



**MINISTÉRIO DA EDUCAÇÃO**

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DO AMAZONAS**

**CAMPUS MANAUS DISTRITO INDUSTRIAL**

**CURSO DE ENGENHARIA DE CONTROLE E  
AUTOMAÇÃO**



**INSTITUTO FEDERAL  
AMAZONAS**  
Campus Manaus-Distrito Industrial

**NORA NEYSE TORRES DA CUNHA**

**UM SISTEMA PARA COMUNICAÇÃO NAS DEPENDÊNCIAS DE  
INSTITUIÇÕES USANDO NODEMCU - VISION**

**MANAUS  
2020**

**NORA NEYSE TORRES DA CUNHA**

**UM SISTEMA PARA COMUNICAÇÃO NAS DEPENDÊNCIAS DE  
INSTITUIÇÕES USANDO NODEMCU - VISION**

Trabalho de Conclusão de Curso de graduação em Engenharia de Controle e Automação do Campus Manaus Distrito Industrial, Instituto Federal de Educação, Ciência e Tecnologia do Amazonas - IFAM, como requisito obrigatório para a obtenção do título de Bacharel em Engenharia de Controle e Automação.

Orientador: Prof. Dr. Vitor Bremgartner da Frota.

**MANAUS  
2020**

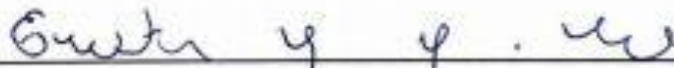
NORA NEYSE TORRES DA CUNHA

**UM SISTEMA PARA COMUNICAÇÃO NAS DEPENDÊNCIAS DE  
INSTITUIÇÕES USANDO NODEMCU - VISION**

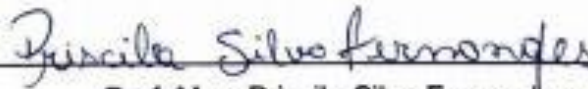
Trabalho de conclusão de curso apresentado ao curso de Engenharia de Controle e Automação do Campus Manaus Distrito Industrial do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas – IFAM, como requisito obrigatório para obtenção do grau de Bacharel em Engenharia de Controle e Automação.

Aprovado em 28 de Fevereiro de 2020.

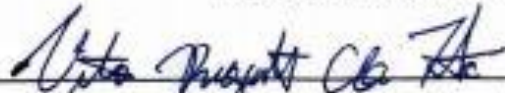
**BANCA EXAMINADORA**



Prof. Esp. Ewerton Andrey Godinho Ribeiro  
Professor avaliador



Prof. Msc. Priscila Silva Fernandes  
Professora Avaliadora



Prof. Dr. Vitor Bremgartner da Frota  
Orientador

**MANAUS  
2020**

*Aos meus pais pelo amor, apoio,  
proteção, suporte e incentivo constante.*

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, por ser essencial em minha vida, meu guia e autor do meu destino.

Aos meus pais, Jose Augusto da Cunha Junior e Maria Edyvanda Torres da Cunha por terem me dado força e sustentabilidade financeira desde o início do curso. Aproveito também a oportunidade para dizer que os dois foram os meus maiores incentivos pra nunca desistir dos meus sonhos. Obrigada Pai e Mãe.

Aos meus irmãos Thessyvanda Torres e Paulo Ramon pela assistência, parceria e oportunidades de aprendizagem.

Ao meu namorado Bruno Silva da Costa pela dedicação oferecida, o carinho, colaboração, cumplicidade e por sua capacidade de me trazer paz na correria de cada semestre.

Aos amigos, por confiarem em minha capacidade, e em todo o suporte que me proporcionaram ao longo desta jornada.

Agradeço ao professor Dr. Vitor Bremgartner pela oportunidade de me orientar na conclusão deste trabalho, ao corpo docente e a todos os servidores do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas - IFAM, pela convivência e aprendizagem durante o período do curso.

Aos meus colegas de turma pelas experiências acadêmicas compartilhadas.

Finalmente, aos meus avós paternos e maternos, que não estão presentes, mas guardo nossas lembranças como pérolas dentro do meu coração.

*“A vida é um processo fluente e em alguns lugares do caminho coisas desagradáveis ocorrerão. Podem deixar cicatrizes, mas a vida continua a fluir. É como a água fluente, que ao estagnar-se, torna-se podre; não pare! Continue bravamente... porque cada experiência nos ensina uma lição.”*

*Bruce Lee.*

## **RESUMO:**

A comunicação é fundamental para a interação e diálogo entre as pessoas que convivem entre si, permitindo compartilhar conhecimentos e experiências. Dessa forma, este trabalho apresenta o desenvolvimento de um sistema de comunicação nas dependências do Instituto Federal do Amazonas (IFAM), permitindo a troca de informações entre os servidores e alunos sem que houvesse a necessidade dos funcionários se deslocarem até as salas de aula, evitando atrasos e contratempos na passagem de comunicados. É notório que todos os dias surgem notícias ou informes a serem compartilhados, mas devido a dispersão dos alunos na faculdade, torna-se difícil o meio pelo qual os dados devam ser anunciados, de forma que todos vejam simultaneamente. Além disso, o processo na divulgação de avisos ou notícias de última hora muitas vezes não são transmitidos publicamente. Atualmente, não existe no Instituto Federal do Amazonas um dispositivo que propicie esse tipo de comunicação demonstrado neste trabalho. Sendo assim, criamos um sistema de baixo custo e embarcado para comunicação exclusiva do IFAM entre alunos, servidores e professores, que supra a necessidade atual de uma tecnologia com o objetivo de tornar ágil, planejada e eficaz a troca de informações. A falta deste tipo de sistema, em ambientes escolares, enfatiza a pesquisa e o desenvolvimento na área tecnológica proporcionando uma melhoria contínua em processos de comunicação, mediante a preocupação na falta de transparência com os alunos e professores. Dessa maneira, utilizamos como materiais para o desenvolvimento do sistema de comunicação o microcontrolador ESP8266, embarcado à plataforma de *hardware* NodeMCU, que é responsável pelo recebimento do arquivo em JSON, que interpreta exibindo uma string (texto) numa matriz de LED. Este microcontrolador é comumente utilizado e tem a capacidade de receber códigos através do WiFi, sendo adequado para o projeto, pois é utilizado a rede local do IFAM para o envio de mensagens. Portanto, com um produto final de baixo custo, facilidade de uso e confiabilidade nas respostas, foram realizados testes em salas de aula, cujos resultados foram satisfatórios durante todo o processo da passagem de informações.

**Palavras-chave:** Sistema de Comunicação, Página *Web*, ESP8266, NodeMCU.

## **ABSTRACT:**

Communication is fundamental for interaction and dialogue between people who live together, allowing to share knowledge and experiences. Thus, this work aims to develop a communication system in the dependencies of the Federal Institute of Amazonas, allowing the exchange of information between servers and students without the need for staff to travel to classrooms, avoiding delays and mishaps in passing communications. It is noticeable that news or reports are being shared every day, but due to the dispersal of students in college, the means by which data must be announced is difficult for all to see simultaneously. In addition, the process of disclosing warnings or breaking news is not publicly transmitted. Currently, the Federal Institute of Amazonas does not have a device that provides the objectives demonstrated by this study, the proposal to create a system for exclusive communication of IFAM between students, servers and teachers of low cost and embedded, which meets the current need for a technology. vision to make the exchange of information agile, planned and effective. The lack of this type of system in school environments emphasizes research and development in the technological area providing continuous improvement in communication processes, due to the concern about the lack of transparency with students and teachers. Thus, the ESP8266 microcontroller, embedded on the NodeMCU hardware platform, is responsible for receiving the JSON file, which interprets by displaying the string (text) in the LED array. This microcontroller is commonly used and has the ability to receive codes via WiFi and is suitable for the project as it uses the IFAM LAN for sending messages. Therefore, with a low cost final product, ease of use and reliability of answers, tests were carried out in classrooms, whose results were satisfactory during the whole process of passing information.

**Keywords:** Communication System, Web page, ESP8266, NodeMCU.

## LISTA DE FIGURAS

Figura 1: Módulo ESP-12E. ....	21
Figura 2: NodeMCU. ....	23
Figura 3: Pinagem do NodeMCU ESP-12.....	23
Figura 4: Módulo RTC.....	24
Figura 5: Exemplos de letreiros com LEDs. ....	26
Figura 6: Matriz de led 32x8.....	26
Figura 7: Pinagem da matriz 8x8. ....	27
Figura 8: Tela inicial.....	29
Figura 9: Opções na tela inicial.....	29
Figura 10: Tela inicial da IDE. ....	30
Figura 11: Tela para inserção da URL do ESP8266. ....	31
Figura 12: Instalação da Esp8266. ....	32
Figura 13: Seleção da placa NodeMCU.....	32
Figura 14: Exemplo de codificação em PHP.....	33
Figura 15: Exemplo da linguagem CSS. ....	34
Figura 16: Exemplo do código SQL. ....	35
Figura 17: Arquitetura do sistema. ....	39
Figura 18: Página index. ....	40
Figura 19: Páginas cadastro, principal e dispositivo. ....	40
Figura 20: Componentes utilizados.....	42
Figura 21: Placa 01 simulada com <i>Fritzing</i> . ....	43
Figura 22: Placa 02 simulada com <i>Fritzing</i> . ....	43
Figura 23: Montagem do projeto elétrico em protoboard. ....	44
Figura 24: Placa 01.....	44
Figura 25: Placa 02.....	45
Figura 26: Função <code>intervalo()</code> ; .....	48
Figura 27: Função <code>Buzzer()</code> ; .....	48
Figura 28: Função <code>printDateTime()</code> ; .....	49
Figura 29: Função <code>handleSetCommand()</code> ; .....	49
Figura 30: Painel de controle do XAMPP.....	50
Figura 31: Página <i>phpMyAdmin</i> . ....	50
Figura 32: Tabela <i>device</i> .....	51

Figura 33: Tabela <i>user</i> .	52
Figura 34: Página de login.	53
Figura 35: Página Cadastro.	54
Figura 36: Página Dispositivos.	54
Figura 37: Página principal.	55
Figura 38: Modelo de impressora Cloner DH+.	56
Figura 39: Modelagem 3D da caixa.	56
Figura 40: Caixa com o cabo de força.	57
Figura 41: Dispositivo Vision ligado.	57
Figura 42: Rede ESP.	58
Figura 43: Acesso a rede embarcados.	58
Figura 44: Rede Servidor.	59
Figura 45: Servidor XAMPP.	59
Figura 46: Página <i>web</i> .	60
Figura 47: Página de login.	60
Figura 48: Efetuando cadastro.	61
Figura 49: Enviando mensagens.	62
Figura 50: Vision recebendo mensagem.	62
Figura 51: Versão final do circuito eletrônico do sistema.	63
Figura 52: Finalizada do dispositivo com a caixa fechada.	64
Figura 53: Matriz exibindo mensagens da interface <i>web</i> .	64
Figura 54: Dois dispositivos de comunicação no IFAM.	65

## LISTA DE TABELAS

Tabela 1 – Especificações ESP-12E.....	21
Tabela 2 – Descrição dos pinos do módulo RTC.....	25
Tabela 3 – Descrição dos pinos da matriz de led MAX7219.....	27
Tabela 4 – Orçamento dos componentes do projeto.....	45

## LISTA DE ABREVIATURAS E SIGLAS

CAD	Desenho Auxiliado por Computador
CAE	Coordenação de Administração Escolar
CI	Circuito Integrado
CSS	Folha de Estilo em Cascatas
HTML	Linguagem de Marcação de Hipertexto
HTTP	Protocolo de Transferência de Hipertexto
IDE	Ambiente de desenvolvimento integrado
IFAM	Instituto Federal do Amazonas
IOT	Internet das Coisas
IP	Protocolo de internet
LAN	Rede de Área Local
LED	Diodo Emissor de Luz
MQTT	Fila de Mensageria de Transporte de Telemetria
PHP	Linguagem de Programação <i>Web</i>
PLA	Poliácido Láctico
RTC	Relógio em Tempo Real
SPI	Interface Periférica Serial
SQL	Linguagem de Consulta Estruturada
URL	Localizador Uniforme de Recursos
USB	Barramento Serial Universal
XAMPP	Plataforma de Servidor Independente
WEB	Rede Mundial de Computadores
WIFI	Fidelidade Sem Fio

## LISTA DE SÍMBOLOS

$\Omega$  Unidade de medida da Resistência elétrica

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>16</b>
1.1	OBJETIVOS .....	18
1.2	JUSTIFICATIVA .....	18
1.3	METODOLOGIA.....	19
1.4	ORGANIZAÇÃO DO TEXTO.....	20
<b>2</b>	<b>REFERENCIAL TEÓRICO.....</b>	<b>21</b>
2.1	HARDWARE.....	21
2.1.1	Esp8266 .....	21
2.1.2	NodeMCU.....	22
2.1.3	Módulo RTC DS3231 .....	24
2.1.4	Matriz de LED MAX7219 .....	25
2.2	SOFTWARES.....	28
2.2.1	Programação da NodeMCU .....	28
2.2.1.1	IDE do Arduino.....	28
2.2.2	Programação web.....	33
2.2.2.1	Linguagem de programação PHP.....	33
2.2.2.2	Linguagem de Folhas de estilo CSS.....	34
2.2.2.3	Linguagem de banco de dados SQL.....	35
2.2.2.4	Linguagem de marcação HTML.....	35
2.2.3	XAMPP .....	36
2.2.4	AutoCAD .....	37
2.2.5	Fritzing.....	37
<b>3</b>	<b>METODOLOGIA .....</b>	<b>38</b>
3.1	VISÃO GERAL DO PROJETO .....	38

3.2	CIRCUITO ELETRÔNICO .....	41
3.2.1	Montagem das placas .....	42
3.2.2	Levantamento de custo .....	45
3.3	PROGRAMAÇÃO DA NODEMCU .....	46
3.3.1	Configurações iniciais.....	46
3.3.2	Linhas de comando em <i>loop</i> . .....	47
3.3.2.1	Função <code>intervalo()</code> .....	47
3.3.2.2	Função <code>Buzzer()</code> .....	48
3.3.2.3	Função <code>printDateTime()</code> .....	48
3.3.2.4	Função <code>handleSetCommand()</code> .....	49
3.4	CRIAÇÃO DO BANCO DE DADOS .....	50
3.5	PÁGINAS <i>WEB</i> .....	52
3.5.1	Index.....	53
3.5.2	Cadastro.....	53
3.5.3	Dispositivos .....	54
3.5.4	Principal.....	55
3.6	PROJETO MECÂNICO .....	55
3.7	OPERAÇÃO DO SISTEMA .....	57
<b>4</b>	<b>RESULTADOS.....</b>	<b>63</b>
<b>5</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>66</b>
5.1	TRABALHOS FUTUROS .....	67
<b>6</b>	<b>REFERÊNCIAS.....</b>	<b>68</b>

## 1 INTRODUÇÃO

A internet das coisas ou “Internet of Things” (IoT) tem como finalidade conectar equipamentos eletrônicos na internet, para a troca de informações, controle, ou até mesmo a substituição de seres humanos em determinadas atividades. Esse conceito de IoT vem crescendo com o passar dos anos e diversas aplicações podem ser associadas a esse assunto, como por exemplo, supervisão artificial, automação residencial, carros inteligentes, wearables, entre outros. Segundo Deimar, Sobreira e Lima (2017), essa evolução tecnológica gera a união de um mundo real ao imaginário, fazendo com que se estabeleça a comunicação entre pessoas e objetos.

