

ESP32 to The Moon

Digital Internet of Things Course

Chiang Mai Technical College

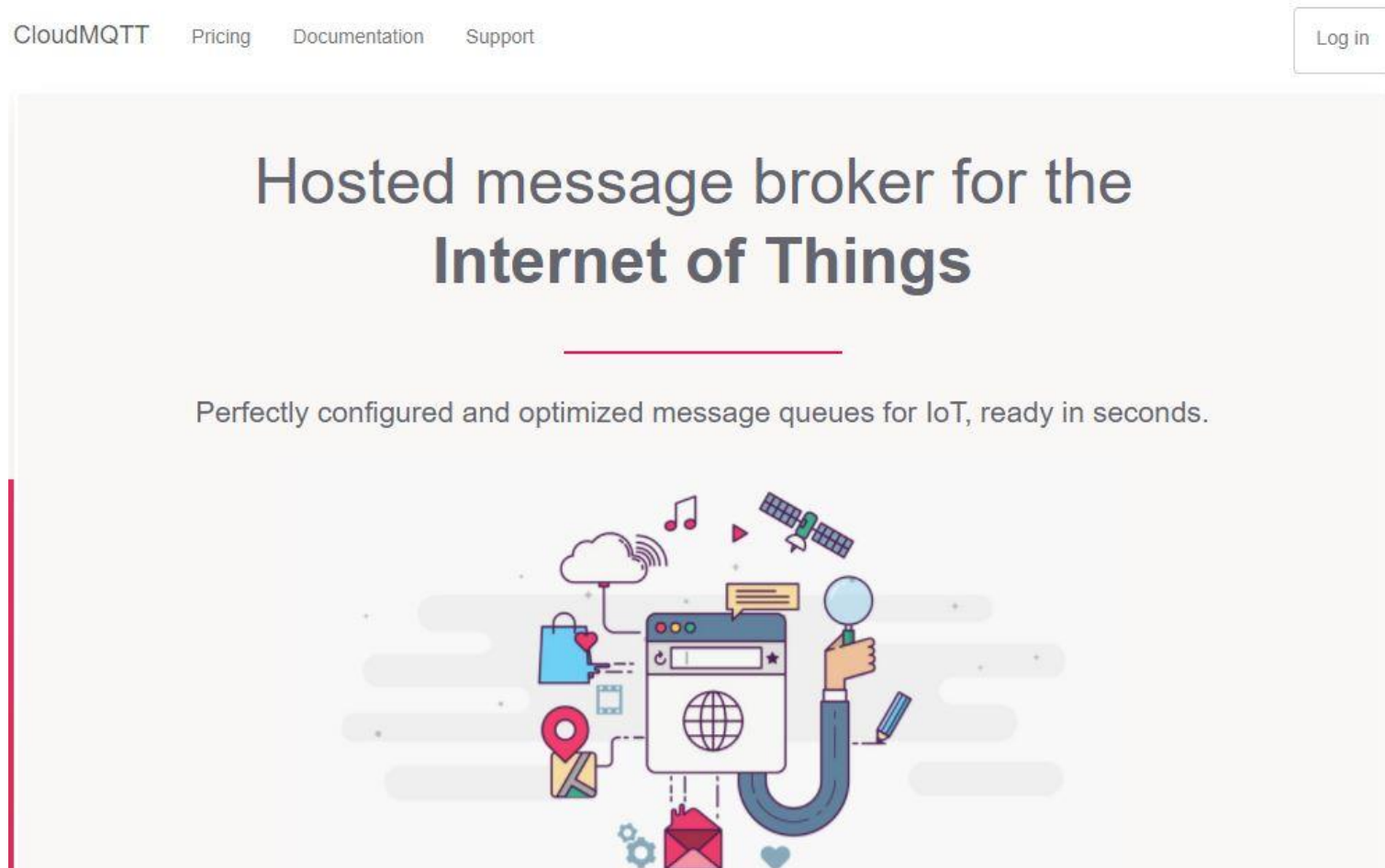


Introduce

- ESP32 to [Thingspeak.com](https://thingspeak.com)
- ESP32 to [CloudMqtt.com](https://cloudmqtt.com)
- ESP32 to [Line Notification](#)



ESP32 to CloudMqtt.com



ESP32 to CloudMqtt.com



Log in


☐ Remember me [Forgot your password?](#)

Create an account

Authenticate through a third-party service



ESP32 to CloudMqtt.com

 **CloudMQTT** List all instances ▼ apiruk326@gmail.com ▼

Create new instance

No credit card Please [add a credit card](#) if you want to subscribe to a paid plan


Name

Plan

Cute Cat (Free) ▼


Data center

US-East-1 (Northern Virginia) ▼


powered by
amazon
web services

Create New Instance

Plan


Cute Cat

See the [plan page](#) to learn about the different plans.

ESP32 to CloudMqtt.com



List all instances ▼

 apiruk326@gmail.com ▼

Instances

[+ Create New Instance](#)


Name	Plan	Datacenter	Actions
<u>Apirak Sang-ngenchai</u>	Cat	Amazon Web Services US-East-1 (Northern Virginia)	Edit




CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com

 **CloudMQTT**

Apirak Sang-ngenchai ▾

 apiruk326@gmail.com ▾

DETAILS USERS BRIDGES AMAZON KINESIS STREAM LOG WEBSOCKET UI

Details

[Statistics ↗](#)

Server

m12.cloudmqtt.com

User

qoknbhqq

↻ Restart

Password

jwxt_5n9NN70

Port

10970

SSL Port

20970


Websockets Port (TLS only)

30970

Connection limit

10

Active Plan




Cute Cat

Upgrade Instance



ESP32 to CloudMqtt.com

	Server	m12.cloudmqtt.com
	User	qoknbhgg <button>Restart</button>
	Password	jwxt_5n9NN70
	Port	10970
	SSL Port	20970
	Websockets Port (TLS only)	30970
	Connection limit	10



ESP32 to CloudMqtt.com



Apirak Sang-ngenchai ▾



apiruk326@gmail.com ▾

[DETAILS](#) [USERS](#) [BRIDGES](#) [AMAZON KINESIS STREAM](#) [LOG](#) [WEBSOCKET UI](#)

Users and ACL

Users

Name

Password

Create

Name

esp32Tester

Delete



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

Name

esp32Tester

Delete



ESP32 to CloudMqtt.com

ACLs

Note:

- You have to set a acl rule for a custom user before it can access anything
- Use # for multi level wildcard ACL
- Use + for single level wildcard ACL

For API docs look at HTTP API

User	Topic	Read	Write	
esp32Tester	/esp32/#	true	true	Delete

esp32Teste ▼

Topic

☐ Read Access?

