



南臺科技大學  
Southern Taiwan University of Science and Technology

# AHT15溫濕度感測器

## LINE 應用



古都土城仔綠電創能與智動養殖  
之跨界整合永續淨零發展計畫

# 目錄

---

- 全部程式碼
- LINE介紹
- 申請LINE Notify權杖
- 程式講解
- 最終執行成果

# 全部程式碼如下：

```
#include <Wire.h>
#include <WiFi.h>
#include <WiFiClientSecure.h>

#define AHT15_ADDRESS 0x38
#define AHT15_RESET 0xBA
#define AHT15_INIT 0xBE
#define AHT15_START_MEASUREMENT 0xAC

char SSID[] = "SSID"; // WiFi名稱
char PASSWORD[] = "SSIDPassword"; // WiFi密碼
String Linetoken = "iIFWvROwfBrYC7jVTvrVlmfJfJgVwiwDtvcdki9X2k6"; // LINE通知的權杖
WiFiClientSecure client; // 宣告一個加密的WiFi 連線端
char host[] = "notify-api.line.me"; // 宣告Line API 網站
```

# 全部程式碼如下：

```
void setup() {  
  Wire.begin();  
  Serial.begin(115200);  
  
  Serial.print("Connecting Wifi: ");  
  Serial.println(SSID);  
  WiFi.begin(SSID, PASSWORD); // 連接到WiFi  
  while (WiFi.status() != WL_CONNECTED) {  
    Serial.print(".");  
    delay(500);  
  }  
  // 連線成功，顯示取得的IP  
  Serial.println("");  
  Serial.println("WiFi connected");  
  Serial.println("IP address: ");  
  IPAddress ip = WiFi.localIP();  
  Serial.println(ip);  
  client.setInsecure(); // 讀者若使用ESP32 1.0.6 版核心須加上這句避免SSL 問題
```



# 全部程式碼如下：

```
// Reset sensor
Wire.beginTransmission(AHT15_ADDRESS);
Wire.write(AHT15_RESET);
Wire.endTransmission();
delay(20);

// Initialize sensor
Wire.beginTransmission(AHT15_ADDRESS);
Wire.write(AHT15_INIT);
Wire.write(0x08);
Wire.write(0x00);
Wire.endTransmission();
delay(10);
}
```

# 全部程式碼如下：

```
void loop() {  
  // Start measurement  
  Wire.beginTransmission(AHT15_ADDRESS);  
  Wire.write(AHT15_START_MEASUREMENT);  
  Wire.write(0x33);  
  Wire.write(0x00);  
  Wire.endTransmission();  
  delay(80);  
  // Read data  
  Wire.requestFrom(AHT15_ADDRESS, 6);  
  uint8_t data[6];  
  for (int i = 0; i < 6; i++) {  
    data[i] = Wire.read();  
  }  
  // Calculate humidity and temperature  
  uint32_t rawHumidity = ((uint32_t)data[1] << 12) | ((uint32_t)data[2] << 4) | (data[3] >> 4);  
  float humidity = rawHumidity * 100.0 / (1 << 20);  
  
  uint32_t rawTemperature = (((uint32_t)data[3] & 0xF) << 16) | ((uint32_t)data[4] << 8) | data[5];  
  float temperature = ((200.0 * rawTemperature) / (1 << 20)) - 50;  
}
```

## 全部程式碼如下：

```
// Print results
Serial.print("Humidity: ");
Serial.print(humidity);
Serial.print("% , ");
Serial.print("Temperature: ");
Serial.print(temperature);
Serial.println("°C");

if ((int)temperature >= 30 || (int)humidity >= 80) {
    // 組成Line 訊息
    String message = "檢測環境發生異常，請協助儘速派人查看處理，目前環境狀態：";
    message += "\r\n 溫度=" + String((int)temperature) + "°C";
    message += "\r\n 濕度=" + String((int)humidity) + " H";
    Serial.println(message);
}
```

