物聯網核心技術
期末報告

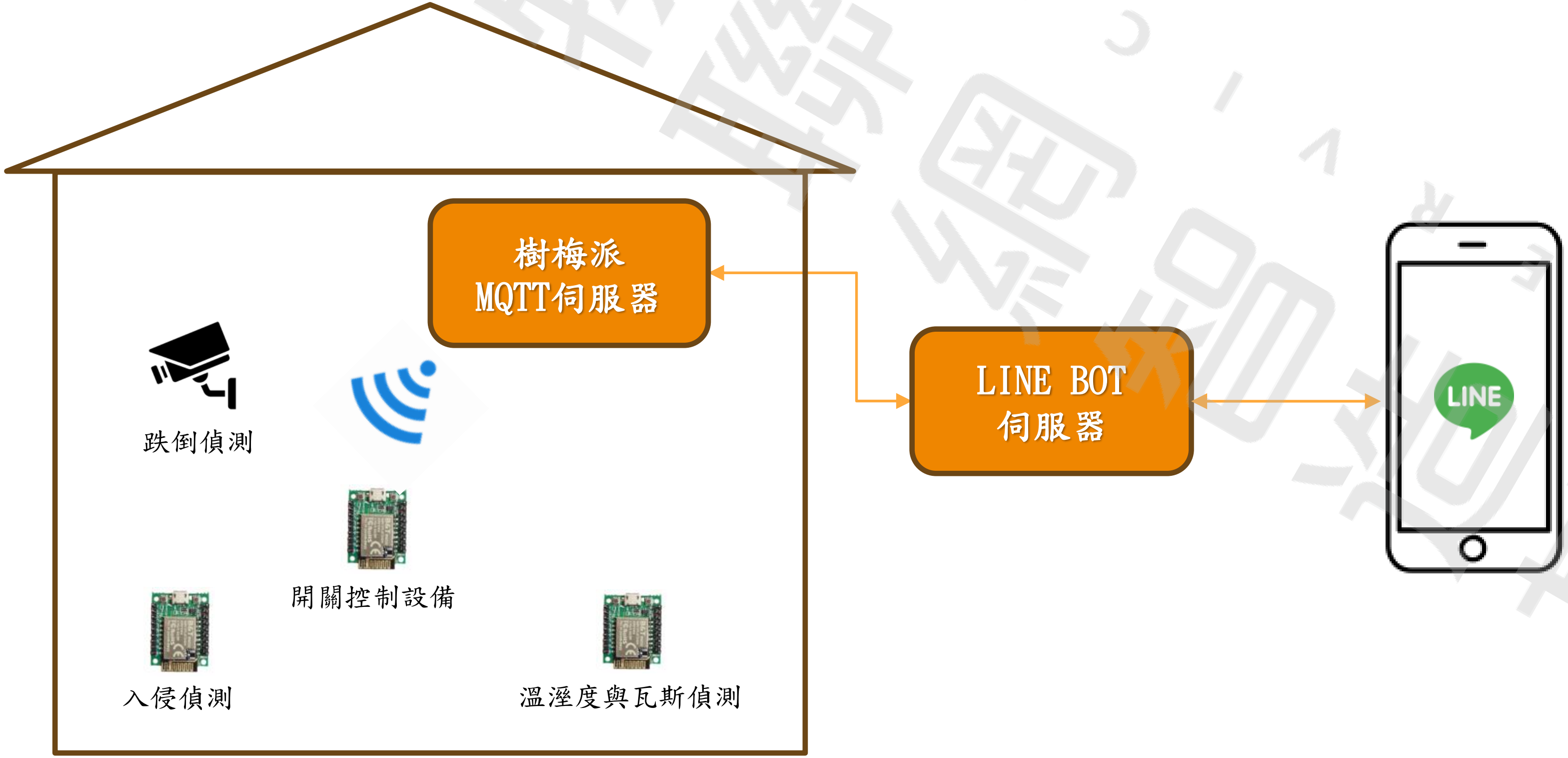
智慧居家安全

研究動機

之前在介紹有關智慧居家監視器系統時，而有感而發想製作與之相關的設計，也因為我們手邊的研究與其相似，能做到深度學習來辨識跌倒偵測。

所以我們決定研究:入侵偵測回報、溫溼度感測、煙霧感測器、智慧電燈開關、跌倒偵測再將其整合到LINE BOT。

設備通訊架構



樹梅派
MQTT伺服器

LINE BOT
伺服器

跌倒偵測

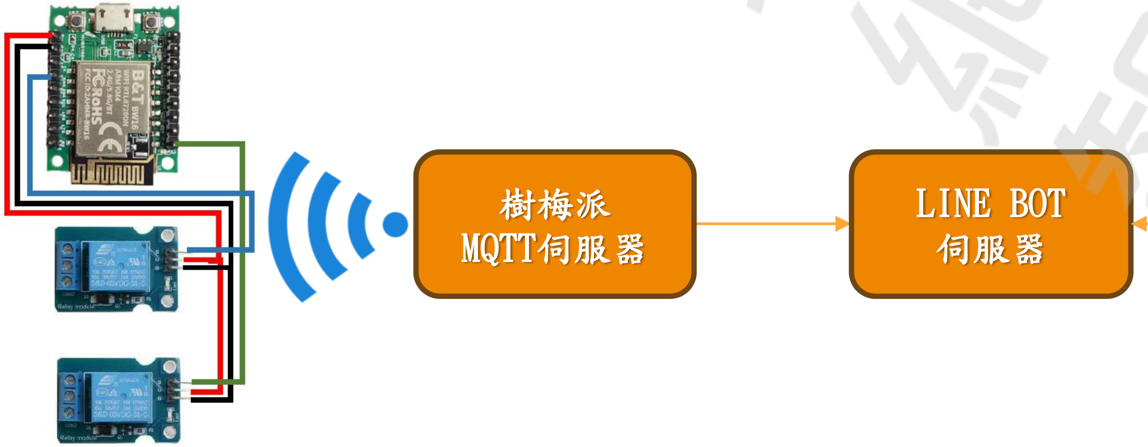
開關控制設備

入侵偵測

溫溼度與瓦斯偵測

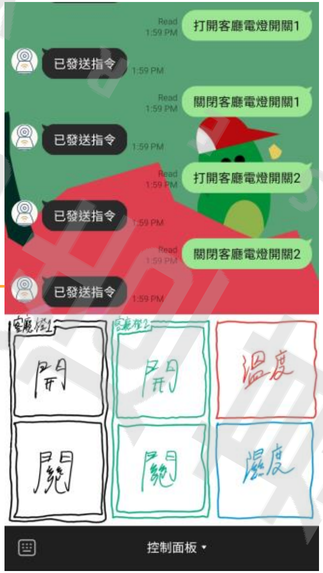
智慧電燈開關

繼電器連接到HUB 5168+，可透過LINE控制伺服器端的電燈開關，伺服器端在與家中樹梅派做同步，樹梅派在透過MQTT發送至HUB 5168+



樹梅派
MQTT伺服器

LINE BOT
伺服器



打開客廳電燈開關1 Read 1:59 PM

已發送指令 1:59 PM

關閉客廳電燈開關1 Read 1:59 PM

已發送指令 1:59 PM

打開客廳電燈開關2 Read 1:59 PM

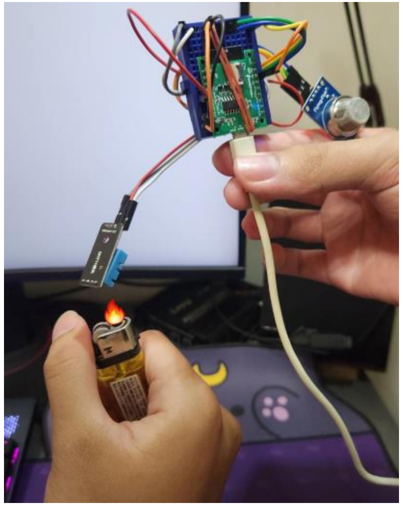
已發送指令 1:59 PM

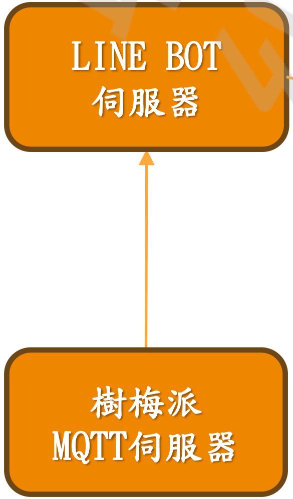
關閉客廳電燈開關2 Read 1:59 PM

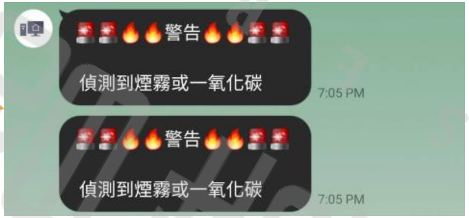
已發送指令 1:59 PM

溫溼度與煙霧感測器回報

溫溼度感測器模組 & 煙霧感測器連接到HUB 5168+，這部分是作為偵測室內和LINE通知的部分是有伺服端接收LINE的訊息(固定的)，並針對接收到的訊息做出相對應的回覆。