A IoT viabiliza soluções em diferentes cenários como aplicações na área da saúde, transporte, monitoramento de tráfego, agronegócios, indústria 4.0, entre outras (Ashton, 2009). Certamente, a tendência é aumentar o uso para outras áreas devido ao seu potencial de aplicação. Segundo Santos et. al. (2016), a Internet das Coisas, em poucas palavras, nada mais é que uma extensão da Internet atual, que proporciona aos objetos do dia-a-dia (quaisquer que sejam), mas com capacidade computacional e de comunicação, se conectarem à Internet.

A conexão possibilita o controle de variáveis e consequentemente ações de objetos ou atuadores que se conectam entre si sem a necessidade da interferência humana, cooperando nas atividades do dia-a-dia, como por exemplo, ligar/desligar automaticamente um ar-condicionado, televisões, geladeiras e projetos desenvolvidos para tal, bem como na obtenção de dados dos dispositivos em tempo real. Estas habilidades geram inúmeras possibilidades tanto no âmbito acadêmico quanto no ramo industrial e residencial. Segundo Leite, Martins e Ursini (2017), a IoT realiza uma nova transformação digital, conectando dispositivos, incrementando valores de negócios, redefinindo organizações e gerando redes de comunicações.

De modo geral, a comunicação é fator determinante no convívio social entre as pessoas, que estabelecem diálogos para a troca de informações, seja por meio da internet, mensagens, cartas, ligações ou e-mails. Existe por trás um ramo de tecnologias usadas, que torna isso possível, que são o WiFi, Bluetooth, IEEE, NFC e RFID. Essas tecnologias ampliaram a capacidade dos seres humanos de observarem e transmitir conteúdo entre si. Segundo Santos, Gomes e Duarte

(2016), o espaço virtual vem demonstrando ser um importante dispositivo para o processo de disponibilização das informações, um veículo rápido e acessível para o provimento de respostas.

A partir desses conceitos, o presente trabalho considera a aplicação da Internet das Coisas para facilitar a troca de informações entre os servidores e alunos do IFAM. No setor administrativo é rotineiro que os funcionários se desloquem até as salas de aula para anunciar comunicados, seja direcionado para alunos ou professores. Outra maneira é através de papéis impressos e fixados nas paredes dos corredores do Instituto.

O IoT foi a tecnologia escolhida porque garante rapidez e eficiência no que se refere ao tempo de resposta, eliminando o uso de terceiros, como o e-mail ou WhatsApp, pois nem todos os professores estão disponíveis 24 horas para responder as mensagens, visto que, estão na maior parte do tempo dentro das salas de aula. Da mesma forma, nem todos os alunos possuem o número pessoal dos docentes ou servidores para tirar dúvidas ou receber comunicados.

Os comunicados variam de acordo com as necessidades e prioridades do dia-a-dia, podendo ser informativos institucionais, comunicados de última hora, reclamações, chamadas internas para programas, acontecimentos, divulgação de bolsas, e demais informações relevantes. Não ter um planejamento bem definido na propagação de notícias e comunicados pode acarretar insatisfação, ausência de conteúdo, falta de audiência, desinformação e perda de tempo, visto que os comunicados internos são importantes dentro de qualquer organização. Segundo Barro e Matos (2015) para garantir o êxito da comunicação, é necessário que o fluxo desta comunicação ocorra de maneira planejada.

Com base nessa problemática, o objetivo deste trabalho é desenvolver um dispositivo para otimizar a comunicação da Coordenação de Administração Escolar (CAE) no que diz respeito a transmissão de avisos para a comunidade de alunos e professores. Além disso, o dispositivo pode favorecer também os docentes, que podem solicitar do responsável o envio de uma mensagem específica para seus alunos que estão em espera, como o nome do laboratório onde a aula vai ser ministrada, reduzindo o tempo de todos os envolvidos.

O projeto conta com uma página *web* que pode ser acessada em uma rede local pelos funcionários do Instituto e um dispositivo de Hardware, denominado de Vision, que deve ser instalado nas salas de aula, o qual possibilita a visualização

da mensagem transmitida a longa distância, bem como na elaboração de um alarme sonoro. A comunicação está centrada na rede local da Instituição, disponibilizada via WiFi gratuitamente para todos que fazem parte do IFAM. Dessa forma, a internet é um dos principais elementos que precisa estar disponível para realizar a conexão entre os componentes e o chip controlador ESP8266.

## 1.1 OBJETIVOS

Este trabalho tem por objetivo desenvolver um sistema de comunicação de baixo custo, composto por um chip controlador, um sistema *web* e alarme para anunciar informações por meio de um exibidor de mensagens constituído por uma matriz de LED, com a finalidade de auxiliar na comunicação interna entre alunos, servidores e professores no desenvolvimento social dentro da Instituição.

Os objetivos específicos do trabalho são:

- Diminuir custos na criação de sistemas de comunicação eficientes e operativos;
- Ampliar a rede de comunicação do IFAM e de outras instituições;
- Manter os discentes e professores das instituições informados com a distribuição de comunicados;

## 1.2 JUSTIFICATIVA

A demora na transmissão de comunicados verbalmente ou conteúdos distorcidos são fatores de motivação para a realização deste trabalho, onde o tempo é um elemento significativo quando se trata de divulgar uma mensagem ao mesmo tempo para tantas pessoas. Além disso, o IFAM é composto por mais de 20 salas de aulas, divididas em cursos do Tecnólogo, Engenharia, Subsequente, Integrado, EJA, Especialização e salas de laboratórios.

Neste sentido, é explícito que existe uma quantidade elevada de alunos no campus do Instituto Federal do Amazonas localizado no Distrito Industrial. Este local é estruturado de áreas amplamente largas, ocasionando uma

distância entre as salas de aulas e o departamento administrativo, incluindo as salas de professores.

A CAE é responsável pelas questões pedagógicas aos discentes, o que inclui na passagem de informações importantes. Porém, os servidores utilizam meios de comunicação arcaico, como por exemplo, protocolos via papel na solicitação de alguma coisa referente a instituição, bem como imprimir avisos em papéis e colocar nos corredores, sucedendo numa espera indeterminada e muitas vezes despercebida pelo aluno.

Tendo em vista que os servidores possuem diversas tarefas e funções no dia-a-dia, o tempo de espera por notícias pode se estender mais do que o planejado, atrapalhando uma relação direta com os estudantes, visto que, muitos problemas escolares são resultados de alguma falha na comunicação, sendo este fator, uma questão presente na Instituição.

A prevenção para evitar esse tipo de situação é uma das principais motivações deste trabalho, pois não somente é pensando no bem-estar dos alunos, servidores e professores, mas também na rapidez, eficiência e prazo a serem cumpridos, uma vez que estes elementos podem comprometer o desempenho das atividades na instituição. Além disso, o projeto aqui demonstrado não se aplica somente ao IFAM, mas pode servir para outras instituições, públicas ou privadas.

Dessa forma, as ferramentas escolhidas para a transmissão de mensagens entre o receptor e o emissor, são fatores decisivos em questão de melhorias e modernidade, uma vez que novas plataformas estão surgindo com o passar dos anos.

### 1.3 METODOLOGIA

O trabalho é elaborado inicialmente com o uso de um software que permite desenhar objetos em 3D, simulando um projeto real. Logo, o dispositivo Vision é desenhado para a impressão do mesmo em uma impressora 3D. Em seguida, um estudo para idealizar a montagem do projeto elétrico e como ficará distribuído os componentes utilizados dentro do dispositivo.

Visto que a função é transmitir informações e agilizar este processo de comunicação dentro do Instituto, o desenvolvimento de uma página web foi também elaborado para permitir que os responsáveis possam escrever uma mensagem e em seguida carregar para o dispositivo Vision.

Dessa forma, fez-se necessário o uso de um banco de dados para possibilitar o cadastramento de informações a serem solicitadas como requisito de acesso às páginas principais. Assim, torna o sistema seguro contra eventuais acessos indevidos por pessoas não autorizadas. Estes acessos são possíveis com o uso de uma rede local disponibilizada pelo IFAM.

#### 1.4 ORGANIZAÇÃO DO TEXTO

O trabalho é dividido em seis capítulos, que compreende todo o processo de pesquisa e evolução do projeto para alcançar os resultados finais. Inicia-se com a Introdução, onde é descrito os objetivos a serem alcançados e as justificativas que levou a tal desenvolvimento do projeto. Em seguida, o Referencial teórico que é baseado nas pesquisas e assuntos relacionados aos componentes utilizados, englobando a parte do Hardware e Software do sistema.

O terceiro capítulo é relacionado a parte principal do trabalho, que consiste na Metodologia, ou seja, engloba os métodos utilizados para alcançar todas as funções do dispositivo. Além disso, é descrito detalhadamente os processos de maneira clara e objetiva. Essa parte comporta a visão geral do projeto, o circuito eletrônico, programações, criação do banco de dados, projeto mecânico e a operação do sistema.

Por fim, os últimos capítulos são para contemplar e concluir os resultados finais do trabalho, as metas alcançadas e as futuras, seguidas pelas Referências.

## 2 REFERENCIAL TEÓRICO

Neste trabalho o referencial teórico expõe uma breve apresentação dos componentes utilizados, além de exibir conceitos fundamentais do software e hardware do projeto.

### 2.1 HARDWARE

Os principais hardwares utilizados na elaboração do projeto mecânico e elétrico do sistema foram: Esp8266, NodeMCU, Módulo RTC DS3231 e a Matriz de LED MAX7219. Todos esses componentes são descritos detalhadamente a seguir.

#### 2.1.1 Esp8266

O microcontrolador ESP8266 segundo Oliveira (2017) está no mercado operando desde o ano de 2014 e é composto por um sistema de comunicação WiFi próprio, sendo este o seu grande diferencial no mercado. É utilizado para produzir sistemas embarcados, e possui diversas vantagens quanto à praticidade, custo e aplicação. Devido a esses fatores é amplamente utilizado em projetos de IoT.

Encontra-se atualmente diversos módulos desse microcontrolador, como por exemplo, ESP-01, ESP-02, ESP-03, ESP-04, dentre outros. Neste trabalho, foi utilizado o módulo ESP-12E, embarcada em uma plataforma NodeMCU, exibida com mais detalhes a seguir. A Figura 1 exibi o modelo ESP-12E.

**Figura 1: Módulo ESP-12E.**



Fonte: AI-THINKER.

A programação pode ser feita usando as plataformas de programação: IDE do Arduino ou LUA. É um componente com o WiFi embutido, além de ser capaz de integrar sensores e demais componentes com aplicações específicas usando os GPIOs (*General Purpose Input Output*), sendo este um fator de destaque. A Tabela 1 ilustra as especificações composta pela ESP-12E.

**Tabela 1: Especificações ESP-12E.**

Protocolo de comunicação	<i>Wireless</i> padrão 802.11 b/g/n
Tensão de operação	3.3V
Consumo de corrente	70 mA
Quantidade de GPIOs	11 portas
Conversor analógico digital (A/D)	10 bits
Velocidade de processamento	80 ~160 MHz
Alcance máximo	90 metros
Limite de velocidade de conexão	300 MBps
Dimensões	24 x 16 x 3,2 mm

Fonte: AI-THINKER.

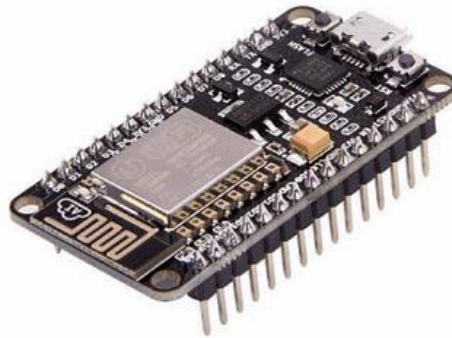
### 2.1.2 NodeMCU

A plataforma NodeMCU é uma placa produzida para facilitar o desenvolvimento de aplicações ou projetos integrada ao módulo ESP-12E. A junção das duas placas se dá por meio da soldagem, e é composto por uma interface usb-serial e um regulador de tensão 3.3V.

Algumas características importantes que compõe essa plataforma são o custo baixo, o tamanho reduzido, a facilidade de uso, e a quantidade de funcionalidades que possui, bem como na integração com sensores e atuadores

existentes no mercado, tornando-a um componente relevante na utilização desse projeto. A Figura 2 ilustra a plataforma NodeMCU.

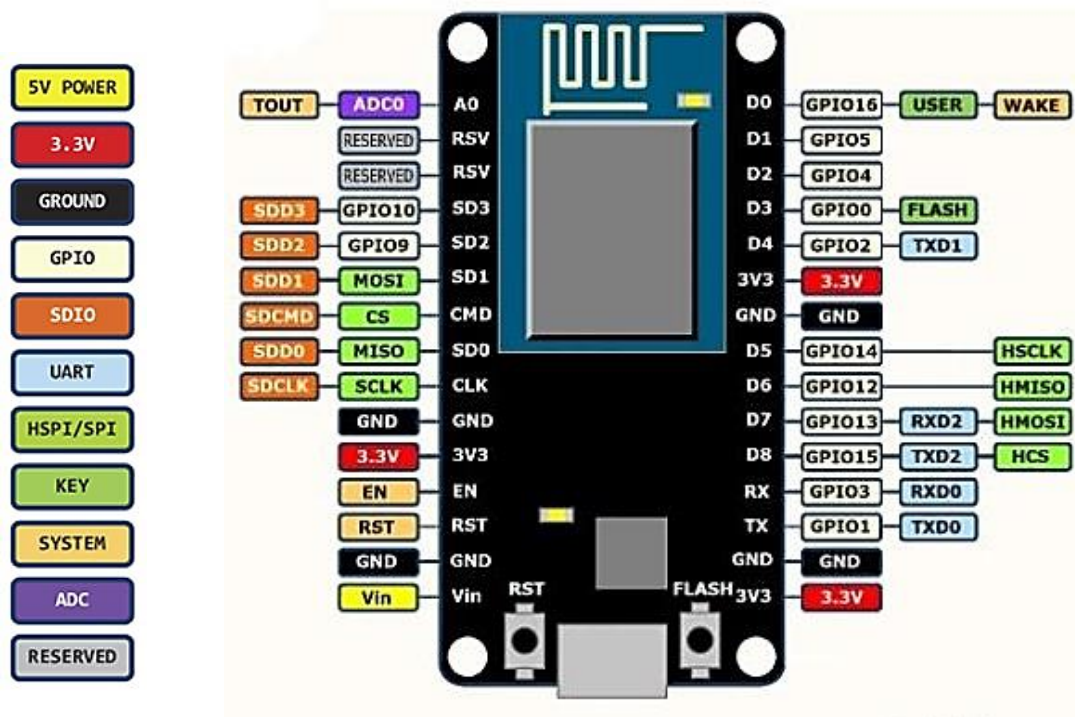
Figura 2: NodeMCU.



Fonte: CALDAS (2018).

O uso dessa plataforma pode abranger áreas residenciais, monitoramentos, rede de sensores, localização, dentre outros. De acordo com Murta (2018) a grande vantagem dessa placa NodeMCU é que ela funciona como se fosse um Arduino. Você conecta o cabo USB, e com a IDE Arduino carrega-se os programas na placa. A Figura 3 mostra a pinagem do NodeMCU.

Figura 3: Pinagem do NodeMCU ESP-12.



Fonte: ARDUNAUT.

### 2.1.3 Módulo RTC DS3231

O módulo RTC (do inglês, *Real Time Clock*) que traduzido para o português significa *Relógio em tempo real*, é um componente pequeno e compacto, fabricado para que os terminais entrem e saiam com facilidade nos protoboards e placas. É compatível com diversos microcontroladores.

