

## การทดลองที่ 8

### การใช้งาน IFTTT ด้วย NodeMCU โดยการใช้ Arduino IDE

IFTTT คืออะไร

IFTTT หรือ IF This Then That คือบริการออนไลน์ที่ช่วยให้ผู้ใช้สามารถตั้งค่าการทำงานได้อย่างยืดหยุ่น ตามเงื่อนไขที่เราต้องการ ยกตัวอย่างเช่น ให้เก็บข้อมูลที่อ่านได้จาก Sensor ไว้ที่ Google Sheets หรือถ้ามีเหตุการณ์เกิดขึ้นให้แจ้งไปที่ Line Notify

กล่าวคือ IFTTT เป็นเว็บและแอป ที่นำเอา API ของ Service หลายเจ้าในโลกใบนี้เข้ามาใช้ด้วยกันได้ สามารถสร้างสูตร (Recipe) ขึ้นมาได้อย่างอิสระ และนำไปแชร์ให้คนอื่นใช้ได้อีกด้วย IFTTT มีทั้งบนเว็บไซต์ และบนแอป iOS, Android สามารถโหลดไปทำพร้อมกันได้เลย ใน IFTTT นี้มี Applet ที่มีผู้สร้างไว้มากมาย และเราสามารถสร้างเองได้

ตัวอย่างนี้เป็นแสดงให้เห็นถึงวิธีการเก็บข้อมูลที่อ่านได้จากเซ็นเซอร์ ไปไว้ที่ Google Sheets โดยใช้ ESP8266 Node MCU ร่วมกับ IFTTT (ในที่นี้ข้อมูลไม่ได้อ่านจากเซ็นเซอร์จริงใช้การ Random ตัวเลขแทน)

การสร้าง Applet ด้วยตัวเอง

1. สมัครใช้งานก่อนที่เว็บไซต์ <https://ifttt.com/> ถ้ามี Gmail อยู่แล้วก็สามารถใช้งานได้เลย

IFTTT

Sign in Sign up

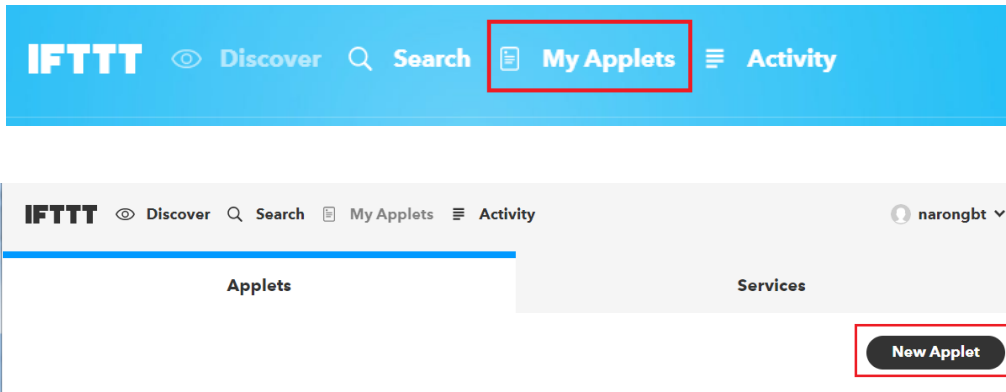
## A world that works for you

IFTTT is the free way to get all your apps and devices talking to each other. Not everything on the internet plays nice, so we're on a mission to build a more connected world.

