

UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

## **Automação Residencial Utilizando IoT**

Trabalho de Conclusão de Curso

**Leones Moura dos Santos**

São Cristóvão – Sergipe

2017

UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Leones Moura dos Santos

## **Automação Residencial Utilizando IoT**

Trabalho de Conclusão de Curso II submetido ao Departamento de Engenharia Elétrica da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Engenharia Eletrônica.

Orientador: Prof. Dr. Antonio Ramirez Hidalgo

São Cristóvão – Sergipe

2017

**Leones Moura dos Santos**

## **Automação Residencial Utilizando IoT**

Trabalho de Conclusão de Curso II submetido ao Departamento de Engenharia Elétrica da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Engenharia Eletrônica.

Trabalho aprovado. São Cristóvão – Sergipe, 22 de Setembro de 2017:

---

**Prof. Dr. Antonio Ramirez Hidalgo**  
Orientador

---

**Prof. Dr. Carlos Alberto Villacorta Cardoso**  
Convidado

---

**Prof. Me. Felix Estevam de Jesus Brito**  
Convidado

São Cristóvão – Sergipe  
2017

# Agradecimentos

...

*As invenções são, sobretudo,  
o resultado de um trabalho teimoso.  
(Santos Dumont)*

# Resumo

Com a evolução da automação em geral, principalmente nas áreas de telecomunicações e microeletrônica, tornou-se possível o desenvolvimento de aplicações que dão conforto, segurança e aumentam a qualidade de vida das pessoas em suas próprias residências. O acesso remoto à informação é uma característica não só desejável como também, na maioria das vezes, necessária, seja para segurança, controle de eletrodomésticos, controle de iluminação entre tantos outros. É notável a carência de tecnologias modernas nas áreas residenciais, porém este paradigma vem sendo mudado aos poucos. A automação industrial e o avanço das telecomunicações foram as responsáveis por favorecer a evolução da domótica. Com o suporte da atual tecnologia de internet, a aplicação da mesma, para o controle remoto e comunicação entre as "coisas", está tomando seu lugar no universo da automação residencial. A domótica, utilizando o padrão de comunicação sem fio (WiFi), está ocupando um espaço considerável no mercado e mudando o cenário da automação nas residências. Sendo assim, estamos prestes a entrar em uma nova era tecnológica, na qual áreas de automação, telecomunicações, construção civil e arquitetura residencial estarão unidas em uma só. Neste trabalho, é desenvolvido um projeto para automatização de uma residência utilizando a "internet das coisas", para controle de iluminação, tomadas, eletrodomésticos e um pequeno sistema de segurança. Ao final é exposto os resultados obtidos e os principais tópicos para futuras aplicações no mercado e na ciência.

**Palavras-chave:** Domótica, IoT, WiFi, Controle, Automação Residencial.

# Abstract

With the evolution of automation in a general manner, especially in areas such as telecommunications and microelectronics, it has become possible to develop several applications that provide comfort, safety and improve people's quality of life in their own residences. The remote access to information is not only desirable, but mostly, a necessary characteristic for security, home appliance control, illumination control, among others. It is noticeable the lack of modern technologies in residential areas, however, this paradigm has been changing gradually. Industrial automation as well as the advancement of telecommunication were responsible for favoring the evolution of home automation. With the support of the current Internet technology, the application of it for remote control and communication among "things" is taking its place in the universe of residential automation. The home automation, utilizing the pattern of wireless communication (WiFi), is occupying a considerable space in the market and changing the residential automation scenario. Therefore, we are about to enter a new technological era in which automation, telecommunications, civil construction, and residential architecture areas will be united in one. In this work, a project for the automation of a residence be developed using the "Internet of Things" to control illumination, plugs, home appliances, and a small security system. At the end is exposed the results obtained and the main topics for future applications in the market and science.

**Keywords:** Dómotica, IoT, WiFi, Control, Home automation.

# **Lista de ilustrações**

Figura 1 – <i>Gráfico da cobertura de redes móveis e tecnologias em evolução</i> (SANOU, 2016).	2
Figura 2 – <i>Estrutura da Automação Residencial</i> (BOLZANI, 2004).	5
Figura 3 – <i>Adaptação de Iluminação em um Ambiente</i> (QUINDERÉ, 2009).	7
Figura 4 – <i>Ilustração de objetos conectados pela IoT</i> (APLEX, 2016).	9
Figura 5 – <i>Evolução do número de dispositivos</i> (BOLZANI, 2004).	12
Figura 6 – <i>Camadas da Arquitetura da internet das coisas</i> (WAKA, 2015).	13
Figura 7 – <i>Chip ESP8266EX e suas portas</i> (ADAPTADA PELO AUTOR).	16
Figura 8 – <i>ESP8266 modelo ESP-01</i> (CURVELLO, 2015).	16
Figura 9 – <i>Variantes do ESP8266</i> (CURVELLO, 2015).	17
Figura 10 – <i>Placa NodeMCU</i> (AUTOCORE, 2017).	18
Figura 11 – <i>Portas ESP-01</i> (ROVAI, 2016).	18
Figura 12 – <i>Portas do ESP-12</i> (ADAPTADA PELO AUTOR).	18
Figura 13 – <i>OrangePI Lite</i> (ORANGEPI, 2017).	20
Figura 14 – <i>Módulo conversor USB serial</i> (PRODUZIDA PELO AUTOR).	20
Figura 15 – <i>Estrutura de um relé eletromecânico</i> (FINDER, 2014).	21
Figura 16 – <i>Estrutura de um relé de estado sólido</i> (FINDER, 2014).	22
Figura 17 – <i>Módulo Sensor MQ-2</i> (ELABORADA PELO AUTOR).	23
Figura 18 – <i>Webcam com visão noturna.</i> (ELABORADA PELO AUTOR).	23
Figura 19 – <i>Espectro Eletromagnético Visível</i> (LOMBARDI, 2006).	24
Figura 20 – <i>Formas de transmissão e recepção de sinais IR</i> (LOMBARDI, 2006).	25
Figura 21 – <i>LED Emissor IR 5mm</i> (PRODUZIDA PELO AUTOR).	25
Figura 22 – <i>Receptor IR TS1838</i> (PRODUZIDA PELO AUTOR).	25
Figura 23 – <i>Modulação NEC de níveis lógicos</i> (SINGH, 2013).	26
Figura 24 – <i>Protocolo NEC</i> (SINGH, 2013).	26
Figura 25 – <i>Estrutura de uma válvula solenoide</i> (SILVA, 2011).	27
Figura 26 – <i>Válvula solenoide 12v 1/2"</i> (JDR, 2017).	27
Figura 27 – <i>IDE Arduino</i> (ARDUINO, 2017).	32
Figura 28 – <i>Planta Baixa</i> (ACENGENHARIA, 2017).	33
Figura 29 – <i>Planta Elétrica</i> (ACENGENHARIA, 2017).	34
Figura 30 – <i>Diagrama macro da rede</i> (ELABORADA PELO AUTOR).	34
Figura 31 – <i>Diagrama de DI's de cada cômodo na rede</i> (ELABORADA PELO AUTOR).	36
Figura 32 – <i>Instalação elétrica de alguns cômodos</i> (RÔMULO, 2013).	37
Figura 33 – <i>Sistema de controle com IR</i> (ELABORADA PELO AUTOR).	37
Figura 34 – <i>Adicionamento do ESP no Arduino IDE</i>	39
Figura 35 – <i>Adicionamento das placas ESP no Arduino IDE</i>	40

Figura 36 – Seleção da placa nodeMCU para programação. . . . .	40
Figura 37 – Formatação do Cartão SD. . . . .	41
Figura 38 – Gravação do SO no cartão SD. . . . .	42
Figura 39 – Configuração de rede WiFi . . . . .	42
Figura 40 – Informações da Rede Conectada . . . . .	43
Figura 41 – Configuração para IP estático no roteador. . . . .	43
Figura 42 – Configuração MobbaXterm . . . . .	44
Figura 43 – Esquemático Host 1. . . . .	46
Figura 44 – Circuito para Acionamento do LED IR. . . . .	47
Figura 45 – Esquemático Host 2 . . . . .	47
Figura 46 – Esquemático Host 3 . . . . .	48
Figura 47 – Esquemático Host 4 . . . . .	48
Figura 48 – Circuito para Acionamento da válvula solenoide . . . . .	49
Figura 49 – Esquemático Host 5 . . . . .	49

# **Lista de abreviaturas e siglas**

IoT	Internet of Things, no português, "Internet das Coisas".
DI	Dispositivos Inteligentes.
WI-FI	Wireless Fidelity, no português, "Fidelidade sem Fio".
TCP/IP	Protocolo de Controle de Transmissão / Protocolo de Internet.
IR	Infra-Red, no português, "Infra-Vermelho".
RF	Rádio Frequência.
RGB	Red-Green-Blue, no português, "Vermelho-Verde-Azul".
I/O	In/Out, no português, "Entrada/Saída".
UDP	User Datagram Protocol, no português, "Protocolo de Datagrama do Usuário".
UART	Universal Asynchronous Receiver Transmitter, no português, "Transmissão e Recepção universal assíncrono".
LED	Light Emitting Diode, no português, "Diodo Emissor de Luz".
USB	Universal Serial Bus, no português, "Barramento Serial Universal".
SSR	Solid State Relay, no português, "Relé de Estado Sólido".
IDE	Integrated Development Environment, no português, "Ambiente de Desenvolvimento Integrado".
LAN	Local Area Network, no português, "Rede de Área Local".
WAN	Wide Area Network, no português, "Rede de Área Alargada".
LTE	Long Term Evolution, no português, "Evolução de Longo Prazo".
SO	Sistema Operacional.
HTML	HyperText Markup Language, no português, "Linguagem de Marcação de Hipertexto".
CSS	Cascading Style Sheets, no português, "Estilo de Folha em cascata".
JS	JavaScript.

SSH	Secure Shell, no português, "capsula Segura".
DDNS	Dynamic Domain Name System, no português, "Nome Dinâmico do Domínio".
HTTP	Hypertext Transfer Protocol, no português, "Protocolo de Transferência de Hipertexto".

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Objetivos	1
1.2	Organização do Trabalho	2
1.3	Metodologia	3
<b>2</b>	<b>Automação Residencial</b>	<b>4</b>
2.1	Domótica	4
2.2	Evolução e Perspectivas Futuras	5
2.3	Atuação	5
2.3.1	Controle de Iluminação	6
2.3.1.1	Lâmpadas Liga/Desliga	6
2.3.1.2	Lâmpadas com Dimerização	6
2.3.2	Controle de Tomadas	6
2.3.3	Controle de Eletrodomésticos	6
2.3.4	Segurança Patrimonial	7
2.3.4.1	Controle de Acessos	7
2.3.4.2	Alarme	7
2.3.5	Controle de Fluidos	7
2.4	Normas Básicas	7
<b>3</b>	<b>Internet das Coisas</b>	<b>9</b>
3.1	Evolução e perspectivas futuras	10
3.2	Arquitetura da Internet das coisas	12
3.2.1	Camada de Borda	12
3.2.2	Camada de Gateway de acesso	12
3.2.3	Camada de Internet	13
3.2.4	Camada de Middleware	13
3.2.5	Camada de Aplicação	13
<b>4</b>	<b>Projeto</b>	<b>15</b>
4.1	Hardware	15
4.1.1	ESP8266	15
4.1.1.1	Família ESP	16
4.1.1.2	Descrição de portas	17
4.1.1.3	Consumo	18
4.1.2	Microcomputador	19

4.1.3	Interface USB-Serial . . . . .	19
4.1.4	Relés . . . . .	21
4.1.4.1	Relé Mecânico . . . . .	21
4.1.4.2	Relé de Estado Sólido . . . . .	22
4.1.5	Sensor de gases MQ-2 . . . . .	22
4.1.6	Câmera de vídeo . . . . .	23
4.1.7	Comunicação IR . . . . .	24
4.1.8	Válvula Solenoide . . . . .	26
4.1.9	Fonte de Alimentação . . . . .	27
4.1.10	Componentes Diversos . . . . .	28
4.2	Software . . . . .	29
4.2.1	Plataformas IoT . . . . .	29
4.2.2	NodeJS . . . . .	31
4.2.3	HTML, CSS e JS . . . . .	31
4.2.4	Arduino IDE . . . . .	31
4.2.5	MobaXterm . . . . .	32
4.2.6	Softwares diversos . . . . .	32
4.3	Visão do Projeto . . . . .	33
4.4	Rede e Conexões . . . . .	33
4.4.1	Conexões . . . . .	35
4.5	Esquematização . . . . .	36
4.5.1	Lâmpadas e Tomadas . . . . .	36
4.5.2	Eletrodomésticos Controlados por Infra-Vermelho . . . . .	37
4.5.3	Detecção de Incêndio . . . . .	37
4.5.4	Torneira Autônoma . . . . .	37
<b>5</b>	<b>Execução em Software . . . . .</b>	<b>39</b>
5.1	Arduino IDE . . . . .	39
5.2	Módulos ESP8266 . . . . .	41
5.3	Servidor . . . . .	41
5.4	Interface com o Usuário . . . . .	45
<b>6</b>	<b>Execução em Hardware . . . . .</b>	<b>46</b>
6.1	Módulos ESP . . . . .	46
6.1.1	Host 1 . . . . .	46
6.1.2	Host 2 . . . . .	46
6.1.3	Host 3 . . . . .	46
6.1.4	Host 4 . . . . .	46
6.1.5	Host 5 . . . . .	46

<b>7 Resultados</b>	50
<b>8 Conclusão e Sugestões para trabalhos futuros</b>	51
<b>Referências</b>	52
<b>ANEXO A Programação ESP8266 HOST 2</b>	55
<b>ANEXO B Página HTML</b>	59
<b>ANEXO C Página CSS</b>	60
<b>ANEXO D Servidor NodeJS</b>	61

# 1

## Introdução

A automação, de acordo com o dicionário, é a prática de criar autômatos<sup>1</sup>, ou seja, utilizar máquinas para alguma finalidade. Para o controle e supervisão de qualquer sistema automático, é necessário um meio de comunicação e com a evolução da internet das coisas, essa comunicação se torna um fator revolucionário para este ramo.

A rede de internet, atualmente, já atingiu grande parte da população mundial, de acordo com ([SANOU, 2016](#)) no referido ano de publicação, tinha-se que 95% da população global vivia em uma área coberta pela rede de internet. A figura 1 representa graficamente essa evolução em um range de aproximadamente 10 anos, ou seja, de 2007 até 2016. As tecnologias móveis de internet são representadas nesta figura como uma medida de cobertura, seja em 2g, 3g ou LTE; neste caso por exemplo, em 2016 a tecnologia 2g era capaz de cobrir aproximadamente 6,5 bilhões de pessoas, enquanto a 3g aproximadamente 5,2 bilhões e a LTE 3,5 bilhões.

O tracejado em vermelho representa o crescimento populacional, atualmente medindo em torno de 7 bilhões de pessoas; Já a linha verde representa a quantidade de usuários.

A perspectiva futura é que exista cobertura da rede de internet em todo o planeta e que qualquer "coisa" possa ser conectado a ela. Isto acarretaria em um mundo totalmente conectado, no qual a distância seria um parâmetro irrisório, para qualquer ação humana.

### 1.1 Objetivos

O Propósito geral deste trabalho é detalhar as vertentes da automação residencial e da *IoT*<sup>2</sup>, explorando funcionalidades, normas, estruturas e componentes que as compõem para então comprovar na prática o funcionamento de um sistema automático residencial utilizando a *IoT*, para comunicação, supervisão e controle. É esperado que ao final deste trabalho obtenha-se clareza em toda teoria, resultados satisfatórios para o sistema implementado e contribuição para projetos científicos, acadêmicos ou profissionais.

<sup>1</sup> Autômatos: Qualquer equipamento que opere de maneira autônoma.

<sup>2</sup> IoT: do inglês "*internet of things*" que significa internet das coisas.

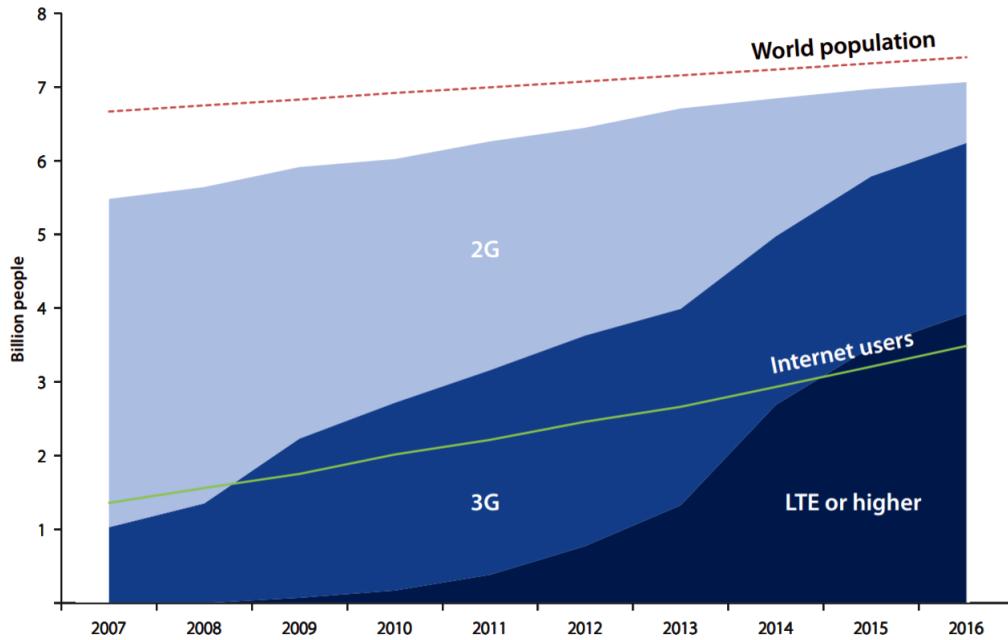


Figura 1 – Gráfico da cobertura de redes móveis e tecnologias em evolução (SANOU, 2016).

## 1.2 Organização do Trabalho

O Trabalho foi desenvolvido em oito capítulos, nos quais os três primeiros são de estado da arte e revisões bibliográfica, o quarto descreve o projeto, o quinto e sexto como foi feito e o sétimo e oitavo são as considerações finais.

O capítulo 1 faz uma breve introdução, explanando os objetivos, descrevendo como será o trabalho e os métodos utilizados para sua realização.

