

Monitoramento de Ar-condicionado para Pequenos Provedores utilizando MQTT

Denis N. Conceição¹, Joaquim P. Filho¹

¹Faculdade de Computação e Informática – Universidade Presbiteriana Mackenzie
Rua da Consolação, 930 – 01.302-907 – São Paulo – SP – Brasil

denis.nascimento@mackenzista.com.br, joaquim.pessoa@mackenzie.br

Abstract. *This article presents an air conditioning monitoring system for small ISP, performing periodic readings of certain sensors, collecting information that defines full operation such as temperature, electrical current, flame sensor and an IR emitter which will allow control of the air conditioner per application. Currently the local provider market Brazil is growing, and many do not have some kind of monitoring system. In this article we used the MQTT protocol, the Raspberrypi, Esp8266, DHT22 hardware, we also used a simple MQTT dashboard for viewing the collected data.*

Resumo. *Este artigo apresenta um sistema de monitoramento de ar-condicionado para pequenos provedores, realizando leituras periódicas de determinados sensores, coletando informações que define o pleno funcionamento tais como temperatura, corrente elétrica, possíveis focos de chamas e um emissor de IR que irá permitir o controle do Ar-condicionado por aplicativo. Atualmente o mercado de provedores locais no Brasil vem crescendo, e muitos não possuem algum tipo de sistema de monitoramento. Neste artigo foi utilizado o protocolo MQTT, os hardwares Raspberrypi, Esp8266, DHT22 e utilizado também um simples dashboard MQTT para visualização dos dados coletados.*

1. Introdução

A internet das coisas(IoT) fornece recursos que podem ser utilizados em vários sistemas para automação, logo vi a necessidade de utilizar IoT em um sistema para controlar e monitorar o Ar-condicionado de uma pequena central de provedor de internet, fazendo o uso do protocolo MQTT[Kodali and Soratkal 2016].

Nos últimos anos o mercado de provedores de internet no Brasil vem crescendo ano a ano de forma acelerada, oferecendo espaço para pequenas operadoras regionais. A maioria destas operadoras regionais implanta suas centrais em pequenos bairros, muitas sem nenhuma estrutura para uma central de provedor, com isso alguns vem tendo problema com o ambiente da central, que necessita de uma temperatura específica para ter o pleno funcionamento.

Com a inoperabilidade de um ar-condicionado na central, pode acarretar um mal funcionamento de equipamentos, danificando ou causando baixo desempenho, pois precisam de baixas temperaturas para funcionar adequadamente.

A automação proporciona uma série de benefícios aos usuários, muitas vezes a implementação possui um alto custo de investimento e complexidade na implementação. Este artigo busca simplificar a implantação junto ao baixo custo de investimento.

2. Objetivo geral

O objetivo geral deste trabalho é desenvolver um sistema para controlar e monitorar um ar-condicionado, abordando microcontroladores de baixo custo, aplicativos gratuitos e fácil implantação.

3. Objetivo específico

Os objetivos específicos são: implementar sensores de temperatura, corrente e chama, utilizando o Microcontrolador Esp8266 e a placa Wemos d1 mini. Instalar sistema operacional na Raspberry e o servidor Mosquito MQTT Server que efetuará a comunicação entre os sensores. Efetuar o clone do controle do ar-condicionado e implantar em um emissor de infravermelho utilizando o ESP8266 como microcontrolador para controlar o ar-condicionado remotamente, instalar aplicativos MQTTclient gratuitos nas plataformas de Smartphones e Notebooks disponibilizando uma dashboard que irá apresentar os dados coletados.

4. Referencial Teórico

Neste projeto iremos utilizar diversos componentes já utilizados no modelo de IoT. A ideia é interconectar diversos sensores com um servidor, que irá coletar e disponibilizar os dados para os clientes. Para este projeto Foi utilizado uma Raspberrypi, wemos d1 mini(esp8266), roteador 300Mbps, sensor de chama, sensor de temperatura DHT22, emissor de IR, sensor de corrente não invasivo, plataforma de desenvolvimento Arduino e o servidor MQTT mosquito.

