

SISTEMA IOT PARA MONITORAMENTO E CONTROLE DA UMIDADE DO AR

Vinícius Alexandre Soares¹, Me. Gustavo José Correa Gonçalves ²

RESUMO

Este artigo apresenta um sistema de propósito geral e baixo custo baseado no princípio da Internet das Coisas para monitoramento de temperatura e umidade e acionamento de carga em ambientes remotos. O *Hardware* do sistema é composto por um microcontrolador ESP8266 que controla um sensor DHT11 para coletar dados de temperatura e umidade e um relé, que atua em um umidificador. O controle de estados e o gerenciamento destes dispositivos ocorre por meio do aplicativo em tempo real da plataforma Blynk. O projeto foi testado em um local fechado com o objetivo de validar o projeto proposto neste artigo.

Palavras-chave: Umidade; Temperatura; IoT; Sensor; Automação.

ABSTRACT

This article presents a low-cost general-purpose system based on the IoT principle for temperature and humidity monitoring and load triggering in remote environments. The system hardware consists of an ESP8266 microcontroller that controls a DHT11 sensor to collect temperature and humidity data and a relay that operates on a humidifier. State control and management of these devices takes place through the Blynk platform real-time application. The project was tested in a closed place in order to validate the project proposed in this article.

Keywords: Humidity; Temperature; IoT; Sensor; Automation.

¹Graduando do curso de Engenharia de Controle e Automação. Faculdade Estácio de Sá de Ourinhos. Universidade Estácio de Sá de Ourinhos, Ourinhos, São Paulo. E-mail: vinicius-a-soares@outlook.com;

²Prof. Me. no curso de Engenharia de Controle e Automação. Faculdade Estácio de Sá de Ourinhos. Universidade Estácio de Sá. Ourinhos, São Paulo. E-mail: gustavo.correa@estacio.br;

INTRODUÇÃO

Um dos principais fatores para a preocupante qualidade do ar hoje em dia, é o fato de que a porcentagem da umidade do ar está cada vez mais baixa, isso por conta dos longevos períodos sem chuva. Tal fato colocou a capital Paulista em estado de atenção por conta dos baixos índices de umidade do ar, informação dada pelo CGE (Centro de Gerenciamento de Emergências), órgão ligado à prefeitura de SP.

Os Valores ficaram abaixo dos 30%, sendo que segundo a OMS (Organização Mundial da Saúde), índices inferiores a 60% são prejudiciais à saúde. (Folha de São Paulo, 2019).

Esse fato se agrava em demasiados problemas respiratórios e empecilhos no dia-a-dia das pessoas, as consequências com a saúde vão desde ardência e irritação nos olhos e garganta, até casos mais graves como a piora nos casos de Asma em pessoas que já possuam a doença crônica; também surgem crises alérgicas como rinites, sinusites e inflamações das vias respiratórias. (Secretaria do Estado da Saúde, 2011)

Uma das soluções para tal situação em locais fechados, é a utilização de umidificadores de ar, porém, para a efetividade dessa solução, se faz necessário o acompanhamento dos níveis de umidade presentes no ar por meio de sensores como o sensor de umidade e temperatura DHT11, para que assim seja possível atuar o umidificador sempre que necessário de acordo com as informações aferidas pelo sensor.

Importante ressaltar também que deixar o umidificador ligado o tempo todo não é o ideal, assim relata o Hospital Paulista, afinal a alta taxa de umidade ocasionada pelo uso excessivo do umidificador, pode acarretar em demasiados problemas, esses problemas vão desde o surgimento de fungos como também a proliferação de bactérias. (Hospital Paulista, 2018)

Dito isso, por meio da automação proposta com esse projeto, foi feito o controle da qualidade do ar de forma autônoma, deixando o ambiente com a umidade do ar ideal para os seres humanos estipulada de acordo com as recomendações da OMS (Organização Mundial de Saúde).

De acordo com o Hospital Paulista, os umidificadores de ar, trazem muitos benefícios à saúde, desde que sejam usados de forma eficaz. Afirmam que “Quando o aparelho fica ligado por períodos longos causa um excesso de umidade, o que pode

trazer mais problemas do que o alívio, uma vez que os fungos e bactérias se proliferam em alta umidade”. (Hospital Paulista, 2018)

Através da implementação da IoT (*Internet of Things*) junto ao módulo ESP8266 NodeMCU e a utilização do sensor DHT11, foi possível fazer o acompanhamento do funcionamento dos equipamentos e também à análise em tempo real do que está sendo aferido a partir do sensor utilizado, tendo assim, acesso a partir de qualquer local à toda informação a respeito da qualidade do ar no local em questão e também podendo atuar o sistema de qualquer local e a qualquer momento.

Com a ascensão de novas tecnologias computacionais e de automação, unidas à consolidação de paradigmas como IoT (Internet of Things) e WSN (Wireless Sensors Network), é cada vez mais comum que tais mecanismos sejam combinados para realizar o monitoramento de ambientes e o controle de objetos de forma remota. (Lima M. S. et al., 2019, p. 1)

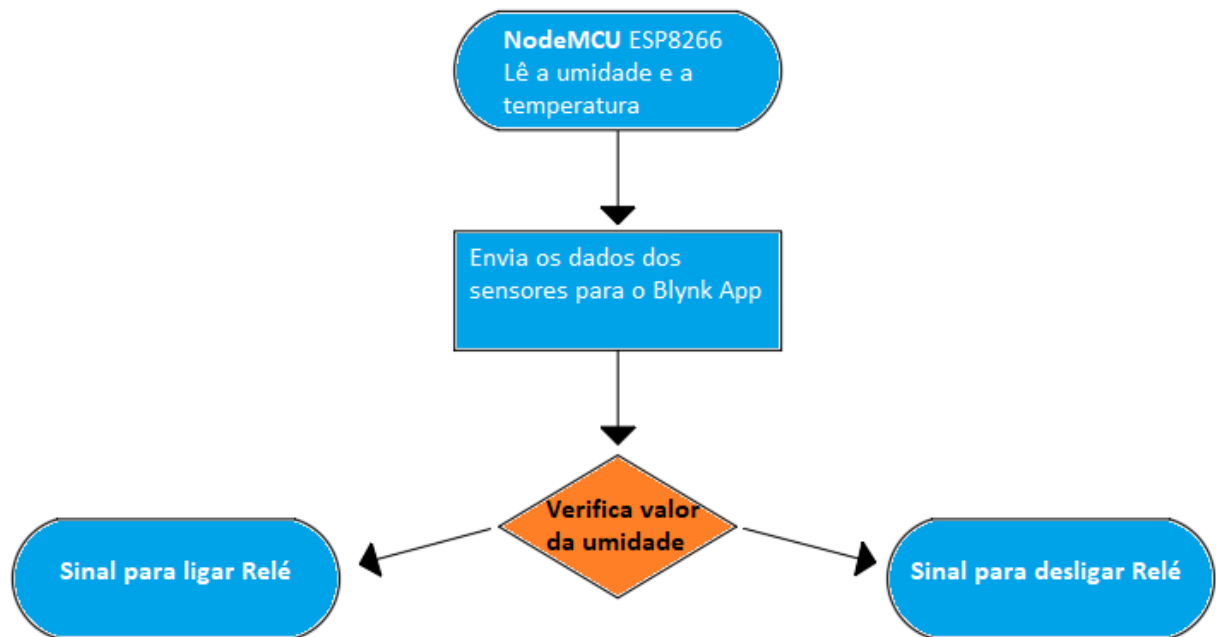
Para a atuação via rede, já tratando o sistema como um sistema IoT e também como uma forma de automatizar o umidificador, foi utilizado um módulo Relé, que foi atuado da forma programada na função lógica fornecida pelo aplicativo Blynk em comunicação com o microcontrolador ESP8266 NodeMCU e passou a informação sobre seu funcionamento à rede.

