

SMARTHOME: UTILIZANDO IOT EM CASA

Andressa Lúcia Machado

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP
Capivari, SP, Brasil

Resumo

O objetivo deste artigo demonstra o desenvolvimento de um sistema o qual prevê possibilitar mais autonomia e conforto aos moradores de uma residência em suas atividades cotidianas relacionadas a iluminação de sua casa. Utilizando o conceito de Internet das Coisas (*IoT*), um protótipo foi elaborado baseado em hardware e software livre a fim de simular sua atuação no sistema de iluminação de uma residência. Para o funcionamento do projeto foram utilizados: o Arduino e o *RaspberryPi*. O *RaspberryPi* faz a função de servidor *web* e ponte de conexão com o Arduino e o Arduino e seus componentes atenderão os comandos enviados pelo usuário através da internet. Além disso, foi desenvolvido um sistema *web* capaz de ser executado em qualquer navegador, inclusive em dispositivos móveis para acesso e controle da *Smarthome*.

Palavras-chave: *RaspberryPi*; Arduino; Automação; *OpenSource*; Iluminação.

SMARTHOME: USIGN IOT AT HOME

Abstract

The objective of this article demonstrates the development of a system which provides for greater autonomy and comfort for residents of a residence in their daily activities related to the lighting of their home. Using the Internet of Things (*IoT*) concept, a prototype was developed based on hardware and free software in order to simulate its performance in a home's lighting system. For the operation of the Project were used: Arduino and *RaspberryPi*. The *RaspberryPi* acts as a web server and connection bridge with the Arduino and the Arduino and their components will answer the commands sent by the user through the internet. In addition, a web system capable of running in any browser, including mobile devices for access and control by *Smarthome*, was developed.

Keywords: *RaspberryPi*; Arduino; Automation; *OpenSource*; Lighting.

1. INTRODUÇÃO

É comum vermos a interação das pessoas e agora, principalmente a interação de componentes eletrônicos de uma residência com a tecnologia. Diante desse cenário, a procura por facilitar o trabalho humano, gerando mais comodidade e conforto e, com isso, ganhando mais tempo para que ele seja dedicado na realização de outras tarefas é praticamente inevitável.

Do inglês *Internet of Things (IoT)*, a Internet das Coisas propõe a facilidade de fornecer um intermédio entre pessoas e os componentes interligados à internet. A utilização dessa tecnologia se dá tanto para pessoas como para empresas e é uma realidade que vem conquistando espaço no mundo inteiro.

Trazendo os conceitos de *IoT*, a *Smarthome* tem o objetivo de auxiliar o(s) morador(es) de uma residência em suas atividades cotidianas relacionadas a iluminação da casa. Com isso, o usuário tem maior comodidade e conforto: pois não irá se esforçar como antes para realizar uma determinada tarefa; acessibilidade: principalmente para usuários com limitações físicas, tornando a vida dos mesmos mais independente e com eficiência energética: pois o consumo energético dos componentes utilizados é baixo.

2. MATERIAIS E MÉTODOS

A proposta deste trabalho consiste em simular a atuação do sistema de iluminação de uma residência através de um protótipo. Uma aplicação foi desenvolvida com o intuito de controlar a iluminação de uma residência utilizando um servidor *web*. Através da aplicação, o morador enviar comandos para alteração do estado das lâmpadas. Dessa forma, o usuário pode realizar o acesso através de um computador ou dispositivo móvel com conexão à internet para acender ou apagar as luzes de sua casa.

Para tal, foi realizada inicialmente uma pesquisa bibliográfica e o levantamento dos materiais necessários, a fim de familiarizar com o tema específico, focalizando principalmente nos termos: *RaspberryPi*; Arduino; Aplicação, Servidor e Protocolo.

A seguir, estão descritos os materiais e detalhes da implementação.