# 全部程式碼如下：

```
// 連線到Line API 網站
if (client.connect(host, 443)) { // 嘗試連接到LINE API
    int LEN = message.length(); // 計算訊息的長度
    // (1) 傳遞網站
    String url = "/api/notify"; // Line API 網址
    client.println("POST " + url + " HTTP/1.1"); //向LINE API發送一個HTTP POST請求
    client.print("Host: "); client.println(host); // 在HTTP請求的標頭中加入主機名稱
    // (2) 資料表頭
    client.println("Authorization: Bearer " + Linetoken);
    // 在HTTP請求的標頭中加入授權權杖
    // (3) 內容格式
    client.println("Content-Type: application/x-www-form-urlencoded");
    // 設定HTTP請求的內容類型
```

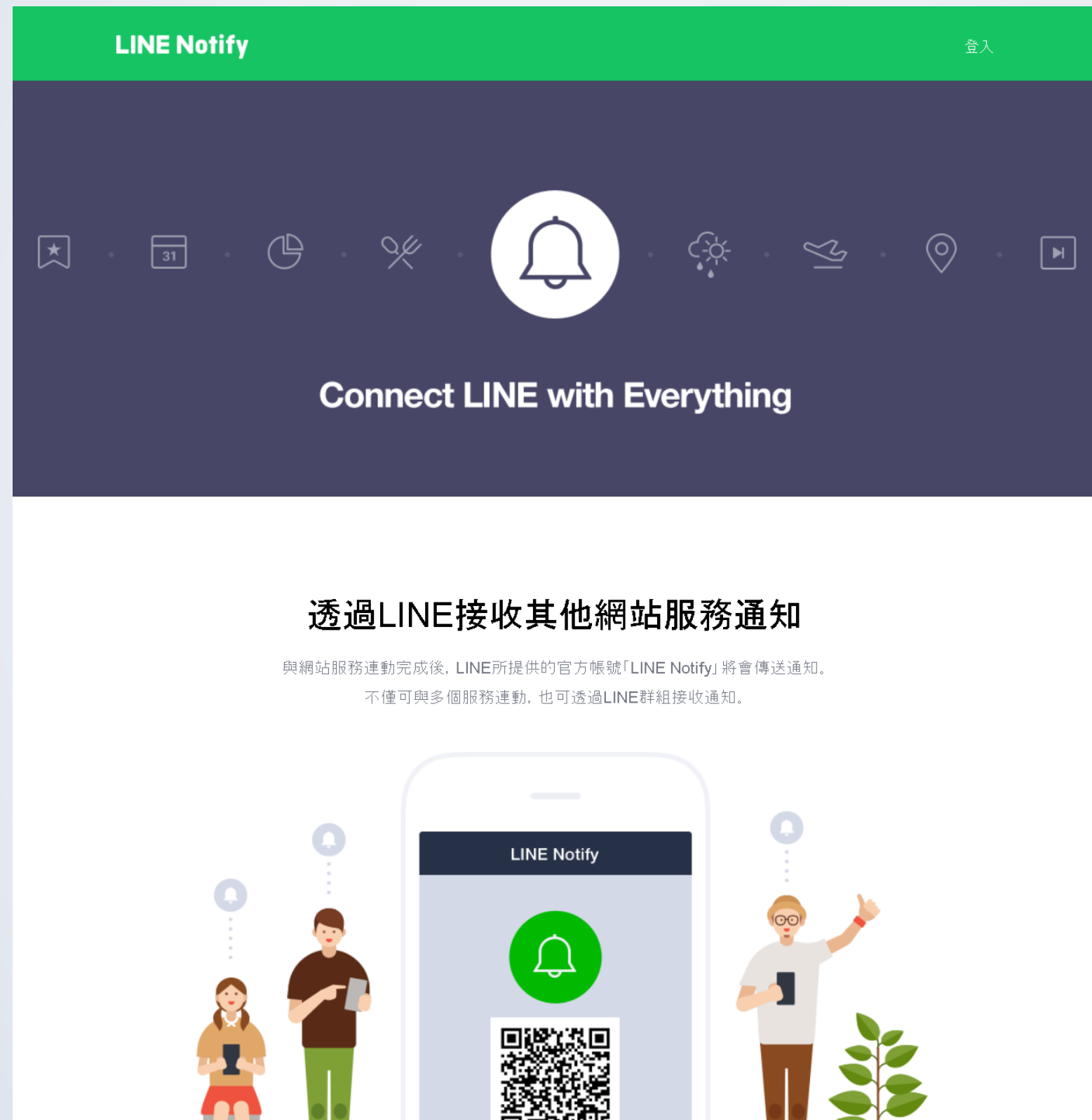


# 全部程式碼如下：

```
// (4) 資料內容
client.println("Content-Length: " + String((LEN + 8)) ); // 設定HTTP請求的內容長度
client.println(); // 空行，表示標頭和內容的分隔
client.println("message=" + message); // 將訊息加入到HTTP請求的內容中
// 等候回應
delay(2000); // 讓程式暫停2秒，等待LINE API的回應
String response = client.readString(); // 讀取LINE API的回應
// 顯示傳遞結果
Serial.println(response); // 將LINE API的回應輸出到序列埠
client.stop(); // 關閉與LINE API的連接
}

else {
    // 傳送失敗
    Serial.println("connected fail");
}
}
delay(10000);
}
```