☐ Write Access?

+ Add



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

ACLs

User	Topic	Read
------	-------	------

New Rule

User

Topic

Read Access? ☐

Write Access? ☐


Save




ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

User	Topic	Read	Write	
esp32Tester	/esp32/#	true	true	<div>Delete</div>

ESP32 to CloudMqtt.com

 **CloudMQTT**

Apirak Sang-ngenchai ▾

 apiruk326@gmail.com ▾

DETAILS USERS BRIDGES AMAZON KINESIS STREAM LOG **WEBSOCKET UI**

Websocket

Send message

Topic

Message

Send

Received messages

Topic	Message
-------	---------



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

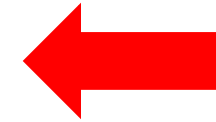


ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

```
esp32Cloudmqtt | Arduino 1.8.5
File Edit Sketch Tools Help

esp32Cloudmqtt $

1 #include <WiFi.h>
2 #include <PubSubClient.h>
3
4 #define WIFI_STA_NAME "xxxxxxxx"
5 #define WIFI_STA_PASS "xxxxxxxx"
6
7 #define MQTT_SERVER "xxxxxxxx"
8 #define MQTT_PORT xxxx
9 #define MQTT_USERNAME "xxxxxxxx"
10 #define MQTT_PASSWORD "xxxxxxxx"
11 #define MQTT_NAME "ESP32_CMTC"
12
13 #define LED_PIN 12
```



ESP32 to CloudMqtt.com

```
7 #define MQTT_SERVER "xxxxxxxx"
8 #define MQTT_PORT xxxx
9 #define MQTT_USERNAME "xxxxxxxx"
10 #define MQTT_PASSWORD "xxxxxxxx"
11 #define MQTT_NAME "ESP32_CMTC"
```

Server m12.cloudmqtt.com

User qoknbhgg

Restart

Password jwxt_5n9NN70

Port 10970



CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com



Apirak Sang-ngenchai ▾

 apiruk326@gmail.com ▾

[DETAILS](#) [USERS](#) [BRIDGES](#) [AMAZON KINESIS STREAM](#) [LOG](#) [WEBSOCKET UI](#)

