



Universidade Presbiteriana Mackenzie

Objetos Inteligentes Conectados

Automação Residencial

Nome: Bernard Prado

TIA: 31693210

Nome: Gabriel de Carvalho

TIA: 314331550

Nome: Isadora Prado

TIA: 31610277

Turma: 5K12

São Paulo

Maio/2018

1. Automação residencial

a. Definição

Este documento apresenta uma solução de software para o projeto de Automação residencial, apresentando os problemas a serem solucionados, as necessidades dos principais envolvidos, o alcance do projeto e as funcionalidades esperadas.

O projeto “Automação residencial” tem a finalidade de facilitar o usuário em sua residência. Nosso foco é dar a possibilidade do usuário poder controlar as luzes de sua residência via desktop/mobile em qualquer lugar que se encontre.

b. Tendência e influências

A maior influência para esse projeto é o futuro, onde nossa visão é dar nas mãos do usuário a possibilidade de controlar sua casa com apenas alguns cliques

c. Objetivo

Controlar todas as lâmpadas, ventiladores, etc, de todos os cômodos de uma casa

d. Problema

Descrição do problema: Muitas vezes, as pessoas esquecem de apagar as luzes da sua casa. Excesso de iluminação causa diversos problemas de saúde. Pode alterar os ritmos circadianos, causando problemas de sono. Além disso, a disfunção dos ciclos circadianos também pode estar associada a problemas como depressão e câncer. O efeito atrativo que a luz exerce sobre os insetos também tem consequências sanitárias. Insetos vetores de doenças tais como malária, leishmaniose e mal de Chagas podem ser atraídos por luzes, aproximando-se de populações humanas. Além disso, a própria iluminação aumenta a atividade das pessoas em locais externos durante a noite, aumentando sua exposição a esses insetos.

- Quem é afetado pelo problema: Vida humana e custo elevados de energia.
- Impacto no negócio: Não há impacto.
- Benefícios de uma boa solução: Evita problemas de saúde e economia de dinheiro.

