

MQTT Dash



Prof. Michel Chaparro

IoT

MQTT Dash

mqtt dash



MQTT Dash (IoT, Smart Home)

Instalado

Abrir

4,8★

5 mil avaliações



Classificação Livre

Mais de 100 mil

Downloads

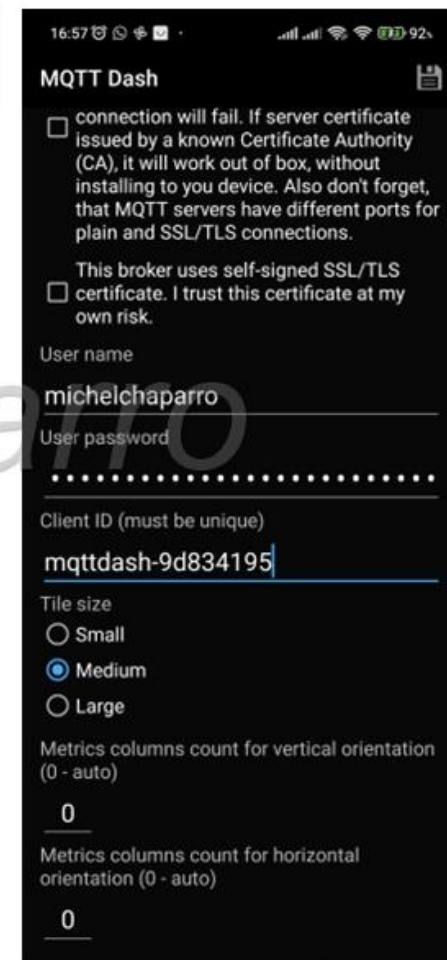
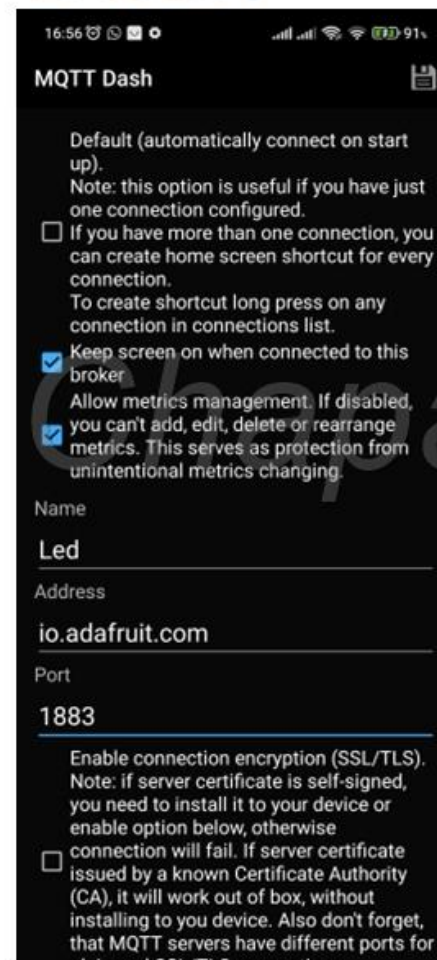
Controle e exibir dados de MQTT habilitado dispositivos e aplicativos (Internet das coisas, o Smart Home)



