

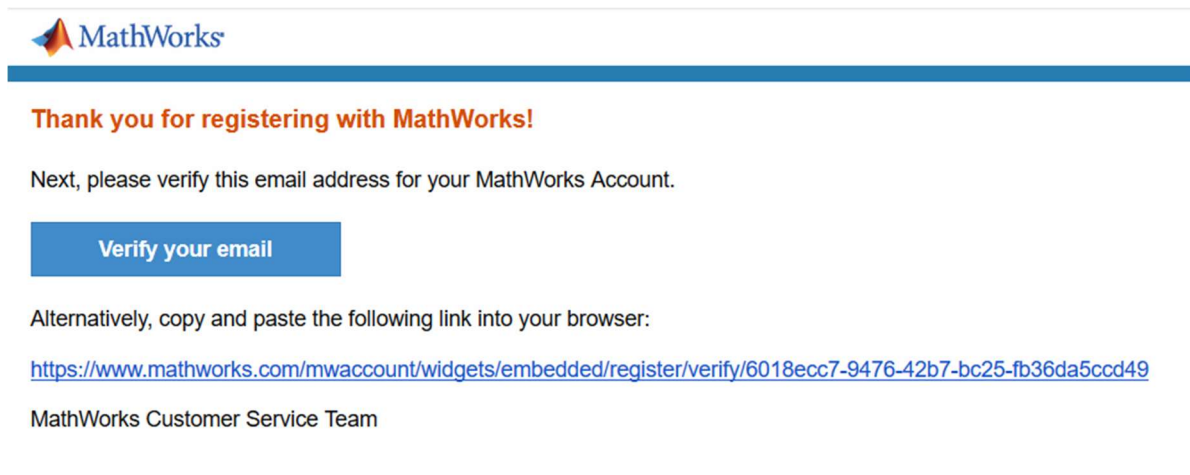
[ HTTP-發 ] 請求到http:// ?

積木用途：發送 HTTP 請求將資料送到外部網站，使資料能共享並做進一步的應用。

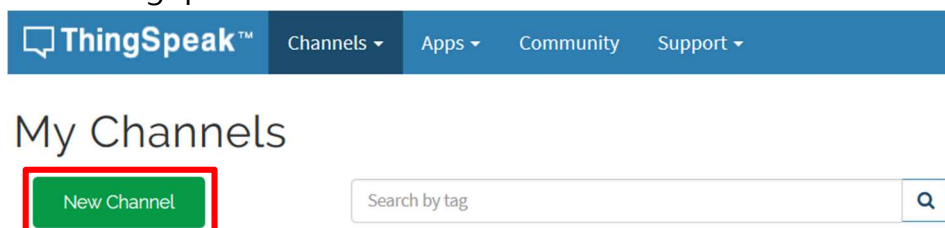
說明：物聯網的應用千變萬化，在此僅介紹兩個應用：一是收集資料、分析、共享的 ThingSpeak；另一個是串接網路服務的 IFTTT。

## 一、 ThingSpeak

1. 連上 Thingspeak 網站，網址是 <https://thingspeak.com/>，並註冊一個帳號。  
註冊並沒有特殊程序，填完資料後，會發一封確認信到 e-mail 信箱，按下確認接鈕即可，ThingSpeak 使用 MathWork 的帳號，所以收到的信件是由 MathWorks 寄發的。



2. 回到 ThingSpeak，登入之後即可建立一個新的 Channel。



3. 接著填寫 Channel 的內容，簡單填寫 Name(資料表名稱)、Field1(資料項標題)，要收集幾項資料就填幾欄，然後就可以按下表格最下方的 Save Channel。

## New Channel

Name

Description

Field 1  ☒

Field 2  ☒

Field 3  ☒

4. Save Channel 之後可以看到三個資料項的統計圖框已經建立好，接著請按下 API Keys，看看如何讀寫資料。

### 台南地區空氣品質監測資料

Channel ID: 849002

Author:

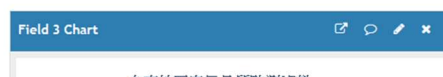
Access: Private

Private View Public View Channel Settings Sharing **API Keys** Data Import / Export

#### Channel Stats

Created: about a minute ago

Entries: 0



5. 這裡我們比較關注的是右下角的 API Requests，尤其是 Write a Channel Feed，是我們要寫進程式裡的資訊，請把 Write a Channel 欄位裡從“api\_key”以後的文字全部選取並複製起來。

