

การใช้งาน ESP8266 ควบคุม LED ผ่าน MQTT โดยใช้ Arduino IDE (C++)

HTML Content

ในบทความนี้เราจะมาใช้ MQTT ในเป็นโปรโตคอลตัวกลางในการสื่อสารระหว่างคอมพิวเตอร์ หรืออุปกรณ์ต่างๆ ผ่านอินเทอร์เน็ต ไปที่ ESP8266 กันครับ หลังจากอ่านบทความนี้จบแล้ว ท่านสามารถจะนำ MQTT ไปใช้งานอื่นๆได้อีกมากมาย เช่น การสื่อสารกันเองระหว่าง ESP8266 กับ ESP8266 ในการแชร์ข้อมูลซึ่งกันและกัน มอนิเตอร์อุณหภูมิแบบเรียลไทม์ ติดตามการทำงานของเครื่องจักรแบบไร้สาย นอกจากนี้ยังสามารถนำไปประยุกต์ใช้อะไรที่ไม่เกี่ยวข้องกับ ESP8266 อย่างการทำห้องแชทออนไลน์ได้อีกด้วย

รู้จักกับ MQTT

MQTT เป็นโปรโตคอลที่ใช้สำหรับการสื่อสาร ต่างภาษา ต่างอุปกรณ์กัน หรือภาษาเดียวกัน อุปกรณ์เดียวกัน ซึ่งมีไลบรารีให้พร้อมสำหรับภาษาต่างๆ ได้แก่ Heroku Ruby Python NodeJS Java Go .NET NodeMCU(Lua) และ PHP ทำงานอยู่บนฐานของโปรโตคอล TCP โดยปกติแล้ว MQTT จะสามารถใช้งานได้ก็ต่อเมื่อมีเซิร์ฟเวอร์ตัวกลางเท่านั้น ซึ่งท่านสามารถติดตั้งบนเซิร์ฟเวอร์ของตัวเองก็ได้ แต่ในบทความนี้จะเลือกใช้บริการเซิร์ฟเวอร์ MQTT จากเว็บ [CloudMQTT](https://cloudmqtt.com/) ซึ่งสามารถใช้งานได้ฟรี 10 การเชื่อมต่อ กรณีต้องการนำไปใช้งานจริงควรเผื่อจำนวนการเชื่อมต่อไว้มากๆหากนำไปใช้กับสาธารณะ

ก่อนที่จะใช้งานได้ เราจำเป็นต้องมารู้ศัพท์ที่ใช้ และหลักการใช้งานกันก่อนครับ ก่อนอื่นเลย Username Password ที่ใช้ล็อกอินเข้าใช้งานโปรโตคอลจะมีอยู่ด้วยกัน 2 ระดับ คือ ระดับหัวหน้า และระดับผู้ดูแลห้อง (เป็นชื่อระดับที่ผมตั้งขึ้นเองเพื่อให้เข้าใจได้ง่าย) ข้อแตกต่างของ 2 ระดับนี้คือ

- **ระดับหัวหน้า** เปรียบได้กับผู้บริหาร หรือเจ้าของบริษัท มีสิทธิ์ที่จะฟัง และประกาศ ไปยังห้อง (Topic) ไหนก็ได้ที่มีการสร้างขึ้น ซึ่ง Username Password จะถูกกำหนดมาแล้ว หลังจากสมัครสมาชิกในระบบของ CloudMQTT จะมีประกาศในหน้าแรก
- **ระดับผู้ดูแลห้อง** ระดับนี้มีสิทธิ์ที่จะฟังอย่างเดียว ประกาศอย่างเดียว หรือทั้งฟังและประกาศ ได้ในห้อง (Topic) ที่กำหนดไว้แล้วเท่านั้น ซึ่ง Username Password สามารถกำหนดได้เองโดยการสร้าง

ต่อมา ก็มาทำความรู้จักกับคำว่า **Topic** กันครับ คำนี้ผมจะขอใช้ว่าเป็น "ชื่อห้อง" น่าจะเข้าใจได้ง่ายกว่า ก่อนที่เราจะส่งข้อมูล หรือรอฟังข้อมูล เราจะต้องทราบชื่อห้องซะก่อนครับ เปรียบเสมือนว่าหากเราต้องการจะคุยกับกลุ่มไหน เราจำเป็นต้องรู้ชื่อกลุ่มซะก่อน ใน