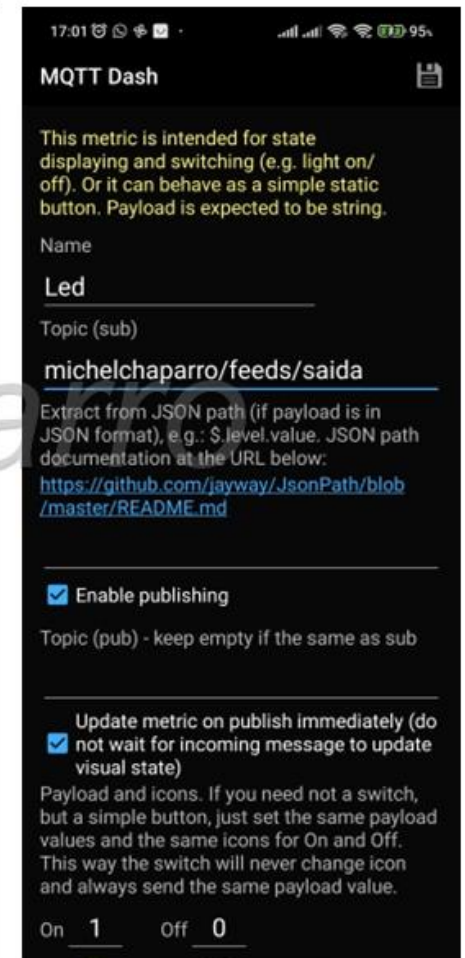
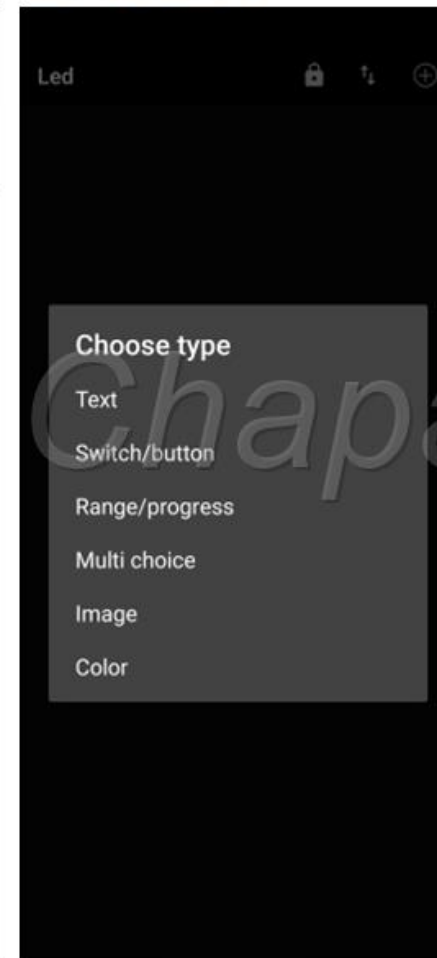
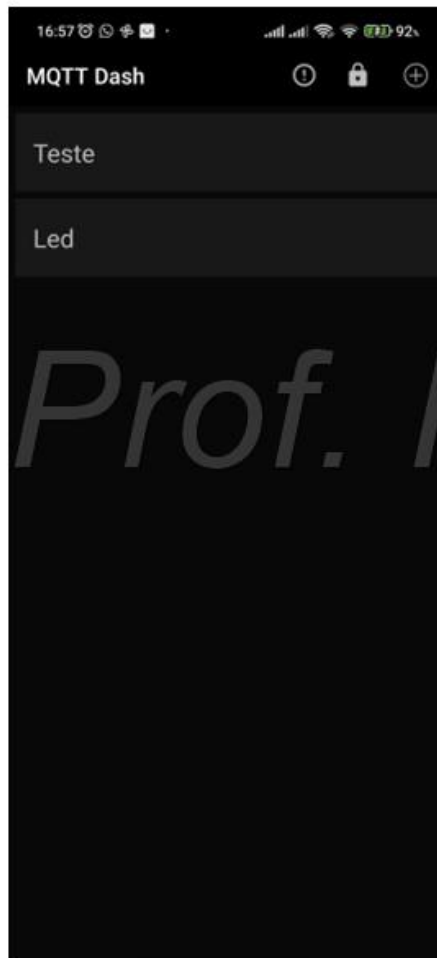
My Watermark (unregistered version)