MATERIAIS E MÉTODOS

Esse trabalho utilizou ferramentas e equipamentos que estão sempre presentes em projetos de automação residencial, que buscam automatizar o ambiente e auxiliar/facilitar no dia-dia das pessoas. O sistema autônomo foi criado visando o monitoramento da umidade e temperatura de um ambiente fechado através do sensor de DHT11 e para o seu funcionamento autônomo foi utilizado um módulo relé para o acionamento de um umidificador.

A automação partiu do microcontrolador NodeMCU Esp8266 carregado com a programação elaborada pelo autor, junto com a utilização de bibliotecas disponíveis pelo software do próprio controlador, que fazendo uso do sistema IoT, mandou pela rede as informações aferidas pelo sensor para o aplicativo Blynk instalado em um Smartphone, possibilitando o monitoramento da qualidade do ar em qualquer lugar com conexão à Internet.

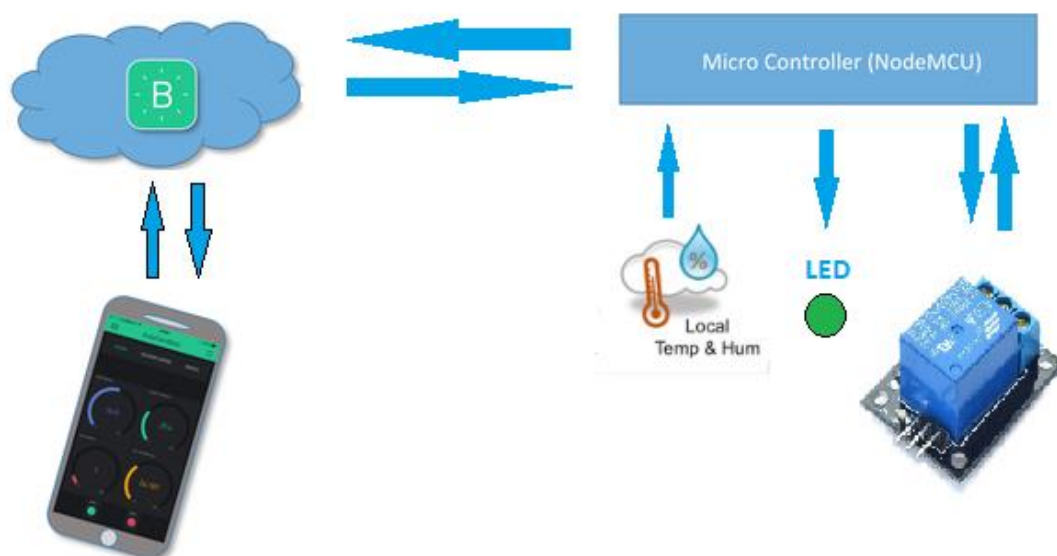
Figura 1 – Fluxograma do sistema.



Fonte: Autoria própria.

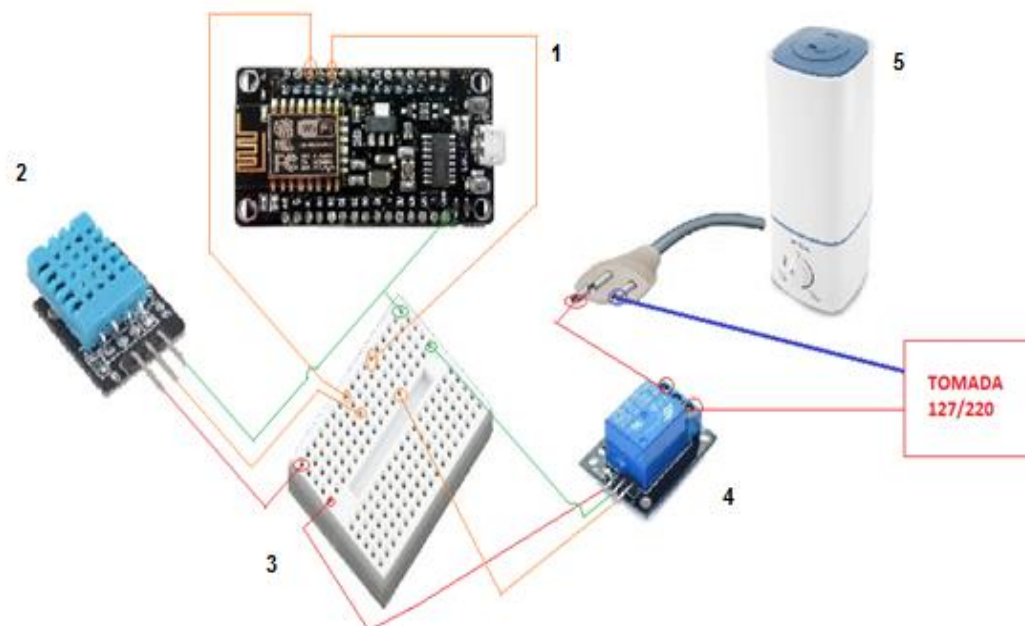
As figuras 2 e 3 mostram os principais componentes do projeto e como será feita sua conexão aos demais dispositivos.

Figura 2 – Fluxograma do funcionamento do sistema.



Fonte: Autoria própria.

Figura 3 - Demonstração das ligações elétricas do circuito.



Fonte: Autoria própria.

