

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET  
CÂMPUS GUARAPUAVA

GUILHERME RIBAS CARNEIRO

**SMART WINDOW - SENSOR DE JANELAS PARA A IOT**

PROPOSTA DE TRABALHO DE CONCLUSÃO DE CURSO DO CURSO  
SUPERIOR EM TECNOLOGIA EM SISTEMAS PARA INTERNET

GUARAPUAVA  
2017

GUILHERME RIBAS CARNEIRO

## **SMART WINDOW - SENSOR DE JANELAS PARA A IOT**

Proposta de Trabalho de Conclusão de Curso Do Curso Superior em Tecnologia em Sistemas para Internet de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 1, do Câmpus Guarapuava - TSI - da Universidade Tecnológica Federal do Paraná - UTFPR - Câmpus Guarapuava, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Hermano Pereira

Coorientador: Maurício Barfknecht

GUARAPUAVA  
2017

# 1 PROPOSTA DE TRABALHO DE CONCLUSÃO DE CURSO

## 1.1 TÍTULO

Smart Window - Sensor de janelas para IoT.

## 1.2 MODALIDADE DO TRABALHO

Desenvolvimento de Sistemas

## 1.3 ÁREA DO TRABALHO

Redes de Sensores sem fio, Microcontroladores, Monitoramento de ambiente e Programação em Lua.

## 1.4 RESUMO

Na atualidade a necessidade de segurança em imóveis é fundamental, onde assim, é possível encontrar soluções básicas no âmbito de sensores sem fio para portas e janelas. Os sensores hoje utilizados possuem uma tecnologia que transmite um sinal de rádio para uma central, onde muitas vezes há a necessidade de mão de obra especializada para a instalação da mesma. Esse projeto tem por objetivo o desenvolvimento de um sensor fazendo uso da plataforma ESP8266, sendo capaz de enviar dados sobre a situação atual de uma janela/porta em que este esteja instalado. A essência desse projeto é a fácil instalação, configuração e consistência dos dados obtidos.

## 2 GLOSSÁRIO

<b>IoT</b>	Internet Of Things
<b>MCU</b>	Microcontroller Unit
<b>USB</b>	Universal Serial Bus
<b>FTDI</b>	Future Technology Devices International
<b>TTL</b>	Transistor–Transistor Logic
<b>WPA</b>	Wi-Fi Protected Access
<b>REST</b>	Representational State Transfer
<b>PHP</b>	Hypertext Preprocessor
<b>HTTPS</b>	Hyper Text Transfer Protocol Secure
<b>API</b>	Application Programming Interface
<b>HTML</b>	HyperText Markup Language
<b>CSS</b>	Cascading Style Sheets

## 3 DESCRIÇÃO DA PROPOSTA

### 3.1 INTRODUÇÃO

O termo Sistemas Embarcados surgiu em 2007, onde define que estes possuem um desenvolvimento complexo e altas taxas de processamento computacional. Os Embarcados, são dispositivos com propósitos únicos para toda a sua vida útil como por exemplo, os sensores. Os sensores de ambientes tem como principal função monitorar a situação em que determinado objeto ou ambiente se encontra; onde caso haja alguma alteração quando comparada aos padrões definidos, medidas como enviar dados para o usuário ou até mesmo acionar um determinado serviço de emergência sejam tomadas (EMBARCADOS, 2013).

A necessidade de informações sobre o que esta acontecendo em um imóvel faz com que o desenvolvimento de sensores tenha prioridade, tornando possível que o proprietário do imóvel seja capaz de monitorá-lo de qualquer lugar.

No desenvolvimento do projeto proposto, será oferecida uma solução que monitore a real situação de um objeto, fornecendo assim, dados para o usuário através de uma interação realizada pelo *WebService* ou através de gatilhos pré-definidos. Esses gatilhos são caracterizados como, alteração de status (aberto para fechado ou vice-versa).

### 3.2 OBJETIVOS

#### 3.2.1 Objetivo Geral

Apresentar um sensor *Wireless* capaz de informar se uma janela está aberta ou fechada através de um *WebService*.

#### 3.2.2 Objetivos Específicos

- Implementar o monitoramento fazendo uso do nodeMCU ESP8266.
- Certificar-se do status atual da janela com o sensor Reed Switch.
- Estudar e implementar *WebService* para comunicação cliente-servidor.
- Integrar o nodeMCU ESP8266 com o sensor através da programação em Lua.
- Fazer uso de bateria para alimentar o sistema.