ทาง MQTT จะเรียกชื่อห้องว่า **Topic** ซึ่งสามารถกำหนดเป็นอะไรก็ได้ แต่โดยส่วนใหญ่จะกำหนดในลักษณะคล้าย Path เช่น /ESP/LED /ESP/TEMP และอื่นๆ การตั้ง ควรเป็นชื่อที่สื่อความหมาย และเข้าใจได้ง่ายด้วยครับ

พอร์ต (Port) ใน CloudMQTT จะแบ่งพอร์ตออกเป็น 3 ชนิด ดังนี้

- **Port** เป็นพอร์ตที่โปรโตคอล MQTT ใช้งาน ใช้งานกับภาษาที่อยู่ในฝั่ง Back-end เช่น PHP C++ และภาษาอื่นๆที่ได้กล่าวไปแล้ว (พอร์ตที่จะให้ ESP8266 คือพอร์ตชนิดนี้)
- **SSL Port** เป็นพอร์ตที่โปรโตคอล MQTT ใช้งาน แต่จะถูกใช้เมื่อต้องการเชื่อมต่อแบบ SSL ซึ่งผมแนะนำให้ใช้พอร์ตนี้หากอุปกรณ์รองรับ SSL (ESP8266 ไม่ค่อยจะรองรับ จึงไม่ได้ใช้)
- **Websockets Port (TLS only)** เป็นพอร์ตที่ใช้กับหน้าเว็บ ซึ่งหน้าเว็บนั้นไม่รองรับโปรโตคอล MQTT หรือเชื่อมต่อ TCP โดยตรง แต่มีเทคโนโลยีเรียกว่าท่อนที่ออกแบบมาสำหรับเว็บอยู่แล้ว ชื่อ WebSocket ในการที่จะส่งข้อมูลผ่านหน้าเว็บจะต้องคอนฟิกพอร์ตเป็นพอร์ตชนิดนี้

Connection limit เป็นการจำกัดจำนวนผู้เข้ามาเชื่อมต่อ (Clients) ในวินาทีนั้นๆ ตัวอย่างเช่น เปิดหน้าเว็บที่เชื่อมต่ออยู่ไว้ 2 หน้า และมี ESP8266 ที่เชื่อมต่ออีก 1 ตัว รวมแล้วจะนับเป็น 3 การเชื่อมต่อ หากใช้งานฟรีเซิร์ฟเวอร์ MQTT จาก CloudMQTT จะถูกจำกัดไว้ที่ 10 การเชื่อมต่อ

สมัครใช้งานฟรีเซิร์ฟเวอร์ MQTT จาก CloudMQTT

เข้าไปที่ลิง <https://customer.cloudmqtt.com/login> ตรง Sign up กรอกอีเมลของท่านลงไปก่อน จากนั้นไปเช็คเมล ระบบจะส่งลิงสำหรับสมัครสมาชิกไปให้ จากนั้นคลิกลิงที่อยู่ในเมล แล้วกรอกรายละเอียดต่างๆเป็นอันเสร็จครับ (ผมไม่มีรูปภาพให้ดูเป็นขั้นตอนเนื่องจากได้สมัครสมาชิกไปแล้วครับ)

เลือก Plan ใช้งาน และเตรียมพร้อมก่อนเขียนโค้ด

เมื่อเข้าสู่ระบบแล้ว จะพบกับหน้าจัดการทั้งหมดประมาณรูปด้านล่างนี้ กดปุ่ม + Create

CloudMQTT Instances for @hotmail.com (Log out)

[Account](#) (Edit)@hotmail.com
N/A[PayPal](#) (Change)

Not set up.

[Invoices](#) (All)[+ Create](#)

Name

Plan

Region

You don't have any instances yet, do you want to [create](#) one?

จากนั้นตั้งชื่อ (ชื่ออะไรก็ได้ ไม่มีผลต่อโค้ด) ในข้อที่ 1 เลือกที่ตั้งของเซิร์ฟเวอร์ในข้อที่ 2 (มีให้เลือก US กับ EU ผมเลือก US) และสุดท้าย Plan กรณีต้องการใช้ฟรีให้คง Cute Cat ไว้ไม่ต้องแก้ หากต้องการใช้งานแบบเสียเงินจะต้องเชื่อม Paypal แล้วเลือก Plan ที่ลิง <https://www.cloudmqtt.com/plans.html> จากนั้นกดปุ่ม Create

Create new CloudMQTT Instance

PayPal not enabled. Please [enable PayPal](#) if you want to subscribe to a paid plan

Name

Name to describe your instance

1

Data center

Amazon US East (Northern Virginia)

2

Plan

Cute Cat

3

To learn more about the different plans please visit: www.cloudmqtt.com/plans.html[Create](#)[Cancel](#)

จากนั้นระบบจะพากลับมาหน้าเดิม ให้กดที่ปุ่ม Details

CloudMQTT Instances for max30012540@hotmail.com ([Log out](#))

[Account](#) ([Edit](#))
max30012540@hotmail.com
N/A

[PayPal](#) ([Change](#))
Not set up.

