



**UNIVERSIDADE FEDERAL DO PARÁ**  
**INSTITUTO DE TECNOLOGIA**  
**FACULDADE DE ENGENHARIA DA COMPUTAÇÃO E TELECOMUNICAÇÕES**  
**CURSO DE ENGENHARIA DA COMPUTAÇÃO**

**AUTOMAÇÃO RESIDENCIAL UTILIZANDO CONCEITOS DE IOT FOCADO  
EM EFICIÊNCIA ENERGÉTICA**

Matheus Castelo Branco Queiroz

**BELÉM**

**2018**

Dados Internacionais de Catalogação na Publicação (CIP)  
Sistema de Bibliotecas da Universidade Federal do Pará  
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)

---

Q3a Queiroz, Matheus  
Automação Residencial Utilizando Conceitos de IoT com Foco em Eficiência Energética / Matheus  
Queiroz. - 2018.  
73 f. : il. color.

Trabalho de Conclusão de Curso (Graduação) - Faculdade de Engenharia da Computação e  
Telecomunicações, Instituto de Tecnologia, Universidade Federal do Pará, Belém, 2018.  
Orientação: Prof. Dr. Adalbery Rodrigues Castro

1. Automação Residencial. 2. Internet das Coisas. 3. Eficiência Energética. I. Rodrigues Castro,  
Adalbery, *orient.* II. Título

---

CDD 629.89

Matheus Castelo Branco Queiroz

**AUTOMAÇÃO RESIDENCIAL UTILIZANDO CONCEITOS DE IOT FOCADO EM  
EFICIÊNCIA ENERGÉTICA**

**Trabalho de Conclusão de curso apresentado ao Curso de Engenharia da Computação, Faculdade de Engenharia da Computação e Telecomunicações, Instituto de Tecnologia, Universidade Federal do Pará, como requisito parcial para a obtenção do título de Bacharel Engenheiro da Computação.**

**Orientador: Adalbery Rodrigues Castro  
Coordenador: Marcos César da Rocha Seruffo**

**BELÉM  
2018**

**MATHEUS CASTELO BRANCO QUEIROZ**

**AUTOMAÇÃO RESIDENCIAL UTILIZANDO CONCEITOS DE IOT FOCADO EM  
EFICIÊNCIA ENERGÉTICA**

Este trabalho foi julgado em \_\_\_/\_\_\_/\_\_\_\_\_ adequado para obtenção do Grau de Engenheiro da Computação, aprovado em sua forma final pela banca examinadora, que atribuiu o conceito \_\_\_\_\_.

Banca Examinadora

---

**Prof. Dr. Adalbery Rodrigues Castro – ORIENTADOR**

---

**Prof. Dr. Francisco Carlos Bentes Frey Müller – MEMBRO**

---

**Prof. Dr. Lamartine Vilar de Souza – MEMBRO**

**VISTO:**

---

**Prof. Marcos César da Rocha Seruffo – Diretor da Faculdade de Engenharia da  
Computação e Telecomunicações**

**“A maior qualidade de um vencedor**

**É nunca desistir”**

(Autor Desconhecido)

# AGRADECIMENTOS

Agradeço primeiramente à Deus por todo o amparo, sabedoria e força de vontade que me acompanharam durante toda essa caminhada acadêmica.

Agradeço também à minha família, que sempre esteve ao meu lado, comemorando minhas conquistas, me ensinando sobre o mundo e a vida, dando exemplo, dando forças, me inspirando e me apoiando em minhas decisões, até nas mais difíceis.

Agradeço também ao meu orientador, Prof. Dr. Adalbery Rodrigues Castro, que foi um excelente orientador não apenas durante o TCC, mas durante toda a minha graduação. Foi um grande exemplo de mestre, me ensinou sobre diversos quesitos da academia e sempre estava me ajudando em relação às dúvidas e problemas que surgiram durante toda essa caminhada.

Agradeço à Danielle Cruz de Sousa, minha companheira de vida que sempre esteve ao meu lado, me ajudando nas mais diversas maneiras, me dando apoio, me cobrando quando era necessário, me confortando nos momentos difíceis e me inspirando com a sua sabedoria e seu modo de viver.

Agradeço aos companheiros de curso, que sempre ajudaram uns aos outros, tornando a caminhada rumo à graduação mais leve e alegre. São pessoas que compreenderam o sentido de amizade e companheirismo no âmbito acadêmico. Tenho certeza que na faculdade surgiram amizades que levarei para o resto da vida.

Agradeço à Ingresso Rápido e ao Grupo Movable, empresas que faço parte atualmente, e empresas que entenderam a minha situação complexa em relação ao desenvolvimento e apresentação deste trabalho, que fizeram o máximo possível para que esta situação fosse resolvida da melhor forma, e que sempre tentaram me tranquilizar e me manter confortável em relação a tudo isso.

Agradeço também ao programa Ciência Sem Fronteiras, o qual fiz parte durante esta graduação e me proporcionou experiências riquíssimas, nos mais diversos campos e que com certeza mudaram a minha vida e meu modo de ver o mundo. Foram experiências que eu nunca esquecerei.

Matheus Castelo Branco Queiroz

# RESUMO

Este trabalho apresenta o desenvolvimento de um projeto que engloba os conceitos de Automação Residencial, Internet das Coisas e Eficiência Energética. Trata-se de um sistema robusto e com baixo custo de implementação. Utilizando tecnologias recentes, foi criado um sistema que permite ao usuário final o acesso remoto às tomadas de sua residência, verificando o consumo das mesmas em tempo real e se há algum equipamento ligado em cada tomada. O sistema ainda possibilita ao usuário ativar ou desativar as tomadas remotamente. O sistema possui três componentes principais: o Hardware, construído utilizando um microcontrolador ESP8266, um sensor de corrente ACS712 e um módulo relé; o software embarcado desenvolvido utilizando a Arduino IDE; e o Website, desenvolvido utilizando as tecnologias HTML, CSS e Javascript. Além disso, foi utilizado o protocolo MQTT para o envio e recebimento de dados entre essas partes. Ao longo deste trabalho são explicados os conceitos principais que foram utilizados durante o desenvolvimento deste projeto, são descritos individualmente os recursos de hardware e software utilizados, como se deu o desenvolvimento do mesmo, seus resultados e também são expostos alguns trabalhos que se pretende desenvolver baseados neste.

**PALAVRAS-CHAVE:** Automação Residencial, Internet das coisas, Eficiência Energética

# ABSTRACT

This paper presents the development of a project that encompasses the concepts of Residential Automation, Internet of Things and Energy Efficiency. It is a robust system with a low implementation cost. Using recent technologies, a system has been created that allows the end user to remotely access their home sockets, verifying their consumption in real time and if there is any equipment connected at each outlet. The system also enables the user to remotely turn the outlets on or off. The system has three main components: Hardware, built using an ESP8266 microcontroller, an ACS712 current sensor and a relay module; the embedded software developed using the Arduino IDE; and the Website, developed using HTML, CSS and Javascript technologies. In addition, the MQTT protocol was used to send and receive data between these parts. Throughout this work the main concepts that were used during the development of this project are explained, the hardware and software resources used are described individually, as well as the development of the project, its results and some projects that are intended to be developed based on this project.

**KEYWORDS:** Residential Automation, Internet of Things, Energy Efficiency

## LISTA DE FIGURAS

Figura 3.1 – ESP8266 NodeMCU 12-E.....	27
Figura 3.2 – Sensor de corrente ACS712. ....	29
Figura 3.3 – Fonte Ajustável YWRobot Breadboard Power Supply.....	30
Figura 3.4 – Funcionamento do protocolo MQTT.....	34
Figura 4.1 – Integração da Fonte Ajustável YWRobot Breadboard Power Supply ao projeto. ....	37
Figura 4.2 – Integração do microcontrolador ESP8266 ao projeto.....	38
Figura 4.3 – Integração do sensor de corrente ACS712 ao projeto. ....	39
Figura 4.4 – Integração do módulo relé ao projeto.....	40
Figura 4.5 – Hardware do projeto completo. ....	41
Figura 4.6 – Footprint do hardware. ....	41
Figura 4.7 – Esquemático do hardware.....	42

## **LISTA DE TABELAS**

Tabela 3.1 – Especificações técnicas do ESP8266. ....	28
---	----

## LISTA DE EQUAÇÕES

Equação 4.1 – Equação para obtenção do valor medido pelo sensor de corrente ...	44
Equação 4.2 – Equação para obtenção da tensão RMS do circuito .....	44
Equação 4.3 – Equação para obtenção da corrente RMS do circuito .....	44

## **LISTA DE ACRÔNIMOS**

HTML5 – *Hypertext Markup Language*

CSS3 – *Cascading Style Sheet*

IoT – *Internet of Things*

# SUMÁRIO

1	INTRODUÇÃO.....	14
1.1	Justificativa .....	14
1.2	Motivação .....	15
1.3	Objetivos.....	15
1.3.1	Objetivos gerais .....	15
1.3.2	Objetivos específicos .....	15
1.4	Estrutura do trabalho.....	16
2	FUNDAMENTAÇÃO TEÓRICA .....	17
2.1	Introdução .....	17
2.2	Internet das coisas (IoT) .....	17
2.3	Automação Residencial .....	20
2.4	Eficiência Energética .....	23
3	DISPOSITIVOS E RECURSOS UTILIZADOS .....	26
3.1	Introdução .....	26
3.2	Hardware.....	26
3.2.1	ESP8266 NodeMCU .....	26
3.2.2	Sensor de Corrente ACS712.....	28
3.2.3	Fonte Ajustável YWRobot Breadboard Power Supply .....	29
3.2.4	Outros Componentes .....	31
3.3	Software .....	31
3.3.1	Arduino IDE .....	32
3.3.2	Protocolo MQTT .....	33
3.3.3	Website .....	34
4	DESENVOLVIMENTO DO PROJETO.....	36
4.1	Integração do Hardware.....	36

4.1.1	Confecção do sistema de energia (extensão).....	36
4.1.2	Integração da Fonte Ajustável .....	36
4.1.3	Integração do ESP8266 .....	37
4.1.4	Integração do Sensor de Corrente.....	38
4.1.5	Integração do módulo relé .....	39
4.1.6	Hardware completo.....	40
4.2	Software Embarcado .....	42
4.2.1	Conexão WiFi.....	42
4.2.2	Leitura do Sensor de Corrente ACS712 .....	43
4.2.3	Comunicação com o módulo relé .....	44
4.2.4	Envio e recebimento de dados utilizando o protocolo MQTT .....	45
4.2.5	Códigos Complementares.....	46
4.3	Website .....	47
4.3.1	Conexão com o servidor MQTT.....	47
4.3.2	Leitura, interpretação e envio de Dados .....	48
4.3.3	HTML5 e CSS3 .....	48
5	RESULTADOS .....	50
6	CONCLUSÃO.....	53
6.1	Trabalhos Futuros .....	54
	REFERÊNCIAS .....	56
	ANEXO I – CÓDIGO FONTE DO SOFTWARE EMBARCADO .....	60
	ANEXO II – CÓDIGO FONTE DA PÁGINA WEB .....	67
	ANEXO III – SCRIPT TMO.JS .....	71

## 1 INTRODUÇÃO

### 1.1 Justificativa

O desperdício de energia elétrica há muito tempo vem sendo um grande inconveniente. Por um lado, é inconveniente ao produtor de energia, que consome cada vez mais recursos para produzir uma energia que não vai ser consumida da maneira correta. Da mesma forma é inconveniente para o consumidor final, que acaba tendo diversos prejuízos devido à energia que desperdiçou.

Os dados sobre o desperdício de energia elétrica no Brasil são alarmantes. Segundo a Associação Brasileira das Empresas de Serviços de Conservação de Energia (ABESCO, 2017) o desperdício de energia elétrica nos últimos três anos custou R\$ 61,7 bi ao consumidor brasileiro, o que é um grande prejuízo a todos os consumidores. Segundo o Programa Nacional de Conservação de Energia Elétrica (PROCEL, 2017) O total de energia elétrica desperdiçada no país chega a 40 milhões de KW, sendo os consumidores responsáveis por 22 milhões destes mesmos.

Com o intuito de combater este desperdício o campo de estudo da eficiência energética está se desenvolvendo cada vez mais. Este campo visa propor soluções em diversas áreas de forma a promover um uso mais eficiente da energia elétrica, diminuindo o desperdício da mesma e fazendo mais tarefas sem a utilização de mais recursos.

Eficiência energética também é parte de diversos projetos de automação residencial, um conceito que visa trazer maior conforto, comodidade e praticidade ao cotidiano das pessoas. Este conceito pode se manifestar de várias formas, incluindo dispositivos inteligentes que atuam e se comunicam de forma a tornar a interação com o usuário final mais natural e o dia do mesmo mais agradável.

Para a comunicação entre estes dispositivos surgiu o conceito de Internet das coisas (IoT). A Internet das coisas permite que diversos dispositivos se comuniquem e sejam controlados através da rede. Isso torna muito mais fácil

desenvolver soluções inovadoras e capazes de proporcionar um maior conforto e facilidade ao usuário final.

O projeto que é objeto deste trabalho une esses três conceitos. Este projeto é um projeto de automação residencial que utiliza Internet das coisas visando proporcionar uma melhoria à vida do usuário final através da aplicação de conceitos de Eficiência energética em tarefas do seu cotidiano.

## **1.2 Motivação**

A primeira razão pela escolha deste tema para este projeto foi a sua capacidade real de impactar a vida de diversas pessoas. Caso este projeto seja popularizado milhares de pessoas irão perceber diferenças nas suas vidas, pois este projeto é capaz de fornecer uma maior comodidade e praticidade aos seus usuários, além de diminuir os prejuízos e desperdícios.