# LINE介紹：



在使用Line傳訊前，必須取得權杖（Token）。權杖和訊息一同送至Line API，即可傳送訊息。

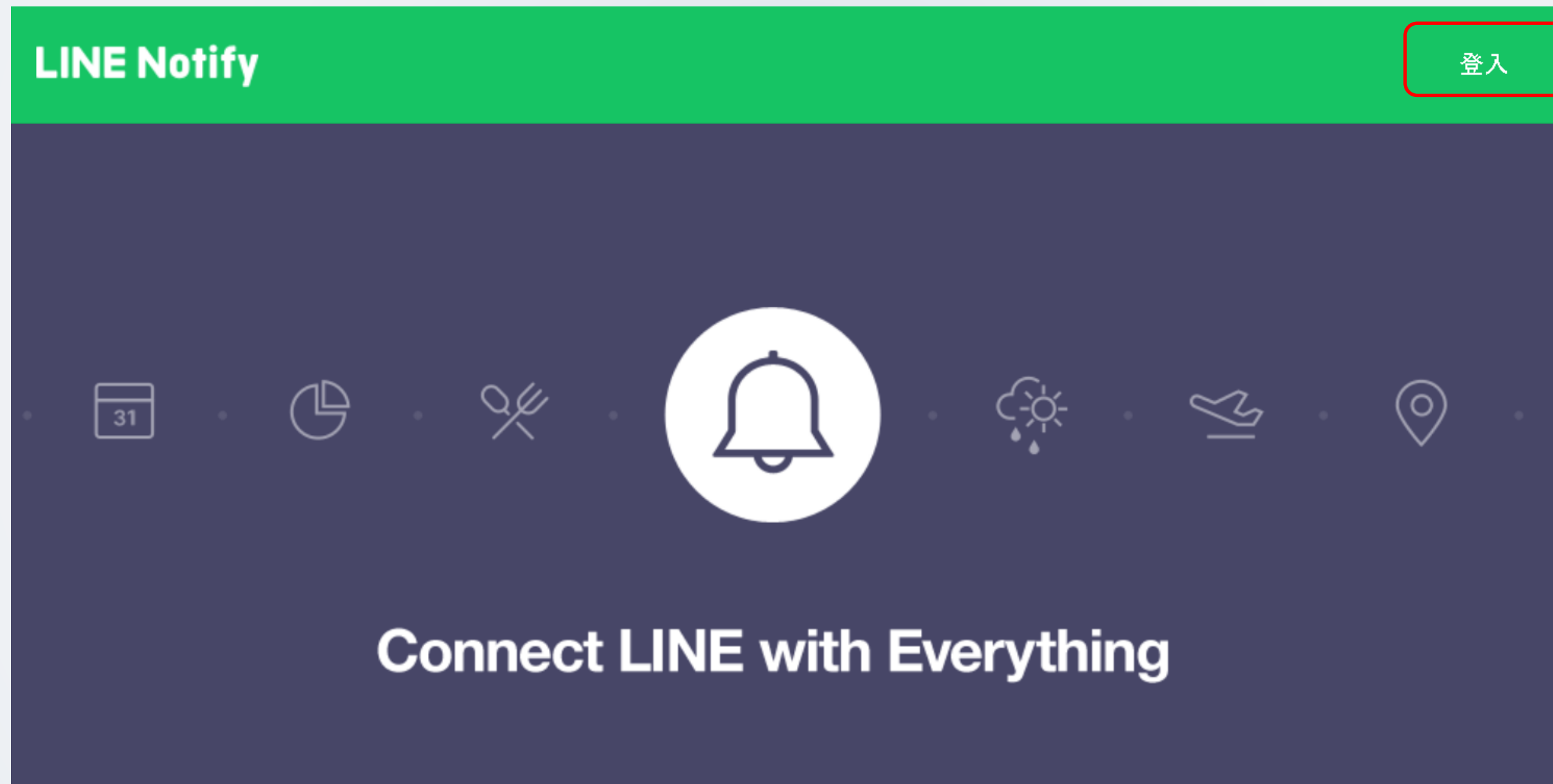
Line提供兩種傳訊工具：

1. Line Notify：僅能發送訊息。
2. Line Bot：能接收並回應訊息，適合製作應答機器人。但設定較複雜。

本次以Notify為例。

# 申請LINE Notify權杖

首先我們先到Line Notify 的網站申請開通服務，首先在瀏覽器中輸入網址：  
<https://notify-bot.line.me/>，並點選右上角的「登入」。