Private View Public View Channel Settings Sharing API Keys Data Import / Export

### Write API Key

Key [redacted]

Generate New Write API Key

### Read API Keys

Key [redacted]

Note

Save Note Delete API Key

Generate New Read API Key

### Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

### API Keys Settings

- **Write API Key:** Use this key to write data to a channel. If you feel your key has been compromised, click **Generate New Write API Key**.
- **Read API Keys:** Use this key to allow other people to view your private channel feeds and charts. Click **Generate New Read API Key** to generate an additional read key for the channel.
- **Note:** Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

### API Requests

**Write a Channel Feed**

```
POST https://api.thingspeak.com/update?api_key=[redacted]&field1=0
```

**Read a Channel Feed**

```
GET https://api.thingspeak.com/channels/849002/feeds.json?api_key=LYO
```

**Read a Channel Field**

```
GET https://api.thingspeak.com/channels/849002/fields/1.json?api_key=
```

**Read Channel Status Updates**

```
GET https://api.thingspeak.com/channels/849002/status.json?api_key=LY
```

6. 回到 NKNU BLOCK，這次會用到 CHT11 溫溼度感測器與 PMS5003 空氣品質感測器。先寫好讀取感測器資料的部分，並依照說明完成 HTTP Request 參數字串的組合。

當被點擊

重複無限次

變數 temperature 設為 溫濕度感應器 腳位 2 的 溫度 °C

變數 humidity 設為 溫濕度感應器 腳位 2 的 濕度

變數 pm2.5 設為 讀 PM2.5 (µg/m³) 以 一般大氣顆粒 標準, Rx 10 Tx 11 模組 PMS5003G5

等待 15 秒

ThingSpeak 的免費授權寫入資料時，每筆資料至少須間隔 15 秒

字串組合 api\_key=[redacted]&field1= temperature

字串組合 &field2= humidity

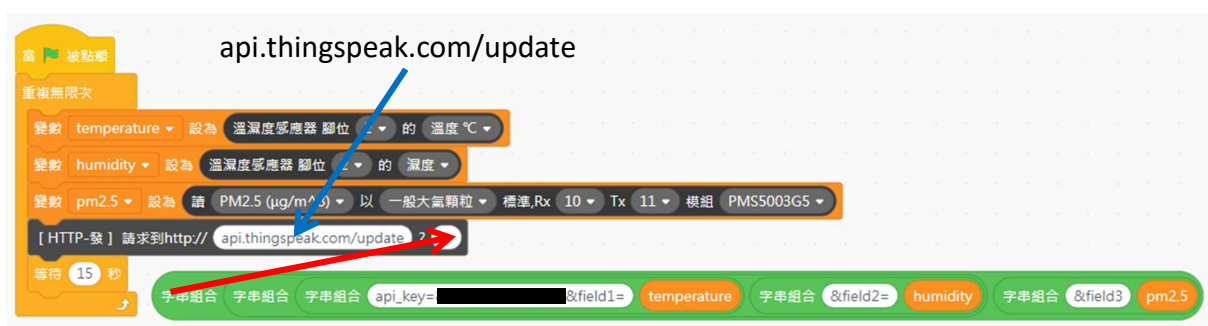
字串組合 &field3= pm2.5

把剛剛複製的 Write a Channel Feed 字串貼在這裡並刪掉最後一個字“0”