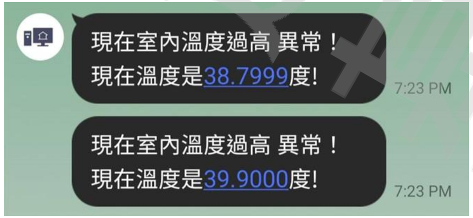


🚨🚨🔥🔥警告🔥🔥🚨🚨

偵測到煙霧或一氧化碳 7:05 PM

🚨🚨🔥🔥警告🔥🔥🚨🚨

偵測到煙霧或一氧化碳 7:05 PM

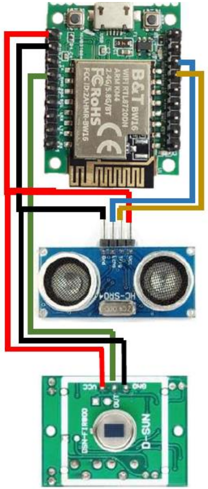


現在室內溫度過高 異常！
現在溫度是38.7999度! 7:23 PM

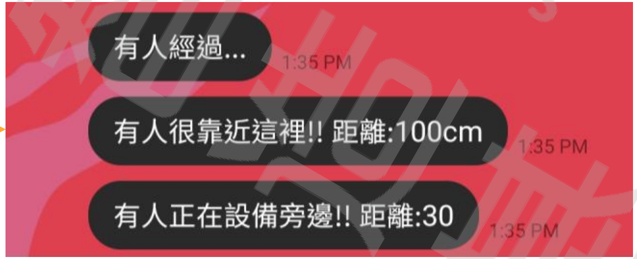
現在室內溫度過高 異常！
現在溫度是39.9000度! 7:23 PM

入侵警報系統

將超音波測距模組 & 人體紅外線感測模組連接到HUB 5168+，透過樹梅派傳送警報到LINE BOT伺服器上。



樹梅派發送警報
至
LINE BOT伺服器



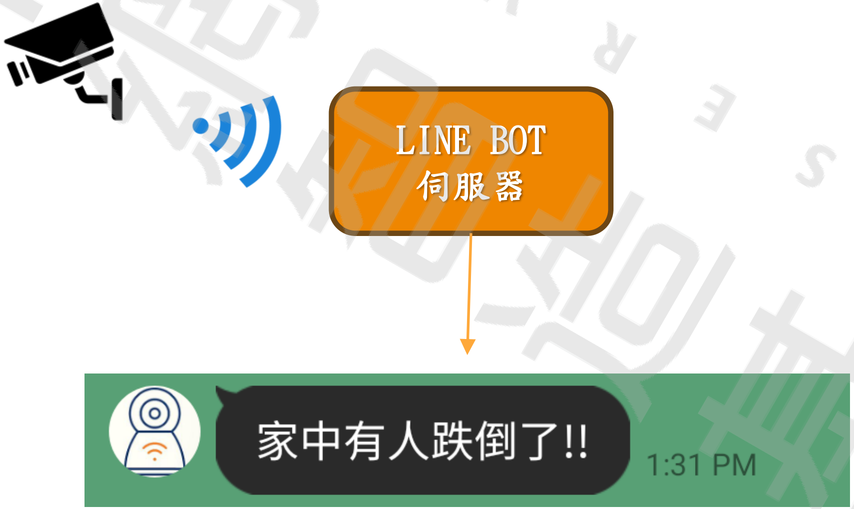
有人經過... 1:35 PM

有人很靠近這裡!! 距離:100cm 1:35 PM

有人正在設備旁邊!! 距離:30 1:35 PM

跌倒偵測系統

用YOLO訓練跌倒偵測模組。在家中攝影機開啓的狀態下可以即時判斷畫面中是否有人摔倒，若有偵測到將發送警報至LINE。準確率約75%

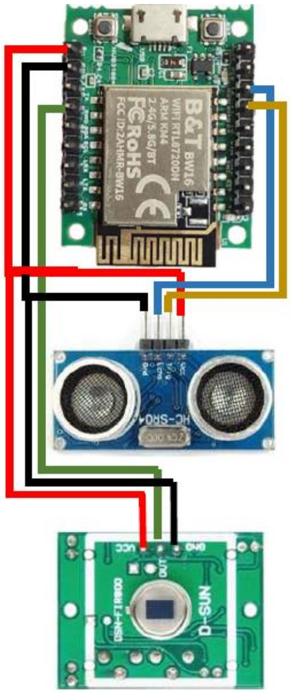


LINE BOT
伺服器

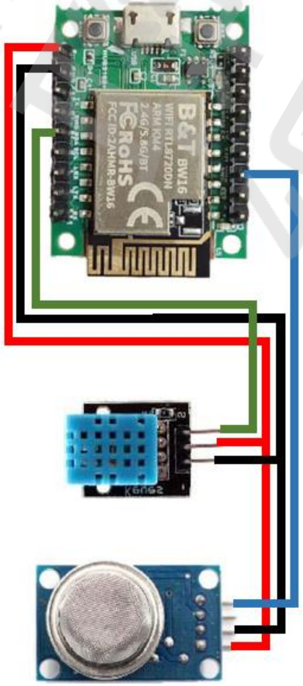
家中有人跌倒了!!

