)) {
        //Se a conexão foi bem sucedida
        Serial.println("Conexão a Plataforma estabelecida!");
    }
    else {
        //Se ocorreu algum erro
        Serial.print("error = ");
        Serial.println(client.state());
        connectMQTTServer(); //tenta conectar novamente
    }
}
}

```

APÊNDICE C - Código final ESP32 para mão mecânica

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <DNSServer.h>
#include <WebServer.h>
#include <WiFiManager.h>
#include <Servo.h>

const char* ssid = "20ever";
const char* password = "17752010";
const char* host = "data.sparkfun.com";

//no site do Watson IoT
const String ORG = "rw18q1";
const String DEVICE_TYPE = "ESP32";
const String DEVICE_ID = "001";
#define DEVICE_TOKEN "mkDD1WHESeyqXQPFNy"

//ID que usaremos para conectar
const String CLIENT_ID = "d:"+ORG+": "+DEVICE_TYPE+": "+DEVICE_ID;

//Server MQTT que iremos utilizar
const String MQTT_SERVER = ORG + ".messaging.internetofthings.ibmcloud.com";

//Comandos para o Servo 1
#define COMMAND_TOPIC_S1 "iot-2/cmd/commandR1/fmt/json"

//Comandos para o Servo 2
#define COMMAND_TOPIC_S2 "iot-2/cmd/commandR2/fmt/json"

//Comandos para o Servo 3
#define COMMAND_TOPIC_S3 "iot-2/cmd/commandR3/fmt/json"

//Comandos para o Servo 4
#define COMMAND_TOPIC_S4 "iot-2/cmd/commandR4/fmt/json"

//Comandos para o Servo 5
#define COMMAND_TOPIC_S5 "iot-2/cmd/commandR5/fmt/json"

//Comandos para botão posição de corte
#define COMMAND_TOPIC_S6 "iot-2/cmd/commandR6/fmt/json"

//Comandos para botão pegar e soltar
#define COMMAND_TOPIC_S7 "iot-2/cmd/commandR7/fmt/json"

//Comandos para botão posição de paz
#define COMMAND_TOPIC_S8 "iot-2/cmd/commandR8/fmt/json"
```

```

//Comandos para botão posição rock
#define COMMAND_TOPIC_S9 "iot-2/cmd/commandR9/fmt/json"

//Pinos dos Servos
static const int servosPins[5] = {22, 14, 27, 23, 19};

//Servo servo1,servo2,servo3,servo4,servo5;
Servo servos[5];

//Cliente WiFi que o MQTT irá utilizar para se conectar
WiFiClient wifiClient;

PubSubClient client(MQTT_SERVER.c_str(), 80, wifiClient);

void setServos(int degrees) {
  for(int i = 0; i < 5; ++i) {
    servos[i].write(degrees);
  }
}

void setup()
{
  Serial.begin(115200);

  for(int i = 0; i < 5; ++i) {
    if(!servos[i].attach(servosPins[i])) {

    }
  }

  // Serial.println();
  // Serial.println();
  // Serial.print("Conectando a ");
  // Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    // Serial.print(".");
  }

  // Serial.println("");
  // Serial.println("WiFi conectado!");

```

```

// Serial.println("Endereço IP: ");
// Serial.println(WiFi.localIP());

    connectMQTTServer();
}

void loop()
{
    client.loop();

    WiFiClient client;
    const int httpPort = 80;
    if (!client.connect(host, httpPort)) {
        // Serial.println("conexão falhou");
        return;
    }

    // Serial.println();
}

//Função responsável por conectar ao server MQTT
void connectMQTTServer() {
    // Serial.println("Connecting to MQTT Server...");
    //Se conecta com as credenciais obtidas no site do Watson IoT
    //quando cadastramos um novo device
    if (client.connect(CLIENT_ID.c_str(), "use-token-auth", DEVICE_TOKEN)) {
        //Se a conexão foi bem sucedida
        // Serial.println("Connected to MQTT Broker");
        //Quando algo for postado em algum tópico que estamos inscritos
        //a função "callback" será executada
        client.setCallback(callback1);

        //Se inscreve nos tópicos de interesse
        client.subscribe(COMMAND_TOPIC_S1);
        client.subscribe(COMMAND_TOPIC_S2);
        client.subscribe(COMMAND_TOPIC_S3);
        client.subscribe(COMMAND_TOPIC_S4);
        client.subscribe(COMMAND_TOPIC_S5);
        client.subscribe(COMMAND_TOPIC_S6);
        client.subscribe(COMMAND_TOPIC_S7);
        client.subscribe(COMMAND_TOPIC_S8);
        client.subscribe(COMMAND_TOPIC_S9);
    }

    else {