Websocket

Send message

Topic

Message

Send

Received messages

Topic

Message

/esp32/humid

44

/esp32/temp

27

/esp32/humid

44

/esp32/temp

27

/esp32/humid

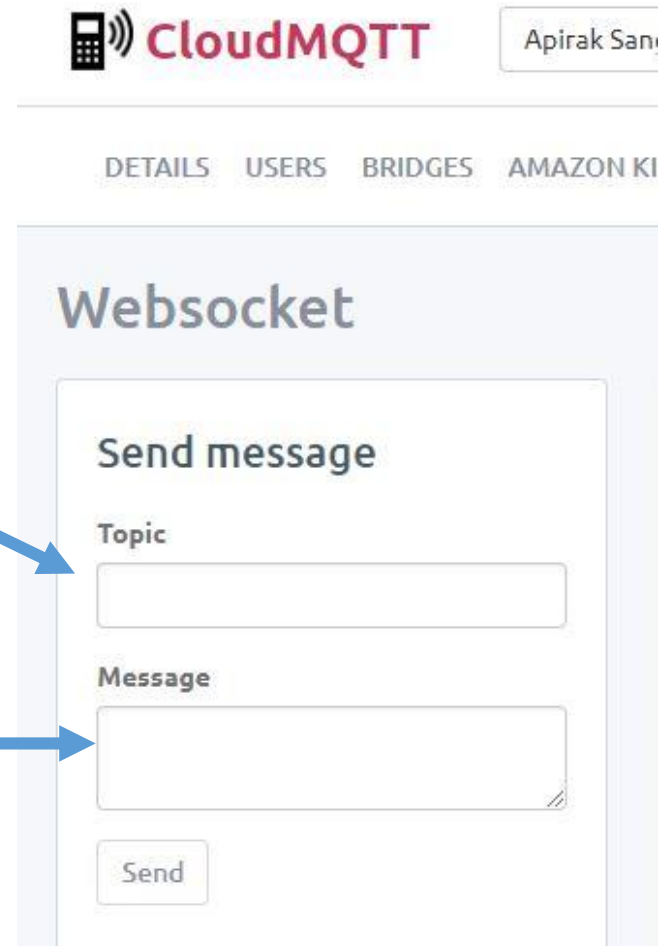
44



ESP32 to CloudMqtt.com

/esp32/LED

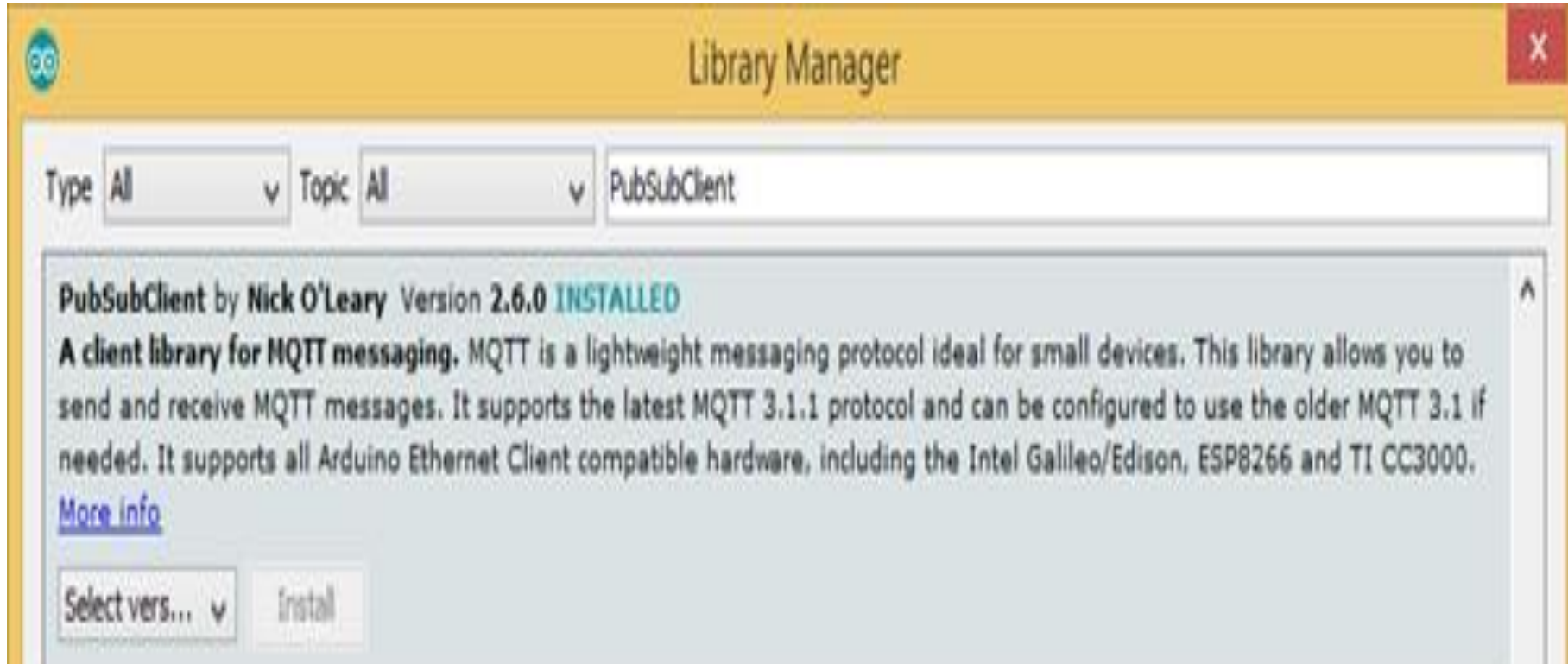
ON & OFF



The screenshot shows the CloudMQTT website interface. At the top, there is a logo with a mobile phone icon and the text "CloudMQTT". To the right of the logo is a user profile box containing the name "Apirak San". Below the logo, there is a navigation bar with links: "DETAILS", "USERS", "BRIDGES", and "AMAZON KI". The main content area is titled "Websocket". Inside this section, there is a "Send message" form. The form has two input fields: "Topic" and "Message". Below these fields is a "Send" button. A blue arrow points from the text "/esp32/LED" to the "Topic" input field. Another blue arrow points from the text "ON & OFF" to the "Message" input field.



ESP32 to CloudMqtt.com



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

```
52 mqtt.setServer(MQTT_SERVER, MQTT_PORT);  
53 mqtt.setCallback(callback);
```



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

```
56 void loop() {  
57     if (mqtt.connected() == false) {  
58         Serial.print("MQTT connection... ");  
59         if (mqtt.connect(MQTT_NAME, MQTT_USERNAME, MQTT_PASSWORD)) {  
60             Serial.println("connected");  
61             mqtt.subscribe("/esp32/LED");  
62         } else {  
63             Serial.println("failed");  
64             delay(5000);  
65         }  
66     } else {  
67         // coding here  
68         mqtt.loop();  
69     }
```



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)



```
66     } else {  
67         // coding here  
68         mqtt.publish("/esp32/text", "CMTC");  
69         mqtt.loop();  
70     }
```



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

```
18 void callback(char* topic, byte* payload, unsigned int length) {  
19     payload[length] = '\0';  
20     String topic_str = topic, payload_str = (char*)payload;  
21     Serial.println "[" + topic_str + "]: " + payload_str;  
22  
23     digitalWrite(LED_PIN, (payload_str == "ON") ? HIGH : LOW);  
24     digitalWrite(LED_BUILTIN, (payload_str == "ON") ? HIGH : LOW);  
25 }
```



CHIANG MAI
MAKERCLUB



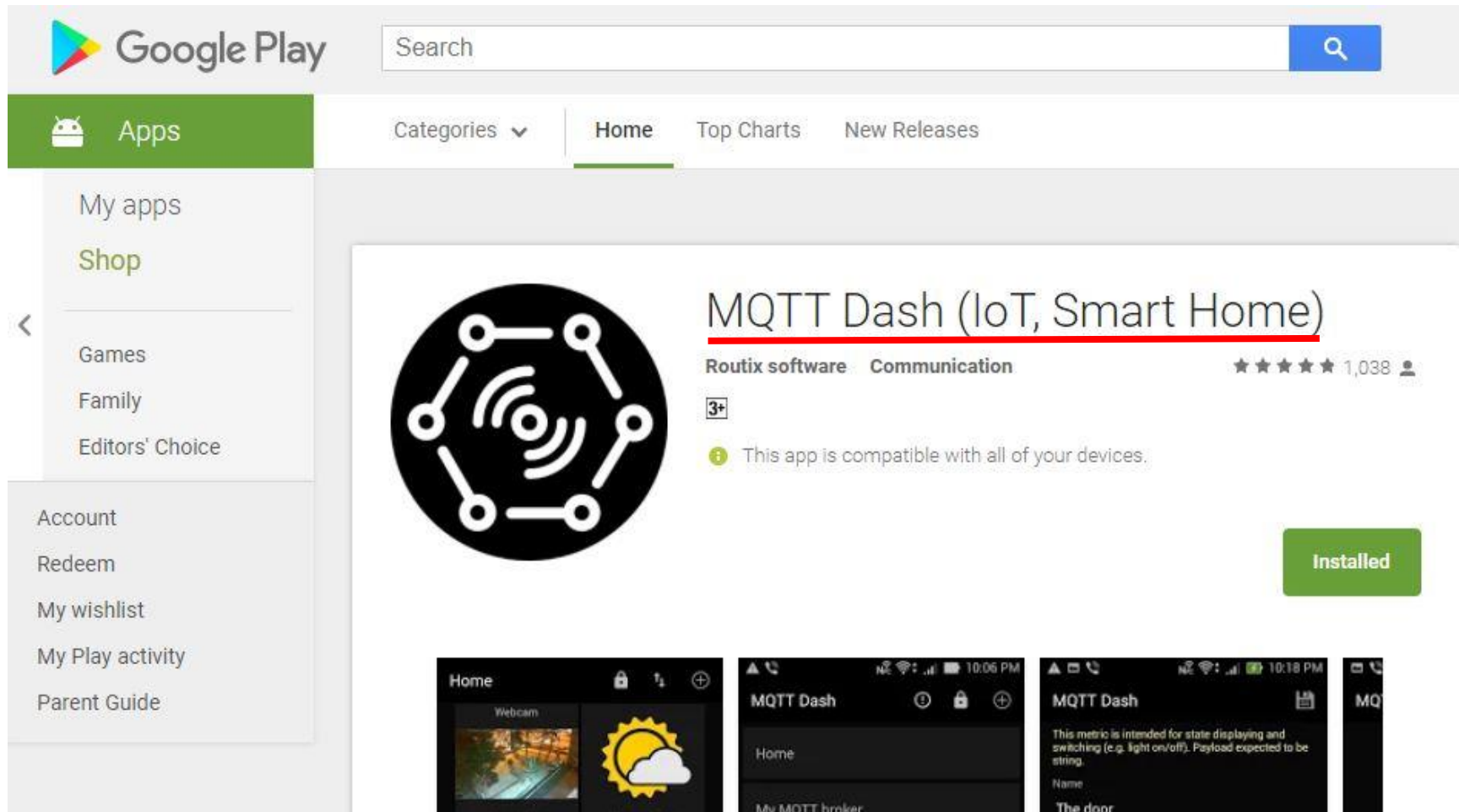
ESP32 to [CloudMqtt.com](https://cloud.mqtt.com)