1:31 PM

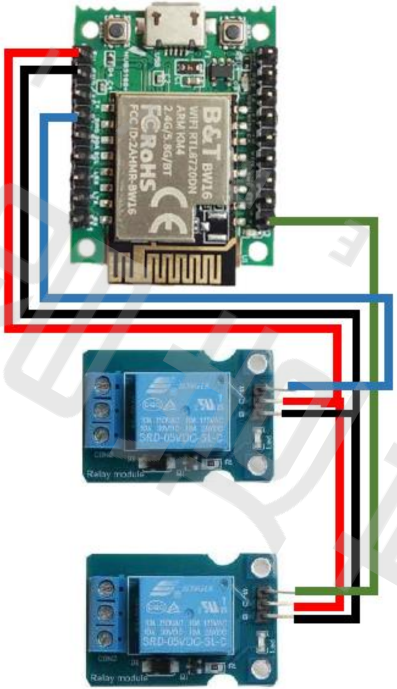
電路圖



超音波&人體紅外線
連HUB 5168+



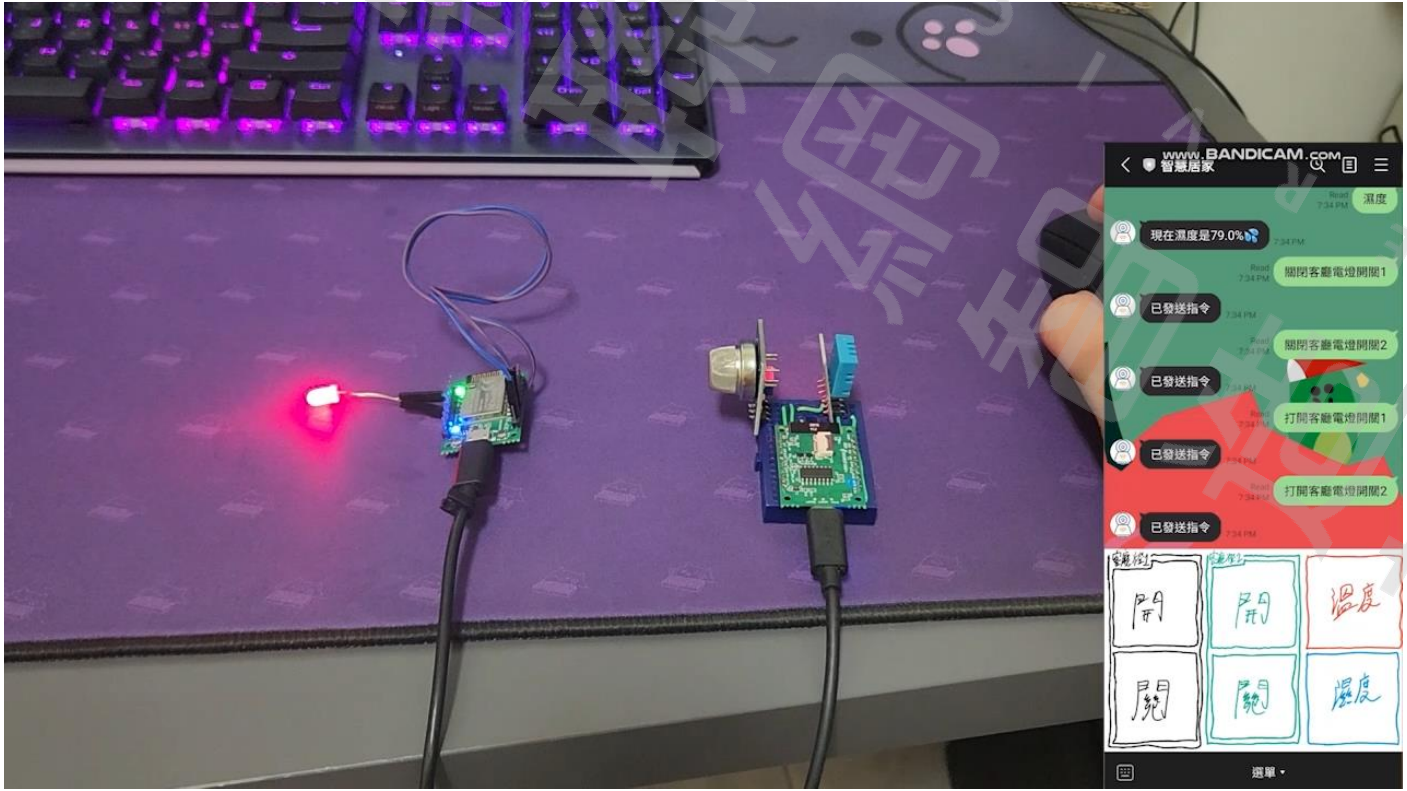
溫溼度&煙霧感測
連HUB 5168+



繼電器開關x2
連HUB 5168+

示範影片

溫溼度　瓦斯偵測　電燈遠程開關



濕度 Read 7:34 PM

現在濕度是79.0%💦 7:34 PM

關閉客廳電燈開關1 Read 7:34 PM

已發送指令 7:34 PM

關閉客廳電燈開關2 Read 7:34 PM

已發送指令 7:34 PM

打開客廳電燈開關1 Read 7:34 PM

已發送指令 7:34 PM

打開客廳電燈開關2 Read 7:34 PM

已發送指令 7:34 PM

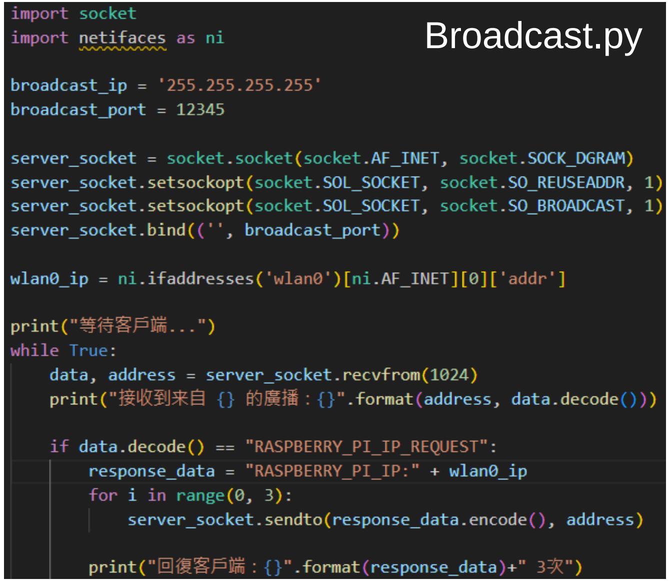
示範影片

跌倒偵測

程式碼

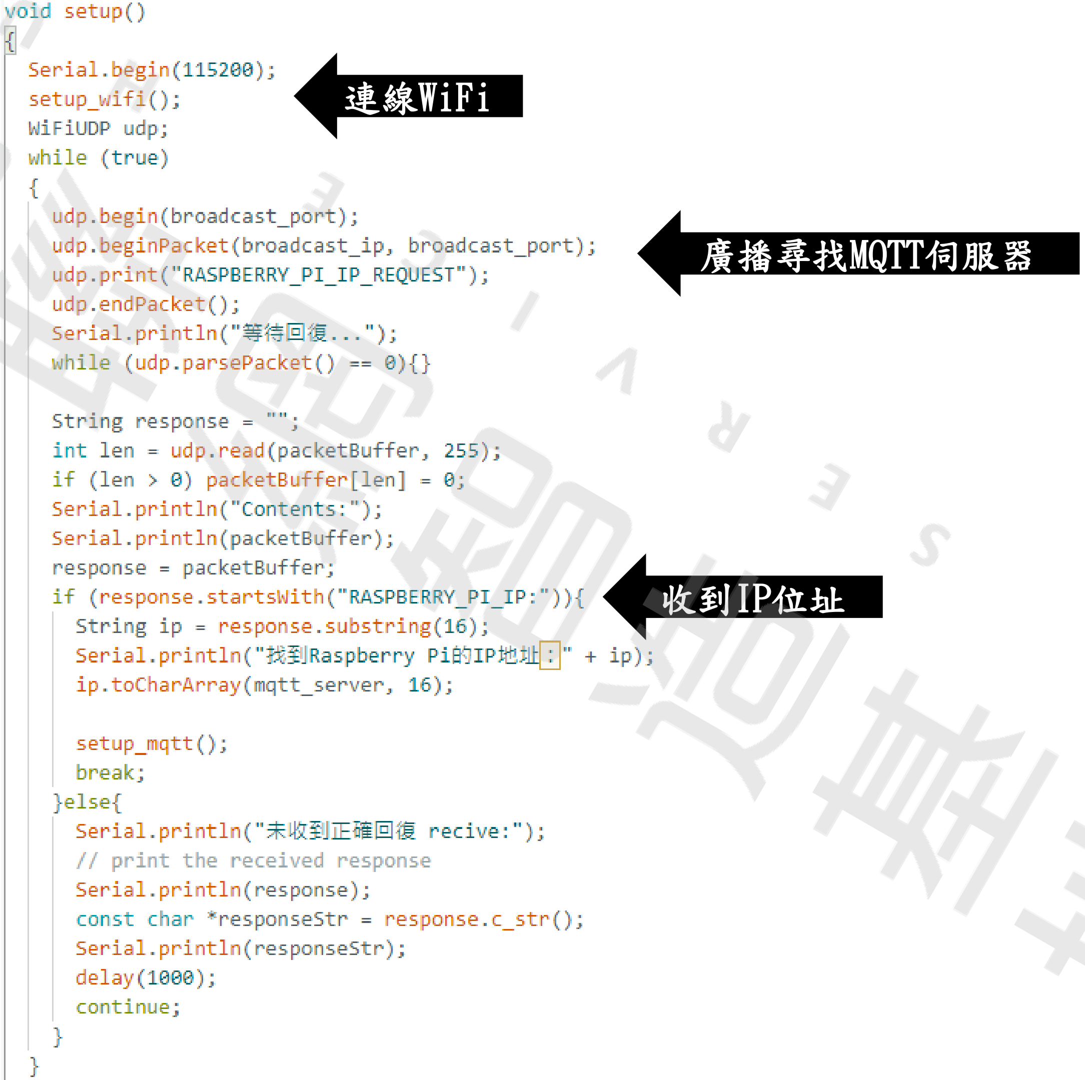
程式碼

特殊設計：物聯網設備自動尋找樹梅派IP，可以免去手動燒錄每個設備



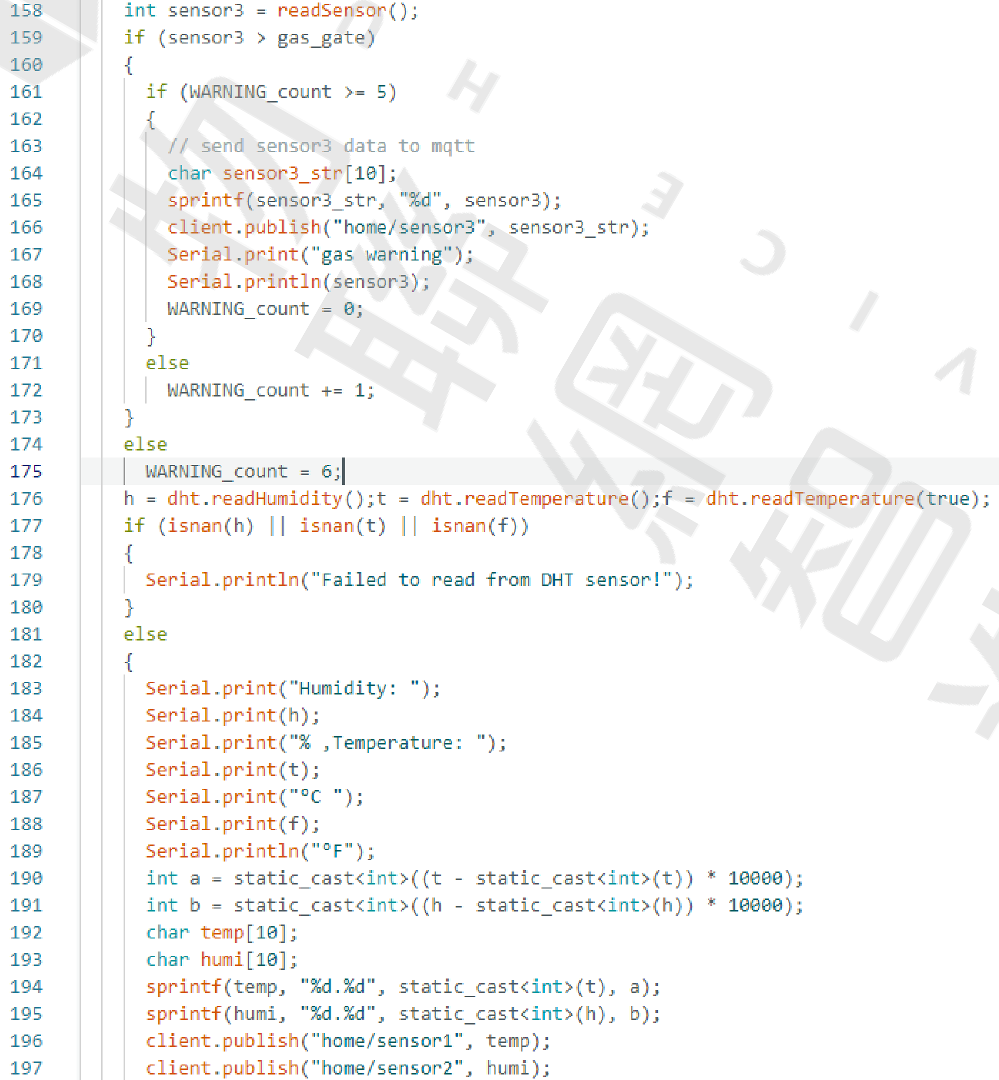
Broadcast.py

```python
import socket
import netifaces as ni

broadcast_ip = '255.255.255.255'
broadcast_port = 12345

server_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