A segunda razão pela escolha deste tema para este projeto foi seu pioneirismo. Após muita pesquisa, não foram encontrados projetos similares a este, e nem produtos como este à venda no mercado nacional (foram encontrados similares apenas no mercado internacional). Este projeto possui um grande potencial de inovação, e vários outros projetos podem ser desenvolvidos a partir deste. Alguns destes projetos serão exemplificados na sessão de trabalhos futuros.

## **1.3 Objetivos**

### **1.3.1 Objetivos gerais**

Elaborar um sistema de tomada inteligente, onde a mesma fornecerá diversas informações (corrente sendo transmitida pelo sistema, se existe algo conectado à mesma e estado da tomada) ao usuário final através de um website. Neste mesmo website o usuário também será capaz de ativar ou desativar a tomada remotamente.

### **1.3.2 Objetivos específicos**

- Criar um sistema de tomada;
- Acoplar um medidor de corrente ao sistema de tomada;

- Acoplar um módulo relé ao sistema de tomada;
- Acoplar todo o sistema de tomada ao microcontrolador ESP8266;
- Conectar o microcontrolador ESP8266 à internet via wifi;
- Criar um website para exibição de dados referentes ao sistema de tomada e para possibilitar o controle remoto do mesmo;
- Utilizar o protocolo MQTT para possibilitar que o microcontrolador ESP8266 envie as informações referentes ao sistema de tomada para o website e receba comandos do mesmo.

#### **1.4 Estrutura do trabalho**

Além do capítulo introdutório, este trabalho está organizado da seguinte maneira:

O capítulo 2 irá abordar e exemplificar os principais conceitos que serão utilizados no decorrer deste trabalho.

O capítulo 3 irá apresentar os dispositivos e recursos utilizados neste projeto, dando as especificações técnicas dos mesmos, explicando suas formas de operação e mostrando as razões pela escolha dos mesmos.

O capítulo 4 aborda o desenvolvimento do projeto, mostrando de maneira mais profunda a montagem do hardware, o desenvolvimento do software embarcado e o desenvolvimento do website.

O capítulo 5 apresenta os resultados, análises e conclusões obtidos após o desenvolvimento deste projeto.

O capítulo 6 apresenta as considerações finais deste trabalho e as ideias que serão desenvolvidas em projetos futuros.

## **2 FUNDAMENTAÇÃO TEÓRICA**

### **2.1 Introdução**

Este projeto aborda três temas de grande relevância e discussão nos dias de hoje: automação residencial, Internet das coisas(IoT) e eficiência energética. Estes temas foram de grande importância na formação deste projeto, pois são temas bastante contemporâneos, inovadores e temas que estão sendo bastante estudados nos dias de hoje.

Este capítulo será dedicado a apresentar cada um desses temas, dar uma breve explicação sobre os mesmos, mostrar qual a importância deles nos dias de hoje, exemplifica-los e expor qual a relação de cada tema com o projeto.

### **2.2 Internet das coisas (IoT)**

A União Internacional de Telecomunicações define a Internet das coisas da seguinte forma: “A Internet das coisas (IoT) é a rede de dispositivos físicos, veículos, eletrodomésticos e outros itens que possuem eletrônicos, software, sensores, atuadores e conectividade de rede que permite que esses objetos se conectem e troquem dados” (ITU, 2015). Ou seja, de modo mais simples, é uma rede que não é populada por seres humanos, mas sim objetos de diversos tipos.

A internet das coisas possibilita que diversos dispositivos se comuniquem e sejam controlados através da rede, proporcionando maior integração do mundo real com o mundo virtual, resultando em maior eficiência, precisão, economia e menor intervenção humana em processos. Pesquisadores do assunto afirmam que em 2020 haverá em torno de 30 bilhões de dispositivos conectados à rede (NORDRUM, 2016).

A Internet das coisas já é aplicada em diversas áreas atualmente. Um bom exemplo de utilização de internet das coisas é a forma como a mídia a utiliza. Utilizando os dados registrados por diversos aparelhos, um veículo de mídia é capaz de determinar qual a melhor propaganda ou conteúdo que deve ser apresentado para cada indivíduo. E além disso ainda são feitas análises sobre a reação de cada indivíduo em relação à determinada propaganda, de modo a aperfeiçoar mais ainda o seu sistema.

Outra área em que a internet das coisas atua extensivamente é a área de veículos autônomos e veículos inteligentes. Estes veículos se conectam à rede para a troca de informações das mais variadas possíveis. Informações de rota, de condição das estradas, do clima, das condições do veículo e alguns mais avançados são capazes até de sintonizar rádios ou músicas de acordo com o gosto do usuário que está dentro do veículo (WAYMO, 2017).

E a utilização da internet das coisas em veículos autônomos vai ainda além. Empresas de transporte como a Uber, Lyft, Didi-Chuxing e a montadora de carros Tesla estão desenvolvendo formas para que o motorista não seja mais necessário no processo de carona (JOHNSON, 2018). Ou seja, o carro fará todo o processo de busca, transporte e desembarque dos passageiros sem a necessidade de um motorista. Uma das vantagens dessa tecnologia é que o dono do carro poderá estar no trabalho, em casa ou passeando enquanto o carro “trabalha por ele”.

O conceito de veículos autônomos e veículos inteligentes também não se restringe somente à carros. Já existem caminhões que utilizam a rede para reportar suas localizações, condições da carga que estão levando, clima, condições da estrada, entre outros(UBER, 2016). Existem também máquinas agrícolas capazes de mapear toda a plantação, traçar a melhor rota para fazer a colheita e fazer a colheita inteira sem a necessidade de motorista(FRANCO, 2016). Há também navios que reportam suas localizações, condições de carga e verificam as condições marítimas para determinar qual a melhor rota a ser seguida(KONGSBERG, 2017). O conceito de veículos autônomos e inteligentes que utilizam internet das coisas pode ser expandido para os mais diversos tipos de veículos existentes.

A internet das coisas também é bastante utilizada no gerenciamento de infraestrutura e tráfego. Nesse quesito a mesma é utilizada para controle de trilhos de trem, controle de pontes, monitoramento de fazendas de energia eólica e solar, controle de tráfego de vários tipos de veículo, monitoramento de frotas, sistemas automatizados de pedágio, etc. Nesse quesito a Internet das coisas também pode ser utilizada para verificar qual o melhor horário para o

agendamento de serviços de manutenção de redes públicas (energia, água, saneamento, entre outros).

O monitoramento ambiental também faz bastante uso da internet das coisas. Aplicações como monitoramento de florestas e queimadas, monitoramento das condições do solo, do ar e da água, monitoramento de animais selvagens e seus habitats e sistemas de monitoramento e alerta de terremotos, tsunamis e outros desastres naturais fazem grande uso de sensores e dispositivos para coletar dados e enviar os mesmos para a internet.

Outro campo onde a internet das coisas é bastante utilizada é o campo da medicina e bem-estar. No campo da medicina, hoje em dia já existem diversos aparelhos para monitoração à distância de idosos. Estes aparelhos são capazes de monitorar pressão sanguínea, batimentos cardíacos, insulina e alguns são capazes até de informar caso o usuário esteja em alguma situação de emergência (tenha caído, desmaiado, entre outros...) (APPLE, 2015). Existem também aparelhos de audição, marca passos e óculos inteligentes capazes de fazer análises de dados com o intuito de proporcionar uma melhor qualidade de vida aos seus usuários.

No campo do bem-estar existem diversos aparelhos que fazem monitoramento de vários quesitos como quantidade de passos por dia, calorias queimadas, distância caminhada por dia, tempo de exercício, qualidade do sono, batimento cardíaco, pressão sanguínea, alimentação, peso, altura, quantidade de gordura, e outros. Utilizando-se destes dados os aparelhos são capazes de traçar um plano para o usuário com o intuito de melhorar sua qualidade de vida. Ou seja, se o usuário estiver se alimentando mal, o aparelho o sugere uma dieta mais equilibrada; se o usuário estiver se exercitando pouco, o aparelho o sugere uma rotina de exercício; se o usuário não tem dormido o suficiente, o aparelho lhe dá dicas para que ele tenha uma melhor noite de sono, entre outros (FITBIT, 2015). Este tipo de aparelho já é bastante popular, pois proporcionam uma melhora real na qualidade de vida e é possível encontrar modelos destes dispositivos por preços razoáveis.

E outro campo onde Internet das coisas é bastante utilizado é o campo de automação residencial, que será o campo que o projeto deste trabalho fará parte.

A Internet das coisas é essencial para o campo de automação residencial, pois diversos dispositivos residenciais autômatos e inteligentes utilizam a mesma como meio de comunicação. Hoje já existem diversos dispositivos residenciais que utilizam a internet das coisas para a transferência de dados, comunicação com o usuário final e tomada de decisões. Alguns exemplos destes dispositivos são: lâmpadas, geladeiras, termostatos, aspiradores de pó e alimentadores automáticos de animais. O campo de automação residencial será abordado com mais detalhes no tópico seguinte a este.

### **2.3 Automação Residencial**

Segundo TEZA (2002) “A automação residencial se fundamenta na integração dos sistemas constituintes de uma residência, como por exemplo, seus eletrodomésticos, equipamentos de áudio, vídeo e informática, persianas, portões e portas automáticas, dentre outros, que são os sistemas autônomos, presentes hoje em dia nas residências. Dessa forma, o termo automação residencial também denominada automação doméstica ou domótica, representa a utilização de processos automatizados em casas, apartamentos e escritórios”.

A automação residencial ainda está em um estágio muito inicial, e ainda é tida como novidade por muitos, mesmo no mundo tecnológico. Muitas pessoas ainda ficam com um certo receio, pois o conceito insere muitas tecnologias novas ao cotidiano das pessoas. Porém, muitos afirmam que essa tecnologia em alguns anos já estará bem inserida no cotidiano das pessoas, como diversas tecnologias que antes eram vistas como futuristas e hoje já são vistas como algo natural pelas pessoas.

Uma das grandes barreiras encontradas e que ainda impede a popularização da automação residencial é a desconfiança e rejeição por parte do usuário final. Muitas pessoas acreditam que trazer mais tecnologia para dentro de suas residências irá trazer mais malefícios do que benefícios. As pessoas tem medo de que equipamentos mal projetados acabem fazendo mal a alguém, ou de que os usuários podem estar sendo vigiados dentro de suas próprias casas por hackers ou empresas, ou simplesmente acreditam que os aparelhos vão mais atrapalhar que ajudar. Porém, acredita-se que com os

avanços da tecnologia essa desconfiança e rejeição podem diminuir e a automação residencial se popularize cada vez mais.

Uma segunda barreira enfrentada pela automação residencial é o custo. Por ser uma tecnologia muito recente, os preços dos produtos que compõem essa tecnologia ainda são bastante elevados. Vários produtos “inteligentes” chegam a custar mais que o dobro das suas versões comuns, o que acaba tornando eles menos atrativos aos clientes. Neste quesito, o projeto que é desenvolvido neste trabalho pode ser considerado um avanço, pois seu custo é bastante inferior ao custo de produtos “inteligentes” similares encontrados no mercado, apesar de ainda ser muito superior ao de uma versão comum do mesmo (uma versão “não inteligente”).

Segundo MURATORI; DAL BÓ (2012) os sistemas de automação residencial podem ser classificados de 3 formas: autônomos, integrados e complexos. A definição de cada sistema segundo os autores é a que segue:

- Autônomos: Não há interação entre os dispositivos e estes são projetados apenas para o liga/desliga de um subsistema ou um dispositivo específico de acordo com um ajuste predefinido. Cada dispositivo é tratado de forma independente sem que tenham relação um com o outro;
- Integrados: Há integração entre os subsistemas por meio de uma central de automação que controla os dispositivos. O processamento pode ser centralizado ou distribuído pela rede;
- Complexos: A integração do sistema é feita através de softwares e o sistema como um todo deixa de ser apenas um controlador remoto e passa a ser um gerenciador dos dispositivos e grupos de dispositivos.

As aplicações de automação residencial são as mais diversas possíveis. Automação residencial pode ser aplicada, por exemplo, em sistemas de segurança. Com a insegurança cada vez mais crescente, já são encontradas fechaduras automatizadas, câmeras de segurança e sistemas de proteção residencial cada vez mais integrados e com a possibilidade de acesso e controle remotos, de modo a fazer com que o usuário fique mais tranquilo e tenha uma maior sensação de segurança em relação à sua residência.

Outro exemplo de aplicação, e que já existe e é utilizado atualmente, são os sistemas de climatização automáticos. Estes sistemas são capazes de verificar a temperatura externa do local e fornecer um ambiente mais agradável ao usuário, aquecendo ou resfriando o ambiente. Alguns sistemas mais avançados são capazes de serem controlados remotamente e alguns ainda possuem ferramentas para aprender o comportamento e horário do usuário, de forma a apenas serem ligados perto dos horários em que serão necessários, poupando energia(NEST, 2017).

Os sistemas de entretenimento também estão cada vez mais interligados com a automação residencial. Estes são capazes de aprender com as preferências de cada usuário e sugerir conteúdos personalizados para cada usuário. Estes sistemas também estão sendo otimizados de forma a serem ativados de qualquer lugar da residência, e existem inclusive dispositivos de áudio capazes de serem instalados em diversos cômodos da casa e reproduzirem o mesmo áudio simultaneamente (GOOGLE, 2016).

Existem também diversos sistemas de cuidado e bem-estar. Existem, por exemplo, geladeiras que verificam o que está em falta, alertam o usuário final e dão a opção de comprar aqueles produtos(SAMSUNG, 2015). Também já existem sistemas para hortas domésticas, que são capazes de verificar as condições do solo, o nascimento de ervas daninhas, a quantidade de água nas plantas e fazer a irrigação das plantas caso seja necessário(FARMBOT, 2016).

