



Temperatura e Iluminação por Movimento: Conforto e Segurança

Michele Teixeira, José Marcos Pereira, Prof. Wilian França Costa

¹Universidade Presbiteriana Mackenzie (UPM)
Rua da Consolação, 930 Consolação, São Paulo - SP, 01302-907 – Brazil

michele.queiroz@mackenzista.com.br, 10918501011@mackenzista.com.br

Abstract. *This article describes a temperature and lighting control system using a motion sensor controlled via a smartphone. The temperature sensor will activate the fan / exhaust fan and a light when movement is detected in the room. It will be possible to program through the smartphone which will be the temperature that will activate these devices and which temperature will turn them off.*

Resumo. *Este artigo descreve um sistema de controle de temperatura e iluminação por sensor de movimento controlado via smartphone. O sensor de temperatura irá acionar o ventilador/exaustor e uma luz quando for detectada algum movimento no local. Será possível programar pelo smartphone qual será a temperatura que irá acionar esses dispositivos e qual temperatura que irá desligá-los.*

Introdução

Automação residencial com ventilador e luz, ativadas por movimento não é algo novo. Hoje em dia podemos encontrar muitas empresas que oferecem este serviço. Esta automação tornou-se nos dias hoje um hábito, vemos muito disso quando entramos em uma garagem, no hall de um estabelecimento e muito mais. A (Smarthouse, 2021), umas das empresas que pesquisamos, oferece essa automação para casa inteira e controlado por um único sistema.

Existem grupos de pesquisa que estão desenvolvendo projetos nesta área como a Philips através do projeto HomeLab, para melhorar a experiência humana com o ambiente, tornando as casas mais inteligentes. A Microsoft desenvolve o projeto EasyLiving que implementa um ambiente inteligente capaz de monitorar os moradores da residência. (Silva de Freitas,, Larico Chávez, , Higa, & Iano, 2012, p. 42)

As vantagens que são oferecidas com essa automação, no nosso caso somente de ventilador e luz, é a praticidade e comodidade. Entrando numa sala ou quarto imediatamente se acende a luz e o ventilador é ligado. Outra vantagem importante é a economia de energia elétrica. Com o sistema integrado a um dispositivo de presença, os equipamentos podem não ser acionados ocasionando em economia elétrica e reduzindo desgastes.

2. Materiais e Métodos

Apresentação Geral

2.1 Componentes utilizados para o projeto

- Arduino UNO genérico placa de interface baseada em microcontrolador;
- Sensor de vibração
- Relês para acionamento elétrico
- Sensor de temperatura
- Ventilador
- Luminária
- Smartphone de controle e monitoramento
- Computador
- Protoboard
- Fios de conexão

Para este projeto serão usados os equipamentos e periféricos conforme listado com suas respectivas funções:

Uma placa controladora ARDUINO UNO para interface do sistema com os componentes e equipamentos periféricos. O sistema usará seus sensores para que, conforme configurados, eles executem ou não ações.



Imagem 1. Placa controladora ARDUINO UNO.

Duas placas de circuito “protoboard”, para a montagem de protótipo e testes.

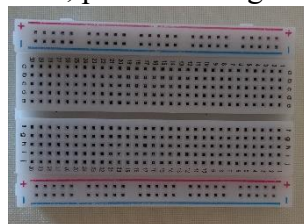


Imagem 2. Placa de circuito “protoboard”

Dois relês eletrônicos para controle dos equipamentos periféricos, estes relês são capazes de, ao receber uma instrução eletrônica, ligar ou desligar um equipamento elétrico. No projeto controlarão o ventilador e a luz do ambiente.



Imagem 3. Relê eletrônico

Estes componentes serão conectados e alimentados ao sistema conforme o seguinte diagrama.