server_socket.bind(('', broadcast_port))

wlan0_ip = ni.ifaddresses('wlan0')[ni.AF_INET][0]['addr']

print("等待客戶端...")
while True:
    data, address = server_socket.recvfrom(1024)
    print("接收到来自 {} 的廣播：{}".format(address, data.decode()))

    if data.decode() == "RASPBERRY_PI_IP_REQUEST":
        response_data = "RASPBERRY_PI_IP:" + wlan0_ip
        for i in range(0, 3):
            server_socket.sendto(response_data.encode(), address)

        print("回復客戶端：{}".format(response_data)+" 3次")
```



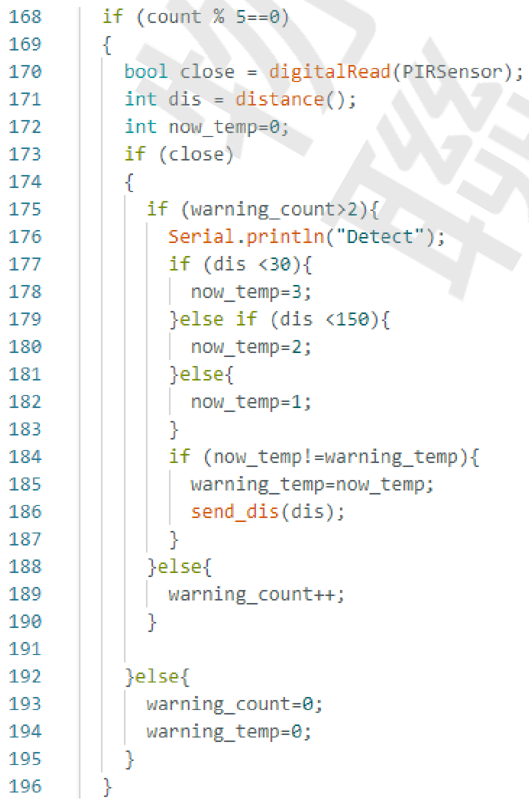
```cpp
void setup()
{
  Serial.begin(115200);
  setup_wifi();
  WiFiUDP udp;
  while (true)
  {
    udp.begin(broadcast_port);
    udp.beginPacket(broadcast_ip, broadcast_port);
    udp.print("RASPBERRY_PI_IP_REQUEST");
    udp.endPacket();
    Serial.println("等待回復...");
    while (udp.parsePacket() == 0){}

    String response = "";
    int len = udp.read(packetBuffer, 255);
    if (len > 0) packetBuffer[len] = 0;
    Serial.println("Contents:");
    Serial.println(packetBuffer);
    response = packetBuffer;
    if (response.startsWith("RASPBERRY_PI_IP:")){
      String ip = response.substring(16);
      Serial.println("找到Raspberry Pi的IP地址：" + ip);
      ip.toCharArray(mqtt_server, 16);

      setup_mqtt();
      break;
    }else{
      Serial.println("未收到正確回復 recive:");
      // print the received response
      Serial.println(response);
      const char *responseStr = response.c_str();
      Serial.println(responseStr);
      delay(1000);
      continue;
    }
  }
```