Quick test

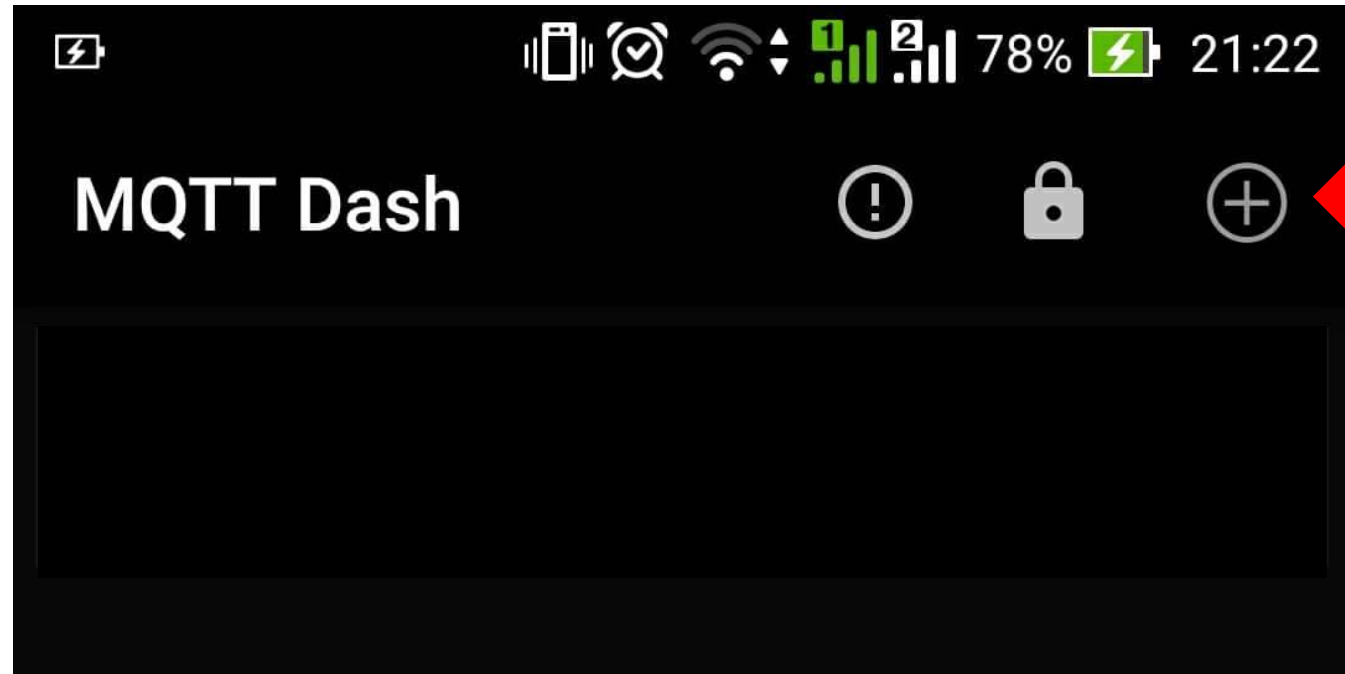
จงเขียนโปรแกรมอ่านค่าอุณหภูมิ และความชื้น จากโมดูล
เซนเซอร์ DHT อัดโหลดข้อมูลไปยัง CloudMQTT



ESP32 to CloudMqtt.com



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)



ESP32 to CloudMqtt.com

MQTT Dash

Default (automatically connect on start up).
Note: this option is useful if you have just one connection configured.

☐ If you have more than one connection, you can create home screen shortcut for every connection.
To create shortcut long press on any connection in connections list.

☒ Keep screen on when connected to this broker

Allow metrics management. If disabled, you can't add, edit, delete or rearrange metrics. This serves as protection from unintentional metrics changing.

☒

Name

Address

Port

1883

Enable connection encryption (SSL/TLS).
Note: if server certificate is self-signed, you need to install it to your device or enable option below, otherwise connection will fail. If server certificate issued by a known Certificate Authority (CA), it will work out of box, without installing to your device. Also don't forget that