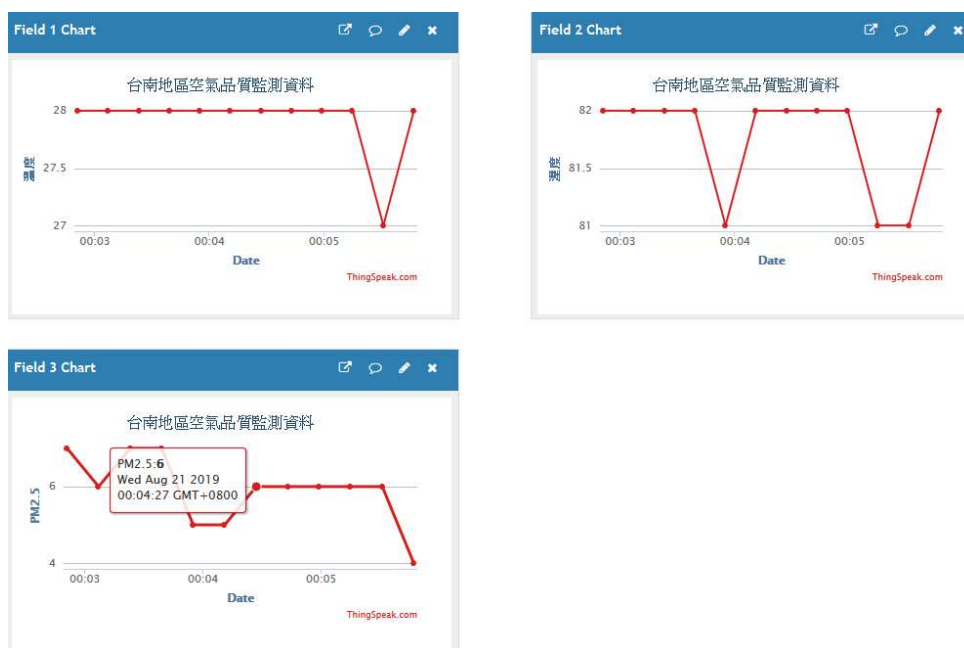
## 7. 進一步組合 HTTP Request 參數字串



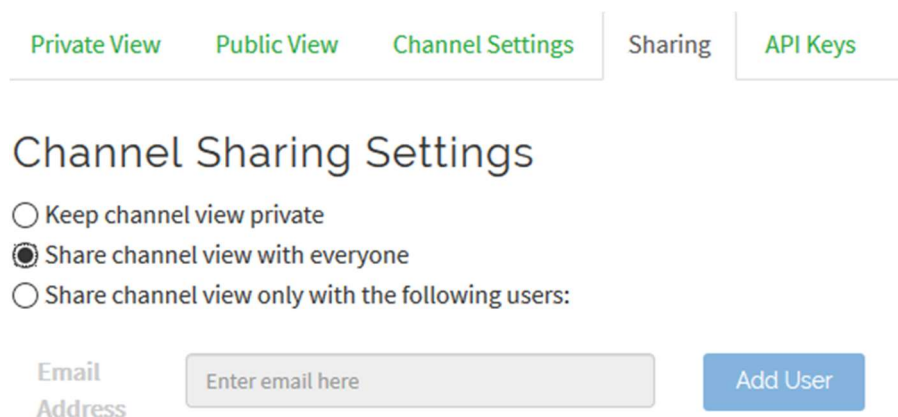
## 8. 接著就可以把發送 http 請求積木拉進程式中，記得在“http://”後面填上網址“api.thingspeak.com/update”，並將組合好的參數字串拉進積木的第2欄。



## 9. 執行程式後，就可以在 ThingSpeak 網站看到即時繪製的統計圖了。您也可以將程式燒錄到 Arduino Nano，只要給予電源，不需連接電腦也可以運作。



10. 不過這些資料只有上傳者才看得到，如果要分享給別人，請到 Sharing 項目中改變分享方式。



Private View Public View Channel Settings Sharing API Keys

### Channel Sharing Settings

☐ Keep channel view private

☒ Share channel view with everyone

☐ Share channel view only with the following users:

Email Address

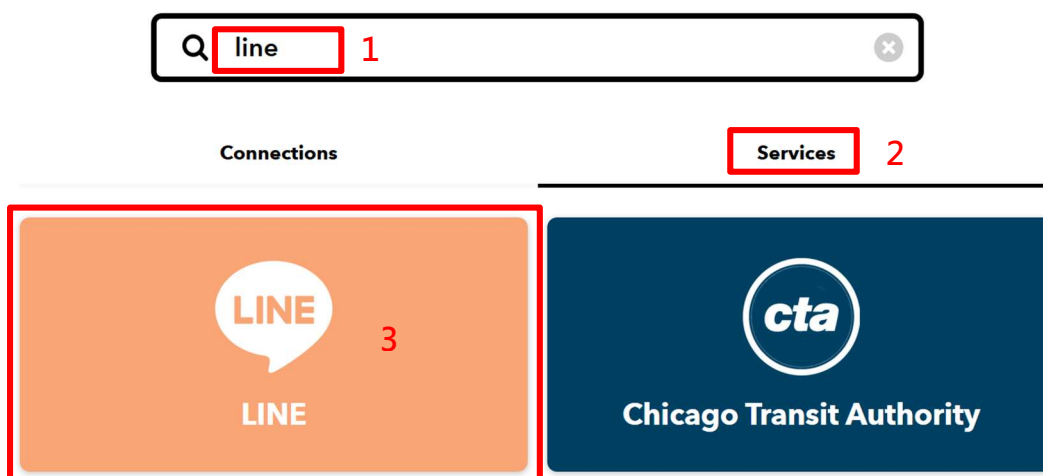
11. 除了運用感測器即時監控，一筆一筆加進資料外，也可在“Data Import/Export” 整批上傳或下載資料。