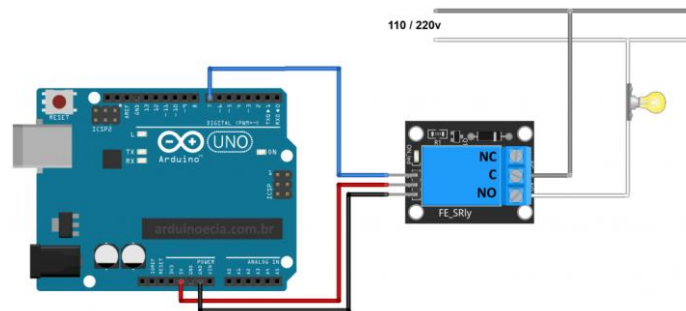


Figura 1. Diagrama eletrônico para os relês. (Arduinoecia, 2013)

Um sensor de movimento e vibração para monitoramento. Este sensor transforma a recepção de energia mecânica em impulsos eletrônicos, ele é capaz de detectar movimentos, vibrações e medir sua intensidade. Isso torna possível, através de parâmetros programados, ignorar falsos disparos.



Imagem 4. Sensor de movimento e vibração com regulação de intensidade. (USINAINFO, 2021)

Estes componentes serão conectados e alimentados ao sistema conforme o seguinte diagrama.

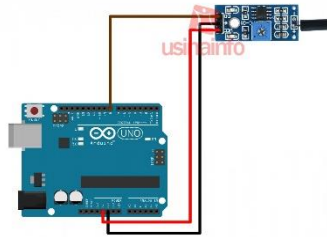


Figura 2. Diagrama eletrônico para o sensor de vibração.

Um sensor de luminosidade para monitoramento. Este sensor será empregado para detectar a presença e a intensidade da iluminação no ambiente, permitindo ou não o acionamento da luz de segurança ao detectar movimentação.

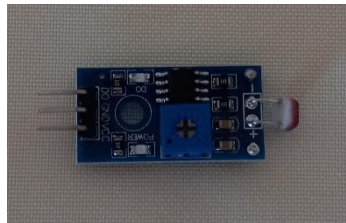


Imagem 5. Sensor de iluminação com controle de intensidade. (Usinainfo, 2021)

Estes componentes serão conectados e alimentados ao sistema conforme o seguinte diagrama.

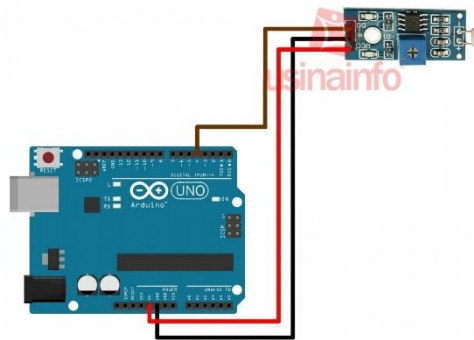


Figura 3. Diagrama eletrônico para o sensor de luz.

Um sensor de temperatura para monitoramento. Este sensor transforma energia de temperatura em sinais eletrônicos, isso torna possível que o sistema leia e interprete a temperatura no ambiente, acionando ou desligando o ventilador.

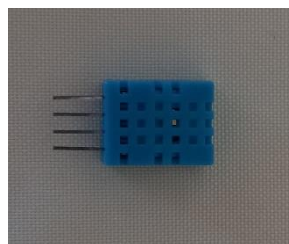


Imagem 6. Sensor de temperatura

Estes componentes serão conectados e alimentados ao sistema conforme o seguinte diagrama.

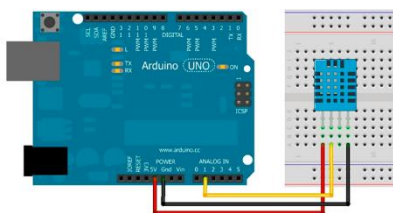


Figura 4. Diagrama eletrônico para o sensor de temperatura. (Filipeflop, 2013)

Nesse projeto quando a temperatura no ambiente superar a temperatura programada, o sistema acionará o ventilador pelo relê 1. Esta temperatura poderá ser programada através de aplicativo no celular pela internet e quando a luminosidade será acionada quando o ambiente estiver escuro.

Caso o sistema detecte um movimento, mas o sensor de luz acusar luminosidade mínima prévia, o sistema não aciona a luz, relê 2, com a detecção do movimento. Os parâmetros, luminosidade mínima e intensidade da detecção de movimento, serão ajustados por aplicativo de celular pela internet.

O sistema demonstrará por painel de controle, no celular, as variações de temperatura do ambiente e os movimentos detectados. Além das ações disparadas de acionamento e desligamento da luz e do ventilador.