[Invoices](#) ([All](#))

[+ Create](#)

จากนั้นรายละเอียดต่างๆจะแสดงขึ้นมาในหัวข้อ Instance info (สามารถอ่านรายละเอียดได้ในหัวข้อที่แล้ว)

CloudMQTT Console

[Overview](#)[Websocket UI](#)[Server log](#)[Statistics](#)[Restart](#)

Instance info

Server	m12.cloudmqtt.com
User	
Password	
Port	
SSL Port	
Websockets Port (TLS only)	
Connection limit	10

ต่อมา เราจะมาสร้าง Username Password ระดับหัวหน้าห้องกันครับ เลื่อนมาที่หัวข้อ Manage Users แล้วตั้ง Username Password ที่ต้องการ หากทำครั้งแรกแนะนำให้ตั้งตามผมไปก่อนคือ User : TEST และ Pass : 12345 แล้วกดปุ่ม Save

Manage Users

TEST 1 2 Save

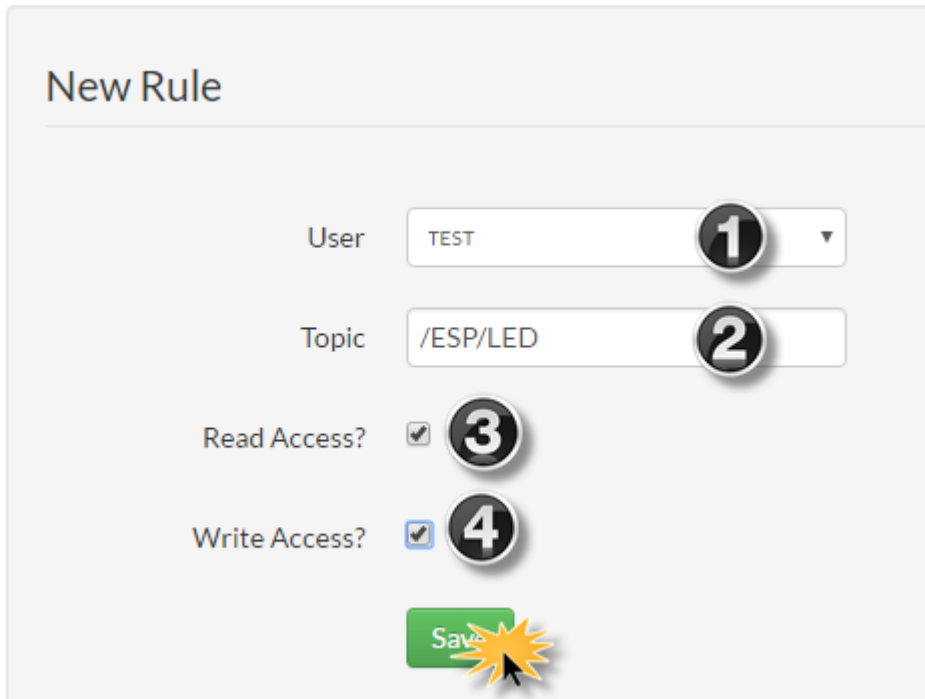
แล้วจะมีการรีเฟรชหน้าเว็บชั่วคราว จากนั้นก็จะแสดงรายชื่อที่เราได้สร้างไว้ทั้งหมดออกมา

Manage Users

• TEST

[Delete](#)[Save](#)

เลื่อนลงมาที่ New Rule ตรงส่วนนี้จะเป็นส่วนที่ไว้กำหนดว่าผู้ดูแลห้อง (User) ใด สามารถ ฟัง (Read) หรือประกาศ (Write) ไปที่ห้อง (Topic) ใดได้บ้าง เลือก User ในข้อที่ 1 จากนั้น ตั้งชื่อห้อง (Topic) ในช่องที่ 2 หากทำครั้งแรกแนะนำให้ตั้ง /ESP/LED ตามผมไปก่อน แล้ว เลือกสิทธิ์ในการใช้งานห้อง (Topic) ในช่อง 3 และ 4 จากนั้นกด Save



