



## Porta jóias automatizado

Amanda Ferreira Ramalho<sup>1</sup>, Gustavo Rosas Boteon<sup>2</sup>, Wilian França Costa<sup>1</sup>

<sup>1</sup>Universidade Presbiteriana Mackenzie (UPM)

Rua da Consolação, 930 Consolação, São Paulo - SP, 01302-907 – Brazil

[41904567@mackenzista.com.br](mailto:41904567@mackenzista.com.br) , [41913469@mackenzista.com.br](mailto:41913469@mackenzista.com.br)

**ABSTRACT.** *This article describes the IoT design for a smart jewelry box. Using an infrared sensor, servo motor and a NodeMCU along with some resistors and two LEDs, one red and one green. We developed a system in which through the cell phone, using a mobile application, it is possible to enter a password or register it to open the container. This information is connected through the MQTT Dashapp, which can receive commands from the user via a mobile device to open the lid of the jewelry box.*

**KEYWORDS:** *Internet of Things, NodeMCU, MQTT Dash, password.*

**RESUMO.** *Este artigo descreve o projeto de IoT para um porta jóias inteligente. Usando um sensor infravermelho, o servo motor e um NodeMCU junto com alguns resistores e dois LED, um vermelho e outro verde. Desenvolvemos um sistema em que através do celular utilizando um aplicativo móvel é possível inserir uma senha ou cadastrá-la para abrir o contêiner. Essas informações são conectadas através do aplicativo MQTT Dash, que pode receber comandos do usuário através de um aparelho celular para abrir a tampa do porta joias.*

**PALAVRAS-CHAVE:** *Internet das Coisas, NodeMCU, MQTT Dash, Porta jóias, senha.*

### 1. Introdução

Com as diversas possibilidades trazidas pela Internet das coisas este projeto busca atender aqueles que desejam proteger seus bens com facilidade e através de um aparelho móvel. Este projeto está sendo desenvolvido para atender a uma necessidade de proteger bens de valor. Para tanto, será possível cadastrar uma senha e desbloquear o porta joias por meio do aplicativo de celular MQTT Dash.



## **2. Materiais e Métodos**

O projeto tem como componentes de *Hardware*:

1 NodeMCU: Microcontrolador com conexão Wi-Fi, onde será responsável pela *software* da aplicação.



**Figura 1. Representação do NodeMCU**

1 Cabo Micro USB: Cabo que faz a conexão entre o NodeMCU e o computador, funcionando também como a fonte de energia.



**Figura 2. Representação do Micro USB**

1 LED Vermelho: Um condutor de energia elétrica, que quando energizado, emite luz visível na cor vermelha. Será usado como confirmação visual para o usuário.



**Figura 3. Representação do LED Vermelho**

1 LED Verde: Um condutor de energia elétrica, que quando energizado, emite luz visível na cor verde. Será usado como confirmação visual para o usuário.



**Figura 4. Representação do LED Verde**



1 Servo motor 9g: Um motor que pode ajustar seu ângulo de abertura, será responsável tanto pela trava do container quanto sua abertura.



Figura 5. Representação do NodeMCU. Disponível em:

<https://www.baudaeletronica.com.br/micro-servo-9g-sg90-tower-pro.html>

Arduino IDE: Aplicação utilizada para criação e inserção de um *software* para a placa microcontroladora compatível com códigos arduino.