# 申請LINE Notify權杖

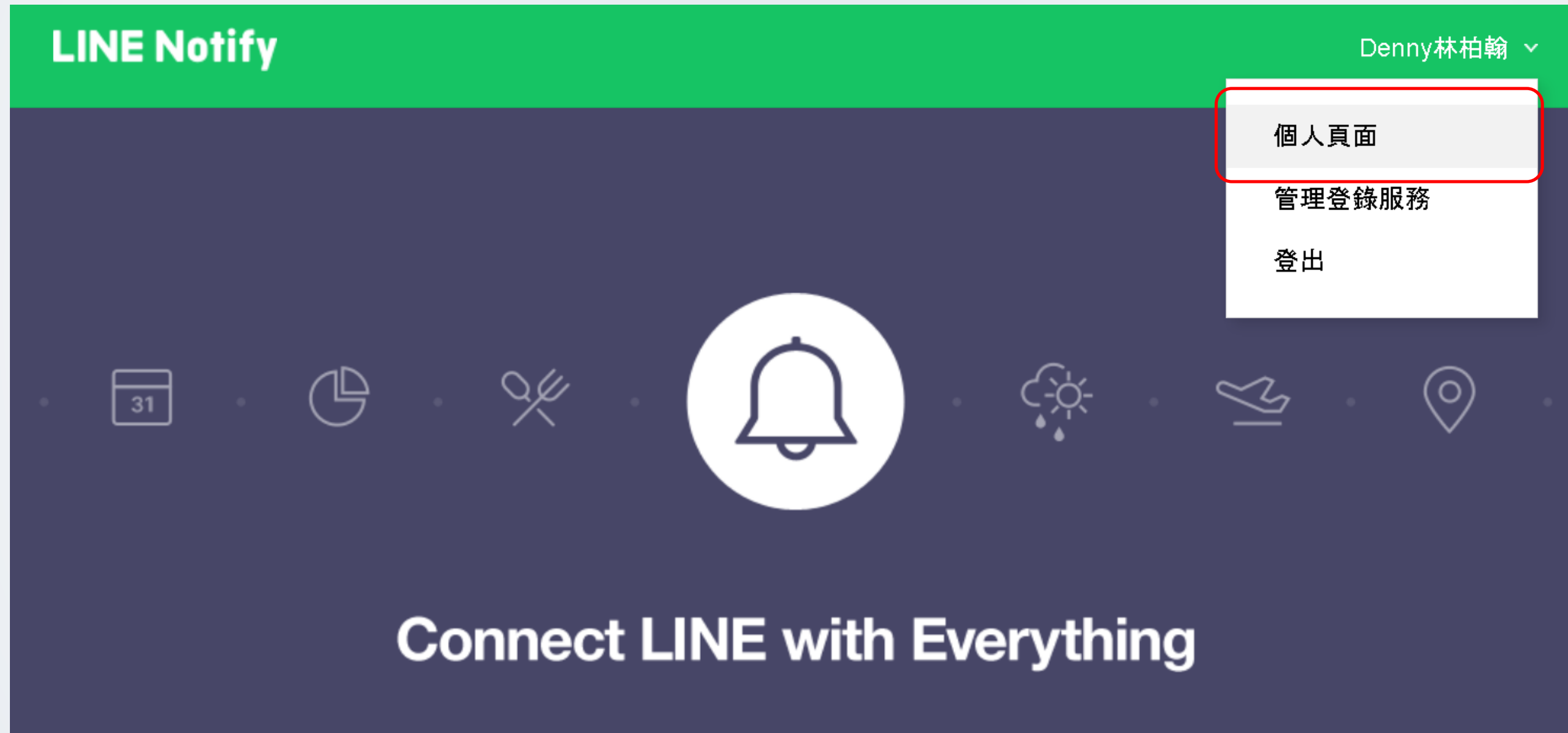
輸入自己的Line 帳號密碼後，按下方的「登入」。



The image shows the LINE login interface. At the top, the word "LINE" is displayed in large green letters. Below it, there are two input fields. The first field is labeled "電子郵件帳號" (Email Account) and contains the text "YOUR@gmail.com". The second field is labeled "密碼" (Password) and contains a series of dots. Below the password field is a large green button with the text "登入" (Login) in white. The button is highlighted with a red rounded rectangle.

# 申請LINE Notify權杖

完成登入，點選右上角的登入帳號/ 個人頁面，即可進入「已連動服務」管理頁面。







# 申請LINE Notify權杖

在頁面下方，點選「發行權杖」按鈕。

LINE Notify

Denny林柏翰 ▾