## 二、 IFTTT

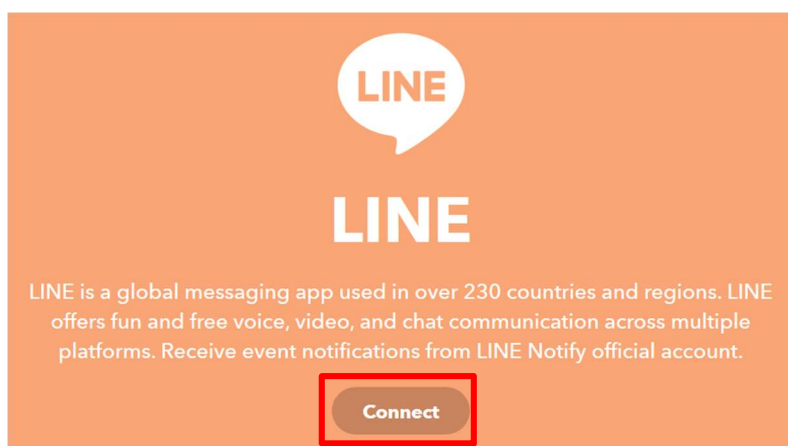
IFTTT(If this then that)是一個網路自動聯結服務，讓我們不用寫程式也可以將常用的網路服務串在一起，例如 Facebook、LINE、Gmail、Google Drive.....等，當某個網路服務發生某個條件後，觸發另一個網路服務進行某個反應行動。底下就以“當溫度、溼度異常時，就在 LINE 上通知我”為例來介紹 IFTTT 的操作過程。

1. 首先到 IFTTT 網站註冊一個帳號，可以使用 Google 或 Facebook 帳號登入，或使用 email 註冊一個新帳號。
2. 登入後會顯示 IFTTT 可以串接的服務，因為我們需要使用 LINE，所以直接搜尋 LINE，接著點右方的 Service，就可以找到 LINE，請點選 LINE 圖示。

### Get more from the services you love



3. 點選 Connect 按鈕，以便連結 LINE 官方所提供的 Notify 通知服務。



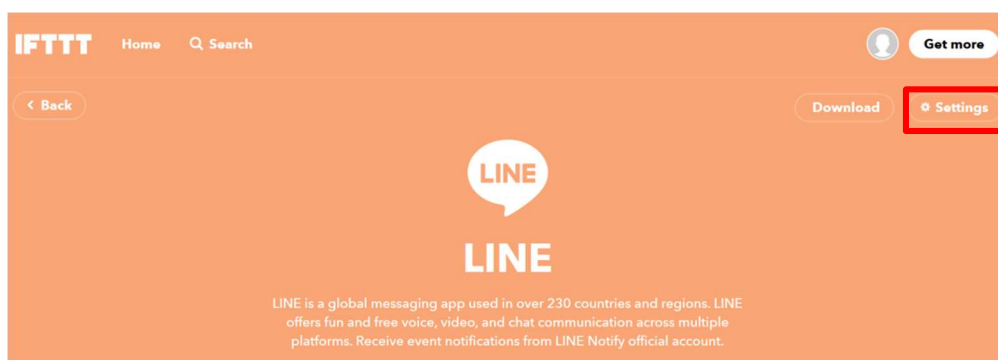
4. 接下來會進入 LINE 的登入畫面，請用自己的帳號密碼登入。

The image shows the LINE login interface. At the top is the green 'LINE' logo. Below it are two input fields: the first is labeled '電子郵件帳號' (Email account) and the second is labeled '密碼' (Password). At the bottom is a large blue button labeled '登入' (Login).