2.2 Modelo de Montagem do Projeto

No projeto existem dois sensores para analisar o ambiente um sensor que capta temperatura no ambiente e outro que capta a luminosidade do ambiente e de acordo com a luminosidade ascende ou não a luz da luminária através de um relê. O sensor de temperatura quando atinge uma determinada temperatura ele aciona o ventilador se através de um relê, se a temperatura cair ele desliga o ventilador, sendo assim são dois circuitos um para iluminar o ambiente através de sensor de movimento e outro de temperatura para ligar o ventilador de acordo com a temperatura do ambiente. Esse parâmetro da temperatura será ajustado através do aplicativo no celular, exemplo se a temperatura chegar a 22° liga o ventilador, se for menor que 22° desliga o ventilador.

Todos os sensores e atuadores ficam ligados nas portas digitais do Arduino.

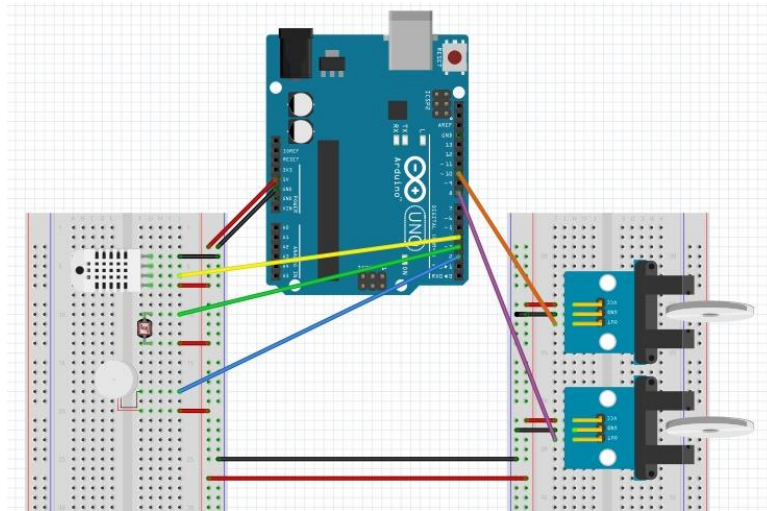


Figura 2.2 Modelo de Montagem de Projeto.

2.3 Software – Descrição do módulo de comunicação

Seguiremos o seguinte método para este projeto. Os sensores de luz, temperatura e vibração fornecerão ao sistema as informações sobre o ambiente. Cada sensor enviará suas leituras por porta MQTT específica, e estes sinais serão interpretados para o acionamento e desligamentos dos equipamentos periféricos através dos relês.

Precisamos carregar o sketch StandardFirmata no software do Arduino. Esse sketch vem como exemplo quando você instala o ambiente do Arduino, portanto podemos carregá-lo utilizando o menu arquivo – exemplo – firmata – StandardFirmata, como mostrado na figura1-1.

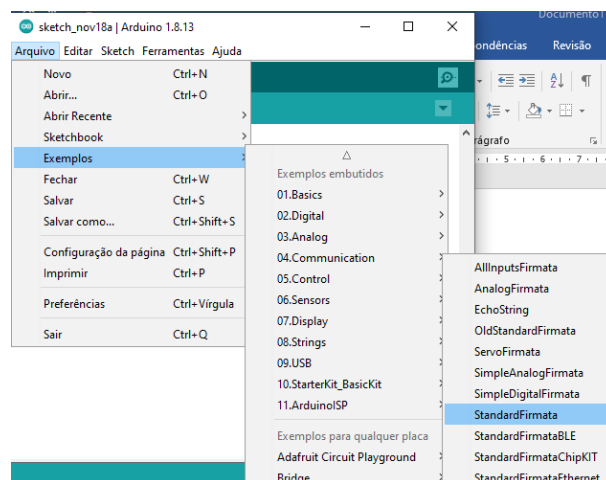


Figura1.1 sketch StandardFirmata

2.3.1 Programação dos sensores no Arduino

Para a programação dos sensores no Arduino foi usado o node red e programação no sketch do Arduino.