## 已連動的服務

沒有已連動的服務。

可連動的服務



GitHub >



IFTTT >



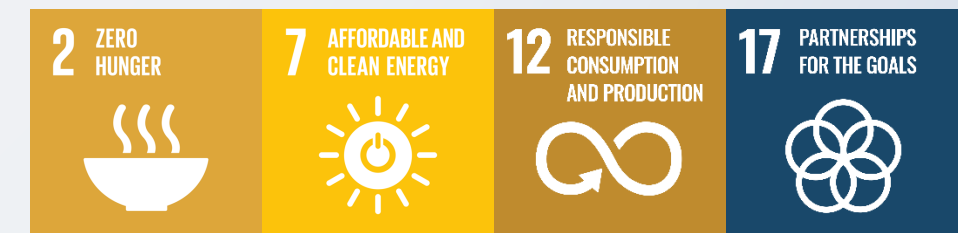
Mackerel >

## 發行存取權杖(開發人員用)

若使用個人存取權杖，不須登錄網站服務，即可設定通知。

發行權杖

LINE Notify API Document



# 申請LINE Notify權杖

1. 名稱：發送訊息通知時出現的名稱，也就是這個Line 機器人的名稱。
2. 對象：要發送訊息的群組，此處選擇「透過1 對1 聊天接收LINE Notify 通知」，也就是傳訊給自己。
3. 完成輸入後，按下方的「發行」即可獲得一組密碼，此即為權杖。

發行權杖

請填寫權杖名稱(將於傳送提醒時顯示)

AHT15溫濕度監測

1 輸入通知機器人的名稱

請選擇您要接收通知的聊天室。

Search by group name

2 選擇要通知的群組

 透過1對1聊天接收LINE Notify的通知

 106金城-體育班912

 110英語配音

 111學年度(112)申請入學群組

 112-1 四訊一乙 歷史與文化(一)

※若公開個人存取權杖，第三者將能取得您所連動的聊天室名稱及個人資料上的姓名。

3

發行

# 申請LINE Notify權杖

拿到權杖後，點選下方的「複製」按鈕將密碼複製起來，存在記事本中避免遺失，若遺失也不用緊張，可重新再申請一次，不過每次的權杖密碼都是不相同的，雖然如此，後申請的密碼也不會讓前一次申請的失效。