### 3.3 REVISÃO LITERÁRIA

#### 3.3.1 ESP8266

A empresa *Espressif Systems* desenvolveu uma solução de baixo custo e tamanho reduzido, o ESP8266, o qual é um chip voltado para aplicações no campo da Internet das Coisas (IoT) devido ao seu baixo consumo de energia. Seu consumo é baixo se comparado

ao *Chip Wifi* (HDG204) do Arduino, sendo no ESP8266 em torno de 20 microampéres no modo *deep-sleep*, aproximadamente 60mA enquanto recebe pacotes e picos de 215mA durante a transmissão de dados, enquanto que no HDG204 o consumo é de aproximadamente 30 microampéres, 41mA e 165mA respectivamente. Este chip é implementado em módulos que custam em torno de 3,50 dólares<sup>1</sup>, razão pela qual a solução foi selecionada para a proposta em questão (DOS SANTOS, p. 25, 2015), (MINATEL, 2016).

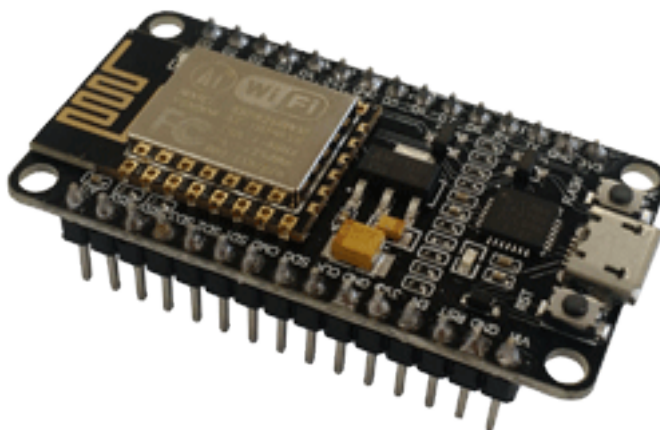


Figura 1 – NodeMCU ESP8266

Existem diversos modelos de módulos para diferentes implementações, dentre eles estão destacados os seguintes (Conforme Quadro 1): Obs.: Valores obtidos nos sites [www.ebay.com](http://www.ebay.com) e [www.filipeflop.com.br](http://www.filipeflop.com.br). Acesso em: 27 abr. 2017.

Quadro 1 – Comparativo entre módulos ESP8266.

	<b>nodeMCU ESP8266</b>	<b>Wemos D1 R2</b>	<b>Adafruit HUZZAH</b>	<b>ESP-201</b>
Portas de Comunicação	11	11	9	9
Regulador de Tensão	de 3,3v	de 3,3v	de 3,3v	Não possui
Conector	Micro USB- Serial	Micro USB- Serial	FTDI Serial TTL-232	FTDI Serial
Dimensões	49mm x 25,5mm x 7mm	70mm x 53,5mm x 12,5mm	25mm x 38mm x 5mm	34mm x 25,5mm x 3,5mm
Custo	R\$11,00	R\$59,90	R\$31,12	R\$44,90
Fabricante	AI-Thinker	Wemos Eletro- nics	Adafruit	AI-Thinker

Mediante a análise de requisitos e possíveis expansões futuras, para este projeto foi selecionado o módulo NodeMCU ESP8266 (Figura 1), devido ao fato de que este modelo

<sup>1</sup>[www.ebay.com](http://www.ebay.com)

possui: suporte a WPA/WPA2, maior número de portas de comunicação de entrada e saída, regulador de tensão 3.3v e alimentação por bateria.

### 3.3.2 Reed Switch

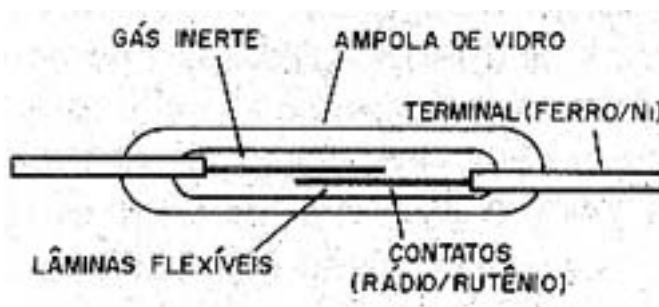


Figura 2 – Reed Switch (NEWTONCBRAGA, [20-?-?])