O capítulo 2 faz uma revisão bibliográfica sobre a automação residencial, descrevendo seu surgimento, o que proporciona aos usuários, empresas que atuam no ramo, normas que a regem e algumas perspectivas futuras.

O capítulo 3 aborda a internet das coisas, fazendo um estudo bibliográfico, desde as ideias futuristas até as suas aplicações. Também é explorado nesse capítulo a arquitetura utilizada e o que é esperado dessa tecnologia para o futuro.

O capítulo 4 refere-se ao projeto, nele é descrito todos os componentes que foram utilizados, fazendo descrições teóricas sobre cada um deles, tanto em hardware quanto em software; como será a conexão entre os periféricos e um apanhado geral de como funciona o sistema, seguindo todas as normas estabelecidas para esse tipo de projeto.

O capítulo 5 é composto pelo desenvolvimento do software. Nele é descrito de forma detalhada tudo que foi feito para programar a central, os terminais e a interface com o usuário.

O capítulo 6 é composto pelo desenvolvimento do hardware. Nele são feitas análises dos circuitos e como eles atuam no sistema.

O capítulo 7 traz os resultados do trabalho; fazendo breves comentários das dificuldades encontradas e explanando de formas gráficas e por tabelas algumas informações sobre o sistema.

O último capítulo, capítulo 8, trás as conclusões do trabalho e sugestões para trabalhos futuros.

### **1.3 Metodologia**

Para a realização deste trabalho, foram feitos estudos qualitativos utilizando livros, artigos científicos, teses de mestrado, pesquisas na internet e todo conhecimento adquirido durante a graduação. Para o entendimento dos componentes em hardware foram feitos estudos baseados em datasheets, fornecidos pelos fabricantes, e também através de livros práticos. Para melhor compreensão do assunto, foram utilizadas figuras ilustrativas, tabelas e citações.

# 2

## Automação Residencial

Após a invenção da máquina a vapor na Inglaterra, no século XVIII, a produção de produtos manufaturados obteve um crescimento nunca visto na história. Esse período ficou conhecido como a revolução industrial, e foi a partir de então que se estabeleceu o início da automação ([SILEVIRA, 2003](#)). Ao decorrer dos anos a automação foi tornando-se indispensável para industria.

Apenas em meados da década de 80, com o surgimento dos microcontroladores e computadores pessoais, que a tecnologia foi capaz de suprir projetos de pequeno porte, de forma eficiente. Finalmente foi possível aplicar a automação em residências. As primeiras companhias a aplicar esse novo conceito foram a *Leviton* e *X10 corporate*.

A automação residencial se tornou um ramo muito promissor, com o surgimento da internet e o barateamento de hardware. Esses fatores favoreceram para interligar todos os equipamentos de uma residência, fazendo com que trabalhem em conjunto. Isso é o que realmente define uma residência automatizada ([BOLZANI, 2004](#)).

A automação surgiu da necessidade de que máquinas façam trabalhos que sejam repetitivos para os seres humanos e que, além disso, tenha rapidez, precisão e exatidão. O ramo da automação aplicada em residências tem o intuito de tornar a vida das pessoas mais eficiente, cômoda, segura, econômica e também de auxiliar pessoas com deficiências.

### 2.1 Domótica

Um sistema residencial automático controlado através de uma central recebe o nome de "*Domótica*". Esse termo se origina da junção das palavras "*Domus*" (que significa "casa" em latim) e "*Robótica*" ([BOLZANI, 2004](#)).

Segundo ([BRUGNERA, 2007](#)), "A domótica é um recurso utilizado para controle de um ou mais aparelhos eletrônicos por meio de uma central computadorizada".

Atualmente, a Domótica vem mostrando sua grande capacidade de mercado, principalmente com os avanços da *IoT*. No qual a "central computadorizada" passa a ser à rede de internet, ou seja, a partir de um webserver será possível ter o controle do sistema através de algum dispositivo com acesso a internet (smartphones, tablets, laptops, computadores, smartTVs, ipods, entre outros) ([BRUGNERA, 2007](#)).

A figura 2 ilustra uma visão macro de um sistema domótico, na qual todos os subsistemas de atuação são ligados a uma central computadorizada, representada nesta figura, como o integrador.

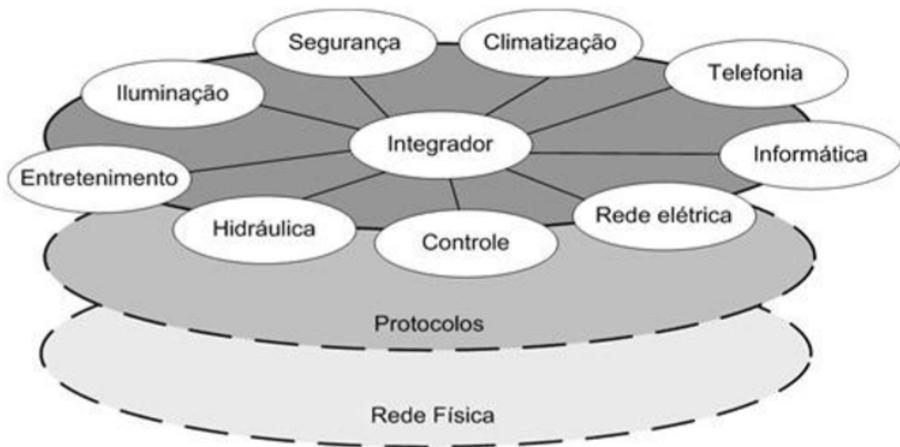


Figura 2 – Estrutura da Automação Residencial ([BOLZANI, 2004](#)).

## 2.2 Evolução e Perspectivas Futuras

Indubitavelmente, a Domótica, utilizando centrais computadorizadas, existe a muitos anos, mas ainda não é totalmente disseminada pelo mundo. A domótica utilizando *IoT* tem uma situação ainda pior. Estamos longe de alcançar um patamar no qual tudo será inteligente. Segundo ([DOUKAS, 2012](#)) a *IoT* ainda tem muitos desafios a serem vencidos para que se possa ter um nível plausível de tecnologia e de aplicações para suprir todo o mundo.

Todavia, cada vez mais o empreendedorismo e pesquisas científicas nesta área específica vem evoluindo. No Brasil, empresas como a *iot soluções* e a *home manager*, são exemplos de empreendimentos que trabalham especificamente com a Domótica utilizando-se de artifícios da *IoT*.

Com o avanço da *IoT* e de hardwares específicos para controle via web, a tendência da domótica, utilizando rede de internet, é liderar o mercado da automação não somente residencial e predial como qualquer tipo de automação ([TWITTY, 2016](#)).

## 2.3 Atuação

A Domótica, utilizando-se da *IoT*, vem sendo amplamente pesquisada e a cada dia ganhando novas possibilidades. As aplicações que empresas e pesquisas científicas mais sugerem são as seguintes: controle de iluminação, controle de tomadas, controle de eletro-domésticos e segurança patrimonial. Cada uma delas são detalhadas abaixo.

### 2.3.1 Controle de Iluminação

A iluminação, tanto natural como artificial é algo que sem exceção é encontrada em todas as residencias. O controle sobre elas atualmente é feito por interruptores físicos ou janelas, persianas e etc. Tornando-se então o seguimento na qual a domótica é mais aplicada.

#### 2.3.1.1 Lâmpadas Liga/Desliga

Além de controle manual do usuário com interruptores, o controle ON/OFF de iluminação pode ser feito através de sensoriamento luminoso, por relés ou detectores de presença, isso acarreta em um sistema inteligente e autônomo, ou seja, alguns cômodos só precisam de iluminação com a presença de algum indivíduo, neste caso o detector de presença será responsável pelo acionamento da lâmpada. Utilizando a internet, o usuário pode ligar ou desligar lâmpadas no caso dos relés, ou supervisionar para os casos de sensoriamento e detecção de presença ([WAKA, 2015](#)).

#### 2.3.1.2 Lâmpadas com Dimerização

Em caso de lâmpadas dimerizáveis, o ambiente é adaptado a preferência do usuário, podendo o controle ser feito em malha aberta, na qual o usuário define a intensidade da luz, ou um controle em malha fechada no qual utiliza-se sensores de luminosidade para indicar a intensidade de luz no local. Neste caso, o usuário seleciona o nível de luminosidade desejada e o sistema automaticamente manipula tudo que envolve a iluminação do local, seja mudando a intensidade da luz, fechando persianas ou "vidros inteligentes"<sup>1</sup>. Uma outra aplicação da dimerização é para o controle de cor, em lâmpadas RGB. As mesmas são compostas basicamente de três emissores de luz juntos, um vermelho, um verde e um azul. A dimerização de cada uma delas em diferentes níveis, podem gerar qualquer cor no ambiente.

A mudança de luminosidade no ambiente pode ser vista na figura 3.

### 2.3.2 Controle de Tomadas

O controle feito na tomada se faz acionando ou não uma chave eletrônica. Esse artifício é capaz de ligar, desligar ou dimerizar a mesma. Pode-se controlar qualquer equipamento ligado a ela, exemplos: ventilador, forno elétrico, motor para persianas, micro system, refrigeradores, entre tantos outros.

### 2.3.3 Controle de Eletrodomésticos

Repetidores de controle remoto, seja IR ou RF, faz o controle de sistemas de multimídia, como: Televisores, ar-condicionados, ou qualquer equipamento que utiliza controle remoto.

<sup>1</sup> Vidros inteligentes: Nova tecnologia capaz de mudar a cor do vidro, filtrando raios UV



Figura 3 – Adaptação de Iluminação em um Ambiente ([QUINDERÉ, 2009](#)).

### 2.3.4 Segurança Patrimonial

O setor que mais impulsiona o mercado da automação residencial é o de segurança; visto que é um setor no qual não é supérfluo e é uma necessidade cada vez mais crescente em todo o mundo.

#### 2.3.4.1 Controle de Acessos

Fechaduras inteligentes acionadas por reconhecimento facial, impressão digital ou scanner de retina.

#### 2.3.4.2 Alarme

Disparo sonoro ou alerta virtual para alguns eventos, como: incêndios, invasão, entre outros. Utilização do Chaveiro do Pânico, que é um acionamento no qual o usuário ou o sistema pode emitir para que sejam acessas todas as luzes do local, soe um alarme ou efetue uma ligação de emergência automática ([BOLZANI, 2004](#)).

### 2.3.5 Controle de Fluidos

Controle de vazão de fluidos, como: gás de cozinha e principalmente água, em chuveiros, irrigadores ou controle de água quente.

## 2.4 Normas Básicas

Para todo e qualquer projeto, é fundamental seguir normas técnicas, na qual garantem a padronização, qualidade, intercambiabilidade, respeito ao meio ambiente e principalmente a

segurança, tanto dos usuários como dos prestadores de serviço.

Algumas normas que abrangem o escopo da automação residencial são listadas a seguir:

- ABNT (Associação Brasileira de Normas Técnicas)
  - ([NBR16264, 2014](#)) *Cabeamento Estruturado Residencial*.
  - ([NBR5410, 2004](#)) *Instalações Elétricas em Baixa Tensão*.
  - ([NBR7198, 1993](#)) *Projeto e Execução de instalações prediais de água quente*.
  - ([NBR5413, 1992](#)) *Iluminação de Interiores*.
- ANSI/TIA/EIA (American National Standards Institute)/(Telecommunication Industry Association)/(Energy Information Administration)
  - ([ANSI/TIA/EIA-570-A, 2003](#)) Residential Telecommunication Cabling Standard (Sistemas de Cabeamento Residencial).
  - ([ANSI/EIA-600, 1995](#)) Consumer Electronic Bus (Padrão Utilizado nos Equipamentos de Automação).

# 3

## Internet das Coisas

A *IoT* é uma nova tecnologia em expansão na qual tem o objetivo de conectar DI's<sup>1</sup> do nosso dia-a-dia na rede de internet. Qualquer dispositivo, em rede, pode trocar informações e acionar eventos de forma inteligente. Uma ilustração da *IoT* pode ser vista na figura 4.



Figura 4 – Ilustração de objetos conectados pela IoT (APLEX, 2016).

De acordo com (DOUKAS, 2012), *IoT* é definida como uma revolução tecnológica, na qual a atual interação homem-máquina passa a ser uma interação máquina-máquina.

Com a *IoT*, é possível que sensores e atuadores presentes em objetos possam interagir entre si, através da internet. O potencial da internet das coisas é imensurável, pois as possibilidades de aplicações são de acordo com a criatividade. Se cada objeto pode se comunicar entre si, quaisquer união de equipamentos torna-se um Cluster<sup>2</sup>.

<sup>1</sup> DI's: Dispositivos inteligentes. segundo (BOLZANI, 2004). Qualquer eletroeletrônico que desenvolva: uma tarefa básica, efetue troca de informação e possibilite comando remoto. É considerado um dispositivo inteligente.

<sup>2</sup> Cluster: termo em inglês mais usado no meio técnico, representando a ideia de "conjunto de equipamentos"

Com o avanço tecnológico da instrumentação eletrônica, sensores e atuadores cada vez mais vem tornando-se robustos. Este fato implica que qualquer "*coisa*" pode ser adaptada para enviar e receber informações, através de uma rede. Portanto, é possível tornar qualquer "*coisa*" em um DI. As "*coisas*" podem ser um chip transmissor, um localizador, um marca passo, uma câmera, um sensor no motor do carro, uma fechadura, uma chave seletora, um eletrodoméstico, um cômodo, um animal de estimação ou qualquer "*coisa*" que possa ser capaz de enviar e receber informações através de uma rede sem fio ou cabeadas.

### 3.1 Evolução e perspectivas futuras

A ideia de *IoT* já existia a décadas como pensamentos futuristas, a exemplo disto, foram os pensamentos de dois grandes cientistas, MICHIO KAKU e NIKOLA TESLA. O físico norte-americano Michio Kaku ouviu trezentos dos principais cientistas do planeta para poder desenvolver sua pesquisa, e então fazer previsões precisas para o futuro da internet ([CARVALHO, 2012](#)).

Em uma palestra na Campus Party em 2012, MICHIO KAKU fez a seguinte citação:

"Os computadores que conhecemos hoje deixarão de existir, e a internet estará em tudo. Incluindo os seus óculos, que serão capazes de reconhecer os rostos das pessoas e ver suas biografias. Elas vão falar chinês e você vai ler as legendas do idioma bem diante dos seus olhos"

KAKU nesta citação, nos leva a perceber que o futuro da computação, cada vez mais, será moldado a internet. Ele, através deste exemplo não apenas mostrou que teremos internet em tudo, mas sim, o impacto que ela terá na vida das pessoas, com informações surgindo de todos os lugares.

Uma outra citação futurista é do brilhante engenheiro Nikola Tesla para a revista *COLLIER'S MAGAZINE* em 1926 ([NOVAK, 2015](#)):

"Quando a comunicação sem fio for aplicada perfeitamente toda a terra será convertida em um enorme cérebro, que na verdade é, todas as coisas sendo partículas de um todo real e rítmico... e os instrumentos através dos quais seremos capazes de fazer isso serão incrivelmente simples em comparação com o nosso telefone atual. Um homem será capaz de transportar um no bolso do colete"(NIKOLA TESLA, 1926).

Tesla, no século passado, já tinha ideias de como seria um mundo conectado, ele não só teve a visão de como seria um mundo com um mesmo padrão de comunicação "internet" como também, que tudo a nossa volta estaria conectado como "Partículas de um todo".

---

*interconectados*" A tradução direta seria "*agrupamento*".

Apenas depois de 1974 com a invenção da internet e com as padronizações de protocolos de comunicação, que o nível de tecnologia foi se desenvolvendo para se tornar capaz de colocar em prática DI's na internet. Mas não só a internet foi necessária para tornar realidade a *IoT*, os desenvolvimentos tecnológicos nas áreas de: microeletrônica, robótica, sensoriamento e telecomunicação vem tornando-se não menos importante para suas aplicações.

A origem do nome IoT é atribuída a Kevin Ashton. O nome internet das coisas foi o tema de uma apresentação feita por ele para a empresa Procter&Gamble (P&G) em 1999. Em 2009 Ashton em um artigo publicado pela "RFID Journal" cita sua apresentação e define o que é a IOT ([BALLAGUER, 2014](#)).

“... Se tivéssemos computadores que soubessem de tudo o que há para saber sobre coisas, usando dados que foram colhidos, sem qualquer interação humana, seríamos capazes de monitorar e mensurar tudo, reduzindo o desperdício, as perdas e o custo. Gostaríamos de saber quando as coisas precisarão de substituição, reparação ou atualização, e se eles estão na vanguarda ou se tornaram obsoletas (ASHTON ,2009)”.

De acordo com ([ANDERSON, 2014](#)), 83% dos especialistas em tecnologia dos Estados Unidos acreditam que o conceito *IoT* se tornará uma tendência geral no nosso cotidiano até 2025.

Atualmente o padrão da comunicação sem fio é o *IEEE 802.11*, mais conhecido como WI-FI<sup>3</sup> nele é utilizado os protocolos TCP/IP<sup>4</sup> ([IEEE, 2017](#)). O IP é conhecido como a "*identidade*" de um dispositivo, ou seja, é o nome atribuído a ele. Neste caso, uma mensagem direcionada a um dispositivo específico necessita saber para qual enviar em meio a tantos, portanto, na mensagem enviada é informado para qual IP ela está sendo direcionada.

A *IoT* ainda é vista como uma internet do futuro, mas os desafios para um mundo conectado já está sendo vencido, a troca do protocolo ipv4 para o protocolo ipv6, na camada de rede do TCP/IP, já é um forte indício do crescimento da IoT. Isso por que, a transferência de endereços de protocolo do ipv4 é de apenas 32 bits, ou seja, é capaz de ter aproximadamente 4,29 bilhões de ip's; Neste caso 4,29 bilhões de dispositivos. Enquanto eram utilizado apenas computadores pessoais na rede, esse numero era suficiente, mas o crescimento de dispositivos conectados se eleva exponencialmente, como pode ser visto através da figura 5.

Com o número de Ip's do ipv4, seria impossível conectar a quantidade atual de dispositivos na rede. Então o Ipv6 surgiu para suprir esta necessidade, utilizando-se de 128 bits de transferências de endereços de protocolo, ou seja, aproximadamente  $3,4 \cdot 10^{38}$  ip's. Este número é portanto o máximo de equipamentos que poderá existir na rede.

<sup>3</sup> WI-FI: uma abreviação de “Wireless Fidelity”, que significa fidelidade sem fio, em português.