De acordo com a figura 3, os itens numerados são:

- 1 – Microcontrolador NodeMCU ESP8266-12
- 2 – Sensor de temperatura e umidade DHT11
- 3 – *Protoboard*
- 4 – Módulo relé Arduino
- 5 – Umidificador de ar bud02B 2.2l Britânia

A figura 1 exemplifica como é feita a tomada de decisão para o acionamento do relé. Na figura 2 verifica-se como se dá a comunicação do aplicativo com os *hardwares* utilizados. Com a figura 3 verifica-se como foi conectado os dispositivos utilizados no projeto. E também como estão interligados e relacionados os componentes existentes. Todos os componentes estão descritos na tabela 1.

Tabela 1 – Relação de materiais utilizados e seus respectivos custos.

Quantidade	Nome do Item	Custo Unitário	Custo Total
1	NodeMCU ESP8266	R\$ 33,00	
1	Sensor DHT11	R\$ 13,90	
1	Modulo Relé	R\$ 8,90	
1	Protoboard 170 pontos	R\$ 5,50	
1	Umidificador de ar bud02B 2,2l Britânia	R\$ 105,00	
1	Cabos e conectores	R\$ 30,00	
			R\$ 193,30

Fonte: Autoria própria.

1. Sensor de temperatura e umidade DHT11

O módulo sensor digital de temperatura e umidade DHT11 permite medir temperaturas de 0 a 50 graus Celsius com um erro de +/- 2 graus, e mede a umidade na faixa de 0% a 90%, com erro de +/- 5%. O módulo inclui um sensor de umidade capacitivo, sendo ele do tipo HR202 e um sensor de medição de temperatura do tipo NTC. O DHT11 é conectado a um controlador de 8 bits de alto desempenho que converte os sinais de temperatura e umidade aferidos pelos sensores e o converte em um sinal serial que é enviado ao microcontrolador. (Aosong, 2019).

Para o pleno funcionamento do sensor com o microcontrolador, aderindo a programação criada no IDE do Arduino, se faz necessário o uso da biblioteca Adafruit DHT *humidity and temperature unified sensor library*. (Arduino, 2019).

Os dados aferidos pelo sensor são exibidos em tempo real pelo aplicativo Blynk, toda informação foi enviada pela rede, assim se caracterizando IoT.

2. NodeMCU ESP8266-12:

É dado o nome de ESP8266 aos microcontroladores que possuem todos os dispositivos necessários para que exista uma conexão com a internet via *wireless*. Seu modo de funcionamento é semelhante ao do Arduino, diferindo-se no fato de possuir a conexão *wireless* sem a necessidade de um módulo extra. Além também de

possuir um poder de processamento maior, conversor serial, regulador de tensão próprio e pinos próprios para I2C, SPI, Analógico e outros. Seu o módulo *wireless* ESP 12F, é uma tecnologia de segurança desenvolvida pela Letter, o módulo suporta o protocolo padrão IEEE802.11 b / g / n. [ELECTRODRAGON, 2017].

Os microcontroladores NODEMCU ESP8266-12, são pequenos, viabilizando assim projetos pequenos, como o proposto com este artigo.

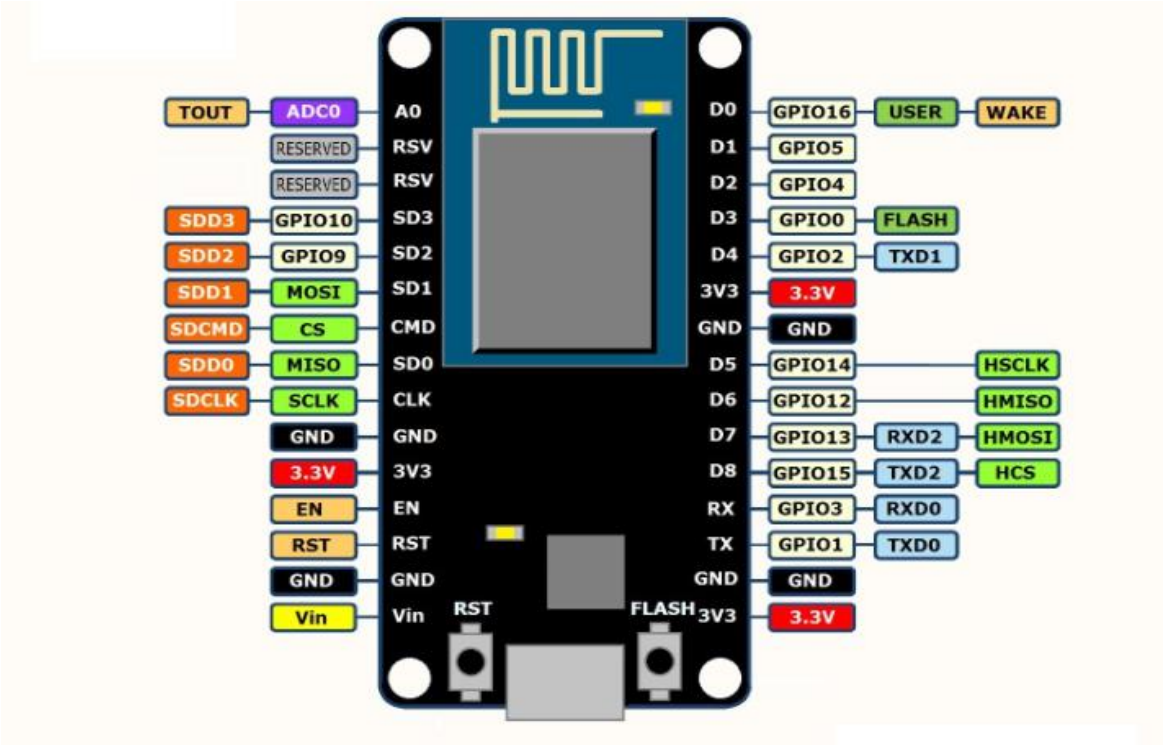
Figura 4 - NodeMCU ESP8266-12.



Fonte: Autoria própria.

Para efetuar a conexão de todos os componentes de forma correta, foi necessário o conhecimento a respeito da pinagem do NodeMCU ESP8266-12. A figura 5 exibe a pinagem e todas as funções em cada pino do microcontrolador.

Figura 5 - Pinagem NodeMCU.



Fonte: FelipeFlop, 2016.

A figura 6, exibe todas definições de GPIO's do ESP8266.

Figura 6 - Definições de GPIO's do ESP8266.