O *reed switch* (Figura 2) é composto por um ampola de vidro (bulbo) e duas lâminas. A ampola, possui no seu interior um gás nobre inerte, este gás por sua vez, tem a função de proteger as lâminas evitando que elas oxidem. Já as lâminas, são formadas por um material ferroso onde assim, torna-se possível que um ímã aja sobre elas (NEWTONCBRAGA, [20-?-?]).

Quando há a aproximação de um ímã ao bulbo do *reed switch*, um campo magnético é formado e a força gerada pelo polo norte do ímã passa pelas lâminas e chega ao polo sul deste, e, com isso, as lâminas se atraem uma a outra. Quando há o contato elétrico, pode-se dizer que o interruptor magnético está fechado, caso contrário, ele estará aberto (NEWTONCBRAGA, 20-?-?), (MAGALDI, p. 132, 2008).

### 3.3.3 LUA

A Linguagem de programação Lua foi projetada e desenvolvida no Brasil, inicialmente seu foco era suprir as necessidades da Petrobrás. Esta linguagem é de alto-nível, rápida e de fácil de incorporação em uma aplicação desenvolvida em outras linguagens devido a sua API. A Lua é *Open-Source*, ou seja, código aberto. Lua oferece suporte para programação orientada a objetos e a dados, onde sua semântica se destaca devido ao fato de ser extensível, permitindo a adição de novas funcionalidades sem modificar anteriores (CELES, DE FIGUEIREDO, IERUSALIMSKY), (LUA).

O grande destaque da linguagem Lua, deve-se ao fato que ela utiliza tabela para formar sua estrutura de dados. A tabela tem por função armazenar qualquer valor e indexar os elementos (HARDWARE, 2013).

### 3.3.4 Webservice REST e PHP

*REST* é uma ordem de princípios de arquitetura para que seja possível a construção de um *Webservice*, onde este vai consumir recursos de um determinado sistema ou aplicação.

Diversas linguagens implementam os princípios *REST* e dentre elas esta o PHP. Devido ao fato do PHP trabalhar com requisições (GET, POST, DELETE, PUT) facilita a implementação do *REST* evitando assim, o uso de *frameworks*. Atualmente, os *WebServices* implementados com o uso do *REST* também utilizam HTTP como camada de transporte. O HTTP funciona muito bem em conjunto com esta arquitetura devido a sua grande abrangência em servidores (PASQUA, 2015).



Figura 3 – Fluxo de mensagens (CHEMAXON, [201-?])

Na Figura 3 podemos ver o fluxo de troca de mensagens entre o Cliente e o Servidor REST, onde o HTTP é o intermediador destas requisições e respostas. Por padrão, para ocorrer a troca de mensagens, o cliente é quem inicia a conversação e o servidor apenas responde (FISCHER, 2013).

### 3.3.5 Produtos Relacionados

No mercado atual, encontra-se alguns sensores direcionados para a área de segurança. A empresa ECP desenvolveu o Sensor Magnético Sem Fio alimentado por uma pilha mini de 12v, o qual por sua vez tem a função de enviar um sinal de radio através da frequência 433,92Mhz para um central em uma distância de 15 a 25 metros onde, assim, será disparado uma sirene identificando que houve a abertura de um determinado setor (janela ou porta) (TUDOFORTE, [201-?]).

Um projeto desenvolvido por (SANTOS, 2016) tem por objetivo monitorar uma porta usando *reed switch* via cabo. O autor fez uso da plataforma Arduino para monitorar o status do *reed switch* onde, caso o ímã se afaste do bulbo do reed switch ele é detectado como "aberto" e em consequência, um led vermelho é aceso em sua *protoboard*.

O entusiasta (DIAMOND, 2016) desenvolveu um sistema para identificar se suas encomendas foram colocadas em sua caixa de correspondência. Mike fez uso do modulo ESP8266-07 juntamente com um *reed switch*, onde caso a tampa da sua caixa de correspondência seja



aberta, o ímã nela colocado se afasta do bulbo do *reed switch* e envia um sinal para a aplicação *Blynk* instalada em seu celular. O sistema é alimentado através de uma bateria com capacidade de 110mAh e 3.7v. O autor fez uso do recurso *DeepSleep* para economia de energia, esse recurso é implementado pelo ESP8266 e tem por função deixar o módulo dormindo enquanto não há nenhuma interação com o *reed switch* poupando, assim, energia.