Prof. Michel Chaparro

MQTT Dash



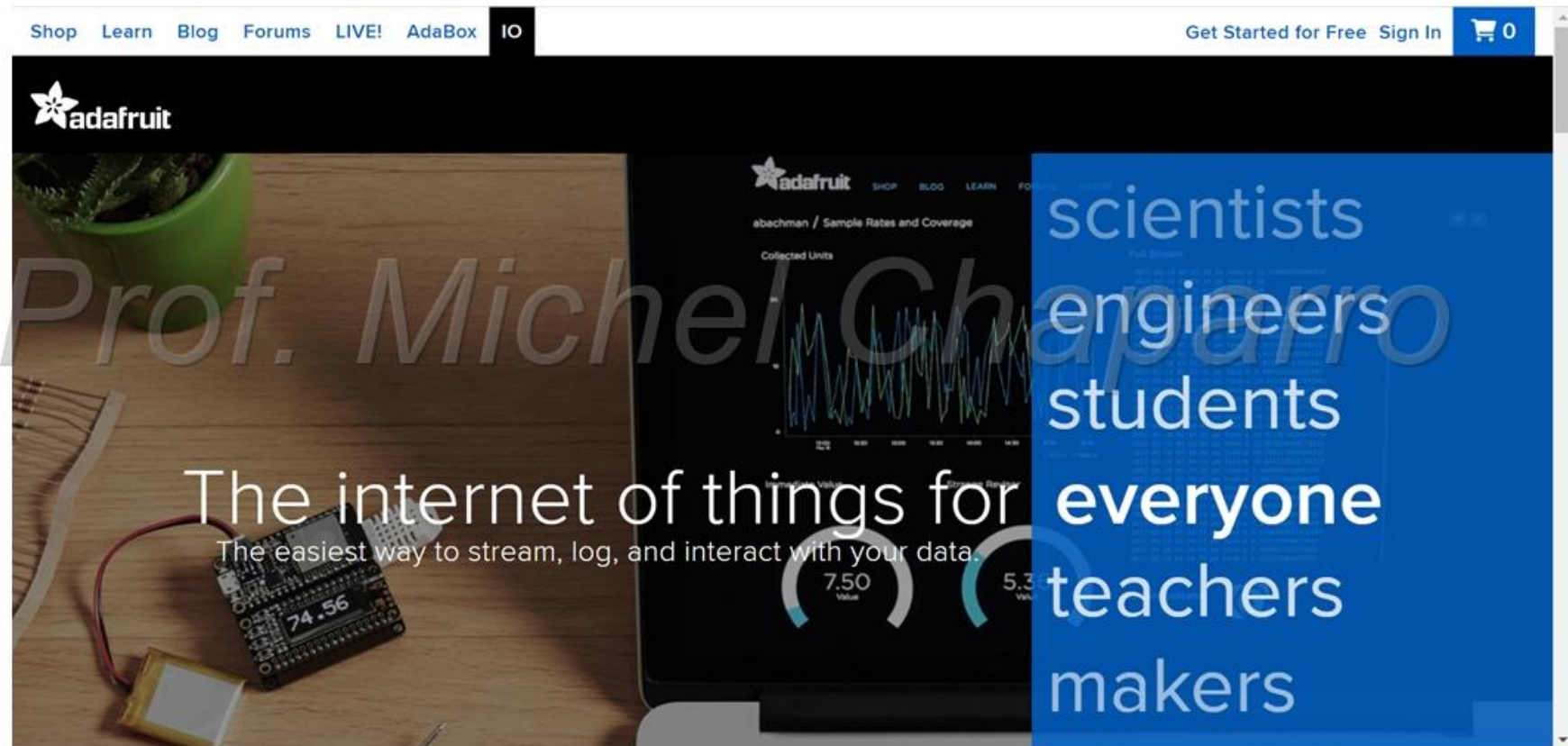
MQTT Dash



MQTT Dash



Plataforma IOT



Plataforma IOT

Acessar a Plataforma:

Prof. Michel Chaparro
<https://io.adafruit.com/>

Plataforma IOT

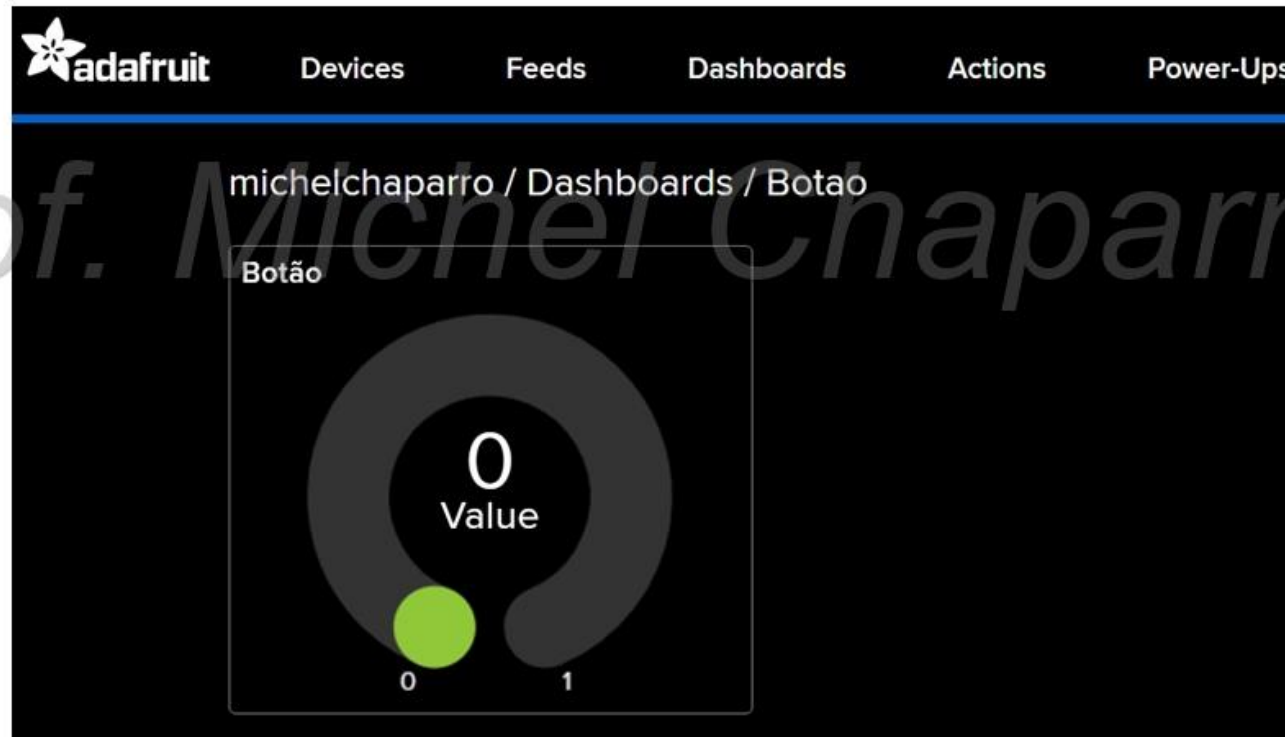
Criar um Feed → digital

Criar uma Dashboard → botao

Prof. Michel Chaparro

Plataforma IOT

Editar Dashboard:



Plataforma IOT

API Key:

YOUR ADAFRUIT IO KEY ×

Your Adafruit IO Key should be kept in a safe place and treated with the same care as your Adafruit username and password. People who have access to your Adafruit IO Key can view all of your data, create new feeds for your account, and manipulate your active feeds.



If you need to regenerate a new Adafruit IO Key, all of your existing programs and scripts will need to be manually changed to the new key.

Username

Active Key

REGENERATE KEY

Programação

```
#include "config.h"
```

```
#define BUTTON_PIN 4 // pino 5
```

```
// estados dos botões
```

```
bool current = false;
```

```
bool last = false;
```

```
AdafruitIO_Feed *digital = io.feed("digital");
```

Programação

```
void setup()  
{  
  pinMode(BUTTON_PIN, INPUT);  
  Serial.begin(115200);  
  while(! Serial);  
  Serial.print("Connecting to Adafruit IO");  
  io.connect();  
}
```

Programação

```
while(io.status() < AIO_CONNECTED)
{
    Serial.print(".");
    delay(500);
}

Serial.println();
Serial.println(io.statusText());
}
```


Programação

```
void loop()
{
    io.run();

    if(digitalRead(BUTTON_PIN) == LOW)
        current = true;
    else
        current = false;

    if(current == last)
        return;
```

Programação

```
Serial.print("sending button -> ");  
Serial.println(current);  
digital->save(current);
```

```
    last = current;  
}
```

Arquivo config.h

```
// configuração Adafruit
#define IO_USERNAME  "michelchaparro"
#define IO_KEY       "aio_RcEK44LxGbqjv4MlopMEccvk3gzu"

// configuração wifi
#define WIFI_SSID "IOT"
#define WIFI_PASS "IOT20222"

#include "AdafruitIO_WiFi.h"
```

Arquivo config.h

```
#if defined(USE_AIRLIFT) || defined(ADAFRUIT_METRO_M4_AIRLIFT_LITE) || \
    defined(ADAFRUIT_PYPORTAL)
// Configure the pins used for the ESP32 connection
#if !defined(SPIWIFI_SS) // if the wifi definition isnt in the board variant
// Don't change the names of these #define's! they match the variant ones
#define SPIWIFI SPI
#define SPIWIFI_SS 10 // Chip select pin
#define NINA_ACK 9 // a.k.a BUSY or READY pin
#define NINA_RESETN 6 // Reset pin
#define NINA_GPIO0 -1 // Not connected
#endif
AdafruitIO_WiFi io(IO_USERNAME, IO_KEY, WIFI_SSID, WIFI_PASS, SPIWIFI_SS,
                  NINA_ACK, NINA_RESETN, NINA_GPIO0, &SPIWIFI);
#else
AdafruitIO_WiFi io(IO_USERNAME, IO_KEY, WIFI_SSID, WIFI_PASS);
#endif
```