GPIO	Inst Name	Function 0	Function 1	Function 2	Function 3	Function 4	At Reset	After Reset	Sleep
0	GPIO0 U	GPIO0	SPICS2			CLK OUT	oe=0, wpu	wpu	oe=0
1	U0TXD U	U0TXD	SPICS1		GPIO1	CLK_RTC	oe=0, wpu	wpu	oe=0
2	GPIO2 U	GPIO2	I2SO_WS	U1TXD		U0TXD	oe=0, wpu	wpu	oe=0
3	U0RXD U	U0RXD	I2SO_DATA		GPIO3	CLK XTAL	oe=0, wpu	wpu	oe=0
4	GPIO4 U	GPIO4	CLK XTAL				oe=0		oe=0
5	GPIO5 U	GPIO5	CLK_RTC				oe=0		oe=0
6	SD CLK U	SD_CLK	SPICLK		GPIO6	U0CTS	oe=0		oe=0
7	SD DATA0 U	SD_DATA0	SPIQ		GPIO7	U1TXD	oe=0		oe=0
8	SD DATA1 U	SD_DATA1	SPIQ		GPIO8	U1RXD	oe=0		oe=0
9	SD DATA2 U	SD_DATA2	SPIHD		GPIO9	HSPICLK	oe=0		oe=0
10	SD DATA3 U	SD_DATA3	SPIWP		GPIO10	HSPICLK	oe=0		oe=0
11	SD CMD U	SD_CMD	SPICS0		GPIO11	U0CTS	oe=0		oe=0
12	MTDI U	MTDI	I2SI_DATA	HSPICLK	GPIO12	U0DTR	oe=0, wpu	wpu	oe=0
13	MTCK U	MTCK	I2SI_BCK	HSPICLK	GPIO13	U0CTS	oe=0, wpu	wpu	oe=0
14	MTMS U	MTMS	I2SI_WS	HSPICLK	GPIO14	U0DSR	oe=0, wpu	wpu	oe=0
15	MTDO U	MTDO	I2SO_BCK	HSPICS	GPIO15	U0RTS	oe=0, wpu	wpu	oe=0
16	XPD_DCDC	XPD_DCDC	RTC_GPIO0	EXT_WAKEUP	DEEPSLEEP	BT_XTAL_EN	oe=1, wpu	oe=1, wpu	oe=1

Fonte: Vida De Silício, 2017.

3. IDE Arduino

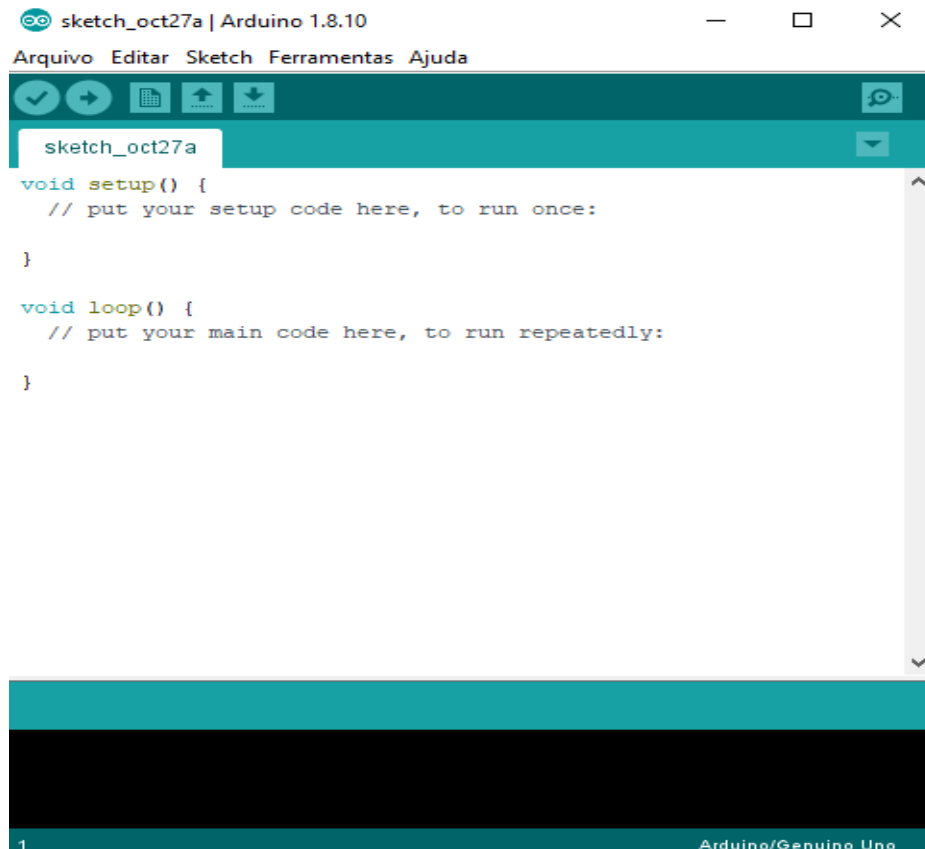
A IDE (*Integrated development environment*) ou Ambiente integrado de desenvolvimento do Arduino é um compilador multiplataformas que foi desenvolvido na linguagem Java e funciona nos sistemas operacionais Windows, macOS e Linux.

A função do compilador é a de elaborar e fazer upload de programas em placas compatíveis com Arduino, núcleos de terceiros e outras placas de desenvolvimento. O IDE do Arduino suporta as linguagens C e C++ usando regras especiais de estruturação de código. [Arduino, 2019]

A IDE do Arduino possui bibliotecas de software da plataforma de prototipagem *Wiring*, que fornece muitos procedimentos comuns de entrada e saída.

A programação gravada pelo usuário requer apenas duas funções básicas para iniciar, o esboço e o *loop* principal, que são compilados e vinculados à uma função do programa denominada como *main*. A IDE do Arduino converte o programa executável em um arquivo de texto em codificação hexadecimal que é carregada na placa do microcontrolador por um programa carregador no *firmware* da placa. [Arduino, 2019]

Figura 7 – IDE Arduino.