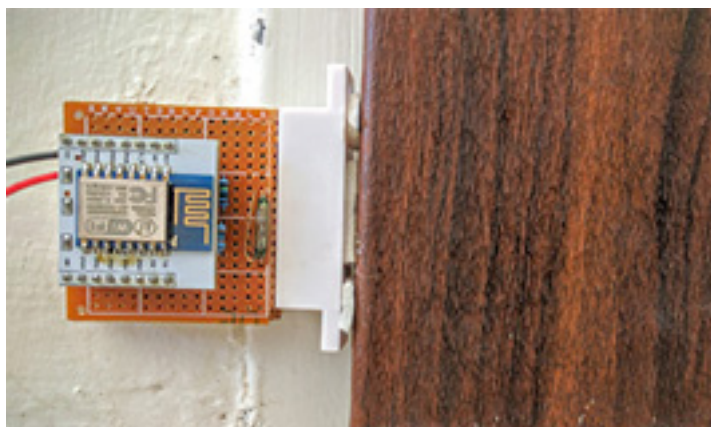


Figura 4 – Monitoramento (DIAMOND, 2016)

O autor (DIAMOND, 2016) propôs também, um sistema para monitorar sua casa, onde ele fez uso do ESP8266-12e juntamente com uma placa de adaptação, com o intuito de supervisionar o status de uma janela ou porta. Nesse projeto fez-se uso do aplicativo *Blynk* como interface para o usuário. O esquema físico montado pode ser visto na Figura 4. Pode-se notar o uso do *reed switch* como sendo o sensor de aberto/fechado, também nota-se que o sensor é energizado via cabos.

### 3.4 DIFERENCIAL TECNOLÓGICO

Neste projeto, a comunicação entre o Sensor e o Cliente será através de uma interface propriamente desenvolvida, com o intuito de facilitar o seu uso. Quando comparado ao produto desenvolvido pela empresa ECP, a grande diferença tecnológica é o uso do *Wi-Fi* ao invés de frequência de radio 433,92Mhz.

Comparando-se com o desenvolvido por (SANTOS, 2016), o presente projeto irá enviar dados para o usuário ao invés de apenas acender um *led* e, será utilizado o Microcontrolador ESP8266 ao contrário de fazer uso do ATmega (Arduino).

Em comparação com o Sensor de porta proposto por (DIAMOND, 2016), o presente projeto se destaca por ter um tamanho reduzido e sua modularização, onde a alimentação será por meio de uma bateria dispensando, assim, o uso de cabeamento.

O usuário terá facilidade em utilizar o *Smart Window* devido às suas características específicas, tendo apenas que colocar o sensor em sua janela ou porta. Após isso será necessário conectar-se ao sensor através de um *smartphone* para informar qual a rede *Wi-Fi* o sensor deverá se comunicar, para que seja possível o envio de informações para o *WebService*.

### 3.5 PROCEDIMENTOS METODOLÓGICOS/METODOLOGIA

Para o desenvolvimento deste projeto serão desenvolvidas as seguintes atividades:

- O projeto será iniciado com a programação em linguagem Lua fazendo uso do software *Explorer* para a integração do nodeMCU ESP8266 com o *Reed Switch*, onde assim será possível obter o status real em que uma janela/porta equipada com o *Smart Window* se encontra.
- Após o desenvolvimento do sensor serão realizados estudos e desenvolvimento do *WebService* REST e PHP, o qual será o nó<sup>1</sup> principal do sistema. O *WebService* vai intermediar as requisições feitas ao sensor e coletar de dados vindos dele.
- Para a exibição do status, o desenvolvimento de um interface básica torna-se necessária. Portanto, será produzida uma página web para apresentar o resultado obtido do sensor por intermédio do *WebService*. A página em questão, será produzida através do uso das linguagens de programação web (como por exemplo: HTML, CSS e PHP).
- Testes e análises serão feitas com o intuito de obter informações de cunho relevante nos quesitos: Bateria e Estabilidade. A análise da bateria deve-se ao fato da necessidade de saber quanto tempo o sensor irá ficar operante com uma bateria de 9v sendo ela alcalina ou não. E quanto a estabilidade, testes serão feitos para verificar se o reed switch está identificando perfeitamente o status e se o sistema está enviando estes dados corretamente para o *WebService*.
- Devido ao fato de uma janela/porta não possuir um padrão, a distância entre o sensor e o ímã que age sobre o *reed switch* pode variar, devido a isso, serão desempenhados testes físicos com diferentes tipos de ímãs para que, assim, utilize-se o mais eficiente possível.
- Após a realização de todos os testes acima descritos será feito um teste de usabilidade com uma pessoa que ainda não conhece a solução.