รอซักครู่หนึ่ง จะมีการรีเฟรชหน้าเว็บ จากนั้นเลื่อนลงมาดูในหัวข้อ ACLs ก็จะมีข้อมูลที่เรา ได้ตั้งไว้ครับ

User	Topic	Read	Write	
TEST	/ESP/LED	true	true	<button>Delete</button>

มาถึงตรงนี้ก็เรียบร้อยแล้วลองกดไปที่ Websocket UI

CloudMQTT Console

Overview **Websocket UI** Server log Statistics Restart

Instance info

Server m12.cloudmqtt.com

แล้วเปิดลิ้งขึ้นมา 2 หน้าคู่กัน ลองใส่ Topic เป็น /ESP/LED แล้วใส่ข้อความลงไป กดปุ่ม Send หากส่งจากหน้าหนึ่ง อีกหน้าหนึ่งจะมีการแสดงข้อมูลอัตโนมัติด้วย ซึ่งในหน้านี้จะใช้สำหรับดูว่าโค้ดที่เราได้เขียนนั้นได้มีการส่งข้อมูลมาที่เซิร์ฟเวอร์หรือเปล่า สามารถใช้สำหรับ ตีบักได้ในกรณีเกิดปัญหา

CloudMQTT Console

Overview Websocket UI Server log Statistics Restart

Websocket

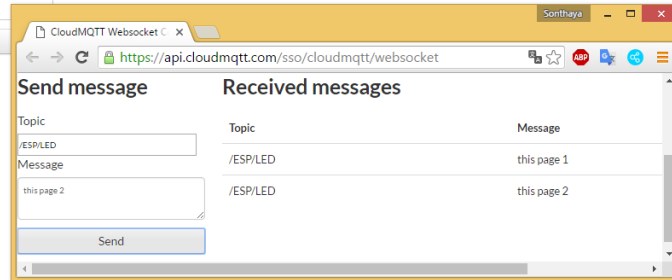
Send message

Topic
/ESP/LED
Message
this page 1

Send

Received messages

Topic	Message
/ESP/LED	this page 1
/ESP/LED	this page 2



เขียนโค้ดด้วย Arduino IDE

หากยังไม่ได้ลงบอร์ดให้โปรแกรม Arduino ท่านสามารถอ่านวิธีลงบอร์ดได้ที่บทความ [เริ่มพัฒนาเฟิร์มแวร์ด้วย Arduino IDE](#)

ใน Arduino จะไม่มีไลบรารีเชื่อมต่อ MQTT มาให้ในตัว สามารถดาวน์โหลดมาติดตั้งจากลิง [Arduino Client for MQTT](#)

ก๊อปปี้โค้ดด้านล่างนี้ลงโปรแกรม Arduino IDE ไป

```
// Coding By IOXhop : http://www.ioxhop.com/
```

```
#include <ESP8266WiFi.h>
```

```
#include <PubSubClient.h>
```

```
// Update these with values suitable for your network.
```

```
const char* ssid = "YOUR AP SSID";
```

```
const char* password = "YOUR AP PASSWORD";
```

```
// Config MQTT Server
```

```
#define mqtt_server "m12.cloudmqtt.com"
```

```
#define mqtt_port YOURMQTTPORT
```

```
#define mqtt_user "TEST"
```

```
#define mqtt_password "12345"
```

```
#define LED_PIN 2
```

```
WiFiClient espClient;
```

```
PubSubClient client(espClient);
```

```
void setup() {
```

```
pinMode(LED_PIN, OUTPUT);
```

```
Serial.begin(115200);
```

```
delay(10);
```

```
Serial.println();
```

```
Serial.print("Connecting to ");
```

```
Serial.println(ssid);
```

```
WiFi.begin(ssid, password);
```

```
while (WiFi.status() != WL_CONNECTED) {
```

```
    delay(500);
```

```
    Serial.print(".");
```

```
}
```

```
Serial.println("");
```

```
Serial.println("WiFi connected");
```



```
Serial.println("IP address: ");
```

```
Serial.println(WiFi.localIP());
```

```
client.setServer(mqtt_server, mqtt_port);
```

```
client.setCallback(callback);
```

```
}
```

```
void loop() {
```

```
if (!client.connected()) {
```

```
Serial.print("Attempting MQTT connection...");
```

```
if (client.connect("ESP8266Client", mqtt_user, mqtt_password)) {
```

```
Serial.println("connected");
```

```
client.subscribe("/ESP/LED");
```

```
} else {
```

```
Serial.print("failed, rc=");
```

```
Serial.print(client.state());
```

```
Serial.println(" try again in 5 seconds");
```

```
delay(5000);
```

```
return;
```

```
}
```

```
}
```

```
client.loop();
```

```
}
```

```
void callback(char* topic, byte* payload, unsigned int length) {
```

```
Serial.print("Message arrived [");
```

```
Serial.print(topic);
```

```
Serial.print("] ");
```

```
String msg = "";
```

```
int i=0;
```

```
while (i<length) msg += (char)payload[i++];
```

```
if (msg == "GET") {
```

```
client.publish("/ESP/LED", (digitalRead(LED_PIN) ? "LEDON" : "LEDOFF"));
```

```
Serial.println("Send !");
```

```
return;
```

```
}
```

```
digitalWrite(LED_PIN, (msg == "LEDON" ? HIGH : LOW));
```

```
Serial.println(msg);
```

```
}
```