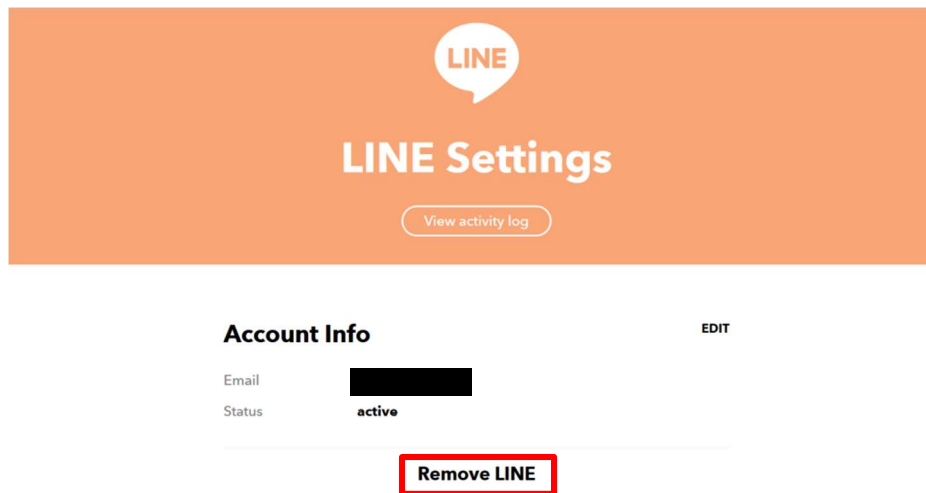
5. 接著會徵求您同意連動，同意後會將「LINE Notify」官方帳號加入好友。



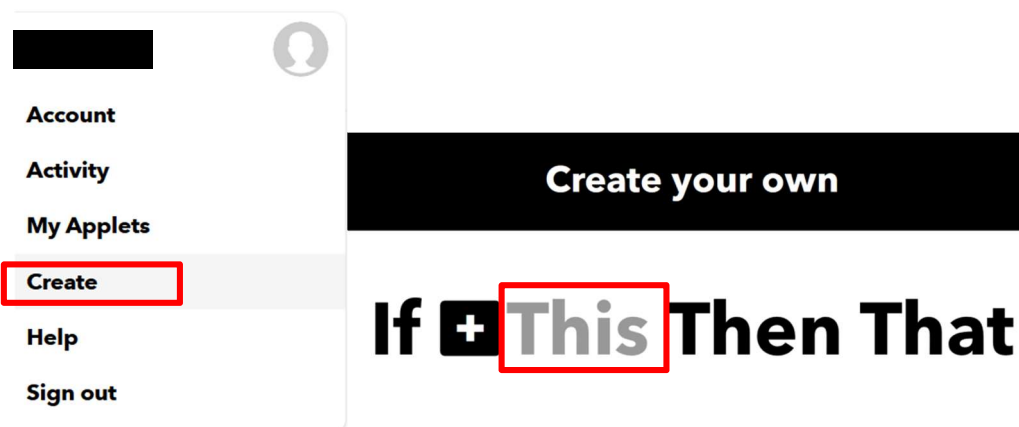
6. 連動完會回到 IFTTT 畫面，點擊右上方的 Settings, 就可以看到 Line 服務已啟動，如果想取消，點擊畫面下方的 Remove LINE 即可。



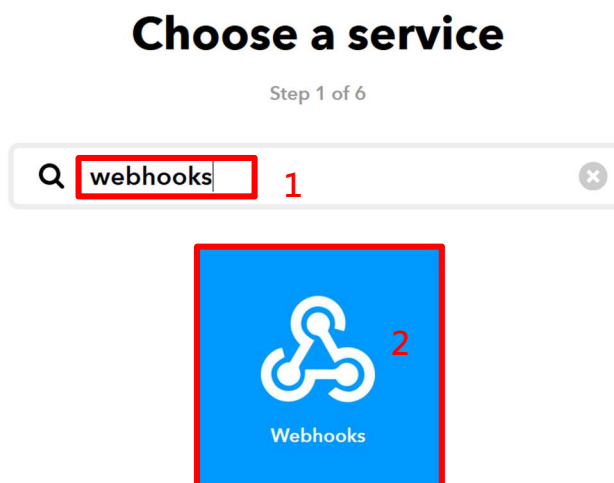




7. 建立好 LINE Notify 的連結後，回到 IFTTT 首頁，要開始處理發送訊息的服務。請點擊右上方帳號人頭圖示，選擇 Create，接著點選 if + This Then That 的“ This”。



8. 搜尋 Webhooks 服務，找到後點擊 Webhooks 圖示。





9. 連接 Webhooks 服務之後，點選 Receive a web request 觸發服務。


## Connect Webhooks

Step 1 of 6

Integrate other services on IFTTT with your DIY projects. You can create Applets that work with any device or app that can make or receive a web request. If you'd like to build your own service and Applets, check out the IFTTT platform.