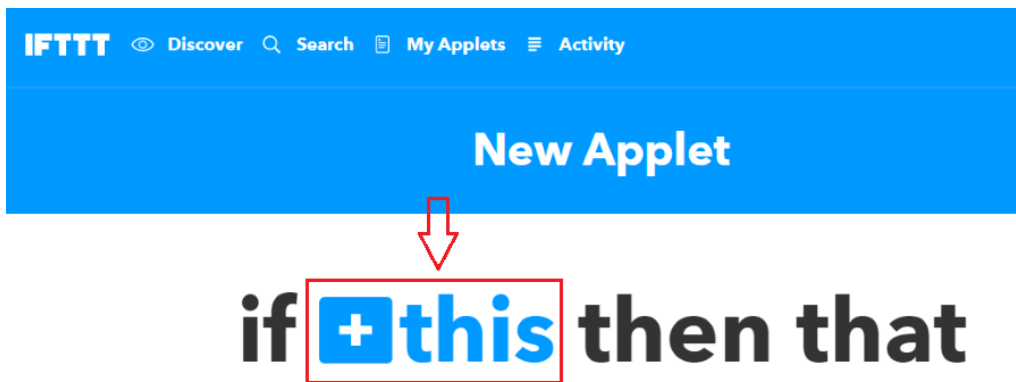


Ggssss

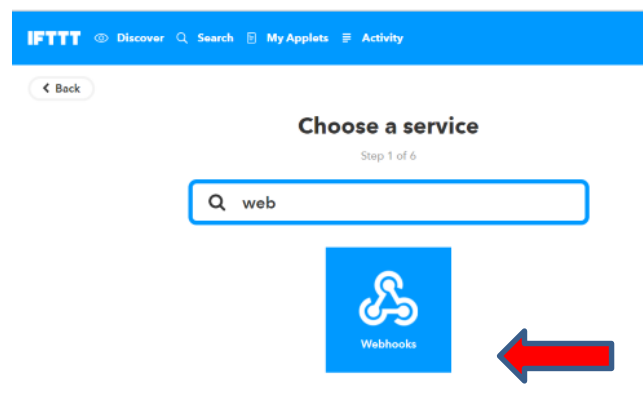
2. เมื่อสมัครใช้งานแล้วระบบจะ login ให้อัตโนมัติ ให้คลิกที่ My Applets และคลิกที่ New Applet เพื่อเริ่มต้นใช้งาน



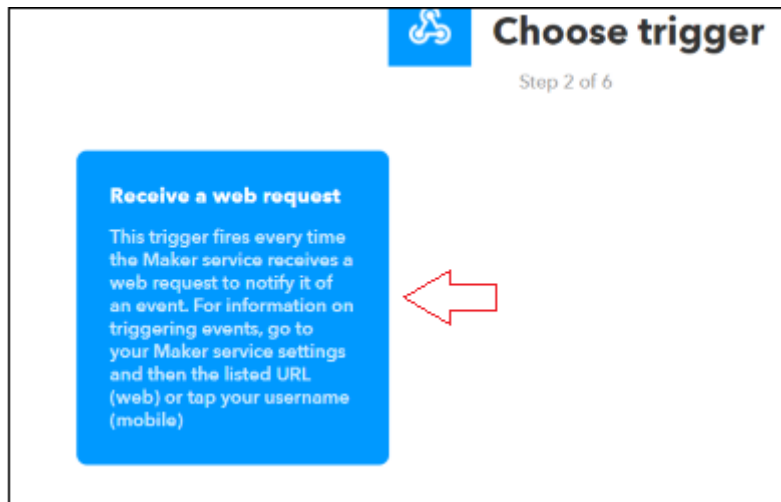
3. ระบบจะแสดงหน้าจอคังภาพนี้ ให้คลิกที่อักษรสีฟ้าที่เขียนว่า this เพื่อกำหนดเหตุการณ์(Trigger) ที่จะ เป็นจุดกำเนิดของการแจ้งเตือน