2.1. Raspberry Pi

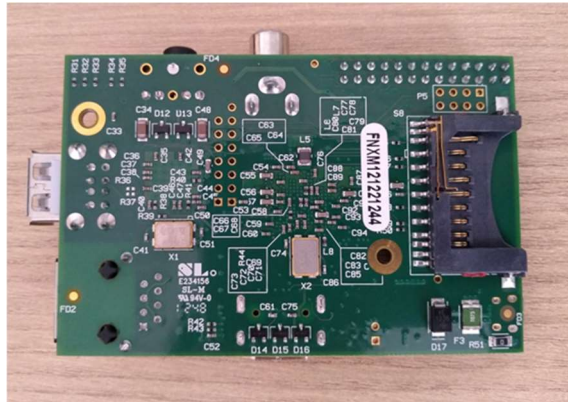
É um pequeno dispositivo que permite que pessoas de todas as idades explorem a computação e aprendam a programar em linguagens como Scratch e Python (RASPBERRY, 2015).

O *Raspberry Pi* utilizado é do modelo B (Figuras 1 e 2) e foram utilizados os seguintes recursos: uma entrada de cartão SD para armazenamento de dados (devido não possuir memória não volátil, como um disco rígido), uma porta Ethernet para conexão com a rede e duas portas USB para ligação de dispositivos para uso de teclado e mouse. Neste projeto, o *Raspberry Pi* é usado como servidor *web*.

Figura 1- Raspberry Pi (Parte Frontal)



Figura 2 - Raspberry Pi (Parte traseira)



Para uso do *Raspberry Pi* como servidor foi necessária a instalação de alguns pacotes, sendo eles: A IDE do Arduino, o Lighttpd e PHP. Além disso, houve a necessidade de dar as devidas permissões de leitura e gravação para que os dados enviados pelo Raspberry Pi pudessem trafegar para o Arduino e vice-versa. Para isso, o módulo FastCGI foi habilitado para o PHP e, após instalado, foi necessário reiniciar o serviço do Lighttpd.

2.2. Raspberry Pi OS

O *Raspberry Pi* necessita de um sistema operacional para funcionar. Para a realização do projeto foi necessário realizar o download e instalação do *Raspberry Pi OS*, anteriormente

chamado de Raspbian. O *Raspberry Pi OS* é um sistema operacional *open source* baseado no Debian otimizado para uso no *Raspberry Pi*.

Para sua instalação foi necessário fazer o download de um arquivo com a imagem do sistema operacional. O sistema utilizado neste projeto foi o *Raspberry Pi OS with desktop and recommended software*: Essa versão possui softwares que foram utilizados no projeto, além de ter interface gráfica.

2.3. Arduino Uno

Arduino é uma plataforma eletrônica *open source* tanto para *hardware* quanto para *software*. Em termos práticos, um Arduino é um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele (MCROBERTS, 2010).

O Arduino Uno foi desenvolvido e baseado no microcontrolador ATmega328P, possui 14 portas de entrada e saída, das quais 6 podem ser utilizadas como saída PWM, 6 entradas analógicas, um pequeno ressoador cerâmico de 16 MHz, uma conexão USB, um conector de alimentação, uma conexão ICSP, e um botão de reset. (ARDUINO, 2013).

Dentre as versões disponíveis, o Arduino Uno foi utilizado pois, é a placa encontrada com mais facilidade e a quantidade de entradas e saídas contemplam a necessidade do projeto. A Figura 3 demonstra a versão utilizada.

Figura 3 – Arduino Uno



O Arduino foi conectado no *Raspberry Pi* que é de onde virão os comandos enviados pela *web* para realização das funções desejadas pelo usuário, bem como é por onde ele pode ser programado. Para isso, o código com as funções necessárias foi gravado no Arduino.

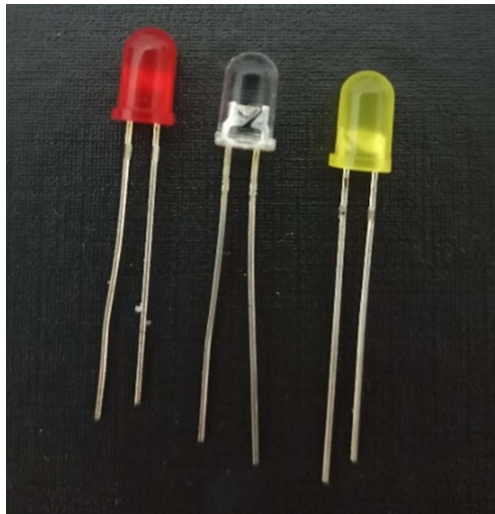
2.4. Sensores

Termo empregado para designar dispositivos sensíveis à alguma forma de energia do ambiente que pode ser luminosa, térmica, cinética, relacionando informações sobre uma grandeza física que precisa ser mensurada (medida), como: temperatura, pressão, velocidade, corrente, aceleração, posição etc. (WENDLING, 2010).