Para quem possui animais de estimação existem sistemas de alimentação automática para os mesmos e até sistemas de comunicação otimizados para animais, onde os usuários serão capazes de ver e se comunicar com seus animais de estimação. Já existe, por exemplo, uma câmera com um botão onde o animal aperta o botão e o usuário recebe uma notificação em seu smartphone para comparecer à uma videoconferência com seu animal(PETCHATZ, 2017).

Para a limpeza da residência existem também os robôs de limpeza, que estão cada vez mais populares. Estes robôs são programáveis e autônomos. Neles, o usuário é capaz de configurar dias e horários para a limpeza, e quando é chegada a hora o robô se liga automaticamente, começa a limpeza da

residência e ao terminar, retorna para sua base. Tudo de forma autônoma e sem nenhuma necessidade de interação humana(IROBOT, 2015).

Neste quesito o projeto que é objeto deste trabalho se encaixa nas aplicações de controle à distância e também na área de eficiência energética, pois com esta possibilidade de controlar as tomadas à distância, vem também o benefício da diminuição do desperdício de energia. Este benefício se dá pelo fato de que objetos que por alguma razão foram esquecidos ligados poderão ser desligados remotamente. O conceito de eficiência energética e alguns exemplos de aplicações que envolvem o mesmo serão dados no tópico que segue.

#### **2.4 Eficiência Energética**

Segundo a Associação Brasileira das Empresas de Serviços de Conservação de Energia (ABESCO, 2015) “Eficiência energética é uma atividade que busca melhorar o uso das fontes de energia. A utilização racional de energia chamado também simplesmente de eficiência energética, consiste em usar de modo eficiente a energia para se obter um determinado resultado. Por definição, a eficiência energética consiste da relação entre quantidade de energia empregada em uma atividade e aquela disponibilizada para sua realização”.

O termo Eficiência Energética pode ser resumido simplesmente em “Fazer mais com menos energia”. Como foi explicado no capítulo 1.1 deste trabalho, o desperdício de energia no Brasil e no mundo são grandes, e é para diminuir estes desperdícios que existem os estudos no campo de eficiência energética.

Em todo o país existem diversos centros de pesquisa e desenvolvimento dedicados a este tema. Dentro da Universidade Federal do Pará, por exemplo, temos o Centro de Excelência em Eficiência Energética da Amazônia (CEAMAZON). Este centro, além de conduzir diversos estudos sobre o assunto, também é responsável por ministrar cursos para instruir profissionais sobre o assunto (FRAZÃO, 2017), e realiza ações para conscientizar famílias de baixa renda sobre o assunto e ajuda-las a diminuir o desperdício de energia e os prejuízos financeiros que esse desperdício gera (FRAZÃO, 2016).

Existem diversos exemplos da eficiência energética aplicada ao cotidiano. Um bom exemplo disso são as lâmpadas do tipo LED, que são capazes de gerar a mesma quantidade de luz gerada por uma lâmpada incandescente, porém gastam em torno de 90% a menos de energia, possuem uma vida útil até 50 vezes maior e produzem menos calor, o que acaba gerando economia de energia nos sistemas de climatização, que consumirão menos energia para resfriar os ambientes.

Desenvolvido na Suécia, o sistema OrbSys (Orbital Systems, 2017) também é um excelente exemplo de eficiência energética. Este sistema se trata de um chuveiro que é capaz de captar a água já utilizada naquele banho, filtrá-la e reutilizá-la neste mesmo banho. Segundo a empresa responsável pelo projeto, este chuveiro é capaz de reutilizar 90% da água já utilizada. Sua eficiência energética se dá ao fato de que a água que é reutilizada não precisa ser aquecida novamente, o que segundo a empresa gera uma economia de até 80% na energia utilizada em cada banho.

Mas a eficiência energética não é proporcionada apenas por equipamentos de última geração. O conceito de eficiência energética também abrange a melhor utilização de recursos naturais. Por exemplo, é bastante estimulada a melhor utilização da luz natural. Substituir a luz elétrica pela luz natural nos horários possíveis é uma boa forma de se reduzir o consumo desnecessário de energia elétrica.

Outra área que a eficiência energética abrange é a de melhorias e manutenções. Esta área pode se manifestar de diversas formas, como em uma melhoria no projeto elétrico de um local, uma manutenção em equipamentos que podem estar consumindo mais energia que o necessário ou uma substituição de equipamentos ultrapassados por equipamentos mais novos e de menor consumo. Neste quesito existem várias atitudes que podem ser tomadas de forma a diminuir o desperdício de energia.

Eficiência energética também pode ser aplicadas em empresas e indústrias. Alguns exemplos são o isolamento térmico de equipamentos quentes, a eliminação de fugas de energia, o reaproveitamento de combustíveis e compostos residuais, a implementação dos sistemas de cogeração e a troca de

motores convencionais por motores de alto rendimento (esses capazes de gerar economias de 20% a 30% de energia).

O envolvimento do projeto que é objeto deste trabalho no conceito de eficiência energética se dá utilizando o conceito em si. O conceito diz que eficiência energética é, resumidamente, “Fazer mais com menos energia”. E o projeto se encaixa neste conceito devido ao fato que os usuários deste projeto estarão consumindo uma quantidade de energia igual ou inferior à que já é utilizada em seus cotidianos, porém estarão fazendo um uso mais consciente de energia, evitando o desperdício da mesma.

### **3 DISPOSITIVOS E RECURSOS UTILIZADOS**

#### **3.1 Introdução**

Após ter feito todo o planejamento do que estaria incluso no projeto e como ele deveria operar, foi pesquisado qual seria o material (hardware e software) necessário para a execução do mesmo. Foram verificadas diversas formas de implementar o projeto, com diversos equipamentos e diversas abordagens. Após muito analisar, foram escolhidos os equipamentos que segue.

Neste capítulo serão apresentadas cada uma das ferramentas utilizadas, suas especificações técnicas, seu funcionamento e a razão delas terem sido escolhidas. Serão abordadas tanto as ferramentas de software como as de hardware.

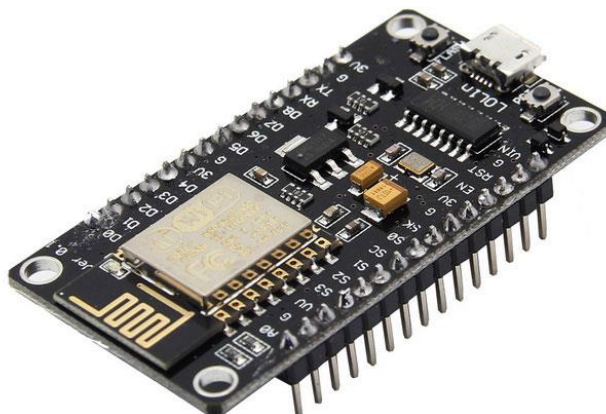
#### **3.2 Hardware**

Para um projeto de automação residencial como este, se faz necessária a utilização de equipamentos de hardware, pois será necessário fazer medições de corrente, interferências na rede de energia (permitir ou cortar corrente), recebimento de comandos e envio de informações via rede Wi-Fi.

Após diversas análises foi visto que os equipamentos necessários para executar essas funções seriam: sensor de corrente (medições de corrente), relé (interferências na rede de energia), e um micro controlador com chip Wi-Fi (recebimento de comandos e envio de informações via rede Wi-Fi). Feito isso foram analisados diversos equipamentos que poderiam ser utilizados nesse projeto. Os equipamentos que foram escolhidos, por melhor se adequarem ao projeto, serão apresentados a seguir.

##### **3.2.1 ESP8266 NodeMCU**

O microcontrolador escolhido para fazer parte desse projeto foi o ESP8266. Produzido pela Espressif Systems, o ESP8266 é tido hoje como um dos chips que irá impulsionar a evolução da Internet das coisas. O ESP8266 possui chip Wi-fi integrado capaz de fazer conexões TCP/IP simples, conexão micro-usb, consegue operar em 3 modos diferentes (STA/AP/STA+AP), 11 portas GPIO, um ADC, entre outros (especificações técnicas na tabela 3.1).

**Figura 3.1 – ESP8266 NodeMCU 12-E.**

**Fonte: Deal Extreme, 2017**

O ESP8266 foi escolhido por diversas razões. A primeira delas é que ele não é um shield (ou seja, um complemento que pode ser encaixado em outros microcontroladores) e sim um MCU (Micro Controller Unity) conseguindo assim atuar sozinho, sem a necessidade de complementos para exercer suas funções, tendo em vista que ele também já possui um chip Wi-Fi Integrado. Isso faz muita diferença no projeto, pois elimina custos e diminui a quantidade de equipamentos necessários para sua implementação.

Outra razão para a escolha do mesmo foi a diversidade de linguagens em que ele pode ser programado. O ESP8266 pode ser programado em mais de 10 linguagens diferentes, entre elas linguagens populares como Arduino, Lua e Python. Essa diversidade o põe em posição de destaque em relação a outros microcontroladores, pois o fato de haver mais opções acaba tornando o trabalho mais fácil de ser executado, e também proporciona a possibilidade de programar tarefas mais complexas para serem executadas no mesmo.

Uma terceira razão para a escolha do mesmo é que devido ao fato de ser bastante utilizado em diversos projetos, o ESP8266 já possui uma vasta

biblioteca de manuais, projetos e fóruns, seja na internet ou até em livros físicos do mesmo. Isso ajuda bastante, pois um vasto material de qualidade como referência proporciona diversas possibilidades para executar o projeto, e também ajuda a solucionar problemas que possam vir a ocorrer durante a execução do mesmo.

**Tabela 3.1 – Especificações técnicas do ESP8266.**

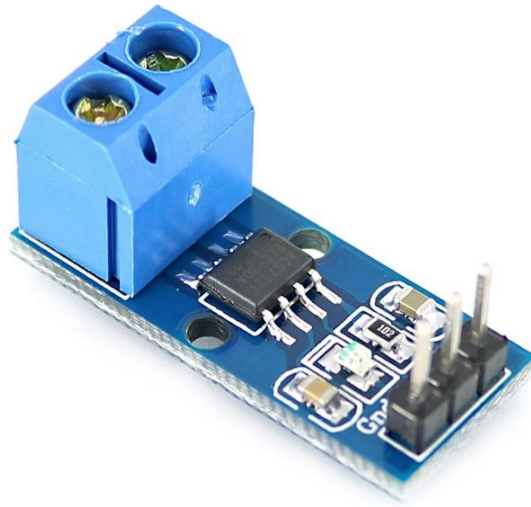
ESP8266	
Wireless Integrado	Sim, padrão 802.11 b/g/n
Conector	Micro-usb
Modos de operação	STA/AP/STA+AP
Portas GPIO	11
Tensão de operação	3 ~ 3,6V (recomendado 3.3V)
Taxa de transferência	110-460800bps
Conversor Analógico Digital	Sim
Protocolos de rede	IPv4, TCP, UDP, HTTP, FTP

**Fonte: Espressif Systems, 2017**

### 3.2.2 Sensor de Corrente ACS712

Para este projeto foi escolhido o sensor de corrente ACS712 produzido pela empresa Allegro MicroSystems. A versão que seria utilizada inicialmente no projeto seria a de -5 A a +5 A, porém houve dificuldade na hora de achar essa versão à venda. Sendo assim, foi utilizada a versão de -30 A a + 30 A, que foi a versão encontrada com mais facilidade para a compra.

Pensou-se também em utilizar outro modelo de sensor de corrente, que fosse não-invasivo. Porém essa ideia foi descartada por três razões. A primeira razão é que já iria ser feita uma invasão no sistema devido à utilização do relé, então não havia razão de comprar um sensor não invasivo. A segunda razão é que os sensores invasivos geralmente são mais precisos que os não invasivos. A terceira razão é o custo, pois um sensor não invasivo chega a custar até quatro vezes mais que um sensor invasivo.

**Figura 3.2 – Sensor de corrente ACS712.**

**Fonte: Filipe Flop, 2017**

Este sensor de corrente possui uma boa precisão, com taxa de erro de apenas 1,5 %, taxa considerada razoável para o projeto. Sua medição é feita utilizando o efeito Hall para detectar o campo magnético gerado pela passagem de corrente. Sua sensibilidade de saída pode variar de 66 mV/A (sensor de 30 A) à 185 mV/A (sensor de 5 A). Possui 3 pinos (VCC, GND e Out) para efetuar a medição e possui largura de banda de 80 kHz. A entrada que é utilizada para a medição da passagem de corrente suporta tensões de 127 V, 220 V e 380 V. Sua tensão de operação é de 5V, e ele é capaz de operar tanto com correntes AC como com correntes DC.

### **3.2.3 Fonte Ajustável YWRobot Breadboard Power Supply**

No projeto inicial não se previa a utilização deste equipamento. Porém, durante a execução do mesmo foi verificado que utilizando a alimentação USB havia muita variação na alimentação do circuito, o que acabava ocasionando erros nas medições de corrente. Logo, percebeu-se a necessidade de algo que pudesse fornecer uma corrente de alimentação mais estável.

Diversas razões levaram à escolha deste equipamento. A primeira delas foi a disponibilidade, tendo em vista que é muito fácil encontrar esse produto à venda e com um custo bastante acessível (Durante as pesquisas foi possível achar este equipamento por R\$ 10,90).

**Figura 3.3 – Fonte Ajustável YWRobot Breadboard Power Supply.**



**Fonte: Amazon Company, 2017.**

O segundo fator foi a capacidade que este componente tem de fornecer as tensões necessárias para todos os equipamentos deste experimento, pois ele fornece tensões de 3,3 V (necessária para o ESP8266) e 5 V (necessária para o relé e o sensor de corrente). Sendo assim, temos todos os equipamentos do experimento operando com as correntes de alimentação corretas e com nível de variação na alimentação baixo.

