

**CENTRO PAULA SOUZA**

**FATEC SANTO ANDRÉ**

**Mecatrônica Industrial**

**Mauricio Santana Barbosa**

**Diego Munhoz Rodrigues**

**AUTOMAÇÃO RESIDENCIAL POR CONTROLE DE VOZ**

**Santo André**

**2023**

**Mauricio Santana Barbosa**  
**Diego Munhoz Rodrigues**

## **AUTOMAÇÃO RESIDENCIAL POR CONTROLE DE VOZ**

Trabalho de Conclusão de Curso apresentado ao Curso de Tecnologia em Mecatrônica Industrial da Fatec Santo André como requisito parcial para a obtenção do título de Tecnólogo em Mecatrônica Industrial.

Prof. Orientador: Dr. Edson Caoru Kitani

**Santo André**  
**2023**

## FICHA CATALOGRÁFICA

B238a

Barbosa, Mauricio Santana

Automação residencial por controle de voz / Mauricio Santana  
Barbosa, Diego Munhoz Rodrigues. - Santo André, 2023. – 75f.  
il.

Trabalho de Conclusão de Curso – FATEC Santo André.  
Curso de Tecnologia em Mecatrônica Industrial, 2023.

Orientador: Prof. Dr. Edson Caoru Kitani

1. Mecatrônica. 2. Automação residencial. 3. Internet das coisas. 4. Projeto. 5. Tecnologia. 6. Acessibilidade. 7. Segurança. 8. Controle de processos. 9. Microcontrolador. 10. Casas inteligentes. I. Rodrigues, Diego Munhoz. II. Automação residencial por controle de voz.

629.89

**LISTA DE PRESENÇA**

Santo André, 12 DE JUNHO DE 2023.

LISTA DE PRESENÇA REFERENTE À APRESENTAÇÃO DO  
TRABALHO DE CONCLUSÃO DE CURSO COM O TEMA:  
“AUTOMAÇÃO RESIDENCIAL POR CONTROLE DE VOZ” DOS  
ALUNOS DO 6º SEMESTRE DESTA U.E.

**BANCA**PRESIDENTE:  
PROF. EDSON CAORU KITANI \_\_\_\_\_MEMBROS:  
PROF. PAULO TETSUO HOASHI \_\_\_\_\_

PROF. FERNANDO GARUP DALBO \_\_\_\_\_

**ALUNOS:**

DIEGO MUNHOZ RODRIGUES \_\_\_\_\_

MAURÍCIO SANTANA BARBOSA \_\_\_\_\_

Dedicamos este trabalho aos professores,  
amigos, familiares e todos que acreditaram  
em nosso potencial e apoiaram nossas  
escolhas

## **AGRADECIMENTOS**

Agradeço à grande ajuda e incentivo de nossos familiares e mestres professores por todos os ensinamentos durante todo o curso.

Ao Prof. Dr. Edson Caoru Kitani por suas orientações e correções durante a realização do projeto e ao professor Fernando Garup Dalbo pela ajuda e incentivos constantes ao longo da execução e organização de todo o documento.

“Talvez não tenha conseguido fazer o melhor, mas lutei para que o melhor fosse feito. Não sou o que deveria ser, mas GRAÇAS à DEUS, não sou o que era antes”.

**MARTIN LUTHER KING**

## RESUMO

Casas Inteligentes já são uma realidade e existem várias aplicações simples que conectam alguns dispositivos elétricos aos assistentes digitais. E com o auxílio de assistentes virtuais acessíveis em todo telefone celular, ou em meio físico, como o Google Assistente, a “Siri” e a Amazon Alexa, é possível controlar equipamentos que sejam compatíveis com redes Wireless e a tecnologia da Internet das Coisas (IoT). Com esse avanço tecnológico controla-se processos de uma residência utilizando apenas a voz e o seu telefone celular como acionadores. Porém, será que sem os assistentes ainda é possível automatizar uma residência de forma eficiente e com baixo custo? Este questionamento incentivou esse projeto, que coloca em prática toda atividade citada acima e testa conhecimentos gerais adquiridos durante o curso. Este projeto se trata de um protótipo que visa a utilização da Automação Residencial de baixo custo para auxiliar na acessibilidade, segurança e conforto de uma residência. Neste projeto utilizou-se o microcontrolador NodeMCU ESP8266, que possibilita controlar e monitorar todos os processos localizados externamente ou internamente da residência, sejam eles luzes, portões, ventilações, sensores de presença ou dispositivos sonoros. O resultado foi um equipamento funcional cujo demonstração foi aplicado numa maquete de uma casa, na qual com um aplicativo no telefone celular é possível controlar as luzes dessa maquete.

**Palavras-Chave:** Automação Residencial. Acessibilidade. Controle de Processos. Microcontrolador NodeMCU ESP8266.

## ABSTRACT

Smart Homes are already a reality, and there are several simple applications that connect some electrical devices to digital assistants. And with the help of virtual assistants accessible in every cell phone, or in physical media, such as Google Assistant, "Siri" and Amazon Alexa, it is possible to control equipment that are compatible with wireless networks and the Internet of Things (IoT) technology. With this technological advance you can control processes in a home using only your voice and your cell phone as triggers. However, is it still possible to automate a residence efficiently and at a low cost without assistants? This questioning encouraged this project, which puts into practice all the activities mentioned above and tests general knowledge acquired during the course. This project is a prototype that aims to use low-cost Residential Automation to help with accessibility, safety, and comfort in a residence. In this project the NodeMCU ESP8266 microcontroller was used, which makes it possible to control and monitor all processes located outside or inside the residence, whether they are lights, gates, ventilations, presence sensors or sound devices. The result was a functional device whose demonstration was applied to a scale model of a house, in which, with a cell phone application, it is possible to control the lights of this model.

**Keywords:** Residential Automation. Accessibility. Process Control. ESP8266 Microcontroller.

## LISTA DE FIGURAS

<b>Figura 1</b> - Dr. E. A. Quade demonstrando o Shoebox.....	20
<b>Figura 2</b> - Camadas da IA.....	22
<b>Figura 3</b> - Comunicação Cliente/Servidor .....	27
<b>Figura 4</b> - Pinagem NodeMCU ESP8266.....	30
<b>Figura 5</b> - Módulo Relé 5v 10A com Optoacoplador .....	31
<b>Figura 6</b> - Lâmpada Incandescente 127VAC 7W .....	32
<b>Figura 7</b> - Divisão de campos .....	34
<b>Figura 8</b> - Programação da tela de login (1) .....	35
<b>Figura 9</b> - Programação da tela de login (2) .....	36
<b>Figura 10</b> - Tela de login APP .....	37
<b>Figura 11</b> - Programação tela principal .....	38
<b>Figura 12</b> - Tela principal APP .....	40
<b>Figura 13</b> - Programação genérica de tela de andar (1) .....	41
<b>Figura 14</b> - Programação genérica da tela de andar (2) .....	42
<b>Figura 15</b> - Programação genérica de tela de andar (3) .....	43
<b>Figura 16</b> - Tela de andar APP .....	44
<b>Figura 17</b> - Fluxograma do funcionamento do APP Inventor.....	45
<b>Figura 18</b> – Inserindo a URL (1) .....	46
<b>Figura 19</b> - Inserindo a URL (2) .....	47
<b>Figura 20</b> - Instalando a placa (1).....	47
<b>Figura 21</b> - Instalando a placa (2).....	48
<b>Figura 22</b> - Incluir biblioteca .....	48
<b>Figura 23</b> - Fluxograma da programação do ESP8266.....	49
<b>Figura 24</b> - Maquete .....	50
<b>Figura 25</b> - Vista frontal da caixa de controle .....	52
<b>Figura 26</b> - Alimentação caixa de controle.....	53
<b>Figura 27</b> - Esquema elétrico .....	54
<b>Figura 28</b> - Conexão ESP8266 / Relé .....	55
<b>Figura 29</b> - Residência Padrão .....	56
<b>Figura 30</b> - Cabos de energia posicionados .....	59
<b>Figura 31</b> - Teste térreo (sala) .....	60

<b>Figura 32</b> - Teste primeiro andar (quarto) .....	60
<b>Figura 33</b> - Teste de funcionamento transformador 127VAC/5VDC .....	61
<b>Figura 34</b> - Primeira comunicação funcional sinalizada pelo serial .....	62
<b>Figura 35</b> - Teste térreo APP .....	62
<b>Figura 36</b> - Teste primeiro andar APP .....	63

## LISTA DE QUADROS

<b>Quadro 1</b> - Especificações protocolo IEEE .....	24
<b>Quadro 2</b> - Interligações.....	55

## LISTA DE TABELAS

<b>Tabela 1</b> - Tabela de Custos .....	51
<b>Tabela 2</b> - Tabela de custo unitário do projeto. ....	57
<b>Tabela 3</b> – Custo padronizado em função de área e potência. ....	58

## LISTA DE ABREVIATURAS E SIGLAS

A	Ampere
APP	Aplicativo
DARPA	<i>Defense Advanced Research Projects Agency</i>
GPIO	<i>General Purpose Input/Output</i>
HTTP	Protocolo de Transferência de Hipertexto
IA	Inteligência Artificial
IBM	<i>International Business Machines Corporation</i>
IDE	Ambiente de Desenvolvimento Integrado
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IoT	Internet das Coisas
IP	Protocolo de Internet
LAN	Rede local
LED	Diodo Emissor de Luz
m	Metros
m <sup>2</sup>	Metros quadrados
mA	Miliampere
MDF	<i>Medium Density Fiberboard</i>
mm <sup>2</sup>	Milímetros quadrados
NLP	<i>Natural Language Processing</i>
PWM	Modulação por Largura de Pulso
RFID	<i>Radio Frequency Identification</i>
SMTP	<i>Simple Mail Transfer Protocol</i>
TCP	Protocolo de Controle de Transmissão
TUE	Tomada de Uso Específico
URL	Localizador Uniforme de Recursos
VA	Volt-ampere
VAC	Volt em Corrente Alternada
VDC	Volt em Corrente Contínua
WAN	Rede remota
Wi-Fi	<i>Wireless Fidelity</i>
WLAN	Rede local sem fio

WWW

*World Wide Web*

## SUMÁRIO

1. INTRODUÇÃO .....	17
1.1. OBJETIVO.....	18
1.2. MOTIVAÇÃO .....	18
1.3. CONTEÚDO .....	18
2. DESENVOLVIMENTO .....	19
2.1. FUNDAMENTAÇÃO TEÓRICA .....	19
2.1.1. COMANDO DE VOZ.....	19
2.1.2. RECONHECIMENTO DE LINGUAGEM FALADA.....	21
2.1.3. MEIOS DE COMUNICAÇÃO .....	23
2.2. DISPOSITIVOS DE INTEGRAÇÃO.....	28
2.2.1. NODEMCU ESP8266.....	28
2.2.2. MÓDULO DE RELÉ COM OPTOACOPLADOR.....	30
2.2.3. LÂMPADA INCANDESCENTE .....	31
2.2.4. COOLER.....	32
2.2.5. TRANSFORMADORES .....	32
2.3. SOFTWARE DE APLICAÇÃO.....	33
3. DESENVOLVIMENTO DO PROJETO .....	34
3.1. PROGRAMAÇÃO DO SOFTWARE .....	35
3.2. INSTALAÇÃO DO HARDWARE .....	50
3.2.1. SIMULAÇÃO DE APLICAÇÃO REAL.....	56
4. TESTES.....	59
5. CONCLUSÃO.....	64
5.1. PROPOSTAS FUTURAS.....	64
6. REFERÊNCIAS .....	66
7. APÊNDICE.....	68
APÊNDICE A – Programação do APP .....	68
APÊNDICE B – Programação do ESP8266.....	68
APÊNDICE C – Montagem do protótipo .....	68

## 1. INTRODUÇÃO

Desde os primórdios a humanidade sempre teve a necessidade de evoluir utilizando seus conhecimentos e recursos disponíveis na época. E com as constantes mudanças muita tecnologia foi desenvolvida. Dentro dos diversos ramos da tecnologia encontra-se a automação, que objetiva facilitar e otimizar processos repetitivos ou com grande grau de periculosidade.

A tecnologia de automação foi evoluindo de tal maneira que tornou possível às máquinas executarem atividades e processos que antes só eram possíveis de formas manuais e presenciais.

Atualmente, as atividades realizadas por máquinas são muito mais comuns, tornando os processos cada vez mais rápidos, eficientes, precisos e gradualmente mais autônomos. Esse cenário abriu portas para novas áreas de estudo tais como a Domótica e todas as tecnologias associadas a ela.

O conceito atribuído à automação residencial é a Domótica. Formada pelas palavras “Domus”, que significa casa, e “Robótica”, tem como objetivo o controle e automatização de aparelhos elétricos e eletrônicos localizados dentro de uma residência, para o conforto e acessibilidade, abrangendo também necessidades especiais.

Baseado nas pesquisas recentes, pode se notar um grande crescimento na procura desta tecnologia, com casas cada vez mais tecnológicas utilizando as novas tendências do mercado de internet das coisas (IoT), visando ter mais conforto e praticidade em funções comuns do dia a dia de uma casa. (FOLHA DA REGIÃO, 2021)

O mercado está em constante inovações tecnológicas tanto nos ramos industriais e residenciais, como por exemplo o Google Home que é um aplicativo de telefone celular que pode se comunicar com vários dispositivos eletrônicos de sua casa contanto que eles possuam conectividade com a internet e o assistente virtual da Google (o qual é responsável pela intercomunicação dos aparelhos).

## 1.1. OBJETIVO

Considerando todos os avanços tecnológicos existentes, visamos em elaborar um projeto que atendesse as necessidades humanas para o conforto residencial.

O objetivo deste projeto é desenvolver um aplicativo de telefone celular em uma aplicação código aberto para controlar e monitorar os dispositivos eletrônicos de uma residência, utilizando, em sua maioria, o comando por voz para acionar os dispositivos eletrônicos, mas também disponibilizar uma forma de controlá-los pela tela do telefone celular.

## 1.2. MOTIVAÇÃO

A principal motivação deste projeto é de aumentar a praticidade e conforto de uma residência com a utilização de um telefone celular, e futuramente utilizar os conhecimentos adquiridos para automatizar um edifício ou uma empresa, pois a automação facilita o controle de processos e monitoramento de forma ampla.

## 1.3. CONTEÚDO

No Capítulo 2 são apresentadas as origens teóricas que nortearam o desenvolvimento do trabalho, no Capítulo 3 é apresentado o desenvolvimento do projeto em si, no Capítulo 4 discute-se os testes realizados para validar o funcionamento, bem como uma análise do resultado, no Capítulo 5 a conclusão sobre os resultados do projeto, no Capítulo 6 são apresentadas as referências utilizadas durante a escrita do documento, no Capítulo 7 contém apêndices que possuem imagens do desenvolvimento do protótipo.

## 2. DESENVOLVIMENTO

Para o entendimento básico do conceito do projeto proposto nesta monografia, se faz necessário a fundamentação teórica de algumas tecnologias e *softwares* utilizados no projeto, de forma que a suas comunicações e interligações fiquem compreensivas.

O termo Domótica foi criado para definir a automação residencial que tem o objetivo de facilitar tarefas cotidianas numa residência, seja para aumentar o conforto, seja para aumentar a acessibilidade para idosos e pessoas com necessidades especiais. Naturalmente, o conceito de Domótica vai muito mais além do conforto de acessibilidade, mas também está ligado às questões de eficiência energética e melhor aproveitamento dos recursos naturais.

Assim, as tecnologias que são desenvolvidas para a área da Domótica objetivam criar casas “inteligentes” sob o ponto de vista de melhor gestão dos recursos internos, seja controlando adequadamente a temperatura, bem como evitando desperdícios com lâmpadas acesas em ambiente onde não há pessoas (FERREIRA, 2010), (SILVEIRA et al., 2014).

### 2.1. FUNDAMENTAÇÃO TEÓRICA

O reconhecimento da voz humana passou por diversas fases e inovações para chegar até onde se encontra atualmente, sendo renovada e evoluída conforme a necessidade e interesse da indústria sobre o assunto. Por meio de diversos projetos e aplicações, o conceito da utilização da voz se tornou amplo e futuramente quase indispensável na sociedade, mas onde e quem teve a primeira idealização sobre o uso do comando de voz?