4.1. Protocolo MQTT

MQTT é um protocolo desenvolvido na década de 90 pela IBM, um protocolo de transporte de sistema de mensagens de publicação e assinatura, tornou-se um dos protocolos que mais ganhou força nestes últimos anos com a evolução do IoT, pois é leve e projetado para troca de mensagens em redes de baixa largura de banda, alta latência e pequenos dispositivos, utilizando tcp/ip de um modo de envio confiável. Neste modelo de publicação e assinatura o protocolo MQTT define dois tipos de entidade na rede, broker e clientes. O broker é o servidor que recebe todas as mensagens dos clientes e em seguida envia para os clientes de destino relevantes. Os clientes são quais quer dispositivos que possam interagir com o servidor broker como mostra a figura 1, a interação é feita através de mensagens que são organizadas por tópico, os desenvolvedores definem quais clientes poderão publicar ou efetuar leituras no tópico [Yuan].

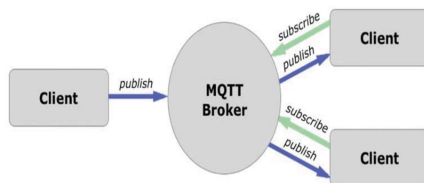


Figura 1. Modelo do protocolo MQTT
fonte:[Openhab.org]

- Publisher MQTT e o dispositivo que gera a informação e envia os dados para o broker MQTT.
- Subscriber MQTT e o assinante do serviço disponibilizado pelo Publisher Mqtt. Deve receber as informações geradas pelo Publisher pelo acesso ao servidor broker.

4.2. Raspberry Pi

Raspberry Pi e um computador que possui o tamanho de um cartão desenvolvida pela fundação Raspberry Pi. Por ser um computador compacto, barato e possuir um bom desempenho de processamento ele e muito utilizado em sistema de IoT, tornou-se um sucesso entre as comunidades. Pode ser utilizado com diversos sistemas operacionais existentes para arquitetura[Bhadoria and Ramos 2017].



Figura 2. Raspberry pi
fonte:[Raspberrypi.org]

4.3. ESP8266(wemos d1 mini)

Esp8266 e um micro controlador fabricado pela Espressif Systems um chip de baixo custo que vem ganhando espaço no cenário IoT pelo seu baixo custo e possuir um wifi integrado, possibilitando a conexão com diversos dispositivos na internet ou local, tais como sensores, roteadores ou computadores. Será utilizado o Esp8266 junto com o modulo Esp12 e a placa wemosd1 mini que permitirá um fácil manuseio, tornando mais pratica e fácil a programação do chip, pois a placa wemos permite programar o chip via interface USB [Kumiawan 2019].

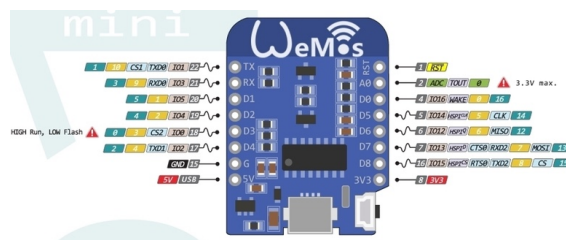


Figura 3. Placa Wemos d1 mini
fonte:[br arduino.org]

4.4. Sensor de Corrente Não Invasivo

O dispositivo SVT-013 é um sensor de corrente AC não invasivo de até 100. É utilizado em muitos projetos de automação residencial como medidor de corrente elétrica, iremos utilizar para identificar quando o Ar-condicionado for desligado, e futuramente já implantar uma medição de corrente, com o mesmo módulo sendo que este sensor não precisa romper o circuito elétrico para efetuar a instalação [Nama et al. 2016].



Figura 4. Sensor de corrente não invasivo
fonte:[filipeflop.com a]

4.5. Plataforma de desenvolvimento Arduino IDE

A plataforma IDE Arduino é o ambiente de desenvolvimento integrado do Arduino, ou seja, é o software utilizado para criação dos scripts que serão carregados no esp8266, a própria IDE tem suporte para efetuar o upload dos scripts para o chip, via cabo USB [Massimo. Banzi 2015] a figura 5 mostra a interface da IDE.

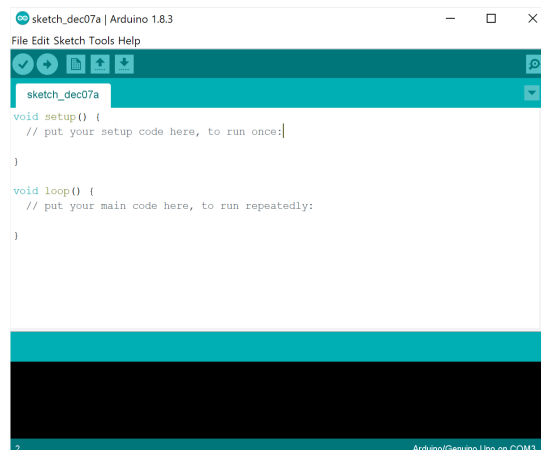


Figura 5. Plataforma de desenvolvimento arduino
fonte:[filipeflop.com b]

4.6. Modulo Sensor de Chama

Sensor de Chama é um sensor que identifica fontes de fogo ou fontes de luz do comprimento de 760nm a 1100nm, e baseado no sensor YG1006 que é um foto transistor de silicóneo NPN de alta velocidade e alta sensibilidade.