Na Figura 5 é apresentado um protótipo do funcionamento e montagem do sensor.

---

<sup>1</sup>O termo **nó** é utilizado para se referenciar a um dispositivo presente na rede.



Figura 5 – Protótipo

### 3.6 CONCLUSÃO/CONSIDERAÇÕES FINAIS

Com o desenvolvimento desse projeto, espera-se obter um sensor capaz de verificar se houve alguma alteração de status do objeto em que ele estará instalado. A fácil instalação e configuração do sensor é um dos fatores determinantes de seu sucesso. No início do desenvolvimento, as dificuldades esperadas são no desenvolvimento do *WebService* e na configuração de rede em que o usuário deverá realizar.

O forte apelo deste projeto está na segurança modular que o sensor poderá propor, onde, caso haja a necessidade de mudar de local, será fácil e suas funcionalidades permanecerão as mesmas, fornecendo assim, uma solução de baixo custo e excelentes benefícios.

## 3.7 PLANEJAMENTO DO TRABALHO

Quadro 2 – Cronograma de Atividades.

Atividades	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1. Revisão dos apontamentos da banca		X			X				X	
2. Revisão bibliográfica	X	X	X	X	X	X	X	X	X	
3. Redação do projeto de TCC			X	X						
4. Defesa do projeto de TCC					X					
5. Escrita da Monografia de TCC						X	X	X		
6. Desenvolvimento do Sensor						X	X	X		
7. Elaboração da apresentação final								X	X	
8. Defesa final do TCC									X	

## 3.8 RECURSOS NECESSÁRIOS

Para o desenvolvimento deste projeto serão necessários os seguintes recursos:

- 1 (um) Computador com o software *Esplorer* instalado
- 1 (um) nodeMCU ESP8266
- 1 (um) *reed switch*
- 1 (um) imã para acionar o *reed switch*
- 1 (uma) mini *protoboard*
- 1 (uma) bateria alcalina 9v

No Quadro 3 estão detalhados os custos do projeto proposto.

Quadro 3 – Custos do Sensor.

Produto	Custo (R\$)
nodeMCU ESP8266	R\$ 11,00
Reed Switch	R\$ 4,25
Imã para Reed Switch	R\$ 4,25
Mini Protoboard	R\$ 4,50
Bateria alcalina 9v	R\$ 13,25
Case de plástico	Indefinido
<b>Total</b>	<b>R\$ 37,25</b>

Todos os itens descritos acima já foram adquiridos onde, assim, não será necessário nenhum custeio da Universidade.

### 3.9 HORÁRIO DE TRABALHO

O horário destinado para realização das atividades do TCC, bem como o horário destinado para a reunião semanal/quinzenal com o orientador estão descritos no cronograma do Quadro 4. Este horário é definido com orientador levando em consideração a complexidade do trabalho a ser desenvolvido.

Quadro 4 – Horário de Trabalho.

<b>Horário</b>	<b>Seg</b>	<b>Ter</b>	<b>Qua</b>	<b>Qui</b>	<b>Sex</b>	<b>Sab</b>
07h30 - 08h20						
08h20 - 09h10						
09h10 - 10h00						
10h10 - 11h00						
11h00 - 11h50						
13h00 - 13h50						
13h50 - 14h40		TCC	TCC	TCC		
14h40 - 15h30		TCC	TCC	TCC		
15h40 - 16h30		TCC	TCC	TCC		
16h30 - 17h20						
17h20 - 18h10						
18h50 - 19h40			Orientação			
19h40 - 20h30			Orientação			
20h30 - 21h20						
21h30 - 22h15						

## 4 REFERÊNCIAS

EMBARCADOS. **Sistema Embarcado - O que é? Qual a sua importância?**. Disponível em <https://www.embarcados.com.br/sistema-embarcado/>. Acesso em 11 abr. 2017.

DOS SANTOS, Nuno Miguel Lopes. **Dispensador de Medicamentos**. Disponível em <https://fenix.tecnico.ulisboa.pt/downloadFile/563345090414517/dissertacao.pdf>. Acesso em: 23 mar. 2017.

MINATEL, Pedro. **Modos de economia de energia no ESP8266**. Disponível em <http://pedrominate.com.br/pt/esp8266/modos-de-economia-de-energia-no-esp8266/>. Acesso em: 9 mar. 2017.