Este equipamento possui botão de ligar/desligar, LED indicador de corrente e um conector USB fêmea multi-propósito. Sua tensão de alimentação pode variar de 6,5 V a 12 V(DC). Este equipamento é capaz de fornecer ao mesmo tempo tensões de 3,3 V e 5 V, sendo cada uma em um dos conjuntos de pinos de saída nas extremidades do equipamento. O equipamento ainda fornece a possibilidade de escolher qual será a tensão de saída em cada um dos

conjuntos de pinos, bastando mover cada jumper seletor que há no equipamento para a tensão desejada.

### **3.2.4 Outros Componentes**

Para a execução desse projeto também foram usados diversos equipamentos que são de uso comum em projetos do tipo. São eles:

- Módulo relé de 5 V e 1 canal, utilizado para cortar a corrente do sistema quando necessário;
- Capacitores de 100  $\mu$ F, para evitar interferências e perdas de corrente. Foram utilizados na saída da fonte ajustável;
- Resistores de alta precisão, para evitar que equipamentos queimassem devido à alta tensão que pudesse ser fornecida;
- Jumpers (cabos) para conectar todos os equipamentos.
- Fonte comum de 12 V e 1.25 A, que será utilizada para fornecer energia à fonte ajustável
- Plugs de tomada (Macho e Fêmea) e cabo de transmissão de energia que serão utilizados para confeccionar uma extensão que atuará como o sistema de fornecimento de energia para os equipamentos que serão utilizados como carga no projeto

### **3.3 Software**

Para um projeto desta complexidade se faz necessário o uso de diversos softwares e novas tecnologias, pois é necessário que um sistema robusto seja executado de maneira eficaz e rápida. Para isto, foi estudada qual seria a melhor maneira de programar um software para o microcontrolador, qual protocolo utilizar para estabelecer uma conexão do mesmo com a internet e que tecnologias utilizar para apresentar os dados ao cliente de forma elegante, simples de entender e que possa ser acessada em qualquer dispositivo.

Para a execução destas tarefas foram escolhidas as seguintes ferramentas: Arduino IDE (Programação do microcontrolador) protocolo MQTT (conexão do microcontrolador com a internet) e um website (para exibir os dados

para o cliente). A seguir serão detalhadas cada uma dessas ferramentas, explicando seus funcionamentos e a razão de terem sido escolhidas.

### **3.3.1 Arduino IDE**

Para programar o microcontrolador foi escolhido o Ambiente de Desenvolvimento Integrado Arduino (Arduino IDE). Vários motivos levaram à essa escolha. O primeiro deles foi a familiaridade com a ferramenta, pois durante a formação acadêmica houveram diversas disciplinas e projetos onde uma das ferramentas utilizadas foi a IDE Arduino. Isso facilita bastante a execução do projeto, pois já sabendo como utilizar a ferramenta elimina-se um gasto de tempo que seria necessário para poder entender o funcionamento caso fosse utilizada uma ferramenta desconhecida.

Outra razão foi a enorme quantidade de aparelhos compatíveis com essa linguagem de programação. Devido ao seu tempo de mercado e à sua popularidade, a Arduino IDE tem bibliotecas que a fazem capaz de se comunicar com boa parte dos equipamentos de automação existentes no mercado. Isso proporciona ao projeto uma maior flexibilidade, uma maior gama de opções e uma maior certeza de que o projeto funcionará, pois grande parte das bibliotecas já foram testadas e aprovadas por sua comunidade.

Comunidade também que faz parte da terceira razão pela qual esta ferramenta foi escolhida. Devido à sua popularidade, a Arduino IDE possui uma vasta quantidade de materiais para estudo sobre a mesma. São blogs, artigos, livros, monografias, entre outros. Além da própria comunidade que é bastante ativa. Existem muitos fóruns sobre as mais diversas formas de aplicação da Arduino IDE e seus componentes. Isso dá um grande suporte ao projeto, proporcionando melhorias ao mesmo e soluções para problemas que possam vir a ocorrer.

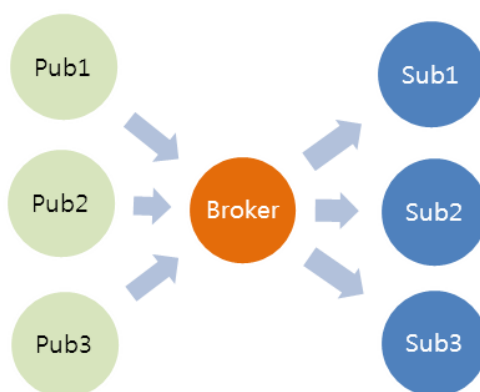
Houveram várias outras razões para a escolha desta ferramenta, como linguagem de programação simples de ser entendida, compilação rápida, programa gerado leve. Porém, as principais razões foram as citadas anteriormente, e elas foram as que mais pesaram para a escolha desta ferramenta.

### 3.3.2 Protocolo MQTT

O protocolo MQTT é hoje um dos principais protocolos utilizados quando se trata de Internet das coisas. De acordo com seus criadores “É um protocolo de mensagens extremamente simples e leve, projetado para dispositivos restritos e redes de baixa largura de banda, de alta latência ou não confiáveis. Os princípios de design são para minimizar a largura de banda da rede e os requisitos de recursos do dispositivo, enquanto também tentam garantir a confiabilidade e algum grau de garantia de entrega. Esses princípios também tornam o protocolo ideal do mundo emergente "máquina-a-máquina" (M2M) ou "Internet das coisas" de dispositivos conectados e para aplicações móveis onde a largura de banda e a energia da bateria são excelentes” (MQTT, 2017).

Para que seja possível a comunicação utilizando o protocolo MQTT são necessárias cinco ferramentas: uma chave (token) única que será utilizada para identificar a informação; um payload, que é a informação que será enviada ou recebida; um (ou mais de um) publisher, que é o dispositivo que é o responsável por disponibilizar as informações; um (ou mais de um) subscriber, que é o dispositivo que receberá as informações; e um broker, que é o servidor MQTT responsável por receber, armazenar e repassar as informações necessárias.

A comunicação é feita da seguinte forma: o publisher envia o payload e o seu respectivo token para o broker, que o armazena. O subscriber se inscreve em um determinado token e recebe do broker toda a informação (payloads) que é enviada para aquele token. Sendo assim os projetos que utilizam o protocolo MQTT necessitam de apenas um servidor, sendo todos os outros componentes clientes MQTT. A figura 3.4 é uma ilustração que demonstra o funcionamento do protocolo MQTT.

**Figura 3.4 – Funcionamento do protocolo MQTT.**

Fonte: BERTOLETI, 2016.

### 3.3.3 Website

Para construção do website foram utilizadas três das principais tecnologias de desenvolvimento web da atualidade: HTML, CSS e Javascript. Todas elas foram escolhidas pelas mesmas razões: suprem satisfatoriamente as necessidades do projeto, são bastante familiares (pois já foram utilizadas em outros projetos no decorrer da vida acadêmica) e são simples de serem escritas e lidas.

O HTML foi utilizado para estruturar o site. Utilizando essa ferramenta foi definido onde cada elemento se posicionaria na página, suas identificações e como agiriam. Essa ferramenta também foi utilizada para tornar a página responsiva, ou seja, capaz de se adaptar a qualquer tela que visitasse o website, tornando assim o site capaz de ser visitado sem problemas utilizando qualquer tipo de dispositivo.

O CSS foi utilizado no design do site. Esta ferramenta foi utilizada para tornar o website mais visualmente agradável ao cliente, além de mais fácil de ser entendido. Foram colocadas diferentes cores e estilos em diversas partes do site com o intuito de tornar a experiência do usuário mais agradável.

O Javascript é a tecnologia responsável pelos dados nesse projeto. O mesmo foi utilizado para conectar ao broker MQTT, receber e enviar dados, interpretá-los e exibí-los corretamente no site. Foram utilizados três scripts diferentes para a execução dessas tarefas, sendo dois que são padrões,

necessários para a conexão com o servidor MQTT e um para recepção/envio de dados, interpretação e exibição dos mesmos no site.

## **4 DESENVOLVIMENTO DO PROJETO**

### **4.1 Integração do Hardware**

De posse das ferramentas adequadas, pode-se iniciar o processo de montagem da parte física do projeto. Durante a montagem deste projeto cada equipamento foi montado e testado individualmente. Apenas após todas as ferramentas terem sido testadas e aprovadas é que foi desenvolvido o código-fonte responsável por integrar todas elas.

Neste capítulo será abordada a integração de cada ferramenta ao projeto, especificando onde e como são feitas as conexões de cada porta de entrada e saída de cada ferramenta utilizada no projeto.

#### **4.1.1 Confeção do sistema de energia (extensão)**

Para proporcionar uma maior segurança a este projeto e também uma maior portabilidade do mesmo, foi confeccionado um sistema de fornecimento de energia para os equipamentos que serão conectados como carga ao projeto. Este sistema funciona como uma extensão, e é constituído por um plug macho que será conectado à fonte de energia (geralmente uma tomada), um cabo que será responsável pela transmissão da energia e um plug fêmea onde serão ligados equipamentos diversos (cargas).

Este equipamento foi criado também pelo fato de serem necessárias invasões à este sistema, como será explicado mais adiante. Com este equipamento, a tarefa de criar invasões se torna menos perigosa, tendo em vista que basta desligá-lo da tomada para que nenhuma corrente esteja passando pelo sistema, diminuindo o risco de choques.

#### **4.1.2 Integração da Fonte Ajustável**

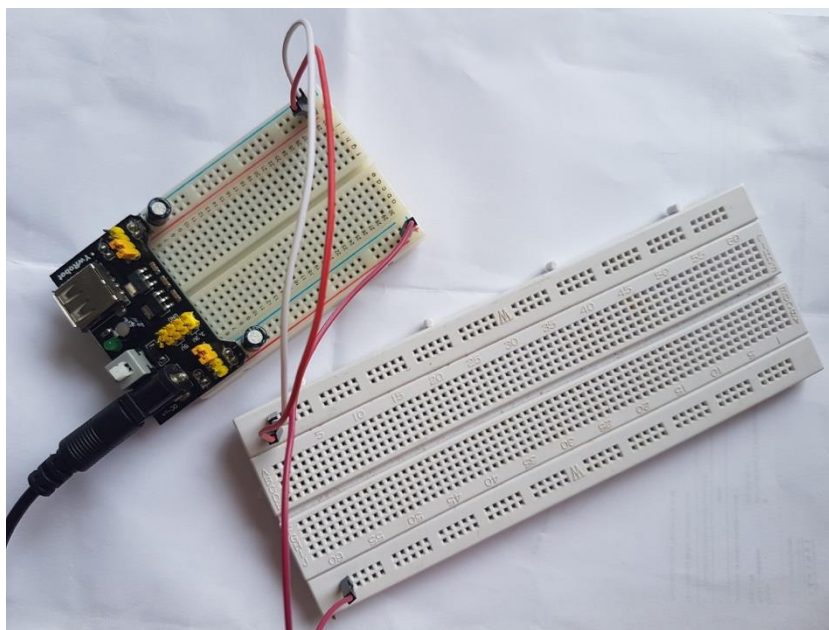
Este equipamento foi o primeiro a ser inserido no projeto, pois ele é responsável por fornecer energia a todos os outros equipamentos que serão utilizados no projeto. Sua conexão se dá de maneira muito simples, bastando encaixá-lo na protoboard, selecionar as tensões que serão fornecidas em cada uma das extremidades onde existem portas de saída e ligar o mesmo na fonte de energia (No caso, uma fonte comum de 12 V e 1.25 A). Para melhorar sua

performance, foram inseridos dois capacitores de  $100\mu\text{F}$  em cada uma das portas de saída, diminuindo a ocorrência de interferências e perda de corrente.

Devido às suas dimensões, este equipamento teve que ser inserido em uma protoboard diferente da que foi utilizada para todos os outros equipamentos utilizados neste projeto. Assim, se fez necessária a conexão desta protoboard com a utilizada pelos outros equipamentos. Esta conexão foi feita utilizando jumpers (cabos) comuns.

A figura 4.1 demonstra como foi feita a integração deste equipamento ao projeto.

**Figura 4.1 – Integração da Fonte Ajustável YWRobot Breadboard Power Supply ao projeto.**



**Fonte: O Autor.**

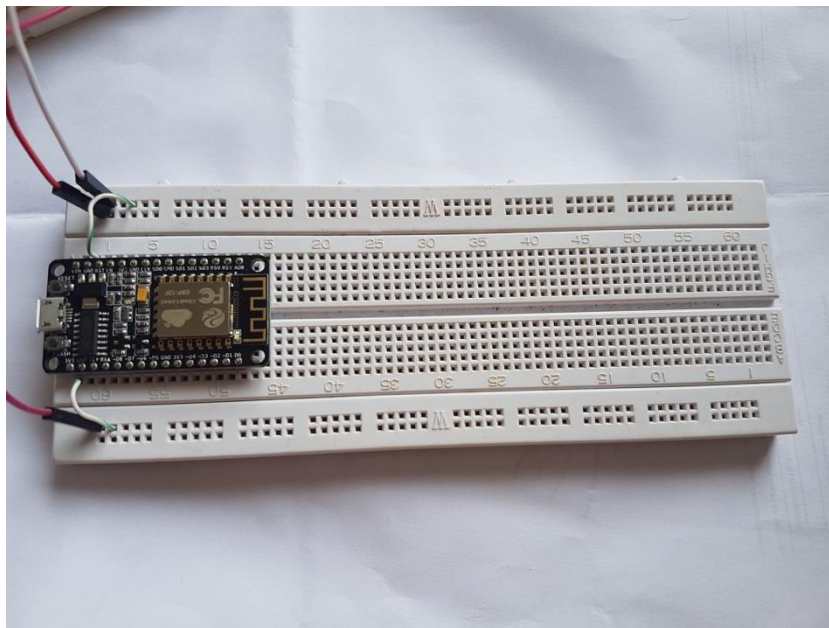
### **4.1.3 Integração do ESP8266**

Em seguida, foi inserido no projeto o microcontrolador ESP8266, por ser considerado a parte central do projeto, pois recebe, processa e envia informações para os equipamentos e para a internet. Sua integração também é feita de maneira simples, bastando encaixá-lo na protoboard, conectar um jumper saindo da sua porta *GND* à linha de aterramento da protoboard e outro

jumper saindo da sua porta 3V3 para a linha de fornecimento de 3,3V da protoboard.