<sup>4</sup> TCP/IP: TCP (Transmission Control Protocol - Protocolo de Controle de Transmissão) e o IP (Internet Protocol - Protocolo de Internet, ou ainda, protocolo de interconexão)

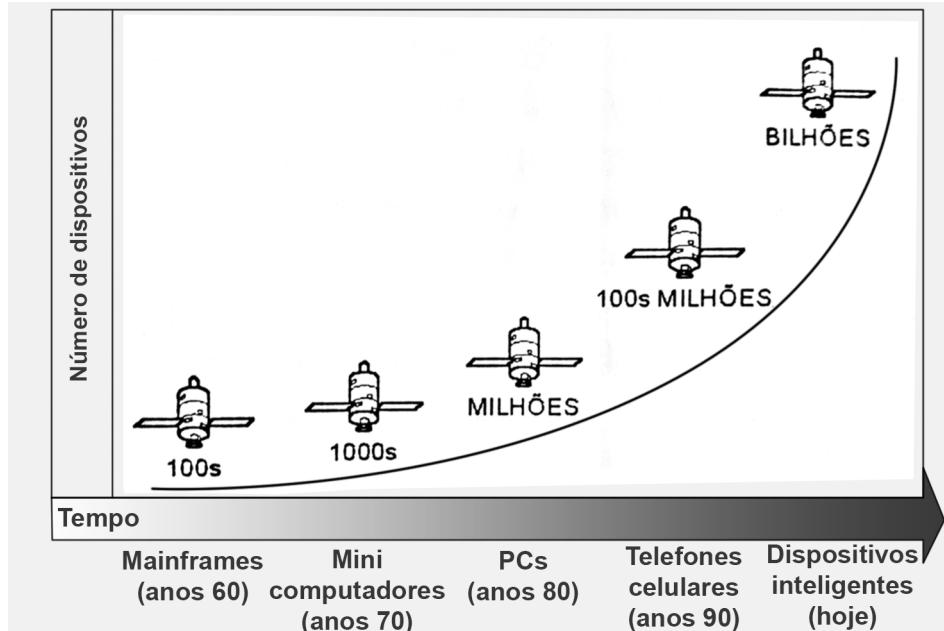


Figura 5 – Evolução do número de dispositivos (BOLZANI, 2004).

## 3.2 Arquitetura da Internet das coisas

Como a IoT tem o objetivo de conectar tudo em uma única rede, se torna necessário uma padronização no modo de conexão na rede virtual, pois cada objeto tem seu padrão de conexão. Não só para que cada dispositivo tenha a capacidade de interagir com a rede mas também para uma maior disseminação da internet das coisas, se torna de fundamental importância ter um padrão fixo. Afinal, como seria se cada sistema tivesse seus próprios padrões? A disseminação deles seriam completamente inviáveis.

Segundo (ZARGHAMI, 2013) a arquitetura da IoT, de forma genérica, pode ser dividida em 5 camadas, como mostra a Figura 6 : camada de borda, camada gateway de acesso, camada de internet, camada de middleware e camada de aplicação.

### 3.2.1 Camada de Borda

Esta camada é vista como a camada de nível mais baixo. Ela está em contato direto com a captação de informação que determinado objeto está designado a processar, ou seja, é nessa camada que se encontra o *Hardware*. Além disso, esta camada também é capaz de processar informação e dar suporte à comunicação. Sistemas embarcados, e sensores no geral são alguns dos hardwares que a compõem (ZARGHAMI, 2013).

### 3.2.2 Camada de Gateway de acesso

Esta camada pode ser traduzida como ”portão de acesso” e é responsável por interligar máquinas que se utilizam de protocolos diferentes. Ou seja, ela é capaz de traduzir uma mensagem



Figura 6 – *Camadas da Arquitetura da internet das coisas* ([WAKA, 2015](#)).

para que sua camada superior ”camada Internet” possa entender. Esta camada é portando uma prestadora de serviço, pois é a responsável por criar a conexão da rede local com a rede mundial ([CONCEIÇÃO, 2016](#)).

### 3.2.3 Camada de Internet

Esta camada prove o acesso a rede mundial. É nesta camada que se encontra os serviços de *Cloud computing*<sup>5</sup> e toda a inteligência de processamento de grandes volumes de dados ([WAKA, 2015](#)).

### 3.2.4 Camada de Middleware

Esta camada têm como funcionalidades principais o agregamento e filtro dos dados recebidos dos dispositivos de *hardware* e descobrir informações para controle de acesso aos dispositivos para aplicações. Esta fica entre a camada de internet e a camada de aplicação ([WAKA, 2015](#)). Segundo ([FERREIRA, 2008](#)) sua função é controlar e monitorar o estado dos dispositivos com a finalidade de se obter a melhor interoperabilidade, escalabilidade e mobilidade possível das partes envolvidas.

### 3.2.5 Camada de Aplicação

Esta camada é a mais próxima do usuário e sua função é processar os dados coletados, no hardware, e em seguida prestar serviços nos objetos aos quais foi predisposto a realizar. É

<sup>5</sup> Cloud computing: no português, Computação na nuvem. É uma tecnologia que se utiliza da memória e da capacidade de armazenamento e cálculo de computadores e servidores compartilhados e interligados por meio da Internet.

nesta camada que observa-se a Internet das Coisas tomando formas reais ([CONCEIÇÃO, 2016](#)).

# 4

## Projeto

Neste capítulo é discutido como é um projeto para a automação de uma residência utilizando a IoT. Descrições detalhadas de todos componentes em hardware e todas as técnicas utilizadas em software é exploradas para que, ao fim deste trabalho, o leitor possa ter clareza no funcionamento de todas as etapas do projeto.

### 4.1 Hardware

Existem no mercado inúmeros modelos de hardware para automação e *IoT*, sejam elas, módulos, sensores, atuadores, ou componentes discretos. Aqui iremos fazer uma descrição apenas dos que são utilizados no projeto.

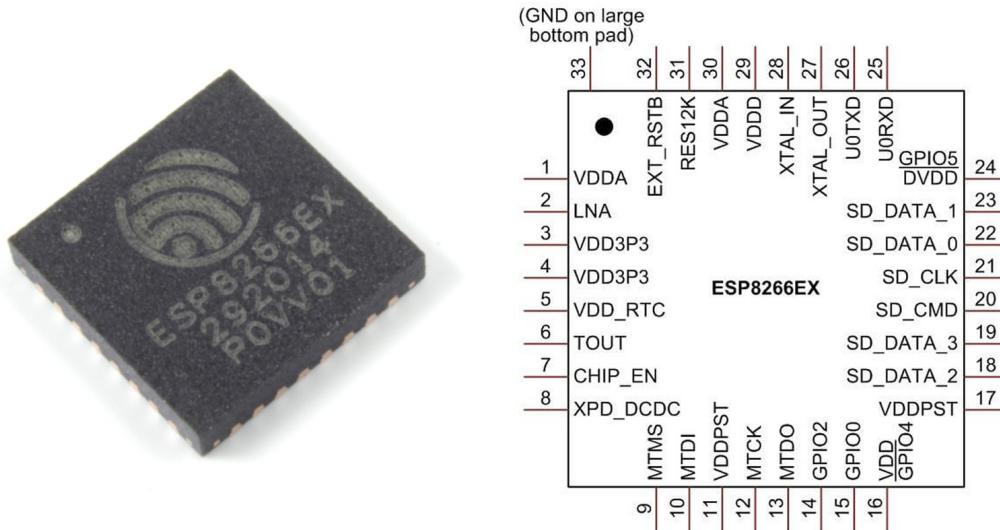
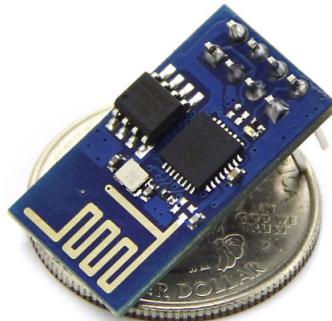
#### 4.1.1 ESP8266

Em busca de um dispositivo pequeno, de baixo custo e de ótimo desempenho. A empresa *Espressif* desenvolveu um chip chamado ESP8266EX (figura 7). Hoje esse pequeno dispositivo domina o mercado no que se refere a módulos de comunicação Wi-Fi. O lançamento foi feito em meados de agosto de 2004. Atualmente, este chip vem integrado em módulos, com alguns periféricos. Os fatores que chamam a atenção nestes são:

1. O tamanho, geralmente pouco maior que uma moeda, como na figura 8.
2. Seu preço, em torno de U\$ 3,00.
3. A facilidade de integralização em sistemas, por conta de ter um microcontrolador interno com comunicação serial UART.

A seguir, algumas das principais características desse poderoso hardware ([ESPRESSIF, 2016](#)):

- É um System-On-Chip com Wi-Fi embutido
- Tem conectores GPIO, barramentos I2C, SPI, UART, entrada ADC, saída PWM e sensor interno de temperatura.

Figura 7 – *Chip ESP8266EX e suas portas (ADAPTADA PELO AUTOR).*Figura 8 – *ESP8266 modelo ESP-01 (CURVELLO, 2015).*

- CPU que opera em 80MHz, com possibilidade de operar em 160MHz.
- Arquitetura RISC de 32 bits.
- 45KBytes de SRAM
- 64KBytes de ROM para boot.
- Possui uma memória Flash SPI Winbond W25Q40BVNIG de 512KBytes.
- O núcleo é baseado no IP Diamond Standard LX3 da Tensilica.
- Existem módulos de diferentes tamanhos e fabricantes.

#### 4.1.1.1 Família ESP

Atualmente os módulos ESP8266 são fornecidos numa grande variedade de modelos, do ESP-01 ao ESP-12, a figura 9 mostra todos eles. Cada modelo tem suas diferenças, principalmente no número de portas I/O's<sup>1</sup> e no tamanho.

<sup>1</sup> I/O's: In/Out, no português, Entrada/Saída. Significa portas de entidade e saída de dados



Figura 9 – Variantes do ESP8266 ([CURVELLO, 2015](#)).

De acordo com ([CURVELLO, 2015](#)) o maior objetivo dos modelos ESP-01 e o ESP-10 é servir como "*ponte Serial-WiFi*", ou seja, o módulo recebe comandos serial (UART) e interage com a rede Wi-Fi por meio de conexão TCP ou UDP<sup>2</sup>. Os outros modelos, podem também trabalhar em modo ponte Serial-WiFi, mas são capazes de desempenhar outras funcionalidades, uma delas é o modo de operação *standalone*<sup>3</sup>, ou seja, funciona como um microcontrolador com WiFi.

Existe no Mercado um variante do ESP8266, chamado de NodeMCU, visto na figura [10](#). Nela se encontra um módulo ESP-12 e alguns periféricos adicionais, como interface USB-Serial e regulador de tensão. Isto promove ao ESP uma maior facilidade de atualização de firmware através do computador, e maior robustez no que se diz respeito a tensão de alimentação.

#### 4.1.1.2 Descrição de portas

As portas são o meio físico no qual o hardware irá se comunicar com o mundo externo. A figura [11](#) e a figura [12](#) ilustram as portas presentes nos módulos ESP-01 e ESP-12. As funções dos principais pinos são as seguintes:

- **VCC:** Tensão de alimentação 3,3V. O módulo consome até 300 mA.
- **GND:** Sinal de Terra GND.

<sup>2</sup> UDP: User Datagram Protocol, protocolo que trabalha no envio de pacotes, semelhante ao TCP. A diferença é que o UDP remove toda a parte de verificação de erros que a outra tecnologia oferece.

<sup>3</sup> STANDALONE: No português, "ficam em pé por si só", é um modo no qual o programa é autossuficiente, e não precisa de programas auxiliares para seu funcionamento

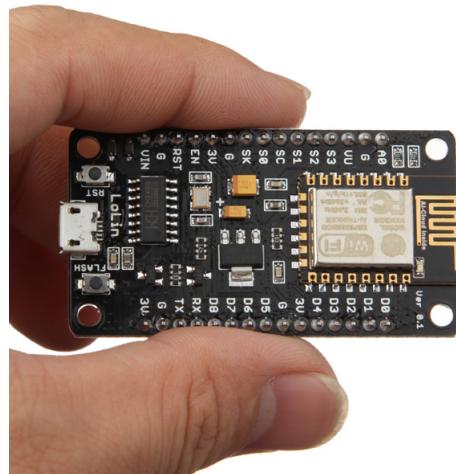


Figura 10 – Placa NodeMCU (AUTOCORE, 2017).

- **TX:** Sinal de Tx do módulo (Sinal em 3,3V).
- **RX:** Sinal de Rx do módulo (Sinal em 3,3V).
- **RST:** Sinal de Reset/Restart acionado em nível baixo (GND).
- **CH\_PD:** Sinal de habilitação do chip (chip enable), usado para gravação de firmware ou atualização. Deve ser mantido em nível ALTO para operação normal.
- **GPIO's:** I/O's que podem ser controladas pelo firmware, ou funções secundárias.

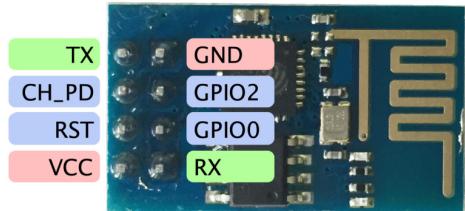


Figura 11 – Portas ESP-01 (ROVAI, 2016).

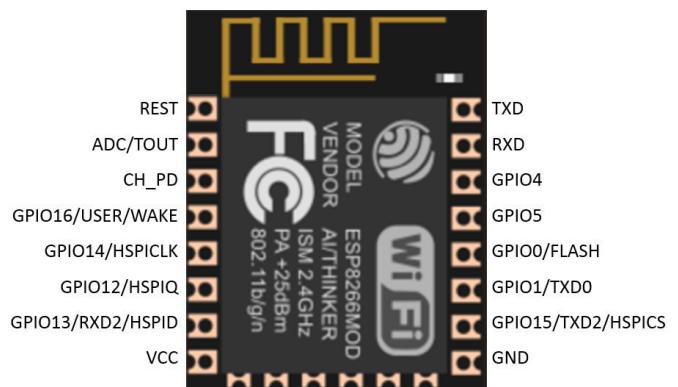


Figura 12 – Portas do ESP-12 (ADAPTADA PELO AUTOR).

#### 4.1.1.3 Consumo

Uma média do consumo de energia do ESP para cada modo de operação é, segundo (CURVELLO, 2015), mostrado na tabela 1.

Modo	Média (mA)
Transmit 802.11b, CCK 1Mbps, POUT=+19.5dBm	215
Transmit 802.11b, CCK 11Mbps, POUT=+18.5dBm	197
Transmit 802.11g, OFDM 54Mbps, POUT =+16dBm	145
Transmit 802.11n, MCS7, POUT=+14dBm	135
Receive 802.11b, packet length=1024 byte, -80dBm	60
Receive 802.11g, packet length=1024 byte, -70dBm	60
Receive 802.11n, packet length=1024 byte, -65dBm	62
Standby	0.9
Deep sleep	0.01
Power save mode DTIM 1	1.2
Power save mode DTIM 3	0.86
Total shutdown	0.0005

Tabela 1 – Tabela de Consumo de energia do ESP.

De acordo com a tabela1, o ESP pode consumir em média até 215mA. Para um dispositivo destas dimensões esse valor é relativamente alto. Portanto, é preciso dimensionar com prudência a fonte de alimentação, para que o sistema possa funcionar de forma eficiente.

#### 4.1.2 Microcomputador

Atualmente, graças a arquitetura ARM, a miniaturização de sistemas complexo é uma realidade. Existem no mercado inúmeros modelos de computadores em miniatura projetados exclusivamente para sistemas embarcados, nas mais variedades de configurações, alguns fabricantes deles são: RaspberryPi, OrangePi e BananaPi. No projeto é usado um microcomputador para atuar como servidor web ("O ESP8266 conseguiria fazer este papel para um sistema pequeno, porém, para ter uma página web mais elaborada e ter maior fluxo de dados, agradável aos usuários e com maior segura, a opção de utilizar um microcomputador se torna a melhor opção. Visto que os módulos ESP8266 é muito limitado em respeito a memória, processamento e etc.").

O modelo utilizado no projeto é da fabricante OrangePi, modelo Lite. A escolha desse modelo se justifica apenas pelo custo benefício, mas o modelo mais inferior que seria o "OrangePi zero" já serviria para este tipo de aplicação. A figura 13 informa algumas das configurações de hardware desse dispositivo. Existe inúmeros sistemas operacionais que esse modelo suporta, mas basicamente são SO baseados em linux ou android. Neste projeto é utilizado o SO armbian, que é baseado em linux. Esta escolha se deve ao fato da facilidade de atualizações, acesso remoto e compatibilidade com outros sistemas que é usado no projeto.

#### 4.1.3 Interface USB-Serial

O módulo ESP utiliza o protocolo serial UART para se comunicar, seja para programação ou troca de informações (ESPRESSIF, 2016). Atualmente os computadores não utilizam mais

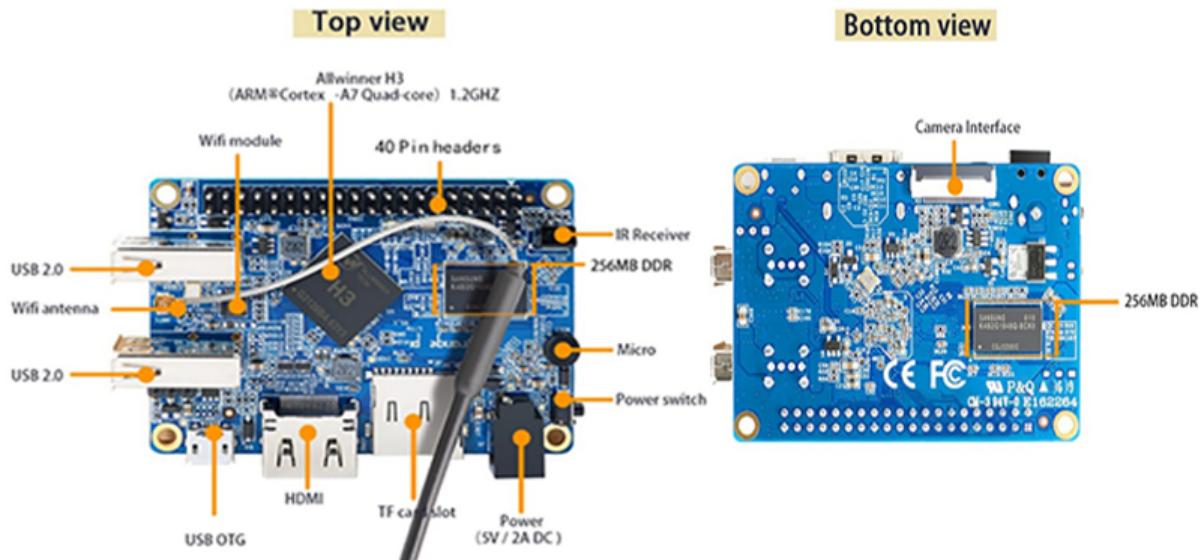


Figura 13 – *OrangePI Lite* (ORANGEPI, 2017).

dessa porta serial, mas sim da USB, logo, para se estabelecer uma comunicação entre o ESP com o computador se torna necessário a utilização de um conversor USB-Serial.