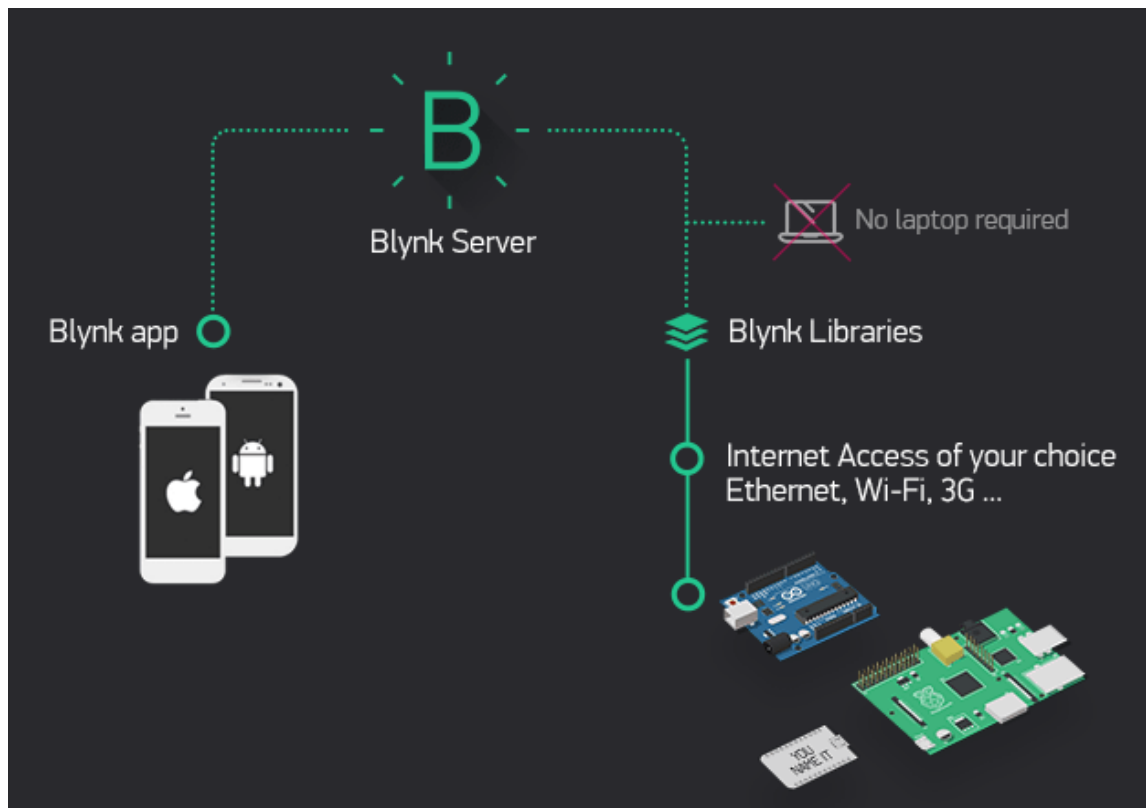


Fonte: Autoria Própria.

4. Blynk

O aplicativo Blynk neste projeto foi tratado como interface com o usuário, por esse aplicativo que o usuário tem acesso e pode monitorar o que é aferido pelos sensores. Esse aplicativo é composto de três partes: o Blynk App, o Blynk Server e a Blynk Library, conforme figura 8:

Figura 8 – Funcionamento aplicativo Blynk.



Fonte: Blynk, 2019.

4.1 Blynk App

É um aplicativo personalizável que permite controlar remotamente um hardware programável, bem como reportar dados do hardware ao aplicativo. Ele se encontra disponível para sistemas Android e iOS, sua interface simples e prática, permite a criação de aplicações e funções de leitura para a interação com hardware. (Blynk, 2019)

4.2 Blynk Server

A comunicação entre o projeto e o aplicativo Blynk se dá através do servidor Cloud Blynk. O servidor faz a transmissão dos dados ao *hardware*, armazena os

estados escolhidos e aferidos no aplicativo e do *hardware* e também armazena os dados aferidos pelos sensores mesmo se o aplicativo estiver fechado. (Blynk, 2019)

4.3 Blynk Libraries

Ao lado do *hardware* existem as bibliotecas Blynk para diversas plataformas de desenvolvimento. É esta biblioteca que gerencia todas as conexões do *hardware* com o servidor Blynk e também monitora todas as requisições de entrada e saída de dados e comandos. (Blynk, 2019).

Na figura 9 é possível verificar o layout criado a partir do Blynk App:

Figura 9 – Interface de visualização do projeto no aplicativo Blynk.



Fonte: Autoria própria.

5. Módulo Relé Arduino

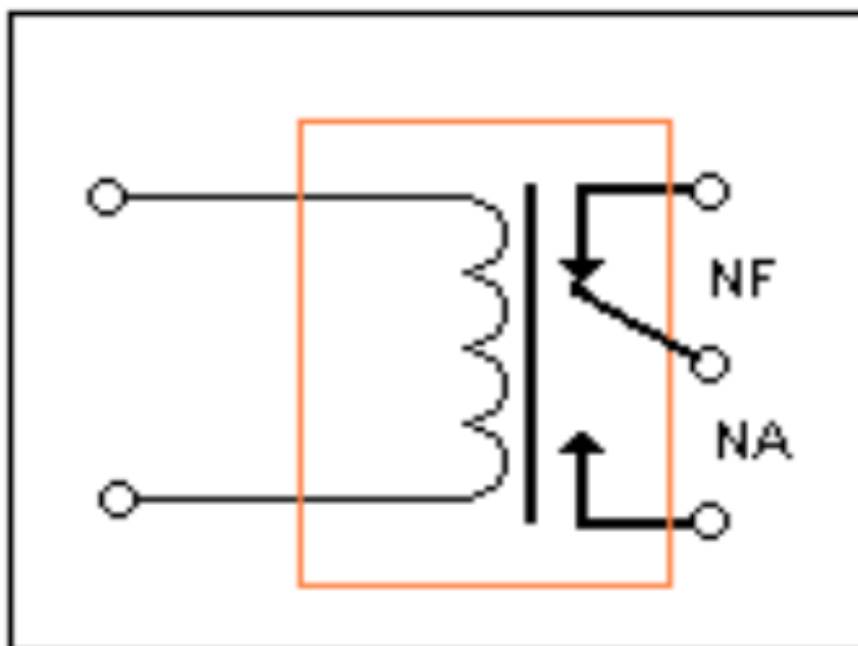
É um componente eletromecânico, onde é possível o acionamento de um interruptor a partir de um sinal. Sendo que o interruptor e o sinal de comando estão sempre completamente isolados entre si.

O interruptor está dentro do relé e geralmente possui uma alta capacidade de tensão e corrente, mesmo em relés pequenos. Funciona exatamente como uma chave (interruptor), possui uma bobina e um contato preso a uma mola de rearme, que conecta com os terminais nas posições NA (Normalmente Aberto) e NF (Normalmente Fechado).

No borne há 3 conexões: NA, C (Comum) e NF. Assim, quando o Módulo Relé estiver "desligado", C estará conectado à NF. Quando estiver ligado, C estará conectado à NA. Quando a bobina é energizada, ela cria um campo eletromagnético, que funciona como um ímã e atrai e desloca o contato. Então o relé passa a desconectar do NF do contato central que passa a estar conectado no NA. Isso acontece sem misturar os sinais, já que a bobina é totalmente isolada dos contatos que serão chaveados. (ATHOSELECTRONICS, 2019)

Na figura 10 é possível verificar que a bobina interna presente em um relé é totalmente isolada dos contatos que serão chaveados.

Figura 10 – Esquema elétrico do modulo relé.