[MQTT_LED.ino](#) hosted with ❤ by [GitHub](#)

[view raw](#)

แก้ไขตรงตัวแปร ssid (บรรทัดที่ 7) ให้เป็นชื่อ AP (ชื่อ WiFi) และแก้ไขตัวแปร password (บรรทัดที่ 8) ให้เป็นรหัสผ่านของ AP (รหัสผ่าน WiFi) กรณีไม่มีรหัสผ่านให้ใส่ตัวเลข 0 (const char* password = 0;)

แก้ค่า mqtt_server (บรรทัดที่ 11) , mqtt_port (บรรทัดที่ 12) ให้ตรงกับหัวข้อ Instance info และกรณีที่สร้าง User Pass ตามผมในขั้นตอนที่แล้วก็ไม่จำเป็นต้องเปลี่ยนตรง mqtt_user (บรรทัดที่ 13) และ mqtt_password (บรรทัดที่ 14) หากไม่ได้สร้าง User Pass ตาม ให้แก้ตรงส่วนนี้ด้วย

ในบรรทัดที่ 16 จะเป็นการกำหนดขาที่ต่อกับ LED แบบ Active Low ซึ่งผมเลือกใช้ขา GPIO5

Instance info

Server	m12.cloudmqtt.com
User	
Password	
Port	11583
SSL Port	21583
Websockets Port (TLS only)	31583
Connection limit	10

```
MQTT
1 // Include the MQTT client library
2 #include <PubSubClient.h>
3
4 // Update these with values suitable for you:
5
6 // const char* ssid = "YOUR AP SSID";
7 // const char* password = "YOUR AP PASSWORD";
8
9
10 // Config MQTT server
11 #define mqtt_server "m12.cloudmqtt.com"
12 #define mqtt_port 11583
13 #define mqtt_user "TEST"
14 #define mqtt_password "12345"
15
16 #define LED_PIN 5
17
18 WiFiClient espClient;
19 PubSubClient client(espClient);
```

เลือกบอร์ด เลือกพอร์ต จากนั้นกด Upload ได้เลย

ในบทความนี้เราจะมาใช้ MQTT ในเป็นโปรโตคอลตัวกลางในการสื่อสารระหว่างคอมพิวเตอร์ หรืออุปกรณ์ต่างๆ ผ่านอินเทอร์เน็ต ไปที่ ESP8266 กันครับ หลังจากอ่านบทความนี้จบแล้ว ท่านสามารถจะนำ MQTT ไปใช้งานอื่นๆได้อีกมากมาย เช่น การสื่อสารกันเองระหว่าง ESP8266 กับ ESP8266 ในการแชร์ข้อมูลซึ่งกันและกัน มอนิเตอร์อุณหภูมิแบบเรียลไทม์ ติดตามการทำงานของเครื่องจักรแบบไร้สาย นอกจากนี้ยังสามารถนำไปประยุกต์ใช้อะไรที่ไม่เกี่ยวข้องกับ ESP8266 อย่างการทำห้องแช่ทออนไลน์ได้อีกด้วย