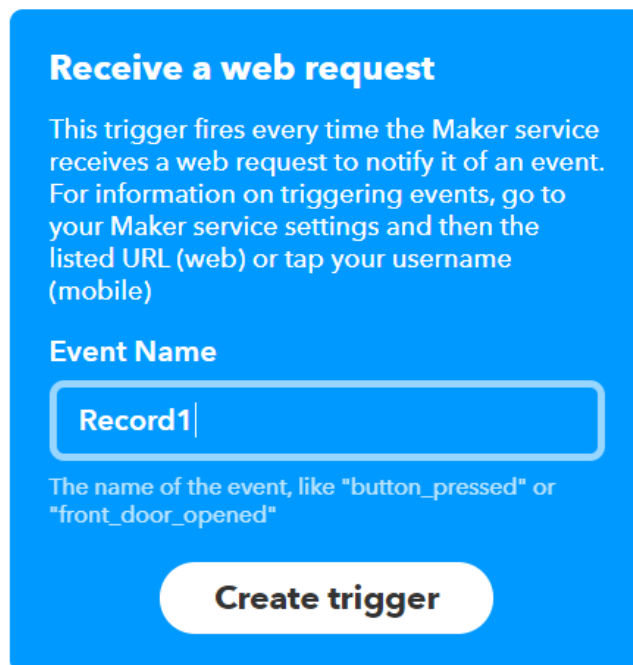
4. สำหรับกรณีนี้เราจะให้ Trigger จาก Arduino และส่งข้อมูลไปที่ Google Sheet ดังนั้นให้เรา Search คำว่า Webhooks เป็น Applet ตัวหนึ่งที่ IFTTT เตรียมไว้ให้ Developer ใช้



5. เมื่อเลือกที่ Webhooks ระบบจะ Connect ไปที่ Server และแสดง Applet ให้เราเลือก ซึ่งปัจจุบันจะมีตัว เดียวคือ "Receive a web request" ให้คลิกเลือกได้เลย



6. กำหนด Event Name เป็น Record1 (ตั้งชื่ออะไรก็ได้แต่ต้องจำไว้ใช้งาน) จากนั้นกดปุ่ม Create Trigger



7. ระบบจะย้อนมาที่หน้าหลักเพื่อให้เราสร้าง Action ให้คลิกเลือก that

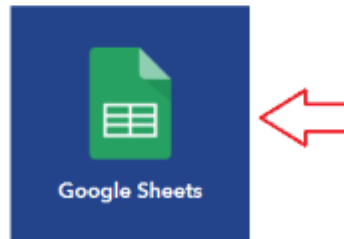


9. ระบบจะขึ้น Action Service ให้เลือก ให้ค้นหาด้วยคำว่า Google Sheets เมื่อได้แล้วให้เลือก Google sheet

## Choose action service

Step 3 of 6

Q Google Sheets"



10. จากนั้นระบบจะขึ้น Action ให้เลือกว่าจะจัดการอย่างไรกับ Sheet ให้คลิก Add row



**Add row to spreadsheet**

This action will add a single row to the bottom of the first worksheet of a spreadsheet you specify. Note: a new spreadsheet is created after 2000 rows.

**Update cell in spreadsheet**

This action will update a single cell in the first worksheet of a spreadsheet you specify. Note: a new spreadsheet is created if the file doesn't exist.

11. ระบบจะขึ้นข้อมูลที่จะกระทำกับ sheet คือ

- Spreadsheet name เป็นการตั้งชื่อ spreadsheet ที่จะทำงาน จะเปลี่ยนหรือไม่ก็ได้ (ควรจำไว้)
- Formatted row เป็นรูปแบบข้อมูลของแต่ละแถวที่จะบันทึกคอลัมน์ คอลัมน์นี้เพิ่มหรือลดก็ได้

OccuredAT หมายถึง เวลาที่ทำการบันทึก

EventName หมายถึง ชื่อสิ่งที่บันทึก

Value1, Value2, Value3 หมายถึงค่าที่ทำการบันทึก

สุดท้าย Drive folder path คือที่เก็บ Sheet

ค่าต่างๆเหล่านี้ปรับแก้ได้  
เมื่อปรับแก้แล้ว (หรือคงไว้) ให้คลิก Create Action

**Add row to spreadsheet**

This action will add a single row to the bottom of the first worksheet of a spreadsheet you specify. Note: a new spreadsheet is created after 2000 rows.