溫溼度更新

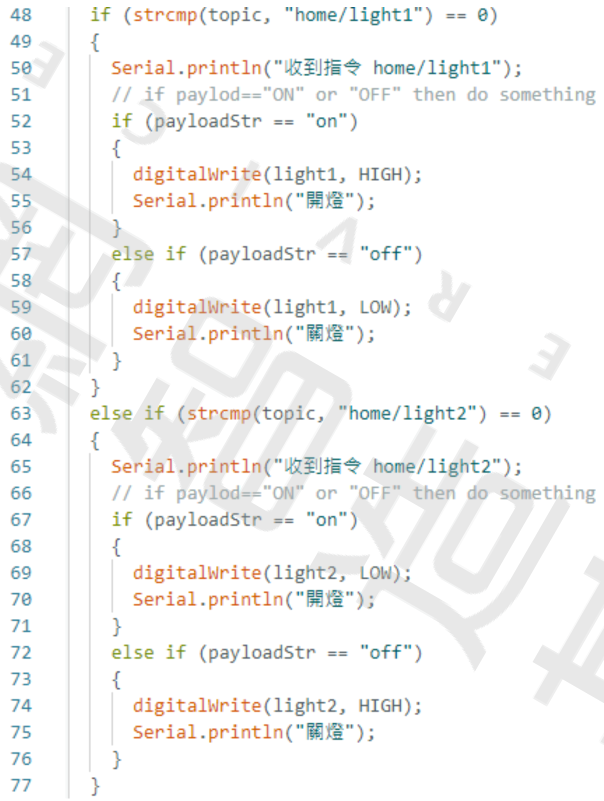


```
int sensor3 = readSensor();
if (sensor3 > gas_gate)
{
  if (WARNING_count >= 5)
  {
    // send sensor3 data to mqtt
    char sensor3_str[10];
    sprintf(sensor3_str, "%d", sensor3);
    client.publish("home/sensor3", sensor3_str);
    Serial.print("gas warning");
    Serial.println(sensor3);
    WARNING_count = 0;
  }
  else
    WARNING_count += 1;
}
else
  WARNING_count = 6;
h = dht.readHumidity();t = dht.readTemperature();f = dht.readTemperature(true);
if (isnan(h) || isnan(t) || isnan(f))
{
  Serial.println("Failed to read from DHT sensor!");
}
else
{
  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print("% ,Temperature: ");
  Serial.print(t);
  Serial.print("°C ");
  Serial.print(f);
  Serial.println("°F");
  int a = static_cast<int>((t - static_cast<int>(t)) * 10000);
  int b = static_cast<int>((h - static_cast<int>(h)) * 10000);
  char temp[10];
  char humi[10];
  sprintf(temp, "%d.%d", static_cast<int>(t), a);
  sprintf(humi, "%d.%d", static_cast<int>(h), b);
  client.publish("home/sensor1", temp);
  client.publish("home/sensor2", humi);
```

距離感測 & 電燈開關



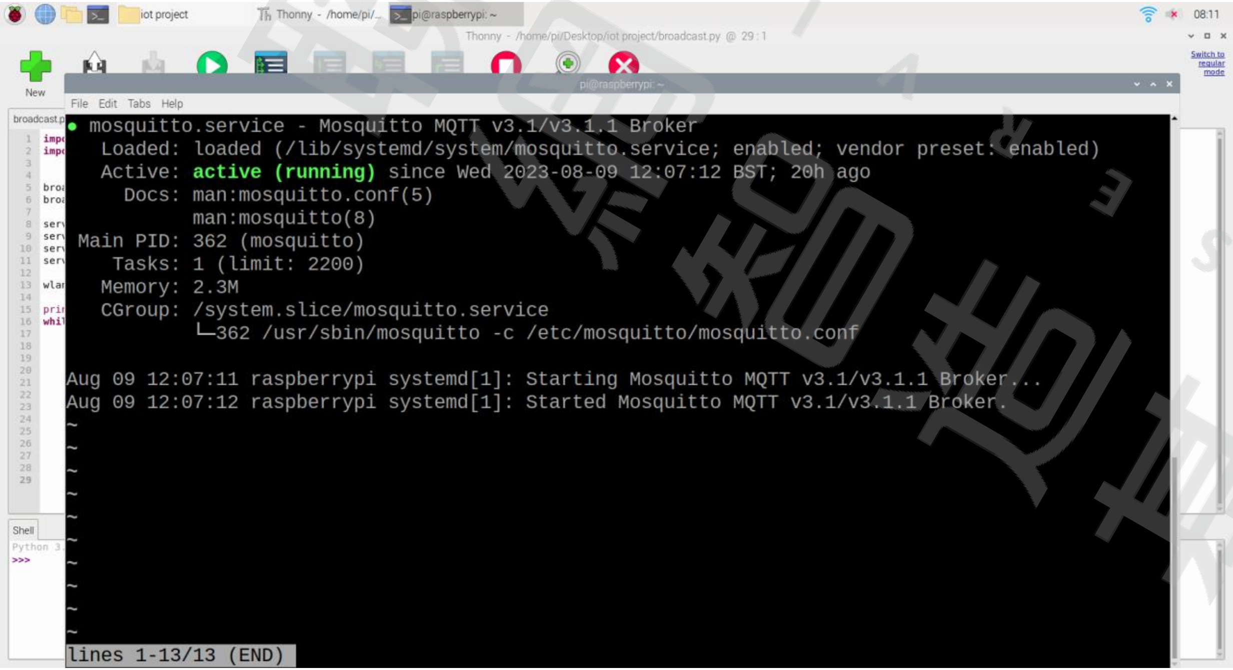
```
if (count % 5==0)
{
  bool close = digitalRead(PIRSensor);
  int dis = distance();
  int now_temp=0;
  if (close)
  {
    if (warning_count>2){
      Serial.println("Detect");
      if (dis <30){
        now_temp=3;
      }else if (dis <150){
        now_temp=2;
      }else{
        now_temp=1;
      }
      if (now_temp!=warning_temp){
        warning_temp=now_temp;
        send_dis(dis);
      }
    }else{
      warning_count++;
    }

  }else{
    warning_count=0;
    warning_temp=0;
  }
}
```



```
if (strcmp(topic, "home/light1") == 0)
{
  Serial.println("收到指令 home/light1");
  // if paylod=="ON" or "OFF" then do something
  if (payloadStr == "on")
  {
    digitalWrite(light1, HIGH);
    Serial.println("開燈");
  }
  else if (payloadStr == "off")
  {
    digitalWrite(light1, LOW);
    Serial.println("關燈");
  }
}
else if (strcmp(topic, "home/light2") == 0)
{
  Serial.println("收到指令 home/light2");
  // if paylod=="ON" or "OFF" then do something
  if (payloadStr == "on")
  {
    digitalWrite(light2, LOW);
    Serial.println("開燈");
  }
  else if (payloadStr == "off")
  {
    digitalWrite(light2, HIGH);
    Serial.println("關燈");
  }
}
```