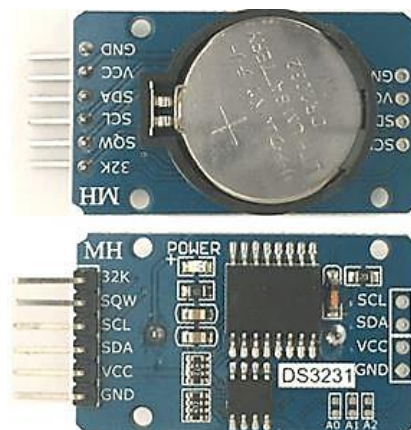
O RTC tem a capacidade de fornecer informações completas que compreende datas e horas. Uma das vantagens desse módulo, é ter acoplado uma bateria de lítio em casos de falta de energia, visando preservar os dados do componente.

O mecânico de manutenção com experiência em Automação e Desenvolvimento de Projetos Matheus Gebert Straub afirma que:

Com este recurso, não é necessário preocupar-se com quedas de energia, por exemplo, pois mesmo sem a alimentação do microcontrolador, o RTC ainda pode continuar acompanhando o tempo e permanecer configurado. (STRAUB, 2019)

Portanto, aspectos de tempo e momento são importantes para o projeto, pois se trata de um ambiente escolar. De acordo com Ferreira e Junior (2017) as informações de data englobam dia da semana, dia do mês e ano, além, é claro, das funções de horas, minutos e segundos, nos formatos de 12 ou 24 horas. O módulo RTC é exibido na Figura 4.

Figura 4: Módulo RTC.



Fonte: <https://components101.com/modules/ds3231-rtc-module-pinout-circuit-datasheet>

Pode-se observar através da Figura 4 que o módulo RTC DS3231 é composto por 6 pinos principais, cuja funções são mostradas na Tabela 2 a seguir.

**Tabela 2: Descrição dos pinos do módulo RTC**

<b>PINO</b>	<b>NOME</b>	<b>FUNÇÃO</b>
1	32K	Saída de 32KHz
2	VCC	Entrada de alimentação 3,3V ~ 5V
3	SQW	Saída de interrupção para alarmes
13	GND	Tensão de referência (terra)
15	SDA	Entrada / saída de dados serial I2C
16	SCL	Entrada de relógio serial I2C

Fonte: Datasheet DS3231.

#### **2.1.4 Matriz de LED MAX7219**

O módulo Matriz de LED MAX7219 é um componente da eletrônica ideal para disseminar informações, composto por Diodos emissores de luz (LED). Essa matriz é constantemente utilizada em letreiro de lojas, relógios digitais, sinais de trânsito, entre outros, com base na iluminação temática de LEDs.

Assim, a escolha da matriz para o projeto, deu-se devido as suas características importantes que possibilita mostrar informações, e principalmente chamar atenção devido a cor intensa das letras. De acordo com Wright:

A percepção da cor pelo ser humano é uma experiência subjetiva, só fazendo sentido se o mesmo puder percebê-la. Nem mesmo a cadeia de eventos envolvendo receptores específicos na retina e o córtex visual do cérebro são suficientes para determinar sua existência, a qual o ser humano toma conhecimento somente após a interpretação pelo seu consciente (Wright, 1963, p. 20).

Essa percepção da cor permite que os usuários percebam a média ou curta distância informações importantes. Dessa maneira, a vantagem de utilizar a matriz de LED de acordo com Silva (2013) é a falta resistência ao ar livre, fácil fabricação, manutenção e baixo consumo. A Figura 5 mostra exemplos das matrizes de LED no dia a dia.

**Figura 5: Exemplos de letreiros com LEDs.**



Fonte: <https://www.mundodeled.com.br/painel-de-led-letreiro-digital-100cmx20cm-alto-brilho>

A matriz de led utilizada foi uma junção de 4 matrizes 8x8 interligadas em cascata, para dar maior visibilidade para o público discente e além disso, esse modelo é compatível com a maioria dos sistemas microcontroladores, inclusive a Esp8266. Dessa forma, o tamanho da matriz ficou por 32x8. A Figura 6 ilustra a montagem.

**Figura 6: Matriz de LED 32x8.**



Fonte: <https://www.filipeflop.com/produto/modulo-4-matrizes-de-led-8x8-com-max7219/>

Como dito anteriormente, o módulo é uma montagem de quatro matrizes 8x8 (8 linhas e 8 colunas), do mesmo modo, a pinagem é padrão para qualquer matriz montada em cascata, exibido na Figura 7.

**Figura 7: Pinagem da matriz 8x8.**



Fonte: <https://www.filipeflop.com/produto/modulo-matriz-de-led-8x8-com-max7219/>

Essa matriz é ideal para montagem de painéis de led, com tamanhos grandes, médios e pequenos. Como o tamanho da matriz é 32x8, significa que são 256 leds controlados simultaneamente. A pinagem, o nome e a função do módulo são exibidos na Tabela 3.

**Tabela 3: Descrição dos pinos da matriz de led MAX7219.**

<b>PINO</b>	<b>NOME</b>	<b>FUNÇÃO</b>
1	DIN	Entrada de dados seriais
4, 9	GND	Tensão de referência (terra)
12	CS	Entrada de seleção de chip
13	CLK	Entrada de relógio serial
19	VCC	Tensão de alimentação positiva 5V

Fonte: Datasheet MAX7219.

## 2.2 SOFTWARES

Os softwares utilizados para elaboração e funcionamento do projeto foram: Arduino Software (IDE), XAMPP, Linguagens PHP, CSS, SQL e HTML, além do AutoCAD e Fritzing. Todos são descritos detalhadamente a seguir.

### 2.2.1 Programação da NodeMCU

Como dito anteriormente, a plataforma NodeMCU pode ser programada de duas formas: usando a linguagem LUA e a IDE do Arduino. O escolhido para este projeto foi a IDE do Arduino, pela facilidade de uso e possuir uma vasta aplicabilidade, além de bibliotecas já existentes.

#### 2.2.1.1 IDE do Arduino

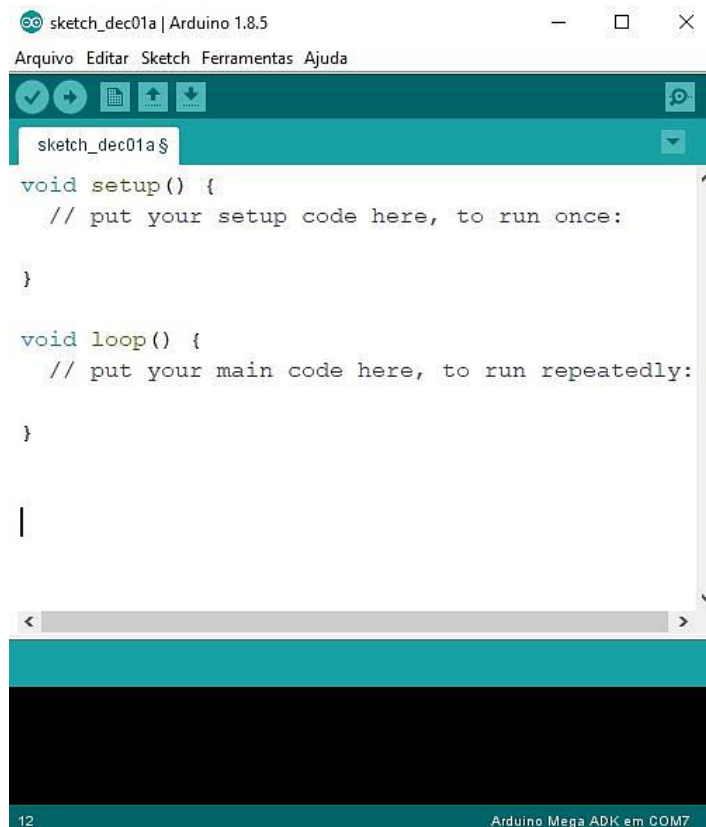
O ambiente de desenvolvimento integrado do Arduino IDE (*Integrated Development Environment*) compõe uma interface amigável e recursos que permitem manusear facilmente. É nesse software que se escreve o programa a ser compilado pelo microcontrolador.

É um código aberto e pode ser executado em plataformas como *Windows*, *Mac OS* e *Linux*. Segundo Perez e Darós (2013) as principais funcionalidades do IDE do arduino são salvar o código do programa, escrever o código do programa, compilar um programa e transportar o código compilado para a placa do Arduino.

Uma das vantagens é o uso da linguagem baseada em C/C++, comumente utilizada nas áreas de engenharia e programação. Essa linguagem é fácil e possui uma estrutura simples. Além disso, existem diversos exemplos disponíveis na internet com programações para todas as bibliotecas padrão.

A versão do IDE para a plataforma Arduino usada foi a 1.8.5. Disponível em: [www.arduino.cc](http://www.arduino.cc). A área onde se desenvolve o programa é chamado de *Sketch* e a tela inicial é exibido na Figura 8.

**Figura 8: Tela inicial.**



Fonte: Autor.

Pode-se observar pela Figura 8 que existem duas partes principais: o *void setup()* e o *void loop()*, respectivamente. O primeiro é dedicado a declaração das variáveis, que são executadas apenas uma vez, e o segundo são para os comandos que devem ser executados repetidamente, sendo este a função principal do programa.

Para complementar as informações sobre a IDE do Arduino no que se refere ao layout, segue abaixo em destaque as principais opções fixadas na página inicial, representada pela Figura 9.

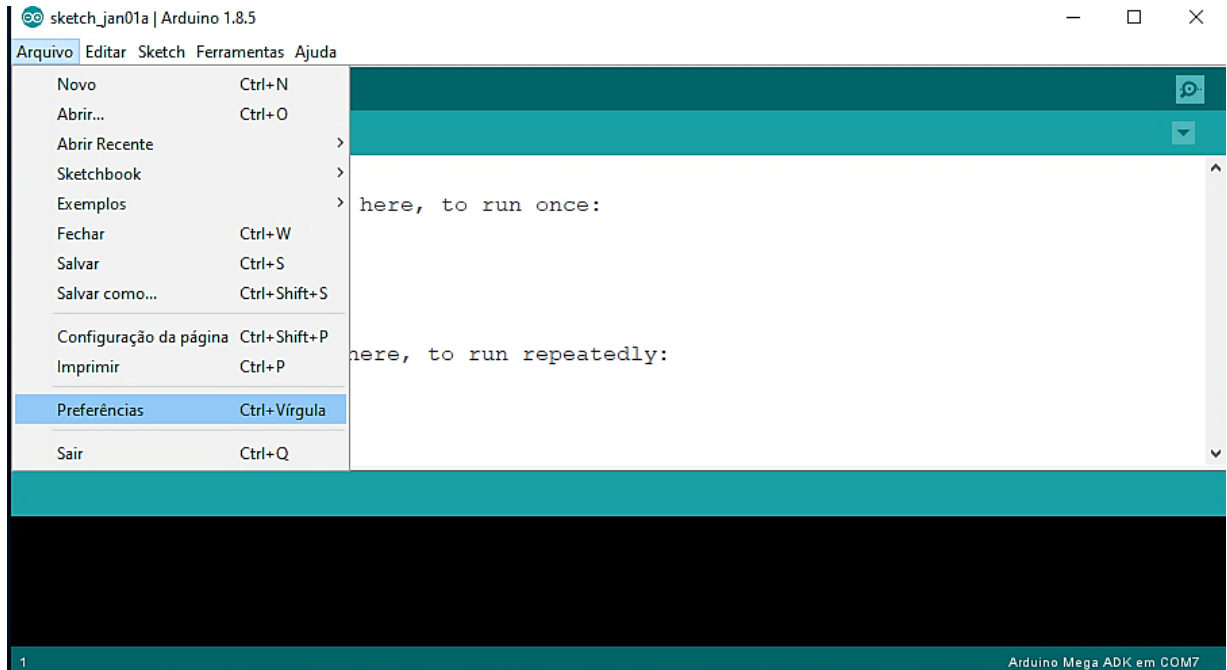
**Figura 9: Opções na tela inicial.**



Fonte: Autor.

Antes de iniciar a programação no ambiente de desenvolvimento integrado do arduino, é necessário configurar alguns dados referentes a família ESP8266. Primeiramente, deve-se selecionar a opção Arquivo localizado na barra superior da IDE, e clicar em Preferências como exibi a Figura 10.

**Figura 10: Tela inicial da IDE.**

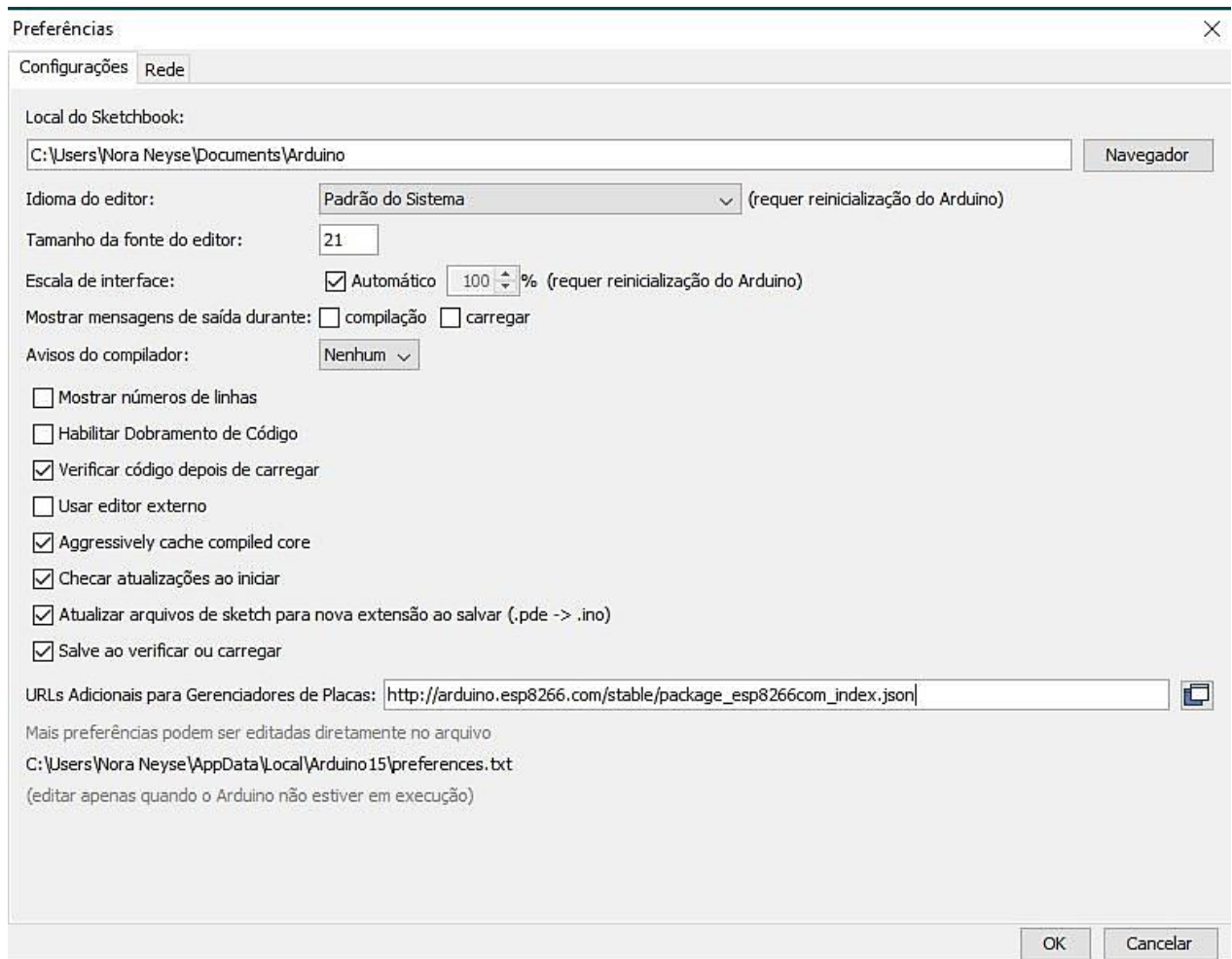


Fonte: Autor.