รู้จักกับ MQTT

MQTT เป็นโปรโตคอลที่ใช้สำหรับการสื่อสาร ต่างภาษา ต่างอุปกรณ์กัน หรือภาษาเดียวกัน อุปกรณ์เดียวกัน ซึ่งมีไลบรารีให้พร้อมสำหรับภาษาต่างๆ ได้แก่ Heroku Ruby Python NodeJS Java Go .NET NodeMCU(Lua) และ PHP ทำงานอยู่บนฐานของโปรโตคอล TCP โดยปกติแล้ว MQTT จะสามารถใช้งานได้ก็ต่อเมื่อมีเซิร์ฟเวอร์ตัวกลางเท่านั้น ซึ่งท่านสามารถติดตั้งบนเซิร์ฟเวอร์ของตัวเองก็ได้ แต่ในบทความนี้จะเลือกใช้บริการเซิร์ฟเวอร์ MQTT จากเว็บ [CloudMQTT](#) ซึ่งสามารถใช้งานได้ฟรี 10 การเชื่อมต่อ กรณีต้องการนำไปใช้งานจริงควรเผื่อจำนวนการเชื่อมต่อไว้หลายๆหากนำไปใช้กับสาธารณะ ก่อนที่จะใช้งานได้ เราจำเป็นต้องมารู้ศัพท์ที่ใช้ และหลักการใช้งานกันก่อนครับ ก่อนอื่นเลย Username Password ที่ใช้ล็อกอินเข้าใช้งานโปรโตคอลจะมีอยู่ด้วยกัน 2 ระดับ คือ ระดับหัวหน้า และระดับผู้ดูแลห้อง (เป็นชื่อระดับที่ผมตั้งขึ้นเองเพื่อให้เข้าใจได้ง่าย) ข้อแตกต่างของ 2 ระดับนี้คือ

- **ระดับหัวหน้า** เปรียบได้กับผู้บริหาร หรือเจ้าของบริษัท มีสิทธิ์ที่จะฟัง และประกาศ ไปยังห้อง (Topic) ใหนก็ได้ที่มีการสร้างขึ้น ซึ่ง Username Password จะถูกกำหนดมาแล้ว หลังจากสมัครสมาชิกในระบบของ CloudMQTT จะมีประกาศในหน้าแรก
- **ระดับผู้ดูแลห้อง** ระดับนี้มีสิทธิ์ที่จะฟังอย่างเดียว ประกาศอย่างเดียว หรือทั้งฟังและประกาศ ได้ในห้อง (Topic) ที่กำหนดไว้แล้วเท่านั้น ซึ่ง Username Password สามารถกำหนดได้เองโดยการสร้าง

ต่อมา ก็มาทำความรู้จักกับคำว่า **Topic** กันครับ คำนี้ผมจะขอใช้ว่าเป็น "ชื่อห้อง" น่าจะเข้าใจได้ง่ายกว่า ก่อนที่เราจะส่งข้อมูล หรือรอฟังข้อมูล เราจะต้องทราบชื่อห้องซะก่อนครับ เปรียบเสมือนว่าหากเราต้องการจะคุยกับกลุ่มไหน เราจำเป็นต้องรู้ชื่อกลุ่มซะก่อน ในทาง MQTT จะเรียกชื่อห้องว่า **Topic** ซึ่งสามารถกำหนดเป็นอะไรก็ได้ แต่โดยส่วนใหญ่จะกำหนดในลักษณะคล้าย Path เช่น /ESP/LED /ESP/TEMP และอื่นๆ การตั้ง ควรเป็นชื่อที่สื่อความหมาย และเข้าใจได้ง่ายด้วยครับ

พอร์ต (Port) ใน CloudMQTT จะแบ่งพอร์ตออกเป็น 3 ชนิด ดังนี้

- **Port** เป็นพอร์ตที่โปรโตคอล MQTT ใช้งาน ใช้งานกับภาษาที่อยู่ในฝั่ง Back-end เช่น PHP C++ และภาษาอื่นๆที่ได้กล่าวไปแล้ว (พอร์ตที่จะให้ ESP8266 คือพอร์ตชนิดนี้)

- **SSL Port** เป็นพอร์ตที่โปรโตคอล MQTT ใช้งาน แต่จะถูกใช้เมื่อต้องการเชื่อมต่อแบบ SSL ซึ่งผมแนะนำให้ใช้พอร์ตนี้หากอุปกรณ์รองรับ SSL (ESP8266 ไม่ค่อยจะรองรับ จึงไม่ได้ใช้)
- **Websockets Port (TLS only)** เป็นพอร์ตที่ใช้กับหน้าเว็บ ซึ่งหน้านั้นไม่รองรับโปรโตคอล MQTT หรือเชื่อมต่อ TCP โดยตรง แต่มีเทคโนโลยีเรียวทามที่ออกแบบมาสำหรับเว็บอยู่แล้ว ชื่อ WebSocket ในการที่จะส่งข้อมูลผ่านหน้าเว็บจะต้องคอนฟิกพอร์ตเป็นพอร์ตชนิดนี้