#### 2.1.1. COMANDO DE VOZ

A utilização do comando de voz teve início na década de 60, mais precisamente em 1962, e a *International Business Machines Corporation* (IBM) foi responsável pelo lançamento do dispositivo pioneiro batizado *Shoebox*. O *Shoebox* tinha o tamanho aproximado de uma caixa de sapatos, reconhecia 16 palavras e dígitos de 0 a 9.

Este dispositivo inovador reconheceu e respondeu a 16 palavras faladas, incluindo os dez dígitos de "0" a "9" [...] O *Shoebox* era operado falando num microfone, que convertia sons de voz em impulsos elétricos. Um circuito de medição classificou estes impulsos de acordo com vários tipos de sons e ativou uma máquina de adicionar anexada através de um sistema de relé. (IBM, 2003).

Na Figura 1 é possível ver o diretor do grupo de tecnologia avançada do laboratório de desenvolvimento de sistemas avançados da IBM demonstrando o dispositivo *Shoebox*.

**Figura 1** - Dr. E. A. Quade demonstrando o *Shoebox*.



Fonte: (IBM, 2003)

Nos anos 70, a *Defense Advanced Research Projects Agency* (DARPA) dos Estados Unidos da América criou o *Harpy*, o *software* mais avançado da época, mais avançado que o próprio *Shoebox* da IBM. Ele conseguia interpretar até aproximadamente 1000 palavras. Na década de 90, mais precisamente em 1993, alguns modelos de *Macintosh* já conseguiam utilizar reconhecimento de fala.

No início dos anos 2000, o Google lançou o "*Google Voice Search*", que é *software online* capaz de reconhecer a voz do usuário para fazer pesquisas na *internet*. (ILUMEO, 2019)

O comando por voz se tornou uma ferramenta extremamente prática e útil para a sociedade atual, podendo ser aplicada em diversos âmbitos, que variam de *softwares* para ajuda à digitação (auxiliando necessidades especiais), controle de dados, informações e segurança.

A lista é tão extensa que é possível imaginar o uso do comando por voz em diversas áreas e situações, e a automação conseguiu utilizá-lo de uma forma que além de dar espaço para as infinitas possibilidades ainda é possível facilitar ações triviais de nosso cotidiano.

### 2.1.2.RECONHECIMENTO DE LINGUAGEM FALADA

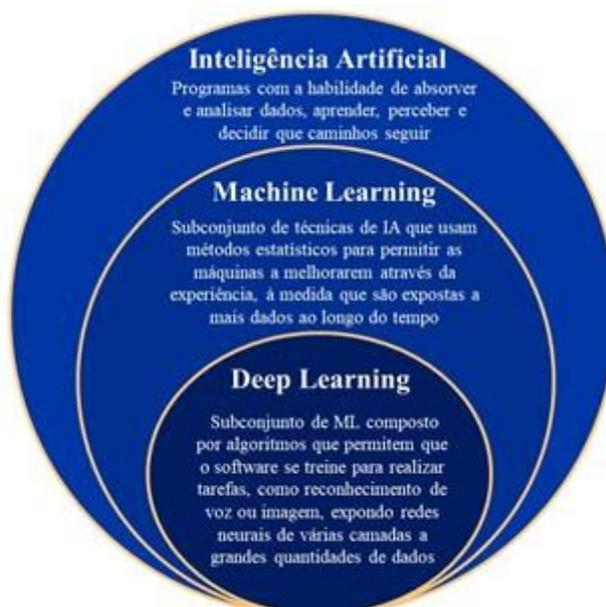
Os assistentes virtuais utilizam *softwares* baseados em Inteligência Artificial (IA), e em conjunto com outros dispositivos inteligentes é possível expandir a área de funcionamento de um *software* individual, utilizando comunicações por meio de redes *Wireless* ou conexões *Bluetooth*. Como exemplo de funcionamento podemos citar um telefone celular conectado à alto-falantes inteligentes, no qual o controle é realizado por meio do comando de voz ao receber informações do usuário e executá-las.

Muitas das automatizações utilizam a ideia da *Natural Language Processing* (NLP), um ramo da IA que une a programação computacional comum e modelos estatísticos de *Machine Learning* e *Deep Learning* para captar e processar a linguagem humana, seja por forma de texto ou voz, de forma que seja compreensível o entendimento homem-máquina.

O conceito da *Machine Learning* ou aprendizado de máquina, é um conceito associado à IA, como uma forma de dar inteligência para a máquina de tal forma que ela possa executar uma tarefa e com as repetições seu funcionamento é otimizado, o que gera melhorias aos processos de execução.

O *Deep Learning* é a camada mais profunda da IA, que por sua vez é capaz de analisar e tomar decisões por si própria baseada em várias camadas de algoritmos, sendo assim uma exemplificação bem simplória da *Deep Learning* é defini-la como um cérebro digital. A ideia das camadas da IA pode ser compreendida pela Figura 2, sendo vista como conceitos englobados para formar um conceito final.

**Figura 2 - Camadas da IA**



Fonte: (Sixty Degrees, 2021)

Considerando o conceito amplo de IA, mais especificamente *Machine Learning*, surgem dúvidas sobre formas de ensinar à máquina atividades, sejam elas de fácil realização ou não, e claro que as respostas assim como as dúvidas são praticamente infinitas. Por meio do uso do microcontrolador denominado como ESP8266 a automatização se transforma em um processo menos complexo, e por conta do tamanho do próprio microcontrolador e componentes, mais compacto. No entanto para o funcionamento com uso de voz é necessário entender e aplicar o conceito de IoT.

Segundo Santos (2018), “Internet das Coisas significa dispositivos que podem detectar aspectos do mundo real – como temperatura, iluminação, presença ou ausência de pessoas ou objetos, etc. – e relatar dados do mundo real, ou agir sobre isto.” e com o uso da IoT o microcontrolador consegue realizar uma conexão com o telefone celular, intermediado pelo sinal de *Wireless Fidelity* (Wi-Fi), e o usuário poderá controlar e monitorar o funcionamento de todo o protótipo em tempo real.

Com a IoT como base do projeto, todo o funcionamento restante é constituído pela conexão elétrica e eletrônica do microcontrolador com os componentes instalados no protótipo. Por necessitar de uma conexão *wireless* o microcontrolador citado foi selecionado por já dispor de um módulo de Wi-Fi interno.

### 2.1.3.MEIOS DE COMUNICAÇÃO

Apesar das controvérsias sobre a origem da comunicação Wi-Fi estar relacionada com os avanços tecnológicos da Segunda-Guerra Mundial, como uma invenção criada por Hedy Lamarr e George Antheill, em 1942, e que foi rejeitada devido sua complexidade, os registros indicam que o Wi-Fi com o protocolo 802.11 surgiu em 1997.

O Wi-Fi é constituído de três camadas físicas que determinam os modos de transmissão alternativos, *Direct Sequence Spread Spectrum* (DSSS), *Frequency Hopping Spread Spectrum* (FHSS) e infravermelho. Cada uma dessas camadas é responsável pela definição do meio de transmissão, por codificar e decodificar os sinais, além de receber e transmitir *bits* (LUGLI & SANTOS, 2014).

Ao longo dos anos surgiram variações, como exemplo: o protocolo 802.11a e o 802.11b no ano de 1999, 802.11g em 2003, 802.11n em 2009. Cada protocolo utiliza uma tecnologia diferente de radiofrequência.

O protocolo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) 802.11, inicialmente fornecia uma velocidade de transmissão de 2 Mbps na frequência de 2,4 GHz em redes locais sem fio (WLAN). Tal velocidade foi superada pelos protocolos sucessores. O protocolo 802.11b permitia 11 Mbps, o 802.11a fornecia no máximo 54 Mbps em uma frequência de 5 GHz, o 802.11g fornece os mesmos 54 Mbps em relação a uma frequência de 2.4 GHz, e ainda há o 802.11n que pode atuar em redes com frequência de 2,4 e/ou 5 GHz fornecendo cerca de 65 a 600 Mbps (RUFINO, 2007). A velocidade pode variar de 1 a 600 Mbps e a radiofrequência pode ser de 2,4 GHz, 5 GHz e, atualmente está sendo testado o uso da radiofrequência de 6 GHz.

Dependendo da necessidade, alguns tipos de radiofrequência são específicos para cada situação baseado na frequência x largura de banda x taxa de dados (TANENBAUM, 2011).

O Quadro 1 apresenta informações mais detalhadas sobre diversos protocolos IEEE padronizados, definindo a frequência trabalhada, a banda e a taxa de dados para cada protocolo.

**Quadro 1 - Especificações protocolo IEEE**

<b>Protocolo IEEE</b>	<b>Frequência</b>	<b>Banda</b>	<b>Taxa de Dados</b>
802.11ax	2,4 ou 5 GHz	20, 40, 80, 160 MHz	2,4 Gbps
802.11ac wave2	5 GHz	20, 40, 80, 160 MHz	1,73 Gbps
802.11ac wave1	5 GHz	20, 40, 80 MHz	866,7 Mbps
802.11n	2,4 ou 5 GHz	20, 40 MHz	450 Mbps
802.11g	2,4 GHz	20 MHz	54 Mbps
802.11a	5 GHz	20 MHz	54 Mbps
802.11b	2,4 GHz	20 MHz	11 Mbps
802.11	2,4 GHz	20 MHz	2 Mbps

Fonte: Autores (2022)

No entanto, com o crescimento exponencial das redes foi necessário desenvolver uma maneira de conectá-las, e para isso foi criado e aplicado o Protocolo de controle de transmissão/Protocolo de internet (TCP/IP), que é um conjunto de protocolos de comunicação, que é definido como uma linguagem utilizada para fazer dois dispositivos se comunicarem. Este protocolo é definido em cinco camadas, conforme listado a seguir:

- 1- Camada Física – Esta camada demonstra as características físicas da comunicação, assim como a natureza do meio usado para ela (cabos ou radiofrequência) e todos os detalhes relacionados com os sinais (modulações, comprimentos de onda, níveis de sinal, distâncias máximas etc.);
- 2- Camada Link – Esta camada tem como principal função a adaptação do modelo TCP/IP aos diversos tipos de redes, devido à enorme variedade de tecnologias de rede possíveis, é uma camada sem padrão do modelo TCP/IP, portanto, é possível a ligação e operação com redes sem uma normalização de protocolo;
- 3- Camada Rede – A Camada Rede é uma camada em que o único protocolo utilizado é o IP. Esta camada é responsável pelo endereçamento, roteamento e controle de envio e recepção dos dados. O IP é o protocolo responsável por definir o caminho que um pacote de dados deverá percorrer desde remetente até o destinatário, porém não garante que os dados chegarão, muito menos se estarão em ordem de envio, passando por uma ou várias redes;

- 4- Camada Transporte – Esta camada é responsável pelas funções de transporte que incluem os mecanismos necessários para garantir a entrega sequencial de dados, sem alterações na ordem de envio. O acesso das diversas aplicações desta camada é feito através de portas que possuem números inteiros distintos para cada tipo de aplicação, como opções podem ser utilizados dois protocolos distintos para o transporte, o TCP e o Protocolo de Datagramas do Usuário (UDP). O TCP é recomendado para conexão enquanto o UDP não, pois não oferece garantias de entrega, nem de uma sequência correta de envio. Normalmente o UDP só é utilizado em aplicações que necessitam de elevados volumes de tráfego na Internet;
  
- 5- Camada Aplicação – Camada formada por um grande conjunto de protocolos que permitem o funcionamento dos diversos serviços e aplicações do modelo TCP/IP. Esta camada não possui um protocolo comum para todas as aplicações, ou seja, considera o serviço em questão e varia conforme o protocolo que vai o atender, como exemplo, o serviço e-mail utiliza o protocolo *simple mail transfer protocol* (SMTP). Sempre que este serviço é requerido pelo TCP/IP (envio ou recepção de e-mail), é este protocolo que se encarrega de atendê-lo, portanto, por trás de cada aplicação existe um protocolo específico.

A comunicação é iniciada na Camada Aplicação, responsável por receber a requisição de um *software* e processá-la. Utilizando o protocolo TCP, a Camada Transporte adquire os dados enviados pela Camada Aplicação, os divide em pacotes e os envia para a camada inferior, enquanto o recebimento está operando a Camada Transporte também ordena os pacotes recebidos e verifica se o conteúdo deles se encontra intacto.

Na Camada Rede, o protocolo IP é responsável por direcionar os pacotes recebidos da Camada Transporte através da adição de endereços virtuais, nomeados como IP, desta forma, os pacotes recebidos são adicionados aos endereços de remetente e destinatário, e em seguida são transportados para a Camada Rede, sendo chamados de datagramas. Nas Camadas Link e Camada Física, o TCP/IP não determina nenhum protocolo específico, abrangendo todos os protocolos padrões e patenteados. Nota-se que durante a interconexão de uma rede TCP/IP, a rede conectada pode ser local (LAN) ou remota (WAN). (FOROUZAN, 2008)

Conforme a sociedade evolui a tecnologia acompanha essa evolução, e obviamente, a tecnologia de comunicação também. Porém, até o momento existe apenas uma maneira de gerar a interconexão entre milhares computadores, a *Internet*.

Até a década de 1990 seu uso era limitado apenas a transferências de arquivos, enviar e receber correio eletrônico ou notícias, no entanto, nesse mesmo período, uma nova aplicação foi colocada em uso, a *World Wide Web* (WWW).

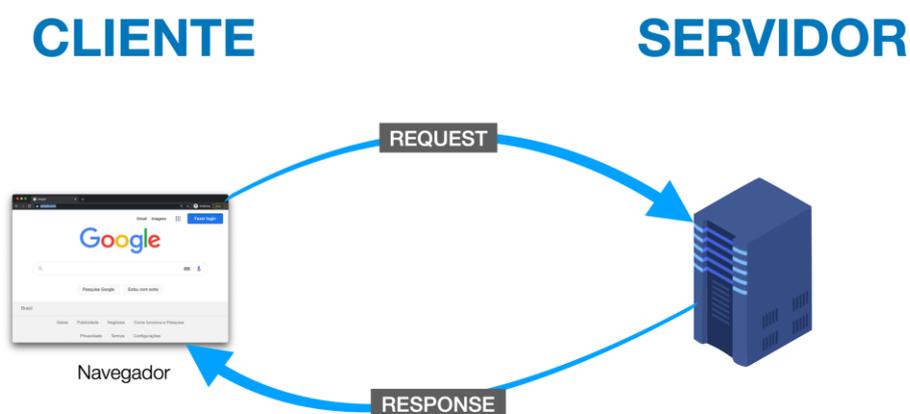
Segundo Kurose (2010) “a Web é a aplicação da *Internet* que chamou a atenção do público em geral. Ela transformou drasticamente a maneira como pessoas interagem dentro e fora de seus ambientes de trabalho. Alçou a *Internet* de apenas mais uma entre muitas redes de dados para, essencialmente, a única rede de dados”.

A utilização da Web trouxe muitos benefícios para usuários, de forma que eles recebem e enviam dados sobre o que querem, quando querem, além da facilidade de disponibilizar e encontrar dados na Web, mas para o funcionamento da Web é necessário a utilização de um protocolo conhecido como Protocolo de Transferência de Hipertexto (HTTP).

O HTTP é um protocolo de comunicação utilizado para a transferência de dados pela Web, como padrão ele é dividido em duas partes, o cliente e o servidor, os dois são executados em máquinas diferentes e conversam um com o outro. O cliente é responsável por enviar uma solicitação ao servidor e o servidor é responsável por responder com os dados solicitados, no caso da comunicação Web o cliente é o navegador e o servidor é a própria página Web.

A Figura 3 representa um exemplo da comunicação entre o cliente e o servidor, exemplificando o cliente como um navegador. Para o nosso projeto o cliente será o Aplicativo (APP) desenvolvido para o smartphone e o servidor será o NodeMCU ESP8266, ou seja, a requisição será realizada pelo APP e será respondida pelo ESP8266 com os dados requisitados.

**Figura 3 - Comunicação Cliente/Servidor**



Fonte: (Metrópole Digital, s.d.)