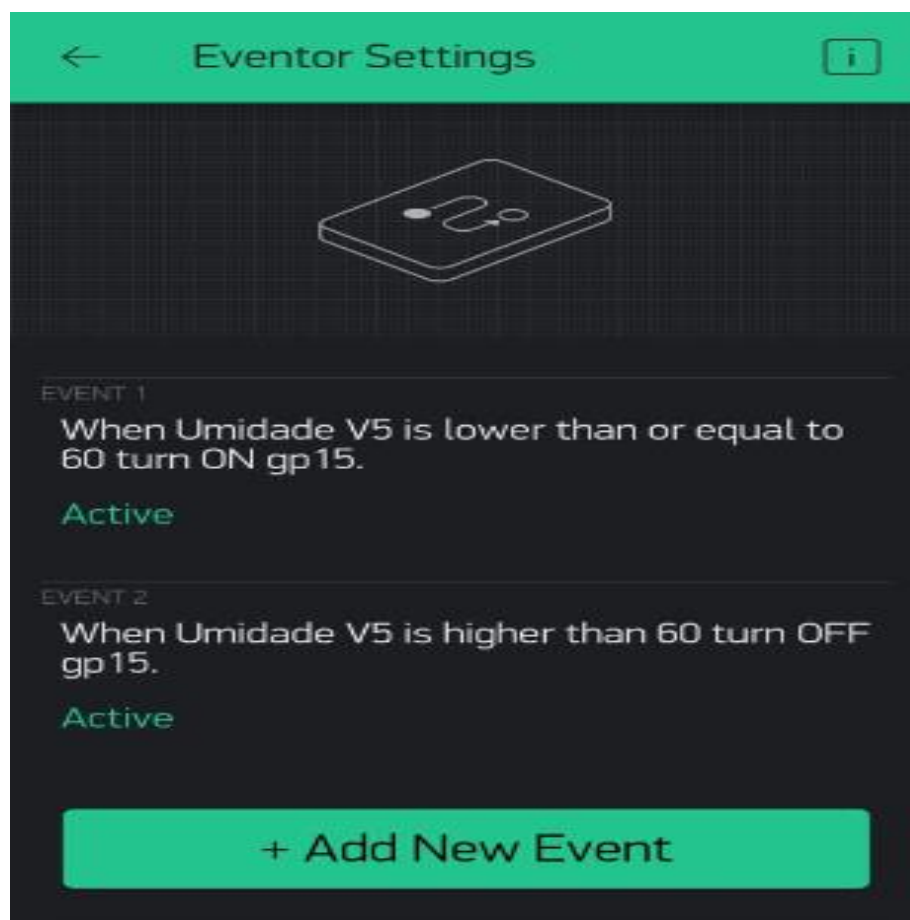


Fonte: AthosElectronics (2019).

Neste projeto, o módulo Relé foi utilizado para o acionamento do umidificador com base nos valores aferidos pelo sensor DHT11. Todo o decorrer do evento foi sinalizado no Blynk App. Uma função interna inclusa no Blynk app foi a responsável pela atuação do relé.

A figura 11 demonstra como é tratada essa função no aplicativo

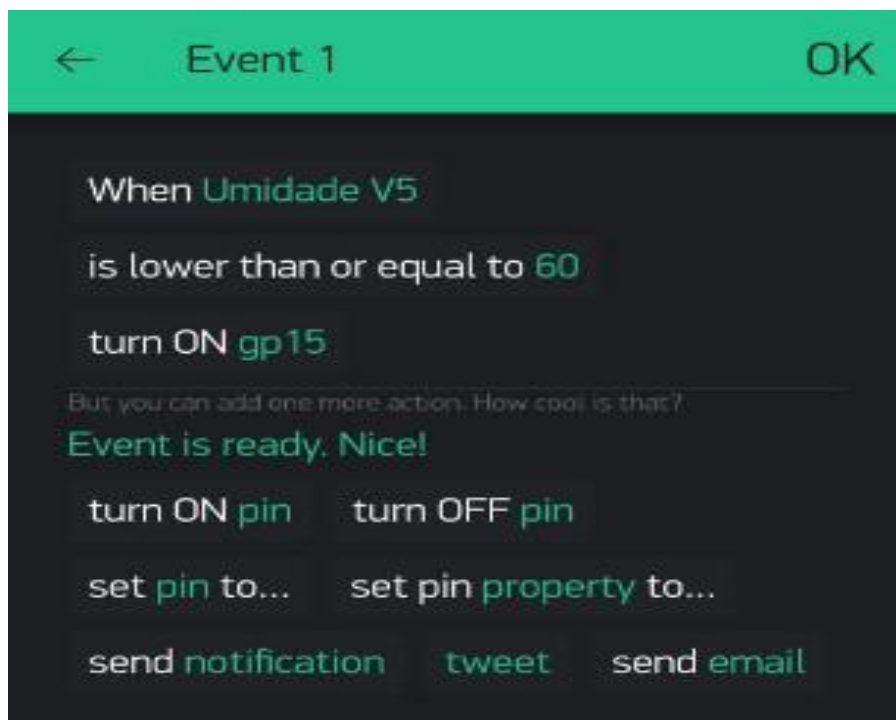
Figura 11 – Função acionamento Relé no Blynk.



Fonte: Autoria própria.

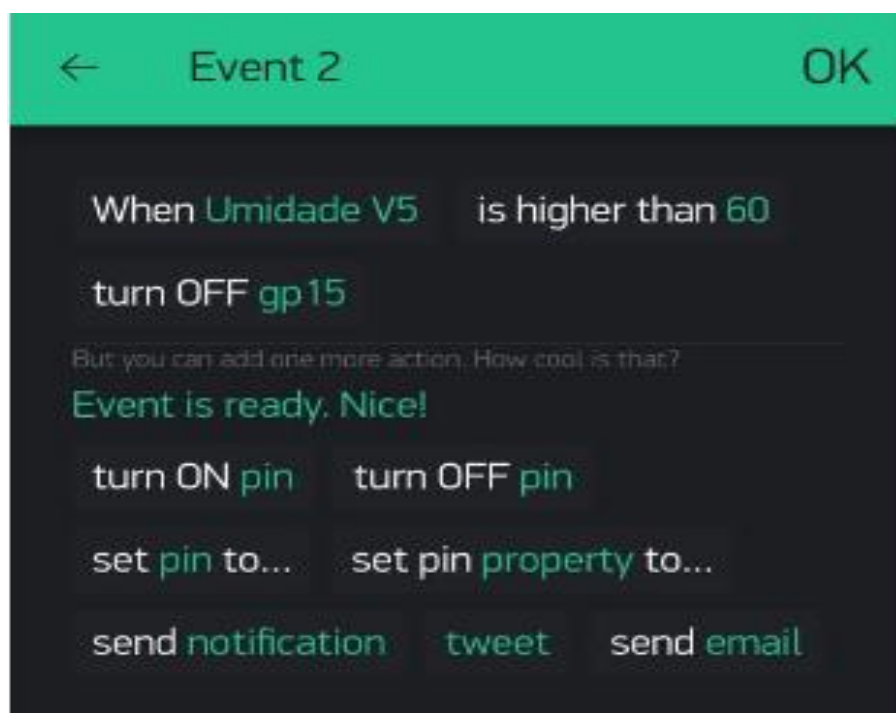
As figuras 12 e 13 exibem como é a programação para os eventos 1 e 2 exibidos com a Figura 11.