☐

Server m12.cloudmqtt.com

User qoknbhgg Restart

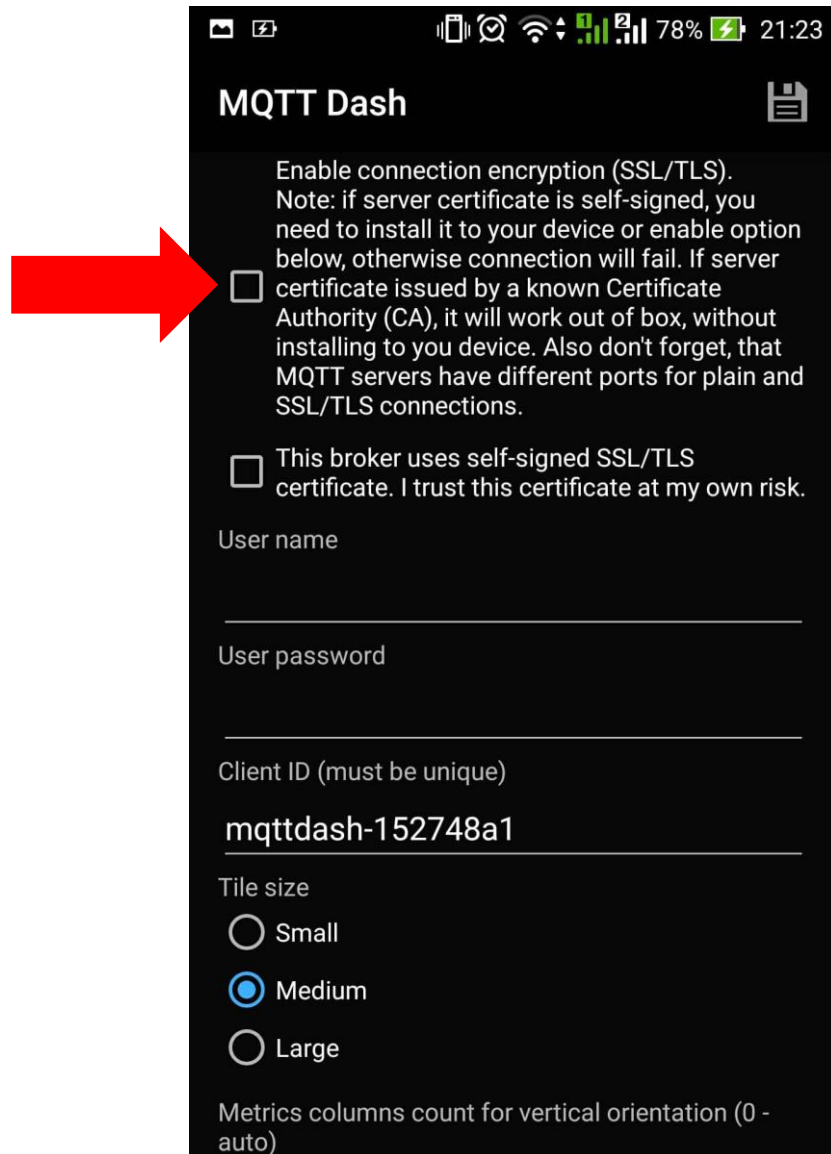
Password jwxt_5n9NN70

Port 10970

SSL Port 20970



ESP32 to CloudMqtt.com



MQTT Dash

Enable connection encryption (SSL/TLS).
Note: if server certificate is self-signed, you need to install it to your device or enable option below, otherwise connection will fail. If server certificate issued by a known Certificate Authority (CA), it will work out of box, without installing to you device. Also don't forget, that MQTT servers have different ports for plain and SSL/TLS connections.

☐ This broker uses self-signed SSL/TLS certificate. I trust this certificate at my own risk.

User name

User password

Client ID (must be unique)

mqttdash-152748a1

Tile size

☐ Small

☒ Medium

☐ Large

Metrics columns count for vertical orientation (0 - auto)

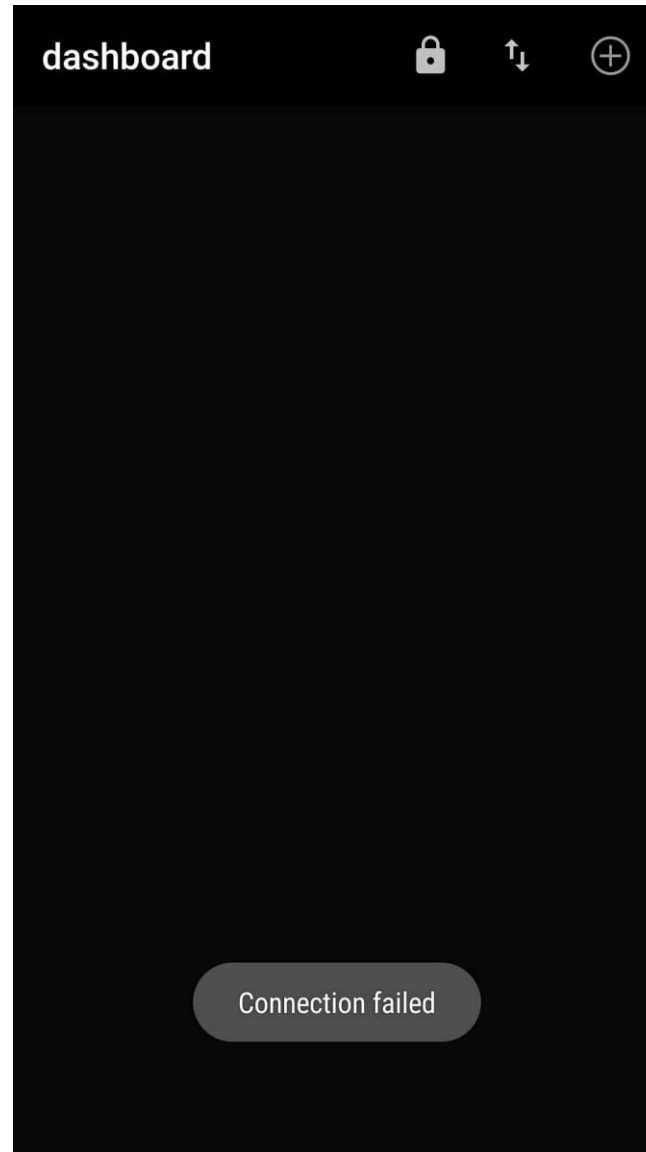


Name

esp32Tester



ESP32 to CloudMqtt.com



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

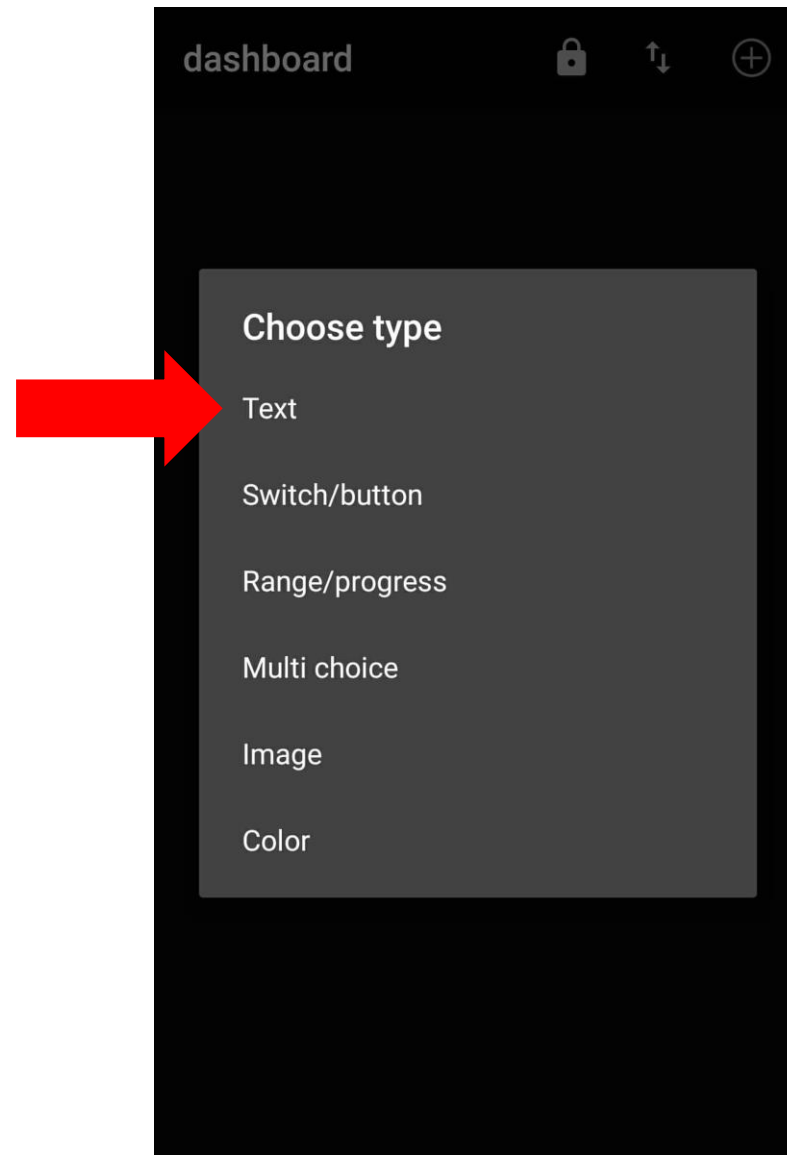
Connection failed




CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)



