

# 第十四屆旺宏科學獎

## 成果報告書

參賽編號：SA14-164

作品名稱：行動多功能智慧盆栽系統

姓名：徐唯哲

關鍵字：行動智慧盆栽、ArduinoWeb Server、LEGO

## 壹、研究動機

台灣氣候多變，陰晴不定，土壤的乾濕度與日照通風和植物生長息息相關，若不多加注意可能導致植物死亡。一般日常生活中，植物大多由人力來照顧，但對於生活忙碌或者記性不佳的人來說，想讓植物能長久存活幾乎是不可能的事。因而產生「自動幫使用者照顧植物」的想法，以自動照顧為起點出發發現了「物聯網」的概念，所謂的物聯網就是透過網路將生活中的事物串聯起來，讓使用者能藉此讀取物件的資訊，增加生活的便利性或工作的效率等。於是開始思考如何開發一個完善具有人性化、便利性、實用性的系統。

## 貳、研究目的

現今的時代，講求自動、遙控與行動裝置配合的功能，所以此專題以雲端控制來取代人力為目標，能藉由機器的運作來幫使用者照顧植物盆栽。利用遠端控制技術，即使人離開盆栽，也可利用行動裝置藉由網路接收此溼度值和溫度值，進一步控制溫、溼度的變化，在將來能應用在各種裝置上，為生活帶來更多的便利。因此訂定了以下六個目標：

- 一、使用 App inventor2 來撰寫行動裝置的應用程式
- 二、透過溫濕度感測器及網路伺服器，傳回溫度與濕度的數值
- 三、以 LEGO 積木完成澆水及遮罩等動作
- 四、利用網路伺服器架設表單及繼電器控制燈泡與風扇
- 五、架設網路攝影機觀看盆栽的即時影像
- 六、區分手動、自動及語音模式方便使用者因應情況操作

## 參、研究過程

### 一、研究步驟

經過討論後，因為台灣的氣候多變與植物息息相關、具有非常豐富的植物種類，很適合在家中種植各類植物，但因為有許多上班族等生活忙碌的族群而無法照顧家中的盆栽導致植物通常都生存不久，進而決定製作關於植物的照顧服務系統。依據下圖的流程開始著手研究，從專業書籍或網路中找尋相關的概念、技術所需器材等；利用相關實習課程及組員的技術設計整個系統大概的架構和各項細節；將整個雛型製作出來後開始調整、改良各種功能細節；依據研究過程的記錄及數據製作書面資料。



圖 1 研究流程圖

## 二、系統概念

利用溫濕度感測器取得植物的相關數據後回傳至 Arduino，並透過撰寫程式架設網站及無線網路基地台將資料送到行動裝置上的 APP 介面中。接下來依照使用者的選擇來作對應的動作，假如為自動模式，植物的數據經過伺服器判斷後會送出訊號給 NXT 讀取執行自動澆水、升降遮罩的功能防止植物枯死；如果是手動模式則會讓使用者透過行動裝置連線至 NXT 直接對系統進行基本的功能操作；最後的語音模式能取代手動按鈕的方式直接以判斷關鍵字的方式去作對應的動作。