Para a conexão completa entre o cliente e o servidor os seguintes passos devem ser seguidos: O cliente HTTP iniciar uma conexão TCP com o servidor utilizando a porta 80 (número de porta padrão para HTTP), após a conexão o cliente HTTP pode enviar uma requisição ao servidor. O servidor receberá a mensagem de requisição, irá extrair o dado transmitido, encapsulará o dado em uma mensagem de resposta HTTP e envia a resposta ao cliente, quando o TCP possuir certeza de que o cliente recebeu a resposta intacta a conexão é encerrada. (KUROSE, 2010)

Para o navegador fazer uma requisição ele se comunica com a página Web em que deseja receber a resposta, mas para a conexão com a página é necessário um *Uniform Resource Locator*, ou Localizador Uniforme de Recursos (URL), para identificar e localizar o recurso requisitado na Web. O URL é uma forma eficiente de localizar o recurso que se está requisitando, pois é composto por uma sequência de caracteres que respeitam uma estrutura específica para fornecer informações para o cliente e o servidor, facilitando o encontro. Segundo Gourley e Totty (2002):

- A primeira parte de um URL indica como acessar o recurso, no caso, o URL informa a utilização do protocolo HTTP (Exemplo: "http://");
- A segunda parte de um URL é a localização do servidor, que indica ao cliente onde deve ser buscada a requisição (Exemplo: "www.site.com.br");

- A terceira parte de um URL é o caminho de recurso, que indica qual recurso local particular do servidor está sendo requerido (Exemplo: “/pasta/arquivo.jpg”);
- Caso o URL exemplificado ocorresse de ser digitado completamente: “http://www.site.com.br/pasta/arquivo.jpg”.

O padrão de um endereço URL pode ser reaproveitado em diversas ocasiões, como exemplo a comunicação HTTP utilizando um IP. O conceito é semelhante, porém para ser funcional seria essencial descobrir o IP no qual os dados irão ser recepcionados e respondidos e substituir o domínio da estrutura da URL pelo IP em questão, ou seja, suponhamos que o IP da comunicação seja “192.168.1.0”, para a comunicação ser funcional a URL seria: “http://192.168.1.0/mensagem.html”.

No entanto, por mais que este método seja funcional não é recomendado o uso no lugar de um domínio da Web por conta de suas limitações, afinal caso o servidor esteja hospedado em um IP dinâmico, por exemplo, o IP pode mudar e interromper a comunicação, gerando mais dificuldades em manter a conexão e reconexão.

Por conta da praticidade e versatilidade, o protocolo HTTP tornou-se o protocolo padrão para a comunicação Web.

## 2.2. DISPOSITIVOS DE INTEGRAÇÃO

A construção do *hardware* baseado no conceito de Domótica, simulando uma comunicação IoT, necessita de componentes elétricos e eletrônicos. Todos os componentes comerciais que foram utilizados na elaboração e execução do projeto estão descritos a seguir.

### 2.2.1. NODEMCU ESP8266

O microcontrolador NodeMCU ESP8266 tem um funcionamento semelhante à um “cérebro” dentro do circuito do protótipo, possibilitando comandar os sinais elétricos que são transmitidos. O NodeMCU é formado por um ESP12E, é uma placa criada pela ALTHINKER no qual ainda possui um microchip ESP8266EX feito pela empresa

chinesa *Espressif Systems*, esse microchip possui Wi-Fi integrado e baixo consumo de energia, e um processador RISC Tensilica L 106 32bit. (KOYANAGI, 2017a)

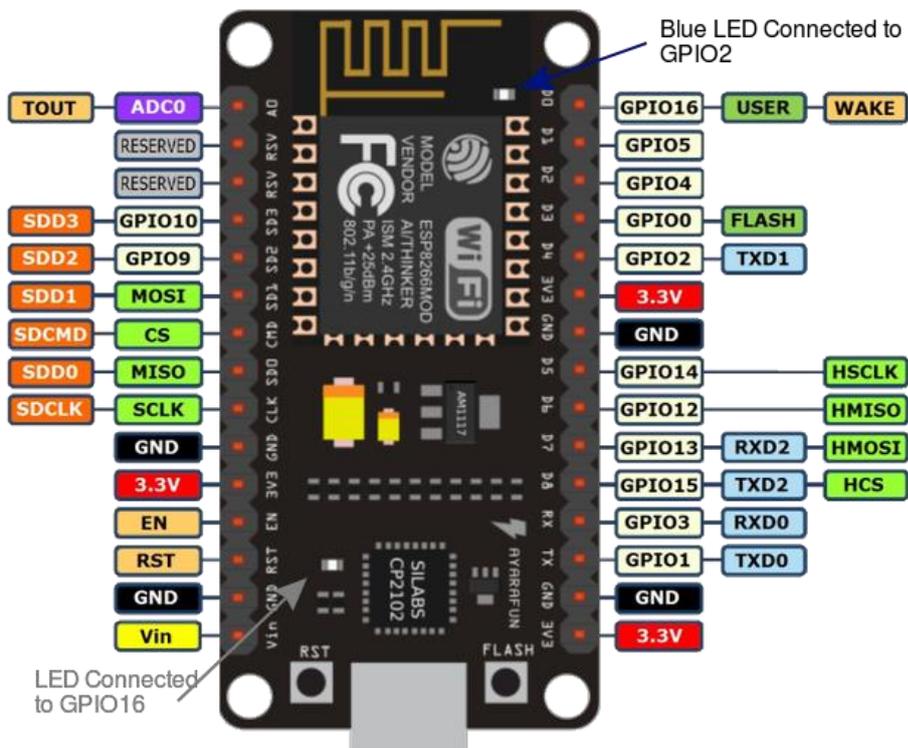
Esse microcontrolador possui 30 pinos, sendo deles 16 considerados para *General Purpose Input/Output* (GPIO), ou seja, podem ser programados como entradas ou saídas com múltiplas funções (podem variar dependendo do pino, mas em geral são saídas PWM, SPI, UART etc.) gerando assim, possibilidades de escolha da quantidade de entradas ou saídas utilizadas, sendo totalmente a critério do desenvolvedor. PWM em inglês, significa *Pulse-Width Modulation*, ou seja, modulação por largura de pulso, é um controlador de potência, como uma espécie de um *dimmer* de um abajur, escurecendo e clareando, este mecanismo te possibilita também, por exemplo, ligar um driver de um motor. (KOYANAGI, 2017b)

SPI significa *Serial Peripheral Interface* (interface periférica serial), é uma comunicação serial síncrona para pequenas distâncias, essa *interface* utiliza uma linha para sincronizar os dados entre o dispositivo que envia e o dispositivo que recebe. A SPI é uma comunicação do tipo *leader/follower* (líder/seguidor), onde só pode existir um líder que controla a comunicação e inúmeros seguidores que são controlados. A SPI não define um protocolo para interpretar os dados recebidos /enviados, sendo assim, fica a critério de quem criar os dispositivos decidir como isso será feito. (GUIMARÃES, 2019)

A UART (*Universal Asynchronous Receiver/Transmitter* ou Receptora/Transmissora Assíncrona Universal) é um termo universal referido ao fato do formato do dado e a velocidade serem configuráveis. Porém, somente um *bit* de informação pode ser transmitido ou recebido por vez, pois o sistema não possui *clock* ou um qualquer sincronismo na comunicação, se tornando necessário a sincronização da velocidade entre os dispositivos. (BALDASSIN, 2019)

A Figura 4 apresenta a pinagem do microcontrolador NodeMCU ESP8266, com representações de cada pino e suas funções específicas.

**Figura 4 - Pinagem NodeMCU ESP8266**



Fonte: (circuits4you, 2017)

## 2.2.2. MÓDULO DE RELÉ COM OPTOACOPLADOR

O módulo de relé é um dispositivo eletromecânico utilizado para ligar e desligar cargas elétricas. Como é formado por contatos elétricos sem potencial, pode ser usado para alimentar qualquer carga em qualquer nível de tensão, bastando respeitar os limites de tensão corrente dos contatos. O módulo de relé pode conter ilimitados número de relés, porém, será utilizado um módulo com 8 relés e outro módulo com 4 relés. O funcionamento do relé se baseia no circuito de comando que aciona o eletroímã, abrindo ou fechando o contato.

A Figura 5 ilustra um relé com optoacoplador. No protótipo foram utilizados módulos com mais relés, portanto a figura representa apenas um relé dos módulos citados.

**Figura 5** - Módulo Relé 5v 10A com Optoacoplador



Fonte: (Smartkits, s.d.)

Este módulo de relé possui um optoacoplador, o qual serve como um sistema de segurança para proteção em casos de descargas elétricas, pois elimina a necessidade de contato entre os dois circuitos do relé, possibilitando também o trabalho com tensões variadas. O relé será utilizado entre o microcontrolador e o circuito elétrico do protótipo, como forma de proteção, para aplicar a tensão de 127 Volt em Corrente Alternada (VAC) e possibilitar o controle de funcionamento por meio do microcontrolador.

### 2.2.3. LÂMPADA INCANDESCENTE

A lâmpada incandescente é um dispositivo criado em 1879, por Thomas Edison, capaz de transformar a energia elétrica em luminosidade e calor, por meio do efeito Joule. Este tipo de lâmpada definitivamente não é o mais eficiente, pois grande parte da conversão da energia é transformada em calor, no entanto foi o modelo mais acessível, dentre as opções que se adequavam às dimensões reduzidas do protótipo.

A Figura 6 apresenta uma lâmpada incandescente semelhante a utilizada no protótipo.

**Figura 6** - Lâmpada Incandescente 127VAC 7W



Fonte: (Brasfort, 2018)

#### 2.2.4. COOLER

Um cooler é um dispositivo utilizado para dissipar o calor. No protótipo será utilizado como medida de estabilizar a temperatura da caixa de controle, semelhante a um ar-condicionado.

#### 2.2.5. TRANSFORMADORES

O transformador é um dispositivo usado para reduzir ou aumentar a tensão e a corrente elétrica do circuito. Em sua constituição, ele é composto por duas bobinas e um núcleo ferromagnético (As bobinas são enrolamentos de fio de cobre), as bobinas são designadas como bobina primária e bobina secundária, sendo a bobina primária posicionada próxima a fonte de tensão e a bobina secundária próxima da carga. O núcleo ferromagnético é composto por um material com propriedades magnéticas.

Quando a bobina primária é energizada por uma fonte de corrente alternada gera um campo magnético variável que divide as espiras da bobina secundária, tornando possível surgir uma tensão induzida na bobina secundária, a tensão e a corrente são controladas pelo número de espiras existentes na bobina secundária, portanto se a bobina secundária possuir menos espiras do que a bobina primária a tensão e corrente geradas serão menores do que os valores que adentram a bobina primária e vice-versa. Para este protótipo foram utilizados dois abaixadores de tensão, 127VAC em 12 Volt em Corrente Contínua (VDC) e 127VAC em 5VDC.

## 2.3. SOFTWARE DE APLICAÇÃO

Para o funcionamento do projeto como um todo foi necessário a criação de um aplicativo para telefone celular capaz de realizar e monitorar todos os processos internos e externos do protótipo.

Graças à aplicação de código aberto nomeada como MIT App Inventor, ou apenas App Inventor, foi possível a conexão entre celular e protótipo de forma simples e prática, afinal apenas necessita a conexão do telefone celular e do microprocessador (ESP8266) com a mesma rede de internet. O funcionamento do App Inventor é, de certa forma, pensado para que a programação seja produzida de forma totalmente intuitiva e sem necessidade de nenhum conhecimento prévio sobre linguagens de programação, e com o uso da programação em blocos a programação intuitiva é possível.

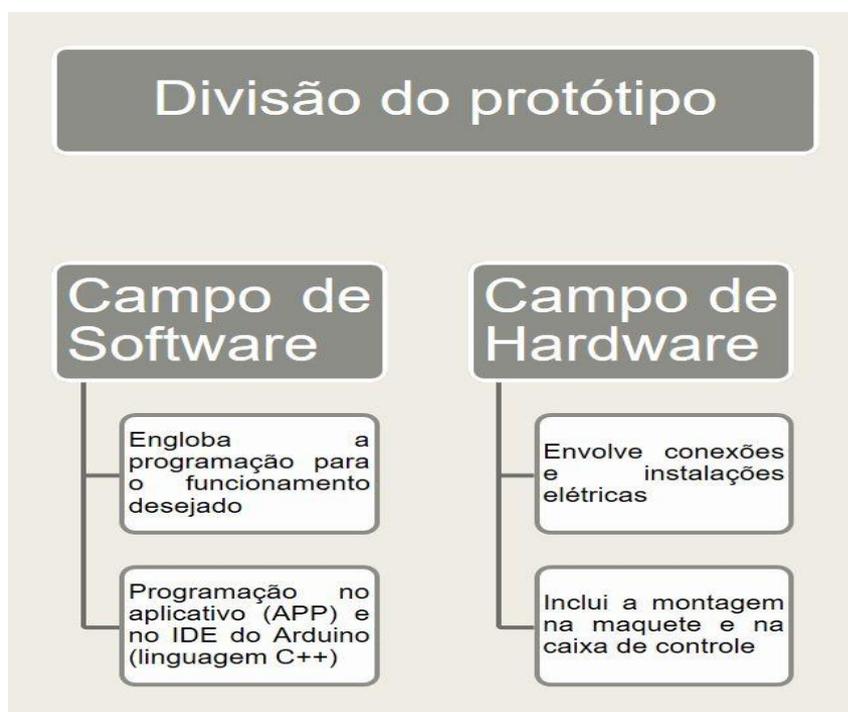
Considerando as necessidades apresentadas neste projeto, acreditamos que o aplicativo seria otimizado separando as áreas do protótipo, no entanto devido ao limite de abas ser apenas 10 a divisão ficou mais restrita.

### 3. DESENVOLVIMENTO DO PROJETO

Com toda a teoria introduzida é possível iniciar os planejamentos necessários, pré-montagem e a própria montagem, pondo em prática todos os estudos e conhecimentos adquiridos. Com estudos e planejamentos concluímos que o protótipo será realizado de forma mais prática se for dividido em campos, portanto deliberamos a divisão em dois campos.

O primeiro campo a ser tratado será o campo de *software*, que engloba toda a programação requisitada para o funcionamento desejado, sendo ela dentro do APP introduzido previamente ou no Ambiente de Desenvolvimento Integrado (IDE) desenvolvida para o microcontrolador Arduino para a programação na linguagem C++. O segundo campo é o de *hardware*, que envolve toda conexão e instalação elétrica, sendo ela dentro da maquete ou na caixa de controle. A Figura 7 demonstra a divisão entre os campos de *software* e *hardware*.

**Figura 7 - Divisão de campos**



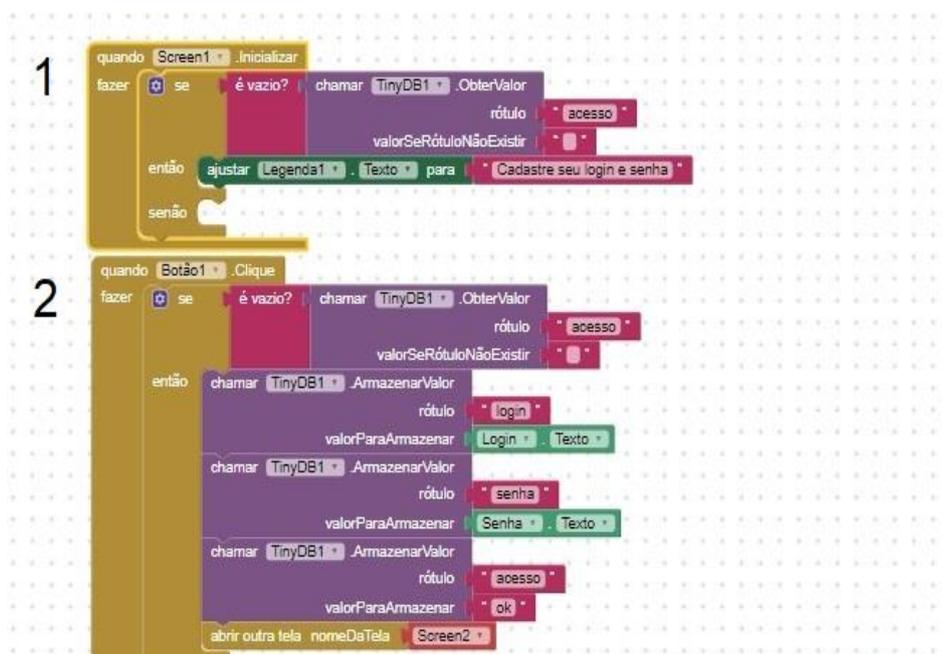
Fonte: Autores (2023)

### 3.1. PROGRAMAÇÃO DO SOFTWARE

O APP para telefone celular, como foi introduzido, foi desenvolvido em uma aplicação de código aberto em que disponibiliza a programação por meio de blocos, tornando o desenvolvimento muito mais intuitivo e visual.