ESP32 to CloudMqtt.com



MQTT Dash

This metric is intended for displaying payload text (e.g. temperature displaying). Payload is expected to be string.

Name

Message

Topic (sub)

/esp32/text

Extract from JSON path (if payload is in JSON format), e.g.: \$.level.value. JSON path documentation at the URL below:
<https://github.com/jayway/JsonPath/blob/master/README.md>

☒ Enable publishing

Topic (pub) - keep empty if the same as sub

☒ Update metric on publish immediately (do not wait for incoming message to update visual state)

Prefix Postfix



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)

44%

14:23

MQTT Dash

Main text size

☐ Small

☐ Medium

☒ Large

Main text color

Other settings

☒ QoS(0)

☐ QoS(1)

☐ QoS(2)

☐ Retained

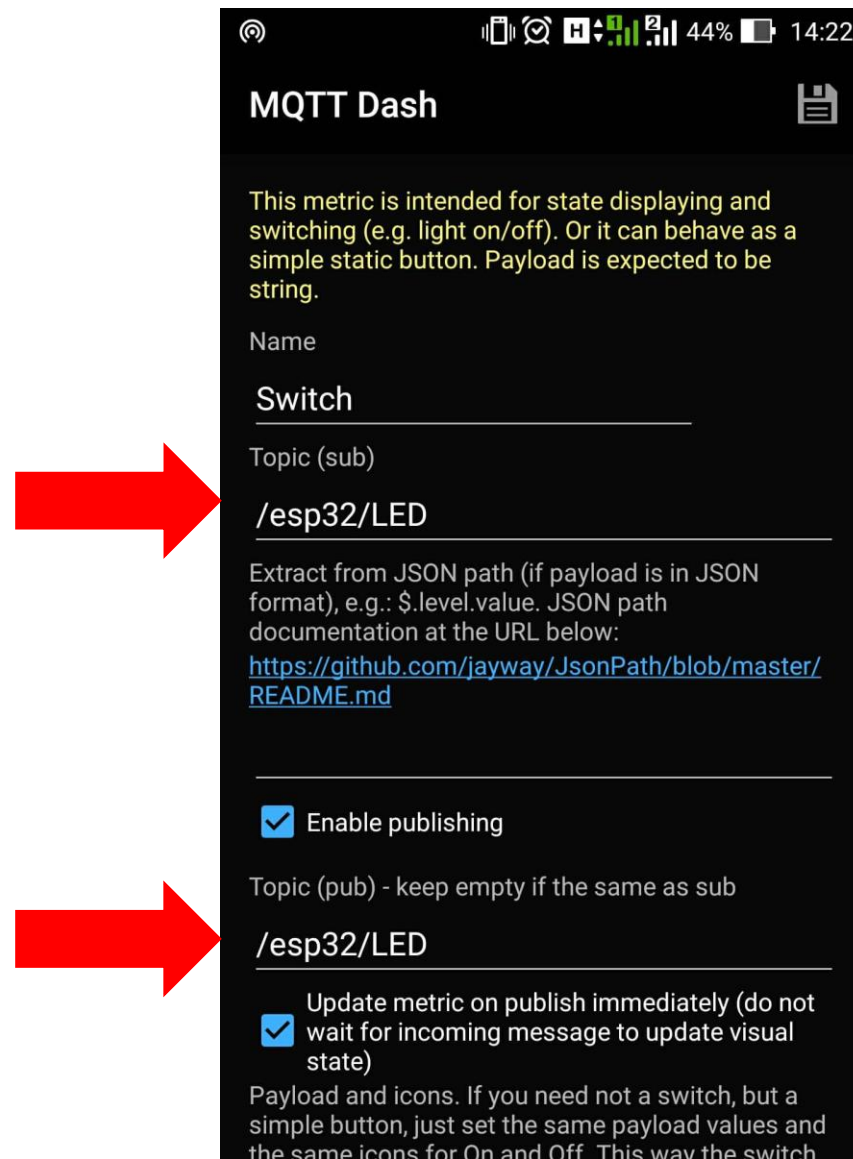
Blink tile to draw attention, if the expression evaluates to 'true'.
Expression can be any valid JavaScript expression which evaluates to boolean (true|false).
You can use 'val' and 'secs' constants in your expression.
'val' contains either, received raw payload or extracted from JSON raw value, if JSON path specified.
'secs' contains numbers of seconds since last



CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com



MQTT Dash

This metric is intended for state displaying and switching (e.g. light on/off). Or it can behave as a simple static button. Payload is expected to be string.

Name

Switch

Topic (sub)

/esp32/LED

Extract from JSON path (if payload is in JSON format), e.g.: \$.level.value. JSON path documentation at the URL below:
<https://github.com/jayway/JsonPath/blob/master/README.md>

☒ Enable publishing

Topic (pub) - keep empty if the same as sub

/esp32/LED

☒ Update metric on publish immediately (do not wait for incoming message to update visual state)