Conforme a Figura 10 mostrada acima, existem várias opções dentro dessa função, como criar um novo ambiente, salvar, abrir, fechar, dentre outras. Uma das características positivas da IDE são os exemplos que contem já montados dentro da própria plataforma, que exibe comentários para facilitar o entendimento daqueles que estão conhecendo o ambiente.

Por conseguinte, para adicionar as placas da família ESP, deve ser preenchido o seguinte campo: "URLs Adicionais para Gerenciadores de Placas" com o link: [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json), assim como mostrado na Figura 11.

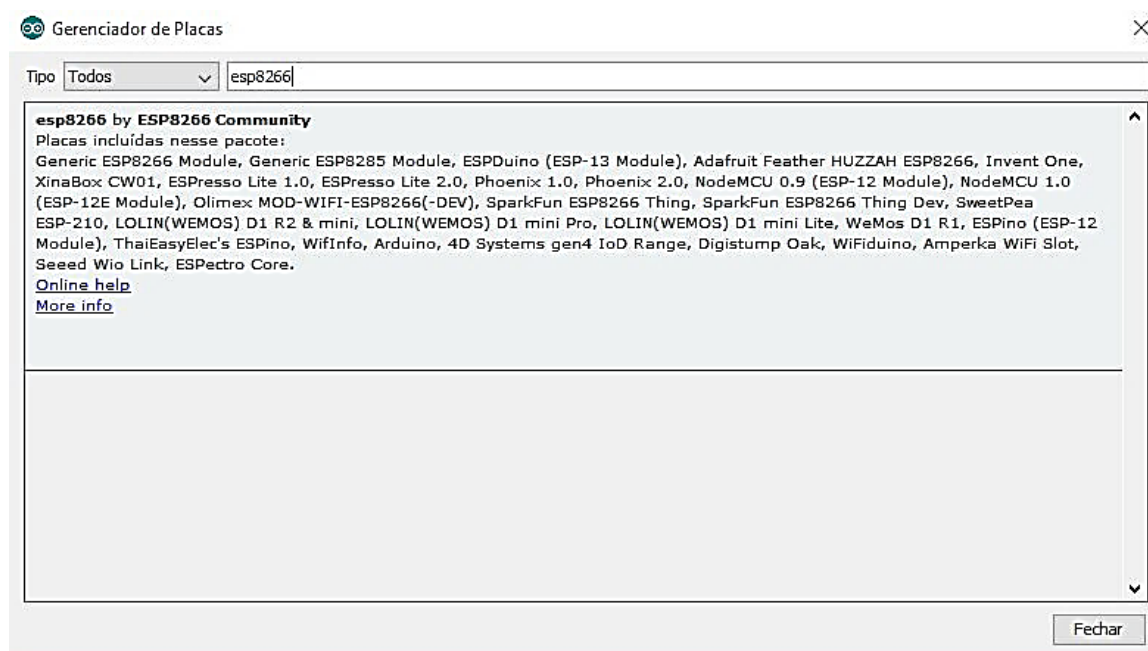
Figura 11: Tela para inserção da URL do ESP8266.



Fonte: Autor.

A próxima etapa é para realizar a instalação da placa propriamente dita na IDE do arduino, clicando na opção do menu: “Ferramentas” → “Placa” → “Gerenciador de Placas”. Em seguida, realizar uma busca com o nome: “esp8266”. Ao carregar as opções, escolher “esp8266 by esp8266 Community” e instalar a última versão de acordo com a Figura 12.

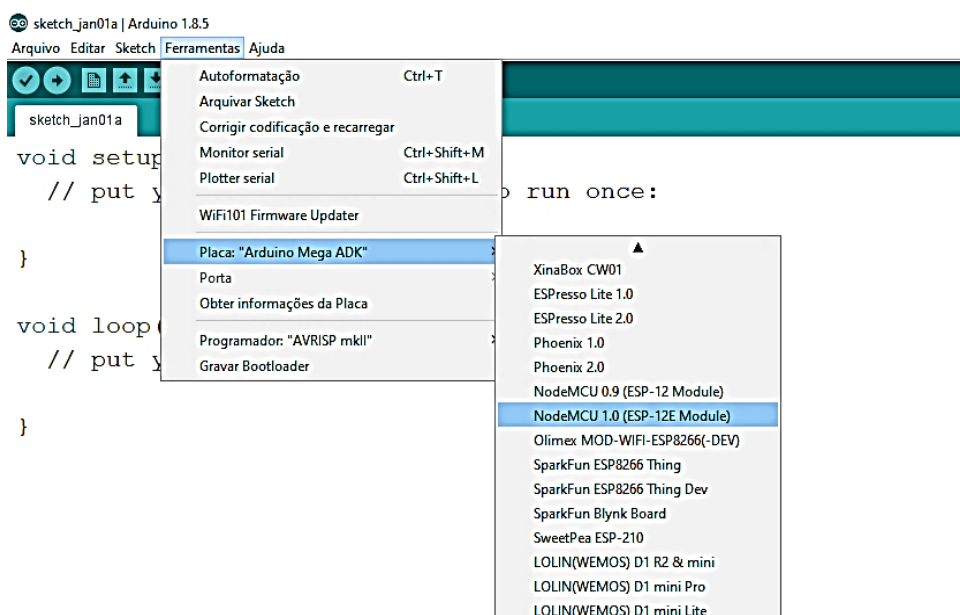
Figura 12: Instalação da Esp8266.



Fonte: Autor.

Após a instalação e todos os passos descritos anteriormente efetuados com sucesso, é possível iniciar a programação da NodeMCU, mas antes é fundamental selecionar a placa instalada. Para isso, deve-se selecionar a opção: “Ferramentas” → Placa → NodeMCU 1.0 (ESP-12E Module) como ilustra a Figura 13.

Figura 13: Seleção da placa NodeMCU.



Fonte: Autor.

## 2.2.2 Programação web

### 2.2.2.1 Linguagem de programação PHP

PHP (originalmente do inglês, *Personal Home Page*) é uma linguagem de programação que permite criar sites *web* dinâmicos, possibilitando uma interação de aplicações com o usuário através de formulários ou questionários. É amplamente conhecido por ser uma linguagem versátil, fácil de aprender e manusear.

Certamente, o PHP possui uma vantagem significativa em relação as outras, de tornar qualquer informação em um dado confidencial, seja senhas ou códigos fornecido pelas pessoas. De acordo com Barreto (2000) é possível interagir com o banco de dados e aplicações existentes no servidor, com a vantagem de não expor o código fonte para o cliente.

As principais características da linguagem PHP são listadas a seguir:

- Código aberto: todo o código fonte está disponível, inclusive o próprio site fornece documentações: [http://www.php.net.](http://www.php.net.;);
- Custo: é fornecido para o público de forma gratuita no site oficial;
- Suporte: permite a manipulação com diversos bancos de dados como MySQL, Oracle, entre outros;
- Integração com HTML: criação de páginas HTML, podendo inserir códigos em PHP;

A Figura 14 apresenta um exemplo com algumas linhas de código usando a linguagem PHP.

**Figura 14: Exemplo de codificação em PHP.**

```
1 <html>
2   <head>
3     <title> Teste </title>
4   </head>
5   <body>
6     <?php echo "TCC IFAM CMDI Manaus"; ?>
7   </body>
8 </html>
```

Fonte: Autor.

### 2.2.2.2 Linguagem de Folhas de estilo CSS

A linguagem CSS (do inglês, Cascading Style Sheets) que, traduzido para o português, significa “Folhas de Estilo em Cascata”, tem como função estilizar as páginas *web* da maneira que o operador achar mais atraente, incluindo conteúdos interativos, elementos em movimento, efeitos visuais, dentre outros.

Para Silva (2013) é uma linguagem de estilo utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como HTML ou XML. Dessa forma, pode-se determinar que a página *web* é baseada principalmente na linguagem CSS e HTML.

Ainda de acordo com Silva (2013) o desenvolvedor cria um link (ligação) para uma página que contém os estilos dentro da programação HTML, procedendo de forma idêntica para todas as páginas de um portal. Logo, quando o indivíduo quiser alterar a aparência do conteúdo da sua página, basta modificar um arquivo.

É possível alterar a cor do texto e do fundo, fonte e espaçamento entre parágrafos. Também pode criar tabelas, usar variações de layouts, ajustar imagens para suas respectivas telas e assim por diante. (Ariane, 2019). É uma linguagem fácil de aprender e utiliza as marcações de HTML. A Figura 15 ilustra como exemplo o formato utilizado.

**Figura 15: Exemplo da linguagem CSS.**

```
1  /*Cadastro*/
2  .form3
3  {
4      position: relative;
5      width: 300px;
6      height: 200px;
7      left: 250px;
8      top: 0px;
9      border-radius: 20px;
10     background-color: rgba(255,255,255,0.4);
11     margin: 2px ;
12     padding: 15px;
13     border: 10px;
14 }
```

Fonte: Autor.

### 2.2.2.3 Linguagem de banco de dados SQL

A Linguagem SQL (do inglês, *Structured Query Language*) que, traduzido para o português significa, “Linguagem de Consulta Estruturada”, tem uma ligação direta com o armazenamento de dados em um banco organizado por tabelas, que por sua vez, possui colunas, e conseqüentemente, permite que estes dados sejam armazenados em registros ou linhas.

A SQL tem por função acessar e estruturar os dados, seja para inserir, remover, editar ou atualizar. Segundo Fernandes (2017) a linguagem SQL é o padrão adotado em banco de dados mundialmente, e trata-se de uma linguagem declarativa, através dela especificamos como o “resultado” será apresentado.

Este projeto emprega o banco de dados *MySQL*, gerenciado pelo software livre *phpMyAdmin*, desenvolvido em PHP. É considerado um dos tipos de banco de dados mais utilizados envolvendo aplicações *web*. A Figura 16 ilustra um exemplo de programação SQL.

Figura 16: Exemplo do código SQL.

```
1 <?php
2 /* Variavel do Banco de Dados*/
3 $sala=isset($_GET["sala"])?$_GET["sala"]:0;
4 $ip=isset($_GET["ip"])?$_GET["ip"]:0;
5
6 $dbNomeServidor = "localhost";
7 $dbNomeUsuario = "root";
8 $dbPassword = "";
9 $dbNomeDoBD = "bd";
10
11 $con=mysqli_connect($dbNomeServidor,$dbNomeUsuario,$dbPassword);
12 mysqli_select_db($con,$dbNomeDoBD);
13
14 $vlogin = array();
15 $status=0;
16
17 $sql = "SELECT DISTINCT ip,local FROM device ";
18
19 $res = mysqli_query($con,$sql);
20 ?>
```

Fonte: Autor.

### 2.2.2.4 Linguagem de marcação HTML

HTML (HyperText Markup Language) é uma linguagem de marcação em hipertexto que permite a criação e estruturação de seções, links, cabeçalhos, parágrafos, bem como na formatação de documentos. Segundo Silva (2013) HTML indica basicamente onde colocar cada texto, imagem ou vídeo e a forma que terão ao serem colocados na página *web*.

De acordo com Pedroso (2007) essa linguagem tem o objetivo de formatar textos através de marcações especiais denominadas tags, para que possam ser exibidos de forma conveniente pelos clientes Web.

De modo geral, as tags devem aparecer em pares, como por exemplo, “<p>” e “</p>” ou “<tag>” e “</tag>”, que significam a sintaxe de *abertura* e *fechamento*. O corpo do texto fica delimitado dentro das tags criadas. Na maioria dos casos, a linguagem HTML é vinculada com PHP, evidenciado na Figura 14. Dessa forma, é uma linguagem amplamente usada, gratuita, roda em todos os navegadores e possui código aberto.

### **2.2.3 XAMPP**

A plataforma XAMPP é composta pelos principais servidores e é usada para aplicações PHP, MySQL, Apache, entre outros. É uma ferramenta de desenvolvimento e serve para ajudar os programadores testar as páginas *web* em seus computadores pessoais sem necessitar de Internet.

De forma geral, o XAMPP é um conjunto de aplicativos operando em um só ambiente, e o fator essencial, é que esta plataforma pode ser baixada gratuitamente. Segundo Lossio (2019) através do XAMPP é possível simular um servidor *web* localmente, usando a própria máquina.

Ainda de acordo com Lossio (2019) o nome XAMPP foi inspirado nas iniciais dos seguintes elementos:

- X – Sistemas operacionais (Windows, Mac, Linux e Solaris);
- A – Servidor Apache oferecido como um software livre;
- M – Inicial do MySQL;
- P – Inicial da linguagem PHP;
- P – Inicial da linguagem Pearl.

Esse software é amplamente utilizado devido a facilidade de aprender e usar no dia a dia em qualquer lugar.

#### **2.2.4 AutoCAD**

O software AutoCAD é um dos programas mais conhecidos na área de desenho, pois se trata de uma ferramenta que permite aos usuários criar, desenvolver e modelar desenhos em 2D e 3D. Atualmente existem diversos modelos de desenhos prontos disponíveis na internet, além de videoaulas que mostram passo a passo como manipular e desenvolver ilustrações.

Esse software é fácil de usar, entender e aprender tanto no ramo estudantil quanto profissional. De acordo com Mota e Valle (2011) o AutoCAD é fundamental para a inserção do profissional engenheiro no mercado de trabalho e é comercializado pela Autodesk, sendo uma poderosa ferramenta.

Diante disso, essa ferramenta foi selecionada para desenvolver a estrutura mecânica do projeto, para comportar dentro todos os componentes. A Autodesk fornece uma versão para estudantes com acesso gratuito e pode ser baixada pelo link: <https://www.autodesk.com.br/education/free-educational-software>

#### **2.2.5 Fritzing**

O Fritzing é um software gratuito usado para simulação de montagens de circuitos no protoboard, possibilitando aos usuários idealizar como ficará o resultado final do circuito montado. Assim, pode-se ter noção de como organizar o lugar de cada componente.

Segundo Hoffman (2018) essa ferramenta permite a criação do hardware eletrônico de forma simplificada, possibilitando que o usuário desenvolva circuitos sem complexidade.

Dessa forma, é possível modelar circuitos usando Arduino, Raspberry PI, ESP8266, ou somente alguns componentes eletrônicos, tornando o trabalho mais prático e simples. Além disso, é possível visualizar o layout no formato PCB e o diagrama esquemático. A ferramenta pode ser baixada pelo endereço disponível em: <https://fritzing.org/download/>.

### 3 METODOLOGIA

O projeto descrito neste trabalho tem como objetivo transmitir informações destinadas aos estudantes ou professores, através de uma página *web* para um dispositivo exibidor de mensagens chamado de Vision, baseado no conceito de Internet das coisas. Deste modo, previne-se a perda de conteúdo e a desinformação dentro da instituição.