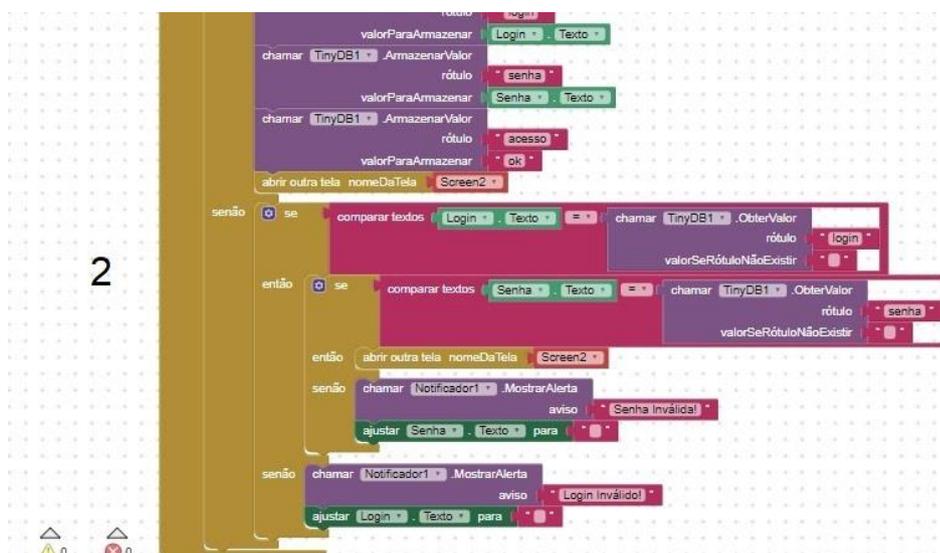
As figuras disponibilizadas abaixo serão acompanhadas de explicações detalhadas sobre o funcionamento principal do APP. Caso seja de interesse, as imagens com a programação em blocos de todas as telas utilizadas estão disponibilizadas no **APÊNDICE A – Programação do APP**. A Figura 8 e a Figura 9 representam a programação em blocos utilizada para a construção da tela de *login*.

**Figura 8 - Programação da tela de *login* (1)**



Fonte: Autores (2022)

**Figura 9 - Programação da tela de *login* (2)**



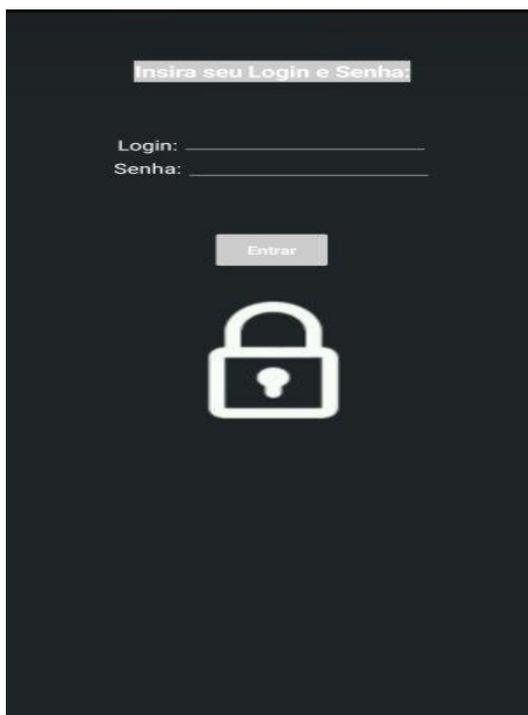
Fonte: Autores (2022)

Para melhor entendimento iremos separar partes diferentes de programação, ou que exigem uma explicação detalhada, por meio de uma numeração.

- 1- A programação durante esta seção utiliza o componente nomeado como TinyDB1 para armazenar informações importantes, no caso desta primeira separação o componente TinyDB1 é configurado para obter o *login* e a senha do usuário caso o rótulo acesso não possua nenhum *login* e senha cadastrado pelo usuário;
- 2- Na segunda separação, durante a Figura 8, pode-se ler, se o Botao1 (botão configurado para o envio do *login* e da senha) for pressionado, e não existir dados de *login* e senha no rótulo acesso, os dados escritos serão guardados como *login* e senha para a utilização e o APP enviará o usuário para a próxima tela. Já na Figura 9, pode-se ler, quando o Botao1 for pressionado, caso existam dados de *login* e senha cadastrados, ele irá compará-los e liberar a passagem somente se os dados forem iguais, em uma situação que não são iguais o APP irá emitir uma notificação para informar que o *login* ou a senha são inválidos, ou seja, não são iguais aos cadastrados no TinyDB1.

A Figura 10 representa a tela de *login*, vista pelo APP desenvolvido para o smartphone. Onde se percebe a presença de caixas de texto para inserir o *login* e a senha, um botão para concluir a ação e uma nota para indicar ao usuário qual será seu passo seguinte.

**Figura 10** - Tela de login APP

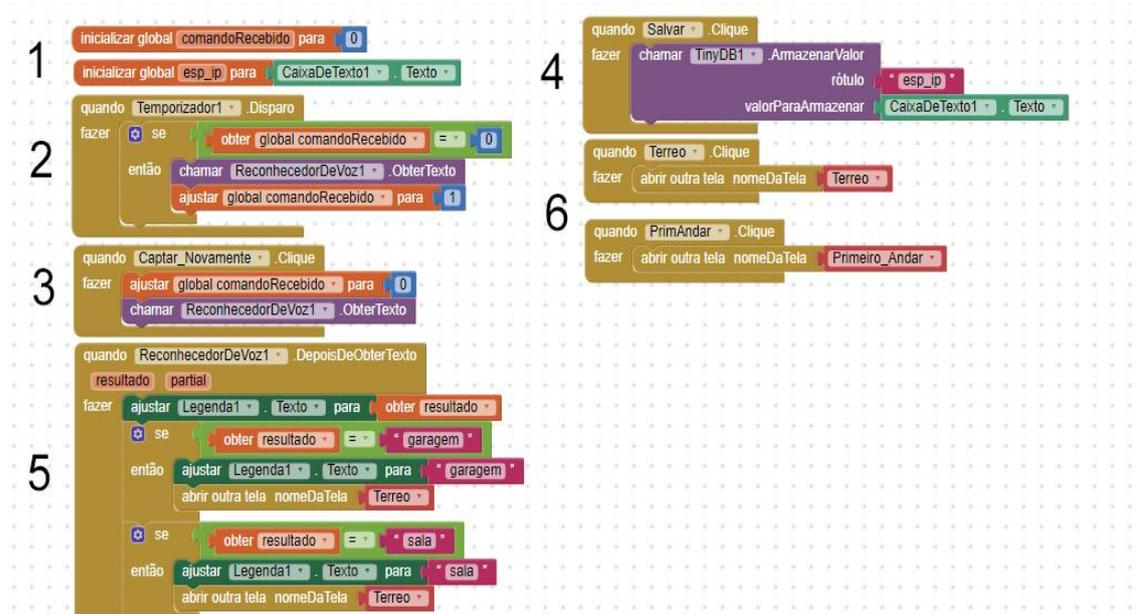


Fonte: Autores (2022)

Mesmo com a utilização de *login* e senha para cada smartphone conectado à rede, é possível realizar o controle do protótipo por diversos usuários simultaneamente, contanto que cada um esteja controlando o ambiente que escolheu por um *smartphone* diferente.

Com a confirmação dos dados cadastrados na tela de *login* o APP envia o usuário diretamente para a tela principal, onde se encontram todos os indicadores para os cômodos automatizados do protótipo. A programação em blocos utilizada na tela principal do programa está representada na Figura 11.

**Figura 11 - Programação tela principal**



Fonte: Autores (2023)

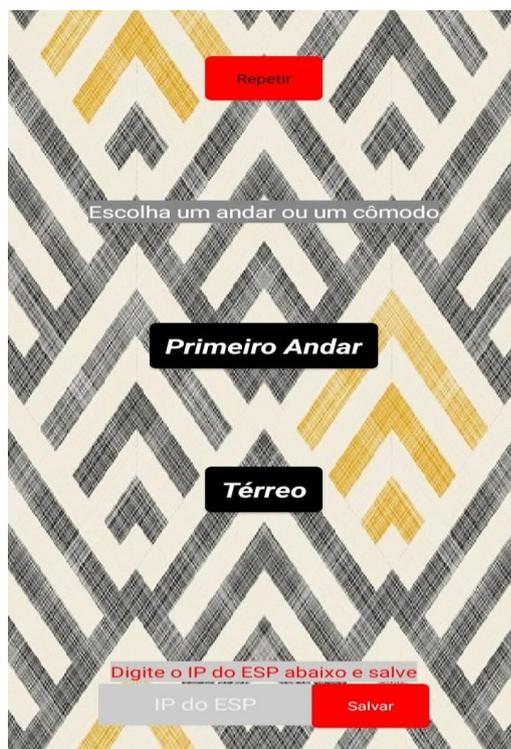
A tela principal do APP tem uma programação dedicada ao direcionamento de outras telas e definição do IP. Partindo da tela principal é possível acessar uma tela específica para cada cômodo do protótipo.

- 1- É necessário definir duas variáveis globais para o funcionamento deste APP: a variável global “comandoRecebido” (inicializada em zero), que indica para o temporizador que é possível chamar o “ReconhecedorDeVoz1” e a variável global “esp\_ip” (inicializada com o mesmo IP que for definido), que disponibiliza IP para a comunicação de forma global, ou seja, para que outras telas também possam acesso;
- 2- Quando o temporizador começar seu funcionamento ele irá conferir se a variável “comandoRecebido” é igual a zero, caso a condição seja verdadeira, o “ReconhecedorDeVoz1” é inicializado e a variável global “comandoRecebido” é definida como um;
- 3- Por meio de testes com o APP nota-se que o temporizador não funciona de forma constante, portanto foi necessário incluímos um

botão nomeado “Captar\_Novamente”. Ao pressionar “Captar\_Novamente” ele define a global “comandoRecebido” para zero e reinicializará o “ReconhecedorDeVoz1”;

- 4- O botão nomeado Salvar é acompanhado de uma caixa de texto (“CaixaDeTexto1”) e ao ser pressionado inicializará o TinyDB1 para armazenar o IP inserido na “CaixaDeTexto1”;
- 5- Quando o “ReconhecedorDeVoz1” obtiver alguma resposta por comando de voz ela será automaticamente armazenada na variável “resultado” e visível na Legenda1, caso o resultado seja igual a um cômodo do protótipo ou o nome de algum andar, como térreo ou primeiro andar, o APP irá direcionar o usuário até a tela respectiva;
- 6- Por conta de praticidade também foram adicionados dois botões que levam para a tela do térreo ou do primeiro andar.

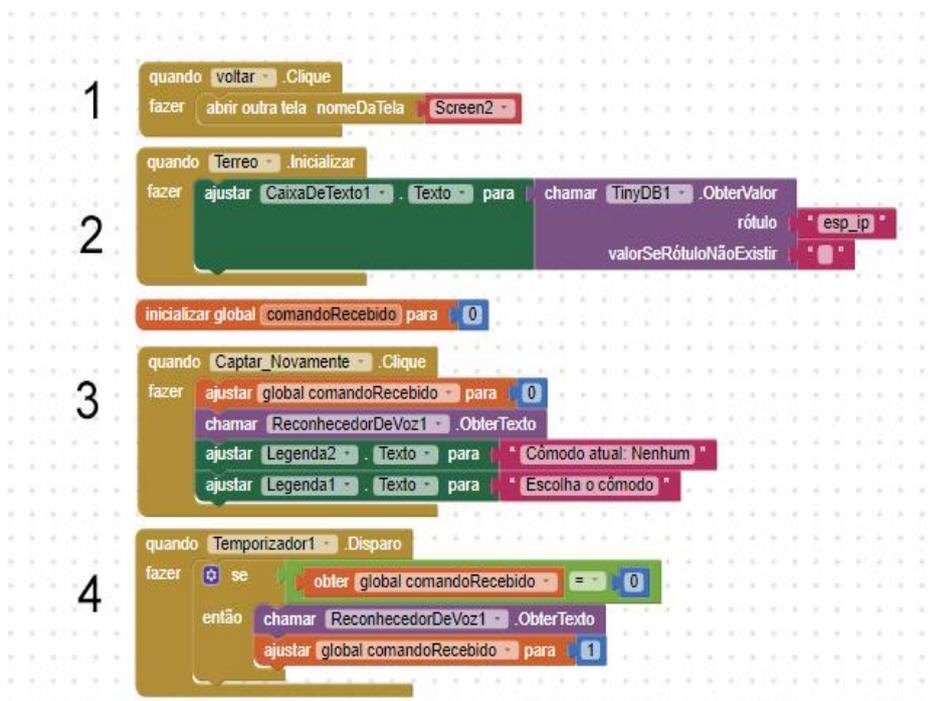
Com toda a programação da tela principal explicada, é possível introduzir uma representação visual da tela principal por meio da Figura 12. Pode-se notar a presença de um botão para reiniciar a captura de voz, nomeado como repetir, logo abaixo uma nota indicando a necessidade de escolher um andar ou cômodo, assim como dois botões representando cada andar do protótipo, e por fim, uma caixa de texto acompanhada de um botão “salvar” para registrar o IP utilizado na comunicação.

**Figura 12 - Tela principal APP**

Fonte: Autores (2023)

A programação apresentada na Figura 13, Figura 14 e Figura 15 será referente ao andar térreo, no entanto, por todas as telas de andares possuírem uma programação semelhante, o uso é dedicado à explicação da programação como um todo, por este mesmo motivo iremos identificá-las como “genérica”.

**Figura 13** - Programação genérica de tela de andar (1)



Fonte: Autores (2023)

Ao iniciar a tela algumas informações serão disponibilizadas pela “Legenda1”, “Legenda2” e “CaixaDeTexto1”, respectivamente, um texto informando que é necessária a seleção de um cômodo, o cômodo atualmente selecionado (por padrão o APP inicia sem a seleção de um cômodo, afinal isto deve ser feito pelo usuário) e o IP definido na tela anterior.