2. Componentes de hardwares e matérias para confecção do projeto

a. Componentes de hardwares utilizados neste projeto

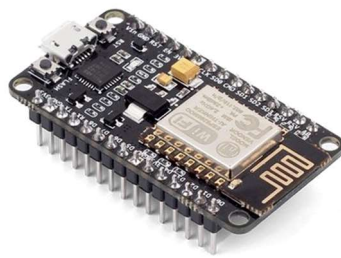
- Lâmpada



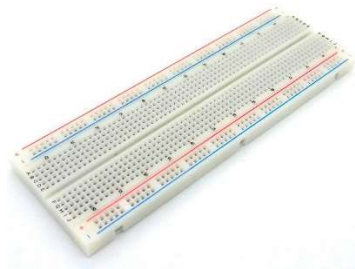
- Socket para lâmpada



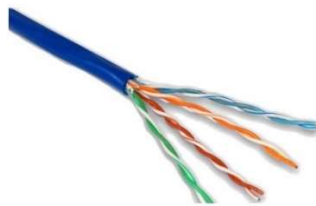
- Nodemcu



- Proboard



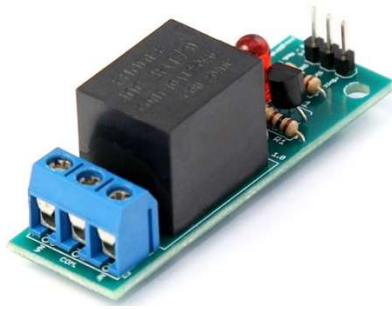
- Cabo



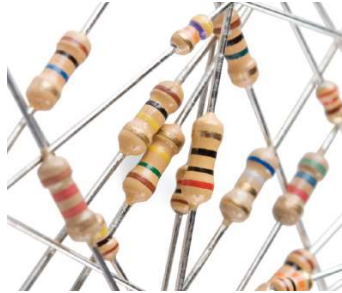
- Jumpers



- Relê



- Resistores

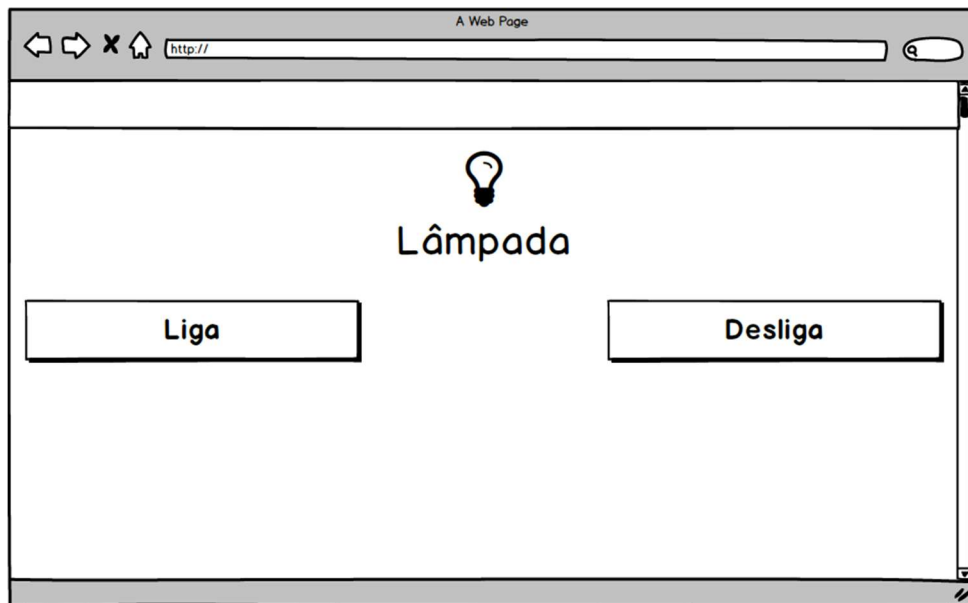


- Casa de madeira

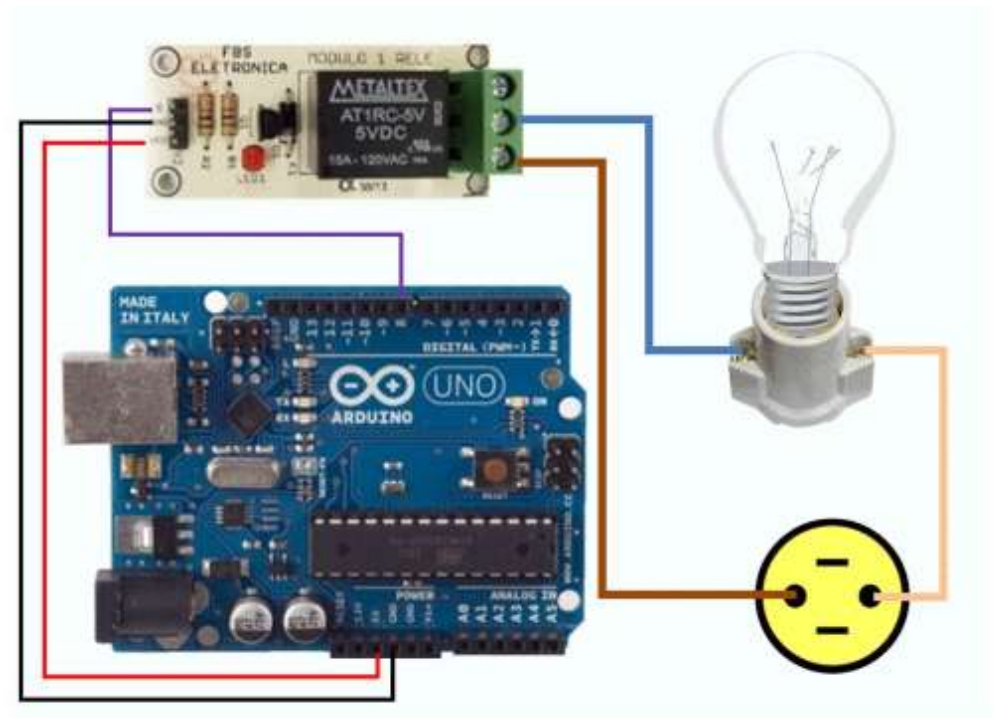


3. Wireframe da interface do usuário

- Botão Liga: liga a lâmpada da residência
- Botão Desliga: desliga a lâmpada da residência



4. Circuito



Código do projeto

```
#include <ESP8266WiFi.h>
```

```
//Nome da sua rede Wifi
```

```
const char* ssid = "iPhone de Isadora helena";
```

```

//Senha da rede
const char* password = "isa12345";

//IP do ESP (para voce acessar pelo browser - voce TEM que mudar este IP tambem)
IPAddress ip(172, 20, 10, 4);

//IP do roteador da sua rede wifi
IPAddress gateway(172, 20, 10, 1);

//Mascara de rede da sua rede wifi
IPAddress subnet(255, 255, 255, 240);

//Criando o servidor web na porta 80
WiFiServer server(80);

```

/*

Equivalencia das saidas Digitais entre nodeMCU e ESP8266 (na IDE do Arduino)

NodeMCU ↔ ESP8266

D0 = 16;

D1 = 5;

D2 = 4;

D3 = 0;

D4 = 2;

D5 = 14;

D6 = 12;

D7 = 13;

D8 = 15;

D9 = 3;

D10 = 1;

*/

```

//Pino do NodeMCU que estara conectado ao rele
const int pin = 4; //Equivalente ao D2 no NodeMCU

//Funcao que sera executada apenas ao ligar o ESP8266

void setup() {

//Preparando o pino, que esta lidado ao rele

pinMode(pin, OUTPUT);

digitalWrite(pin, HIGH);


//Conectando a rede Wifi

WiFi.config(ip, gateway, subnet);

WiFi.begin(ssid, password);


//Verificando se esta conectado,
//caso contrario, espera um pouco e verifica de novo.

while (WiFi.status() != WL_CONNECTED) {

    delay(500);

    }


//Iniciando o servidor Web

server.begin();

}


//Funcao que sera executada indefinidamente enquanto o NodeMCU estiver ligado.

void loop() {

//Verificando se o servidor esta pronto.

WiFiClient client = server.available();

if (!client) {

    return;