Este dispositivo será instalado nas salas de aula do campus, onde o usuário ao acessar a página gerada pelo servidor, escreverá a mensagem que se deseja enviar, e assim será transmitida via rede WiFi para o Vision selecionado pelo indivíduo. Dessa forma, é possível destacar que o projeto é dividido nas seguintes fases:

1. Projeto e montagem do circuito eletrônico;
2. Programação na plataforma NodeMCU;
3. Geração do banco de dados;
4. Criação das páginas *web*;
5. Desenvolvimento do projeto mecânico para o circuito;
6. Conexão da operação do sistema;
7. Testes no ambiente do IFAM;

A seguir, os tópicos citados anteriormente, serão detalhados nas fases de desenvolvimento do projeto, explicados com clareza e objetividade, além dos materiais e métodos utilizados.

#### 3.1 VISÃO GERAL DO PROJETO

O projeto se concentra na rapidez e transparência do fluxo de informações que deve ser transmitido para os alunos ou professores durante os horários normais de aula, colaborando para o setor administrativo do campus utilizar métodos atuais e tecnológicos. Para isso, é ilustrada na Figura 17 a arquitetura do sistema.

Figura 17: Arquitetura do sistema.



Fonte: Autor.

Foi criado um servidor *web* e um banco de dados no software XAMPP. O servidor *web* faz cadastros e consultas de informações do banco de dados através de instruções SQL na linguagem PHP. As requisições realizadas entre a página *web* e os dispositivos acontecem através de requisições HTTP enviadas através do navegador *web* utilizando o protocolo GET, sendo esta uma ferramenta que envia informações pelo método de adição de dados no campo URL da página.

Para a conexão entre servidor e dispositivos de exibição de mensagens, foi necessário o emprego de um roteador para criação de uma rede local que realizará a transmissão dos dados via rede WiFi local.

A seleção dos dispositivos na página *web* é realizada através da marcação por *checkbox*, a qual possui o valor correspondente ao IP da ESP8266 embarcada no dispositivo. A requisição enviada é tratada especificamente para os dispositivos selecionados, onde a função `handleSetCommand()` verifica a existência de mensagem para ser exibida.

Para realizar interação com usuário foi desenvolvido um sistema de navegação de páginas composta por cadastro, dispositivo, principal e index. Na página *index* o usuário é capaz de realizar login para liberação de acesso às demais funções do projeto como mostra a Figura 18.

**Figura 18: Página index.**



Fonte: Autor.

A página *principal* é possível enviar mensagem para as salas já cadastradas. Na área de *cadastro* é utilizada para cadastrar dispositivos novos. E por fim, em *dispositivo* pode-se visualizar os dispositivos registrados, e também excluir como exibi a Figura 19.

**Figura 19: Páginas cadastro, principal e dispositivo.**



Fonte: Autor.

O dispositivo de exibição de mensagens mostrará quatro vezes as mensagens recebidas e soará o buzzer até o término da mensagem. Enquanto o mesmo não estiver executando a função de exibir mensagem o horário atual ficará passando na matriz de Led.

Uma chave de duas posições foi inserida para realizar a comutação entre o modo de configuração e modo de operação. No modo de configuração é possível visualizar na matriz de LED o IP da ESP8266 gerado pelo seu roteador, para efetuar

o cadastro no banco de dados via página *web*. O modo de operação é usado para recepção e exibição de mensagens.

Portanto, para a validação do projeto, a metodologia de desenvolvimento é definida em seis partes: Projeto eletrônico, Programação da NodeMCU, Criação do banco de dados, Páginas *web*, Projeto mecânico e Operação do sistema.

### 3.2 CIRCUITO ELETRÔNICO

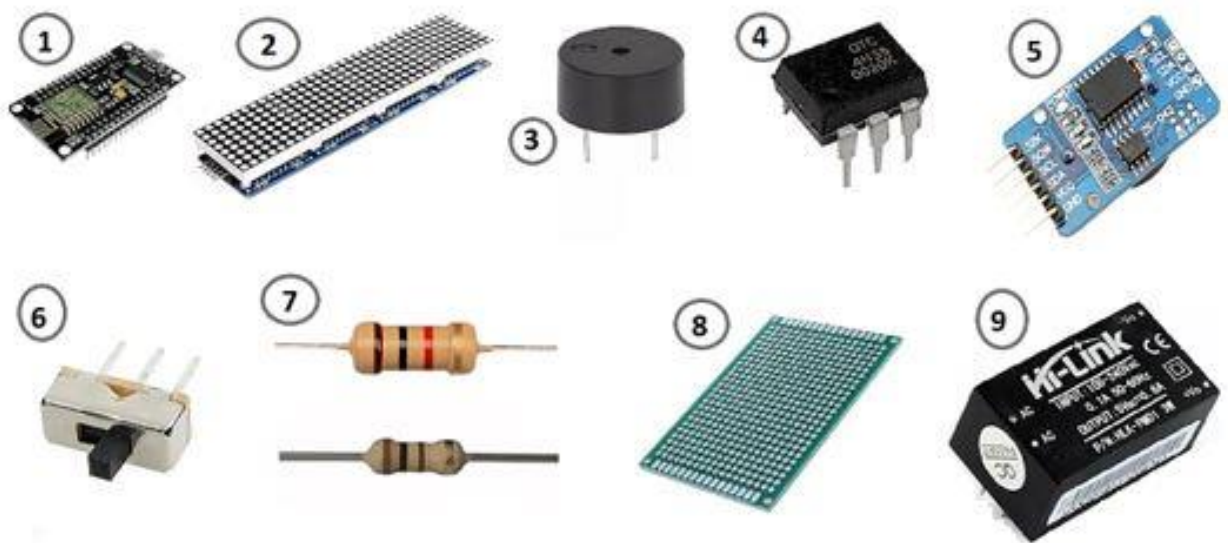
Para a montagem do projeto eletrônico, foi utilizado diversos componentes que auxiliam no funcionamento do circuito, com o objetivo de desempenhar as funções listadas a seguir.

1. NodeMCU: é uma plataforma open source da família ESP8266 criado para ser utilizado no desenvolvimento de projetos IoT.
2. Matriz 32x8: Para o recebimento de informação via página *web*, as quais são feitas pelas portas digitais da matriz de Leds.
3. Buzzer: é um componente de geração de ruídos sonoros a partir da excitação elétrica e irá apitar por um determinado período de tempo quando a matriz de Leds receber uma informação via *web*.
4. Optoacoplador 4N35: O uso desse componente se dá pelo fato de que o microcontrolador (NodeMCU) fornece 3.3 volts e é necessário de 5 volts para acionar o Buzzer.
5. Módulo RTC: Este componente é usado para fornecer “data” e “hora” enquanto não é mandada nenhuma informação para a matriz de leds.
6. Chave interruptora: Chave de duas posições, disponibilizando duas opções ao usuário, o primeiro modo de operação, e a segunda configuração.
7. Resistores: limita a corrente do buzzer e optoacoplador (1K $\Omega$ ), e do botão (100 $\Omega$ ).
8. Placas de CI: Duas placas de circuito impresso para a montagem dos componentes.

9. Fonte de alimentação: A fonte serve para transformar a tensão de 110V proveniente da rede elétrica para 5V.

A partir da aquisição de todos os materiais descritos anteriormente, foi possível realizar a montagem do circuito eletrônico exposto no tópico seguinte. A Figura 20 mostra todos os 9 componentes citados.

**Figura 20: Componentes utilizados.**

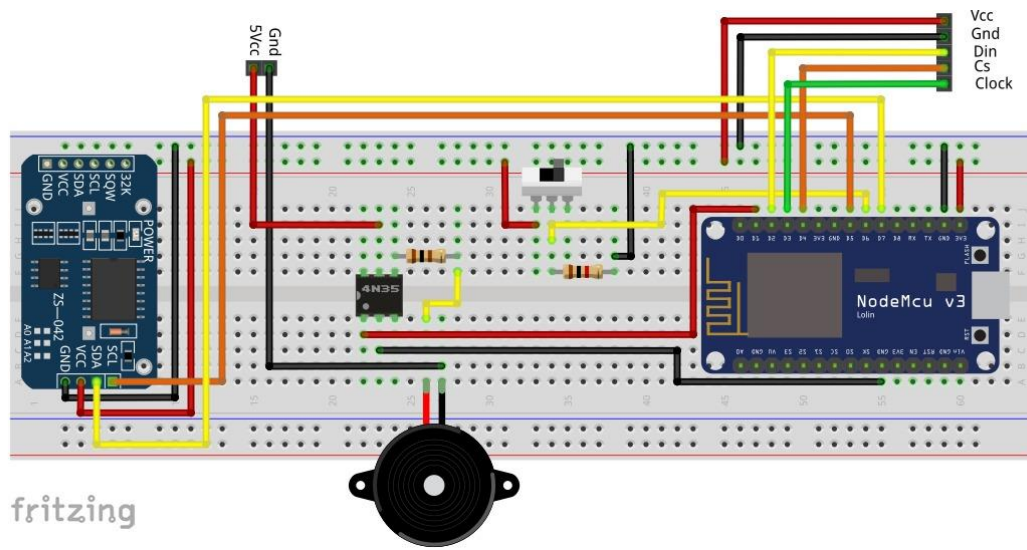


Fonte: Autor.

### 3.2.1 Montagem das placas

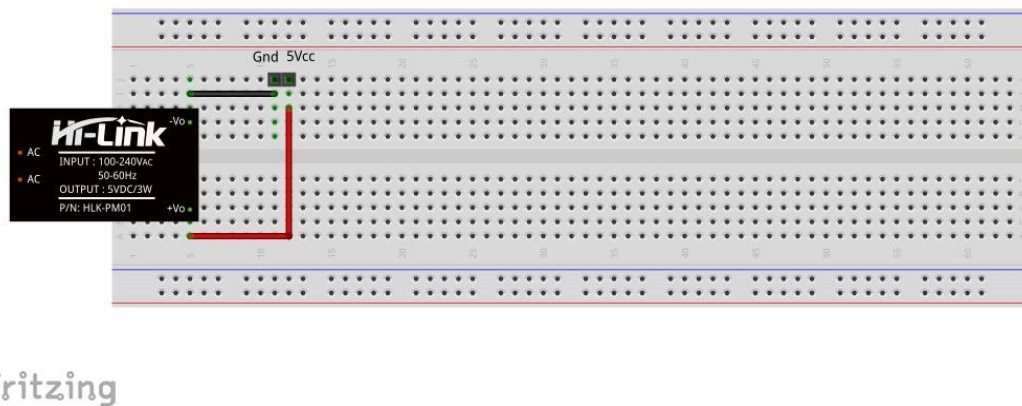
Inicialmente todo o circuito foi montado usando o software *Fritzing* para validar e visualizar a montagem. Dessa forma, o circuito foi dividido em duas partes, sendo uma composta pelos componentes principais exibido na Figura 21, e a outra parte composta de uma fonte de alimentação mostrada na Figura 22.

Figura 21: Placa 01 simulada com *Fritzing*.



Fonte: Autor.

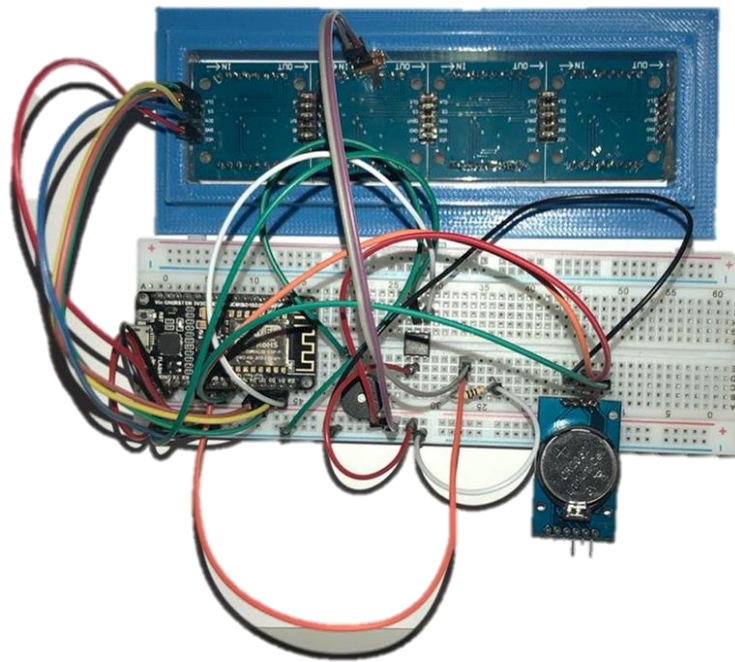
Figura 22: Placa 02 simulada com *Fritzing*.



Fonte: Autor.

A partir do esquema elétrico foi possível montar a primeira versão do circuito em um protoboard para realizar os testes iniciais e simular o funcionamento do sistema como ilustra na Figura 23.

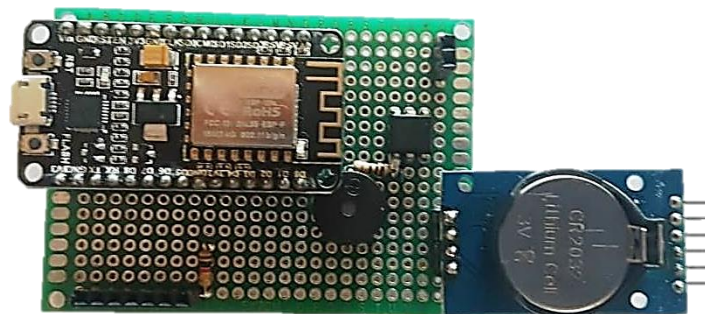
**Figura 23: Montagem do projeto elétrico em protoboard.**



Fonte: Autor.

Em sequência, configurou-se o circuito para duas placas dupla face perfuradas de 4x6cm. A primeira placa comporta o módulo embarcado ESP8266, o optoacoplador, buzzer, módulo RTC, um resistor de 1K $\Omega$  e um de 100 $\Omega$ , como mostrado na Figura 24.

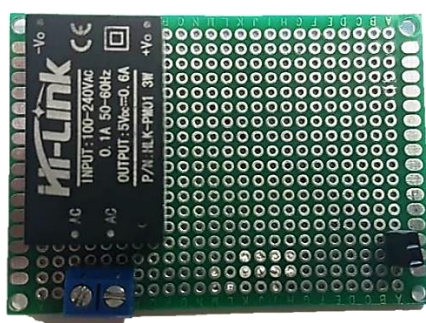
**Figura 24: Placa 01.**



Fonte: Autor.

A segunda placa comporta a mini fonte Hi-Link, responsável por transformar a tensão de 110V da rede elétrica para 5V exibida na Figura 25.

Figura 25: Placa 02.



Fonte: Autor.

### 3.2.2 Levantamento de custo

A Tabela 4 apresenta o orçamento obtido de todos os componentes que integram o circuito eletrônico.

Tabela 4: Orçamento dos componentes do projeto.

Material	Quantidade	Fornecedor	Custo (R\$)
Matriz de LED	02	Mercado Livre	138,00
NodeMCU ESP8266	02	Aliexpress	21,62
Fonte Hi-Link	02	Aliexpress	16,00
Placa perfurada 4x6cm	02	Arduino Manaus	10,00
Optoacoplador	02	Aliexpress	2,54
Buzzer	02	Arduino Manaus	2,00
RTC DS3231	02	Arduino Manaus	30,00
PLA	700g	Aliexpress	52,00
Resistores	04	Mercado Livre	0,40
Chave comutadora	02	Mercado Livre	2,00
<b>Total</b>			<b>274,56</b>

Fonte: Autor.