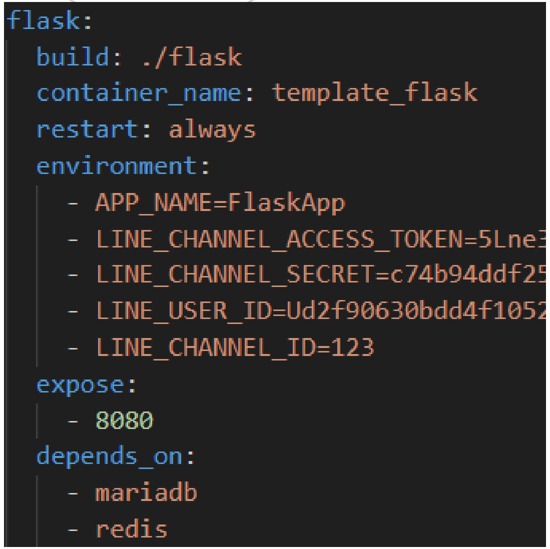
樹梅派上的 MQTT伺服器

Moquitto.conf
Set as
Listener 1883
Allow_anonymous true

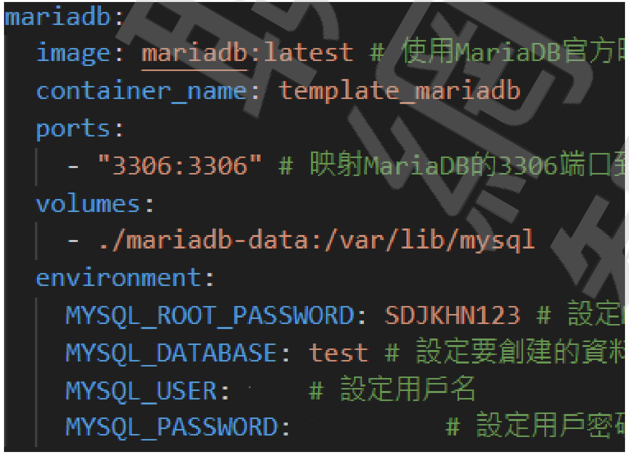


```
● mosquitto.service - Mosquitto MQTT v3.1/v3.1.1 Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-08-09 12:07:12 BST; 20h ago
     Docs: man:mosquitto.conf(5)
           man:mosquitto(8)
 Main PID: 362 (mosquitto)
    Tasks: 1 (limit: 2200)
   Memory: 2.3M
   CGroup: /system.slice/mosquitto.service
           └─362 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

Aug 09 12:07:11 raspberrypi systemd[1]: Starting Mosquitto MQTT v3.1/v3.1.1 Broker...
Aug 09 12:07:12 raspberrypi systemd[1]: Started Mosquitto MQTT v3.1/v3.1.1 Broker.
~
~
~
~
~
~
~
~
~
~
lines 1-13/13 (END)
```

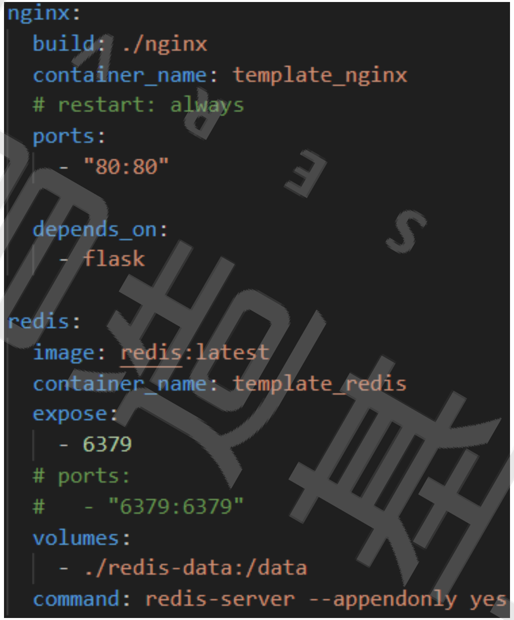
LINE BOT伺服器 設定檔



```
flask:
  build: ./flask
  container_name: template_flask
  restart: always
  environment:
    - APP_NAME=FlaskApp
    - LINE_CHANNEL_ACCESS_TOKEN=5Lne3
    - LINE_CHANNEL_SECRET=c74b94ddf25
    - LINE_USER_ID=Ud2f90630bdd4f1052
    - LINE_CHANNEL_ID=123
  expose:
    - 8080
  depends_on:
    - mariadb
    - redis
```



```
mariadb:
  image: mariadb:latest # 使用MariaDB官方
  container_name: template_mariadb
  ports:
    - "3306:3306" # 映射MariaDB的3306端口
  volumes:
    - ./mariadb-data:/var/lib/mysql
  environment:
    MYSQL_ROOT_PASSWORD: SDJKHN123 # 設定
    MYSQL_DATABASE: test # 設定要創建的資
    MYSQL_USER:       # 設定用戶名
    MYSQL_PASSWORD:            # 設定用戶密
```

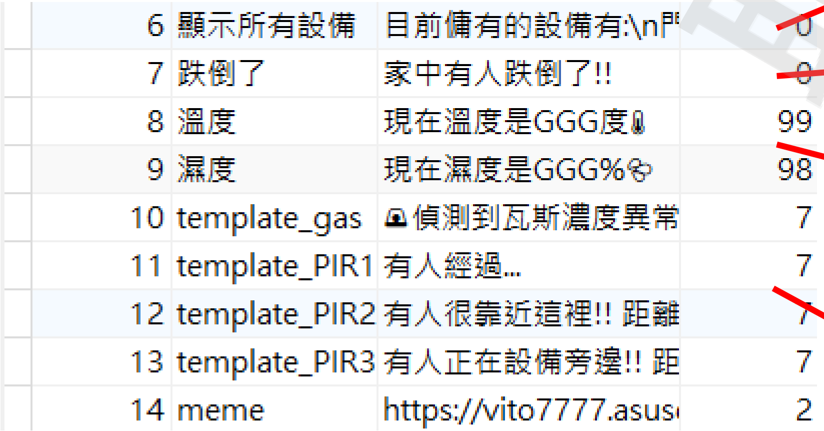


```
nginx:
  build: ./nginx
  container_name: template_nginx
  # restart: always
  ports:
    - "80:80"

  depends_on:
    - flask

redis:
  image: redis:latest
  container_name: template_redis
  expose:
    - 6379
  # ports:
  #   - "6379:6379"
  volumes:
    - ./redis-data:/data
  command: redis-server --appendonly yes
```