2.4.1.LED

O *Light Emitter Diode* ou LED (Em português Diodo emissor de luz) (Figura 4) tem a propriedade de transformar energia elétrica em luz.

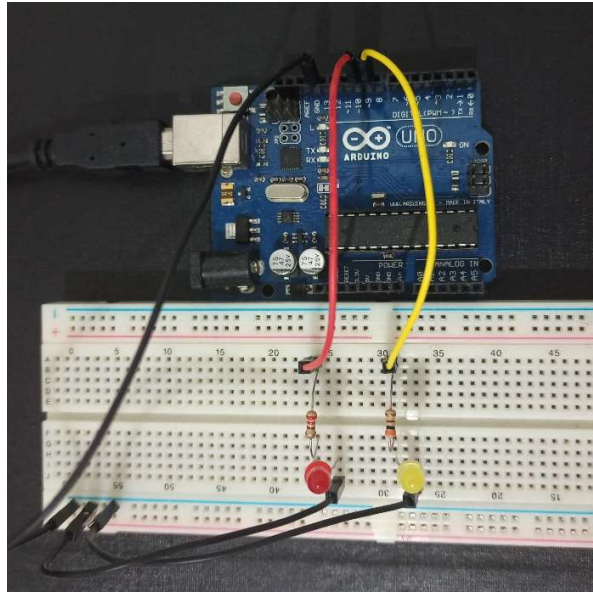
Figura 4 – LED



De acordo com CRUZ, et al. (2016) “O LED é um componente do tipo bipolar, ou seja, tem um terminal chamado anodo e outro, chamado catodo. Dependendo de como for polarizado, permite ou não a passagem de corrente elétrica e, conseqüentemente, a geração ou não de luz.”

A utilização desse sensor tem o objetivo de demonstrar em um protótipo o comportamento de uma lâmpada ao receber o comando enviado pelo usuário. Os LEDs foram encaixados em uma *protoboard*, juntamente com um resistor para cada LED, conforme demonstrado na Figura 5.

Figura 5 – Conexões dos LEDs e resistores no Arduino



2.5. Aplicação

A proposta da aplicação *web* é de facilitar o acesso do morador não o condicionando ao download de um aplicativo em seu dispositivo. Com isso, o usuário irá precisar apenas ter um dispositivo com acesso à internet que possua um navegador para conseguir acessar e enviar comandos para as lâmpadas de sua casa. Considerando a proposta inicial, a aplicação *web* foi desenvolvida com HTML, Sass (CSS) e PHP.

2.5.1.HTML

HTML é uma abreviação de *Hypertext Markup Language* - Linguagem de Marcação de Hipertexto. Resumindo em uma frase: o HTML é uma linguagem para publicação de conteúdo (texto, imagem, vídeo, áudio etc.) na *web* (FERREIRA, 2013).

Neste trabalho o HTML foi utilizado para a criação das páginas que o usuário interage, enviando comandos para as lâmpadas de sua residência através dos botões disponíveis em cada cômodo da casa. No protótipo temos: quarto e sala.

2.5.2.CSS

Utilizado para estilizar todas as páginas da aplicação *web*, o CSS foi gerado a partir de arquivos utilizando o SASS.

Sua maior finalidade, é separar o estilo da página, de seu conteúdo, deixando assim, o código mais limpo e facilitando a sua manutenção e escrita (MURPHY et al., 2012). O termo cascata indica que é possível usar vários documentos CSS para compor o visual da página.

2.5.3.SASS

O SASS É um pré-processador de CSS, ele permite o uso de variáveis regras, funções e muito mais. Com o uso do SASS é possível manter grandes folhas de estilo bem organizadas e com compatibilidade com vários navegadores. Sua sigla em inglês corresponde a “*Syntactically Awesome Style Sheets*” que significa, em uma tradução livre, Folhas de Estilo Sintaticamente Impressionantes. A sintaxe do SASS utilizada será a formatação em bloco, como a do CSS. Seu arquivo é editado e salvo com a extensão .scss.