- 1- O botão nomeado voltar levará o usuário de volta à tela principal caso seja pressionado;
- 2- Quando a tela (considerando a Figura 13, a tela que foi utilizada para referência é a tela térreo) inicializar, a CaixaDeTexto1 é ajustada para o valor que foi salvo como “esp\_ip” anteriormente. A variável global “comandoRecebido” é ajustada para zero;
- 3- O botão nomeado “Captar\_Novamente” tem a mesma função do botão com a mesma nomenclatura da tela principal;

- 4- O temporizador também possui a mesma funcionalidade do temporizador utilizado na tela principal;

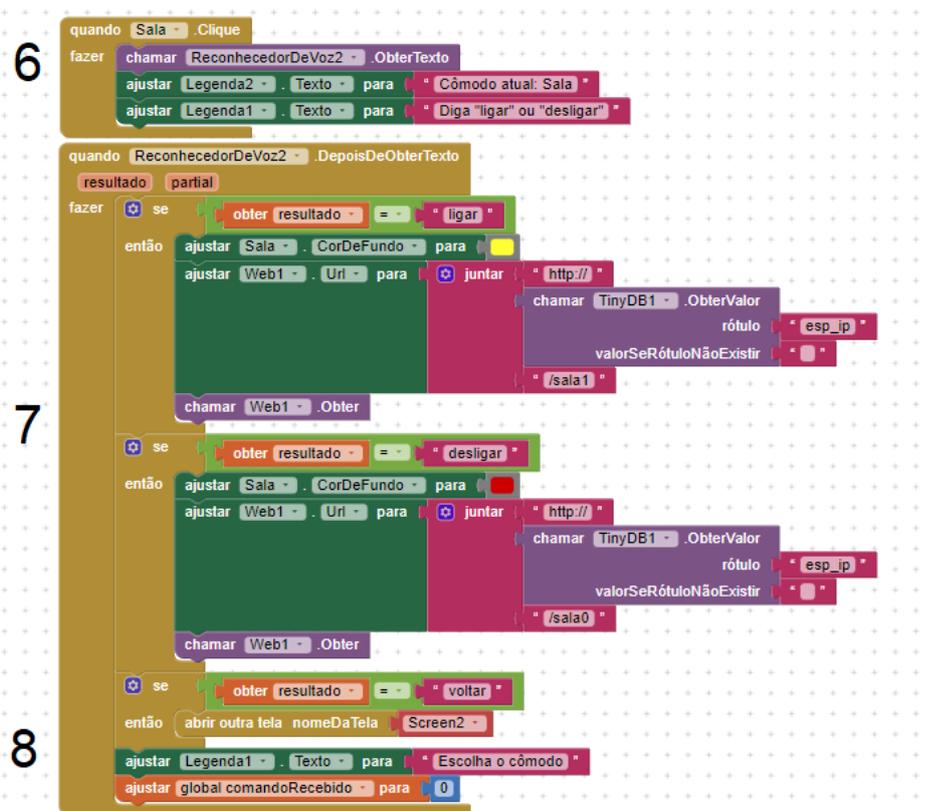
**Figura 14 - Programação genérica da tela de andar (2)**



Fonte: Autores (2023)

- 5- Quando o “ReconhecedorDeVoz1” obter algum resultado haverá uma comparação com os cômodos nomeados no protótipo, caso o texto obtido seja igual ao nome de um cômodo, a Legenda2 será ajustada para exibir o cômodo em que foi selecionado pelo resultado do “ReconhecedorDeVoz1”, o “ReconhecedorDeVoz” respectivo ao cômodo escolhido será inicializado (utilizando como referência a Figura 13, caso o ambiente em questão seja a sala, o “ReconhecedorDeVoz2” será inicializado e assim por diante) e a “Legenda1” será ajustada para retratar o comando a ser dito;

**Figura 15** - Programação genérica de tela de andar (3)



Fonte: Autores (2023)

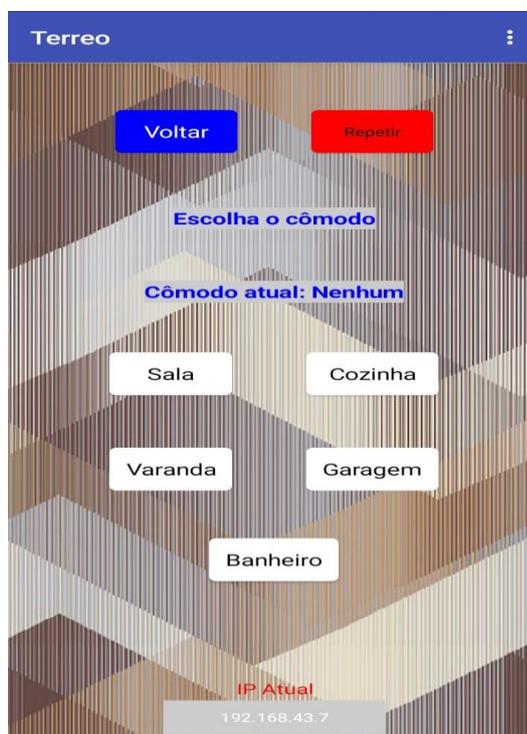
- 6- Para adicionar praticidade ao APP foram incluídos botões para cada cômodo. Cada botão direciona o usuário para o cômodo que o botão foi nomeado, ou seja, inicia o respectivo “ReconhecedorDeVoz” e ajusta as legendas um e dois para indicar as informações necessárias;
- 7- Cada “ReconhecedorDeVoz” possui apenas três resultados possíveis, ligar, desligar e voltar, caso o resultado seja ligar, o botão que representa o cômodo escolhido terá seu fundo alterado para a cor que representa o estado ligado ou desligado (amarelo representa ligado e vermelho representa desligado) e logo após ajustará a URL da Web para “http://” (início comum para toda URL), logo em seguida o IP que foi definido e por fim o comando que será enviado para o microcontrolador (geralmente definido como o nome do cômodo seguido por zero ou um, indicando o comando definido pelo “ReconhecedorDeVoz”, ou seja, se o resultado do “ReconhecedorDeVoz” for “ligar” o comando será

seguido de um, se o resultado for “desligar” o comando será seguido de zero). Portanto se o cômodo for a sala e comando for para ligar, por exemplo, a URL será “http://, o IP escolhido, /sala1”, se o cômodo for a sala e o comando for para desligar a URL será http://, o IP escolhido, /sala0” e este mesmo padrão se repete para todos os cômodos. O fim da comunicação se dá pelo APP requisitando uma resposta do microcontrolador para confirmar que o dado enviado foi recebido corretamente;

- 8- Caso o resultado do “ReconhecedorDeVoz” seja voltar, o APP redirecionará o usuário para a tela principal.

Como exemplo da visualização de uma tela de andar, a Figura 16 representa a organização de uma tela de andar, no entanto a tela em questão foi desenvolvida para o andar térreo.

**Figura 16 - Tela de andar APP**

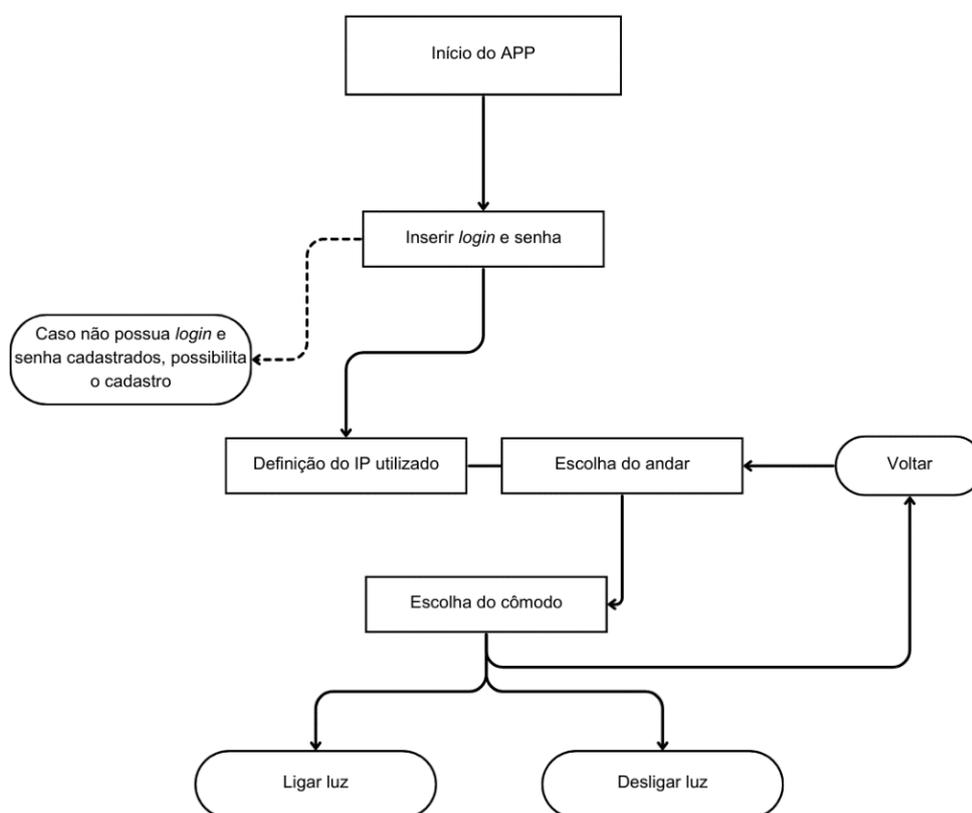


Fonte: Autores (2023)

Nota-se a presença de dois botões ao topo da tela, o primeiro botão representa uma forma de retornar à tela anterior e o segundo representa uma forma de reiniciar a captura de voz, logo abaixo se encontram notas que indicam instruções e indicações para o usuário e botões que representam todos os cômodos do andar, possibilitando a acessibilidade pelo toque ao smartphone, e no final da tela uma caixa de texto foi posicionada para visualização do IP definido atualmente.

Este é o funcionamento detalhado do APP desenvolvido pelos autores, mas para um entendimento geral mais simples foi desenvolvido um fluxograma com todos os passos possíveis, representado na Figura 17.

**Figura 17** - Fluxograma do funcionamento do APP Inventor



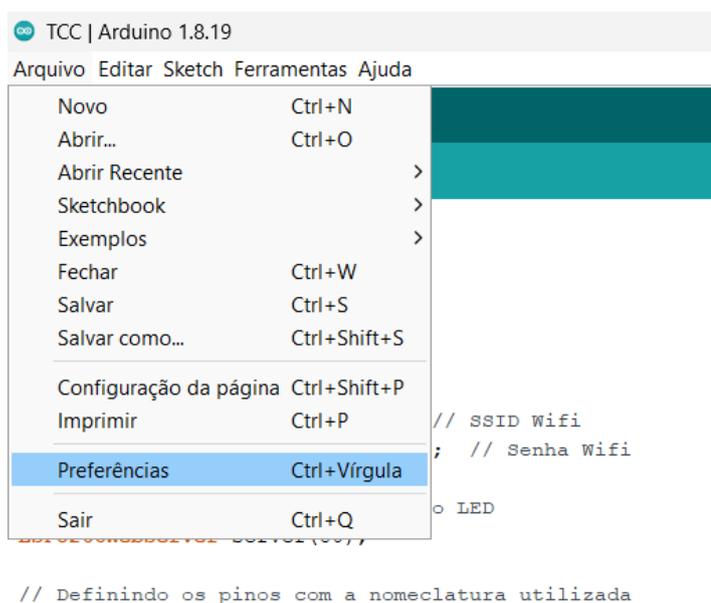
Fonte: Autores (2023)

De agora em diante abordaremos sobre a programação utilizada para a comunicação Web com o microcontrolador ESP8266.

Como mencionado anteriormente, a programação destinada ao microcontrolador foi produzida utilizando o IDE do Arduino, em linguagem C++, porém seu uso é dedicado à programação do próprio microcontrolador Arduino. Com algumas

pesquisas e testes conseguimos uma maneira de utilizar um microcontrolador neste IDE. A Figura 18 representa a opção selecionada no IDE do Arduino para acessar as preferências.

**Figura 18 – Inserindo a URL (1)**



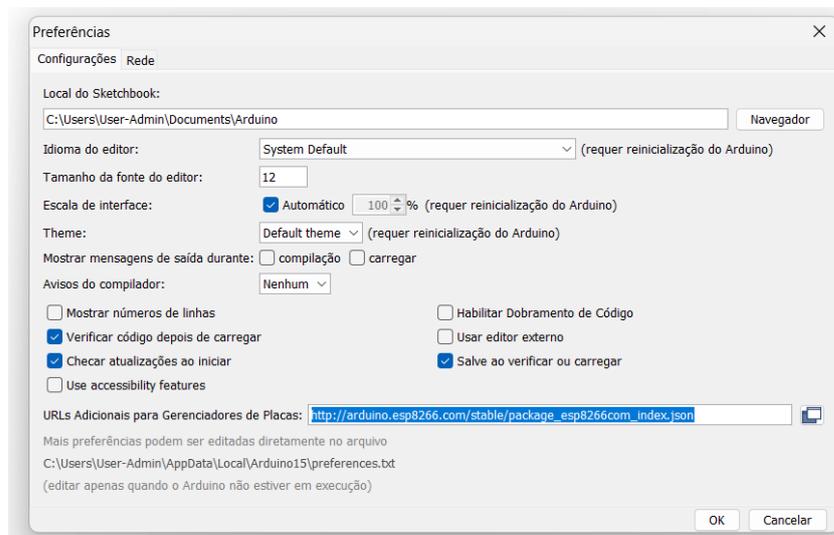
Fonte: Autores (2023)

O primeiro passo para utilizar o IDE com um microcontrolador é definir, em suas preferências, uma URL adicional para o gerenciador de placas, no caso do ESP8266 a URL necessária é:

[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json).

O local no qual a URL é inserida está representado na Figura 19.

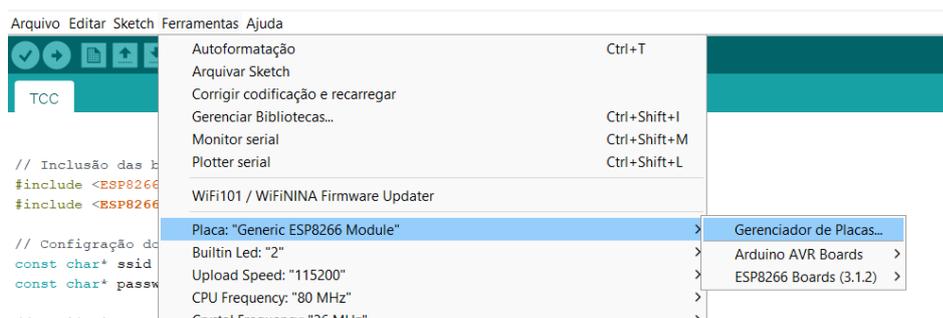
**Figura 19 - Inserindo a URL (2)**



Fonte: Autores (2023)

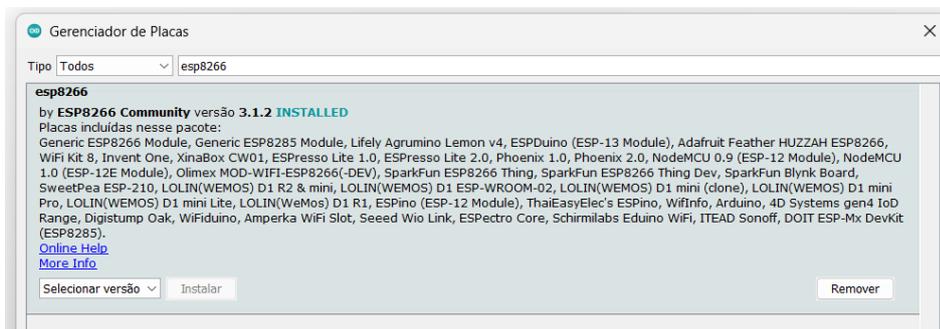
Agora que a URL para o ESP8266 foi configurada no IDE é possível encontrar a placa dentro do gerenciador de placas, como é representado na Figura 20 e Figura 21. A Figura 20 representa a opção selecionada no IDE para acessar o gerenciador de placas e a Figura 21 representa a placa do ESP8266 já instalada no IDE.

**Figura 20 - Instalando a placa (1)**



Fonte: Autores (2023)

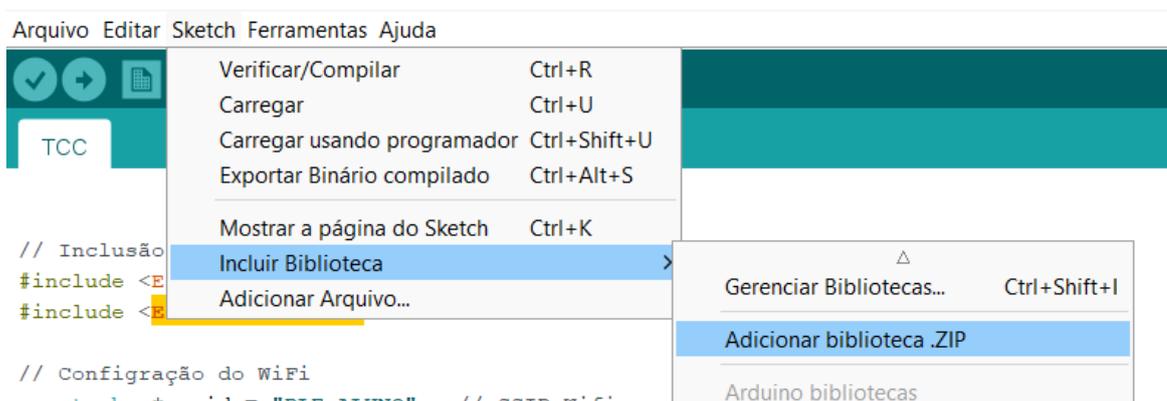
**Figura 21 - Instalando a placa (2)**



Fonte: Autores (2023)

Após instalar a placa podemos começar a programação. Baseado em estudos, descobrimos que para o programa conseguir realizar a comunicação é necessário a instalação de duas bibliotecas no IDE, as bibliotecas utilizadas serão ESP8266WebServer e ESP8266WiFi. Para a instalação das bibliotecas é necessário o *download* pela Internet, após realizar o *download*, dentro do IDE encontra-se uma função para incluir bibliotecas, representada na Figura 22.