Figura 12 – Evento 1 da função de acionamento para o Relé.



Fonte: Autoria própria.

Figura 13 – Evento 2 da função de acionamento do Relé.



Fonte: Autoria própria.

RESULTADOS

Os resultados com o projeto serão verificados com o tempo de atuação do mesmo, podendo verificar sua efetividade no auxílio à melhoria da qualidade do ar junto ao gráfico proposto com o Blynk app.

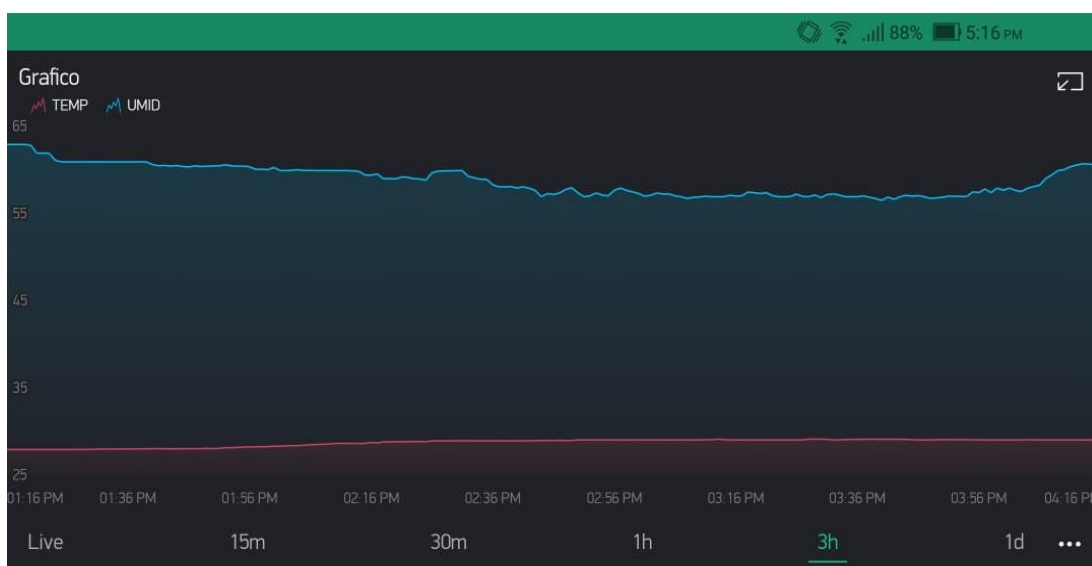
Nos testes realizados, o funcionamento ocorreu conforme planejado, atendendo ao objetivo principal que era obter-se um controle autônomo e eficaz via IoT de um umidificador com a pretensão da melhoria na qualidade do ar, mantendo assim valores aceitáveis de umidade. Os resultados obtidos com o projeto podem ser considerados satisfatórios em relação aos testes de funcionamento, pois conseguiram introduzir um controle eficaz, mantendo a umidade nos valores propostos de umidade relativa do ar e mantendo-se uma boa e estável conexão com os servidores do Blynk.

O projeto em questão atuou somente em áreas fechadas durante todos os testes.

Com os resultados obtidos verifica-se a possibilidade da aplicação do mesmo em qualquer outro local com aspectos semelhantes. Altas quantidades de ventos podem dificultar na umidificação do local, demandando mais tempo para satisfazer as condições do projeto. Afim de atestar o funcionamento e tempo de resposta do projeto, foram alterados em meio a uma simulação, os valores de setpoint necessário para o acionamento do relé, inicialmente o setpoint era 65UR (Umidade Relativa) e foi alterado para 60UR como pode-se notar a diferença no gráfico.

Na figura 14 observa-se o gráfico que demonstra a variação da umidade durante um período de 3 horas de testes.

Figura 14 – Gráfico de umidade num período de 3 horas de funcionamento.



Fonte: Autoria própria.

DISCUSSÃO

Uma das dificuldades encontradas foi a realização de uma pesquisa aplicada sobre o problema da baixa umidade relativa do ar, pois se trata de um assunto de caráter rotineiro e com discussões pouco aprofundadas no assunto.

Para a plena provação das funcionalidades do projeto, se demanda tempo de funcionamento, o sistema precisaria ficar ligado durante demasiadas horas, afim de testar o controle a longo prazo, para assim verificar se os níveis de umidade seriam controlados em faixas aceitáveis em períodos tão longos de funcionamento. Também precisaria ser avaliado com o tempo, o impacto da umidade constantemente em alteração próxima aos dispositivos utilizados.

Outros elementos que possam validar ainda mais ou invalidar a efetividade do sistema vão surgir no decorrer do tempo e de sua utilização.

O baixo custo e fácil manuseamento dos equipamentos foram pontos cruciais para a execução do projeto, sendo assim uma pratica fácil e acessível para uma automação IoT.

Outro ponto positivo foi a utilização do aplicativo Blynk, que possui uma interface simples e cheia de recursos, como a condição de utilizar blocos programáveis em sua interface, que diminuem o tamanho e a complexidade da programação no IDE do microcontrolador. Também abre a possibilidade da criação de um server próprio, não dependendo de seu servidor Nuvem, assim abrindo a possibilidade da criação de projetos maiores, com maior tráfego de informações, de forma confiável e rápida.

CONCLUSÃO

Nos dias atuais a população cada vez mais busca por alternativas de melhorar com sua qualidade de vida, meios que facilitem tarefas básicas do dia-dia, desta forma com as pesquisas realizadas durante o projeto, se observou um crescente aumento no interesse da população com relação à automação residencial, surgindo assim a todo momento novos projetos relacionados a automação com IoT, estes visando sempre o baixo custo e a facilidade em sua criação, possibilitando que qualquer pessoa com interesse e as ferramentas necessárias possam automatizar sua residência da forma que preferirem, obtendo melhorias em sua qualidade de vida, assim como proposto com este projeto.

REFERÊNCIAS BIBLIOGRÁFICA

ARDUINO (2019). O que é Arduino? Disponível em: <<https://www.arduino.cc>>. Acesso em: 20/10/2019.