Após a montagem do circuito eletrônico e do estudo da tabela de custos, a etapa seguinte diz respeito a programação da NodeMCU para o controle dos periféricos do sistema.

### 3.3 PROGRAMAÇÃO DA NODEMCU

A programação da placa NodeMCU é classificada em duas fases: configurações iniciais e linhas de comando em loop.

#### 3.3.1 Configurações iniciais

As configurações iniciais tem relação na declaração de variáveis, importação de bibliotecas e configuração da rede *WiFi*. Fez-se necessária a importação de quatro bibliotecas e a utilização de cinco bibliotecas nativas da própria IDE do Arduino.

Dessa forma, as bibliotecas importadas foram:

- *WiFiManager.h* – Esta biblioteca foi utilizada como gerenciador de conexões wireless;
- *MAX7219\_Dot\_Matrix.h* – Fez-se necessário o uso da biblioteca adicional da Matriz de leds MAX7219, cuja função é facilitar o envio de mensagens utilizando a NodeMCU através de uma interface *web*.
- *bitBangedSPI.h* - O bit banging está criando toda a série de pulsos em software, em vez de depender de um hardware dentro do microcontrolador. O arduino têm um SPI de hardware, e tudo o que precisa fazer é escrever um byte no registro de saída, e o controlador SPI deslocará os dados para fora e, ao mesmo tempo, receberá os dados do escravo, assim pode-se obter uma interrupção quando a transferência for concluída e ler os dados recebidos.
- *RtcDS1307.h* – Utilizou-se essa biblioteca para facilitar o uso do RTC no sistema, que exibirá o horário quando a matriz estiver em modo *standby*.

As bibliotecas nativas são apresentadas a seguir.

- ESP8266WiFi.h – Biblioteca responsável pelo uso das funções associadas à conexão do microcontrolador a uma rede WiFi;
- ESP8266WebServer.h – Cria um servidor *web*, suporta apenas um cliente simultâneo;
- ESP8266mDNS.h - O mDNS usa nomes de domínio com o sufixo local, por exemplo, `http://esp8266.local`. Foi necessário enviar uma solicitação para um nome de domínio que termine em `.local`, conseqüentemente, ele envia uma consulta multicast para todos os outros dispositivos na LAN que ofereçam suporte ao mDNS, solicitando que o dispositivo com esse nome de domínio específico se identifique. O dispositivo com o nome certo responderá com outro multicast e enviará seu endereço IP. No momento que o computador conhecer o endereço IP do dispositivo, ele pode enviar solicitações normais.
- Wire.h – Esta biblioteca permitiu a comunicação com o dispositivo I2C.
- SPI.h – Foi utilizada para comunicação serial com a matriz de LED.

Após a exportação das bibliotecas necessárias na configuração inicial, são empregadas as variáveis do sistema.

### **3.3.2 Linhas de comando em *loop*.**

A programação que está localizada na área de execução do código em *loop* ocorre sempre que o dispositivo estiver ligado. Dessa forma, os comandos presentes na função `loop()` estão diretamente ligados ao acionamento do Buzzer, visualização da hora utilizando o RTC e envio das mensagens para a matriz de leds. As funções para o controle destes componentes do sistema, são exibidos a seguir.

#### **3.3.2.1 Função `intervalo()`;**

A função `intervalo` foi criada para facilitar o uso do Buzzer, podendo assim, criar intervalos por tempos determinados, tanto de quanto tempo o Buzzer estará

acionado e por quanto tempo não, bem como no intervalo do mesmo. Esta função é mostrada na Figura 26.

**Figura 26: Função intervalo();**

```
int intervalo( long TempAlto,long TempBaixo ,long Intervalo,long Ref =0){
    long ajuste = Ref % (TempAlto + TempBaixo);
    long resto =(millis() + TempAlto + TempBaixo - ajuste - Intervalo) % (TempAlto + TempBaixo);
    return (resto < TempAlto ? HIGH : LOW);
}
```

Fonte: Autor.

A variável TempAlto corresponde a quanto tempo o Buzzer soará em volumes altos, a variável TempBaixo corresponde a quanto tempo o Buzzer soará em volumes baixos, e a variável Intervalo corresponde ao tempo de intervalo entre os acionamentos, a Referência sempre será zero.

### 3.3.2.2 Função Buzzer();

A função buzzer aciona o Buzzer propriamente dito. A função responsável é vista na Figura 27.

**Figura 27: Função Buzzer();**

```
void Buzzer (){
    digitalWrite(buzzer,intervalo(1000, 3000, 2000, 0));
    digitalWrite(led,intervalo(2000, 2000, 2000, 0));
}
```

Fonte: Autor.

### 3.3.2.3 Função printDateTime();

Para a exibição de tempo, relacionado a horas, minutos e segundos durante o dia e a noite, período que tem alunos no campus do IFAM, é empregada a função responsável pelo funcionamento e exibição de horário em tempo real. A Figura 28 ilustra o código usado no programa.

**Figura 28: Função printDateTime();**

```
void printDateTime(const RtcDateTime& dt)
{
    char datestring[20];
    snprintf_P(datestring,
               sizeof(datestring),
               PSTR("%02u:%02u:%02u"),
               dt.Hour(),
               dt.Minute(),
               dt.Second());

    Serial.print(datestring);
    Serial.println(); // Teste
}
```

Fonte: Autor.

### 3.3.2.4 Função handleSetCommand();

Para facilitar a compreensão nas linhas de código do programa, a linha *function* foi direcionada para baixo de *response*, ressaltando que o código de comando precisa estar em apenas uma linha para o seu completo funcionamento. A Figura 29 exibe a função responsável por verificar se há ou não mensagens a serem enviadas para o dispositivo Vision.

**Figura 29: Função handleSetCommand();**

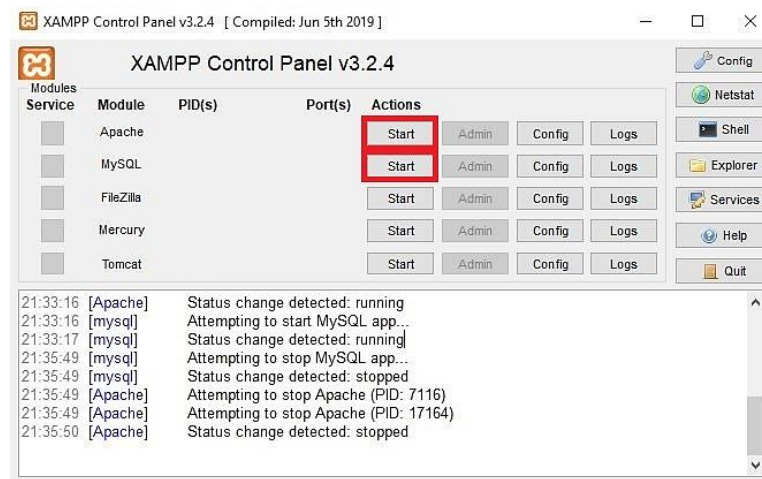
```
82 void handleSetCommand() {
83     String response = "";
84     if (server.args() == 0) {
85         response = "Sem parametro";
86     } else {
87         if (server.argName(0) == "delay") {
88             MOVE_INTERVAL = server.arg("delay").toInt();
89             response = "Tempo de mensagem: ";
90             response += MOVE_INTERVAL;
91         }
92         if (server.argName(0) == "intensity") {
93             intensity = (byte)server.arg("intensity").toInt();
94             response = "Intensidade do brilho";
95             display.setIntensity(intensity);
96             response += intensity;
97         }
98         if (server.argName(0) == "text") {
99             server.arg("text").toCharArray(message, msglen);
100             messageOffset = - chips * 8;;
101             response = "Mensagem enviada teste";
102             //Buzzer ();
103             contMessage= 0;
104
105         }
106         if (response == "" ) {
107             response = "parametro invalido";
108         }
109     }
110     response = "<html><head><meta http-equiv=\"refresh\" content=\"3 ; url=/\> <script language='javascript'>
111     function t(){ window.close(); } </script> </head><body onload='t();'><p>\" + response + \"</p></body></html>\";
112     server.send(200, \"text/html\", response);
113 }
```

Fonte: Autor.

### 3.4 CRIAÇÃO DO BANCO DE DADOS

Este tópico demonstra o procedimento de criação do banco de dados e o preenchimento das tabelas. Para que o computador se torne um servidor do sistema é necessário pressionar o botão de *Start* das opções Apache e MySQL ao abrir o XAMPP conforme a Figura 30.

**Figura 30: Painel de controle do XAMPP.**



Fonte: Autor.

Em seguida, deve-se iniciar um navegador *web* e acessar a URL: <http://localhost/phpmyadmin/>. No lado esquerdo da página, é exibida uma lista de exemplos de banco de dados. Para a criação de um novo banco de dados, basta clicar em *novo*, digitar um nome da escolha do desenvolvedor no formato `utf8_general_ci` como exibe a Figura 31.

**Figura 31: Página *phpMyAdmin*.**



Fonte: Autor.

Após gerar a base de dados, é carregada uma nova página para a criação de uma tabela. No projeto, é necessário dar origem a duas tabelas, uma para os dispositivos, chamada de *device*, e outra de usuário com o nome *user*.

A tabela do banco de dados *device* possui três campos definidos da seguinte forma:

- *id*: é o índice que indica um número de série do equipamento do tipo inteiro, podendo ter até 2048 dispositivos;
- *ip*: é o campo que será preenchido no formulário com o valor de IP de cada protótipo;
- *local*: é completado com o número da sala de alocação.

A Figura 32 mostra o formato de preenchimento da tabela do banco de dados *device*.

**Figura 32: Tabela *device*.**

Nome	Tipo	Tamanho/Valores*	Predefinido	Agrupamento (Collation)	Atributos	Nulo	Índice	A_
id	INT	11	None			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
ip	TEXT		None			<input type="checkbox"/>	---	<input type="checkbox"/>
local	INT	11	None			<input type="checkbox"/>	---	<input type="checkbox"/>

Comentários da tabela:

Collation: utf8\_general\_ci

Motor de armazenamento: InnoDB

PARTITION definition:

Partition by:  ( Expression or column list )

Partitions:

Pré-visualizar SQL   Guarda

Fonte: Autor.

A tabela nomeada *user* é preenchida com três campos, onde cada coluna é definida como se segue:

- id: índice que será o número do usuário;
- login: campo que terá o nome do cliente;
- password: lacuna que deve ser completada com uma senha definida pelo usuário.

A Figura 33 exibe o formato de preenchimento da tabela *user*.

**Figura 33: Tabela *user*.**

The screenshot shows a database management interface for a table named 'user'. The table structure is defined as follows:

Nome	Tipo	Tamanho/Valores*	Predefinido	Agrupamento (Collation)	Atributos	Nulo	Índice	A_I
id	INT	11	None			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
login	TEXT		None			<input type="checkbox"/>	---	<input type="checkbox"/>
password	TEXT		None			<input type="checkbox"/>	---	<input type="checkbox"/>

Additional settings shown in the interface include:

- Comentários da tabela: (empty)
- Collation: utf8\_general\_ci
- Motor de armazenamento: InnoDB
- PARTITION definition: (empty)

Fonte: Autor.

### 3.5 PÁGINAS WEB

A geração da interface *web* é baseada na linguagem de programação PHP, linguagem de front-end CSS, linguagem de banco de dados SQL e a linguagem de marcação HTML, sendo esta uma das principais propriedades do projeto, pois é utilizada na construção de páginas *web*. Desse modo, quando houver uma nova escrita de palavras na página desenvolvida, a mesma poderá ser visualizada pelos indivíduos presentes nas salas de aula.

O alarme sonoro deve ser acionado no instante que o chip controlador ESP8266 receber um sinal interpretado das informações descritas na página

web. O site possui uma estrutura de manipulação simples e fácil, eliminando qualquer dificuldade que o usuário possa enfrentar.

Para desenvolvimento da página web foi utilizado às linguagens PHP, HTML, SQL, CSS como dito anteriormente, através do editor de texto *Sublime Text*. Para realizar a interação do usuário com o servidor foi desenvolvida a interface web, composta por quatro páginas, *index*, *cadastro*, *dispositivos* e *principal*. O acesso à página inicial se dá através do link: <http://localhost/novidade/>.

### 3.5.1 Index

Nesta página o usuário poderá efetuar o login no sistema, inserindo seu nome e senha, ocorrendo assim à validação dos dados inseridos. A Figura 34 mostra a página index.

**Figura 34: Página de login.**



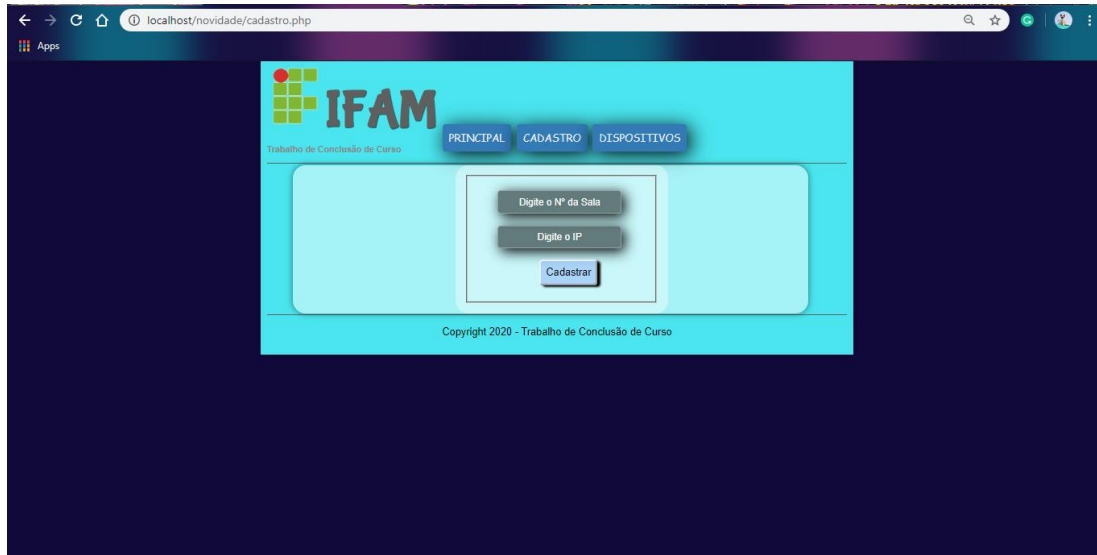
Fonte: Autor.

### 3.5.2 Cadastro

Para cadastrar é necessário preencher dois campos, um é relacionado ao número da sala em que o dispositivo será alocado, e o IP fornecido pela rede. Em seguida pressiona-se o botão “Cadastrar”. Nesta função é realizada uma verificação para que não haja duplicidade de informações dos dispositivos. Caso

algum dos campos não seja preenchido o cadastro não será efetuado. A página é exibida na Figura 35.