Atualmente existe um chip, fabricado pela Empresa FTDI, o FT232, que é capaz de fazer a conversão USB-UART. Este circuito integrado, é bastante utilizado, por sua robustez e velocidade de conversão. Uma característica bastante impactante para a utilização deste circuito neste projeto é o fato dele ter a opção de trabalho em 5V ou 3,3V, visto que o módulo ESP trabalha apenas em 3,3V.

Existem no mercado vários módulos nos quais utilizam o chip FT232, no projeto é utilizado o módulo da fabricante *Deek-Robot* figura 14. Para os Módulos NodeMCU, não será necessário o uso da interface, pois este módulo já tem um conversor como periférico.

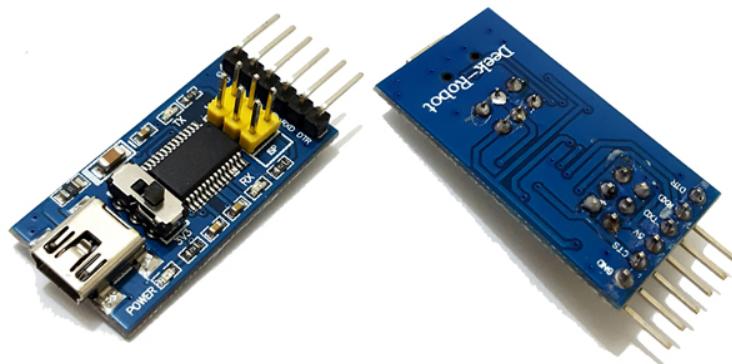


Figura 14 – *Módulo conversor USB serial* (PRODUZIDA PELO AUTOR).

## 4.1.4 Relés

Os relés são dispositivos eletrônicos comutadores, e sua principal função é de chaveamento. Ao longo da história este dispositivo esteve presente nas principais evoluções tecnológicas, seja em grandes centrais de telefonias, ou até mesmo nos primeiros computadores. Para alguns autores, os relés, juntamente com as válvulas, foram os componentes que inspiraram a criação dos transistores. Atualmente, o princípio destes ainda é bastante utilizado, seja em comandos elétricos ou para acionamento de grandes cargas.

### 4.1.4.1 Relé Mecânico

Os relés eletromecânicos são os mais primitivos, mas não menos usados. Seu funcionamento é baseado em um eletroímã<sup>4</sup> chaveando um circuito mecanicamente. A figura 15 ilustra a estrutura de um relé eletromecânico.

Ao passar corrente pela bobina, é criado um campo magnético que atrai o contato móvel, fazendo assim o chaveamento entre dois terminais, caso seja interrompida a corrente elétrica na bobina, o campo magnético deixa de existir e a mola de rearme retorna o contato móvel para seu outro terminal.

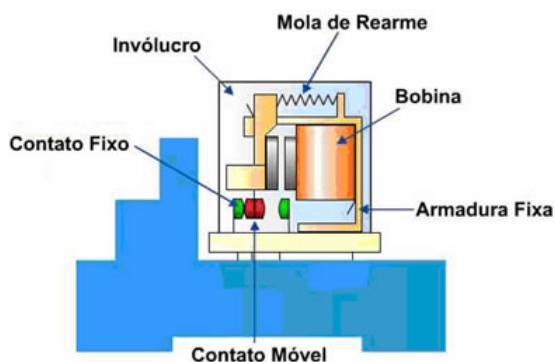


Figura 15 – Estrutura de um relé eletromecânico (FINDER, 2014).

Vantagens	Desvantagens
Isolação elétrica da carga	Desgaste mecânico dos contatos
Não necessita de dissipadores de calor	Velocidade de operação lenta
Baixo custo	Produção de arcos elétricos em cargas muito indutivas
Possibilidade de comutação AC ou DC	Potência de acionamento relativamente alta
Acionamento de várias chaves	Operação ruidosa

Tabela 2 – Tabela de vantagens e desvantagens de um relé eletromecânico.

Para o atual projeto, o relé eletromecânico é utilizado apenas no acionamento de lâmpadas na qual seja necessário o uso de interruptores Three-way ou em tomadas que forneça grandes

<sup>4</sup> Eletroímã: é um dispositivo que utiliza corrente elétrica para gerar um campo magnético, semelhante a um imã natural.

potências. Visto que para um melhor racionamento de energia, a utilização de relés de estado sólido se torna mais viável.

#### 4.1.4.2 Relé de Estado Sólido

Com os avanços da eletrônica de potência, foi desenvolvido um dispositivo com a mesma função do relé eletromecânico, porém sem ter partes mecânicas em movimento, o relé de estado sólido SSR.

Este dispositivo utiliza-se das propriedades dos materiais semi-condutores para o chaveamento, sua estrutura é ilustrada na figura 16. Este é um dispositivo óptico-acoplado, ou seja, se utiliza de um LED e um receptor para fazer o chaveamento, isto implica em uma potência muito baixa para o acionamento. O receptor óptico aciona uma chave eletrônica, esta chave, pode ser, triac's ou Mosfet's, dependendo do modelo.

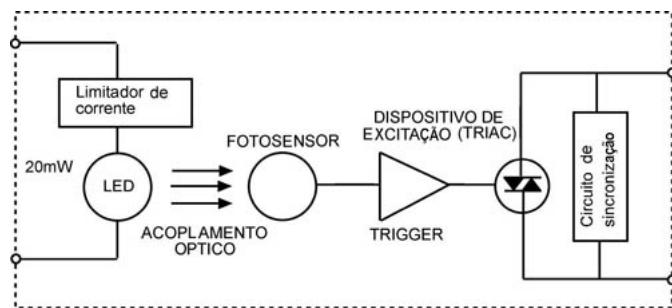


Figura 16 – Estrutura de um relé de estado sólido (FINDER, 2014).

Vantagens	Desvantagens
Potência de acionamento baixa	Resistência de condução alta
Vida útil longa	Necessidade de dissipador de calor para grandes cargas
Velocidade de operação rápida	Sensível a surtos
Possibilidade de comutação AC ou DC	Acionamento de apenas uma chave

Tabela 3 – Tabela de vantagens e desvantagens de um relé eletromecânico.

#### 4.1.5 Sensor de gases MQ-2

Um sensor analógico muito usado em sistemas embarcados e na industria, são os da família MQ. Para o projeto será utilizado o MQ-2, visto na figura 17. Esse sensor é capaz de detectar níveis de concentração de gases inflamáveis como o butano(gás de cozinha), propano e fumaça; Ele utiliza a variação da resistividade dióxido de estanho ( $\text{SnO}_2$ ) para fazer medições eletronicamente.

No projeto esse sensor é utilizado como um dispositivo de segurança, localizado na cozinha.



Figura 17 – *Módulo Sensor MQ-2* (ELABORADA PELO AUTOR).

#### 4.1.6 Câmera de vídeo

Câmeras são a primeira opção quando o quesito é segurança, portanto para um sistema de qualidade uma das principais aplicações é a o acesso remoto de imagens em tempo real. No projeto é utilizado uma simples webcam vista na imagem 18. A câmera é integrada junto ao servidor, que nesse caso é o microcomputador, nele é feito através de software toda a integração para que as imagens possa chegar ao usuário. Existem no mercado inúmeras câmeras de segurança, seja com visão noturna, alta resolução ou até mesmo com wifi integrado. A utilização desse modelo se justifica apenas pelo baixo custo, mas qualquer outra câmera é facilmente integrada ao projeto.



Figura 18 – *Webcam com visão noturna*. (ELABORADA PELO AUTOR).

As principais especificações deste modelo são as seguintes:

- Resolução de 20MP interpolado para fotos.
- Resolução de 5MP interpolado para vídeos.
- Taxa de transmissão de até 30 frames por segundo.

- Modo Noturno com Leds de iluminação.
- Porta USB 2.0 e compatível com USB 1.1.
- Compatível com sistema linux.

#### 4.1.7 Comunicação IR

O infravermelho é uma radiação não ionizante que se encontra um pouco abaixo da radiação visível vermelha no espectro eletromagnético. A palavra "*infra*", significa abaixo, ou seja, a baixo do vermelho. Esta radiação por ter frequência abaixo das radiações visíveis, tem um comprimento de onda maior que as mesmas e fica na faixa de 700 nanômetros até 1 milímetro, como ilustra a figura 19.

Atualmente o IR, através da fotônica<sup>5</sup>, é utilizado como meio de comunicação, detecção de calor, visão noturna, entre tantos outros.

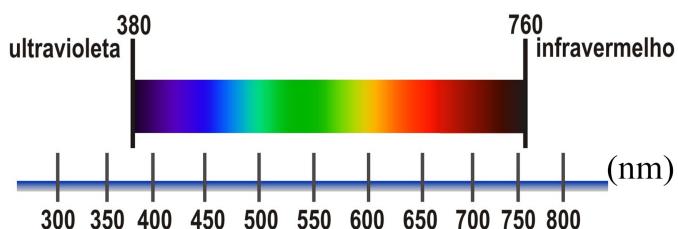


Figura 19 – *Espectro Eletromagnético Visível* (LOMBARDI, 2006).

A comunicação através do infravermelho vem sendo muito utilizada desde a criação do padrão de comunicação por IR da empresa *Infrared Data Association*.

Para a transmissão de dados, os sistemas IR utilizam das maiores frequências possíveis, ou seja, o mais próximo possível da radiação vermelha, e como qualquer luz, o sinal IR não penetra objetos opacos. Portanto existe um grande fator limitante nesta comunicação, que é a distância. Comunicações utilizando infravermelho só é possível a curtas distâncias, a figura 20 mostra os modos de transmissão e recepção destes sinais.

O modo de transmissão e recepção utilizado para controle remoto em eletrodomésticos é o difuso e não direcionado. Portanto é utilizado este modo no projeto.

Para a emissão do sinal, é utilizado um LED infra-vermelho figura 21, a recepção do sinal é feita pelo receptor IR já presente nos equipamentos em que se deseja controlar. Porém para decodificar os sinais dos controles presentes na residencia é utilizado o receptor IR visto na imagem 22, ou então utilizando-se de decodificações presente em um banco de dados separados por marcas encontrado no link: <<http://lirc.sourceforge.net/remotes>>

Existem inúmeras formas de modulação de um sinal, mas o padrão utilizado em eletrodoméstico é o protocolo NEC. Este protocolo utiliza a modulação por amplitude de um sinal

<sup>5</sup> Fotônica: ciência do processamento, geração, detecção e emissão da luz.

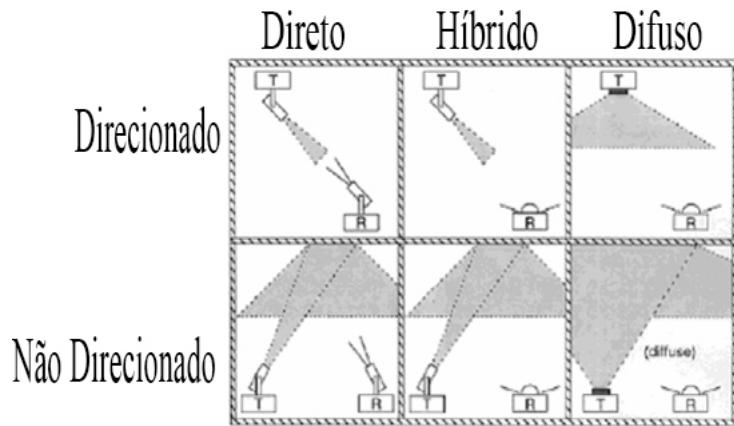


Figura 20 – Formas de transmissão e recepção de sinais IR (LOMBARDI, 2006).



Figura 21 – LED Emissor IR 5mm  
(PRODUZIDA PELO AU-  
TOR).

Figura 22 – Receptor IR TS1838  
(PRODUZIDA PELO  
AUTOR).

digital. O gráfico na figura 23 em tensão por tempo, representa como são os níveis lógicos '0' e '1'. Se o bit for '0' tem-se que  $526\mu s$  são pulsos e em seguida um espaço de  $526\mu s$  sem sinal. Já se o bit for '1' tem-se que  $526\mu s$  são pulsos e  $1.6875ms$  sem sinal.

Um sinal protocolado em NEC é iniciado com 9ms de pulsos e 4.5ms sem sinal. Em seguida são enviados 32 bits separados da seguinte forma:

- 8 bits de endereço do receptor.

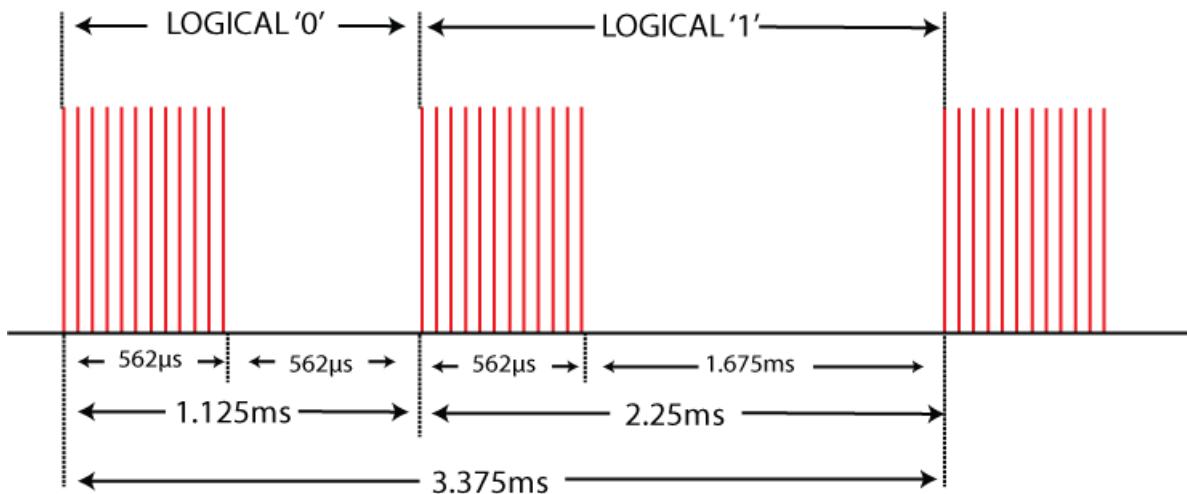


Figura 23 – Modulação NEC de níveis lógicos (SINGH, 2013).

- 8 bits de lógica inversa do endereço do receptor.
- 8 bits do comando.
- 8 bits de lógica inversa do comando.

A figura 24 representa o envio do comando  $10001101_2$  para o endereço  $10110001_2$ . Então pelo protocolo o código do comando será em binário  $10110001010011101000110101110110_2$  ou em hexadecimal  $\text{BD72B14E}_{16}$ .

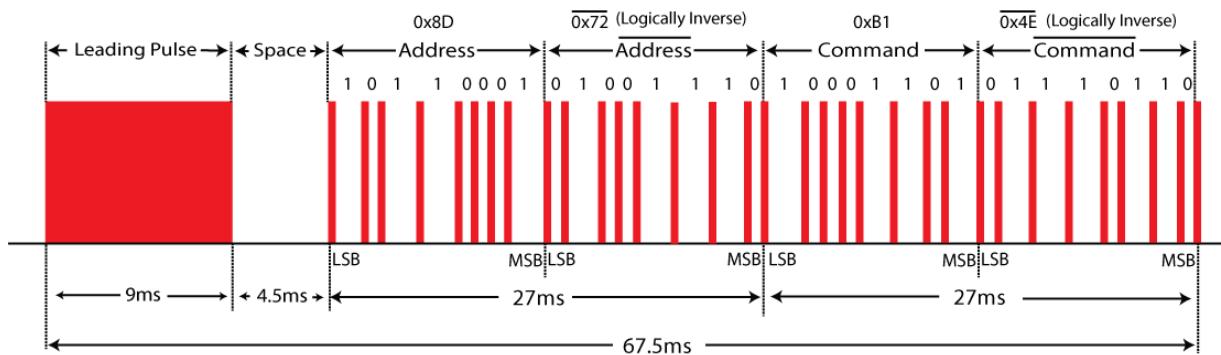


Figura 24 – Protocolo NEC (SINGH, 2013).

#### 4.1.8 Válvula Solenoide

As válvulas solenoide são dispositivos muito utilizados na automação, sua função básica é liberar ou bloquear a passagem de fluidos. Para isso a válvula utiliza um princípio semelhante aos relés eletromecânicos.

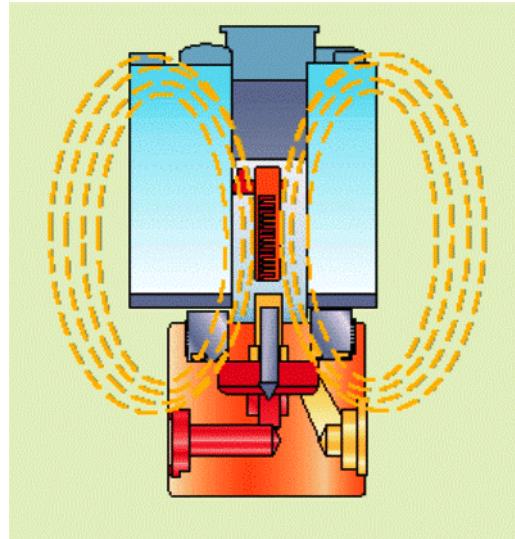


Figura 25 – Estrutura de uma válvula solenoide ([SILVA, 2011](#)).

A figura 25 ilustra o funcionamento de uma válvula solenoide. Ao passar corrente pelo solenoide ("bobina"), é criado um campo magnético, esse campo funciona como um eletroímã atraindo um pino piloto. Com isso, o pino abre e então o fluido tem caminho livre para passar, ao desenergizar a bobina, uma mola de rearme reposiciona o pino e o fluxo é bloqueado.

Para o projeto é utilizado uma válvula solenoide simples, de pequeno porte, com cano de meia polegada, acionado com 12 VDC (figura 26), apenas para irrigação. No entanto, as possibilidades do uso deste dispositivo em uma residencia, são inúmeras.