O SASS foi utilizado no projeto justamente para possibilitar a organização da documentação, a utilização de variáveis e funções com o intuito de diminuir, otimizar as linhas de código e para a geração de estilos compatíveis com os mais diversos navegadores.

2.5.4.PHP

O PHP (*Hypertext Preprocessor*) é uma linguagem de script *open source* de uso geral, adequada para o desenvolvimento *web* e que pode ser embutida dentro do HTML.

Neste trabalho o PHP foi utilizado para realizar a comunicação da página utilizada com o *Raspberry Pi* e com o próprio usuário, fazendo com que ele interaja com os componentes a fim de controlar as luzes de sua residência.

Além disso o projeto utilizara uma biblioteca PHP responsável por realizar a comunicação serial da página *web* com o Arduino. (PHP Serial, 2014).

2.6. Servidor

Para hospedagem da aplicação *web*, há a necessidade de um servidor, para que o acesso seja realizado por todos os clientes através do protocolo HTTP (*Hypertext Transfer Protocol*).

Segundo Dalziel (2015) “a função principal de um servidor *web* é a de entregar documentos aos computadores e dispositivos clientes. Estes documentos usualmente consistem em páginas HTML, que podem incluir imagens, folhas de estilo e scripts, além do seu conteúdo texto.”

Com o intuito de não utilizar muita memória e garantir que o uso ocorra de forma estável e segura, foi instalado o pacote *Lighttpd* para que o *Raspberry Pi* pudesse ser utilizado como servidor para arquivos PHP.

2.7. Protocolo

Basicamente, um protocolo é um acordo entre as partes que se comunicam, estabelecendo como se dará a comunicação (TANENBAUM, 2003).

2.7.1.CGI

O CGI ou *Common Gateway Interface* é um protocolo de comunicação com a Internet, ou seja, é um padrão de interface para aplicações externas através da troca de dados com servidores de informações, como HTTP e servidores *web*. É esse padrão que torna a página dinâmica, possibilitando a execução de informações em tempo real.

A CGI é uma interface padronizada para permitir que servidores da *web* se comuniquem com programas e scripts de *back end* que possam aceitar a entrada (por exemplo, de formulários) e gerar páginas HTML em resposta (TANENBAUM, 2003).

O *Lighttpd* foi utilizado justamente pelo fato de possuir algumas opções avançadas, como por exemplo o CGI que foi utilizado no projeto.

2.7.2.HTTP

O *Hypertext Transfer Protocol* é um protocolo de transferência utilizado na *World Wide Web* (WWW) que possibilita que as pessoas com o acesso à determinada URL na *web* possam ver os conteúdos e dados que nele existem. Quando um navegador deseja uma página da *web*, ele envia o nome da página desejada ao servidor, utilizando o HTTP. Então, o servidor transmite a página de volta (TANENBAUM, 2003).