LIMA, Irving Souza. **NodeMCU (ESP8266) o módulo que desbanca o Arduino e facilitará a Internet das Coisas. . . .** Disponível em: <http://irving.com.br/esp8266/nodemcu-esp8266-o-modulo-que-desbanca-o-arduino-e-facilitara-a-internet-das-coisas/>. Acesso em: 30 mar. 2017.

ELECTROFUN. **Wemos D1 R2 Placa Wi-Fi ESP8266**. Disponível em <https://www.electrofun.pt/wemos-d1-r2-wifi-esp8266>. Acesso em: 30 mar. 2017.

ADAFRUIT. **Adafruit HUZZAH ESP8266 Breakout**. Disponível em <https://www.adafruit.com/product/2471>. Acesso em: 30 mar. 2017.

USINAINFO. **Módulo Wifi ESP8266 Serial - ESP-201**. Disponível em <http://www.usinainfo.com.br/modulos-para-arduino/modulo-wifi-esp8266-serial-esp-201-4782.html>. Acesso em: 4 abr. 2017.

FILIPEFLOP. **Módulo WiFi ESP8266 ESP-201**. Disponível em <http://www.filipeflop.com/pd-2c1419-modulo-wifi-esp8266-esp-201.html>. Acesso em: 4 abr. 2017.

NEWTONCBRAGA. **Como funciona o Reed Switch (ART373)**. Disponível em <http://www.newtoncbraga.com.br/index.php/como-funciona/2462-art373>. Acesso em: 26 mar. 2017.

MAGALDI, Hélio Reis. **Alarmes O livro do instalador**. Disponível em <https://books.google.com.br/books?id=r4M4dJLn1ZYCpg=PA44lpg=PA44dq=livro+sobre+reed+switch-source=blots=gSqxorUghXsig=cLN9K6cudrAP7AN4dhxCKIopd4khl=pt-BRsa=Xved=0ahUK-Ewi9xPDn6PTSAhVJjpAKHTPiBXEQ6AEIRDAKv=onepageqf=false>. Acesso em: 26 mar. 2017.

CELES, Waldemar. DE FIGUEIREDO, Luiz Henrique. IERUSALIMSCHY, Roberto. **A Linguagem Lua e suas Aplicações em Jogos**. Disponível em <https://www.lua.org/doc/w-jogos04.pdf>. Acesso em: 29 mar. 2017.

LUA. **Quais as origens de Lua?**. Disponível em <https://www.lua.org/portugues.html>. Acesso em: 29 mar. 2017.

HARDWARE. **Lua - conhecendo um pouco sobre a linguagem de programação 100% brasileira**. Disponível em <http://www.hardware.com.br/comunidade/lua-linguagem/13->

04947/. Acesso em: 4 abr. 2017.

PASQUA, DOUGLAS V.. **Webservices REST e PHP, Introdução**. Disponível em <http://www.douglaspasqua.com/2015/11/20/webservices-rest-e-php-introducao/>. Acesso em: 5 abr. 2017.

CHEMAXON. Disponível em [https://www.chemaxon.com/app/themes/chemaxon/images/product\\_pages/jws/rest.jpg](https://www.chemaxon.com/app/themes/chemaxon/images/product_pages/jws/rest.jpg). Acesso em: 5 abr. 2017.

FISCHER, Ludovico. **Guia de introdução aos conceitos HTTP e REST**. Disponível em <https://code.tutsplus.com/pt/tutorials/a-beginners-guide-to-http-and-rest-net-16340>. Acesso em: 5 abr. 2017.

TUDOFORTE. **Sensor Magnético Sem Fio ECP Intruder Presença Portas e Janelas**. Disponível em <https://www.tudoforte.com.br/alarmes/sensores-magnetico/ecp-f105547-intruder-magnetico>. Acesso em: 17 mar. 2017.

SANTOS, Rui. **Monitor Your Door Using Magnetic Reed Switch and Arduino**. Disponível em <http://randomnerdtutorials.com/monitor-your-door-using-magnetic-reed-switch-and-arduino/>. Acesso em: 22 mar. 2017.

DIAMOND, Mike. **ESP 8266 Mailbox Notifier using DeepSleep and Blynk**. Disponível em <http://www.whatimade.today/esp-8266-mailbox-notifier-using-deepsleep-and-blynk/>. Acesso em: 29 mar. 2017.

DIAMOND, Mike. **Quick Effective Low Cost Home Protection with Raspberry Pi and ESP8266**. Disponível em <http://www.whatimade.today/quick-effective-low-cost-home-protection-with-raspberry-pi-and-esp8266/>. Acesso em: 5 abr. 2017.