Figura 26 – Válvula solenoide 12v 1/2" ([JDR, 2017](#)).

#### 4.1.9 Fonte de Alimentação

Para qualquer sistema eletrônico a alimentação é inevitável, seja por baterias ou rede de energia. Como o projeto é estático na residência, a fonte utilizada é conectada a rede de energia, dispensando a utilização de baterias. Para isso, por motivos de custo e qualidade, as fontes

utilizadas para alimentação do sistema (Módulos ESP, microcomputador, atuadores, sensores e etc.) são fontes chaveadas.

Estas fontes atualmente está presente na maioria dos eletrônicos, seu funcionamento é baseado no chaveamento eletrônico da rede de energia, aplicação de filtros e controle de tensão de saída. Suas vantagens são o tamanho reduzido e pouca dissipação de energia. Carregadores de celular são exemplos desses tipos de fontes. Por conta da facilidade de encontrar estes carregadores, no projeto é utilizados os mesmos.

#### **4.1.10 Componentes Diversos**

Componentes como transistores, resistores, cabos, interruptores e capacitores são utilizados no projeto para diversos fins. As aplicações dos mesmos são descritas no decorrer dos capítulos seguintes, onde é explanado toda a execução do projeto.

## 4.2 Software

Atualmente, existem várias ferramentas, linguagens de programação e técnicas de softwares. Nesta secção é detalhado o que é usado, no que se diz respeito aos softwares, para o desenvolvimento do projeto.

### 4.2.1 Plataformas IoT

Por ainda ser uma tecnologia relativamente nova, a IoT ainda se encontra com um alto grau de heterogeneidade. Para contornar esse problema, estão surgindo plataformas voltadas diretamente para a IoT. Com isso, começa a surgir padrões, tanto de comunicação quanto armazenamento, tornando-se então desnecessária a criação de servidores de armazenamento e processamento de dados. As funções principais de uma plataforma são: armazenamento de dados, processamento e geração de retorno. Mas com a grande quantidade de plataformas no mercado, cada uma tem suas particularidades, algumas voltadas para o uso empresarial, outras para sensoriamentos ou até mesmo para uso residencial, ou seja, todas tem o mesmo objetivo, mas algumas oferecem serviços e ferramentas diferenciados.

Segundo ([WAKA, 2015](#)), bastante serviços são importantes para o usuário, sendo alguns deles: marcação de tempo dos dados recebidos, definição de regras para que uma atividade seja executada, entre outros.

A citação de ([SANTOS et al., 2014](#)) a seguir, explana como é o funcionamento de uma plataforma.

"A maioria destas plataformas baseiam suas funcionalidades de acordo com os modelos de dados definidos, assim, logo após coletados, os dados quando adequados ao modelo serão armazenados de forma a possibilitar sua consulta subsequente[...] O que geralmente ocorre, na prática, é a utilização de um modelo mais simples e genérico possível, que se adeque ao mais variado número de aplicações. Neste caso, os modelos que se encontram nas principais plataformas são baseados em *key-value* e *markup scheme*. Estes modelos são utilizados para que os usuários possam dar semântica aos seus dados, descrevendo coisas como os tipos dos dados, formatos, etc. Além disso, algumas outras meta informações também podem ser providas, como a localização dos sensores, uma descrição textual do que o sensor representa e algumas tags que poderão ser usadas como palavras-chave para consultas".

A tabela 4 apresenta algumas das principais plataformas utilizadas atualmente. No projeto não foi utilizado nenhuma plataforma específica para IOT pelo fato de que algumas dessas plataformas ainda estão em desenvolvimento; o que representa umas limitação no imenso poder da IoT. Plataformas bem elaboradas que atendem a todas as necessidades tendem a ser custosas financeiramente. Portanto no projeto é desenvolvido um servidor local com protocolos

<b>Plataforma</b>	<b>Endereço</b>	<b>Descrição</b>	<b>Tipo de Conta</b>
AWS IoT	aws.amazon.com/iot	Plataforma da Amazon voltada para empresas	Possui conta gratuita, porém é necessário o cadastro de um cartão de crédito para confirmar.
Linkafy	linkafy.com	Plataforma voltada para o controle de dispositivos residenciais.	Possui conta gratuita, mas limitada a apenas um dispositivo.
Microsoft Azure IoT	microsoft.com/iot	Plataforma da Microsoft voltada a IoT com foco empresarial	Possui conta gratuita de um mês para testes.
PubNub	pubnub.com	Plataforma robusta com diversas funcionalidades voltadas para IoT.	Possui conta gratuita com limitações.
Google Brillo	goo.gl/4k5nDs	Plataforma da Google voltada para o uso geral	Ainda não foi lançada.
ThingSpeak	thingspeak.com	Plataforma robusta, com várias funcionalidades, como sensores públicos e busca por histórico.	Conta gratuita disponível.
ThingSquare	thingsquare.com	Plataforma voltada ao controle de dispositivos e integração via celular.	Possui conta de desenvolvedor gratuita.
Electric imp	electricimp.com	Plataforma em nuvem que integra o conjunto de soluções dos dispositivos <i>electric imp</i> .	Possui conta gratuita.
Cayenne	mydevices.com	Plataforma voltada para automação e controle	Possui conta gratuita.
Johnny-five	johnny-five.io	Plataforma voltada para controle e robótica	Possui código aberto.

Tabela 4 – Tabela de Plataformas IoT.

disseminados pela comunidade científica. Isto acarreta em uma maior liberdade de controle do sistema. Como a IoT tem perceptiva de grande crescimento, é provável que futuramente estas plataformas sejam de baixo custo e de melhor qualidade, podendo ser utilizadas em novas versões do projeto.

### 4.2.2 NodeJS

O NodeJS é um ambiente de execução feito em C++ para aplicações web, construído em cima do motor V8 do google chrome ([NODEJS, 2017](#)). Surgiu com o intuito de melhorar os servidores web focado em *real-time* e executar códigos javascript fora do browser. Atualmente este ambiente é utilizado por inúmeras empresas do ramo da web como microsoft, walmart, ebay, yahoo e tantos outros. Ele tem seu destaque graças a apresentação de uma boa performance com consumo de memória e utilização do máximo poder de processamento dos servidores ([PEREIRA, 2014](#)).

Este ambiente e frameworks como o *express* e *socketio* é utilizado no projeto no micro-computador para atuar como servidor *Backend*<sup>6</sup> e servir a página *Frontend*. Este servidor, assim como toda a comunicação na rede do projeto utiliza os protocolos de requisição HTTP. Toda a comunicação entre servidor, módulos ESP e roteador utiliza este protocolo para se comunicar. Uma visão geral dessa rede pode ser vista na figura [30](#).

### 4.2.3 HTML, CSS e JS

O *frontend*<sup>7</sup> é desenvolvido utilizando as linguagens mais utilizadas para páginas web atualmente; o HTML, CSS e JavaScript. Essas linguagens são diferentes mas interagem entre si e são interpretadas por qualquer navegador web. Cada uma é responsável por partes importantes de uma página. O HTML é responsável pelos textos, estrutura e mídias da página. O CSS aplica estilos a tudo criado pelo HTML, ou seja, define cores, posição, tamanhos e inúmeras outras características de estilo. O JavaScript é responsável pela lógica de funcionamento da página, ou seja, é responsável pelo processamento visível e invisível ao usuário através de eventos.

No projeto é utilizado essas três linguagens para uma página web totalmente robusta, adequada e de fácil utilização para os usuários.

### 4.2.4 Arduino IDE

A IDE arduino (figura [27](#)) é uma aplicação multiplataforma, feita especialmente para pessoas pouco familiarizadas com desenvolvimento de softwares. É uma aplicação feita em JAVA e *open-source*, ou seja, código aberto.

Neste software, se encontra o próprio editor de códigos com recursos de realce de sintaxe, detector de erros e capaz de compilar e carregar programas para alguma placa automaticamente. Ele também contém uma área de mensagem, um console de texto, uma barra de ferramentas com botões para funções comuns e uma série de menus. Esta IDE foi projetada, em especial, para placas arduino, que utilizam microcontroladores *ATMEL*, mas atualmente ele oferece suporte para inúmeras placas microcontroladoras presentes no mercado, inclusive o microcontrolador *ESP8266*.

<sup>6</sup> Backend: programa que processa as requisições do usuário trazidas pelo frontend.

<sup>7</sup> Frontend: programa no qual o usuário faz requisições ao servidor.

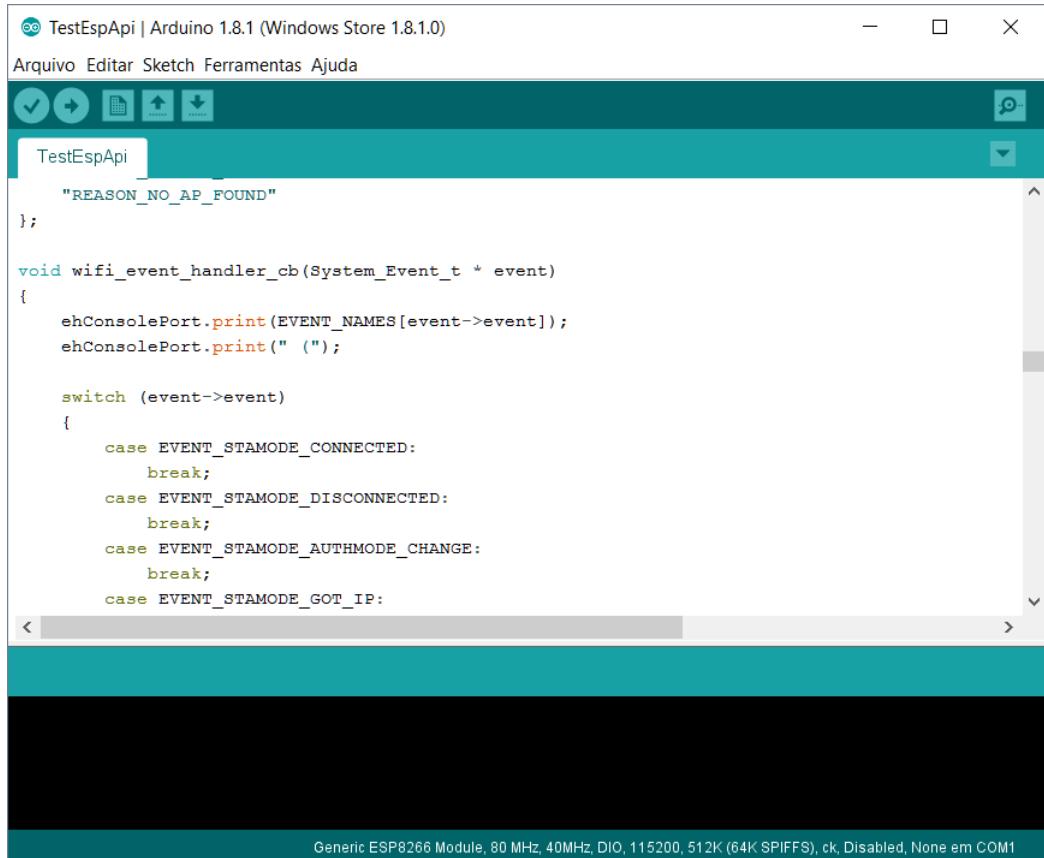


Figura 27 – *IDE Arduino (ARDUINO, 2017)*.

Para programar no Arduino IDE, é necessário escrever o código em linguagem C/C++ , é possível a adição de bibliotecas, o que se encontra em grandes quantidades na internet. Essa interface de desenvolvimento utiliza-se do software *avrdude*, para converter o código feito em um arquivo texto na codificação Hexadecimal, esse arquivo então é carregado por uma aplicação de firmware da placa. Portanto a IDE arduino é o ambiente de desenvolvimento escolhido para a programação dos módulos ESP8266 presentes no projeto.

#### 4.2.5 MobaXterm

O MobaXterm é um software gratuito, na versão básica, que oferece funções de programação remota, ou seja, este software cria terminais remotos de qualquer máquina com suporte a vários protocolos ([MOBAXTERM, 2017](#)).

No projeto, para uma maior agilidade é utilizado este software para programação remota do microcomputador através do protocolo SSH.

#### 4.2.6 Softwares diversos

Para execução do projeto vários programas de menor relevância é utilizado.A aplicações dos mesmos são descritas nos próximos capítulos, onde é explanado toda a execução do projeto.

## 4.3 Visão do Projeto

O projeto é feito com base em um apartamento, representado na figura 28, para exemplificar as principais aplicações de uma residência inteligente.



Figura 28 – Planta Baixa ([ACENGENHARIA, 2017](#)).

No projeto são feitos alguns hosts<sup>8</sup>, de forma a evitar fiação desnecessárias nos eletrodomésticos, ou seja, cada um recebe um módulo ESP8266 e terá seu próprio endereço. Essa distribuição é feita pela análise do projeto elétrico da residência, visto na imagem 29.

## 4.4 Rede e Conexões

O diagrama, na figura 30, representa a rede em uma visão macro, mostrando como é as conexões entre si e como o sistema é ligado a internet.

A figura 31, mostra em diagrama a rede em topologia estrela e apresenta os DI's de cada cômodo. Pela análise da planta elétrica, a distribuição dos DI's nos hosts é definida da seguinte forma:

- Host 1

<sup>8</sup> Host: em português "Hospedeiro" é qualquer máquina ou computador conectado a uma rede.

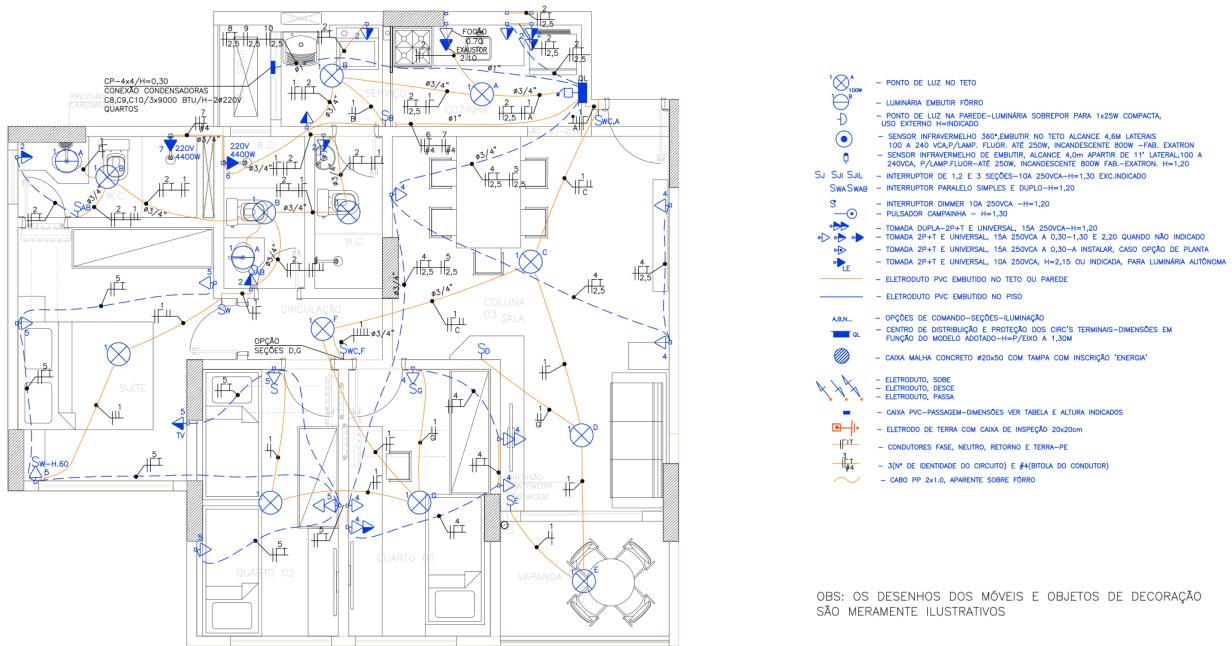


Figura 29 – Planta Elétrica (ACENGENHARIA, 2017).

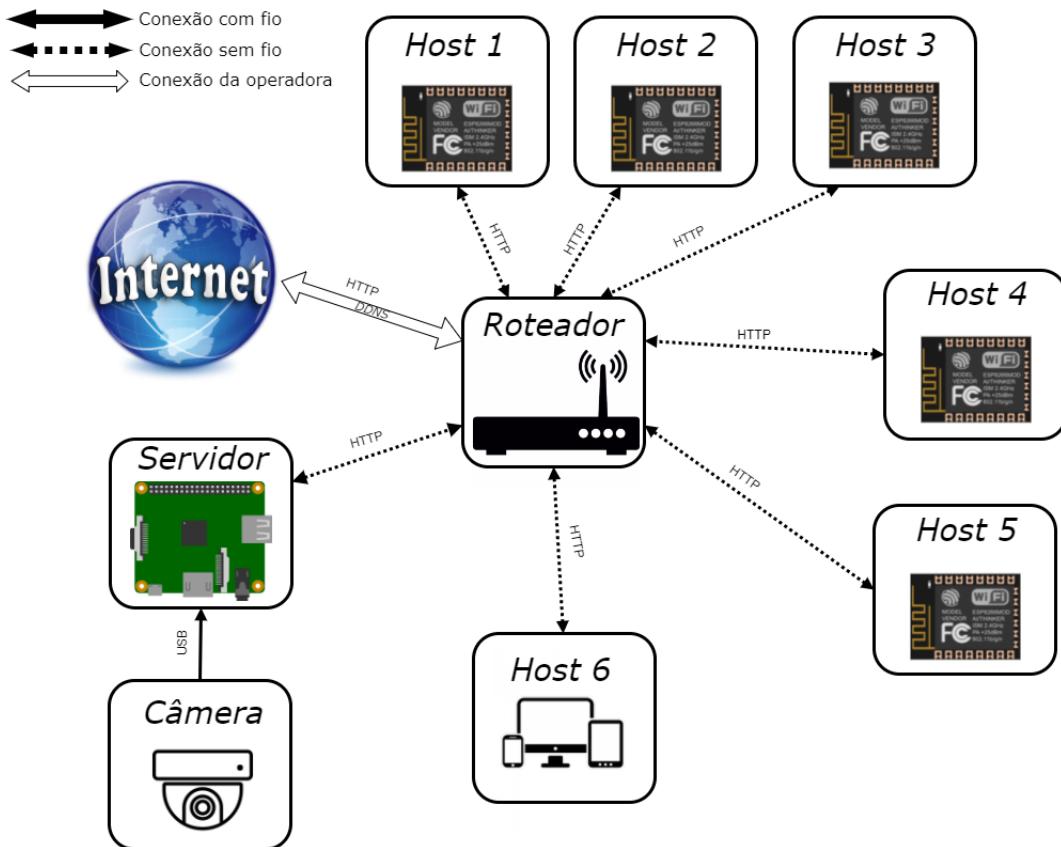


Figura 30 – Diagrama macro da rede (ELABORADA PELO AUTOR).