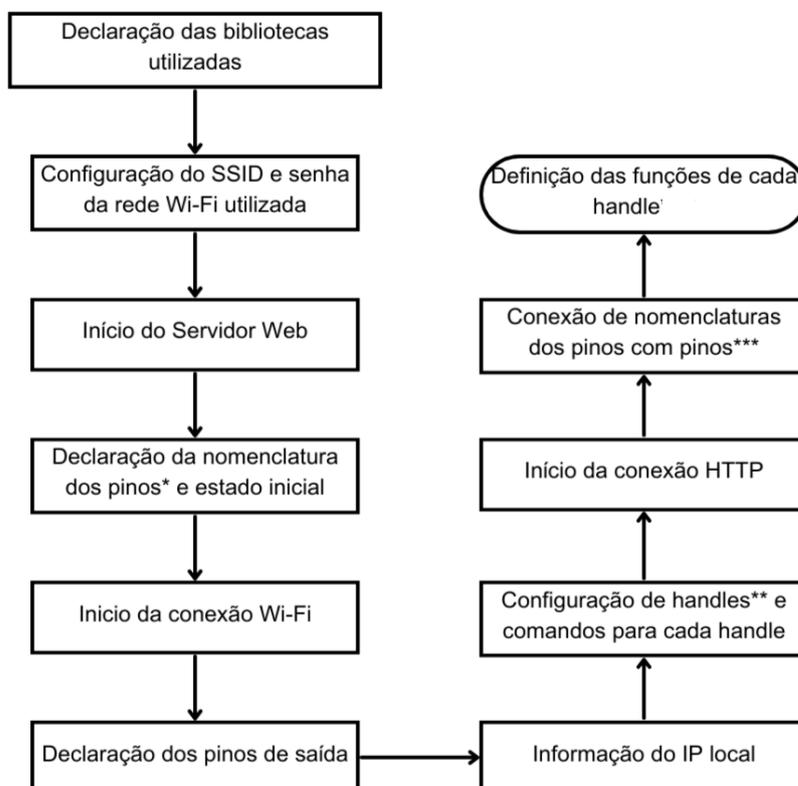
**Figura 22 - Incluir biblioteca**



Fonte: Autores (2023)

Com todas as preparações completas é possível iniciar a programação do microcontrolador. Para um entendimento que não exija conhecer a linguagem de programação C++ produzimos um fluxograma, representado na Figura 23, com os passos utilizados para a programação ser funcional. A programação completa em forma de arquivo .ino, ou seja, representação indicativa do uso do IDE do Arduino, está disponível no **APÊNDICE B – Programação do ESP8266**.

**Figura 23** - Fluxograma da programação do ESP8266



Fonte: Autores (2023)

Para o funcionamento mais intuitivo e como uma forma de identificação mais simples foram usadas algumas nomenclaturas dos pinos (\*), ou seja, ao invés de utilizar os pinos do ESP8266 como sua numeração de GPIO foram utilizadas algumas nomenclaturas definidas pelos autores, como exemplo: “LUZSala”, “LUZCozinha”, etc. Obviamente por conta das nomenclaturas diferentes do padrão é necessário realizar uma conexão entre as nomenclaturas e os pinos (\*\*).

Utilizou-se também *handles* (\*\*), uma forma de identificar e separar funções específicas, e por meio delas também definimos o que cada lâmpada deverá realizar ao receber comandos do APP por meio do IP.

### 3.2. INSTALAÇÃO DO HARDWARE

Compreendendo o funcionamento de todas as programações necessárias, podemos iniciar uma apresentação do *hardware* do protótipo. Caso haja interesse todas as imagens que demonstram a construção do *hardware* estão disponíveis no **APÊNDICE A – Programação do APP**.

Visando a implementação da instalação elétrica de forma mais imersiva e que facilite a visualização do protótipo como um todo foi adquirido uma maquete em *Medium Density Fiberboard (MDF)*, em escala reduzida de 1:25, simulando uma residência com nove ambientes, como representada na Figura 24 - Maquete.

**Figura 24 - Maquete**



Fonte: Autores (2023)

Antes de começarmos as montagens, fizemos um levantamento de preços e materiais que utilizamos no *hardware*, como representa a Tabela 1 - Tabela de Custos.

**Tabela 1 - Tabela de Custos**

Tabela de Custos		
Item	Quantidade	Valor
Módulo relé 4 canais com optoacoplador	1 un.	R\$ 30.00
Módulo relé 8 canais com optoacoplador	1 un.	R\$ 60.00
Transformador 127v/12v	1 un.	R\$ 50.00
Transformador 127v/5v	1 un.	R\$ 10.00
Conector Wago emenda 5 fios	5 un.	R\$ 19.00
Borne de pressão WP4-7	4 un.	R\$ 12.00
Porta fusível	1 un.	R\$ 13.00
Botão chave gangorra redondo	1 un.	R\$ 3.50
Fusível de vidro 1A.	1 un.	R\$ 0.50
Cooler fan 12 vdc 40x40x10 mm	1 un.	R\$ 15.00
Dobradiça de aço	2 un.	R\$ 3.53
Dobradiça porta cadeado	1 un.	R\$ 7.29
Eletroduto corrugado 3/4"	2 mt.	R\$ 5.00
Cabo flexível 1.5 mm <sup>2</sup> Preto/Vermelho/Azul	25 mt.	R\$ 37.50
Conector jumper elétrico	40 un.	R\$ 12.00
Maquete casa 2 andares escala: 1/25	1 un.	R\$ 114.90
Caixa retangular com tampa em acrílico	1 un.	R\$ 44.99
Cabo flexível 3 vias (3x2.5 mm <sup>2</sup> )	2 mt.	R\$ 15.16
Plugue 2P+T 90G 10A.	1 un.	R\$ 8.72
Prensa cabo rosca BSP 1/2"	1 un.	R\$ 6.78
Parafuso rosca máquina com porca e arruela 3.5x40 mm	18 un.	R\$ 8.65
Lâmpada incandescente sinaleiro/abajur 127v 7w	9 un.	R\$ 38.25
NodeMCU Esp8266 30 pinos	1 un.	R\$ 24.99
	<b>TOTAL</b>	<b>R\$ 540.76</b>

Fonte: Autores (2023)

Considerando normas de instalações elétricas, o protótipo se mostrou mais prático e adaptável caso estivesse construído em um formato similar à uma caixa de controle elétrico ou também conhecido como painel elétrico, portanto foi o conceito adotado.

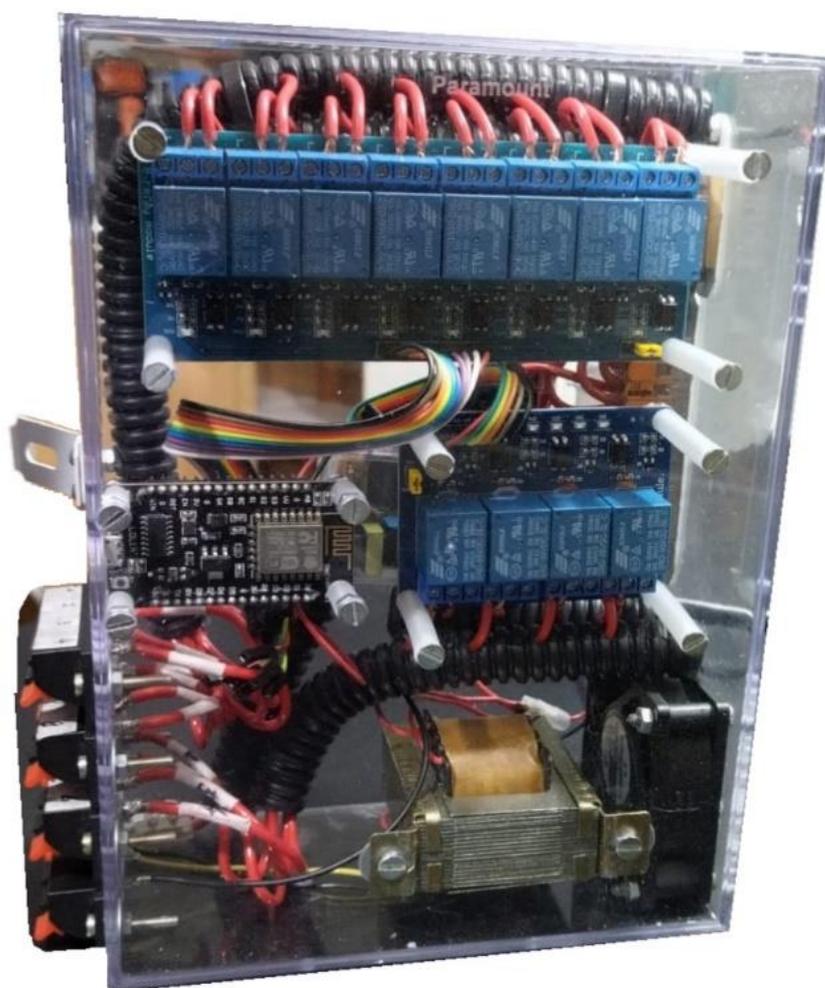
Uma forma que encontramos de simular o protótipo em uma caixa de controle foi utilizando um caixa retangular de acrílico, pois ela seria algo de fácil transporte e que nos permitiria a visualização interna de todos os componentes.

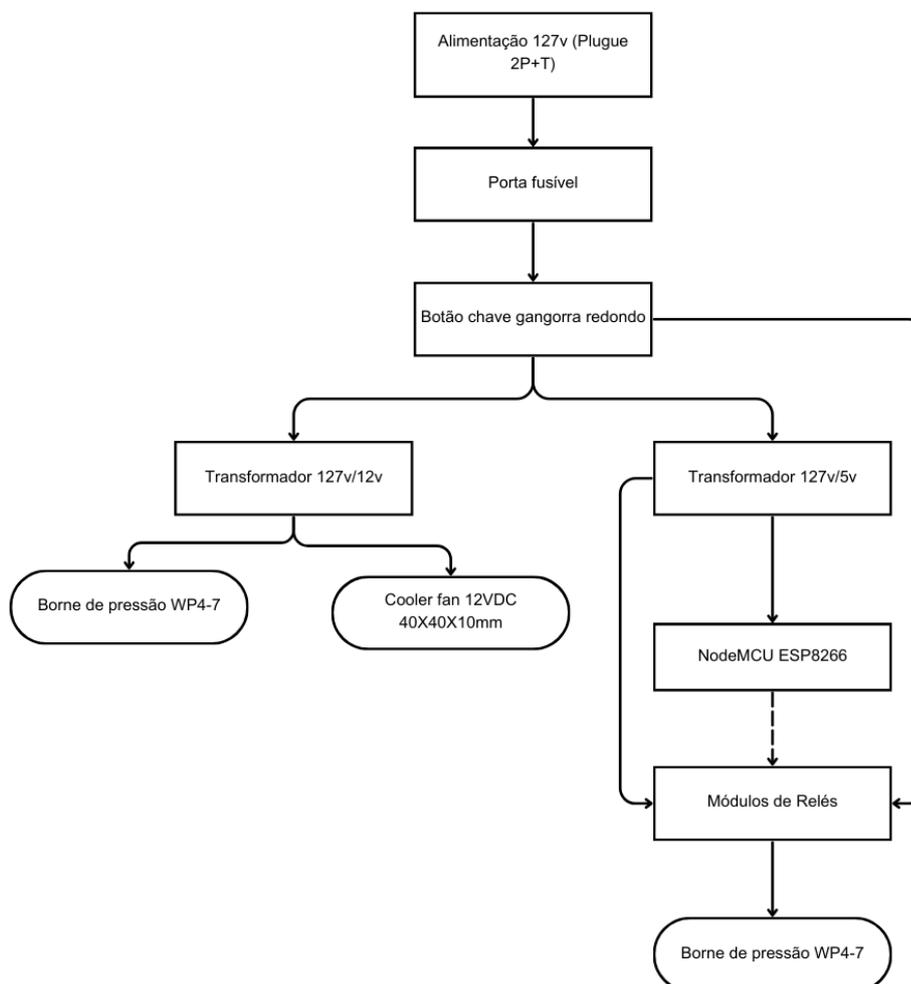
Dentro da caixa posicionamos dois módulos de relés, microcontrolador NodeMCU ESP8266, dois abaixadores de tensão, cooler, conectores de emenda, bornes de pressão, conectores jumpers elétricos, porta fusível, botão chave gangorra e cabos flexíveis. Para o acionamento das lâmpadas, que foram posicionadas dentro da maquete, utilizamos cabos de 1,5 milímetros quadrados (mm<sup>2</sup>) (segundo a norma de iluminação de interiores NBR 5413) que serão conectados nos bornes de pressão posicionados nas laterais da caixa de controle.

No circuito completo do protótipo, cada lâmpada em funcionamento possui 7 Watts, como a tensão aplicada a cada lâmpada é equivalente a 127VAC é necessário o uso da Lei de Ohm. A Lei de Ohm diz que para encontrar uma corrente é necessário dividir a potência da carga pela tensão da carga, simulando no circuito existente: A corrente do circuito é igual a 7 Watts divididos por 127VAC, ou seja, aproximadamente 0,055 Amperes (A), ou 55 Milliampères (mA) para cada lâmpada, considerando a existência de nove lâmpadas no circuito o resultado é multiplicado por nove, resultando em aproximadamente 0,496A, ou 496mA, portanto, o fusível escolhido para a proteção do circuito foi um fusível de vidro de 1A.

A Figura 25 apresenta a vista frontal da caixa de controle e o fluxograma da Figura 26 destaca a ligação de todos os componentes.

**Figura 25 - Vista frontal da caixa de controle**



**Figura 26 - Alimentação caixa de controle**

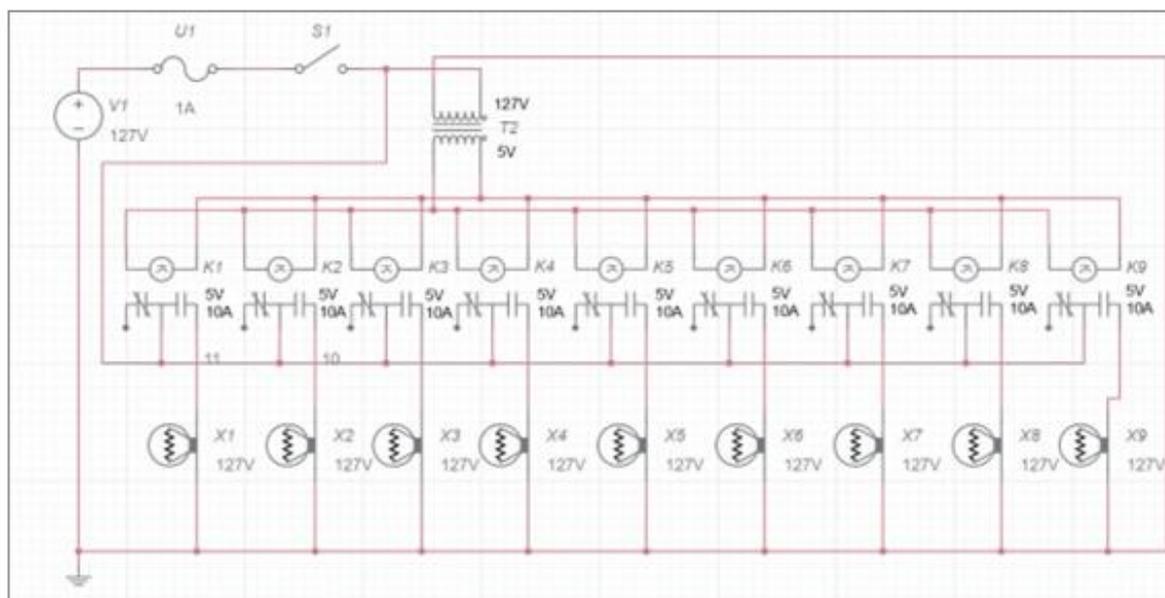
Fonte: Autores (2023)

No fluxograma percebe-se que os módulos de relé são energizados por duas fontes, o NodeMCU ESP8266 apenas controlará se o relé estará ligado ou desligado, os 5VDC energizarão os relés e os 127VAC serão responsáveis pelas saídas, ou seja, o relé fechará a chave para que a tensão 127VAC passe para os bornes de saída. Portanto, unindo a programação do APP e do microcontrolador, é possível gerenciar a iluminação estabelecida na residência com o APP controlando o funcionamento dos relés.