**Spreadsheet name**

IFTTT\_Maker\_Webhooks\_Events

Will create a new spreadsheet if one with this title doesn't exist

Add ingredient

**Formatted row**

OccurredAt		EventName		
Value1		Value2		Value3

Use "|||" to separate cells

Add ingredient

**Drive folder path**

IFTTT/MakerWebhooks/  
EventName

Format: some/folder/path (defaults to "IFTTT")

Add ingredient

**Create action**

ชื่อ Sheet

รูปแบบ Row ที่ต้องการบันทึก

โฟลด์เดอร์

12. ระบบจะขึ้น Review and finish ให้กดปุ่ม Finish เป็นอันเรียบร้อย

## Review and finish

Step 6 of 6



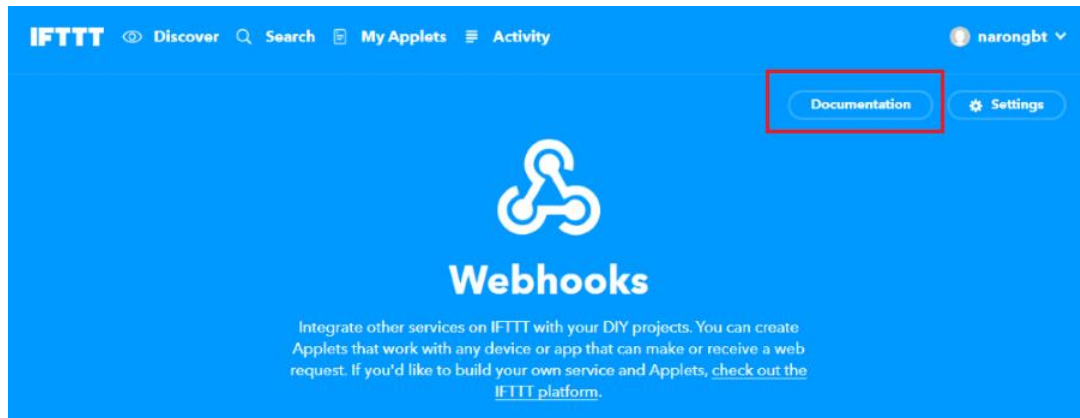
The card features a white icon of three interconnected nodes on a blue background. Below the icon is a white-bordered box containing the text: "If maker Event 'Record1', then add row to Narong Buabthong's Google Drive spreadsheet". To the right of this box is the text "85/140". Below the box is the text "by narongbt". At the bottom right of the card, it says "works with" followed by a small icon of a document with a grid.