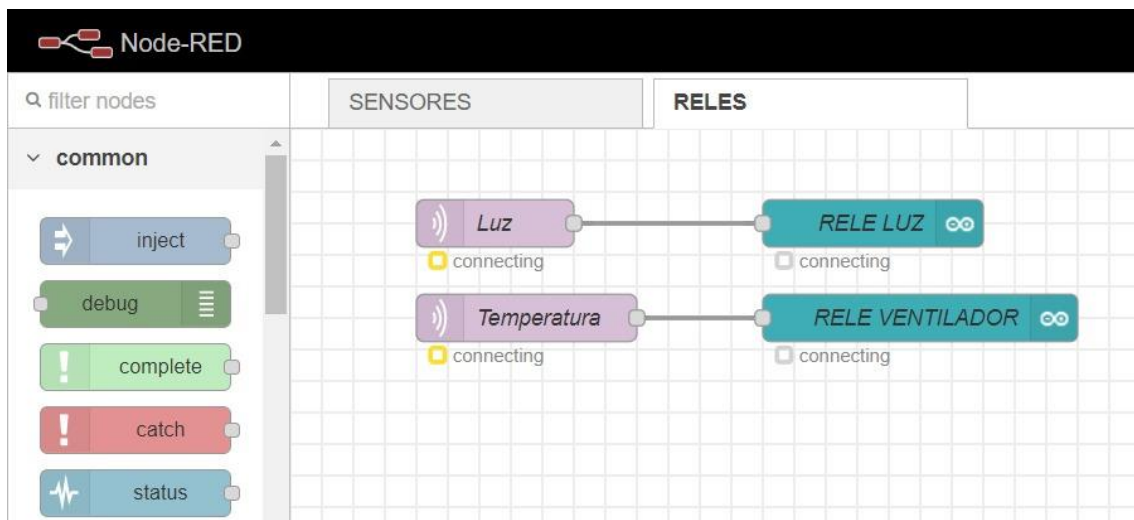


Figura1 Fluxo Node Red Reles

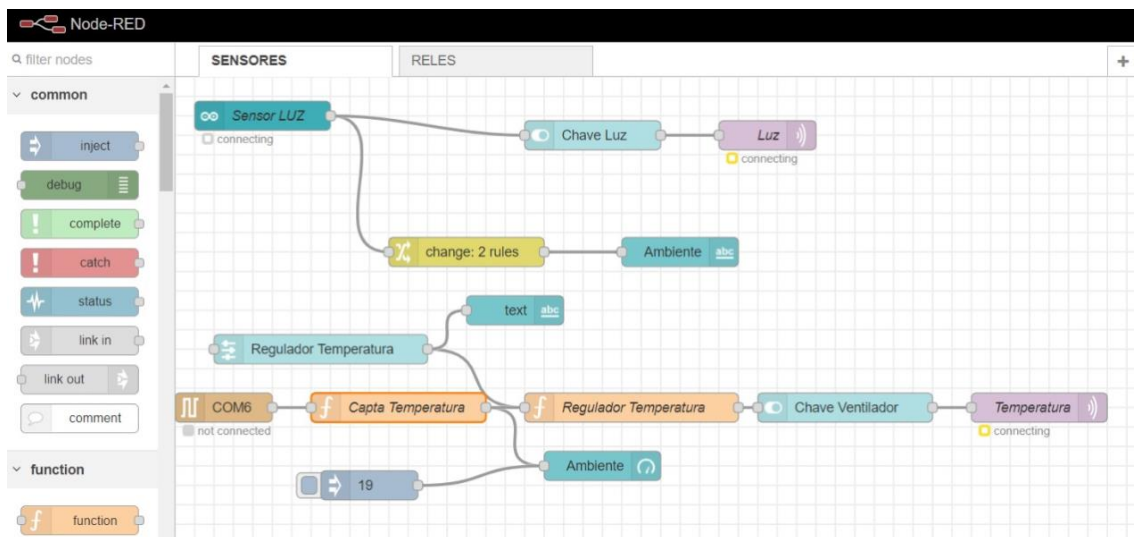


Figura 2 Fluxo Node Red Sensores

Programação no Arduino do sensor de temperatura

```
#include <idDHT11.h>

int idDHT11pin = 2;          //Porta Digital do Arduino onde o Sinal do Sensor DHT esta conectado
int idDHT11intNumber = 0;    //Número da interrupção respectiva à porta definida no parametro anterior

void dht11_wrapper();        // Declaração da função de controle da interrupção.
void loopDHT();              // Atualiza a leitura do sensor

idDHT11 DHT11(idDHT11pin, idDHT11intNumber, dht11_wrapper);    //Instanciação do Objeto de Controle do Sensor

void setup()
{
    Serial.begin(9600);
    Serial.println("Inicio do Sketch");
}

//Variaveis que irao conter os valores lidos no Sensor DHT11
float temperaturaC;
float temperaturaF;
float temperaturaK;
float umidade;
float dewPoint;
float dewPointSlow;

void loop()
{
    loopDHT();

    Serial.print("Temperatura Celcius: ");
    Serial.println( temperaturaC );

    Serial.print("Umidade Relativa: ");
    Serial.println( umidade );

    Serial.print("Ponto de Orvalho: ");
    Serial.println( dewPoint );

    Serial.println();
}

void dht11_wrapper() {
    DHT11.isrCallback();
}

void loopDHT() {
#define tempoLeitura 1000
static unsigned long delayLeitura = millis() + tempoLeitura + 1;
static bool request = false;
```



```
if ((millis() - delayLeitura) > tempoLeitura) {
    if (!request) {
        DHT11.acquire();
        request = true;
    }
}

if (request && !DHT11.acquiring()) {
    request = false;

    int result = DHT11.getStatus();

    switch (result)
    {
    case IDDHTLIB_OK:
        Serial.println("Leitura OK");
        break;
    case IDDHTLIB_ERROR_CHECKSUM:
        Serial.println("Erro\n\r\tErro Checksum");
        break;
    case IDDHTLIB_ERROR_ISR_TIMEOUT:
        Serial.println("Erro\n\r\tISR Time out");
        break;
    case IDDHTLIB_ERROR_RESPONSE_TIMEOUT:
        Serial.println("Erro\n\r\tResponse time out");
        break;
    case IDDHTLIB_ERROR_DATA_TIMEOUT:
        Serial.println("Erro\n\r\tData time out erro");
        break;
    case IDDHTLIB_ERROR_ACQUIRING:
        Serial.println("Erro\n\r\tAcquiring");
        break;
    case IDDHTLIB_ERROR_DELTA:
        Serial.println("Erro\n\r\tDelta time to small");
        break;
    case IDDHTLIB_ERROR_NOTSTARTED:
        Serial.println("Erro\n\r\tNao iniciado");
        break;
    default:
        Serial.println("Erro Desconhecido");
        break;
    }
```

```

}

float valor = DHT11.getCelsius();

if (!isnan(valor)) {
    temperaturaC = valor;
}

valor = DHT11.getHumidity();
if (!isnan(valor)) {
    umidade = valor;
}

valor = DHT11.getFahrenheit();
if (!isnan(valor)) {
    temperaturaF = valor;
}

valor = DHT11.getKelvin();
if (!isnan(valor)) {
    temperaturaK = valor;
}

valor = DHT11.getDewPoint();
if (!isnan(valor)) {
    dewPoint = valor;
}

valor = DHT11.getDewPointSlow();
if (!isnan(valor)) {
    dewPointSlow = valor;
}

delayLeitura = millis();
}
}

```

2.3.2 Passos da Instalação dos Softwares

Na sequência, instalação da IDE do Arduino (www.arduino.cc/en/software). Como o Arduino usado nesse projeto é genérico, temos que corrigir drives do Arduino Genérico - (drive.google.com/file/d/0B2YXepPR2qlAdkR6eXNDMVU3a0E/view). Carregar o Exemplo StandardFirmata no UNO pelo IDE. (Arduino, 2018)

Instalar o Servidor MQTT Mosquitto (<https://mosquitto.org/download/>). Criar servidor MQTT. Acessar Prompt de Comando, depois acessar a pasta onde o Mosquitto foi instalado (Exemplo C:\Program Files\mosquitto), executar o comando de criação: “mosquitto -v”. Criar o servidor MQTT, acessar Prompt de comando a pasta onde o Mosquitto foi instalado (Exemplo C:\Program Files\mosquitto) e executar o comando de criação de tópico:

“mosquitto_sub -h localhost -p 1883 -t “nome do serviço”. (Elipse Knowledgebase, 2021). Foram criados dois servidores, um com o nome de *temperatura* e outro de *luz*.