- Lâmpada Quarto 1
- Lâmpada Quarto 2

- Lâmpada Sala 2

- Host 2

- Lâmpada Sala 1
- Lâmpada Varanda
- Controle IR Sala

- Host 3

- Tomada Quarto 1
- Tomada Quarto 2
- Tomada Suíte

- Host 4

- Lâmpada Cozinha
- Lâmpada Área de Serviço
- Sensor de Gases
- Válvula Solenoide

- Host 5

- Lâmpada Suíte
- Lâmpada Banheiro
- Controle IR Suíte

#### 4.4.1 Conexões

A Conexão entre os módulos ESP8266 com o dispositivo controlador (smartphone, tablet, computador entre outros...) é feita por uma rede WiFi, através de um roteador. Neste caso, cada ESP8266 é configurado no modo *Station*. Caso não utilize roteador, o modo que os módulos ESP devem entrar é o *Acess Point*, ou seja, os próprios módulos rotearão uma rede; porém, no projeto é utilizado o modo *station*. Logo, os hosts poderão ser controlados se estiverem conectados na mesma rede, neste caso, na rede local (LAN). Para que seja possível o controle através da internet, em qualquer lugar do mundo através de uma rede (WAN) é necessário a abertura de portas de acesso no roteador e saber o seu ip externo, porém, o ip externo disponibilizado pelas empresas de fornecimento de internet é em sua maioria, dinâmico. Neste caso, se torna necessário a criação de um DDNS para criar um nome para rede e ela ser identificada na internet. Este serviço pode ser feito através de programas como o NoIp disponível na internet.

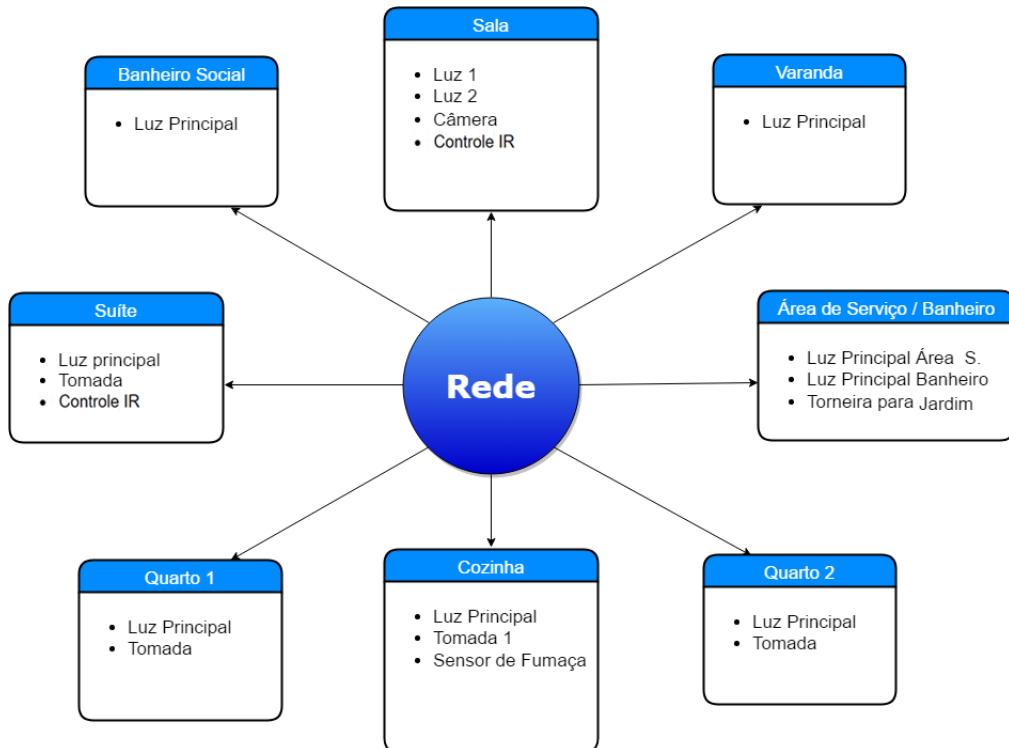


Figura 31 – Diagrama de DI's de cada cômodo na rede (ELABORADA PELO AUTOR).

## 4.5 Esquematização

O módulo ESP de cada cômodo é instalado na caixa da lâmpada principal, ou em caixas de passagem. A conexão entre o módulo com os hardwares de controle (relés e interruptores) é feita fisicamente, através de cabos. Este artifício só é possível para casas construídas com os padrões da ([NBR5410, 2004](#)), na qual os eletrodutos tem espaços para passagem de mais fios e geralmente, todos eles se encontram em um ponto: Telhado ou caixa de passagem. A figura 32 ilustra, de acordo com a norma, uma instalação de eletrodutos de alguns cômodos, e pode-se ver que todos eles se ligam ao ponto onde fica a luz principal.

### 4.5.1 Lâmpadas e Tomadas

Para os dispositivos Liga/Desliga, como luz principal é utilizado relés de estado sólido. Elas podem ser acionadas pelos interruptores ou pelo sistema. Uma alternativa para o acionamento duplo (interruptor e sistema) poderia ser com os interruptores tree-way, mas eles não são tão comuns nas residencias, ter que implementa-los em cada lâmpada tornaria o projeto muito mais custoso. Para superar esse problema, o esquema é utilizar o interruptor para acionar uma interrupção externa no microcontrolador ESP8266 e então ele acionar o relé. Isto faz a mesma função da utilização do interruptor tree-way. O esquema é melhor apresentado nos capítulos posteriores. No caso das tomadas, a implementação será feita semelhante a forma que o sistema de acionamento das lâmpadas. A diferença é que não haverá botão físico de acionamento, visto que a tomada ficará normalmente ligada e o sistema apenas desligará, conforme o desejo do

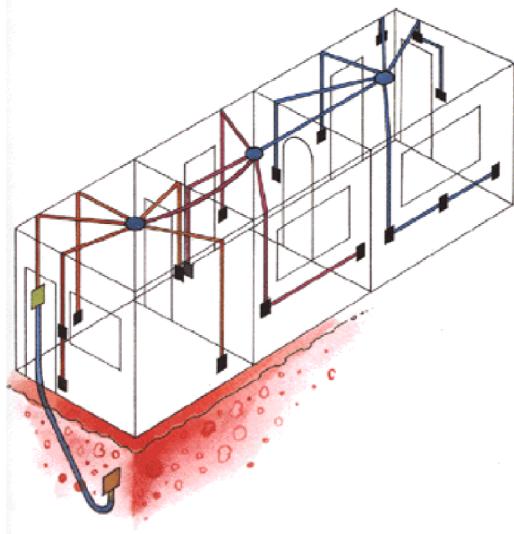


Figura 32 – Instalação elétrica de alguns cômodos (RÔMULO, 2013).

usuário.

#### 4.5.2 Eletrodomésticos Controlados por Infra-Vermelho

Para dispositivos como aparelho de ar condicionado, televisão, receptor parabólica e home theater, o controle é feito através de um LED infra-vermelho, instalado junto ao ESP e direcionado para o equipamento. Primeiramente, com o controle original do equipamento, é feita uma análise dos códigos já existentes para cada função, estes códigos então serão clonados para o ESP. O ESP então ficará encarregado de controlar um LED, com o código para cada função desejada. O método de implementação será como ilustrado na figura 33

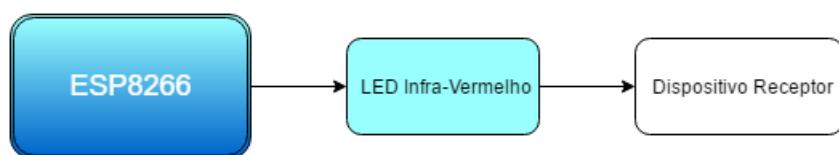


Figura 33 – Sistema de controle com IR (ELABORADA PELO AUTOR).

#### 4.5.3 Detecção de Incêndio

Para a detecção de incêndio, será utilizado um sensor de gás localizado na cozinha, este sensor será ligado ao conversor A/D, presente no módulo ESP. Em caso de presença de fumaça ou algum gás inflamável, o módulo será responsável por criar um alerta através da internet.

#### 4.5.4 Torneira Autônoma

Para regar jardins ou ter controle de vazão de água para qualquer finalidade, é utilizado uma válvula solenoide em uma torneira. Como seu funcionamento é semelhante ao relé, o

acionamento em software pelo ESP será idêntico, porém no hardware é feito circuitos de driver, visto que o microcontrolador não fornece corrente nas suas portas e trabalham com apenas 3,3V. Logo o driver irá fazer com que a porta forneça corrente a 12V. A torneira é programada para todos os dias a uma determinada hora ser acionada por 20 segundos.

# 5

## Execução em Software

### 5.1 Arduino IDE

Para o programar o microcontrolador ESP8266 na plataforma Arduino IDE é necessário fazer alguns procedimentos. O primeiro é fazer com o IDE reconheça o microcontrolador. Com a IDE aberta, no menu "Arquivo > Preferências" coloca-se o seguinte link na caixa "URLs Adicionais de Gerenciadores de placas": "[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)"

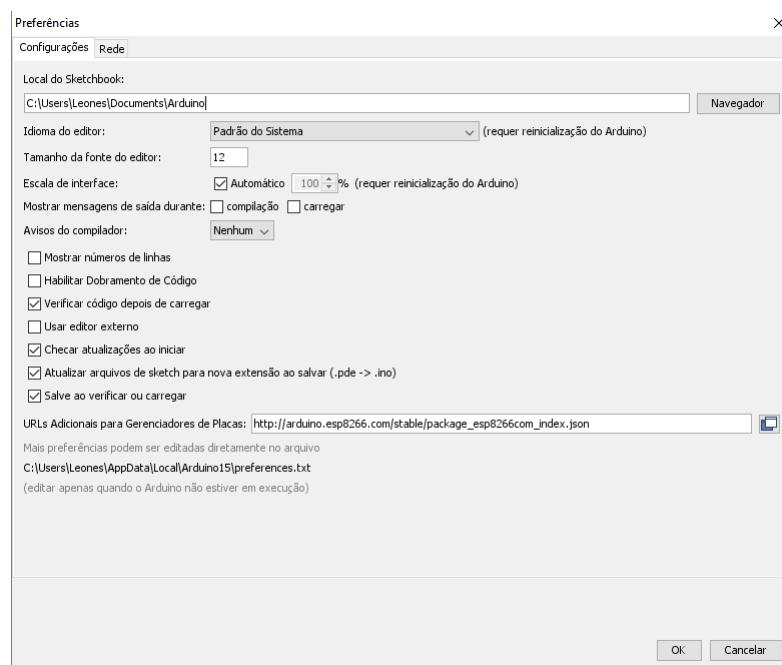


Figura 34 – Adicionamento do ESP no Arduino IDE

Esse procedimento pode ser visto na figura 34. Após ter adicionado o link, clica em "OK" e vai no menu "Ferramentas > Placa > Gerenciador de Placas" Procura o "esp8266 by ESP8266 Community" como mostra a imagem 35 e clica em instalar. É preciso está conectado a internet para poder ser feito o download dos pacotes.

Após o término da instalação o Arduino IDE já está preparado para programar os módulos ESP8266. No menu "Ferramentas > Placa" se encontra todas as placas que o Compilador suporta.

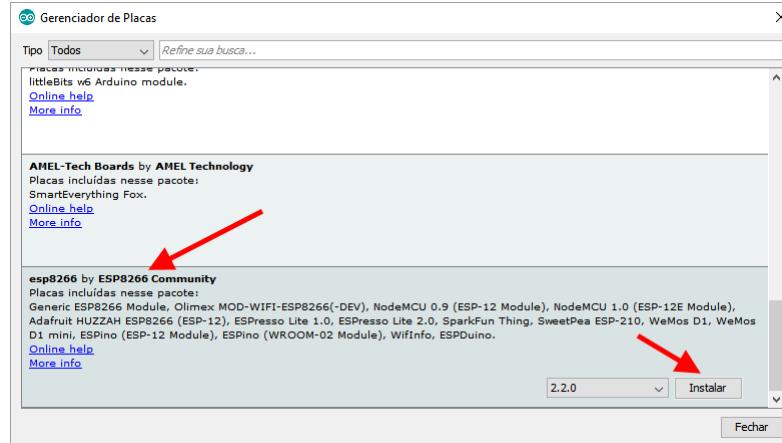


Figura 35 – Adicionamento das placas ESP no Arduino IDE

No projeto é utilizado o ESP8266 da família 12 na placa nodeMCU então é selecionada a mesma, como na imagem 36.

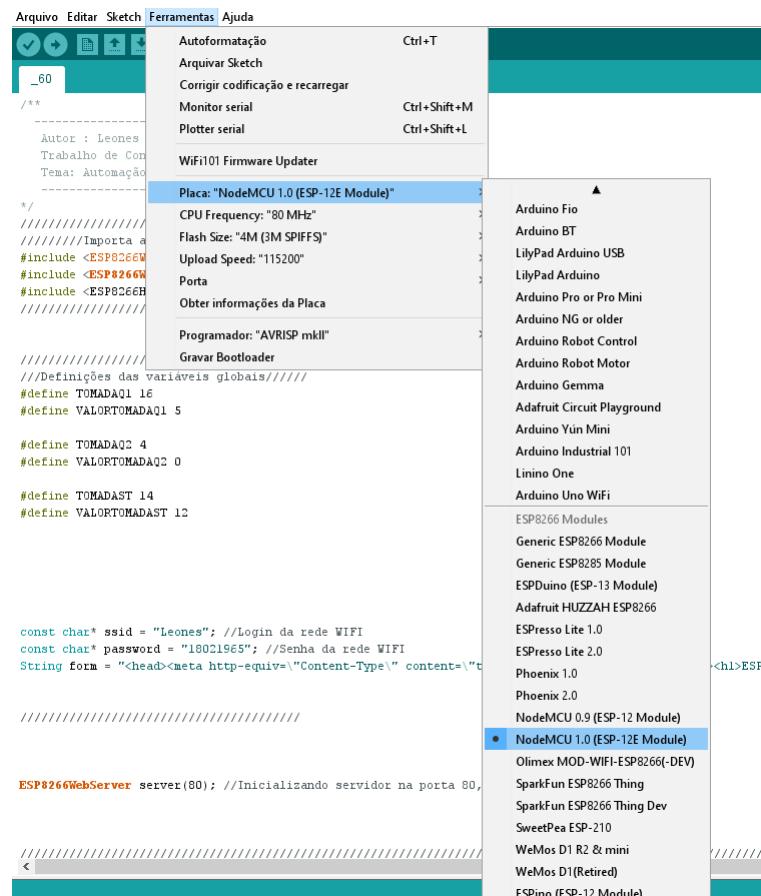


Figura 36 – Seleção da placa nodeMCU para programação.

## 5.2 Módulos ESP8266

### 5.3 Servidor

Como servido é usado um microcomputador orangepi modelo lite. Para instalar o SO é feito os seguintes procedimentos:

- Download da imagem do SO Armbian no site: <<http://www.orangepi.org/downloadresources/>>.
- Formata o microSD com o programa SDFormat, visto na imagem 37.
- Grava a imagem do sistema operacional no microSD com o programa win32diskimage, visto na figura 38.
- Coloca o microSD no Orangepi e o liga na fonte e em um monitor, teclado e mouse.

Após ter iniciado o SO, é feito o procedimento para conectar a WIFI. No terminal, instala-se o pacote wpa\_supplicant com o seguinte comando:

```
» apt-get install wpa_supplicant
```

Logo após configura-se o arquivo como a figura 39 no seguinte caminho:

```
»cd /etc/network/interfaces
```

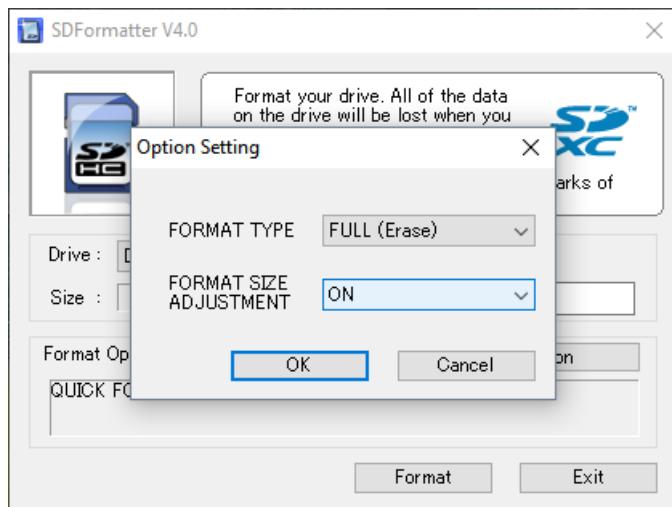


Figura 37 – Formatação do Cartão SD.

Neste caso, o login da rede é "Leones" e a senha é "18021965". Para estabelecer a conexão, o orangepi é resetado com o seguinte comando:

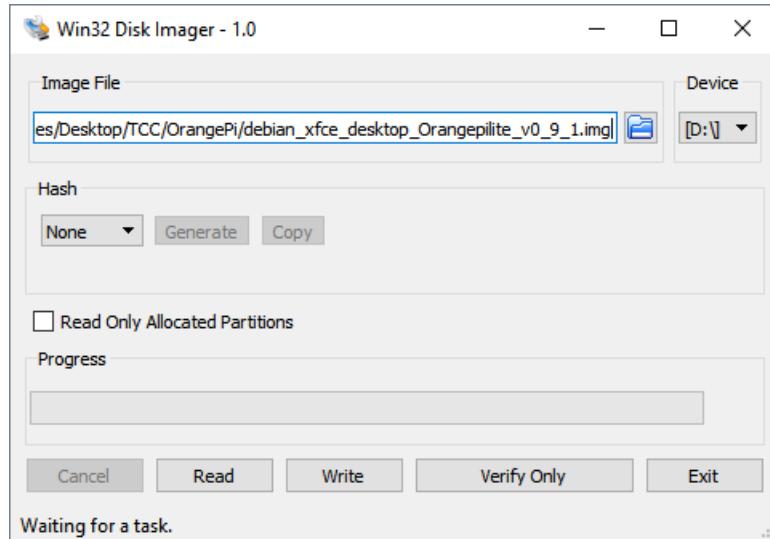


Figura 38 – Gravação do SO no cartão SD.

```
source /etc/network/interfaces.d/*
# This file intentionally left blank
#
# All interfaces are handled by network-manager, use nmtui or nmcli on
# server/headless images or the "Network Manager" GUI on desktop images
auto wlan0
iface wlan0 inet dhcp
    wpa_ssid Leones
    wpa_psk 18021965
```

Figura 39 – Configuração de rede WiFi

» sudo reboot

Com o Wifi configurado, o SO está aberto para ser operado remotamente através do SSH. Porém é preciso saber qual seu ip. Para isso foi utilizado o comando ifconfig, como na figura 40.