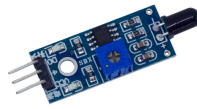


Figura 6. Modulo sensor de chama
fonte:[filipeflop.com c]

4.7. Sensor de Temperatura DHT22

O DHT22 é um sensor de temperatura e humidade de alta precisão. É capaz de medir a temperatura e humidade simultaneamente, utilizando apenas uma saída digital, esta característica o torna muito interessante, pois ele só ocupa um pino digital, podendo utilizá-lo com qualquer GPIO do Esp8266. O Arduino possui uma biblioteca específica para a leitura desse sensor tornando mais fácil a implementação.[Kodali and Mahesh 2016]

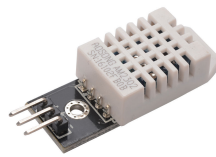


Figura 7. Sensor DHT22
fonte: [filipeflop.com d]

4.8. Led Infravermelho

O Led Infravermelho é um emissor de sinal de luz, emite um sinal IR que pode ser reconhecido por diversos receptores infravermelho, trabalhando numa baixa faixa de luz que o torna não perceptível a olho humano, ele é muito utilizado em sistemas de automação e em diversos controle remotos[Saputra and Lukito 2017].



Figura 8. Led Infravermelho
fonte:[filipeflop.com e]

4.9. Mosquito MQTT server

Mosquito é um software servidor de mensagens Broker que implementa o protocolo MQTT, e faz a intermediação entre a geração de dados e sua publicação. Está disponível na internet para download, operando em sistemas Linux, Mac e Windows[Upadhyay et al. 2016].

5. Metodologia

A arquitetura do sistema está dividida em 3 partes: Servidor, Aplicativo e Sensores conforme o modelo da figura 9.

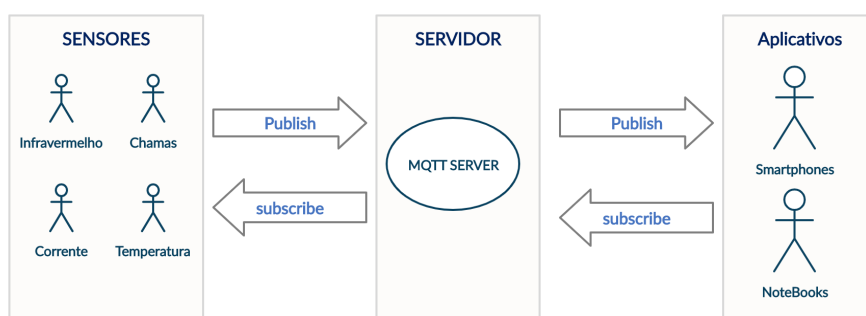


Figura 9. Arquitetura do sistema

fonte: Autoria própria

5.1. Servidor Mosquito

O servidor foi montado com uma Raspberry pi 3 com sistema operacional, sem interface gráfica apenas o básico para evitar processamento desnecessário, existem diversas plataformas nas nuvens disponíveis para ser utilizada como servidor MQTT, porém o intuito deste projeto é ser independente de serviços que agregam valores mensais, após a instalação do servidor mosquito alterar o arquivo de configuração desabilitando o acesso anônimo e adicionando acesso por senha evitando que qualquer outro equipamento indesejado utilize o servidor. Definir um endereço de IP estático, pois quando o acesso é externo, será necessário efetuar um redirecionamento de porta para o endereço ip do servidor, como a Raspberry pi já possui wifi apenas configurar a rede wifi como estática e definir seu devido endereço de rede.

5.2. Sensores

A parte dos sensores são compostos por 4 dispositivos wemos cada um com uma configuração específica, todos conectados em rede via wifi.