Instalar o node-red no Windows, instalar o node.js – (<https://nodejs.org/dist/v12.16.2/node-v12.16.2-x64.msi>) e após essa etapa abrir o prompt de comando e digitar prompt de comando do Windows e deve-se digitar (sem aspas) “npm install -g –unsafe-perm node-red” para que a instalação seja realizada. Para acessar a ferramenta, deve-se abrir novamente o prompt de comando do Windows e digitar node-red. (OpenJS Foundation, 2021)

Em seguida, é necessário abrir o navegador e digite o seguinte endereço: localhost:1880. Instalar no celular o aplicativo MQTT Dash, configurar o Brouker no aplicativo.

3. Resultados

As imagens do protótipo do projeto de luz e temperatura casa e conforto.



Imagem 1 Modelo de Montagem de Projeto

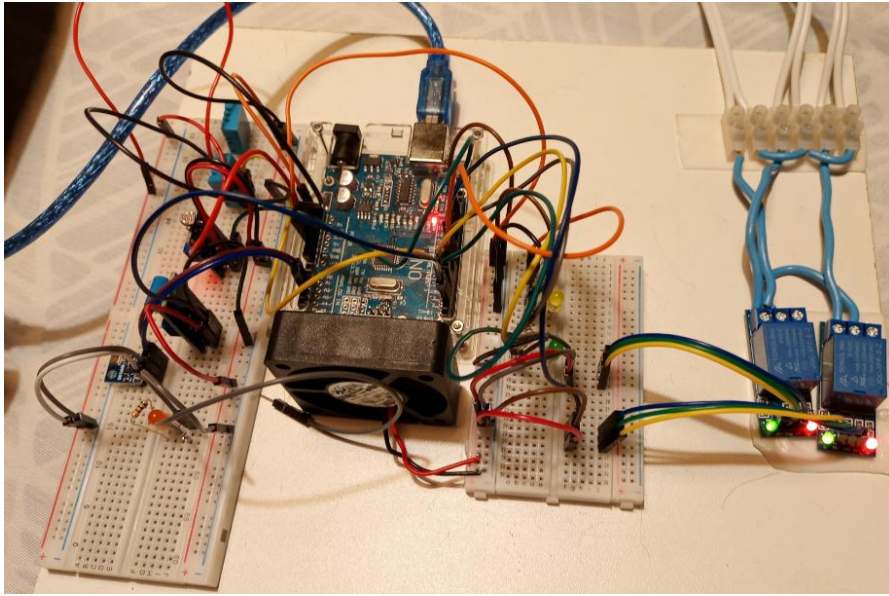


Imagem 1.2 Circuito do projeto

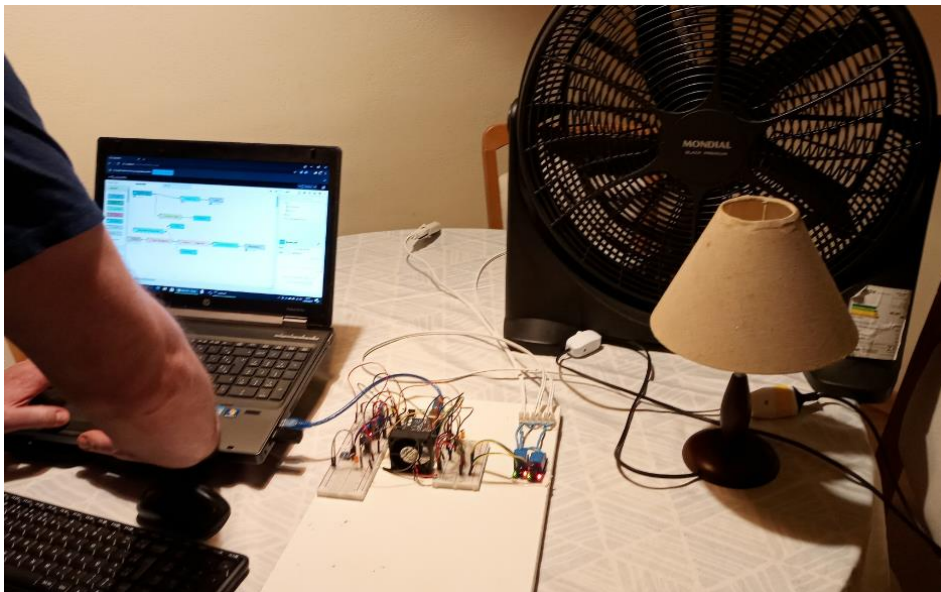


Imagem 1.3 Montagem de Projeto

Link para o youtube: <https://youtu.be/cBocRXDgAyM>

Link para o github: https://github.com/mialq/Projeto_luz_temperatura

4. Conclusões

Nem os objetivos foram alcançados. Não conseguimos implementar o sensor de movimento que faria a luz ascender caso alguém se movimentasse no quarto e estivesse escuro. O sensor simplesmente não funcionava.

Todos os outros objetivos foram alcançados. Os objetivos foram alcançados conseguimos realizar a comunicação entre os componentes do sistema com a internet. Conseguimos controlar a temperatura do ambiente e luz através do Smartphone.

O maior problema foi com a biblioteca do sensor de temperatura que precisou de um script programado diretamente no Arduino para ser reconhecido no node red.

A maior vantagem é o controle dos objetos do ambiente através do smartphone. Desvantagem é o custo que tivemos para fazer todo o projeto funcionar.

O sensor de temperatura podia incluir um sensor de movimento, isso ajudaria a deixar o projeto mais completo, tendo em vista que ao levantar a noite no ambiente, a luz poderia ser acionada através do sensor de movimento.