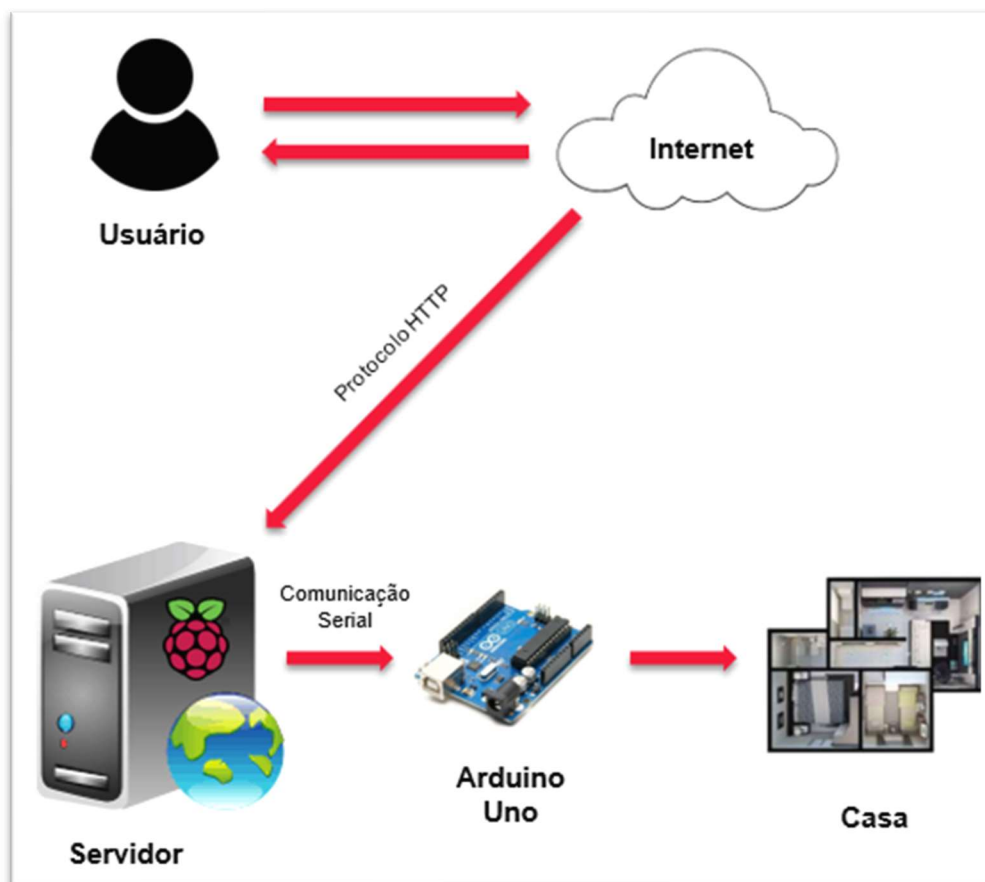
A comunicação do protocolo HTTP é baseada no modelo cliente-servidor, que significa que a comunicação é realizada através de requisição e resposta. As mensagens enviadas pelo cliente, geralmente um navegador da *web*, são chamadas de solicitações (*requests*), ou também requisições, e as mensagens enviadas pelo servidor como resposta são chamadas de respostas (*responses*) (MOZILLA, 2020).

O protocolo HTTP é essencial, visto que através dele será possível o acesso à URL do projeto, possibilitando a visualização dos conteúdos disponíveis na aplicação *web*.

3. RESULTADOS E DISCUSSÃO

O projeto é constituído de uma aplicação *web* hospedada em um minicomputador *Raspberry Pi* que pode ser acessado pela internet e envia comandos para o Arduino, que por sua vez se comunica com os itens a ele conectados. A Figura 6 demonstra o esquema de funcionamento do projeto.

Figura 6 – Esquema de funcionamento do projeto

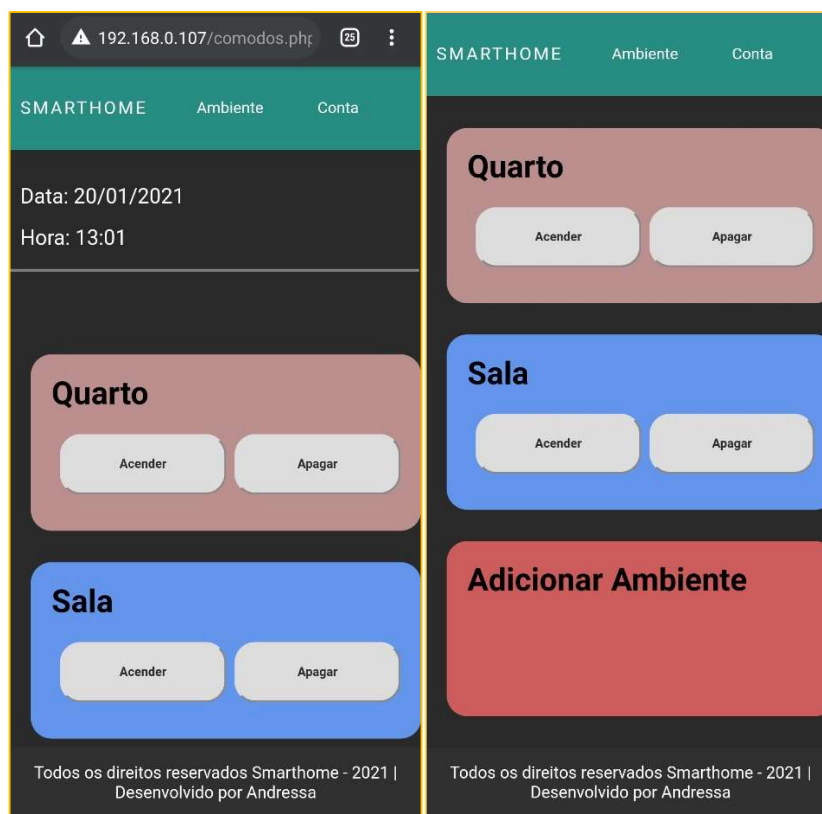


A aplicação possui a tela de controle conforme demonstram as figuras 7 e 8 que é por onde o morador envia comandos para acender e apagar as luzes de sua casa.