A figura 4.2 demonstra como foi feita a integração deste microcontrolador ao projeto.

**Figura 4.2 – Integração do microcontrolador ESP8266 ao projeto**



Fonte: O Autor

#### **4.1.4 Integração do Sensor de Corrente**

Feito isto, foi feita a integração do sensor de corrente ACS712. Sua integração com projeto foi um pouco mais complexa que a dos componentes anteriores por duas razões.

A primeira razão por trás deste aumento de complexidade é que além de precisar integrar o sensor ao microcontrolador, também é necessário inserir o mesmo ao sistema onde serão feitas as medições de corrente. Para inserir este equipamento no sistema, é necessária uma invasão ao mesmo, sendo necessário cortar cabos e manipular tensões maiores que as utilizadas até o momento.

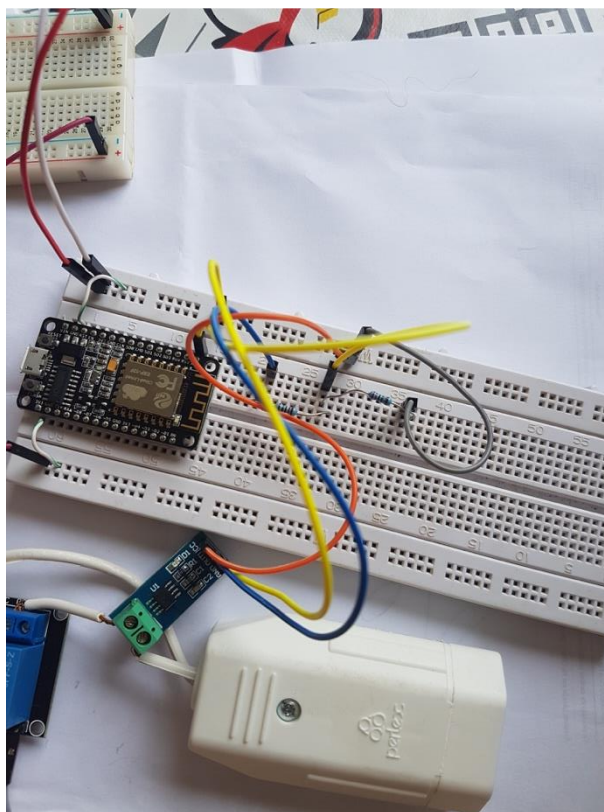
Feito isso ainda foi necessário manipular a tensão de saída fornecida deste equipamento para o microcontrolador, pois este equipamento é capaz de fornecer tensões de saída de até 5 V, porém o ESP8266 suporta tensões de

entrada que vão apenas até 3,3 V. Para regular esta tensão e não causar danos ao microcontrolador, foram inseridos dois resistores de precisão ao projeto, sendo um resistor de 11,6 K $\Omega$  e outro de 1,26 K $\Omega$ . A tensão resultante após a passagem por esses dois resistores é então fornecida à entrada *ADO* do microcontrolador.

Por fim, basta conectar um jumper saindo da sua porta *GND* à linha de aterramento da protoboard e outro jumper conectando a sua porta *IN* com a linha de fornecimento de 5 V da protoboard.

A figura 4.3 demonstra como foi feita a integração deste sensor ao microcontrolador e ao sistema.

**Figura 4.3 – Integração do sensor de corrente ACS712 ao projeto.**



Fonte: O Autor.

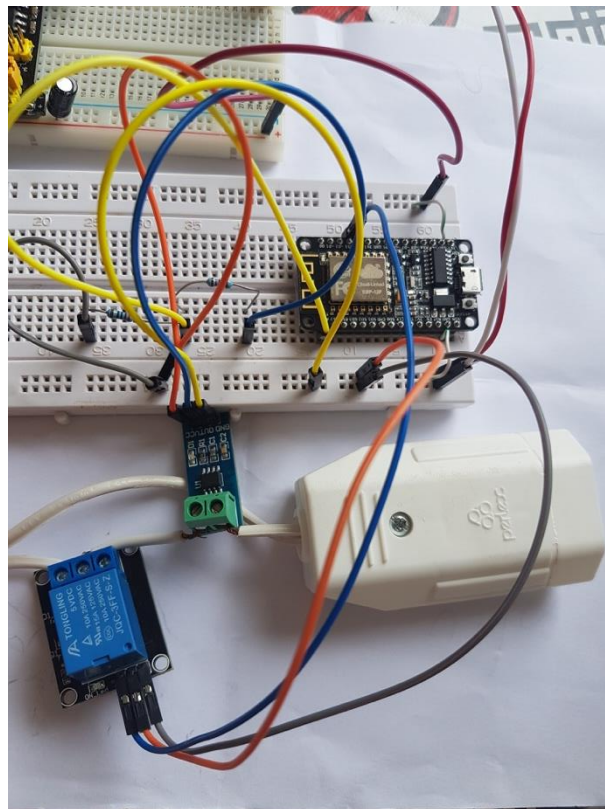
#### 4.1.5 Integração do módulo relé

Para a inserção do módulo relé, também foi necessária uma invasão ao sistema. Esta invasão é necessária para que o equipamento seja capaz de cortar a corrente do sistema quando for necessário. Além disso, é necessário conectar o módulo relé ao microcontrolador. Para fazer esta conexão basta conectar um

jumper saindo da sua porta “-” à linha de aterramento da protoboard, outro jumper conectando a sua porta “S” com a entrada *D3* do microcontrolador, e um terceiro Jumper conectando sua porta “+” com a linha de fornecimento de 5 V da protoboard.

A figura 4.4 demonstra como foi feita a integração do módulo relé ao microcontrolador e ao sistema.

**Figura 4.4 – Integração do módulo relé ao projeto.**

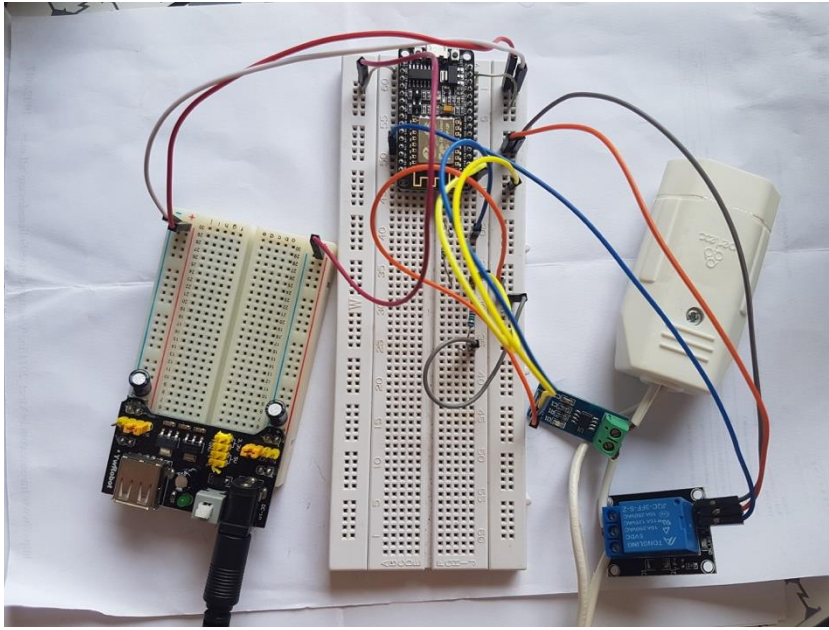


Fonte: O Autor.

#### 4.1.6 Hardware completo

Após todo esse processo, a parte de hardware do projeto está completa. Todos os equipamentos necessários foram devidamente conectados. A figura 4.5 é uma foto da parte de hardware do projeto completa.

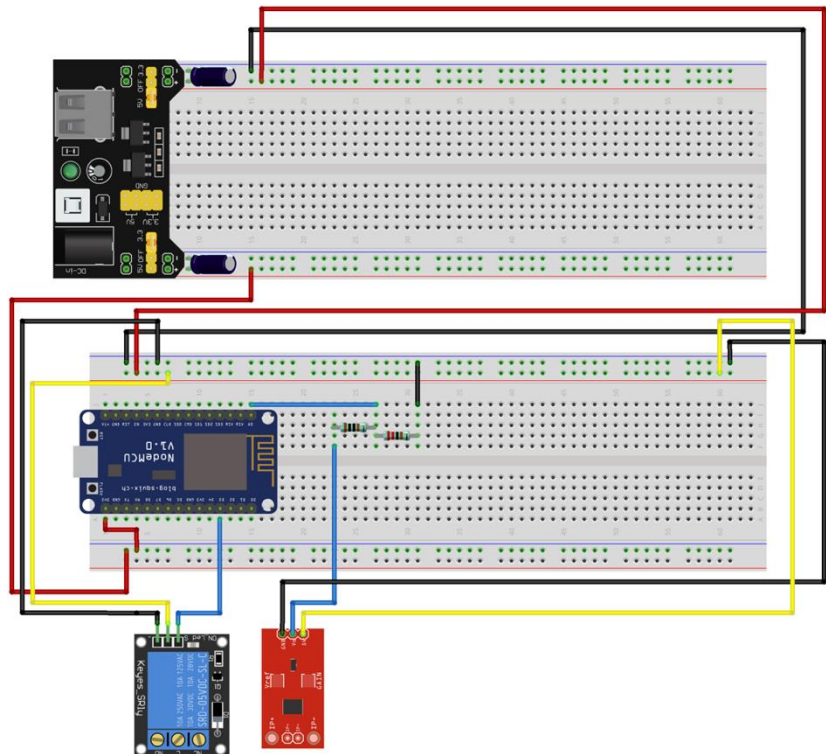
Figura 4.5 – Hardware do projeto completo.



Fonte: O Autor.

Para representar o hardware de forma melhor e de mais fácil entendimento, foi criado um *footprint* e um esquemático do projeto. A figura 4.6 é um *footprint* do hardware e a figura 4.7 é um esquemático do mesmo. Estas duas representações foram criadas utilizando o software Fritzing (Fritzing, 2017).

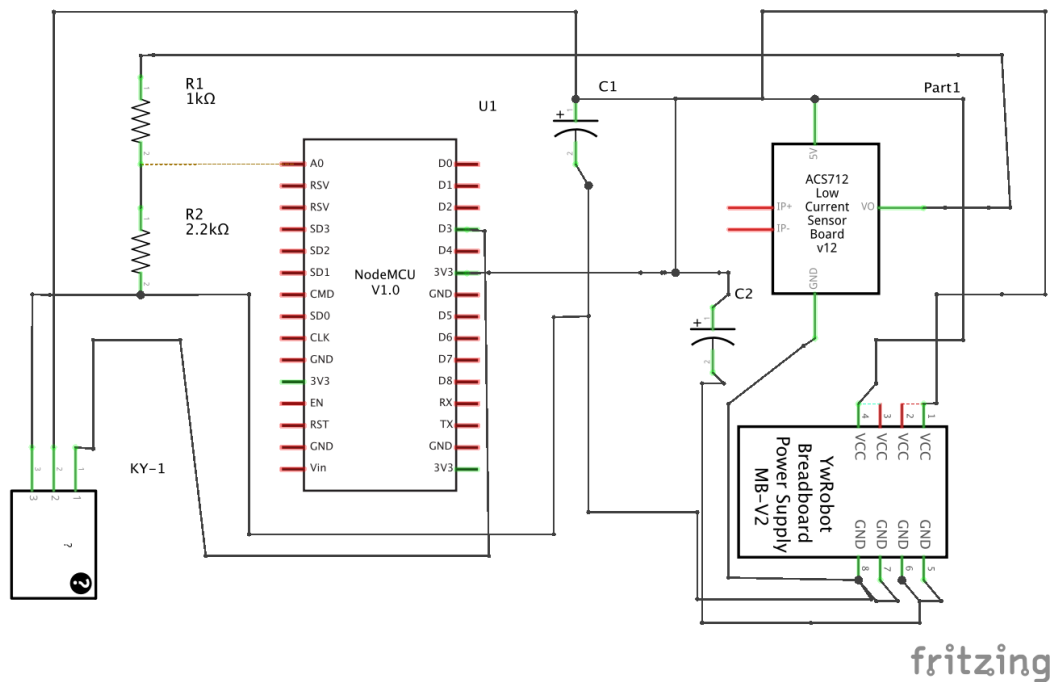
Figura 4.6 – Footprint do hardware.



fritzing

Fonte: O Autor.

Figura 4.7 – Esquemático do hardware.



Fonte: O Autor.

## 4.2 Software Embarcado

Como foi dito anteriormente, foi criado um software para ser embarcado no ESP8266 utilizando a Arduino IDE. Essa IDE gerou um software na linguagem de programação Arduino que, segundo seus criadores, “a linguagem Arduino é meramente um conjunto de funções C / C ++ que podem ser chamadas a partir do seu código” (Arduino, 2017).

Neste capítulo será abordado todo o processo de criação do software embarcado, citando os cálculos e equações necessários para a leitura correta do sensor de corrente, a comunicação com o módulo relé e o envio e recebimento de dados para a web utilizando o protocolo MQTT.

### 4.2.1 Conexão WiFi

O microcontrolador ESP8266 vem com um chip WiFi integrado, o que facilita bastante a execução deste projeto. Porém, para a utilização do mesmo se faz necessário a criação de funções e variáveis específicas no código fonte.