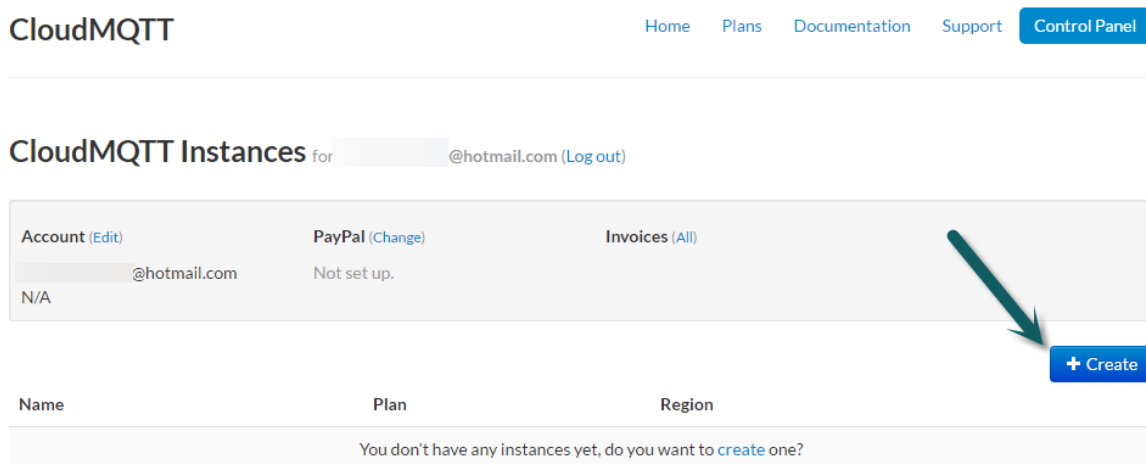
Connection limit เป็นการจำกัดจำนวนผู้เข้ามาเชื่อมต่อ (Clients) ในวินาทีนั้นๆ ตัวอย่างเช่น เปิดหน้าเว็บที่เชื่อมต่ออยู่ไว้ 2 หน้า และมี ESP8266 ที่เชื่อมต่ออีก 1 ตัว รวมแล้วจะนับเป็น 3 การเชื่อมต่อ หากใช้งานฟรีเซิร์ฟเวอร์ MQTT จาก CloudMQTT จะถูกจำกัดไว้ที่ 10 การเชื่อมต่อ

สมัครใช้งานฟรีเซิร์ฟเวอร์ MQTT จาก CloudMQTT

เข้าไปที่ลิง <https://customer.cloudmqtt.com/login> ตรง Sign up กรอกอีเมลของท่านลงไปก่อน จากนั้นไปเช็คเมล ระบบจะส่งลิงสำหรับสมัครสมาชิกไปให้ จากนั้นคลิกลิงที่อยู่ในเมล แล้วกรอกรายละเอียดต่างๆเป็นอันเสร็จครับ (ผมไม่มีรูปภาพให้ดูเป็นขั้นตอนเนื่องจากได้สมัครสมาชิกไว้แล้วครับ)

เลือก Plan ใช้งาน และเตรียมพร้อมก่อนเขียนโค้ด

เมื่อเข้าสู่ระบบแล้ว จะพบกับหน้าจัดการทั้งหมดประมาณรูปด้านล่างนี้ กดปุ่ม + Create



CloudMQTT Home Plans Documentation Support **Control Panel**

CloudMQTT Instances for (Log out)

Account (Edit)	PayPal (Change)	Invoices (All)
<input type="text" value="@hotmail.com"/> N/A	Not set up.	

+ Create

Name	Plan	Region
You don't have any instances yet, do you want to create one?		