已發行的權杖如下。

TFHXST, [REDACTED] JgyFwmGl

若離開此頁面，將不會再顯示新發行的權杖。離開頁面前，請先複製權杖。

複製

關閉



# 申請LINE Notify權杖

確認申請完成，將會產生一個連動服務。

LINE Notify

Denny林柏翰 ▾

## 已連動的服務

此為已連動的服務一覽。若想解除連動，請按「解除」鍵。



>



2024.01.25 12:36

從 **AHT15溫濕度監測**  
傳送至 **Denny林柏翰**

解除

## 發行存取權杖(開發人員用)

若使用個人存取權杖，不須登錄網站服務，即可設定通知。

# 程式講解

這段程式包含：

1. `#include <WiFi.h>` 引入了WiFi函式庫，提供了用於管理WiFi連接的類別。
2. `#include <WiFiClientSecure.h>` 該函式庫提供了一個用於建立安全的客戶端連接的類別。
3. 下段程式碼是前兩行程式碼是設定WiFi的名稱（SSID）和密碼。  
(注意：大部分的ESP32只支援2.4GHz頻段，5.0GHz會收不到)

```
1  #include <Wire.h>
2  #include <WiFi.h>
3  #include <WiFiClientSecure.h>
4
5  #define AHT15_ADDRESS 0x38
6  #define AHT15_RESET 0xBA
7  #define AHT15_INIT 0xBE
8  #define AHT15_START_MEASUREMENT 0xAC
9
10 char SSID[] = "SSID"; // WiFi名稱
11 char PASSWORD[] = "SSID Password"; // WiFi密碼
12 String Linetoken = "Line Token"; // LINE通知的權杖
13 WiFiClientSecure client; // 宣告一個加密的WiFi 連線端
14 char host[] = "notify-api.line.me"; // 宣告Line API 網站
```



# 程式講解

這段程式主要是在寫搜尋網路時顯示" ..... "  
，並在連上網路時顯示網路IP。  
要注意的是如果使用是ESP32 1.0.6版本以上的核心  
必須加上 `client.setInsecure();`  
才能避免SSL問題。

```
16 void setup() {  
17     Wire.begin();  
18     Serial.begin(115200);  
19  
20     Serial.print("Connecting Wifi: ");  
21     Serial.println(SSID);  
22     WiFi.begin(SSID, PASSWORD); // 連接到WiFi  
23     while (WiFi.status() != WL_CONNECTED) {  
24         Serial.print(".");  
25         delay(500);  
26     }  
27     // 連線成功，顯示取得的IP  
28     Serial.println("");  
29     Serial.println("WiFi connected");  
30     Serial.println("IP address: ");  
31     IPAddress ip = WiFi.localIP();  
32     Serial.println(ip);  
33     client.setInsecure(); // 讀者若使用ESP32 1.0.6 版核心須加上這句避免SSL 問題  
34  
35     // Reset sensor  
36     Wire.beginTransmission(AHT15_ADDRESS);  
37     Wire.write(AHT15_RESET);  
38     Wire.endTransmission();  
39     delay(20);  
40  
41     // Initialize sensor  
42     Wire.beginTransmission(AHT15_ADDRESS);  
43     Wire.write(AHT15_INIT);  
44     Wire.write(0x08);  
45     Wire.write(0x00);  
46     Wire.endTransmission();  
47     delay(10);  
48 }
```