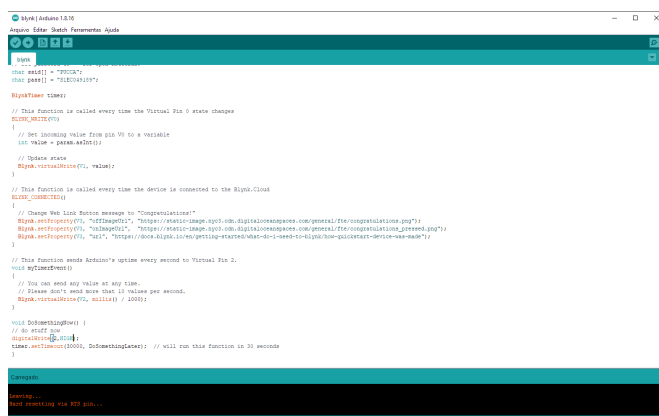


Figura 6. Representação do Arduino IDE, imagem própria autoria

1 Sensor de Obstáculo Infravermelho TCRT5000: Este circuito é composto por um emissor e um receptor IR. para que a placa saiba se o container se encontra fechado ou aberto.

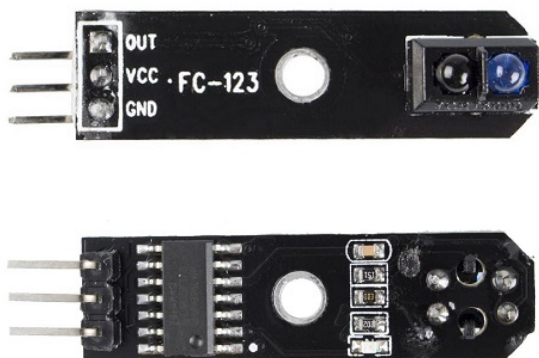




Figura 7. Representação do TCRT5000 Infrared Sensor Module. Disponível em: [http://wiki.sunfounder.cc/index.php?title=TCRT5000 Line Following Tracking Sens or Module](http://wiki.sunfounder.cc/index.php?title=TCRT5000+Line+Following+Tracking+Sens+or+Module)

MQTT Dash: Aplicativo *Mobile* que torna possível a conexão do *smartphone* com o NodeMCU.

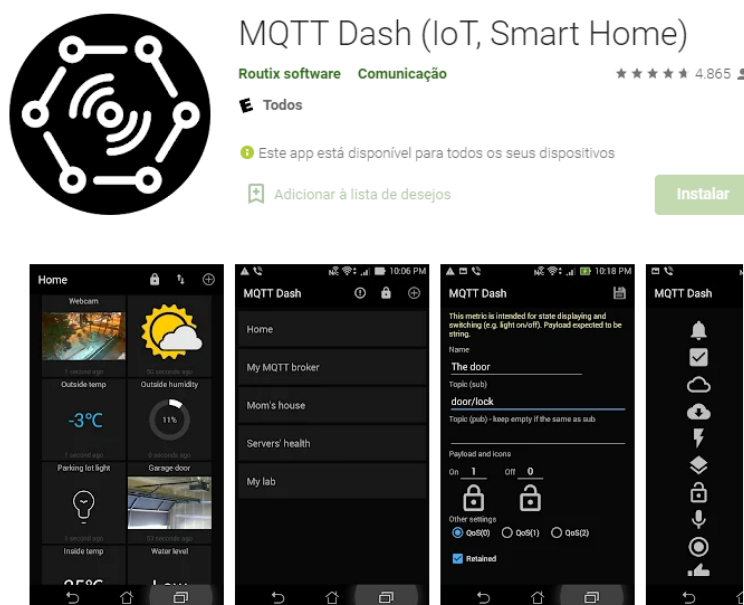


Figura 8. Representação do MQTT Dash via Mobile. Disponível em: [https://play.google.com/store/apps/details?id=net.routix.mqttldash&hl=pt\\_BR&gl=US](https://play.google.com/store/apps/details?id=net.routix.mqttldash&hl=pt_BR&gl=US)

### 3. Resultados

Os primeiros testes do projeto foram a realização da conexão Wi-Fi entre a placa e um aparelho móvel utilizando o MQTT Dash, assim fazendo os primeiros *subscribes* e enviando comandos para a placa através do celular com operações simples de movimento do servo motor e acendendo os leds. Após os testes criamos os *subscribes* de validação de senha e reset de senha responsável pelo cadastro da senha e a validação da entrada do usuário para abertura do container.