O Ip neste caso é o "10.0.0.3". Porém é um ip dinâmico, caso o servidor desligue e ligue, é possível que este ip mude, ainda mais por ser um ip baixo. Então é de extrema importância fixar este ip. Para isso é necessário ter o endereço MAC do microcomputador que de acordo com a imagem 40 é "0e:65:3e:fe:ad:43". Em posse do MAC, entra-se nas configurações do roteador através do seu ip "10.0.0.1", neste caso. O Roteador utilizado é da fabricante Intelbras modelo WRN150. Após o login, no menu DHCP»Lista de Clientes é possível colocar o MAC do microcomputador e atribuir um ip fixo. A figura 41 explana as configurações feitas.

Sabendo então que o Ip fixo é "10.0.0.100"então pode-se utilizar do MobbaXterm para operações remotas através do SSH. Após a configuração do MobbaXterm com esse IP, o ambiente de operação é como visto na figura 42.

Para atualizar o SO são dados os seguintes comandos:

```

root@orangepileite:~# ifconfig
lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING MTU:16436 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

wlan0    Link encap:Ethernet HWaddr 0e:65:3e:fe:ad:43
        inet addr:10.0.0.3 Bcast:10.0.0.255 Mask:255.255.255.0
        inet6 addr: fe80::c65:3eff:fe:ad43/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:48542 errors:0 dropped:475 overruns:0 frame:0
              TX packets:23192 errors:0 dropped:1 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:68103390 (64.9 MiB) TX bytes:2368529 (2.2 MiB)

root@orangepileite:~# ■

```

Figura 40 – Informações da Rede Conectada

The screenshot shows the configuration interface for an Intelbras WRN 150 Wireless Router. The top navigation bar includes links for 'Configurações' (Configurations), 'Wireless', 'DHCP' (selected), 'NAT/DMZ', 'Firewall', 'Roteamento', and 'Sistema'. Below the navigation bar, a sub-menu for 'DHCP' is open, specifically the 'Lista de Clientes' (List of Clients) section.

In the 'Atribuição Estática' (Static Assignment) section, there is a form to assign a static IP address. The 'Endereço IP' field contains '10.0.0.100'. Below it, the 'Endereço MAC' field displays '0E:65:3E:FE:AD:43'. A 'Adicionar' (Add) button is located to the right of the MAC address input.

A table titled 'Ativar' (Activate) lists the assigned static IP entries:

ID	Endereço IP	Endereço MAC	Deletar
1	10.0.0.100	0E:65:3E:FE:AD:43	<input type="button" value="Deletar"/>

Below the table, there is a 'Atualizar' (Update) button. At the bottom of the page, there are 'Salvar' (Save) and 'Cancelar' (Cancel) buttons.

Figura 41 – Configuração para IP estático no roteador.

- » sudo apt-get update
- » sudo apt-get upgrade

Com o sistema atualizado, é instalado o nodeJS e frameworks utilizadas, com os seguintes

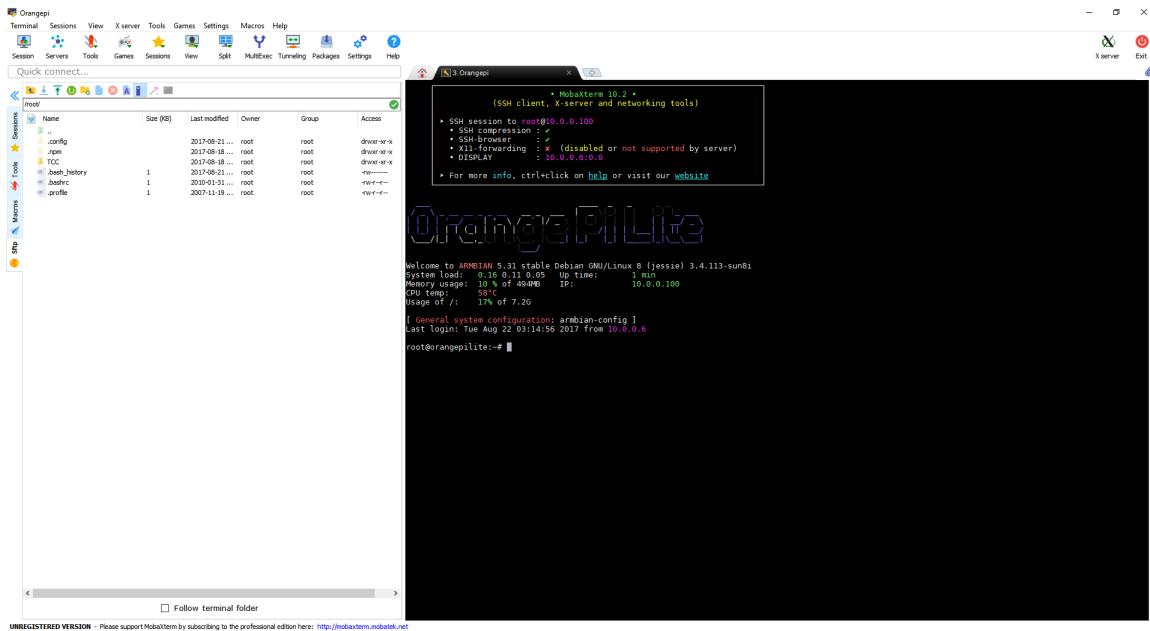


Figura 42 – Configuração MobbaXterm

comandos:

```
» sudo apt-get install npm
» curl -sL https://deb.nodesource.com/setup_7.x | sudo -E bash -
» sudo apt-get install nodejs
» sudo npm install express --save
» sudo npm install request
```

## **5.4 Interface com o Usuário**

# 6

## Execução em Hardware

### 6.1 Módulos ESP

#### 6.1.1 Host 1

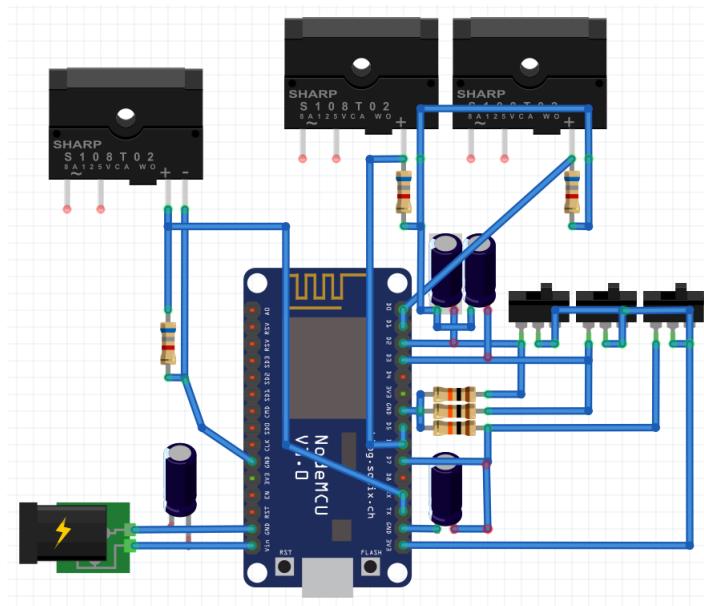


Figura 43 – Esquemático Host 1.

#### 6.1.2 Host 2

#### 6.1.3 Host 3

#### 6.1.4 Host 4

#### 6.1.5 Host 5

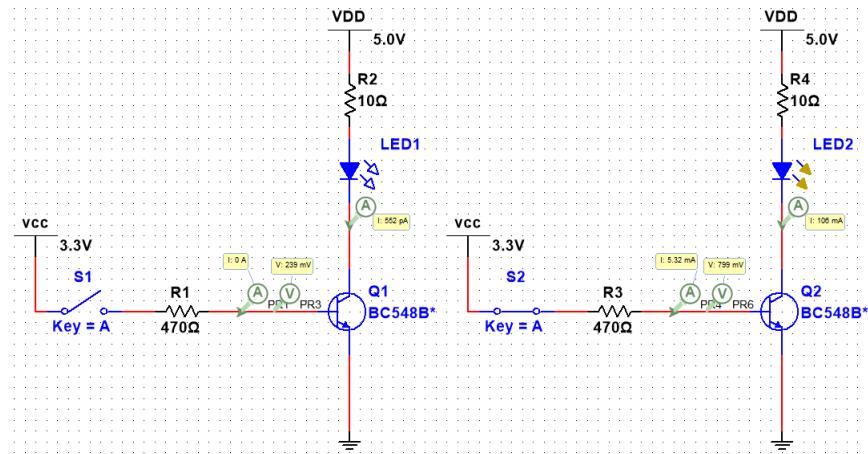


Figura 44 – Circuito para Acionamento do LED IR.

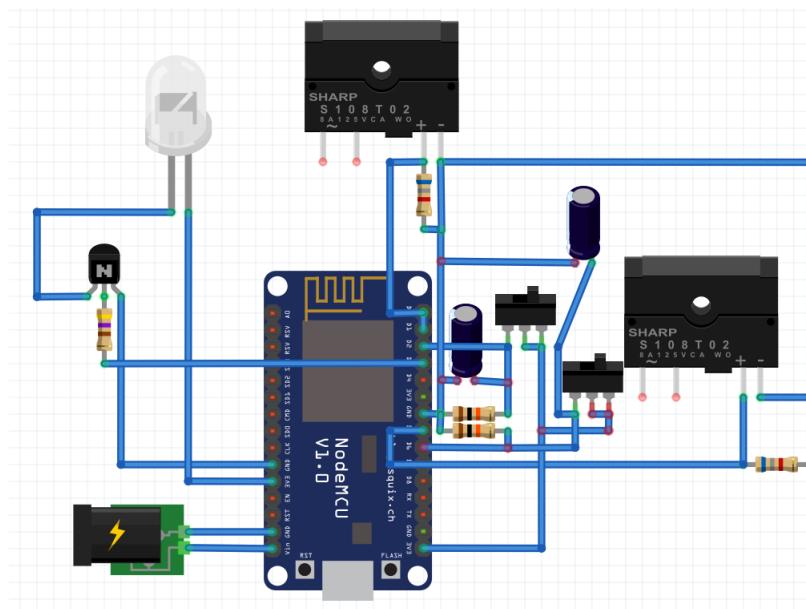


Figura 45 – Esquemático Host 2

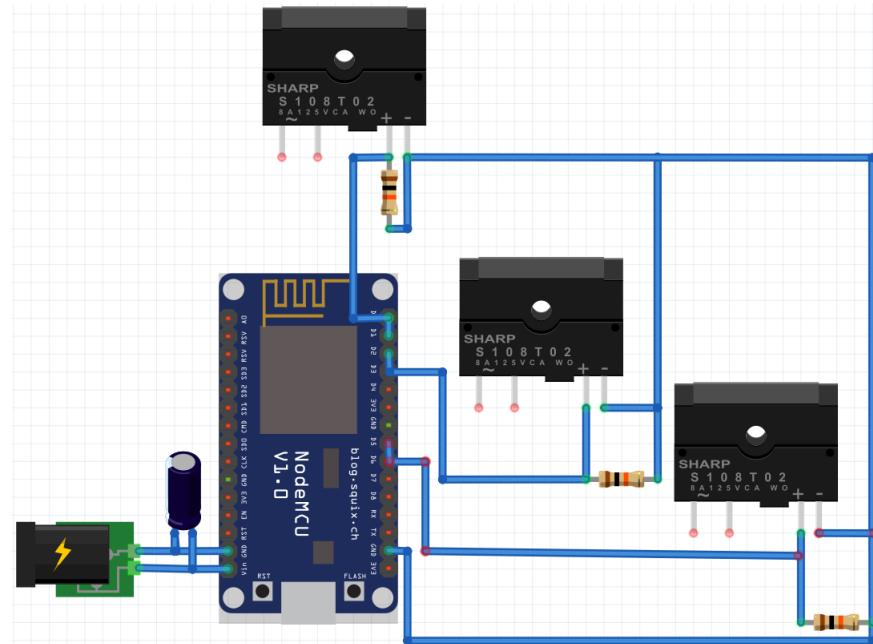


Figura 46 – Esquemático Host 3

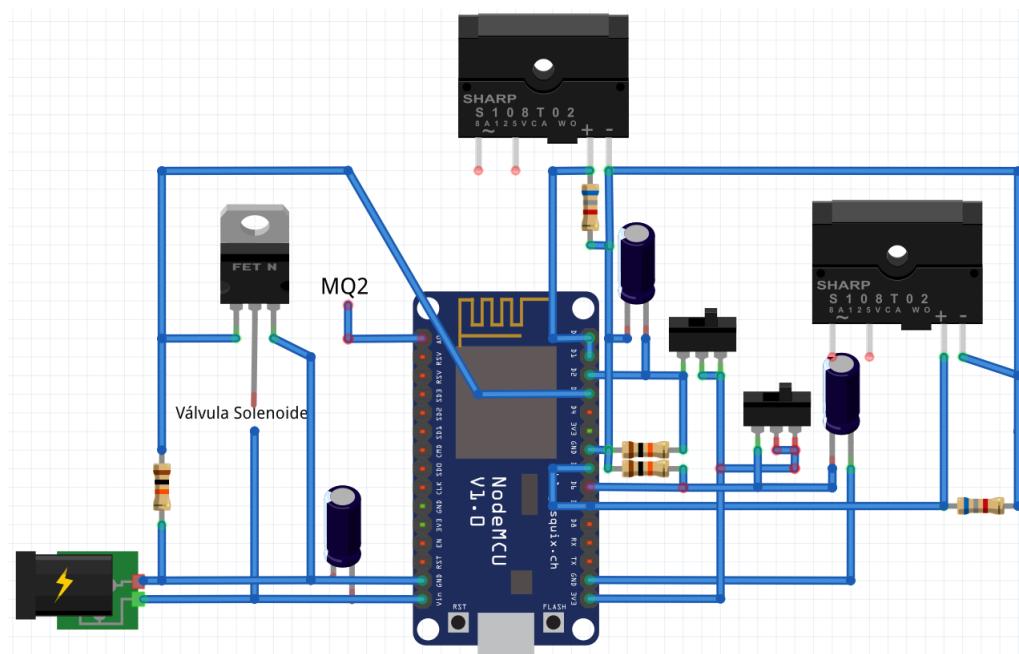


Figura 47 – Esquemático Host 4

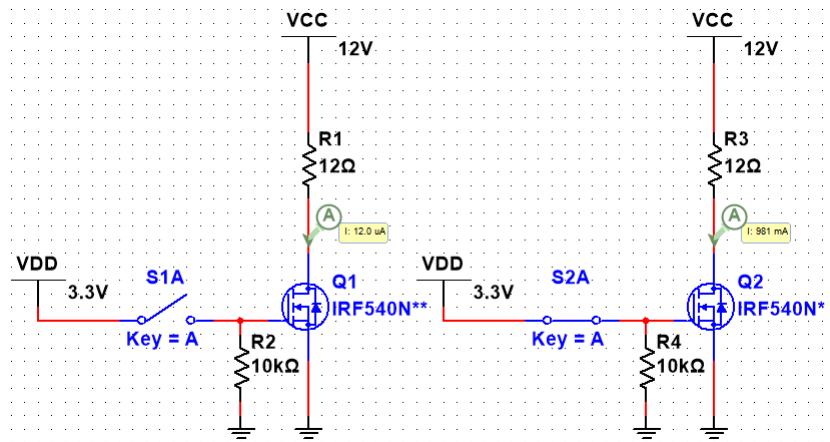


Figura 48 – Circuito para Acionamento da válvula solenoide

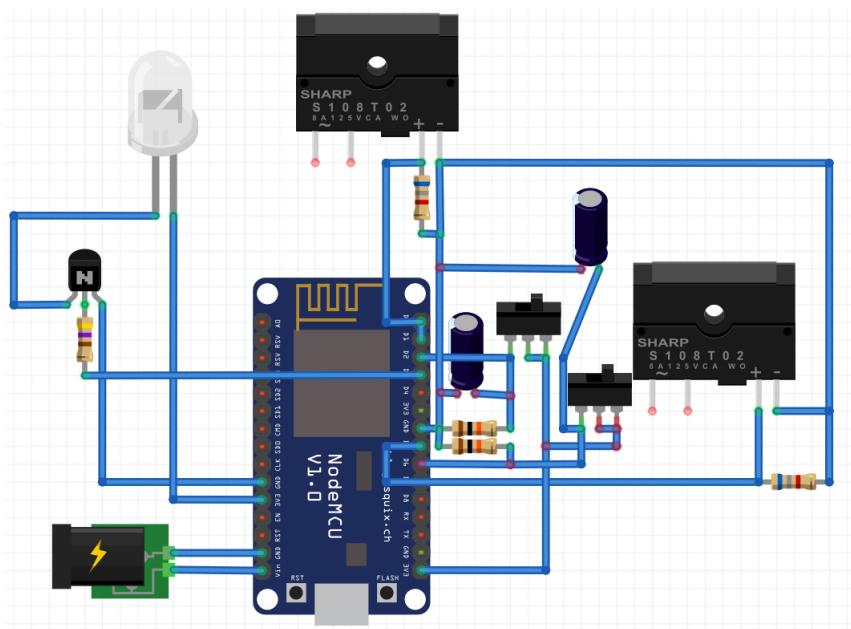


Figura 49 – Esquemático Host 5

# 7

## Resultados

<b>Componente</b>	<b>Especificações</b>	<b>Quantidade</b>	<b>Preço Exterior</b>	<b>Preço Nacional</b>
OrangePi	Modelo Lite	1	U\$ 15.00	R\$ 120
ESP8266	Modelo 12 (nodeMCU)	5	U\$	R\$
WebCam	Modelo não especificado	1	U\$	R\$
SSR	Modelo OMRON G3MB-202P	12	U\$	R\$
Válvula Solenoide	Modelo não especificado	1	U\$	R\$
Roteador	Modelo WRN 150 Intelbras	1	U\$	R\$
LED IR	Modelo TIL32	6	U\$	R\$
Receptor IR	PIC13043	1	U\$	R\$
Transistor	BC548B	2	U\$	R\$
Cabo UTP		20 metros	U\$	R\$
Fonte de Alimentação	Modelo Carregador de Celular	5	U\$	R\$
Resistores	Diversos 1/4W	5	U\$	R\$
Placa para PCI	Feneolite ilhada	5	U\$	R\$
Sensor de Gases	Modelo MQ-2	1	U\$	R\$

Tabela 5 – Tabela de Custos.