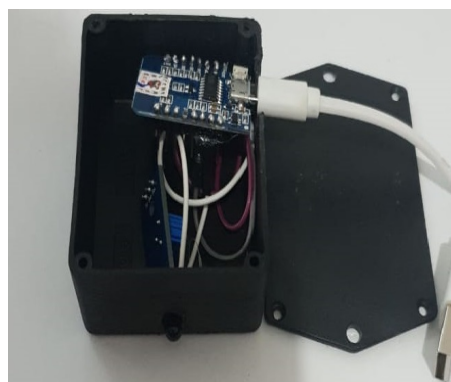
5.2.1. Sensor de Chama

É um conjunto da placa wemos d1 mini e o módulo sensor de chamas, este módulo possui os seguintes pinos D0, GND e VCC, para conectá-lo na placa será necessário efetuar a seguinte ligação o pino D0 do módulo sensor será ligado no D4 da placa wemos, o

pino GND do modulo ligar no GND da placa wemos, o pino VCC do modulo ligar na no 5V da placa wemos conforme figura 10(a), e isso termina a configuração física. Na parte logica foi instalado a biblioteca MQTT na IDE do Arduino, o codigo de conexao com servidor MQTT praticamente ja vem pronto no exemplo da bibliteca MQTT, apenas adptar para funcionar no nosso servidor, esse trecho do codigo será mesmo para a maioria dos dispositivo, alterando apenas o login. adaptado em uma pequena caixa para proteção conforme a figura 10(b).



(a) ligação física

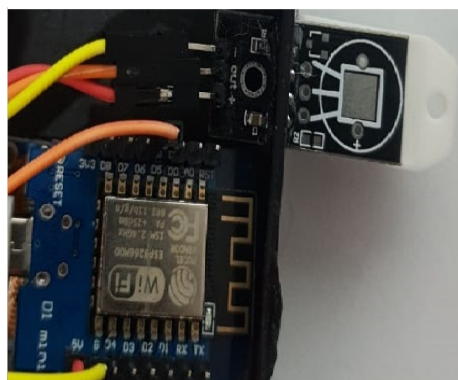


(b) Adaptação

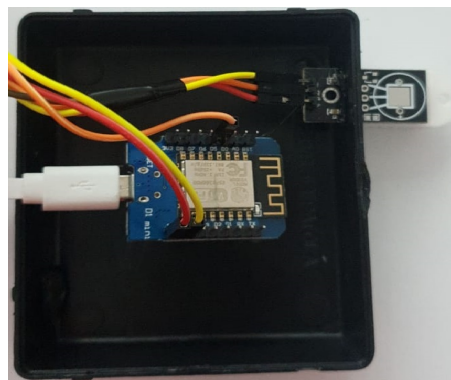
fonte: Autoria própria

5.2.2. Sensor de Temperatura

Para o sensor de temperatura utilizado a Wemos d1 mini com o DHT22, o DHT22 possui as seguintes pinagem: “+”, “-” e “Out” para conectá-lo na placa wemos será necessário ligar o pino “+” no 5V da placa wemos, o pino “-” no GND da placa wemos e o “Out” no pino D5 da placa wemos conforme a figura 10(c). O pino “out” pode ser ligado em qual quer pino digital da placa wemos lembrando que a e necessário efetuar a alteração das portas no script. A plataforma de desenvolvimento do Arduino possui uma biblioteca especifica para o DHT22 basta adicionar e utilizar os métodos “readTemperature() e readHumidity()”.



(c) ligação física



(d) Adaptação

fonte: Autoria própria

5.2.3. LED Infravermelho

Utilizaremos o LED infravermelho para controlar o ar-condicionado remotamente, A ligação física nos pinos do wemos e bem, simples basta ligar um pino do LED no GND da placa wemos o outro pino do LED ligar na porta D6 do wemos, antes de ligar no wemos temos que colocar um resistor de 220k Ω para diminuir a corrente, a plataforma de desenvolvimento do Arduino possuem uma biblioteca chamada “IRremote “ oferecendo total suporte para emitir os comando. Para que seja possível controlar o ar-condicionado será necessário clonar o controle original, adicionando todos os comandos clonados no código.



(e) ligação física



(f) Adaptação

fonte: Autoria própria

5.2.4. Sensor de Corrente

Utilizado o sensor de corrente não invasivo SCT013 junto com a wemos, pela facilidade de leitura e instalação, a plataforma do Arduino possui uma biblioteca chamada “Emon-Lib”, especifica para trabalhar com leitura de corrente, utilizaremos apenas o método de leitura de corrente para identificarmos quando a corrente na tomada do equipamento, tornando possível identificar uma eventual queda de energia. adaptando a placa wemos dentro de uma caixa conforme a figura 10(g) para uma proteção maior.