Primeiramente são criadas as constantes *SSID* e *PASSWORD* (ambas do tipo *char\**), que serão utilizadas como parâmetros para estabelecer a conexão com a rede WiFi que se deseja.

Em seguida é criada a função *initWiFi()* que será a responsável apenas por passar informações de que a tentativa de conexão com a rede WiFi por parte do dispositivo foi iniciada. Já a função *reconnectWiFi()* é a responsável por iniciar a conexão com a rede WiFi e por tentar se reconectar caso a conexão seja perdida.

#### 4.2.2 Leitura do Sensor de Corrente ACS712

A leitura do sensor de corrente neste projeto se dá dentro da função *getVPP()*. Dentro desta função são iniciadas as variáveis *result* (do tipo *float*, sem valor inicial definido), *readValue* (do tipo *int*, sem valor inicial definido), *maxValue* (do tipo *int*, com o valor inicial igual a 0), *minValue* (do tipo *int*, com o valor inicial igual a 1024) e *start\_time* (do tipo *uint32\_t*, que recebe *millis()*, que é o valor que representa em milissegundos o momento em que a função *getVPP()* começou a ser executada no processador).

Após a criação dessas variáveis, é criado um laço *while* que é executado durante um segundo, comparando a função *millis()* atual com a que foi salva na variável *start\_time*. Dentro deste laço a variável *readValue* recebe o valor lido no sensor de corrente ACS712 e compara este mesmo com os valores salvos nas variáveis *maxValue* e *minValue*. Caso o valor contido em *readValue* seja superior ao valor atualmente contido em *maxValue*, o valor contido em *maxValue* é substituído pelo valor contido em *readValue*. O mesmo acontece com a variável *minValue*, com a ressalva que neste caso a substituição ocorre caso o valor contido em *readValue* seja inferior ao valor contido na variável *minValue*.

Obtidos esses valores medidos, eles são inseridos em uma equação para obter um valor único. Nessa função é subtraído o valor mínimo obtido do valor máximo obtido, multiplica-se esse valor pela corrente de entrada (3,3 V no caso) e depois divide-se isso tudo pela resolução de bits do ACS712 (no caso, 1024).

O resultado dessa equação é salvo na variável *result*. A equação utilizada é a equação 4.1.

**Equação 4.1 – Equação para obtenção de resultado enviado pelo sensor de corrente.**

$$result = \frac{(maxValue - minValue) \times 3,3}{1024}$$

Feita essa equação, utilizamos a variável *result* em outra equação para obter a tensão RMS do circuito. Neste caso a equação utilizada foi uma das mais comuns e mais recomendadas para este tipo de aplicação. O resultado dessa equação é salvo na variável *VRMS*. A equação utilizada foi a equação 4.2.

**Equação 4.2 – Equação para obtenção da tensão RMS do circuito.**

$$VRMS = \frac{result}{2} \times 0.707$$

Obtida a tensão RMS do circuito, é aplicada uma terceira equação para obter a corrente RMS do circuito. Nesta equação o valor de *VRMS* é multiplicado por 1000 e depois dividido pela variável *mVperAmp*, que é a sensibilidade do sensor de corrente ACS712 de -30 A a +30 A (neste caso, 66). O resultado dessa equação é salvo na variável *AmpsRMS*. A equação utilizada foi a equação 4.3.

**Equação 4.3 – Equação para obtenção da corrente RMS do circuito**

$$AmpsRMS = \frac{VRMS \times 1000}{mVperAmp}$$

Feito isso, a função retorna o valor de *AmpsRMS*.

### 4.2.3 Comunicação com o módulo relé

A comunicação com o módulo relé é executada dentro da função *mqtt\_callback* de forma simples e efetiva. Dentro desta função é criada a variável *msg* (do tipo *string*, sem valor inicial definido). Essa variável recebe a mensagem (payload) enviada pelo site para o microcontrolador. Caso a mensagem seja apenas a letra “L”, o relé é ligado, o que ocasiona o bloqueio da corrente (corta corrente). Caso a mensagem seja apenas a letra “D”, o relé é desligado, o que ocasiona o desbloqueio da corrente, deixando a corrente passar normalmente.

O uso das letras “L” e “D” para ligar e desligar o relé foi convencionado para que a mensagem enviada pelo site fosse curta, simples e de fácil entendimento pelo processador, gerando rapidez no processo e utilizando poucos recursos do microcontrolador.

#### 4.2.4 Envio e recebimento de dados utilizando o protocolo MQTT

O Protocolo MQTT compõe a maior parte do código, tendo em vista que são necessárias diversas funções, definições e inclusões para a utilização do mesmo. Logo no início do código fonte temos a importação da biblioteca *PubSubClient.h*, que foi criada para facilitar a utilização do protocolo MQTT em diversos microcontroladores por meio de funções e extensões.

Em seguida são definidos *TOPICO\_SUBSCRIBE*, *TOPICO\_PUBLISH* e *ID\_MQTT*. Os dois primeiros são os tópicos MQTT utilizados para receber e enviar dados, respectivamente. Estes devem ser únicos, para que não haja conflito e o microcontrolador não venha a se conectar em outro projeto. O último é utilizado para identificação da sessão. Este também deve ser único, pois no caso de haver mais de um dispositivo conectado simultaneamente e utilizando a mesma identificação, o *broker* MQTT irá fechar a conexão de um deles.

Após isto são definidas as variáveis *BROKER\_MQTT* e *BROKER\_PORT* que são, respectivamente, a url e a porta do *broker* que será utilizado no projeto. Neste caso utilizamos o *broker eclipse*, que é um servidor público e gratuito oferecido pelo próprio MQTT.

Em seguida é criado o objeto *espClient*(do tipo *WiFiClient*, que é utilizado para se conectar à internet via WiFi), que é utilizado na criação do objeto *MQTT* (do tipo *PubSubClient*, utilizado para se conectar aos servidores MQTT). Estes objetos são necessários para que a conexão via WiFi com os servidores MQTT possa ser estabelecida.

Feito isso, é criada a função *initMQTT()* que é responsável por definir o servidor e a função de *callback*, que é a função utilizada quando é recebida uma mensagem do tópico em que o dispositivo está inscrito.

A próxima função a ser criada é a função *mqtt\_callback*, que é exatamente a função responsável por receber e processar as mensagens do tópico em que o dispositivo está inscrito. Como foi dito anteriormente, esta função também é utilizada para se comunicar com o relé.

Na sequência é criada a função *reconnectMQTT()*, que é responsável por tentar reestabelecer a conexão com o servidor caso a mesma seja perdida. Esta função depende da função *VerificaConexoesWifiEMQTT*, que é criada em seguida e é responsável por verificar o estado da conexão WiFi com a internet e o estado da conexão do dispositivo com o servidor MQTT.

Uma função muito importante no projeto, a função *EnviaEstadoOutputMQTT()* é criada em seguida. Ela é responsável por receber o estado de todos os dispositivos utilizados no projeto, montar uma mensagem concatenando todas estas informações e enviar ao servidor, para que esta mensagem possa ser recebida pelo site.

#### 4.2.5 Códigos Complementares

Além dos códigos supracitados, foram necessárias outras definições, funções e variáveis no código, sendo algumas essenciais para o projeto e outras apenas para facilitar a visualização e tornar mais intuitivo o projeto.

Logo no início do código fonte existem vários *#define*, que vão de *D0* a *D10*. Essas definições se fazem necessárias pelo fato do mapeamento das portas digitais do ESP8266 ser feita de forma diferente de como é feito nos microcontroladores Arduino. Dessa forma, se se deseja utilizar o pino *D0*, por exemplo, é necessário digitar 16 (diferentemente de 0, como seria caso fosse um microcontrolador Arduino). A utilização deste método torna o código mais simples de ser escrito por quem está criando e de mais fácil entendimento por quem o está lendo.

Algumas funções padrão do Arduino também são utilizadas, como a função *setup()*, que inicializa as funções que serão utilizadas no código; a função *initSerial()* que habilita a comunicação via Serial Monitor, ajudando bastante na hora de fazer verificações no código, e a função *loop()*, que é

responsável por informar a ordem de execução das funções e repetir a execução das mesmas.

### 4.3 Website

Com o intuito de facilitar a utilização do projeto pelo usuário final, foi criado um website. Este website é responsável por estabelecer uma conexão com o servidor MQTT para onde o ESP8266 está enviando os dados, receber os mesmos, interpretá-los, exibi-los ao usuário final, dar opções ao usuário (ligar ou desligar tomadas) e enviar qualquer decisão tomada pelo usuário ao microprocessador.

Neste subcapítulo será abordada a construção do website, citando a razão da utilização de cada uma das ferramentas utilizadas na construção do mesmo e especificando onde cada uma destas ferramentas atua no website.

#### 4.3.1 Conexão com o servidor MQTT

A conexão do website com o servidor MQTT é feita utilizando dois scripts padrão necessários para grande parte dos projetos que utilizam MQTT.

O primeiro script é o script *mosquitto.js*, criado por Roger Light, um dos criadores do protocolo MQTT. A escolha pela utilização do mesmo foi feita pelo fato de ele ser um script bastante completo, o que facilita muito o trabalho, e por ele ter sido criado por um dos responsáveis pelo MQTT, o que proporciona uma grande credibilidade e confiabilidade ao script e ao projeto. Este script possui diversos parâmetros e funções necessárias para a o envio e recebimento de dados utilizando o protocolo MQTT.

O segundo script a ser utilizado é o *mqttws31.js*, criado por Andrew Banks, que também é um dos criadores do protocolo MQTT. A escolha pela utilização do mesmo foi feita pelas mesmas razões do protocolo anterior, pois este também é um script bastante completo, e que é de grande credibilidade e confiabilidade, já que foi escrito por um dos criadores do protocolo MQTT. Este script é utilizado de forma a proporcionar um melhor interfaceamento entre páginas web e servidores MQTT, pois possui diversas funções que tornam a

criação de páginas web que utilizam o protocolo MQTT um trabalho mais intuitivo e de fácil entendimento.

#### **4.3.2 Leitura, interpretação e envio de Dados**

Para a leitura, interpretação e envio de dados foi criado o script *tmo.js*. Optou-se por criar um script separado para que fosse mantido o conteúdo original dos scripts citados anteriormente, de forma a evitar conflitos e problemas durante o desenvolvimento do projeto.

Dentro deste script são criadas funções que se conectam com os elementos da página através de seus ID's, e atribuem ações que são executadas de acordo com a interação do usuário. Existem casos em que um clique em determinado botão estabelece uma conexão com o servidor MQTT, outros casos em que é enviada uma mensagem para ligar ou desligar a tomada, e outros diversos casos.

Também são criadas funções responsáveis por receber as mensagens enviadas pelo ESP8266 e interpretá-las. Esta "inteligência" é capaz de receber a mensagem, extrair e exibir informações referentes ao sistema como: o estado que o sistema se encontra (cortando ou permitindo corrente), a corrente que está passando naquele momento e se existe ou não algum equipamento externo conectado ao sistema.

Dentro deste script também são criadas funções responsáveis por enviar os parâmetros para efetuar ou desfazer a conexão com o servidor MQTT, enviar também parâmetros para se inscrever ou desfazer a inscrição em um tópico MQTT, exibir mensagens sobre o estado da conexão com o servidor MQTT e atualizar as informações na página web.

#### **4.3.3 HTML5 e CSS3**

Para dar estrutura ao site foram utilizadas duas ferramentas bastante populares, o HTML5 e o CSS3. Utilizando o HTML5 foram inseridos os scripts criados anteriormente para conectar com o protocolo MQTT e interpretar os dados enviados pelo MQTT. Também foram criados os elementos visuais do site,

como campos de preenchimento, botões, textos informativos e campos de exibição de informações.

Utilizando o CSS3 foram criados elementos para facilitar mais ainda o entendimento do site, como adição de cores ao site para diferenciar diferentes elementos, posicionamento correto para que elementos não ficassem espalhados de forma confusa no site e foram adicionadas propriedades de tabulação, de forma a tornar o site responsivo, ou seja, adaptável à diversos tipos de dispositivos e telas.

## 5 RESULTADOS

Após o desenvolvimento deste projeto e utilização do mesmo no cotidiano, foram verificados diversos benefícios que a utilização do mesmo proporcionou. Este capítulo será dedicado à discussão desses resultados, citando os benefícios, como foram alcançados e o impacto deles em diversos aspectos da vida do usuário final.

Porém, antes da verificação dos resultados, é necessário explicar como esse dispositivo foi utilizado no cotidiano. Por ser um dispositivo bastante portátil, esse dispositivo foi utilizado de diversos modos e em diversos locais. Certas vezes era utilizado como uma tomada onde haveria bastante movimentação, com diversos objetos sendo conectados à ela (não simultaneamente), outras vezes era utilizado como um bocal para lâmpadas, e outras vezes era transportado para locais fora da residência para ser testado em outras situações cotidianas.

O primeiro benefício, e mais fácil de ser enxergado, está na economia de energia que este projeto proporciona. Por ele fornecer a opção de desligar um equipamento à distância, muitas vezes um equipamento que estava gastando energia sem necessidade (desperdiçando), era desligado, gerando uma economia de energia substancial.

Vale destacar um caso ocorrido que retrata muito bem esse quesito: certa vez, o usuário final deste sistema estava fora de casa quando ficou em dúvida se havia apagado a lâmpada do seu quarto. Ao entrar no sistema, verificou que a lâmpada estava acesa e clicou para desligá-la. A lâmpada estava acesa há duas horas, e provavelmente permaneceria acesa por mais oito horas, que seria o tempo que levaria para o usuário retornar para sua residência.