# 程式講解

這段程式碼是寫在loop副程式中：

上段程式是在寫通過LINE發送警報的觸發條件為，當環境溫度高於30度或濕度高於80。

剩下的程式是HTTP協議與LINE API進行通信的標準方式。每一步都是必要的，並且順序也不能改變。

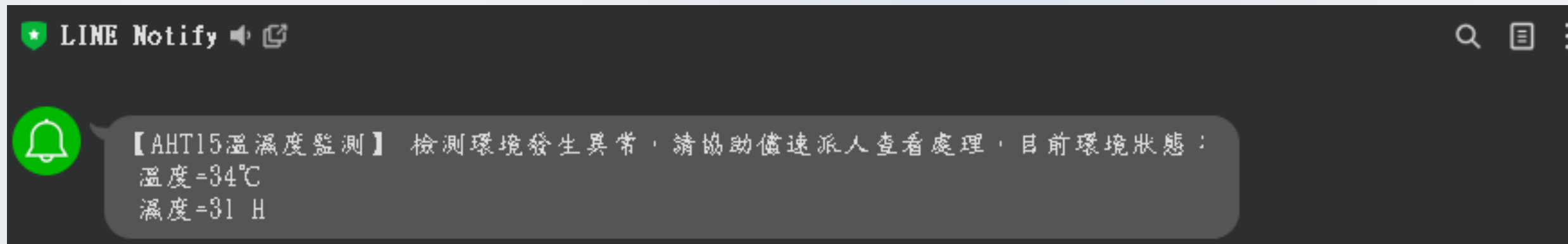
當無法連接到Line API，則顯示"connected fail"。

等待10秒後，再全部重新測量，仍然異常，則重複上述過程。

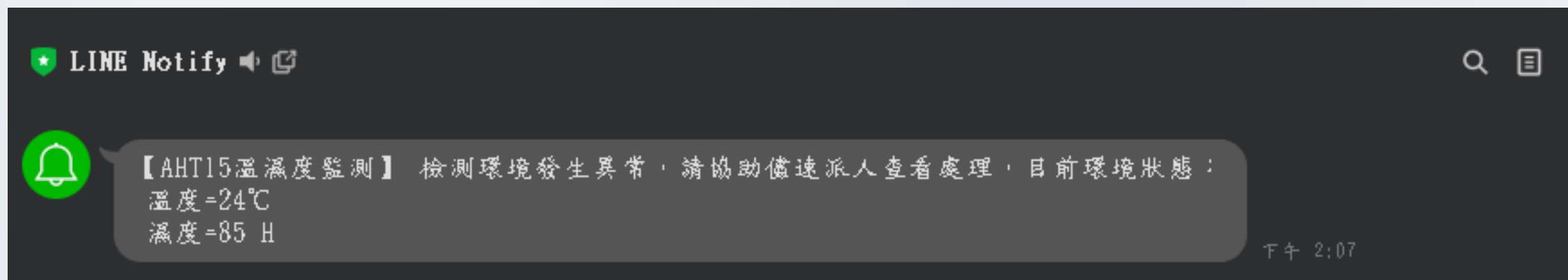
```
81  if ((int)temperature >= 30 || (int)humidity >= 80) {
82      // 組成Line 訊息
83      String message = "檢測環境發生異常，請協助儘速派人查看處理，目前環境狀態：";
84      message += "\r\n 溫度=" + String((int)temperature) + "°C";
85      message += "\r\n 濕度=" + String((int)humidity) + " H";
86      Serial.println(message);
87      // 連線到Line API 網站
88      if (client.connect(host, 443)) { // 連接到LINE API
89          int LEN = message.length();
90          //(1) 傳遞網站
91          String url = "/api/notify"; // Line API 網址
92          client.println("POST " + url + " HTTP/1.1");
93          client.print("Host: "); client.println(host); // Line API 網站
94          //(2) 資料表頭
95          client.println("Authorization: Bearer " + Linetoken); // LINE通知的權杖
96          //(3) 內容格式
97          client.println("Content-Type: application/x-www-form-urlencoded");
98          //(4) 資料內容
99          client.println("Content-Length: " + String((LEN + 8)) ); // 訊息長度
100         client.println();
101         client.println("message=" + message); // 訊息內容
102         // 等候回應
103         delay(2000);
104         String response = client.readString(); // 讀取回應
105         // 顯示傳遞結果
106         Serial.println(response);
107         client.stop(); // 斷線，否則只能傳5 次
108     }
109     else {
110         // 傳送失敗
111         Serial.println("connected fail");
112     }
113 }
114 delay(10000);
115 }
```

# 最終執行結果

## 溫度異常警訊畫面

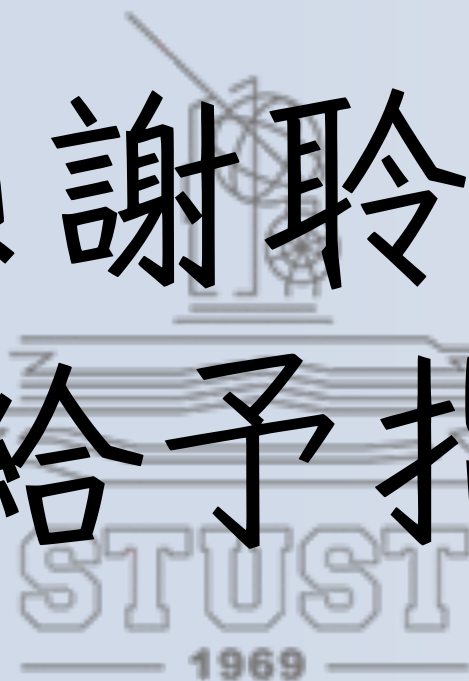


## 濕度異常警訊畫面





感謝聆聽  
給予指導



減碳漁

古都土城仔綠電創能與智動養殖  
之跨界整合永續淨零發展計畫