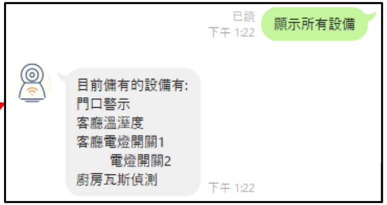
MariaDB
的reply表

我們建立LINE BOT時也建立一SQL來存放

LINE BOT要回復的相對應內容

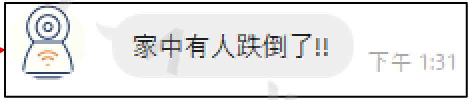


6	顯示所有設備	目前備有的設備有:\n門	0
7	跌倒了	家中有人跌倒了!!	0
8	溫度	現在溫度是GGG度🌡	99
9	濕度	現在濕度是GGG%💦	98
10	template_gas	🚨偵測到瓦斯濃度異常	7
11	template_PIR1	有人經過...	7
12	template_PIR2	有人很靠近這裡!! 距離	7
13	template_PIR3	有人正在設備旁邊!! 距	7
14	meme	https://vito7777.asus	2

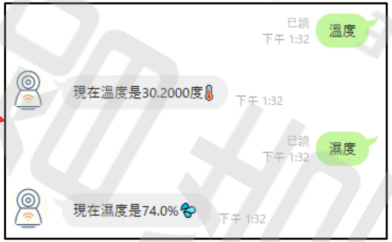


已讀
下午 1:22
顯示所有設備

目前備有的設備有:
門口警示
客廳溫溼度
客廳電燈開關1
電燈開關2
廚房瓦斯偵測
下午 1:22



家中有人跌倒了!!
下午 1:31

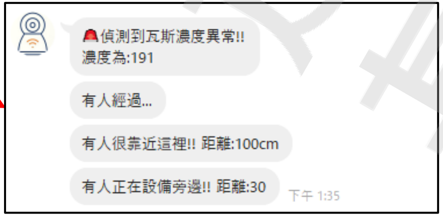


已讀
下午 1:32
溫度

現在溫度是30.2000度🌡
下午 1:32

已讀
下午 1:32
濕度

現在濕度是74.0%💦
下午 1:32



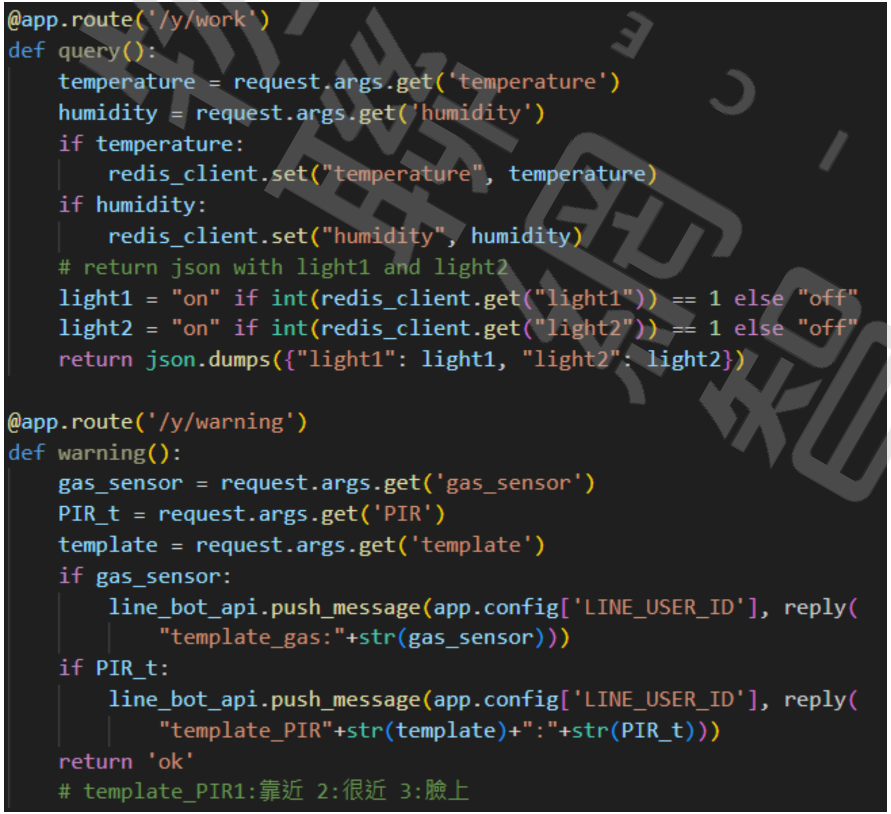
🚨偵測到瓦斯濃度異常!!
濃度為:191

有人經過...

有人很靠近這裡!! 距離:100cm

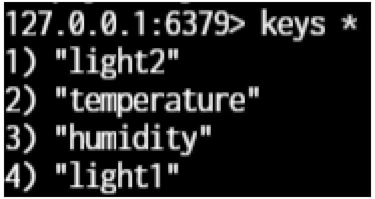
有人正在設備旁邊!! 距離:30
下午 1:35

LINE BOT

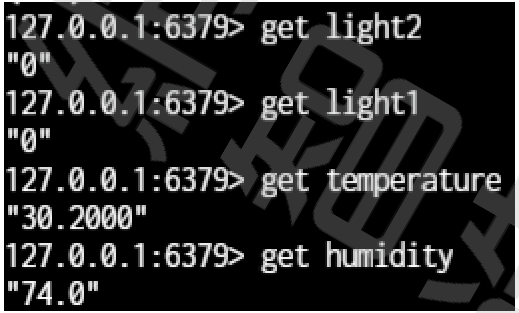


```python
@app.route('/y/work')
def query():
    temperature = request.args.get('temperature')
    humidity = request.args.get('humidity')
    if temperature:
        redis_client.set("temperature", temperature)
    if humidity:
        redis_client.set("humidity", humidity)
    # return json with light1 and light2
    light1 = "on" if int(redis_client.get("light1")) == 1 else "off"
    light2 = "on" if int(redis_client.get("light2")) == 1 else "off"
    return json.dumps({"light1": light1, "light2": light2})

@app.route('/y/warning')
def warning():
    gas_sensor = request.args.get('gas_sensor')
    PIR_t = request.args.get('PIR')
    template = request.args.get('template')
    if gas_sensor:
        line_bot_api.push_message(app.config['LINE_USER_ID'], reply(
            "template_gas:"+str(gas_sensor)))
    if PIR_t:
        line_bot_api.push_message(app.config['LINE_USER_ID'], reply(
            "template_PIR"+str(template)+":"+str(PIR_t)))
    return 'ok'
    # template_PIR1:靠近 2:很近 3:臉上
```

Redis
變數表



```
127.0.0.1:6379> keys *
1) "light2"
2) "temperature"
3) "humidity"
4) "light1"
```



```
127.0.0.1:6379> get light2
"0"
127.0.0.1:6379> get light1
"0"
127.0.0.1:6379> get temperature
"30.2000"
127.0.0.1:6379> get humidity
"74.0"
```

謝謝大家。👏