**Figura 35: Página Cadastro.**

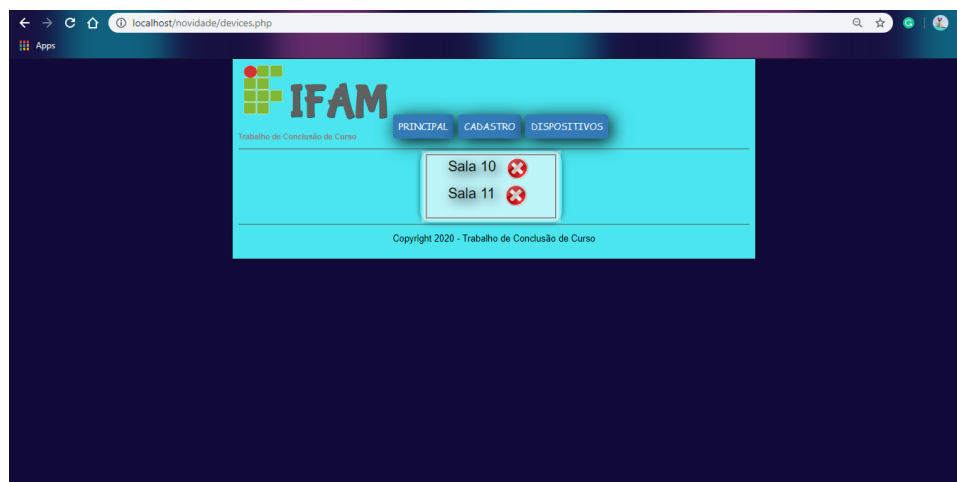


Fonte: Autor.

### 3.5.3 Dispositivos

Na página de dispositivos é possível visualizar os equipamentos que já foram cadastrados, e também os excluir do sistema. A página de dispositivos é mostrada na Figura 36.

**Figura 36: Página Dispositivos.**



Fonte: Autor.

### 3.5.4 Principal

A página principal possui um campo para escrever a mensagem que se deseja enviar. As caixas de *checkbox* devem ser selecionadas para as salas de envio da mensagem. Pode ser selecionada uma sala ou mais. A Figura 37 mostra a página principal.

Figura 37: Página principal.



Fonte: Autor.

## 3.6 PROJETO MECÂNICO

Com a finalidade de tornar o dispositivo visível, preservado, com uma boa aparência e com as placas do circuito protegido, foi usado a impressora 3D do modelo *Cloner DH+* para imprimir a peça em formato de caixa, desenvolvido especificadamente para este projeto.

Essa impressora trabalha com o preenchimento de camadas até formar o modelo final usando um material chamado de PLA (poliácido láctico) com filamento 1,75 mm. O software para utilização da impressora 3D foi o *ClonerGen3D*, fornecido pelo próprio fabricante da impressora. A Figura 38 mostra a impressora utilizada para a confecção da peça.

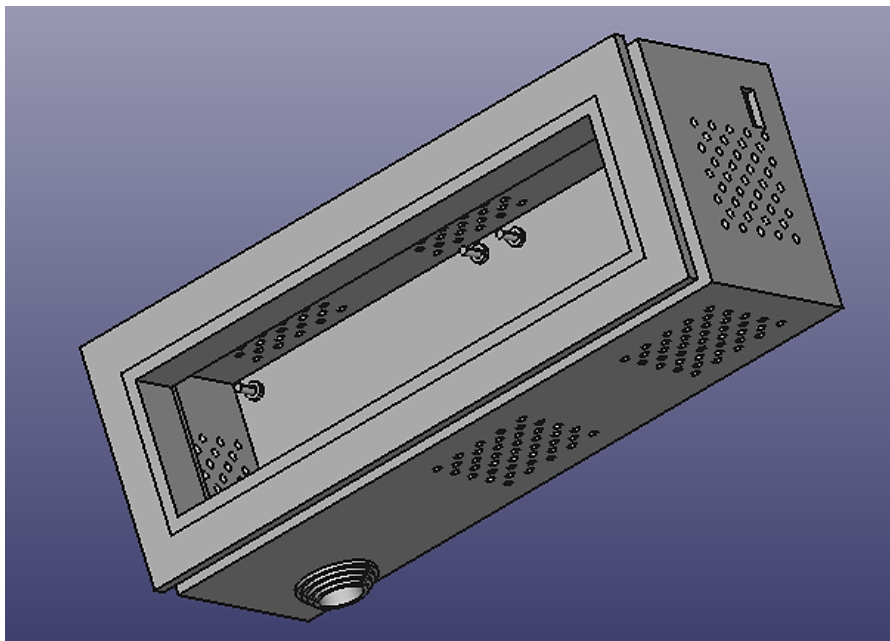
**Figura 38: Modelo de impressora Cloner DH+.**



Fonte: <https://3dclonersp.wixsite.com/3dclonersp/3dclonersp>.

A modelagem 3D do protótipo foi criada utilizando o software livre *AutoCAD*. O desenvolvimento do layout possibilitou a visualização do posicionamento dos componentes que integram o dispositivo principal. Os furos destacados nas laterais da caixa são usados para ventilação e para a não obstrução do sinal WiFi. A Figura 39 mostra o modelo 3D.

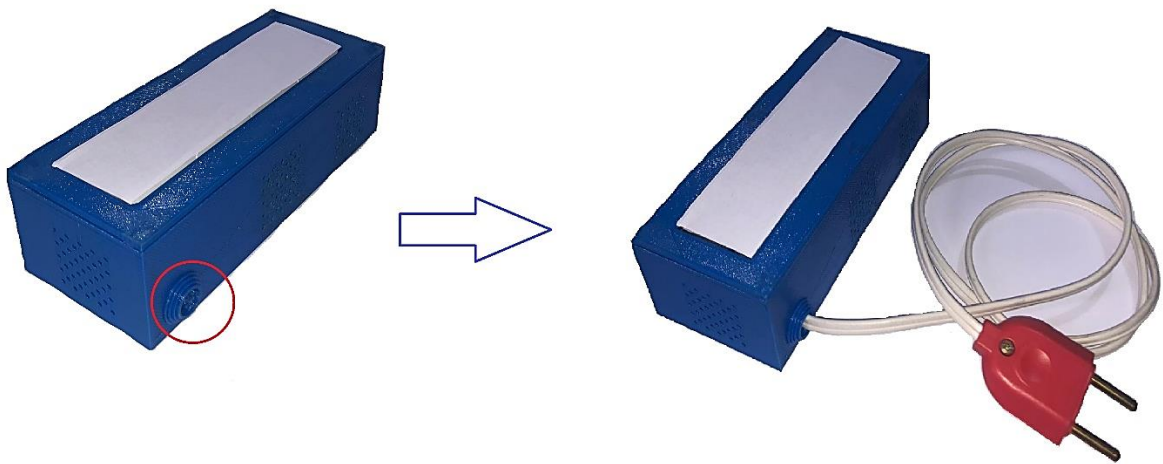
**Figura 39: Modelagem 3D da caixa.**



Fonte: Autor.

Após a impressão da peça por completo, foi utilizado uma lixa para limpar e polir o material, retirando eventuais defeitos ou desnível. A abertura destacada na Figura 40 é reservado para o cabo de força que servirá para a passagem da corrente elétrica.

**Figura 40: Caixa com o cabo de força.**



Fonte: Autor.

### 3.7 OPERAÇÃO DO SISTEMA

Este tópico trata etapa por etapa do processo de inicialização do Vision, para uma correta instalação e funcionamento do mesmo. Dessa forma, as seguintes etapas devem ser executadas:

1. Ligar o dispositivo Vision e o roteador na tomada. Com o Vision energizado os Leds da matriz irão acender, como mostrado na Figura 41.

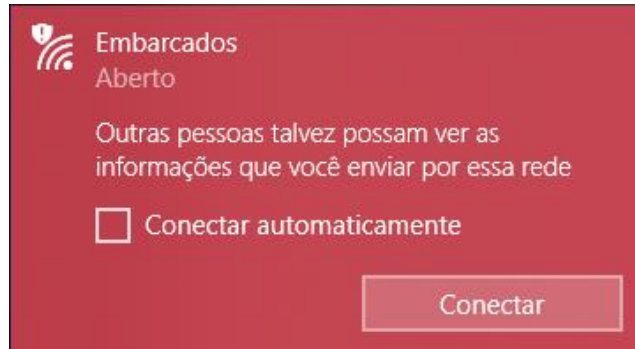
**Figura 41: Dispositivo Vision ligado.**



Fonte: Autor.

2. Com o Vision energizado ele gerará um ponto de acesso chamado “Embarcados”, como exibido na Figura 42.

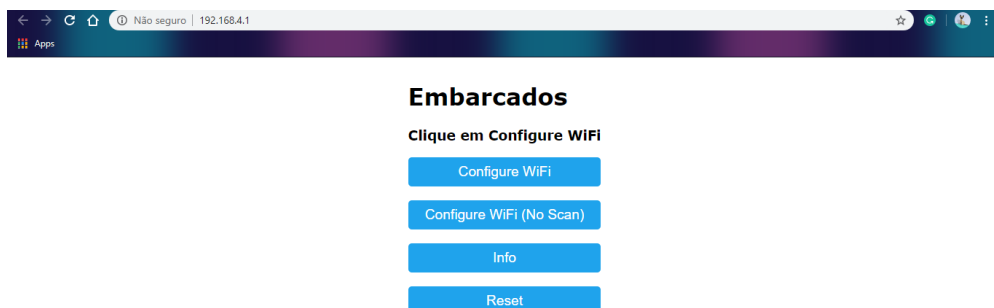
**Figura 42: Rede ESP.**



Fonte: Autor.

3. Depois de conectado, o usuário deverá digitar em um navegador *web* o IP e acessar Configure WiFi ilustrado na Figura 43.

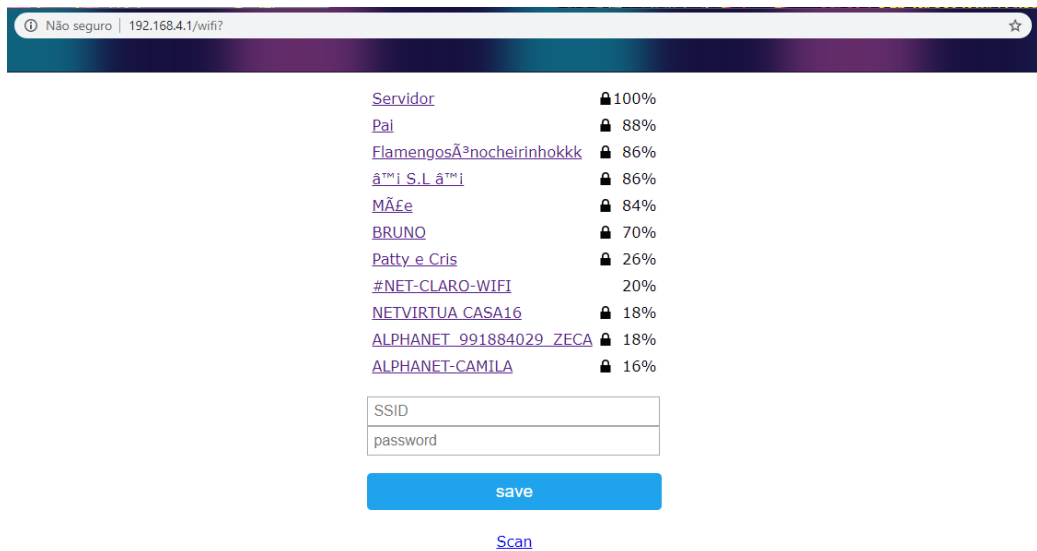
**Figura 43: Acesso a rede embarcados.**



Fonte: Autor.

4. Na opção de configurar WiFi deve-se conectar à rede gerada pelo roteador “Servidor”. A senha para este é “12345678”. A Figura 44 mostra esta opção.

Figura 44: Rede Servidor.

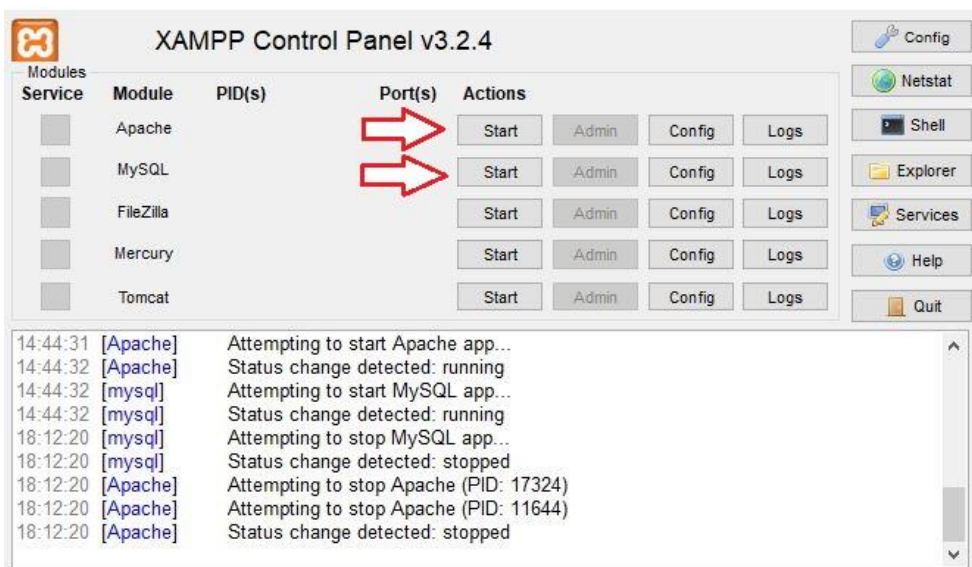


Fonte: Autor.

5. Mudar a rede WiFi do computador para a rede gerada pelo roteador denominada de “Servidor”.

6. Abrir o software XAMPP e pressionar Start nas opções Apache e MySQL, ilustrado na Figura 45. Depois de pressionado, os ícones em destaque ficam na cor verde.

Figura 45: Servidor XAMPP.



Fonte: Autor.

7. Acessar em um navegador *web* o link: <http://localhost/novidade/>, como visto na Figura 46.

**Figura 46: Página web.**



Fonte: Autor.

8. Ao abrir a página inicial deve-se inserir o usuário "adm", a senha "123" e pressionar o botão Login, para ter acesso às demais funções. A Figura 47 mostra como o usuário deve preencher as tabelas de login.

**Figura 47: Página de login.**



Fonte: Autor.

9. Após entrar com usuário e senha, uma nova página é carregada ilustrada na Figura 48. Na primeira caixa deve-se colocar o número da sala em que se encontra o dispositivo Vision e logo abaixo o IP a qual é exibido na tela da matriz de led. Deslocando o botão que se encontra na lateral do dispositivo aparecerá o IP, que deve ser inserido na segunda caixa da página. Ao finalizar o preenchimento, apertar o botão cadastrar. Caso algum dos campos deste formulário não sejam preenchidos, ou tenham conflito com número de sala e endereço de IP, o dispositivo não será cadastrado.

**Figura 48: Efetuando cadastro.**

IFAM  
Trabalho de Conclusão de Curso

PRINCIPAL CADASTRO DISPOSITIVOS

Digite o Nº da Sala

Digite o IP

Cadastrar

Copyright 2020 - Trabalho de Conclusão de Curso

Fonte: Autor.

10. Deslocando o botão comutador de volta para o modo de operação, o usuário para usar o sistema, deve clicar em “principal” e assim inserir a mensagem no campo indicado. A transmissão das mensagens para os dispositivos Vision deve ser marcada no *checkbox* da sala desejada e depois pressionar o botão Enviar como é exibido na Figura 49.

**Figura 49: Enviando mensagens.**



Fonte: Autor.

11.O texto escrito na caixa de mensagens irá aparecer na matriz de Led do Vision selecionado como visto na Figura 50.