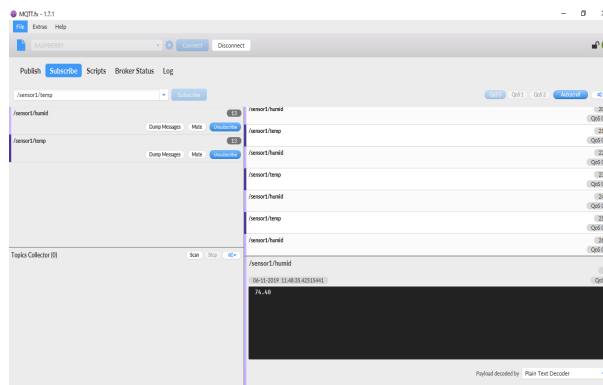


(g) adaptação

fonte: Autoria própria

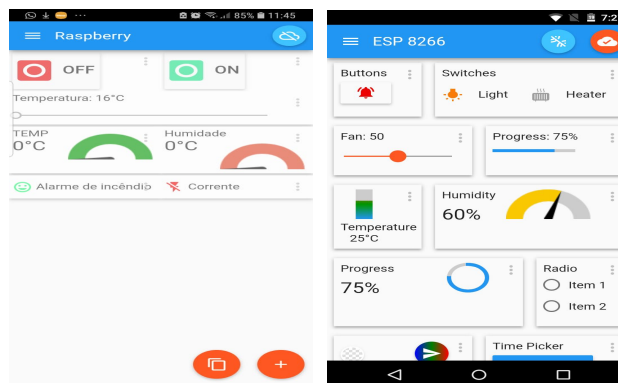
6. Aplicativo cliente MQTT

O protocolo MQTT oferece muita versatilidade, para termos uma interatividade entre um determinado sensor e necessário termos um aplicativo na qual possibilita enviar uma informação ou receber, existem muitos aplicativos open e específicos para MQTT podendo utilizá-los em computadores, smartphones, notebooks. Para visualizarmos a interação das mensagens com o servidor em um computador foi instalado software “MQTTfx” conforme figura 10(h) um software cliente desenvolvido em Java, o funcionamento é bem simples basta conectar no servidor mosquito com um usuário e senha e entrar no tópico onde o sensor esta anunciando as mensagens. Este software é utilizado para monitorar as mensagens e os sensores na implantação, mas existem inúmeros DASHBOARDS MQTT open para pc. Para smartphone foi utilizado um sistema gratuito



(h) MQTTFX, programa cliente MQTT para
fonte: Autoria própria

chamado IoT MQTT Panel, possui uma interface amigável e simples de utilizar, uma das vantagens de utilizar o protocolo MQTT e a facilidade de configurar os aplicativos cliente MQTT, independentemente da plataforma qualquer aplicativo que você adquirir ou desenvolver o processo é sempre o mesmo e não havendo necessidade de efetuar alterações no servidor, Na play store da google existem inúmeros cliente MQTT gratuitos e pagos, a escolha pelo IoT MQTT Panel e a apresentação do Dashboard mais amigável como mostra na figura 10(j).



(i) IoT MQTT panel (j) IoT MQTT panel
fonte: Autoria própria

Tabela 1. Custos

Descrição	QTD	Valor Unitario	Valor Total	local
Raspberry pi	1	R\$ 208,00	R\$ 208,00	amazon.com.br
wemos d1 mini pro com esp8266	4	R\$ 22,00	R\$ 88,00	mercadolivre
sensor de corrente não invasivo	1	R\$ 36,90	R\$ 36,90	mercadolivre
Modulo sensor de chama	1	R\$ 10,39	R\$ 10,39	mercadolivre
Sensor dht22	1	R\$ 32,50	R\$ 32,50	mercadolivre
LedInfravermelho	1	R\$ 2,20	R\$ 2,20	eletrogate.com
pequenas caixas plastica	4	R\$ 7,00	R\$ 28,00	
		Valor Total	R\$ 405,99	

fonte: Autoria própria

7. Resultados

Neste projeto foram abordadas opções fáceis de acesso e baixo custo para montar um sistema de monitoramento de um Ar-condicionado, na tabela 1 e mostrado o custo total do sistema para implantação. Fornecendo segurança, controle e simplicidade de configuração para seus usuários, O sistema foi implantado em uma pequena central em pleno funcionamento, sensores foram instalados próximo ao ar-condicionado, todos os sensores funcionaram perfeitamente, conseguindo coletar os dados periodicamente e enviando para o servidor, o cliente através de um Smartphone com o aplicativo MQTTclient conseguiu visualizar os dados coletados pelos sensores e controlar o ar-condicionado sem nenhum problema, utilizando uma interface amigável e gratuita disponível na internet.