Após a criação desses *subscribes* desenvolvemos o código que usa o sensor para definir se a tampa se encontra aberta ou fechada, criamos a variável *isObstacle* que possui os valores HIGH e LOW que representa se a tampa se encontra fechada ou aberta através do sensor infravermelho, conforme o *digitalRead* do senso.

Os testes subsequentes foram para a responsividade dos led quando o usuário entra com a senha errada ou correta dando uma resposta visual para a interação do usuário com o



dispositivo. Também foi testada a força do motor servo para saber se ele é capaz de abrir a caixa e também mantê-la trancada.

Link do repositório no GitHub: [https://github.com/GBoteon/porta\\_joias\\_automatizado](https://github.com/GBoteon/porta_joias_automatizado).



Figura 9. Representação do Projeto Porta Jóias Automatizado. Disponível em: [https://github.com/GBoteon/porta\\_joias\\_automatizado/tree/main/fotos](https://github.com/GBoteon/porta_joias_automatizado/tree/main/fotos)

#### 4. Fluxograma do Código de Implementação

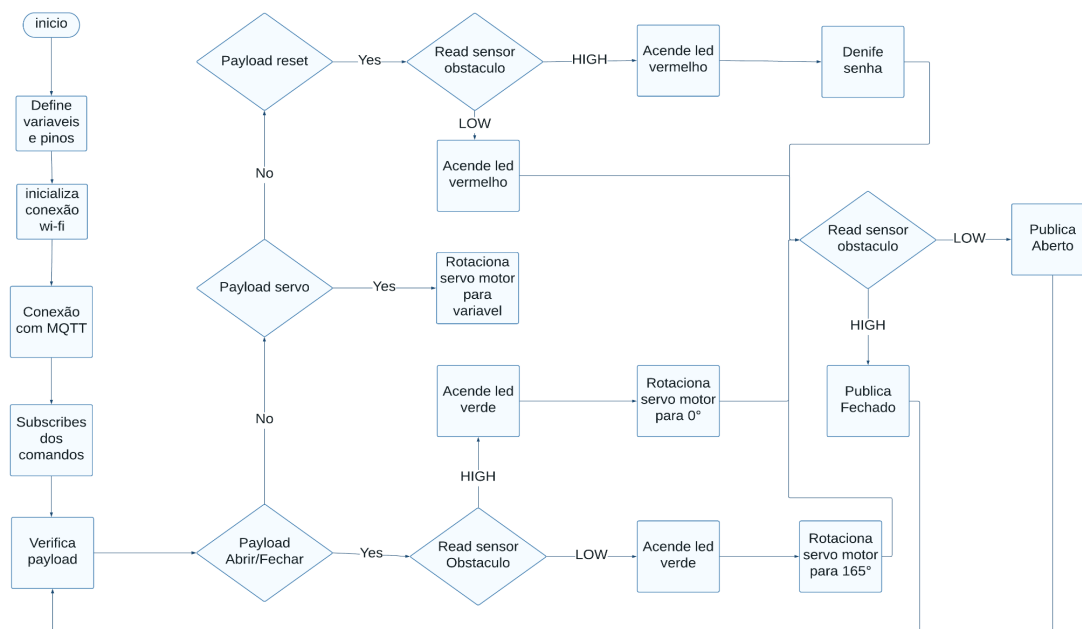


Figura 10. Representação fluxograma do código de implementação

#### 5. Diagrama do Circuito Eletroeletrônico

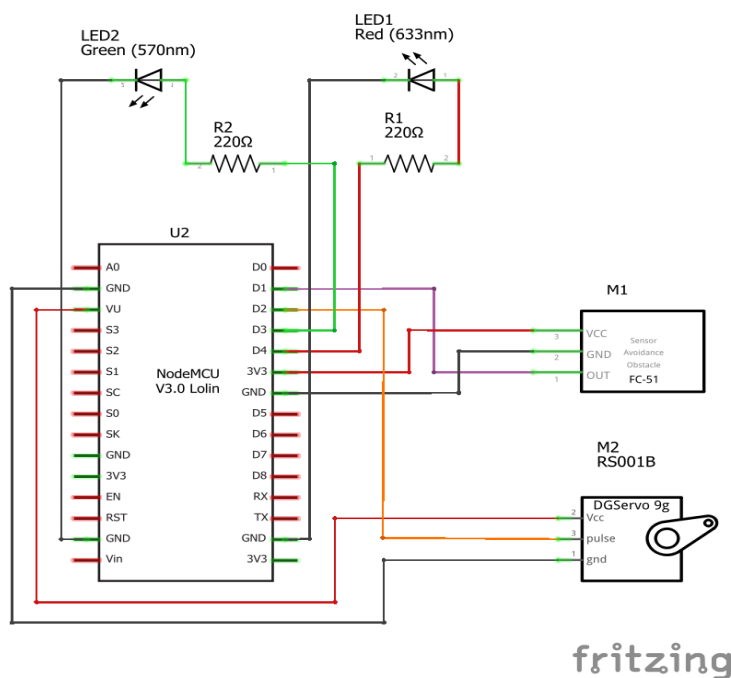


Figura 11. Representação diagrama do circuito eletroeletrônico.

## 6. Conclusões

Com base nos resultados satisfatórios apresentados na fase de testes, podemos concluir que o projeto alcançou seus objetivos. A segurança é o grande benefício deste projeto, podendo trazer reuso deste *software* para atender outros tipos de necessidades similares.