```

```
}
```

```
//Verificando se o servidor recebeu alguma requisicao
```

```
while (!client.available()) {
```

```
    delay(1);
```

```
}
```

```
//Obtendo a requisicao vinda do browser
```

```
String req = client.readStringUntil('\r');
```

```
//Sugestao dada por Enrico Orlando
```

```
if(req == "GET /favicon.ico HTTP/1.1"){
```

```
    req = client.readStringUntil('\r');
```

```
}
```

```
client.flush();
```

```
//Iniciando o buffer que ira conter a pagina HTML que sera enviada para o browser.
```

```
String buf = "";
```

```
buf += "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n<!DOCTYPE  
HTML>\r\n<html>\r\n";
```

```
buf += "<head> ";
```

```
buf += "<meta charset='UTF-8'> ";
```

```
buf += "<meta http-equiv='cache-control' content='max-age=0' /> ";
```

```
buf += "<meta http-equiv='cache-control' content='no-cache' /> ";
```

```
buf += "<meta http-equiv='expires' content='0' /> ";
```

```
buf += "<meta http-equiv='expires' content='Tue, 01 Jan 1980 1:00:00 GMT' /> ";
```

```
buf += "<meta http-equiv='pragma' content='no-cache' /> ";
```

```
buf += "<title>Automa&ccedil;&atilde;o Residencial</title> ";
```

```
buf += "<style> ";
```



```

        buf += "body{font-family:Open Sans; color:#555555;} ";

        buf += "h1{font-size:24px; font-weight:normal; margin:0.4em 0;} ";

        buf += ".container { width: 100%; margin: 0 auto; } ";

        buf += ".container .row { float: left; clear: both; width: 100%; } ";

        buf += ".container .col { float: left; margin: 0 0 1.2em; padding-right: 1.2em; padding-left: 1.2em; } ";

        buf += ".container .col.four, .container .col.twelve { width: 100%; } ";

        buf += "@media screen and (min-width: 767px) { ";

        buf += ".container{width: 100%; max-width: 1080px; margin: 0 auto;} ";

        buf += ".container .row{width:100%; float:left; clear:both;} ";

        buf += ".container .col{float: left; margin: 0 0 1em; padding-right: .5em; padding-left: .5em;} ";

        buf += ".container .col.four { width: 50%; } ";

        buf += ".container .col.tweleve { width: 100%; } ";

        buf += " ";

        buf += " * {-moz-box-sizing: border-box; -webkit-box-sizing: border-box; box-sizing: border-box;} ";

        buf += "a{text-decoration:none;} ";

        buf += ".btn {font-size: 18px; white-space:nowrap; width:100%; padding:.8em 1.5em; font-family: Open Sans, Helvetica,Arial,sans-serif; ";

        buf += "line-height:18px; display: inline-block;zoom: 1; color: #fff; text-align: center; position:relative; ";

        buf += "-webkit-transition: border .25s linear, color .25s linear, background-color .25s linear; ";

        buf += "transition: border .25s linear, color .25s linear, background-color .25s linear;} ";

        buf += ".btn.btn-sea{background-color: #08bc9a; border-color: #08bc9a; -webkit-box-shadow: 0 3px 0 #088d74; box-shadow: 0 3px 0 #088d74;} ";

        buf += ".btn.btn-sea:hover{background-color:#01a183;} ";

        buf += ".btn.btn-sea:active{ top: 3px; outline: none; -webkit-box-shadow: none; box-shadow: none;} ";

        buf += "</style> ";

        buf += "</head> ";

        buf += "<body> ";

        buf += "<div class='container'> ";

        buf += "<div class='row'> ";

```

```

        buf += "<div class='col twelve'> ";
buf += "<p align='center'><font size='10'>Controle de L&acirc;mpadas</font></p> ";
        buf += "</div> ";
        buf += "</div> ";
        buf += "<div class='row'> ";
        buf += "<div class='col four'> ";
        buf += "<a href='?f=on' class='btn btn-sea'>Ligar</a> ";
        buf += "</div> ";
        buf += "<div class='col four'> ";
        buf += "<a href='?f=off' class='btn btn-sea'>Desligar</a> ";
        buf += "</div> ";
        buf += "</div> ";
        buf += "<div class='col twelve'> ";
buf += "<p align='center'><font size='5'>Projeto Objetos Inteligentes Conectados</font></p>
";
        buf += "</div> ";
        buf += "</div> ";
        buf += "</body> ";
        buf += "</html> ";

```

//Enviando para o browser a 'pagina' criada.

```

client.print(buf);
client.flush();

```

//Analizando a requisicao recebida para decidir se liga ou desliga a lampada

```

if (req.indexOf("on") != -1)
{
    digitalWrite(pin, LOW);
}
else if (req.indexOf("off") != -1)
{

```

```
digitalWrite(pin, HIGH);  
    }  
    else  
    {  
        //Requisicao invalida!  
        client.stop();  
    }  
}
```