Os testes foram realizados na rede interna e externa, ao realizar os testes na rede externa foi necessario efetuar um redirecionamento no roteador, onde todos os pacotes entrantes na porta 1883 do roteador reenviar para o endereço do servidor na porta 1883.

8. Conclusão

Com a evolução da Iot um pequeno dispositivo conectado diretamente a internet já é uma realidade no cenário atual, a medida que a tecnologia avança a uma necessidade de mais dispositivos estarem conectados entre si e com uma forte interação disponibilizando informações em tempo real , um exemplo é a chegada da rede 5G que proporcionara uma ampla conectividade com dispositivos, já existem diversos fabricantes com soluções prontas, mas existem muito interesse em soluções customizadas as suas necessidades.

Com o desenvolvimento deste projeto utilizando controladores e protocolos que vem ganhando força no cenário, forneceu uma forte interação com um ar-condicionado que não possui nenhum sistema de monitoramento de fabricação própria, ao concluir este projeto possibilitou monitorar a temperatura, corrente elétrica, detectar focos de chama e controlar o ar-condicionado através de um Smartphone, é possível concluir o quão importante é a automação, onde um pequeno hardware como esp8266 se conecta facilmente com a internet, trazendo e mantendo grandes informações não pelo volume e sim pela importância. Ao utilizar o protocolo MQTT obteve-se uma ideia da facilidade de integração com o front-end e back-end mostrando o quão pequeno é o hardware e já possui uma forte

integração na internet, trazendo e mantendo grandes informações não pelo volume e sim pela importância.

Ao utilizar o protocolo MQTT obteve-se uma ideia da facilidade de integração com o front-end e back-end mostrando o quão pequeno e o hardware e já possui uma forte integração na internet, a muito interesse de pessoas em explorarem esse setor buscando tecnologia de acesso gratuito.

Referências

- Bhadoria, S. and Ramos, R. (2017). *Raspberry Pi 3 Home Automation Projects*, volume 1. Packt Publishing Ltd.
- br arduino.org. Disponível em. <https://br-arduino.org/2016/03/wemos-d1-mini.html>.
- filipeflop.com. Disponível em. <https://www.filipeflop.com/produto/sensor-de-corrente-nao-invasivo-100a-sct-013/>.
- filipeflop.com. Disponível em. <https://blog.arduino.cc/2016/03/09/arduino-ide-1-6-8-available-for-download>.
- filipeflop.com. Disponível em. <https://www.filipeflop.com/produto/sensor-de-chama-fogo>.
- filipeflop.com. Disponível em. <https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22>.
- filipeflop.com. Disponível em. <https://www.filipeflop.com/produto/led-emissor-infravermelho-ir-5mm>.
- Kodali, R. K. and Mahesh, K. S. (2016). Low cost ambient monitoring using esp8266. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 779–782.
- Kodali, R. K. and Soratkal, S. (2016). Mqtt based home automation system using esp8266. In *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pages 1–5.
- Kumiawan, A. (2019). *Internet of Things Projects with ESP32*, volume 1. Packt Publishing Ltd.
- Massimo. Banzi, M. S. (2015). *Primeiros Passos com o Arduino*, volume 1. Novatec Editora.
- Nama, G. F., Despa, D., and Mardiana (2016). Real-time monitoring system of electrical quantities on ict centre building university of lampung based on embedded single board computer bcm2835. In *2016 International Conference on Informatics and Computing (ICIC)*, pages 394–399.
- Openhab.org. Disponível em. <https://www.openhab.org/addons/bindings/mqtt.generic/>. Accessed: 2019-10-1.
- Raspberrypi.org. Disponível em. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>.
- Saputra, L. K. P. and Lukito, Y. (2017). Implementation of air conditioning control system using rest protocol based on nodemcu esp8266. In *2017 International Conference on*

Smart Cities, Automation Intelligent Computing Systems (ICON-SONICS), pages 126–130.

Upadhyay, Y., Borole, A., and Dileepan, D. (2016). Mqtt based secured home automation system. In *2016 Symposium on Colossal Data Analysis and Networking (CDAN)*, pages 1–4.

Yuan, M. Conhecendo o mqtt. <https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>. Accessed: 2019-10-1.