**Connect**

[< Back](#)

 **Choose trigger**  
Step 2 of 6

**Receive a web request**  
This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)

輸入觸發事件名稱後，按下 Create trigger 按鈕就完成 Webhooks 的設定。

## Complete trigger fields

Step 2 of 6

**Event Name**  
  
The name of the event, like "button\_pressed" or "front\_door\_opened"

**Create trigger**

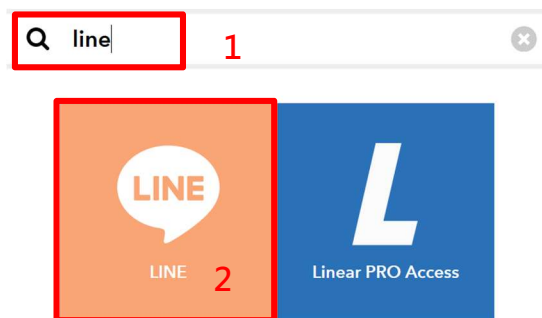
- 10.回到前一個畫面後，可以看到 If This Then That 的 This 已經變成 Webhooks 的圖示，接著請點擊 That 選擇要被觸發的服務。
- 11.

**If**  **Then**  **That**

- 12.因為是要用 Line 發送訊息，所以搜尋 line，找到後按下 LINE 圖示，然後點擊“ Send message”

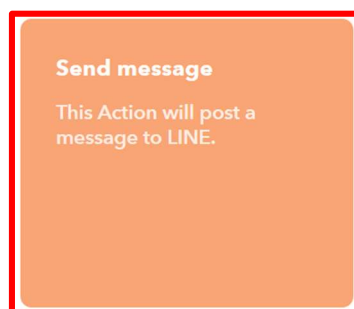
### Choose action service

Step 3 of 6



### Choose action

Step 4 of 6



13. 設定傳送訊息的格式，訊息標籤可以自行修改，由於 DHT11 只能偵測溫度與溼度，所以只要保留{{Value1}}與{{Value2}}，注意外部兩組大括號{}不可刪除，<br>是換行的意思，編輯好確認之後按下 Create action 即可建立被觸發的行為了。