Este caso demonstra o impacto que este projeto é capaz de gerar na diminuição do desperdício de energia, pois com apenas um exemplar do projeto foi possível poupar em torno de 0,48 kWh em um dia (Pois a lâmpada era de 60 W). Isto, segundo a cotação da CELPA de janeiro de 2018, proporcionaria uma economia de R\$ 0,40. Vale ressaltar que se trata de uma única tomada, e que se fossem utilizados mais exemplares do projeto, dando cobertura à uma

residência inteira, o desperdício de energia seria muito menor, gerando uma economia tanto de energia quanto financeira.

O projeto também proporciona uma maior comodidade, tendo em vista que ele dá a possibilidade de desligar qualquer tomada pela internet. Sendo assim, o usuário final é capaz de estar em casa e desligar algum equipamento que ficou ligado em um cômodo distante do qual ele está. Pode-se também utilizar o sistema para, antes de dormir, desligar todos os equipamentos que foram esquecidos ligados.

Também é possível, por exemplo, que o usuário queira deixar um equipamento ligado por um determinado tempo e que após este tempo o equipamento seja desligado. Assim, basta acessar o site do projeto e desligar a tomada do equipamento quando o tempo determinado for atingido.

Neste quesito são inúmeros os resultados positivos que o projeto traz. Além disso, com a comercialização deste projeto e um maior número de clientes, novas formas de utilização do mesmo apareceriam, dependendo apenas da criatividade de cada usuário final, o que proporcionaria resultados que talvez hoje não estejam tão óbvios para quem está por trás do projeto.

O projeto também proporciona uma portabilidade de objetos não vista anteriormente em outros produtos comercializados. Por exemplo: ao invés de comprar uma lâmpada considerada “smart” apenas pelo fato de poder ser ligada ou desligada à distância, basta acoplar a lâmpada já existente em um bocal conectado à tomada inteligente. Sendo assim, diversos objetos poderiam ser portados para a internet das coisas, reduzindo o descarte de objetos que ainda estão em bom estado de utilização. Este quesito também proporciona uma economia financeira, tendo em vista que muitos desses objetos “smart” possuem um custo bastante elevado.

Economia financeira também é um dos resultados, sendo este observado diversas vezes no decorrer desse projeto. O custo de um exemplar do projeto foi de R\$ 97,83. Poucos, ou quase nenhum, dos objetos considerados “smart” possuem um custo inferior ou parecido a este (boa parte chega a custar mais que o dobro disto).

Foram pesquisados diversos objetos, observados seus modos de funcionamento e analisado se a inteligência deles poderia ser substituída por uma tomada inteligente e uma versão não-smart daquele objeto. A conclusão que se chegou é que em diversos casos, a tomada inteligente e a versão não-smart juntas custavam metade ou menos da metade do custo para comprar uma versão smart do mesmo produto.

Também é válido citar que no mercado brasileiro não é encontrado nenhum concorrente, não existem tomadas inteligentes à venda no Brasil. É possível importar tomadas inteligentes de outros países, mas o processo de importação é bastante moroso e complexo. Caso este projeto chegasse a ser um produto comercializado, além de ser um produto de baixo custo ainda seria um produto pioneiro no país. Também, caso fosse comercializado, seu custo diminuiria bastante, devido à produção em massa do mesmo.

## 6 CONCLUSÃO

O trabalho abordou três áreas de muita relevância atualmente: Internet das coisas, automação residencial e eficiência energética. A construção de um projeto que seja capaz de abordar essas três áreas traz muitos benefícios, tendo em vista que estas áreas estão em um ritmo de evolução acelerado.

A escolha do ESP8266 como microcontrolador desse projeto se mostrou uma escolha bastante acertada, tendo em vista que esse dispositivo operou de forma bastante ágil com os outros dispositivos do projeto e com a internet. Este dispositivo se mostrou bastante versátil, independente e com bastante poder de processamento, tornando ele o “cérebro” por trás da inteligência do projeto.

A montagem do projeto também se mostrou bastante simples, sem a necessidade de utilização de tecnologias muito complexas. Basta encaixar corretamente os dispositivos e o projeto funciona.

A elaboração do software embarcado já foi uma parte um pouco mais complexa, pois exige conhecimentos mais aprofundados de programação e redes, com envio e recebimento de dados pela web.

O protocolo MQTT facilitou e acelerou bastante o processo de desenvolvimento deste projeto. Isso se deu devido ao fato deste protocolo ter sido concebido justamente para o fim de conectar dispositivos à internet, facilitando o avanço e expansão da internet das coisas.

O desenvolvimento do website foi bastante simples, pois o mesmo também não utiliza de ferramentas muito complexas, já que este apenas exibe informações e fornece algumas opções ao usuário final. Apenas as ferramentas mais comuns para desenvolvimento web foram utilizadas no desenvolvimento deste website.

Os resultados se mostraram bastante satisfatórios, cumprindo com o que era esperado antes do desenvolvimento deste produto. Estes resultados mostram diversas vantagens na utilização deste projeto e também apontam rumos para o futuro do projeto, o que torna o projeto em si bastante empolgante,

pois os resultados são capazes de mostrar que o projeto possui a capacidade de gerar um impacto real no cotidiano das pessoas.

### **6.1 Trabalhos Futuros**

Como trabalhos futuros muitas melhorias e novas ferramentas podem vir a ser desenvolvidas. A primeira delas seria um aplicativo de smartphones para o projeto. O intuito deste aplicativo é de melhorar ainda mais a interação do usuário com o sistema, proporcionando novas experiências ao usuário.

Com o aplicativo será possível, por exemplo, uma interface mais intuitiva que a do website e que ficará ainda mais ajustada à cada dispositivo. Será possível também criar um sistema de notificação, onde o aplicativo verificará a localização do usuário, verificará a distância entre o usuário e sua residência e, caso o usuário esteja longe de sua residência, o sistema verificará se algum aparelho foi esquecido ligado e, caso a resposta seja positiva, o usuário receberá uma notificação alertando-o sobre isso.

Os projetos futuros também envolverão a criação de um software mais robusto. Pretende-se criar um sistema com criação de usuários, onde cada usuário terá seu login e senha, seus dados serão armazenados em nuvem e não será necessário inserir seus dados toda vez que for utilizar o sistema.

Este sistema mais robusto também contemplará a possibilidade de identificar cada tomada (por exemplo: tomada do quarto, tomada do banheiro, tomada da televisão, etc) e a geração de gráficos de consumo de energia individuais para cada uma das tomadas e também um gráfico de consumo geral, que reunirá o consumo de todas as tomadas.

Um grande passo no futuro será também integrar esse sistema aos grandes serviços de automação residencial. Grandes empresas como Google, Amazon, Apple, Microsoft e Samsung já possuem grandes sistemas inteligentes de automação residencial, e suas respectivas ferramentas de desenvolvimento abertas para que desenvolvedores consigam integrar seus projetos a elas.

Este passo é bastante importante, pois ao fazer essa integração, a quantidade de possíveis clientes aumenta significativamente, tendo em vista que

torna muito mais fácil a integração do projeto com o restante dos objetos inteligentes que o usuário já possui em sua residência.

## REFERÊNCIAS

ABESCO, Associação Brasileira das Empresas de Serviços de Conservação de Energia. “**Desperdício de energia custou mais de R\$60 bi para o Brasil nos últimos três anos**”. 2017. Disponível em: <http://www.abesco.com.br/pt/novidade/desperdicio-de-energia-custou-mais-de-r-60-bi-para-o-brasil-nos-ultimos-tres-anos/>. Acesso em 17/01/2018

ABESCO, Associação Brasileira das Empresas de Serviços de Conservação de Energia. “**O que é Eficiência Energética?**”. 2015. Disponível em: <http://www.abesco.com.br/pt/o-que-e-eficiencia-energetica-ee/>. Acesso em 21/01/2018

Amazon Company. “**3.3V 5V Breadboard Power Supply Module for MB102 Solderless Breadboard**”. 2017. Disponível em: <https://www.amazon.co.uk/Breadboard-Power-Supply-Module-Solderless/dp/B00BXWV2F6>. Acesso em 09/12/2017.

Apple Inc. “**A bold way to look at your health**”. 2016. Disponível em: <https://www.apple.com/lae/ios/health/>. Acesso em 08/02/2018.

Arduino Software. “**Frequently Asked Questions**”. Disponível em: <https://www.arduino.cc/en/Main/FAQ>. Acesso em: 10/12/2017.

BERTOLETI, Pedro. “**Controle e Monitoramento IoT com NodeMCU e MQTT**”. 2016. Disponível em: <https://www.filipeflop.com/blog/controle-monitoramento-iot-nodemcu-e-mqtt/>. Acesso em 09/12/2017.

Deal Extreme. “**NodeMcu Lua ESP8266 Wi-Fi Development Board - Black**”. Disponível em: <http://www.dx.com/p/nodemcu-lua-esp8266-wi-fi-development-board-black-436861#.WmkweqhKvIV>. Acesso em 09/12/2017

Espressif Systems, 2017. **Datasheet do microcontrolador ESP8266**. 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 09/12/2017

**Farmbot**. 2016. Disponível em: <http://farmbot.io/>. Acesso em 08/02/2018

Filipe Flop. “**Sensor de corrente invasivo ACS712**”. Disponível em: [https://www.filipeflop.com/wp-content/uploads/2017/07/sku\\_151391\\_1.jpg](https://www.filipeflop.com/wp-content/uploads/2017/07/sku_151391_1.jpg).

Acesso em 09/12/2017

**Fitbit**. Disponível em: <https://www.fitbit.com/home>. Acesso em 08/02/2018

FRANCO, Edson. “**Máquinas ganham versões autônomas, que dispensam o tratorista**”. 2016. Disponível em:

<http://blogs.canalrural.com.br/marcasemaquinas/2016/10/04/maquinas-ganham-versoes-autonomas-que-dispensam-o-tratorista/>. Acesso em 08/02/2018.

FRAZÃO, Juliane. “**Ceamazon promove minicurso sobre partida de motores elétricos e simulação de defeitos**”. 2017. Disponível em:

<http://pctguama.org.br/index.php/pt/2017/03/07/ceamazon-promove-minicurso-sobre-partida-de-motores-eletricos-e-simulacao-de-defeitos/>. Acesso em 21/01/2018.

FRAZÃO, Juliane. “**Ceamazon oferece projetos de instalação elétrica gratuitos a famílias de baixa renda**”. 2016. Disponível em:

<http://pctguama.org.br/index.php/pt/2016/08/18/ceamazon-oferece-projetos-de-instalacao-eletrica-gratuitos-a-familias-de-baixa-renda/>. Acesso em 21/01/2018.

**Fritzing Software**. Disponível em: <http://www.fritzing.org/home/>. Acesso em 09/12/2017.

Google. “**Chromecast Audio**”. 2015. Disponível em: <https://www.google.com/chromecast/audio/explore/>. Acesso em 08/02/2018.

**IRobot**. 2015. Disponível em: <http://www.irobot.com/For-the-Home/Vacuuming/Roomba.aspx>. Acesso em 08/02/2018.

ITU, International Telecommunication Union. “**Internet of Things Global Standards Initiative**”. 2015. Disponível em: <http://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx>. Acesso em 17/01/2018.

JOHNSON, Leif. “**Uber self-driving cars: everything you need to know**”. 2018. Disponível em: <http://www.techradar.com/news/uber-self-driving-cars> . Acesso em 08/02/2018.

KONGSBERG. “**Autonomous ship project, key facts about YARA Birkeland**”. Disponível em: <https://www.km.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/4B8113B707A50A4FC125811D00407045?OpenDocument>. Acesso em 08/02/2018.

MURATORI, José Roberto; DAL BÓ, Paulo Henrique. “**Automação Residencial: Histórico, Definições e Conceitos**”. 2012. Disponível em: [http://www.instalacoeseletricas.com/download/Automacao\\_residencial1.pdf](http://www.instalacoeseletricas.com/download/Automacao_residencial1.pdf). Acesso em 21/01/2018.

NEST. “**Meet the Nest Learning Thermostat**”. 2017. Disponível em: <https://nest.com/thermostats/nest-learning-thermostat/overview/>. Acesso em 08/02/2018.

NORDRUM, Amy. “**Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated**”. 2016. Disponível em: <https://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated>. Acesso em 17/01/2018.

**Orbital Systems**. Disponível em: <https://orbital-systems.com/>. Acesso em 21/01/2018

**Petchatz**. 2017. Disponível em: <https://petchatz.com/>. Acesso em 08/02/2018

PROCEL, Programa Nacional de Conservação de Energia Elétrica. “**Números sobre o desperdício de energia elétrica no Brasil**”. 2017. Disponível em: <http://www.procelinfo.com.br/main.asp?TeamID=%7B921E566A-536B-4582-AEAF-7D6CD1DF1AFD%7D>. Acesso em 03/01/2018.

SAMSUNG. “**Samsung Smart Refrigerator: Family Hub Touchscreen Fridge**”. 2015. Disponível em: <https://www.samsung.com/us/explore/family-hub-refrigerator/overview/>. Acesso em 08/02/2018.

TEZA, Vanderlei Rabelo. “**Alguns aspectos sobre a Automação Residencial – Domótica**”. Dissertação de Mestrado - Universidade Federal de Santa Catarina, 2002.

**The MQTT Organization**. Disponível em: <https://www.mqtt.org/>. Acesso em 10/12/2017.

Uber. “**The future of trucking**”. 2016. Disponível em: <https://www.uber.com/info/atg/truck/> . Acesso em 08/02/2018.