A Figura 27 representa o esquema elétrico do protótipo. Devido às limitações de representação, o esquema elétrico não possui relé com optoacoplamento e entradas

para o microcontrolador utilizado. A Figura 28 representa uma exemplificação da ligação do microcontrolador ao relé com optoacoplador.

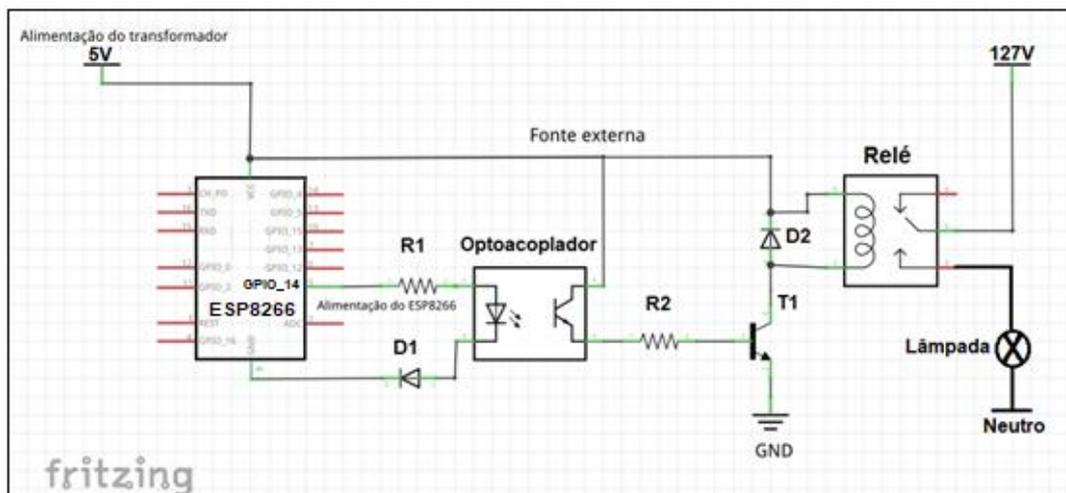
**Figura 27 - Esquema elétrico**



Fonte: Autores (2023)

Considerando o esquema elétrico, todo o circuito é alimentado por uma fonte de 127VAC (V1) e protegido por um fusível de 1A (U1) e um botão chave (S1). A tensão de 127VAC é dividida em dois caminhos, em um dos caminhos a tensão é direcionada à um abaixador de tensão (T2) que gera uma tensão de 5VDC utilizada na alimentação dos relés (K1 à K9), o controle da saída dos relés é realizado por meio do acionamento do optoacoplador controlado pelo ESP8266, como representado na Figura 28. O outro caminho da tensão de 127VAC é direcionado à entrada comum do relé (localizado entre as saídas normalmente aberta e normalmente fechada), portanto, quando o microcontrolador aciona a saída normalmente aberta a tensão de 127VAC é direcionada para a lâmpada (X1 à X9).

**Figura 28 - Conexão ESP8266 / Relé**



Fonte: Autores (2023)

Nota-se que, o relé é responsável pelo controle de uma lâmpada posicionada na maquete, portanto, ao utilizarmos nove lâmpadas também utilizamos nove relés, para realizar o acionamento individual de cada ponto. Considerando o uso de diversos relés foi elaborado um quadro, representado pelo Quadro 2, que identifica a interligação entre os pinos GPIO do microcontrolador, o módulo de relé utilizado e o cômodo em que a lâmpada foi instalada. É importante ressaltar que o ESP8266 está posicionado entre a tensão de 5VDC gerada pelo transformador e o relé.

**Quadro 2 - Interligações**

Interligações				
Pinos ESP8266		Módulo relé 8 canais	Módulo relé 4 canais	Cômodos
D1	5	1		Sala
D2	4	2		Cozinha
D3	0	3		Garagem
D4	2	4		Varanda
D5	14	5		Escada
D6	12	6		Quarto 1
D7	13	7		Quarto 2
D0	16	8		Banheiro 1
RX	3		1	Banheiro 2

Fonte: Autores (2023)

### 3.2.1. SIMULAÇÃO DE APLICAÇÃO REAL

Para uma visualização do protótipo em uma aplicação real, foi necessário o planejamento de custos e materiais que seriam utilizados durante a instalação. Inicialmente a interligação da instalação real será planejada por meio de cabos, porém com estudos feitos durante o planejamento chegamos a uma ideia de utilizarmos um módulo de relés com um ESP-12F integrado para comunicar toda a instalação em diversos ambientes sem a necessidade de instalar cabos adicionais. Considerando o uso do protótipo em uma instalação real, será necessário realizar um planejamento de pontos de iluminação. Com o uso da norma ABNT NBR 5410 é possível calcular a quantidade de pontos de iluminação necessária para cada ambiente baseado na sua área. A Figura 29 representa uma residência “padrão”, composta por dois dormitórios, banheiro, corredor, sala de estar e jantar, cozinha e área de serviço, neste exemplo teríamos 8 pontos de automatização.

**Figura 29 - Residência Padrão**



Fonte: Autores (2023)

Conforme a norma NBR 5410, para o cálculo da carga de iluminação é feita em função da área do cômodo, para área igual ou inferior a 6 m² atribui-se um mínimo de 100VA, para áreas maiores que 6 m² atribui-se 100VA e para cada aumento de 4 m² inteiros 60VA são somados. (PRYSMIAN, 2016)

Neste exemplo de residência teríamos 7 ambientes com carga de iluminação de 100VA e no ambiente da Sala de Estar / Jantar com 220VA considerado pela área de 16,8 m<sup>2</sup>.

Na Tabela 2 apresentamos uma simulação de um orçamento da implementação do protótipo na residência padrão.

**Tabela 2** - Tabela de custo unitário do projeto.

<b>Tabela de Custo de Serviço</b>		
<b>Itens</b>	<b>Quantidade</b>	<b>Valor</b>
NodeMCU Esp8266 30 pinos	1 un.	R\$ 24.99
Módulo relé 8 canais com optoacoplador	1 un.	R\$ 42.86
Cabo flexível 1.5 mm <sup>2</sup>	46 mt.	R\$ 69.00
Interruptor três vias (paralelo)	8 un.	R\$ 99.20
Fonte de alimentação 127v / 5v 2.5A	1 un.	R\$ 19.99
Botão chave gangorra redondo	1 un.	R\$ 3.50
Fusível de vidro 1A	1 un.	R\$ 0.50
Porta fusível	1 un.	R\$ 13.00
Conector jumper elétrico	12 un.	R\$ 5.00
Terminal elétrico tubular 1.5 mm <sup>2</sup>	50 un.	R\$ 2.50
Tubo termo retrátil 1.5 mm <sup>2</sup>	1 mt.	R\$ 2.00
Caixa retangular com tampa em acrílico	1 un.	R\$ 20.00
Serviço (Instalação e programação)		R\$ 250.00
	<b>TOTAL</b>	R\$ 552.54

Fonte: Autores (2023)

O preço calculado baseado nos componentes é apenas uma referência, pois os componentes não estão considerados nas medidas exatas de uma instalação real, afinal a realização e esquematização dos componentes necessários ficam à critério do consumidor, ou seja, o consumidor definirá quantas unidades de cada componente ele julgará necessário.

Para este cálculo consideramos o uso de componentes necessários para a instalação dentro da residência padrão. Segundo a fundação de proteção e defesa do consumidor existe uma margem permitida de lucro baseada em cada tipo de empresa, considerando que o trabalho realizado será na área de serviços, o lucro permitido é de até 20% do valor total, portanto, se estipularmos um lucro de 18% o valor final será de R\$652,00.

O preço final pode ser diferente com o uso de mais ou menos pontos de iluminação automatizados e dependendo da escolha do consumidor de uso de

interruptores ou não, caso os interruptores presentes na instalação não sejam paralelos e o consumidor optar pela opção de controle físico e virtual, os mesmos deverão ser substituídos. Desta forma é possível gerar um padrão de preço baseado na quantidade de componentes utilizados e pontos automatizados em cada cômodo, como representado na Tabela 3.

**Tabela 3 – Custo padronizado em função de área e potência.**

Padrão de custo por cômodo					
Até 6m <sup>2</sup> (100VA)			A cada 4m <sup>2</sup> inteiros adicionais (+60VA)		
Item	Quantidade	Valor	Item	Quantidade	Valor
Módulo de Relé unitário	1 un.	R\$ 5.35	Módulo de Relé unitário	1 un.	R\$ 5.35
Interruptor paralelo (Opcional)	1 un.	R\$ 12.40	Interruptor adicional (Opcional)	1 un.	R\$ 5.00
Cabo flexível 1.5 mm <sup>2</sup>	1 mt.	R\$ 1.50	Cabo flexível 1.5 mm <sup>2</sup>	1 mt.	R\$ 1.50
Terminal elétrico tubular 1.5 mm <sup>2</sup>	8 un.	R\$ 0.40	Terminal elétrico tubular 1.5 mm <sup>2</sup>	6 un.	R\$ 0.30
Tubo termo retrátil 1.5 mm <sup>2</sup>	0,5 mt.	R\$ 1.00	Tubo termo retrátil 1.5 mm <sup>2</sup>	0,5 mt.	R\$ 1.00
	<b>TOTAL</b>	<b>R\$ 20.65</b>		<b>TOTAL</b>	<b>R\$ 13.15</b>

Fonte: Autores (2023)

Para compreender a tabela é necessário entender a forma em que a NBR 5410 classifica a divisão de pontos de iluminação. Um cômodo com até 6 m<sup>2</sup>, segundo a norma, necessita de um ponto de iluminação, representado na tabela por um módulo de relé unitário, este ponto de luz pode ser controlado virtualmente e, caso o consumidor solicite o controle físico, a ligação paralela entre o módulo de relé e o interruptor possibilita o controle. Toda a ligação elétrica dos componentes é feita por meio de cabos flexíveis de 1,5 mm<sup>2</sup>, a metragem necessária só é dimensionada no momento da passagem dos cabos nos eletrodutos da instalação, portanto, demonstramos o valor correspondente a um metro. Cada conexão correspondente necessita de um terminal elétrico para evitar danos à seção do cabo e um tubo termo retrátil para realizar um melhor isolamento.

Caso o cômodo ultrapasse os 6 m<sup>2</sup> um segundo ponto de iluminação se torna necessário, logo um novo módulo de relé e um interruptor serão adicionados mediante a necessidade do consumidor em controlar individualmente cada ponto, se o consumidor não necessitar controlar cada ponto individualmente o relé e o interruptor não serão adicionados, realizando o acionamento por um único interruptor.

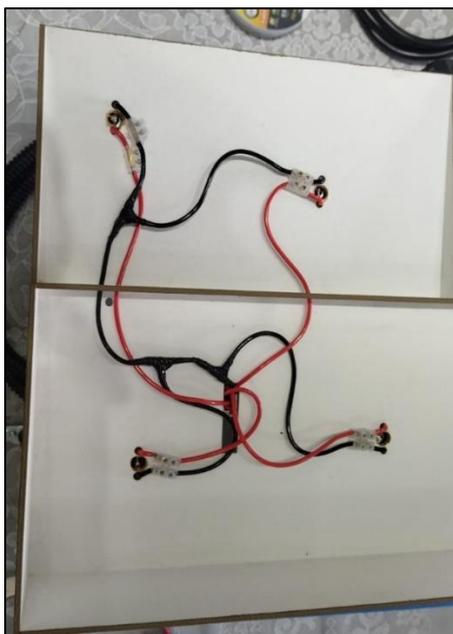
## 4. TESTES

Durante a evolução dos estudos, foram conciliados testes práticos para evitar empecilhos na realização das montagens. A inserção dos testes será realizada de forma cronológica aos acontecimentos.

O primeiro teste realizado foi referente aos módulos de relé, onde foi simulado o comando para ligar e desligar o sinalizador de funcionamento do módulo, que contém um Diodo emissor de luz (LED). Após a comprovação de funcionamento dos 12 módulos foram implementadas lâmpadas e testadas uma a uma, comprovando também o funcionamento da tensão de 127VAC.

Logo após foi iniciada a instalação elétrica dentro da maquete, passando cabos e instalando lâmpadas, como representa a Figura 30.

**Figura 30** - Cabos de energia posicionados



Fonte: Autores (2022)

Foi necessário testar se a instalação dos cabos estava correta. Portanto, foram ligadas algumas lâmpadas para demonstrar o funcionamento, representadas na Figura 31 e Figura 32, que, respectivamente, indicam o teste do térreo e do primeiro andar.

**Figura 31** - Teste térreo (sala)



Fonte: Autores (2022)

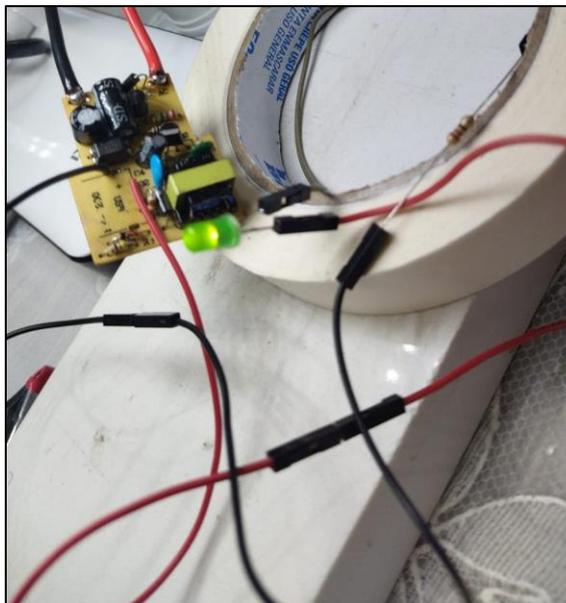
**Figura 32** - Teste primeiro andar (quarto)



Fonte: Autores (2022)

A instalação elétrica da maquete está finalizada, então pode ser começada a montagem da caixa de controle. A caixa de controle necessitou de testes em todos seus componentes antes de serem posicionados, sendo eles: Transformadores, cooler, conectores, bornes, ESP8266 e cabos. A Figura 33 representa o teste realizado

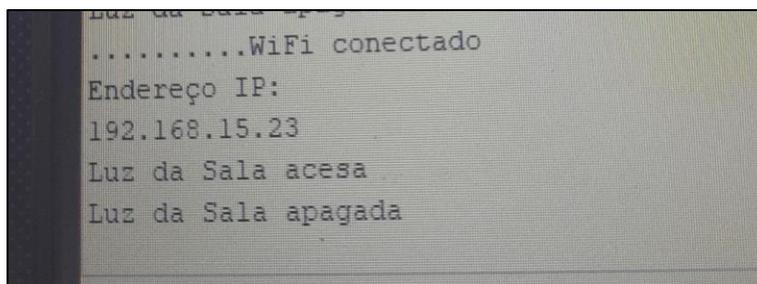
**Figura 33** - Teste de funcionamento transformador 127VAC/5VDC



Fonte: Autores (2022)