## Complete action fields

Step 5 of 6

Recipient

透過1對1聊天接收LINE Not ▼

Message destination

Message

Value 1: {{Value1}}<br>  
Value 2: {{Value2}}<br>  
Value 3: {{Value3}}

Add ingredient

Photo URL

Add ingredient

Create action



## Complete action fields

Step 5 of 6

Recipient

透過1對1聊天接收LINE N ▼

Message destination

Message

<br>溫室警告訊息|<br>  
溫度: {{Value1}}<br>  
溼度: {{Value2}}<br>

Add ingredient

Photo URL

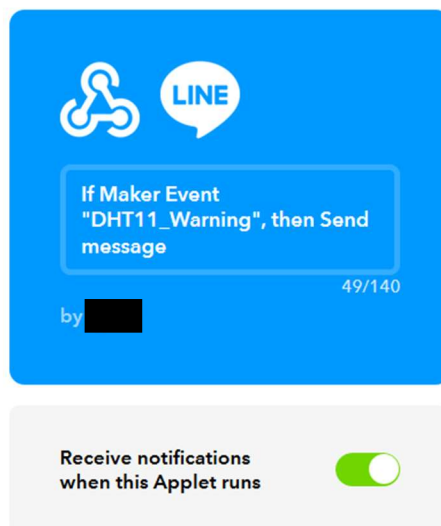
Add ingredient

Create action

14. 按下 Finish 按鈕完成 if Maker Event “DHT11\_Warning” , then Send message 服務連結。

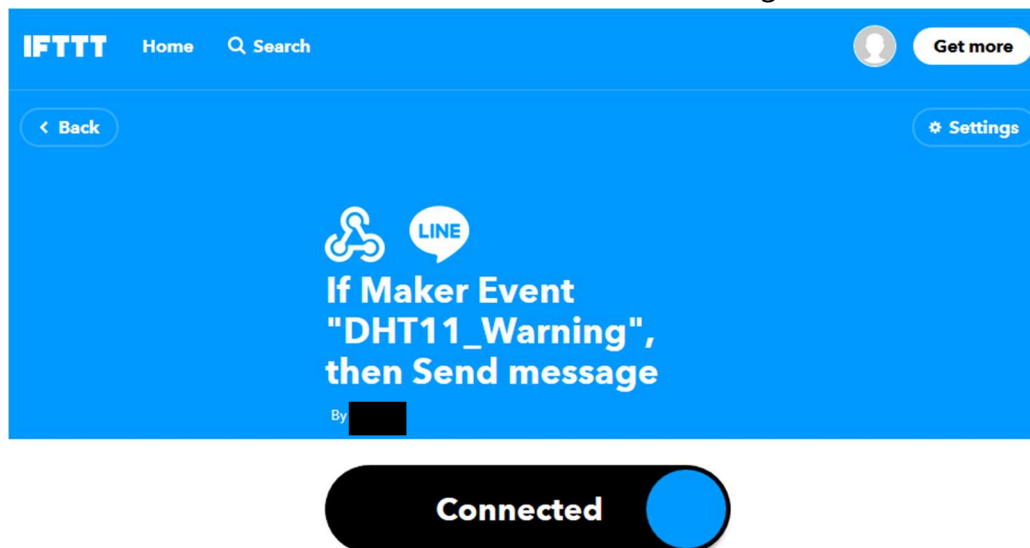
## Review and finish

Step 6 of 6

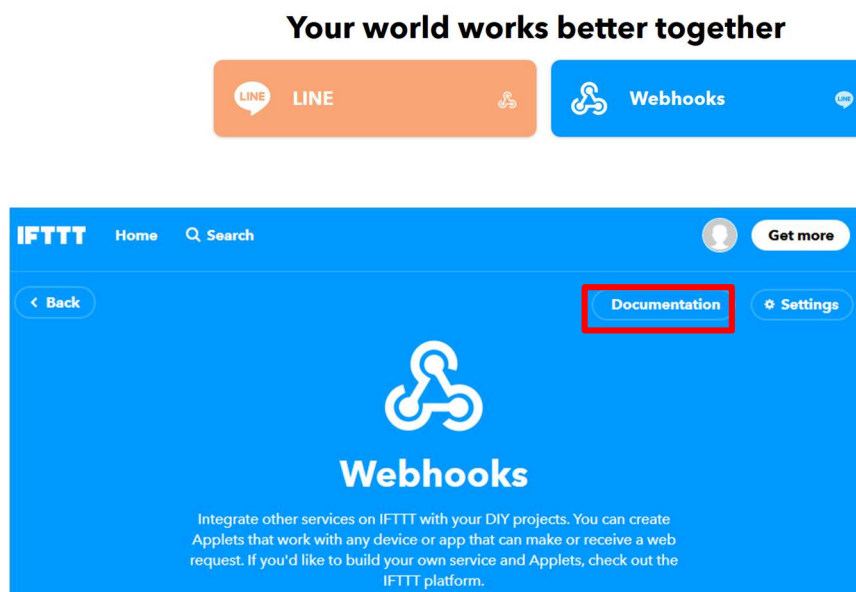


**Finish**

連結完成，若需要修改連結設定，請點擊右上方 Settings。



15. 連結設定完成後，怎麼使用呢？先回到 IFTTT 首頁，點擊右邊的 Webhooks，然後再點擊右上方的 Documentation。



16. 輸入觸發事件名稱與模擬訊息，按下 Test It 按鈕測試是否能將訊息傳送給 LINE。



成功的話，LINE 就會收到底下的訊息。



17.測試成功的話，請框選 Make a POST or GET web request to: 底下欄位中的訊息(從 maker 開始選到最末端)並複製起來。

Make a POST or GET web request to:

```
https://maker.ifttt.com/trigger/DHT11 Warning /with/key/
```

18.NKNU BLOCK 的程式中，先處理好 DHT11 溫度、溼度的偵測，以及需要傳送通知的數值範圍，然後將複製好的字中貼入 http 請求積木的第一個欄位，串接要傳送的參數，放進 http 請求積木的第二個欄位。若測試無誤，可以將程式燒錄至 Arduino Nano 中，就不需要連接電腦了。