5. Referências

- Arduino. (2018). *What is Arduino?* Fonte: <https://www.arduino.cc/en/software>
- Arduinoecia. (2013). *Ligando uma lâmpada com módulo relé Arduino*. Fonte: Arduinoecia: <https://www.arduinoecia.com.br/ligando-uma-lampada-com-modulo-rele-arduino/>
- Eclipse Knowledgebase. (2021). *Aplicação exemplo: Driver MQTT em comunicação com Broker Mosquitto MQTT*. Fonte: <https://kb.eclipse.com.br/https://kb.eclipse.com.br/aplicacao-exemplo-driver-mqtt-em-comunicacao-com-broker-mosquitto-mqtt/>
- Filipeflop. (2013). *Monitorando Temperatura e Umidade com o sensor DHT11*. Fonte: [www.filipeflop.com: https://www.filipeflop.com/blog/monitorando-temperatura-e-umidade-com-o-sensor-dht11/](https://www.filipeflop.com/https://www.filipeflop.com/blog/monitorando-temperatura-e-umidade-com-o-sensor-dht11/)
- OpenJS Foundation. (2021). *Node download*. Fonte: Node: <https://nodejs.org/en/>
- Silva de Freitas,, C. C., Larico Chávez, , R. F., Higa, R. S., & Iano, Y. (2012). *Automação residencial: cenário atual e perspectivas futuras* . UNICAMP.
- Smarthouse. (2021). *Automação Residencial*. Fonte: Smarthouse: www.smarthouse.com.br
- Usinainfo. (2021). *Módulo Sensor de Luminosidade Fotossensitivo LDR*. Fonte: [www.usinainfo.com.br: https://www.usinainfo.com.br/sensor-de-luminosidade/modulo-sensor-de-luminosidade-fotossensitivo-ldr-2539.html](https://www.usinainfo.com.br/https://www.usinainfo.com.br/sensor-de-luminosidade/modulo-sensor-de-luminosidade-fotossensitivo-ldr-2539.html)
- USINAINFO. (2021). *Módulo Sensor de Vibração SW-18015 para Arduino*. Fonte: [www.usinainfo.com.br: https://www.usinainfo.com.br/sensor-de-vibracao-arduino/modulo-sensor-de-vibracao-sw-18015-para-arduino-2543.html](https://www.usinainfo.com.br/https://www.usinainfo.com.br/sensor-de-vibracao-arduino/modulo-sensor-de-vibracao-sw-18015-para-arduino-2543.html)