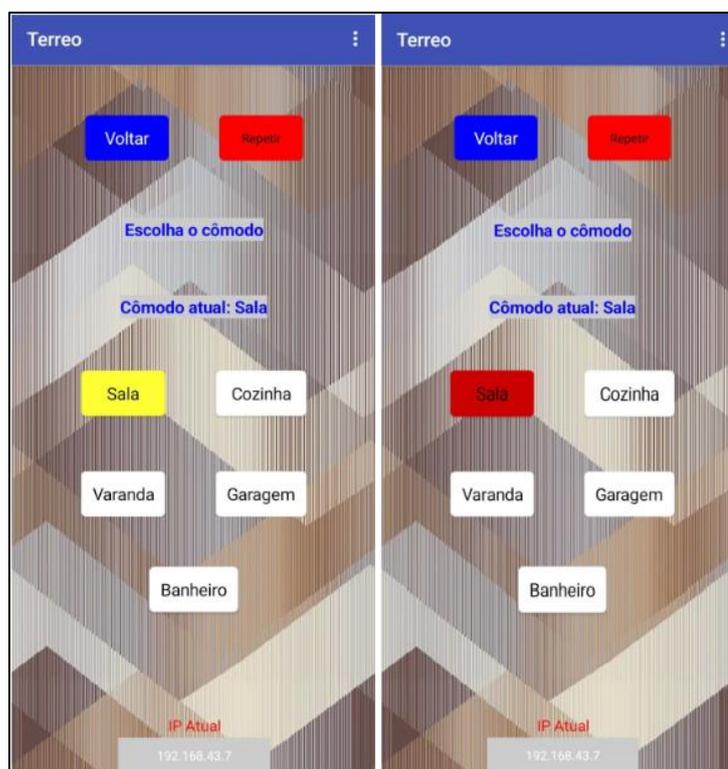
Por exigir a caixa de controle montada, a programação começou a ser desenvolvida após este ponto. A primeira programação utilizada como teste não funcionou como esperado, acredita-se que por falha entre a comunicação do ESP8266 e a rede Wi-Fi, então outras comunicações foram testadas, como exemplo a comunicação UDP (sugerida pelo orientador do trabalho), assim como houve testes utilizando *MicroPython* como linguagem de programação. No entanto, todos os testes realizados com a programação até então foram falhos, ou não se comunicavam da maneira esperada. Enquanto estavam sendo testados modos de comunicação por Wi-Fi com o ESP8266, houve orientações para a necessidade de uma proteção ao circuito e os componentes localizados na caixa de controle, então por meio de cálculos, houve um dimensionamento de um fusível para o circuito. O fusível foi posicionado na caixa de controle por meio de um porta fusível logo após a entrada da tensão, e foi seguido por um botão de liga/desliga.

Ao realizar testes utilizando a conexão Wi-Fi, o protocolo HTTP e TCP/IP, foi possível confirmar a comunicação por meio de comandos enviados pelo IP do dispositivo, como representa a Figura 34.

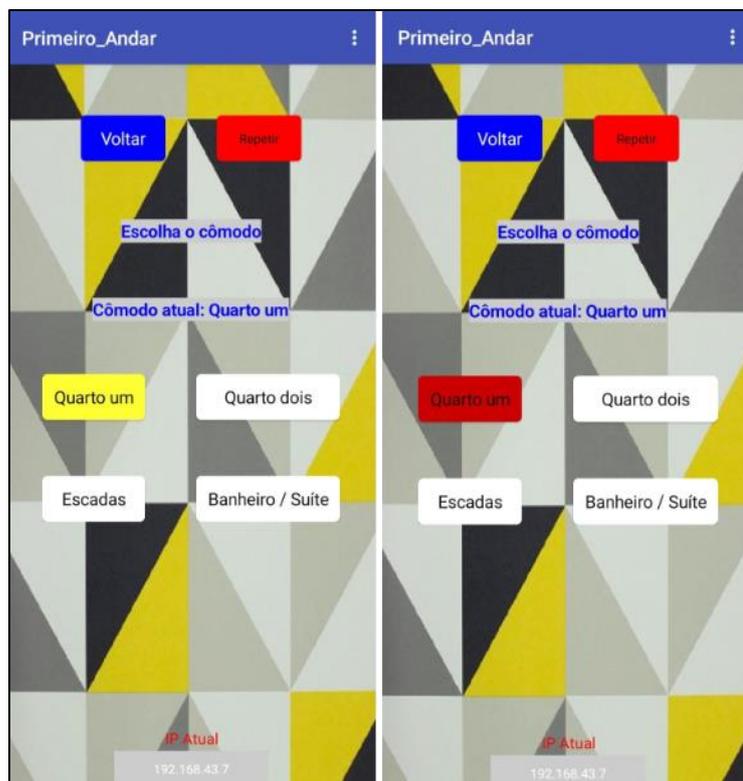
**Figura 34** - Primeira comunicação funcional sinalizada pelo serial

Fonte: Autores (2023)

Com o funcionamento da comunicação entre o APP e o ESP8266 foi apresentada a necessidade de indicar o estado atual da lâmpada, para caso o usuário não conseguisse observar o estado da lâmpada e para confirmar o funcionamento da comunicação. Desta forma quando a lâmpada estiver acesa o botão referente ao ambiente muda seu fundo para amarelo, quando a lâmpada estiver apagada o botão referente ao ambiente muda seu fundo para vermelho. A Figura 35 e a Figura 36 representam, respectivamente, o funcionamento da lâmpada da sala, localizada no térreo, e o funcionamento da lâmpada do quarto um, localizada no primeiro andar.

**Figura 35** - Teste térreo APP

Fonte: Autores (2023)

**Figura 36 - Teste primeiro andar APP**

Fonte: Autores (2023)

## 5. CONCLUSÃO

Esse projeto proporcionou o desenvolvimento da proposta eletrônica juntamente com a montagem e pintura de uma maquete em MDF, o que consumiu um tempo considerável do projeto. Após a montagem da maquete foram realizadas algumas modificações para trabalhar com a tensão de 127VAC, mesmo a maquete sendo projetada para realizar a automatização utilizando a tensão de 5VDC. A maquete permitiu uma visão mais ampla dos espaços de trabalho para a implementação do projeto. Considerou-se a inserção de alguns sensores, como exemplo: sensores de presença, sensores *Radio Frequency Identification* (RFID) e sensores de temperatura, além de um servomotor, com finalidade de aumentar a quantidade de recursos disponíveis no protótipo. No entanto por conta de imprevistos, esses elementos adicionais não foram implementados.

A programação e a montagem da caixa de controle necessitaram uma atenção maior, o que inviabilizou a implementação do restante dos *hardwares*. Além do imprevisto relacionado à organização física, o tamanho da caixa de controle adquirida foi um ponto preocupante, pois durante a montagem eletroeletrônica dos componentes observou-se que o espaço da caixa era mais reduzido do que inicialmente previsto. Naturalmente, um pré-projeto em *softwares* de CAD poderiam ter minimizado esses problemas de layout e distribuição.

### 5.1. PROPOSTAS FUTURAS

Considerando este protótipo como base, é possível realizar algumas evoluções em seu funcionamento, instalação e componentes utilizados, considerando propostas futuras, pode-se citar:

- Implementação de botões dentro do APP, para adicionar praticidade e atender requisitos de pessoas com necessidades especiais;
- Implementação de sensores de presença, para monitorar áreas externas;
- Implementação de sensores de temperatura e umidade, para monitorar superaquecimentos ou resfriamentos extremos no ambiente;
- Implementação de sensores RFID, para abertura dos portões de entrada e portas da residência;

- Implementação do controle de servomotor e futuramente de um motor elétrico, para o funcionamento do portão da garagem;
- Implementação de um sistema de alarme com trava para funções do APP, para evitar que invasores consigam controlar o APP;
- Aplicação do *Nobreak*, como alternativa à falta de energia elétrica;
- Instalação de iluminação externa;
- Testes e montagem em uma residência real, com a utilização de uma TUE (Tomada de uso específico) para a alimentação da caixa de controle;
- Interligação entre os interruptores da residência e os módulos de relé;
- Intercomunicação entre o ESP8266 e módulos Wi-Fi ESP-12F adicionais, para expandir a quantidade de portas utilizáveis e facilitar a instalação;
- Possibilitar o controle do protótipo fora de rede local;
- Desenhar em AutoCAD os conjuntos elétricos e eletrônicos para avaliar o espaço físico necessário para acomodar o projeto numa caixa.

## 6. REFERÊNCIAS

BALDASSIN, A. **Comunicação serial (UART)**. UNESP-Rio Claro, 2019. Disponível em: <http://www1.rc.unesp.br/igce/demac/alex/disciplinas/MicroII/EMA864315-Serial.pdf>. Acesso em: 17 mai. 2023.

**Brasfort**. Disponível em: <https://brasfort.net.br/produto/lampada-sinaleiro/#> . Acesso em: 13 mai. 2023.

BRITO, SAMUEL HENRIQUE B. **Como o sinal Wi-fi é propagado na natureza?** Disponível em: <https://medium.com/ubntbr/como-o-sinal-wifi-%C3%A9-propagado-na-natureza-d87daef39575>. Acesso em: 03 out. 2022.

**Como funciona um servidor HTTP**. Metr pole Digital, aula03. Disponível em: <https://materialpublic.imd.ufrn.br/curso/disciplina/3/78/3/2>. Acesso em 22 mai. 2023.

**Como funcionam as assistentes virtuais**. Ilumeo, 2019. Disponível em: <https://ilumeo.com.br/todos-posts/2019/10/28/como-funcionam-as-assistentes-virtuais>. Acesso em: 10 mai. 22.

FIELDING, H. et.al. **Hypertext Transfer Protocol – HTTP/1.1**. Network Working Group, 1999. Disponível em: <https://datatracker.ietf.org/doc/html/rfc2616#page-7> Acesso em: 15 mai.2023.

FERREIRA, Victor Zago Gomes, **A Dom tica como instrumento para a melhoria da qualidade de vida dos portadores de defici ncia**, Trabalho de Conclus o de Curso apresentado ao Instituto Federal de Educa o, Ci ncia e Tecnologia da Para ba, 2010.

FOROUZAN, B. A. **Protocolo TCP/IP**. 3<sup>a</sup>. Ed. [S.l.]: AMGH, 2008.

GOURLEY, David. et al. **HTTP: The Definitive Guide**. O'Reilly Media. lc, 2002.

GUIMAR ES, F bio de S. **SPI**. Disponível em: <https://mundoprojetado.com.br/spi/>. Acesso em: 27 mai. 2023.

**IBM Shoebox**. IBM, 2003. Disponível em: [https://www.ibm.com/ibm/history/exhibits/specialprod1/specialprod1\\_7.html](https://www.ibm.com/ibm/history/exhibits/specialprod1/specialprod1_7.html). Acesso em: 10 mai. 22.

KOYANAGI, Fernando. **ESP8266 utilizando PWM com Potenci metro**. 2017b. Disponível em: <https://www.fernandok.com/2017/10/esp8266-utilizando-pwm-com-potenciometro.html#:~:text=Para%20quem%20n%C3%A3o%20%C3%A9%20acostumado,um%20driver%20de%20um%20motor>. Acesso em: 07 abr. 2023.

KOYANAGI, Fernando. **Introdu o ao ESP8266**. 2017a. Disponível em: <https://www.fernandok.com/2017/10/introducao-ao-esp8266.html>. Acesso em: 07 abr. 2023.

KUROSE, James F. **Redes de computadores e a Internet: uma abordagem top down**.(5<sup>a</sup> ed.). S o Paulo: Pearson, 2010. Disponível em: <https://www.facom.ufu.br/~sequincozes/referencias/kurose2010.pdf> Acesso em: 15 mar. 2023.

LUGLI, Alexandre B.; SANTOS, Max M. D. **Redes sem fio para Automa o Industrial**. 1 ed. S o Paulo:  rica, 2014.

**Módulo relé 5v 10A 1 canal com optoacoplador.** Disponível em: <https://smartkits.com.br/modulo-rele-5v-10a-1-canal-com-optoacoplador>. Acesso em: 29 mai. 2023.

**Nodemcu pinout.** Circuits4you.com. Disponível em: <https://circuits4you.com/2017/12/31/nodemcu-pinout/>. Acesso em: 07 abr. 2023.

PRYSMIAN. **Instalações Elétricas Residenciais.** Sorocaba, 2016. Disponível em: [https://br.prysmiangroup.com/sites/default/files/atoms/files/Manual\\_Prysmian\\_1\\_0%20%281%29.pdf](https://br.prysmiangroup.com/sites/default/files/atoms/files/Manual_Prysmian_1_0%20%281%29.pdf). Acesso em: 14 jun. 2023.

RUFINO, Nelson M. de O. **Segurança em Redes sem Fio:** Aprenda a proteger suas informações em ambientes Wi-Fi e Bluetooth. (2ª edição). São Paulo: Novatec Editora, 2007.

SANTOS, Sandro. **Introdução à IoT:** Desvendando a Internet das Coisas. [S.I.]: SS Trader Editor, 2018. p. 21.

**Setor de automação residencial projeta crescimento superior a 20% neste ano.** Folha da Região, 2021. Disponível em: <https://www.folhadaregiao.com.br/2021/09/12/setor-de-automacao-residencial-projeta-crescimento-superior-a-20-neste-ano/>. Acesso em: 05 mai. 2022.

SILVEIRA, Sidnei Renato, RIBEIRO, Vinicius Gadis, MARTINS, Márcio Machado, **Uma solução de baixo custo para implementação de Domótica,** em Revista de Sistemas e Computação, Salvador, v. 4, n.2, p 126-133, dezembro 2014.

**Sixty Degrees,** 2021. Disponível em: <https://sixty-degrees.com/weekly-note-20210329-inteligencia-artificial/>. Acesso em: 18 mai. 2022.

TENENBAUM, A. S. **Redes de Computadores.** 5ª. Ed. [S.I.]: Pearson, 2011.

## 7. APÊNDICE

### APÊNDICE A – Programação do APP

Todas as imagens da programação em blocos estão disponíveis no link: [https://github.com/m9sant/Casa\\_Automatica/tree/APP-Inventor-Programa](https://github.com/m9sant/Casa_Automatica/tree/APP-Inventor-Programa)

### APÊNDICE B – Programação do ESP8266

O arquivo .ino com a programação utilizada no microcontrolador ESP8266 está disponível no link: [https://github.com/m9sant/Casa\\_Automatica/tree/main](https://github.com/m9sant/Casa_Automatica/tree/main)

### APÊNDICE C – Montagem do protótipo

Para o início do projeto do protótipo foi necessário a aquisição de uma maquete em MDF de uma casa de dois andares na escala de 1:25, para assim termos uma visão melhor de dimensionamento para a parte elétrica e eletrônica. Os primeiros passos no desenvolvimento do projeto foi a montagem e pintura da maquete.

#### Montagem



Montagem do primeiro andar



Montagem do segundo andar



Montagem do teto



Montagem finalizada da maquete

## Pintura de fundo



*Laje do primeiro andar e segundo andar*



Laje do segundo andar e telhado



Maquete pintura de fundo concluída



Fita crepe para fazer as molduras de portas, janelas e aplicação da cor final nas paredes

### Pintura final



Frontal



Traseira



Perspectiva

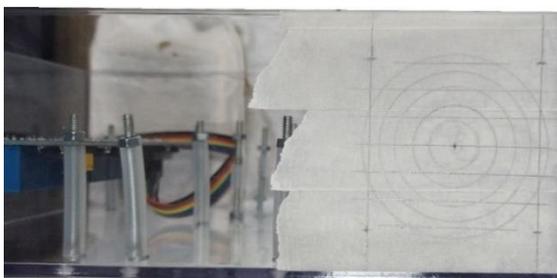
### Caixa de controle: montagem dos componentes elétricos e eletrônicos



Caixa retangular com tampa em acrílico



Instalados módulos de relés, NodeMCU e transformador 127v/12v



Gabarito para furos de fixação do cooler e entrada de ar



Executando os furos



Furos de fixação e entrada de ar finalizados



Gabarito para a execução dos furos de fixação dos bornes de pressão WP4-7



Furos de fixação de bornes realizada



Bornes instalados



Processos para a execução da soldagem dos bornes



Montagem da alimentação do módulo de relés



Vista traseira da caixa de controle finalizada