จากนั้นตั้งชื่อ (ชื่ออะไรก็ได้ ไม่มีผลต่อโค้ด) ในข้อที่ 1 เลือกที่ตั้งของเซิร์ฟเวอร์ในข้อที่ 2 (ให้เลือก US กับ EU ผมเลือก US) และสุดท้าย Plan กรณีต้องการใช้ฟรีให้คง Cute Cut ไว้ไม่ต้องแก้ หากต้องการใช้งานแบบเสียเงินก็ต้องเชื่อม Paypal แล้วเลือก Plan ที่ลิง <https://www.cloudmqtt.com/plans.html> จากนั้นกดปุ่ม Create

Create new CloudMQTT Instance

PayPal not enabled. Please [enable PayPal](#) if you want to subscribe to a paid plan

Name

Name to describe your instance

1

Data center

Amazon US East (Northern Virginia)

2

Plan

Cute Cat

3

To learn more about the different plans please visit: www.cloumqtt.com/plans.html

Create

Cancel

จากนั้นระบบจะพากลับมาหน้าเดิม ให้กดที่ปุ่ม Details

CloudMQTT

[Home](#)

[Plans](#)

[Documentation](#)

[Support](#)

[Control Panel](#)

CloudMQTT Instances for max30012540@hotmail.com ([Log out](#))

Account ([Edit](#))

max30012540@hotmail.com
N/A

PayPal ([Change](#))

Not set up.

Invoices ([All](#))

[+ Create](#)

Name

Plan

Region

TESTROOM

Cat

US

[Details](#)

[Edit](#)

[Delete](#)

จากนั้นรายละเอียดต่างๆจะแสดงขึ้นมาในหัวข้อ Instance info (สามารถอ่านรายละเอียดได้ในหัวข้อที่แล้ว)

CloudMQTT Console

[Overview](#)[Websocket UI](#)[Server log](#)[Statistics](#)[Restart](#)

Instance info

Server	m12.cloudmqtt.com
User	
Password	
Port	
SSL Port	
Websockets Port (TLS only)	
Connection limit	10

ต่อมา เราจะมาสร้าง Username Password ระดับหัวหน้าห้องกันครับ เลื่อนมาที่หัวข้อ Manage Users แล้วตั้ง Username Password ที่ต้องการ หากทำครั้งแรกแนะนำให้ตั้งตามผมไปก่อนคือ User : TEST และ Pass : 12345 แล้วกดปุ่ม Save

Manage Users

1 2 Save

แล้วจะมีการรีเฟรชหน้าเว็บชั่วคราว จากนั้นก็จะแสดงรายชื่อที่เราได้สร้างไว้ทั้งหมดออกมา

Manage Users

- TEST

DeleteSave

เลื่อนลงมาที่ New Rule ตรงส่วนนี้จะเป็นส่วนที่ไว้กำหนดว่าผู้ดูแลห้อง (User) ใด สามารถ ฟัง (Read) หรือประกาศ (Write) ไปที่ห้อง (Topic) ใดได้บ้าง เลือก User ในข้อที่ 1 จากนั้น ตั้งชื่อห้อง (Topic) ในช่องที่ 2 หากทำครั้งแรกแนะนำให้ตั้ง /ESP/LED ตามผมไปก่อน แล้ว เลือกสิทธิ์ในการใช้งานห้อง (Topic) ในช่อง 3 และ 4 จากนั้นกด Save

New Rule

User TEST 1

Topic /ESP/LED 2

Read Access? ☒ 3

Write Access? ☒ 4



รอซักครู่หนึ่ง จะมีการรีเฟรชหน้าเว็บ จากนั้นเลื่อนลงมาดูในหัวข้อ ACLs ก็จะได้เห็นข้อมูลที่เราได้ตั้งไว้ครับ

User	Topic	Read	Write	
TEST	/ESP/LED	true	true	Delete

มาถึงตรงนี้ก็เรียบร้อยแล้วลองกดไปที่ Websocket UI

CloudMQTT Console

Overview

Websocket UI

Server log

Statistics

Restart

Instance info

Server

m12.cloudmqtt.com

แล้วเปิดลิ้งขึ้นมา 2 หน้าคู่กัน ลองใส่ Topic เป็น /ESP/LED แล้วใส่ข้อความลงไป กดปุ่ม Send หากส่งจากหน้าหนึ่ง อีกหน้าหนึ่งจะมีการแสดงข้อมูลอัตโนมัติด้วย ซึ่งในหน้านี้จะใช้สำหรับดูว่าโค้ดที่เราได้เขียนนั้นได้มีการส่งข้อมูลมาที่เซิร์ฟเวอร์หรือเปล่า สามารถใช้สำหรับดีบักได้ในกรณีเกิดปัญหา