Arduino e Cia (2013). Sensor de umidade e temperatura DHT11. Disponível em: <<https://www.arduinoecia.com.br/sensor-de-umidade-e-temperatura-dht11/>>. Acesso em: 10/10/2019.

AthosElectronics (2019). Relé o que é e como funciona. Disponível em: <<https://athoselectronics.com/rele/>>. Acesso em: 20/10/2019.

Bajrami, X.; Murturi, I. *An efficient approach to monitoring environmental conditions using a wireless sensor network and NodeMCU*. E & I Elektrotechnik and Informationstechnik, p.294–301, 2018.

Braga, A. R. et al. 2017. Gerenciamento térmico e elétrico de um centro de dados utilizando sensoriamento IoT. Em SBCUP – 9º Simpósio Brasileiro de Computação. XXXVII Congresso da Sociedade Brasileira de Computação.

Blynk (2019). *Getting Started*. Disponível em: <<https://blynk.io/en/getting-started>>. Acesso em 20/10/2019.

Blynk Library (2019). Blynk App library. Disponível em: <<https://github.com/blynkkk/blynk-library/releases>>. Acesso em 10/10/2019.

Aosong (2019). Aosong Temperature and humidity module DHT11 Product Manual. Disponível em: <<https://datasheet4u.com/datasheet-parts/DHT11-datasheet.php?id=785591>>. Acesso em 10/10/2019.

DHTSensorLibrary (2016). Adafruit DHT humidity and temperature unified sensor library. Disponível em: <https://github.com/adafruit/DHT-sensor-library>. Acesso em 10/10/2019.

Electrodragon (2017). Datasheet ESP8266. Disponível em: <https://www.electrodragon.com/w/ESP-12F_ESP8266_Wifi_Board>. Acesso em 10/10/2019.

EMBARCADOS (2018). Introdução ao Blynk. Disponível em: <<https://www.embarcados.com.br/introducao-ao-blynk-app>>. Acesso em 20/10/2019.

FelipeFlop (2016). *Como programar o módulo ESP8266 NodeMCU*. Disponível em: <<https://www.filipeflop.com/blog/esp8266-nodemcu-como-programar/>>. Acesso em 10/10/2019.

Folha de São Paulo (2019). São Paulo entra em estado de atenção por causa da baixa umidade do ar. Disponível em: < <https://agora.folha.uol.com.br/sao-paulo/2019/09/sao-paulo-entra-em-estado-de-atencao-por-causa-da-baixa-umidade-do-ar.shtml>>. Acesso em 13/10/2019

Hospital Paulista (2018). Você sabe usar corretamente o umidificador de ar? Disponível em: <<http://www.hospitalpaulista.com.br/voce-sabe-usar-corretamente-o-umidificador-de-ar>>. Acesso em: 20/10/2019.

JUNIOR, S. L. S. Internet das coisas: Fundamentos e aplicações em Arduino e NodeMCU: 1. ed. São Paulo: Érica, 2018.

Lima, M. S. et al. Sistema IoT para Monitoramento de Temperatura e Umidade Ambientes e Acionamento Remoto de Cargas. In: [S.l.]: Eixo da Computação – Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Maracanaú, 2019.

Portal do governo (2019). Tempo seco pode causar problemas respiratórios. Disponível em: <<http://www.saopaulo.sp.gov.br/ultimas-noticias/cuidados-com-a-saude-tempo-seco-pode-causar-problemas-respiratorios>>. Acesso em: 20/10/2019.

ROSE, D. *Enchanted Objects: Innovation, Design, and the Future of Technology IOT*. New York: Scribner, 2014.

Secretaria de Estado da Saúde (2011). SP alerta para cuidados com a saúde no tempo seco. Disponível em: <<http://www.saude.sp.gov.br/ses/noticias/2011/maio/sp-alerta-para-cuidados-com-a-saude-no-tempo-seco>>. Acesso em: 20/10/2019.

Silva, A. R. et al. 2016. Rede de sensores para controle inteligente de ambientes. In 8º SBCUP - Simpósio Brasileiro de Computação Ubíqua e Pervasiva - XXXVI CSBC - Congresso da Sociedade Brasileira de Computação.

Vida De Silício (2017). O que é ESP8266 NodeMCU – A Família ESP e o NodeMCU. Disponível em: <<https://portal.vidadesilicio.com.br/o-que-esp8266-nodemcu>>. Acesso em: 10/10/2019.

ANEXO A – Programação enviada ao microcontrolador

```
//TCC - Engenharia de Controle e Automação  
//SISTEMA IOT PARA MONITORAMENTO E CONTROLE DA UMIDADE DO AR  
//Vinicius Alexandre Soares  
//Programação Utilizada.
```

```
#define BLYNK_PRINT Serial  
#include <SPI.h>  
#include <ESP8266WiFi.h>  
#include <BlynkSimpleEsp8266.h>  
#include <SimpleTimer.h>  
#include <DHT.h>  
char auth[] = " "; // Token pessoal.
```

```
char ssid[] = " "; //LOGIN DA REDE  
char pass[] = " "; //SENHA DA REDE
```

```
#define DHTPIN 2
```

```
#define DHTTYPE DHT11  
DHT dht(DHTPIN, DHTTYPE);
```

```
declarado no aplicativo.  
SimpleTimer timer;  
WidgetLED LEDb1(V1);  
WidgetLED LEDb2(V2);
```

```
void sendSensor()  
{  
  float h = dht.readHumidity();  
  float t = dht.readTemperature();
```

```
  Serial.print("temp: ");  
  Serial.println(t);  
  Serial.print("umid: ");  
  Serial.println(h);
```

```
  if (isnan(h) || isnan(t)) {  
    Serial.println("Failed to read from DHT sensor!");  
    return; }  
}
```

```
Blynk.virtualWrite(V5, h);  
Blynk.virtualWrite(V6, t);  
Blynk.virtualWrite(V0, t);  
WidgetLED led1(V1);
```

```
if(h<50){
```

```

    led1.on();
  }
  else{led1.off();}
}
void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  dht.begin();
  timer.setInterval(1000L, sendSensor);
}

void loop()
{
  Blynk.run(); // Initiates Blynk
  timer.run(); // Initiates SimpleTimer
}

```

SOARES, Vinícius Alexandre

S Sistema IoT para monitoramento e controle da
676s umidade do ar / Vinícius Alexandre Soares - Ourinhos -
2019.
20f. il. ; 30cm.

Monografia (Graduação em Engenharia de Controle
e Automação)-Faculdade Estácio de Sá de Ourinhos, 2019.
Orientador: Me. Gustavo José Corrêa Gonçalves

Bibliografia

1. Umidade. 2. Temperatura. 3. IoT. 4. Sensor. 5.
Automação. I. Faculdade Estácio de Sá de Ourinhos. II.
Título.

CDD
621