Receive notifications  
when this Applet runs

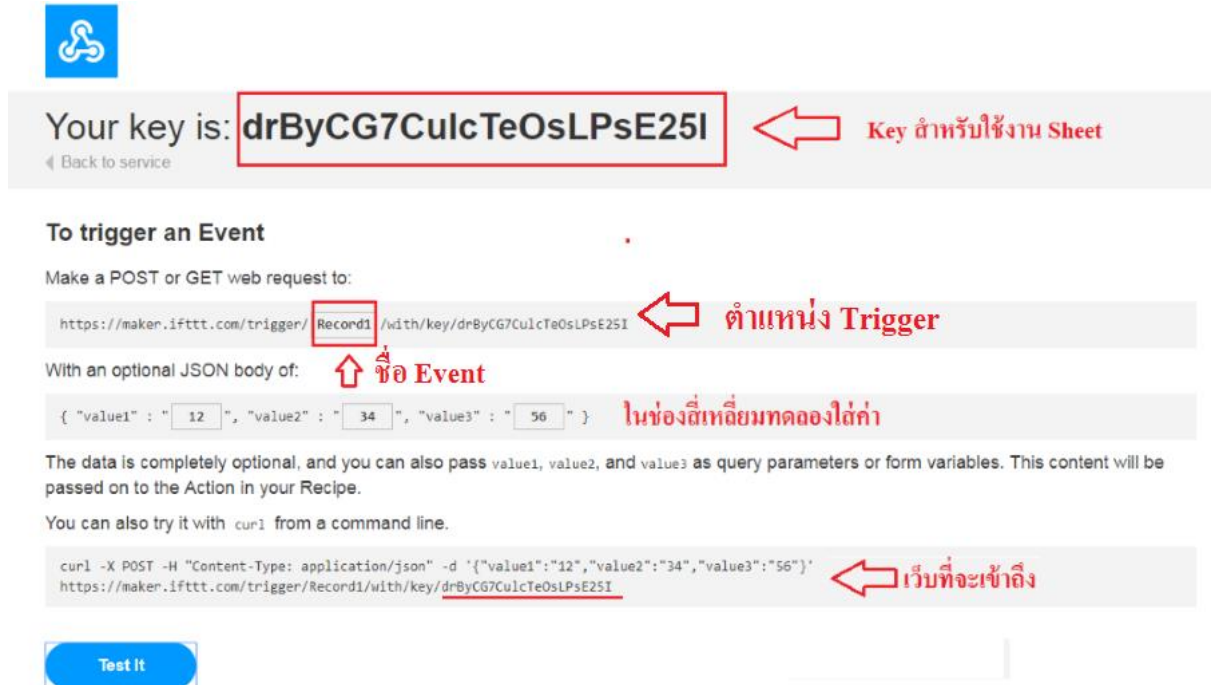


## ทดสอบ Applet

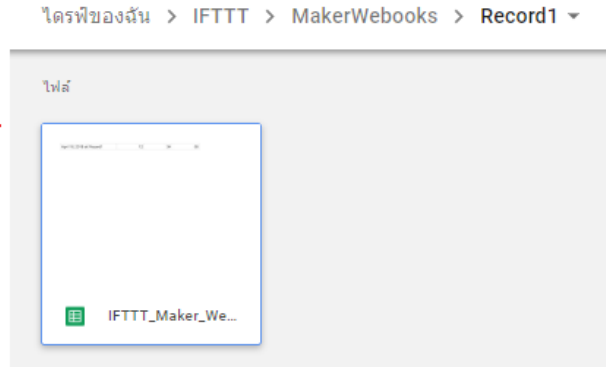
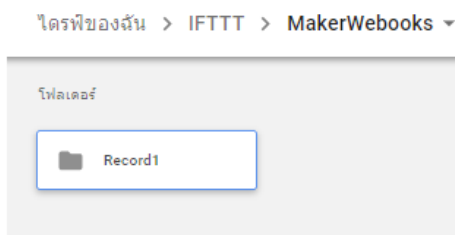
1. ให้ไปที่ [https://ifttt.com/maker\\_webhooks](https://ifttt.com/maker_webhooks)
2. เมื่อปรากฏหน้าเว็บ ให้กดปุ่ม Document



3. ถึงตอนนี้จะได้ API Key และหน้าเว็บ https ที่จะเข้าถึงชื่อ API key นี้ห้ามเผยแพร่ ให้ป้อนชื่อ Event ที่ตั้งไว้ในข้อ 6 ทดลองป้อนค่า ตัวเลข 3 ค่าลงในช่องสี่เหลี่ยมแล้วกดปุ่ม "Test"