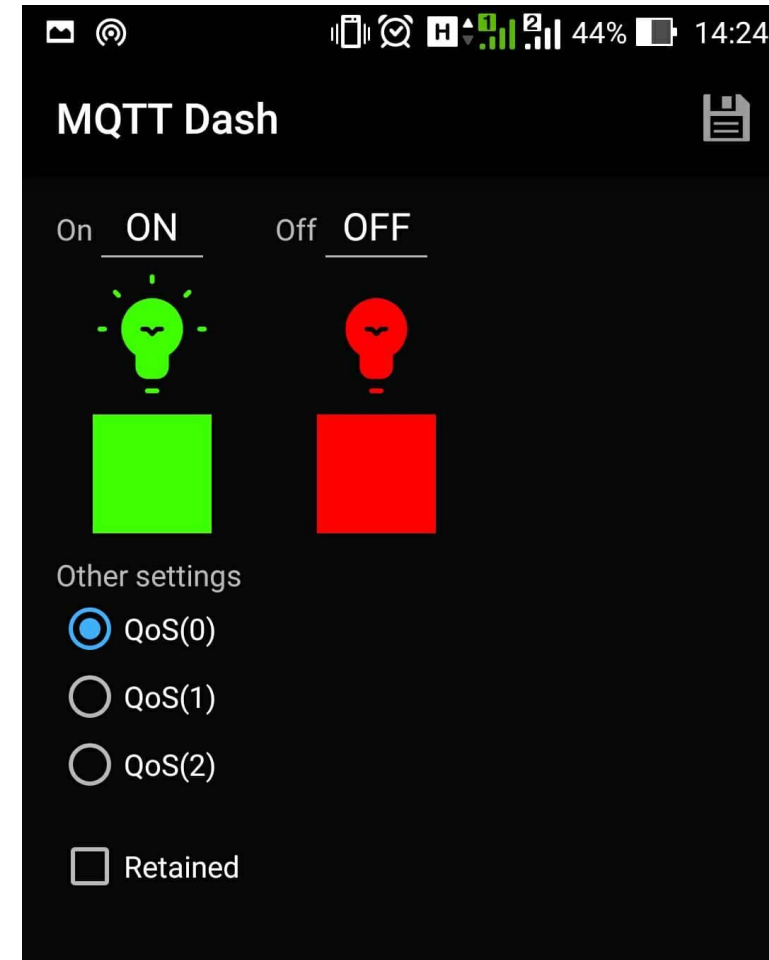
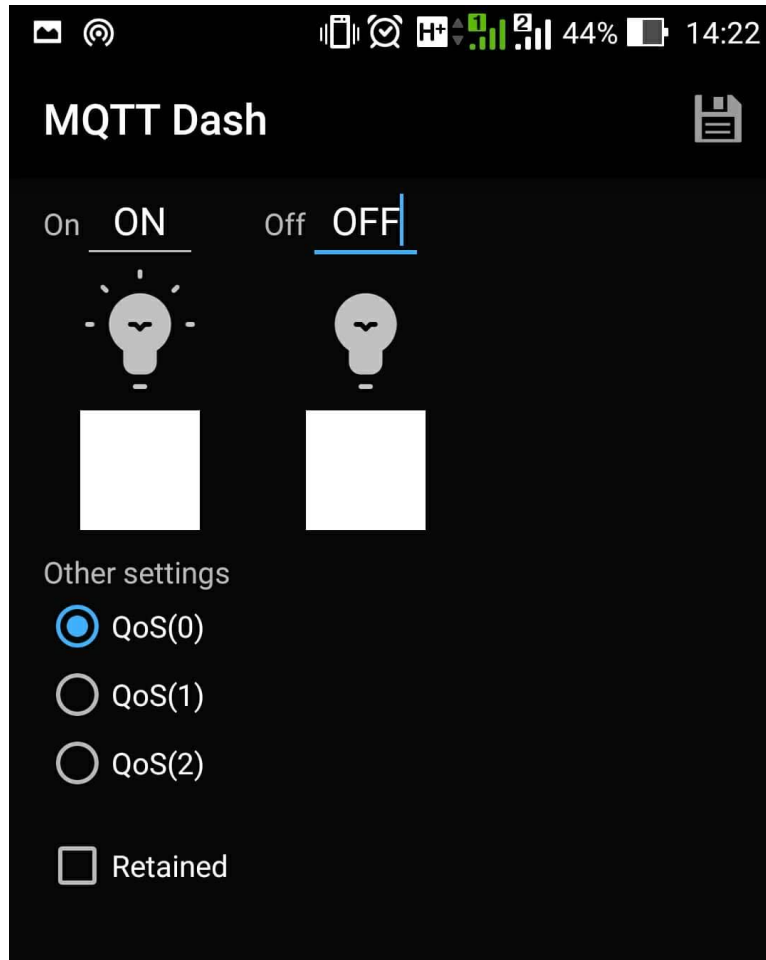
Payload and icons. If you need not a switch, but a simple button, just set the same payload values and the same icons for On and Off. This way the switch




CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)



ESP32 to CloudMqtt.com



MQTT Dash

This metric is intended for numeric progress/range displaying and changing (e.g. thermostat controlling, temperature displaying). Payload is expected to be string.

Name

Temperature

Topic (sub)

/esp32/temp

Extract from JSON path (if payload is in JSON format), e.g.: \$.level.value. JSON path documentation at the URL below:
<https://github.com/jayway/JsonPath/blob/master/README.md>

☒ Enable publishing

Topic (pub) - keep empty if the same as sub

☒ Update metric on publish immediately (do not wait for incoming message to update visual state)

Min 0.0 Max 100.0



CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com

MQTT Dash

Min Max

Prefix Postfix

Precision

☒ Display payload value instead of percentage

Progress color

Other settings

☒ QoS(0)

☐ QoS(1)

☐ QoS(2)

☐ Retained


Blink tile to draw attention, if the expression evaluates to 'true'.
Expression can be any valid JavaScript expression which evaluates to boolean (true/false).
You can use 'val' and 'secs' constants in your expression



CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com



MQTT Dash

This metric is intended for numeric progress/range displaying and changing (e.g. thermostat controlling, temperature displaying). Payload is expected to be string.

Name

Humidity

Topic (sub)

/esp32/humid

Extract from JSON path (if payload is in JSON format), e.g.: \$.level.value. JSON path documentation at the URL below:
<https://github.com/jayway/JsonPath/blob/master/README.md>

☒ Enable publishing

Topic (pub) - keep empty if the same as sub

☒ Update metric on publish immediately (do not wait for incoming message to update visual state)


Min 0.0 Max 100.0



CHIANG MAI
MAKERCLUB



ESP32 to CloudMqtt.com



MQTT Dash

This metric is intended for state displaying and switching (e.g. light on/off). Or it can behave as a simple static button. Payload is expected to be string.