**Figura 50: Vision recebendo mensagem.**

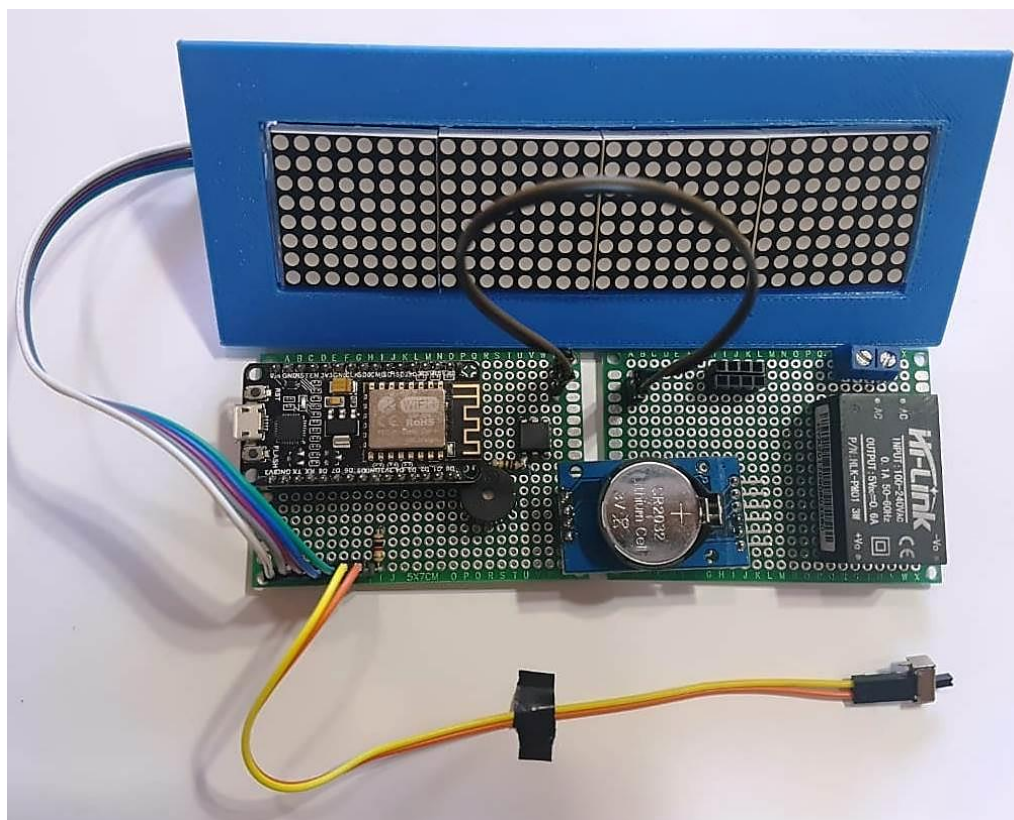


Fonte: Autor.

## 4 RESULTADOS

Depois de realizado os testes de funcionalidade do circuito, os componentes foram aproximados um ao outro, evitando o uso de jumpers, pois dessa forma diminui possíveis interferências nos componentes, que pode afetar diretamente no resultado final. A Figura 51 mostra a versão final do circuito eletrônico montado nas placas.

Figura 51: Versão final do circuito eletrônico do sistema.



Fonte: Autor.

A caixa que comporta todos os componentes foi impresso na cor azul com um formato simples e descomplicado, possuindo orifícios para a circulação de ar, melhorando o sinal WiFi, bem como para não abafar o barulho do alarme sonoro quando tocar.

Além disso, existem mais duas aberturas na caixa, uma para a entrada do fio da rede elétrica e o outro para a chave de operação. A Figura 52 ilustra a

finalização da caixa fechada com todo o circuito ajustado interiormente e parafusada com seis pregos pequenos.

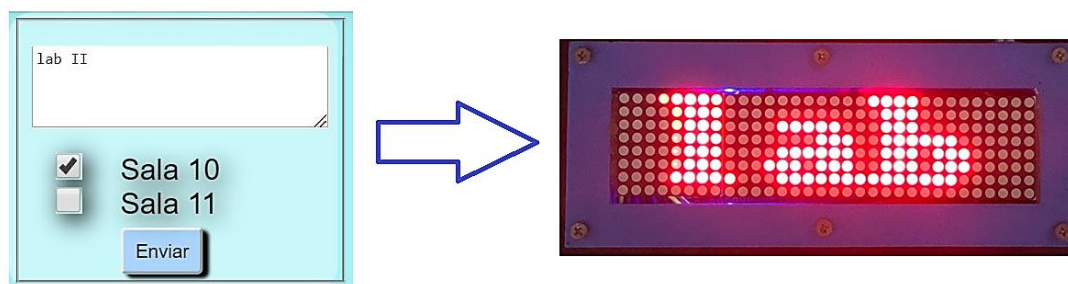
**Figura 52: Finalizada do dispositivo com a caixa fechada.**



Fonte: Autor.

Como apresentado anteriormente, a comunicação através da interface *web* e a matriz são usadas principalmente para que os usuários, alunos ou professores, recebam notificações de avisos ou mensagens importantes no ambiente do Instituto Federal do Amazonas. Uma ilustração do funcionamento é apresentada na Figura 53.

**Figura 53: Matriz exibindo mensagens da interface *web*.**



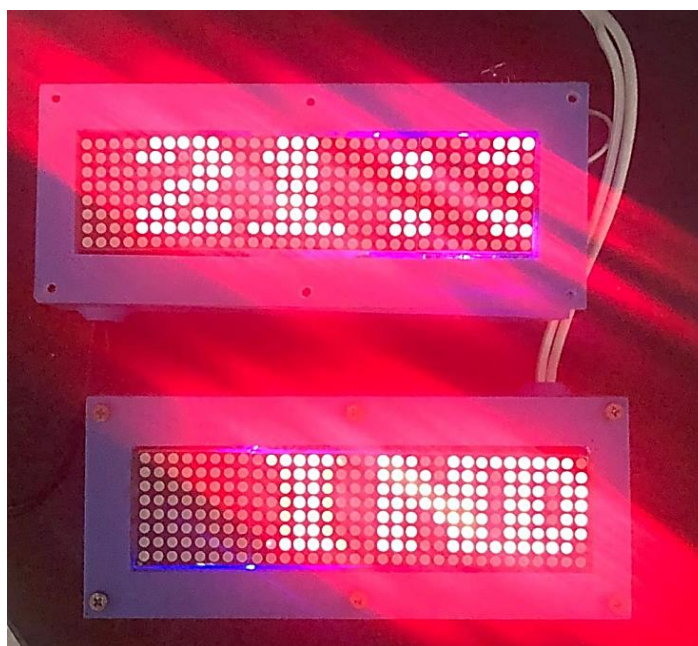
Fonte: Autor.

Dessa forma, a formatação final da interface web apresentou resultado satisfatório, podendo ser acessado rapidamente usando uma rede local do IFAM. As páginas desenvolvidas garantem uma interação que possibilita cadastrar, escolher o dispositivo e enviar informes em tempo real.

Além disso, a página apresenta uma aparência fácil e simples, possibilitando qualquer usuário manipular os dados. O alarme sonoro funcionou em todos os casos de testes, que ocorrem quando o dispositivo é energizado ao ser ligado na tomada e quando é recebida uma mensagem via web.

Quando não há comunicado a ser enviado, o dispositivo Vision permanece em modo de espera, exibindo o horário local enquanto estiver ligado na energia elétrica. De modo geral, a hora deixa de ser exibida para apresentar a nova mensagem e após um tempo programado volta a ser exibido novamente devido ao módulo RTC. Os dois dispositivos são exibidos na Figura 54 com as suas respectivas salas cadastradas anteriormente.

**Figura 54: Dois dispositivos de comunicação no IFAM.**



Fonte: Autor.

Portanto, o sistema apresentou uma alta confiabilidade, constatada a partir dos testes realizados durante o período de validação do sistema, contemplando os objetivos específicos alcançados.

## 5 CONSIDERAÇÕES FINAIS

Foi realizado testes para verificar o funcionamento do sistema, tanto no envio de mensagens para dispositivos de maneira individual quanto para vários dispositivos simultaneamente. Dessa forma, obteve-se êxito no envio das mesmas, tanto no tempo de resposta quanto no desenvolvimento de uma interface simples e intuitiva para que o usuário consiga utilizar o sistema facilmente.

As mensagens foram correspondidas exatamente como na escrita pelo usuário dentro da página *web*. Foi seguido e executado corretamente um padrão de acesso, para que somente os usuários cadastrados no banco de dados tenham acesso quando necessário, evitando a desinformação.

Os softwares usados gratuitamente permitiram um completo estudo para a execução dos testes iniciais e finais, bem como na escolha do material do hardware, sendo útil o levantamento de custo dos materiais, pois resultou em um projeto de baixo custo e prático para quem for executar ou implementar alguma função a mais.

A matriz 32x8 utilizada no projeto possibilitou que as pessoas público-alvo do IFAM olhassem com facilidade o dispositivo, garantindo assim, uma atenção ligada diretamente as informações passadas em tempo real, incluindo um alarme sonoro.

O aviso sonoro foi executado conforme o programado, tocando quatro vezes seguidas ao receber uma nova mensagem. Deste modo, foi importante a execução deste componente, pois alertou os alunos e demais pessoas a quem interessar, visto que o Instituto é grande e possui diversas turmas de ensino fundamental, médio e superior.

A maior dificuldade para o projeto foi trabalhar com a interface *web* utilizando banco de dados e também na maneira de simplificar a interface para o usuário. Outra dificuldade encontrada foi na comunicação do RTC com o sistema, onde este erro foi corrigido reduzindo o caminho de comunicação, retirando os jumpers e ligando o RTC diretamente em seu respectivo borne.

O dispositivo de comunicação funcionou satisfatoriamente, suprimindo os requisitos estabelecidos no trabalho. Sendo assim, os dois sistemas utilizados

exibiram as notificações de comunicados, sem a necessidade do aluno ou professor se deslocar até a administração responsável.

## 5.1 TRABALHOS FUTUROS

Para trabalhos futuros pode ser aplicado o protocolo MQTT no sistema, para fazer com que o usuário consiga tanto enviar dados quanto receber, criando assim, interação entre os usuários do Instituto. Além disso, o sistema poderá ser utilizado em outras instituições, em diferentes cenários, para avaliarmos a satisfação dos usuários com a apresentação de avisos para todos.

Pode ser implementado para alarmes de campanha em horários programados de intervalo, podendo ser nos horários da merenda ou almoço. Outra ideia para trabalhos futuros pode ser na aplicação de componentes atuais, na substituição de componentes eletrônicos para versões mais modernas.

## 6 REFERÊNCIAS

A IMPORTÂNCIA DE TER UMA PÁGINA NA WEB BEM DESENVOLVIDA. Studio AleDesigner, 2011. Disponível em: <<https://studioaledesigner.wordpress.com/2011/02/23/a-importancia-de-ter-uma-pag-na-web-ligada-as-redes-sociais/>>. Acesso em: 20 dez. 2019.

AI-THINKER. **ESP-12E WiFi module**. Datasheet do componente versão 1.0.

ARDUNAUT. **NodeMCU ESP-12**. Development kit V1.0.

ARIANE, G. **O que é CSS? Guia Básico para iniciantes**. Hostinger Blog. 2019. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css/>>. Acesso em: 02 jan. 2020.

ASHTON, Kevin. **That “Internet of Things” Thing**. RFID Journal, 22 jun. 2009. Disponível em: <<http://www.itrco.jp/libraries/RFIDjournal-That%20Internet%20of%20Things%20Thing.pdf>>. Acesso em: 14 dez. 2019.

BARRETO, M.V.S. **Curso de Linguagem PHP**. Comitê de Incentivo à Produção de Software Gratuito e Alternativo – CIPSGA. Universidade Federal de Sergipe, Centro de Ciências Exatas e Tecnologia do Departamento de Estatística e Informática. 2000.

BARROS, Daiane M. L; MATOS, Nailton S.; **A importância da comunicação organizacional interna e dos feedbacks gerenciais**. Revista de Administração |FW| v. 13|n. 23| p.3-20|ago. 2015. Disponível em: <<http://www.revistas.fw.uri.br/index.php/revistadeadm/article/viewFile/1553/1961>>. Acesso em: 15 jan. 2020.

CALDAS, M.M.M.P. **Automação residencial: Controlando dispositivos remotamente com o microcontrolador nodeMCU**. Departamento de Engenharia Elétrica. Pontifícia Universidade Católica do Rio de Janeiro. 2018.

DEIMAR, Gabriel; SOBREIRA, Demóstenes; LIMA, Welton. **Internet das coisas na Educação**. Revista Tecnologias em Projeção, v 8, nº2, ano 2017. p. 78.

FERNANDES, R. **Apostila Fundamento de Banco de Dados**. Núcleo de educação a distância. Nrsystem. 2017. Disponível em: <[http://www.nrsystem.com.br/Fund\\_Banco\\_Dados.pdf](http://www.nrsystem.com.br/Fund_Banco_Dados.pdf)>. Acesso em: 02 jan. 2020.

FERREIRA, E. A; JUNIOR, M. A. F; **Aplicação da plataforma Arduino para a determinação de parâmetros atmosféricos e ambientais**. Ensaios & Diálogos, Rio Claro, v. 10, n. 1, p. 27-47, jan./dez. 2017.

HOFFMAN, J. Mastering Arduino. 1a. ed. Birmingham: Packt Publishing, 2018.

LEITE Emiliano; MARTINS Paulo; URSINI Edson. **A internet das coisas (IoT): Tecnologias e Aplicações**. Brazilian Technology Symposium, 2017. Disponível

em: <<http://lcv.fee.unicamp.br/images/BTSym-17/Papers/76926.pdf>>. Acesso em: 15 dez. 2019.

LOSSIO, R. **O que é e para que serve o XAMPP?**. Oráculo TI. Tecnologia, Informação e Cursos. 2019. Disponível em: <>. Acesso em: 03 jan. 2020.

MOTA, N.A; VALLE, R.M. **Apostila de AutoCAD**. Módulo básico. Universidade Federal de Juiz de Fora. Faculdade de Engenharia. 2011.

MURTA, G. **NodeMCU – ESP12: Guia completo – Introdução (Parte 1)**. Blog Eletrogate. 2018. Disponível em: <<https://blog.eletrogate.com/?s=nodemcu>>. Acesso em: 01 jan. 2020.

OLIVEIRA, R. R. **Uso do microcontrolador ESP8266 para automação residencial**. Universidade Federal do Rio de Janeiro. Escola Politécnica. 2017.

PEDROSO, R. P. **Apostila de HTML**. Programa de Educação Tutorial. Curso de Engenharia de Telecomunicações. Universidade Federal Fluminense. Niterói, Rio de Janeiro. 2007.

PEREZ, A.L.F; DARÓS, R.R. **Programação em Arduino – Módulo básico**. Oficina de Robótica. Universidade Federal de Santa Catarina. 2013

.

SANTOS, Bruno P. et. al. **Internet das coisas: da teoria à prática**. Minicursos SBRC-Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2016. Disponível em: <<http://www.sbrc2016.ufba.br/downloads/anais/MinicursosSBRC2016.pdf>>. Acesso em: 14 dez. 2019.

SANTOS, Raquel.; GOMES, Henriette.; DUARTE Emeida.; **Processo dialógico entre bibliotecários e usuários: reflexão em torno da utilização dos dispositivos de comunicação da web social**. Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação, vol. 21, núm. 45, enero-abril, 2016, pp. 115-129. Universidade Federal de Santa Catarina Florianópolis, Brasil.

SILVA, R. M; **Apostila de programação para web: HTML e CSS**. Instituto Federal de Educação, Ciência e Tecnologia da Bahia Campus Salvador. 2013.