**Waymo**. Disponível em: <https://waymo.com/>. Acesso em 08/02/2018.

# ANEXO I – CÓDIGO FONTE DO SOFTWARE EMBARCADO

```
#include <ESP8266WiFi.h> // Importa a Biblioteca ESP8266WiFi
#include <PubSubClient.h> // Importa a Biblioteca PubSubClient

//defines:
//defines de id mqtt e tópicos para publicação e subscribe
#define TOPICO_SUBSCRIBE "TCCMatheusEngcompUfpa2018Envia" //tópico
MQTT de escuta
#define TOPICO_PUBLISH "TCCMatheusEngcompUfpa2018Recebe" //tópico
MQTT de envio de informações para Broker

#define ID_MQTT "TCCMatheusEngcompUfpa2018" //id mqtt (para
identificação de sessão)

//defines - mapeamento de pinos do NodeMCU
#define D0 16
#define D1 5
#define D2 4
#define D3 0
#define D4 2
#define D5 14
#define D6 12
#define D7 13
#define D8 15
#define D9 3
#define D10 1

// WIFI
const char* SSID = "SEU_SSID"; // SSID / nome da rede WI-FI que
//deseja se conectar
const char* PASSWORD = "SUA_SENHA"; // Senha da rede WI-FI que deseja
//se conectar

// MQTT
const char* BROKER_MQTT = "iot.eclipse.org"; //URL do broker MQTT que
//se deseja utilizar
int BROKER_PORT = 1883; // Porta do Broker MQTT
```

```
//Variáveis e objetos globais
WiFiClient espClient;          // Cria o objeto espClient
PubSubClient MQTT(espClient); // Instancia o Cliente MQTT passando o
                               //objeto espClient
char EstadoSaida = '0'; //variável que armazena o estado atual da saída

//Prototypes
void initSerial();
void initWiFi();
void initMQTT();
void reconnectWiFi();
void mqtt_callback(char* topic, byte* payload, unsigned int length);
void VerificaConexoesWiFIEMQTT(void);
void InitOutput(void);
float getVPP();

//Variaveis para medição da energia
const int sensorIn = A0;
int mVperAmp = 66; // Sensibilidade do ACS712 de 30A
double Voltage = 0;
double VRMS = 0;
double AmpsRMS = 0;

/*
 * Implementações das funções
 */
void setup() {
    //inicializações:
    InitOutput();
    initSerial();
    initWiFi();
    initMQTT();
    ESP.wdtDisable();
    ESP.wdtEnable(WDTO_8S);
}

void initSerial() {
    Serial.begin(115200);
}
```

```
void initWiFi() {
    delay(10);
    Serial.println("-----Conexao WI-FI-----");
    Serial.print("Conectando-se na rede: ");
    Serial.println(SSID);
    Serial.println("Aguarde");

    reconnectWiFi();
}

//Função: inicializa parâmetros de conexão MQTT
void initMQTT() {
    MQTT.setServer(BROKER_MQTT, BROKER_PORT); //informa qual broker e
                                              //porta deve ser
                                              //conectado

    MQTT.setCallback(mqtt_callback); //atribui função de
                                      //callback
}

//Função: função de callback
void mqtt_callback(char* topic, byte* payload, unsigned int length) {
    String msg;

    //obtem a string do payload recebido
    for(int i = 0; i < length; i++) {
        char c = (char)payload[i];
        msg += c;
    }

    //toma ação dependendo da string recebida:
    if (msg.equals("L")){
        digitalWrite(D3, LOW);
        EstadoSaida = '1';
    }

    if (msg.equals("D")){
        digitalWrite(D3, HIGH);
        EstadoSaida = '0';
    }
}
```

```
//Função: reconecta-se ao broker MQTT
void reconnectMQTT() {
    while (!MQTT.connected()) {
        Serial.print("* Tentando se conectar ao Broker MQTT: ");
        Serial.println(BROKER_MQTT);
        if (MQTT.connect(ID_MQTT)) {
            Serial.println("Conectado com sucesso ao broker MQTT!");
            MQTT.subscribe(TOPICO_SUBSCRIBE);
        }
        else {
            Serial.println("Falha ao reconectar no broker.");
            Serial.println("Havera nova tentatica de conexao em 2s");
            delay(2000);
        }
    }
}

//Função: reconecta-se ao WiFi
void reconnectWiFi() {
    //se já está conectado a rede WI-FI, nada é feito.
    //Caso contrário, são efetuadas tentativas de conexão
    if (WiFi.status() == WL_CONNECTED)
        return;

    WiFi.begin(SSID, PASSWORD); // Conecta na rede WI-FI

    while (WiFi.status() != WL_CONNECTED) {
        delay(100);
        Serial.print(".");
    }

    Serial.println();
    Serial.print("Conectado com sucesso na rede ");
    Serial.print(SSID);
    Serial.println("IP obtido: ");
    Serial.println(WiFi.localIP());
}
```

```
//Função: verifica o estado das conexões WiFi e ao broker MQTT.
void VerificaConexoesWiFiEMQTT(void){
    if (!MQTT.connected())
        reconnectMQTT(); //se não há conexão com o Broker, a conexão é
                        //refeita

        reconnectWiFi(); //se não há conexão com o WiFi, a conexão é refeita
}

//Função: envia ao Broker o estado atual do output
void EnviaEstadoOutputMQTT(void){
    //Novo Código
    char mensagem[8];
    int i;

    if (EstadoSaida == '0')
        mensagem[0] = 'L';

    if (EstadoSaida == '1')
        mensagem[0] = 'D';

    //Envio da Corrente
    float Voltage;
    Voltage = getVPP();
    (Voltage<0.4)? mensagem[1]='N':mensagem[1]='S';
    char voltchar[5];
    sprintf      (voltchar,      "%d.%02d      A",      (int)Voltage,
(int) (Voltage*100)%100);

    Serial.println("Voltchar:");
    Serial.println(voltchar);

    for(i=2; i<9; i++){
        mensagem[i] = voltchar[i-2];
    }

    Serial.print("Mensagem: ");
    Serial.print(mensagem);

    //Posso enviar várias coisas em um payload só e processar pelo site
    MQTT.publish(TOPICO_SUBSCRIBE, mensagem, 9);
```

```
Serial.println("- Estado da saída D3 e voltagem enviado ao broker!");
delay(1000);
}

//Loop principal
void loop() {
    //garante funcionamento das conexões WiFi e MQTT
    VerificaConexoesWiFiMQTT();

    //envia o status de todos os outputs para o Broker no protocolo
    EnviaEstadoOutputMQTT();

    //keep-alive da comunicação com broker MQTT
    MQTT.loop();
}

float getVPP(){
    float result;

    int readValue;           //valor lido no sensor
    int maxValue = 0;        // Valor maximo
    int minValue = 1024;     // Valor minimo

    uint32_t start_time = millis();
    while ((millis() - start_time) < 1000){ //Captar valores durante 1
segundo
        readValue = analogRead(sensorIn);

        // Compara o valor atual com o maxValue
        if (readValue > maxValue){
            //Salva o valor máximo do sensor
            maxValue = readValue;
        }

        // Compara o valor atual com o minValue
        if (readValue < minValue){
            //Salva o valor minimo do sensor
            minValue = readValue;
        }
    }
}
```

```
// Subtrai min do max, multiplica pela tensão e divide pela resolução
do ACS712
result = ((maxValue - minValue) * 5.0) / 1024.0;

VRMS = (result / 2.0) * 0.707;           //Equação da tensão RMS
AmpsRMS = (VRMS * 1000) / mVperAmp;    //Equação da corrente RMS
return AmpsRMS;
}
```

## ANEXO II – CÓDIGO FONTE DA PÁGINA WEB

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>SmartLet</title>
<link rel="stylesheet" type="text/css" href="style.css"/>
<script type='text/javascript' src='mosquitto.js'></script>
<script type='text/javascript' src='http://code.jquery.com/jquery-
1.7.2.min.js'></script>
<script type="text/javascript">
</script>
</head>
<body>
  <table border ="0" width="100%">
    <td width="30%">
      <center></center>
    </td>
    <td width="70%">
      <center><font><h1>SmartLet</h1><br>
Aluno: Matheus Castelo Branco Queiroz</font>
</center>
    </td>
  </table>

<table border ="0" width="100%">
  <td bgcolor="#f2021a" width="50%">
    <center>
      <h3>Tomada 1</h3>
      <div id="publish">
        <br>
        <font color="#3D3D3F"><b>1° passo:
</b></font> definir os tópicos MQTT de subscribe e publish:
        <p>Tópico de envio: <input class="txt"
id="pub-topic-text" type="text" width="200%"
value="TCCMatheusEngcompUfpa2018Envia"></p>
        <p>Tópico de recepcão: <input class="txt"
id="pub-subscribe-text" type="text"
value="TCCMatheusEngcompUfpa2018Envia"></p>
        <br>

```

```

</div>

<div id="broker">
    <font color="#3D3D3F"><b>2°
Passo:</b></font> conectar-se ao servidor / broker:<br><br>
    <input class="but" id="connect-button"
type="button" value="Conectar">
    <input class="but" id="disconnect-button"
type="button" value="Desconectar">
</div>
<br><br>
<font color="#3D3D3F"><b>3° passo:</b></font>
escolha a ação desejada:<br><br>
    <input class="but" id="liga-output"
type="button" value=" Ligar ">
    <input class="but" id="desliga-output"
type="button" value="Desligar"><br><br>

<p><font color="#3D3D3F"><b>Status do output
(vindos do NodeMCU via MQTT):</b></font></p>
<p>Corrente Atual: </p>
<div id="corrente"></div>
<p>Tem algo ligado? </p>
<div id="ligado"></div>
<p>Estado do sistema</p>
<div id="status_io"></div>

<p><b>Debug:</b></p>
<div id="debug"></div>
<script src="tmo.js"
type="text/javascript"></script>
</center>
</td>
<td bgcolor="blue" width="50%">
<center>
<h3>Tomada 2</h3>
<div id="publish">
<br>

```

```

        <font color="#3D3D3F"><b>1º passo:</b></font>
definir os tópicos MQTT de subscribe e publish:
        <p>Tópico de envio de informações: <input
class="txt" id="pub-topic-text" type="text"
value="TCCMatheusEngcompUfpa2018Envia"></p>
        <p>Tópico de recepcao de informações: <input
class="txt" id="pub-subscribe-text" type="text"
value="TCCMatheusEngcompUfpa2018Envia"></p>
        <br>

</div>

<div id="broker">
        <font color="#3D3D3F"><b>2º Passo:</b></font>
conectar-se ao servidor / broker:<br><br>
        <input class="but" id="connect-button"
type="button" value="Conectar">
        <input class="but" id="disconnect-button"
type="button" value="Desconectar">
</div>
<br><br>
<font color="#3D3D3F"><b>3º passo:</b></font> escolha
a ação desejada:<br><br>
        <input class="but" id="liga-output" type="button"
value=" Ligar ">
        <input class="but" id="desliga-output" type="button"
value="Desligar"><br><br>

<p><font color="#3D3D3F"><b>Status do output (vindos
do NodeMCU via MQTT):</b></font></p>
<div id="status_io"></div>

<p><b>Debug / respostas do servidor:</b></p>
<div id="debug"></div>

<script src="tmo.js"
type="text/javascript"></script>
</center>
</td>
</table>

```

</body>

</html>

## ANEXO III – SCRIPT TMO.JS

```
(function() {
    window.Main = {};
    Main.Page = (function() {
        var mosq = null;
        function Page() {
            var _this = this;
            mosq = new Mosquitto();

            $('#connect-button').click(function() {
                return _this.connect();
            });
            $('#disconnect-button').click(function() {
                return _this.disconnect();
            });
            $('#subscribe-button').click(function() {
                return _this.subscribe();
            });
            $('#unsubscribe-button').click(function() {
                return _this.unsubscribe();
            });

            $('#liga-output').click(function() {
                var payload = "L";
                var TopicPublish = $('#pub-topic-
text')[0].value;
                mosq.publish(TopicPublish, payload, 0);
            });

            $('#desliga-output').click(function() {
                var payload = "D";
                var TopicPublish = $('#pub-topic-
text')[0].value;
                mosq.publish(TopicPublish, payload, 0);
            });

            mosq.onconnect = function(rc){
                var p = document.createElement("p");
                var topic = $('#pub-subscribe-text')[0].value;
```

```
p.innerHTML = "Conectado ao Broker!";
$("#debug").append(p);
mosq.subscribe(topic, 0);
};

mosq.ondisconnect = function(rc) {
    var p = document.createElement("p");
    var url = "ws://iot.eclipse.org/ws";

    p.innerHTML = "A conexão com o broker foi
perdida";

    $("#debug").append(p);
    mosq.connect(url);
};

mosq.onmessage = function(topic, payload, qos) {
    var p = document.createElement("p");
    var q = document.createElement("p");
    var r = document.createElement("p");
    var acao = payload[0];
    var ligado = payload[1];
    var volt = "";
    var i = 0;

    for(i = 2; i<payload.length; i++){
        volt += payload[i];
    }

    //Escreve o estado do relé conforme informação recebida
    if (acao == 'L')
        p.innerHTML = "<center>Sistema Ativado
(Cortando Corrente)</center>"
    else
        p.innerHTML = "<center>Sistema Desativado
(Corrente Passando)</center>"

    //Escreve o estado da tomada conforme informação recebida
    if (ligado == 'N')
        q.innerHTML = "<center>Provavelmente
Não</center>"
    else
```

```
q.innerHTML = "<center> Provavelmente  
Sim</center>"  
  
r.innerHTML = volt  
  
$("#status_io").html(p);  
$("#ligado").html(q);  
$("#corrente").html(r);  
};  
}  
  
Page.prototype.connect = function(){  
    var url = "ws://iot.eclipse.org/ws";  
    mosq.connect(url);  
};  
  
Page.prototype.disconnect = function(){  
    mosq.disconnect();  
};  
  
Page.prototype.subscribe = function(){  
    var topic = $('#sub-topic-text')[0].value;  
    mosq.subscribe(topic, 0);  
};  
  
Page.prototype.unsubscribe = function(){  
    var topic = $('#sub-topic-text')[0].value;  
    mosq.unsubscribe(topic);  
};  
  
return Page;  
})();  
$(function(){  
    return Main.controller = new Main.Page;  
});  
}).call(this);
```