Um dos problemas encontrados no projeto é o fato que o motor servo não oferece muita resistência a abrir quando a caixa está fechada, mesmo que ele mantenha a última posição comandada, este problema poderia ser resolvido em um projeto futuro usando um motor mais robusto. ou um sistema de tranca

O custo/benefício deste projeto também o torna bem viável, podendo ser adaptado e acoplado a outros *softwares*/produtos, aumentando a sua complexidade e custo futuro.

## 7. Referências

INTRODUÇÃO ao Blynk App. Embarcados, São Paulo, 8 de maio de 2018. Disponível em:<<https://www.embarcados.com.br/introducao-ao-blynk-app/>>. Acesso em: 24 de agosto de 2021.

PRIMEIROS passos com mosquitto (broker MQTT). Youtube, São Paulo, 8 de agosto de 2018. Disponível em:<<https://www.youtube.com/watch?v=nSCJfK3BRCQ>>. Acesso em: 30 de setembro de 2021.

ESP32 e MQTT DASH: controle e monitoramento através de um dashboard MQTT para Android. Filipeflop, São Paulo, 23 de julho de 2019. Disponível



em:<<https://www.filipeflop.com/blog/esp32-e-mqtt-dashboard-android/>>. Acesso em: 30 de setembro de 2021.

FLUXOGRAMA do Código de Implementação. Lucid Software Inc., Estados Unidos. Disponível

em:<[https://lucid.app/lucidchart/cf4b797e-065a-4caa-b655-af67f1349871/edit?viewport\\_loc=-1241%2C-224%2C5935%2C2964%2C0\\_0&invitationId=inv\\_7c6638d1-06fa-45ed-8523-9978a1b3baf](https://lucid.app/lucidchart/cf4b797e-065a-4caa-b655-af67f1349871/edit?viewport_loc=-1241%2C-224%2C5935%2C2964%2C0_0&invitationId=inv_7c6638d1-06fa-45ed-8523-9978a1b3baf)>. Acesso em: 15 de novembro de 2021.

TUTORIAL Fritzing 02. Circuito com NodeMCU ESP8266 y Ultrasonico HC-SR04. Youtube, São Paulo, 1 de abril de 2020. Disponível em:<<https://www.youtube.com/watch?v=WW9srEZkUl8>>. Acesso em: 15 de novembro de 2021.