# 8

## **Conclusão e Sugestões para trabalhos futuros**

# Referências

- ACENGENHARIA. *Planta Baixa*. 2017. Disponível em: <<http://acengenhariase.com.br/leitura-empreendimento/5/perolas-do-luzia>>. Acesso em: 10 jan 2017. Citado 3 vezes nas páginas 7, 33 e 34.
- ANDERSON, J. The internet of things will thrive by 2025. *Pew Research Internet Project*, v. 14, 2014. Citado na página 11.
- ANSI/EIA-600. *Consumer Electronic Bus*. 1995. Citado na página 8.
- ANSI/TIA/EIA-570-A. *Residential Telecommunication Cabling Standard*. 2003. Disponível em: <[http://www.vyskocil.net/firma/dokumenty/data/tct21\\_16\\_TIA570B\\_Residential\\_cabling.pdf](http://www.vyskocil.net/firma/dokumenty/data/tct21_16_TIA570B_Residential_cabling.pdf)>. Acesso em: 04 jan 2017. Citado na página 8.
- APLEX. *Internet of Things*. 2016. Disponível em: <<http://aplex.com.br/sem-categoria/internet-of-things-eset-ressalta-importancia-da-protectao-dos-dispositivos-conectados>>. Acesso em: 17 dez 2016. Citado 2 vezes nas páginas 7 e 9.
- ARDUINO. Website arduino. 2017. Acesso em: 24 jan 2017. Disponível em: <<https://www.arduino.cc/>>. Citado 2 vezes nas páginas 7 e 32.
- AUTOCORE, R. *NodeMCU v3 Lolin*. 2017. Disponível em: <<https://www.autocorerobotica.com.br/nodemcu-v2-kit-de-desenvolvimento-com-esp8266-baseado-em-lua>>. Acesso em: 10 jan 2017. Citado 2 vezes nas páginas 7 e 18.
- BALLAGUER, A. *Internet das Coisas – Das origens ao futuro*. 2014. Disponível em: <[https://www.tiespecialistas.com.br/2014/09/internet-das-coisas-das-origens-ao-futuro/#\\_ftnref2](https://www.tiespecialistas.com.br/2014/09/internet-das-coisas-das-origens-ao-futuro/#_ftnref2)>. Acesso em: 11 fev 2017. Citado na página 11.
- BOLZANI, C. A. M. *Residências inteligentes*. [S.l.]: Editora Livraria da Física, 2004. Citado 5 vezes nas páginas 7, 4, 5, 9 e 12.
- BRUGNERA, M. R. Domótica. 2007. Citado na página 4.
- CARVALHO, C. 'Tudo será a internet'. *Físico faz previsões de como vamos viver daqui 10 anos*. 2012. Disponível em: <<https://img.olhardigital.uol.com.br/noticia/24089/24089?nohits>>. Acesso em: 11 abr 2017. Citado na página 10.
- CONCEIçãO, B. Fundamentos da internet das coisas. 2016. Citado 2 vezes nas páginas 13 e 14.
- CURVELLO, A. Apresentando o módulo esp8266. 2015. Acesso em: 04 jan 2017. Disponível em: <<https://www.embarcados.com.br/modulo-esp8266>>. Citado 4 vezes nas páginas 7, 16, 17 e 18.
- DOUKAS, C. *Building Internet of Things with the ARDUINO*. [S.l.]: CreateSpace Independent Publishing Platform, 2012. Citado 2 vezes nas páginas 5 e 9.
- ESPRESSIF. *ESP8266EX Datasheet*. 2016. Disponível em: <[http://espressif.com/sites/default/files/documentation/0a-esp8266ex\\_datasheet\\_en.pdf](http://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf)>. Acesso em: 06 jan 2017. Citado 2 vezes nas páginas 15 e 19.

- FERREIRA, J. A. O. *Interface homem-máquina para domótica baseado em tecnologias Web.* [S.l.]: Junho, 2008. Citado na página 13.
- FINDER, B. O que você precisa saber sobre relés? 2014. Acesso em: 23 jan 2017. Disponível em: <<http://www.findernet.com/en/node/47658>>. Citado 3 vezes nas páginas 7, 21 e 22.
- IEEE. *IEEE 802.11 WIRELESS LOCAL AREA NETWORKS.* 2017. Disponível em: <<http://www.ieee802.org/11/>>. Acesso em: 11 abr 2017. Citado na página 11.
- JDR. Válvula solenoide 12v. 2017. Disponível em: <<http://www.jdreletronicos.com.br/pd-2700f8-valvula-solenoid-12v.html>>. Acesso em: 24 fev 2017. Citado 2 vezes nas páginas 7 e 27.
- LOMBARDI, R. Controle remoto infravermelho para automação. 2006. Citado 3 vezes nas páginas 7, 24 e 25.
- MOBAXTERM. *MobaXterm.* 2017. Disponível em: <<http://mobaxterm.mobatek.net/>>. Acesso em: 1 fev 2017. Citado na página 32.
- NBR16264, A. *Cabeamento Estruturado Residencial.* 2014. Disponível em: <<https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxhbWF1cnl3YWxiZXJ0fGd4OjM4MDZIMjA5ODFjYmIyNzk>>. Acesso em: 06 jan 2017. Citado na página 8.
- NBR5410, A. *Instalações Elétricas em Baixa Tensão.* 2004. Disponível em: <[http://www.iar.unicamp.br/lab/luz/ld/normas%20e%20relat%F3rios/NRs/nbr\\_5410.pdf](http://www.iar.unicamp.br/lab/luz/ld/normas%20e%20relat%F3rios/NRs/nbr_5410.pdf)>. Acesso em: 05 jan 2017. Citado 2 vezes nas páginas 8 e 36.
- NBR5413, A. *Iluminação de Interiores.* 1992. Disponível em: <<http://www.unicep.edu.br/biblioteca/docs/engenhariacivil/ABNT%205413%20-%20ilumin%C3%A2ncia%20de%20interiores%20-%20procedimento.pdf>>. Acesso em: 05 jan 2017. Citado na página 8.
- NBR7198, A. *Projeto e Execução de Instalações Prediais de Água Quente.* 1993. Disponível em: <[http://fauufrjatelierintegrado1.weebly.com/uploads/1/2/5/9/12591367/nbr\\_7198\\_1993\\_-projeto\\_e\\_execucao\\_instal\\_agua\\_quente.pdf](http://fauufrjatelierintegrado1.weebly.com/uploads/1/2/5/9/12591367/nbr_7198_1993_-projeto_e_execucao_instal_agua_quente.pdf)>. Acesso em: 05 jan 2017. Citado na página 8.
- NODEJS. *NODEJS.* 2017. Disponível em: <[nodejs.org](http://nodejs.org)>. Acesso em: 10 fev 2017. Citado na página 31.
- NOVAK, M. *Nikola Tesla's Incredible Predictions For Our Connected World.* 2015. Disponível em: <<http://paleofuture.gizmodo.com/nikola-teslas-incredible-predictions-for-our-connected-1661107313>>. Acesso em: 11 jan 2017. Citado na página 10.
- ORANGEPI. *OrangePi.* 2017. Disponível em: <<http://www.orangepi.org>>. Acesso em: 1 jul 2017. Citado 2 vezes nas páginas 7 e 20.
- PEREIRA, C. R. *Aplicações web real-time com Node.js.* [S.l.]: Editora Casa do Código, 2014. Citado na página 31.
- QUINDERÉ, P. Casa inteligente – um protótipo de sistema de automação residencial de baixo custo. 2009. Citado na página 7.
- RÔMULO, A. O. Projeto de instalações elétricas. 2013. Citado 2 vezes nas páginas 7 e 37.

- ROVAI, M. Sistema automático para irrigação e calor. 2016. Disponível em: <<https://mjrobot.org/tag/esp8266/>>. Acesso em: 17 jan 2017. Citado 2 vezes nas páginas 7 e 18.
- SANOU, B. *Facts And Figures ICT 2016*. 2016. Disponível em: <<http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2016.pdf>>. Acesso em: 01 mar 2017. Citado 3 vezes nas páginas 7, 1 e 2.
- SANTOS, B. P. et al. Internet das coisas: da teoria à prática. 2014. Citado na página 29.
- SILEVIRA, W. L. L. Um breve histórico conceitual da automação industrial e redes para automação industrial . 2003. Citado na página 4.
- SILVA, C. Válvula solenóide. 2011. Disponível em: <<http://acquaticos.blogspot.com.br/2010/10/valvula-solenoide.html>>. Acesso em: 30 jan 2017. Citado 2 vezes nas páginas 7 e 27.
- SINGH, G. *NEC Protocol IR*. 2013. Disponível em: <<http://www.circuitvalley.com/2013/09/nec-protocol-ir-infrared-remote-control.html>>. Acesso em: 06 set 2017. Citado 2 vezes nas páginas 7 e 26.
- TWITTY, A. *Smart Home Trends for homeowners in 2017*. 2016. Disponível em: <<http://realtybiznews.com/smart-home-trends-for-homeowners-in-2017/98736453/>>. Acesso em: 10 abr 2017. Citado na página 5.
- WAKA, G. Controle remoto de tomadas elétricas baseado nos conceitos da internet das coisas. 2015. Citado 4 vezes nas páginas 7, 6, 13 e 29.
- ZARGHAMI, S. Middleware for internet of things. University of Twente, 2013. Citado na página 12.

# ANEXO A – Programação ESP8266

## HOST 2

```
/*
-----
* Autor : Leones Moura
* Trabalho de Conclusão de Curso em Engenharia Eletrônica
* Tema: Automação Residencial Utilizando IoT
* -----
*/
//////////Importa as bibliotecas//////////
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <ESP8266HTTPClient.h>
#ifndef UNIT_TEST
#include <Arduino.h>
#endif
#include <IRremoteESP8266.h>
#include <IRsend.h>
//////////Definições das variáveis globais////
#define LAMPADAS1 5
#define INTERRUPTORLAMPADAS1 4
#define VALORLAMPADAS1 16
#define LAMPADAV 14
#define INTERRUPTORLAMPADAV 12
#define VALORLAMPADAV 13
#define LEDIR 15
int interruptor1=0;
int interruptor2=0;
//Definindo saída de Sinal IR
IRsend irsend(LEDIR);
const char* ssid = "Leones"; //Login da rede WIFI
const char* password = "18021965"; //Senha da rede WIFI
//Pagina Inicial em html para testes e supervisão
String form = "<html><head><title>Módulo Sala</title> <meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\"> </head><body><h1>ESP8266 Módulo Lâmpadas SALA1, SALA2 e IR </h1><p> Valores --> /all</br>Lâmpada Sala 1 --> /ls1</br>Lâmpada Varanda --> /lv </br>Liga/Desliga IR --> /power</br>Subir canal IR --> /ch+</br>Descer canal IR --> /ch-</br>Subir volume IR --> /v+</br>Descer volume IR --> /v-</br>Confirma IR --> /ok</br>Cima IR--> /up</br>Baixo IR--> /down<br>Esquerda IR--> /left</br>Direita IR--> /right</br>Entradas IR--> /sorce<br> </p></body></html>"; //Pagina Inicial em html para testes
//////////ESP8266WebServer server(80); //Define servidor na porta padrão http (80)
//////////FUNÇÃO PARA CONECTAR A REDE WIFI/////////
void wifiConnect(const char* ssid, const char* password){
int WiFiCounter = 0;
//Desconecta das wifi's e atualiza o modo para Station (conectado a uma rede existente)
WiFi.disconnect();
WiFi.mode(WIFI_STA); //Define o modo Station
```

```
//Inicializa a wifi com a rede e senha
WiFi.begin(ssid, password);
//Define um ip fixo junto com a mascara de sub-rede e gateway
IPAddress subnet(255, 255, 255, 0);
WiFi.config(IPAddress(10,0,0,55),
IPAddress(10,0,0,1), subnet);
//Aguarda conectar-se
while (WiFi.status() != WL_CONNECTED && WiFiCounter < 30) {
delay(1000);
WiFiCounter++;
}
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
}
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
void serverInit(){
//Coloca o servidor para ouvir a página inicial
server.on("/", [](){
server.send(200, "text/html", form);
});
//Define caminhos para chamar eventos
server.on("/all", allFunction);
server.on("/ls1", LampadaS1Function);
server.on("/lv", LampadaVFunction);
server.on("/power", PowerIR );
server.on("/v+", VupIR);
server.on("/v-", VdownIR);
server.on("/ch+", CHupIR);
server.on("/ch-", CHdownIR);
server.on("/up", UPIR);
server.on("/down", DOWNIR);
server.on("/left", LEFTIR);
server.on("/right", RIGHTIR);
server.on("/ok", OKIR);
server.on("/sorce", SOURCEIR);
//Inicializa o servidor.
server.begin();
}
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
void allFunction(){
server.send(200, "text/plain", (" ls1:" + String(digitalRead(VALORLAMPADAS1)) + "\n lv:" + String(digitalRead(VALORLAMPADAS1)) + "\n up:" + String(digitalRead(VALORLAMPADAS1)) + "\n down:" + String(digitalRead(VALORLAMPADAS1)) + "\n left:" + String(digitalRead(VALORLAMPADAS1)) + "\n right:" + String(digitalRead(VALORLAMPADAS1)) + "\n ok:" + String(digitalRead(VALORLAMPADAS1)) + "\n sorce:" + String(digitalRead(VALORLAMPADAS1)) + "\n"));
}
void PowerIR(){
irsend.sendNEC(0xFDC03F, 32);
server.send(200, "text/plain", "Power");
}
void VupIR(){
irsend.sendNEC(0xFD6897, 32);
server.send(200, "text/plain", "V+");
}
void VdownIR(){
irsend.sendNEC(0xFD58A7, 32);
server.send(200, "text/plain", "V-");
}
void CHupIR(){
irsend.sendNEC(0xFD28D7, 32);
```

```
server.send(200, "text/plain", "CH+");
}
void CHdownIR(){
irsend.sendNEC(0xFD18E7, 32);
server.send(200, "text/plain", "CH-");
}
void UPIR(){
irsend.sendNEC(0xFDB847, 32);
server.send(200, "text/plain", "UP");
}
void DOWNIR(){
irsend.sendNEC(0xFDA25D, 32);
server.send(200, "text/plain", "DOWN");
}
void LEFTIR(){
irsend.sendNEC(0xFD02FD, 32);
server.send(200, "text/plain", "LEFT");
}
void RIGHTIR(){
irsend.sendNEC(0xFD42BD, 32);
server.send(200, "text/plain", "RIGHT");
}
void OKIR(){
irsend.sendNEC(0xFD827D, 32);
server.send(200, "text/plain", "OK");
}
void SORCEIR(){
irsend.sendNEC(0xFD7887, 32);
server.send(200, "text/plain", "SORCE");
}
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////FUNÇÕES PARA LAMPADAS/////////////////////////////////////////////////////////////////
void LampadaS1Function(){
if(digitalRead(VALORLAMPADAS1))
{
digitalWrite(LAMPADAS1, LOW);
server.send(200, "text/plain", String(digitalRead(VALORLAMPADAS1)));
}
else{
digitalWrite(LAMPADAS1, HIGH);
server.send(200, "text/plain", String(digitalRead(VALORLAMPADAS1)));
}
}
void LampadaS1Interrupt(){
delayMicroseconds(100000); //evitar debounce
static int state = 0;
state = !digitalRead(VALORLAMPADAS1);
digitalWrite(LAMPADAS1, state);
detachInterrupt(INTERRUPTORLAMPADAS1);
interruptor1=1;
}
void LampadaVFunction(){
if(digitalRead(VALORLAMPADAV))
{
digitalWrite(LAMPADAV, LOW);
server.send(200, "text/plain", String(digitalRead(VALORLAMPADAV)));
}
else{
digitalWrite(LAMPADAV, HIGH);
server.send(200, "text/plain", String(digitalRead(VALORLAMPADAV)));
}
```

```
}

}

void LampadaVInterrupt(){
delayMicroseconds(100000); //evitar debounce
static int state = 0;
state = !digitalRead(VALORLAMPADAV);
digitalWrite(LAMPADAV, state);
detachInterrupt(INTERRUPTORLAMPADAV);
interruptor2=1;
}
/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
void setup() {
//Define portas como entrada ou saída
pinMode(LAMPADAS1, OUTPUT);
pinMode(INTERRUPTORLAMPADAS1, INPUT);
pinMode(VALORLAMPADAS1, INPUT);
pinMode(LAMPADAV, OUTPUT);
pinMode(INTERRUPTORLAMPADAV, INPUT);
pinMode(VALORLAMPADAV, INPUT);
irsend.begin();
wifiConnect(ssid, password); //Conecta-se a wifi
delay(1000);
serverInit(); //Inicia-se o servidor
}
void loop() {
//Interrupções Externas do ESP, acionadas com mudança de estado
attachInterrupt(INTERRUPTORLAMPADAS1, LampadaS1Interrupt , CHANGE);
attachInterrupt(INTERRUPTORLAMPADAV, LampadaVInterrupt , CHANGE);
server.handleClient(); //Deixa o servidor "ouvindo" ações
if(interruptor1==1){
//Get no serviço para informar que o interruptor foi acionado
HTTPClient http;
http.begin("http://10.0.0.100:3000/vls1?vls1=" + String(digitalRead(VALORLAMPADAS1)) );
http.GET();
interruptor1=0;
http.end(); //Close connection
}
if(interruptor2==1){
//Get no serviço para informar que o interruptor foi acionado
HTTPClient http; //Declare an object of class HTTPClient
http.begin("http://10.0.0.100:3000/vlv?vlv=" + String(digitalRead(VALORLAMPADAV)));
http.GET();
interruptor2=0;
http.end(); //Close connection
}
}
```

## **ANEXO B – Página HTML**

## **ANEXO C – Página CSS**

## **ANEXO D – Servidor NodeJS**