```

```

    //Se ocorreu algum erro
    // Serial.print("error = ");
    // Serial.println(client.state());
    connectMQTTServer(); //tenta conectar novamente
}

}

void callback1(char* topic, byte* payload, unsigned int length) {

    String string;
    // Serial.print("topic ");
    // Serial.println(topic);
    for (int i = 0; i < length; i++) {
        string+=((char)payload[i]);
    }

    if (strcmp(topic, COMMAND_TOPIC_S1) == 0){
        // Serial.print(" ");
        int resultado = string.toInt();
        int pos = map(resultado, 0,180,0, 170);
        servos[1].write(pos);
    }
    else if (strcmp(topic, COMMAND_TOPIC_S2) == 0){
        //Serial.print(" ");
        int resultado = string.toInt();
        //int pos = map(resultado, 0,180,0, 170);
        servos[2].write(resultado);
    }

    else if (strcmp(topic, COMMAND_TOPIC_S3) == 0){
        // Serial.print(" ");
        int resultado = string.toInt();
        //int pos = map(resultado, 0,180,0, 170);
        servos[3].write(resultado);
    }

    else if (strcmp(topic, COMMAND_TOPIC_S4) == 0){
        // Serial.print(" ");
        int resultado = string.toInt();
        servos[4].write(resultado);
    }

    else if (strcmp(topic, COMMAND_TOPIC_S5) == 0){
        // Serial.print(" ");
        int resultado = string.toInt();
        int pos = map(resultado, 0,180,0, 150);
        servos[0].write(pos);
    }
}

```

```

//Modo Manual
else if (strcmp(topic, COMMAND_TOPIC_S6) == 0) {

    for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {
        setServos(posDegrees);
        Serial.println(posDegrees);
    }

    delay(1000);

    servos[1].write(180);
    servos[2].write(180);
    servos[3].write(120);
    servos[4].write(80);
    servos[0].write(80);

}

else if (strcmp(topic, COMMAND_TOPIC_S7) == 0) {

    for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {
        setServos(posDegrees);
    }

    delay(1000);

    for(int posDegrees = 0; posDegrees <= 180; posDegrees++) {
        setServos(posDegrees);
    }

    delay(7000);

    for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {
        setServos(posDegrees);
    }

}

else if (strcmp(topic, COMMAND_TOPIC_S8) == 0) {

    for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {
        setServos(posDegrees);
        Serial.println(posDegrees);
    }

    delay(1000);

    servos[1].write(180);

```

```
servos[4].write(180);  
servos[0].write(180);  
  
}  
  
else {  
  
  for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {  
    setServos(posDegrees);  
  }  
  
  delay(1000);  
  
  servos[2].write(180);  
  servos[3].write(180);  
  servos[4].write(180);  
  
}  
}
```

ANEXO A – Termo de Autorização de Publicação de Produção Acadêmica



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
GABINETE DO REITOR

Av. Universitária, 1069 • Setor Universitário
Cidade Postal 98 • CEP 74025-010
Goiânia • Goiás • Brasil
Fone: (62) 3346 1000
www.pucgoias.edu.br • reitoria@pucgoias.edu.br

RESOLUÇÃO n° 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O(A) estudante Debora da Silva Barbosa
do Curso de Engenharia da Computação matrícula 20151003300340,
telefone: (62) 98279-9799 e-mail deborasilvabarbosa@hotmail.com na qualidade de titular dos
direitos autorais, em consonância com a Lei n° 9.610/98 (Lei dos Direitos do autor),
autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o
Trabalho de Conclusão de Curso intitulado
Manipulação de uma mão mecânica utilizando a plataforma
IBM Watson IoT, gratuitamente, sem ressarcimento dos direitos autorais, por 5
(cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial
de computadores, no formato especificado (Texto (PDF); Imagem (GIF ou JPEG); Som
(WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da
área; para fins de leitura e/ou impressão pela internet, a título de divulgação da
produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 10 de dezembro de 2020.

Assinatura do(s) autor(es): Debora da S. Barbosa

Nome completo do autor: Debora da Silva Barbosa

Assinatura do professor-orientador: Ludmilla R. P. dos Santos

Nome completo do professor-orientador: Ludmilla Reis Pinheiro dos Santos