Name
Warning

Topic (sub)
/esp32/warning

Extract from JSON path (if payload is in JSON format), e.g.: \$.level.value. JSON path documentation at the URL below:
<https://github.com/jayway/JsonPath/blob/master/README.md>

☒ Enable publishing

Topic (pub) - keep empty if the same as sub

☒ Update metric on publish immediately (do not wait for incoming message to update visual state)

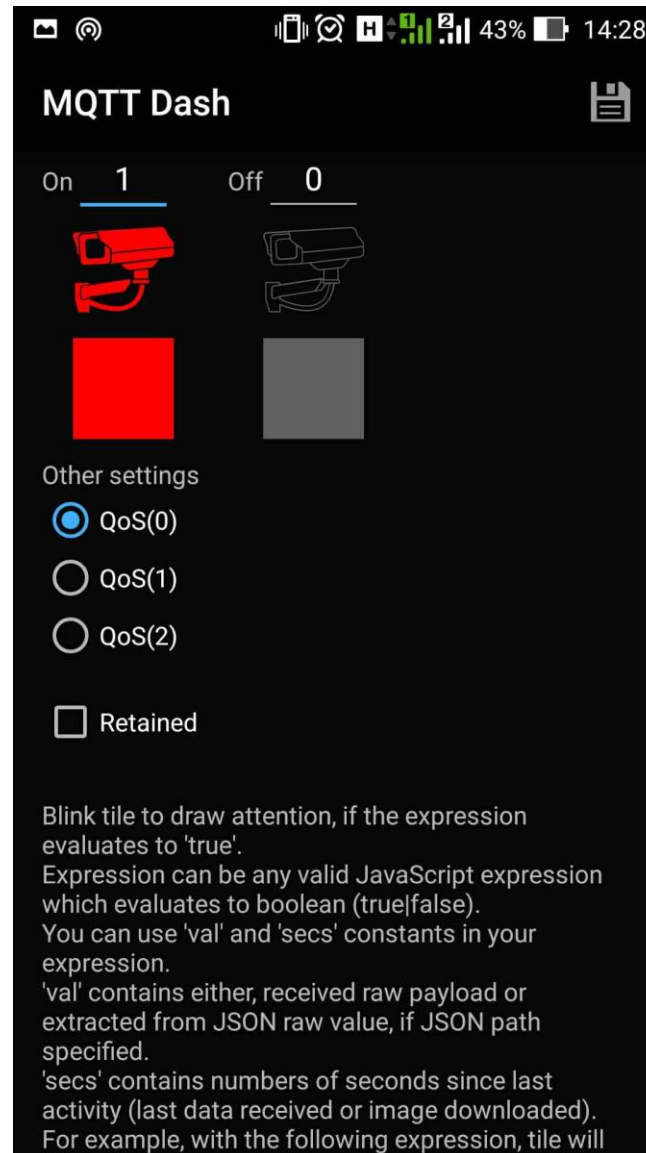
Payload and icons. If you need not a switch, but a simple button, just set the same payload values and the same icons for On and Off. This way the switch



CHIANG MAI
MAKERCLUB



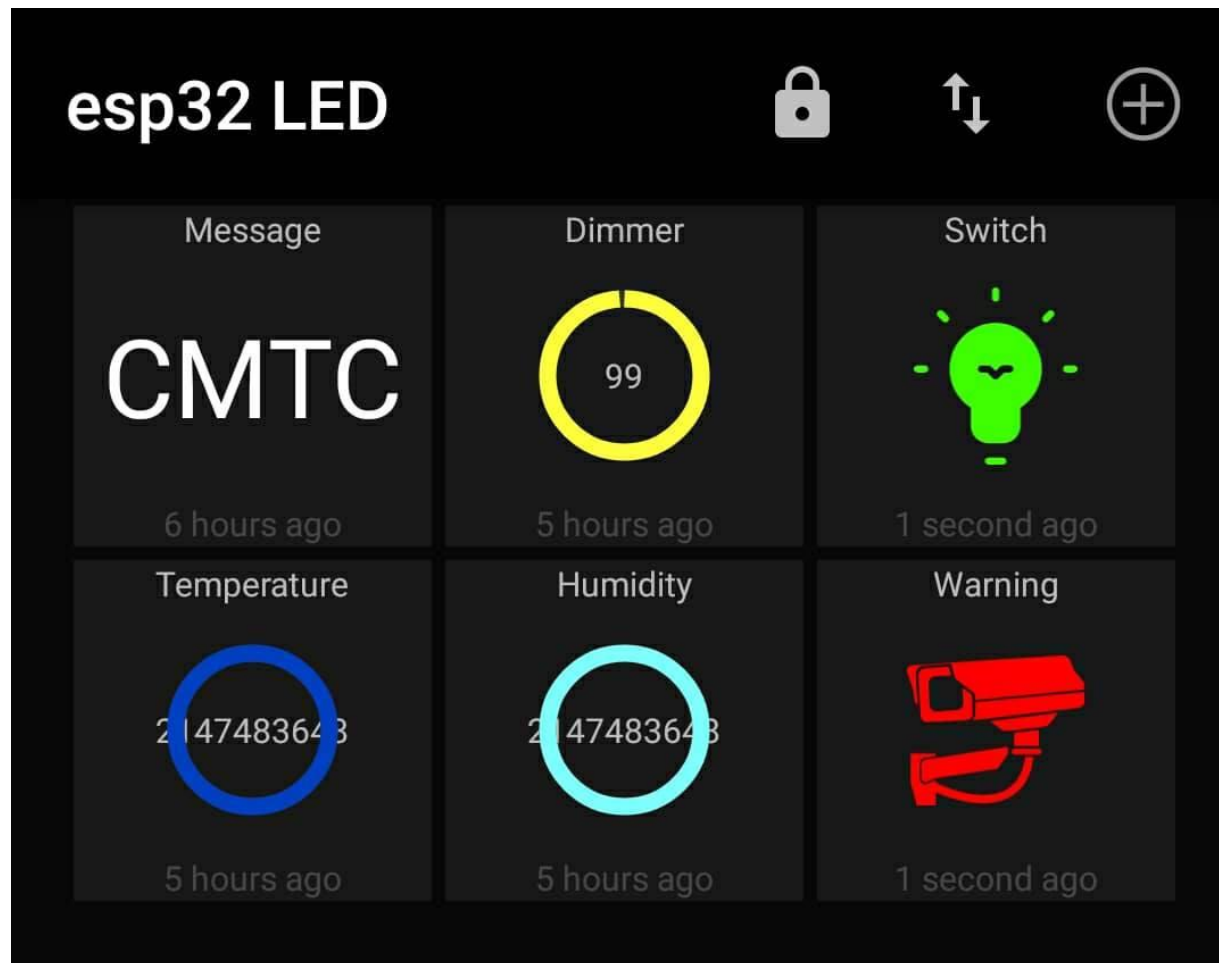
ESP32 to CloudMqtt.com



CHIANG MAI
MAKERCLUB



ESP32 to [CloudMqtt.com](https://cloudmqtt.com)



CHIANG MAI

MAKERCLUB

PLAY WITH US

JOIN US

www.cmmakerclub.com