Figura 7 – Aplicação em um desktop



Figura 8 – Aplicação em um celular



Colocado o projeto em prática, foi possível perceber que a utilização do mesmo está de acordo com o objetivo: proporcionar independência do morador em relação ao controle da iluminação de sua residência de forma remota.

Apesar de ser um sistema simples, a utilização mostrou funcional e prática, em que os objetivos propostos foram concluídos com êxito, possibilitando que pessoas com escasso conhecimento em informática consiga utilizá-lo sem dificuldades.

O código deste trabalho está disponível na página de compartilhamento de código colaborativo *Github* (<https://github.com/andressalmachado/smarthome>) com o intuito de receber melhorias por quem se interessa pelos assuntos abordados e querem colaborar com o projeto ou para eventuais consultas.

4. CONCLUSÃO

Este projeto demonstrou a integração de *hardwares* e *softwares* livres e que, algo de simples montagem e manutenção pode facilitar a vida de muitas pessoas dentro de suas residências, utilizando os conceitos de *IoT*.

Os testes realizados ocorreram de forma satisfatória, todas as implementações almejadas foram alcançadas e funcionaram conforme o planejado. O resultado da validação foi gravado e

disponibilizado no Youtube e pode ser acessado através do seguinte link: <https://youtu.be/nueQBY8rlKE>.

Um determinante para o sucesso desse projeto foi a integração dos conhecimentos adquiridos ao longo do curso e através de pesquisas realizadas na internet para as áreas inicialmente não conhecidas. Com isso, conclui-se que os objetivos do trabalho foram alcançados e a implementação do projeto foi realizada com êxito, obtendo assim uma *Smarthome*, possibilitando aos moradores o uso de *IoT* em casa.

REFERÊNCIAS

- ARDUINO. **Arduino – Introduction**. 2013. Disponível em: <https://www.arduino.cc/en/guide/introduction>. Acesso em 11 dez. 2020.
- CRUZ, H. *et al.* **Projeto de Automação Residencial Utilizando Softwares e Hardwares Livres**. Orientador: Professor Marcos Moulin Valencia. 2016. 85 p. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Controle e Automação) - Instituto Federal de Educação, Ciência e Tecnologia Fluminense. 2016. Disponível em: <http://bd.centro.iff.edu.br/bitstream/123456789/1163/1/Texto.pdf>. Acesso em 27 nov. 2020.
- DALZIEL, H. **How to Attack and Defend Your Website**. 1. ed. Massachusetts: Syngress, 2015. 78 p.
- FERREIRA, S. **Guia prático de HTML5**. 1. ed. São Paulo: Universo dos Livros, 2013. 168 p.
- MCROBERTS, M. **Beginning Arduino**. 2.ed. Nova York: Novatec Editora, 2010.
- MOZILLA. **Uma visão geral do HTTP**. 2020. Disponível em: <https://developer.mozilla.org/pt-BR/>. Acesso em 27 nov. 2020.
- MURPHY, C. *et al.* **Beginning HTML5 and CSS3: The Web Evolved**. Nova York: Apress, 2012. 624 p. Disponível em: <http://books.google.com.br/books?id=IP29IJbC6ooC>.
- PHP SERIAL. CODIGO. Disponível em: <https://github.com/Xowap/PHP-Serial>. Acesso em: 18 dez. 2020.
- RASPBERRYPI. **What is a Raspberry Pi?** 2015. Disponível em: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>. Acesso em 11 dez. 2020.
- TANENBAUM, A. S. **Redes de Computadores**. 4. ed. Rio de Janeiro: Editora Campus (Elsevier), 2003. 600 p.
- WENDLING, M. **Sensores**. 2010. Disponível em: <https://www.feg.unesp.br/Home/PaginasPessoais/ProfMarceloWendling/4---sensores-v2.0.pdf>. Acesso em 27 nov. 2020.