4. เข้าไปที่ Google drive จะปรากฏชื่อ



IFTTT\_Maker\_Webhooks\_Events ☆

ไฟล์ แก้ไข แสดง แทรก รูปแบบ ข้อมูล เครื่องมือ ส่วนเสริม ความช่วยเหลือ

fx April 18, 2018 at 11:00PM

	A	B	C	D	E
1	April 18, 2018 at	Record1	12	34	56
2					
3					
4					

เขียนโปรแกรมแล้วเขียนโปรแกรมบน Arduino IDE หลังจากอัปโหลด Sketch เข้าสู่บอร์ด ESP8266 NodeMCU แล้ว ให้เข้าไปดูใน Serial Monitor จะพบว่าบอร์ดทำการเชื่อมต่อ WIFI ตามที่กำหนดไว้แล้ว แสดงเข้าถึงซีท

```
#include <ESP8266WiFi.h>

int n1,n2,n3;
// Replace with your SSID and Password
const char* ssid = ".....";
const char* password = ".....";

// Replace with your unique IFTTT URL resource
const char* resource =
"https://maker.ifttt.com/trigger/Temperature/with/key/drByCG7CulcTeOsLPsE25I";

// How your resource variable should look like, but with your own API KEY (that API KEY below is just
an example):
//const char* resource = "/trigger/bme280_readings/with/key/nAzjOphL3d-ZO4N3k64-
1A7gTINSrxMJdmqy3";

// Maker Webhooks IFTTT
const char* server = "maker.ifttt.com";

void setup() {
  Serial.begin(115200);
```



```

    delay(2000);
}

void loop() {
    n1 = random(20, 50);
    n2 = random(60, 90);
    n3 = random(95, 120);
    initWifi();
    makeIFTTTRequest();
    delay(5000);
}

// Establish a Wi-Fi connection with your router
void initWifi() {
    Serial.print("Connecting to: ");
    Serial.print(ssid);
    WiFi.begin(ssid, password);

    int timeout = 10 * 4; // 10 seconds
    while(WiFi.status() != WL_CONNECTED && (timeout-- > 0)) {
        delay(250);
        Serial.print(".");
    }
    Serial.println("");

    if(WiFi.status() != WL_CONNECTED) {
        Serial.println("Failed to connect, going back to sleep");
    }

    Serial.print("WiFi connected in: ");
    Serial.print(millis());
    Serial.print(", IP address: ");
    Serial.println(WiFi.localIP());
}

// Make an HTTP request to the IFTTT web service
void makeIFTTTRequest() {
    Serial.print("Connecting to ");
    Serial.print(server);

    WiFiClient client;
    int retries = 5;
    while(!client.connect(server, 80) && (retries-- > 0)) {
        Serial.print(".");
    }
    Serial.println();
    if(!client.connected()) {
        Serial.println("Failed to connect...");
    }

    Serial.print("Request resource: ");
    Serial.println(resource);

    String jsonObject = String("{\"value1\":\\"" + n1 + "\",\"value2\":\\"" + n2 + "\",\"value3\":\\"" + n3
+ "\"}");
}
/*

```

```

// Temperature in Celsius
String jsonObject = String("{\"value1\":\\"" + bme.readTemperature() + "\",\"value2\":\\"" +
(bme.readPressure()/100.0F)
    + "\",\"value3\":\\"" + bme.readHumidity() + "\"}");
*/

// Comment the previous line and uncomment the next line to publish temperature readings in
Fahrenheit
/*String jsonObject = String("{\"value1\":\\"" + (1.8 * bme.readTemperature() + 32) +
"\",\"value2\":\\""
    + (bme.readPressure()/100.0F) + "\",\"value3\":\\"" + bme.readHumidity() + "\"}";*/

client.println(String("POST ") + resource + " HTTP/1.1");
client.println(String("Host: ") + server);
client.println("Connection: close\r\nContent-Type: application/json");
client.print("Content-Length: ");
client.println(jsonObject.length());
client.println();
client.println(jsonObject);

int timeout = 5 * 10; // 5 seconds
while(!client.available() && (timeout-- > 0)){
    delay(100);
}
if(!client.available()) {
    Serial.println("No response...");
}
while(client.available()){
    Serial.write(client.read());
}

Serial.println("\nclosing connection");
client.stop();
}

```

## อ้างอิง

1. <https://randomnerdtutorials.com/esp32-esp8266-publish-sensor-readings-to-google-sheets/>