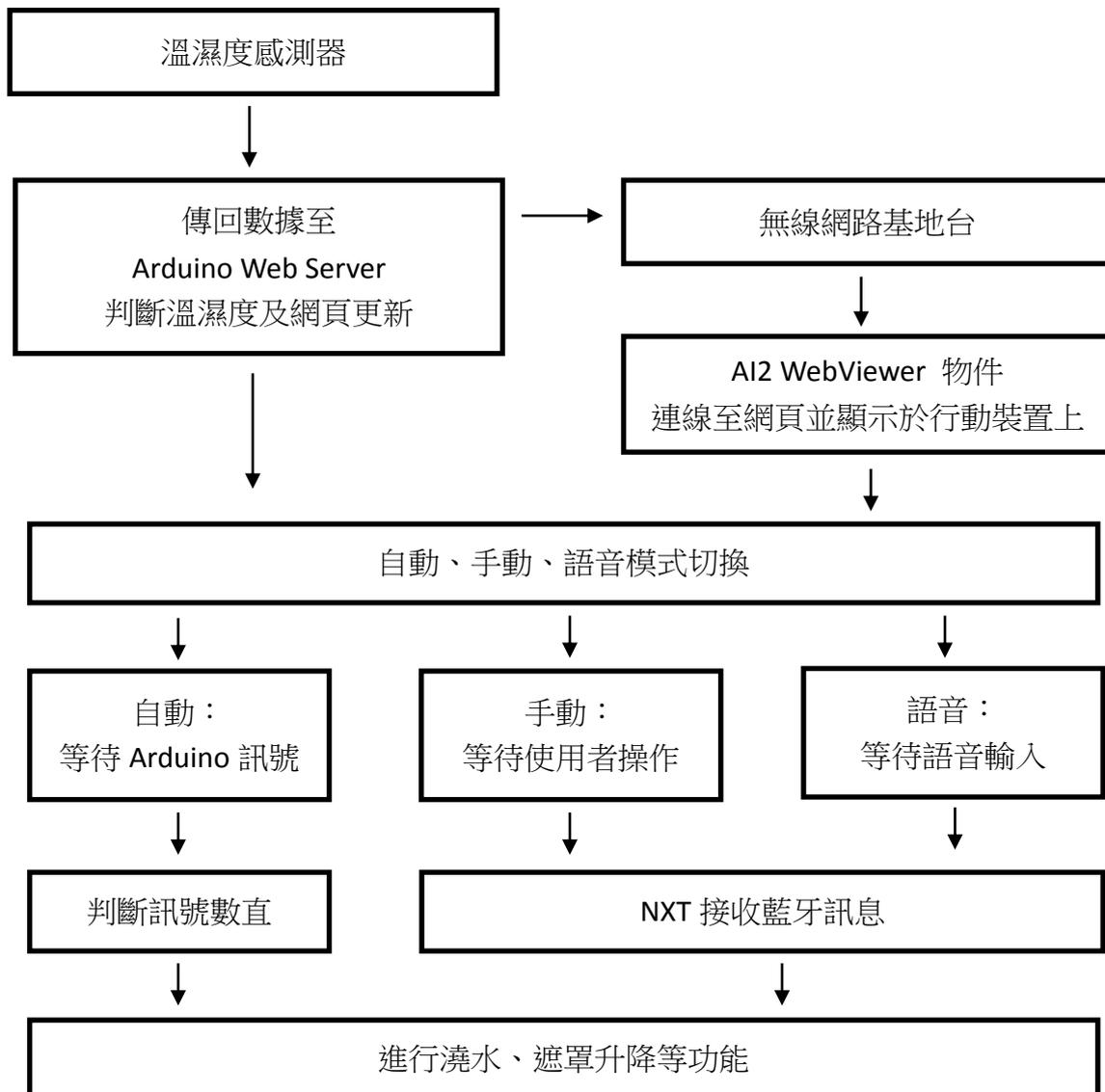


圖 2 系統架構圖

### (一)APP Inventor 控制 LEGO

至今可以編寫 NXT 程式的軟體有很多，而能利用行動裝置來控制軟體中我們選擇了 App Inventor 軟體來控制，主要因為二者皆為麻省理工學院所開發出來的軟體和介面；以圖形化編輯軟體以圖形為主要編輯介面，只要選擇所要的程式圖形並組合起來，NXT 就會依照順序進程式，只要會操縱電腦就會使用，非常簡單且容易上手。

App Inventor 支援樂高機器人控制元件共有 7 種如下圖，且樂高 NXT 支援了 I2C 數位通訊協定，因此大部份感應器控制都使用此協定進行資料傳輸。若有可接於 NXT 主機的 I2C，但 App Inventor 並無對應指令時，可透過 NxtDirectCommands 中的 LS 低速通訊指令來讀取)。而在 App Inventor 中的樂高 MINDSTORMS NXT 相關元件已實作了 Direct Command 通訊指令，不需要編寫任機器人端程式就可以透過 Android 裝置來控制。而因為 LEGO 開放了 NXT 軟體的原始碼所以可以透過位元陣列來直接控制 LEGO NXT 機器人。

表 1 樂高機器人控制元件

元件名稱	說明
NxtDirectCommands	直接控制指令
NxtColorSensor	顏色感應器
NxtLightSensor	光感應器
NxtSoundSensor	聲音感應器
NxtTouchSensor	觸碰感應器
NxtUltrasonicSensor	超音波感應器
NxtDrive	運動及馬達控制指令

由於 Android 在 2.0 以後已完整支援了藍芽功能，而藍芽通訊存在著配對通訊有主動端和被動端的概念，主動端負責發起通訊指令、尋找遠端裝置並建立連線串流，接收端則是等待連線直到開啟串流，因此 Android 行動裝置為主動端，NXT 機器人則是被動的接收端。因為 Android 系統是利用 Dalvik 虛擬機來執行 Java 程式語

言，所以必須在行動裝置(手機、Pad、NXT 主機)安裝 JVM 虛擬平台，以利用 Java I/O 的觀念建立資料串流。

## (二)Arduino 整合周邊

### 1、Arduino

Arduino 是一塊基於開放原始碼的 Simple i/o 介面版，並且具有使用類似 java,C 語言的開發環境。讓您可以快速使用 Arduino 語言與 Flash 或 Processing...等軟體，作出互動作品。Arduino 可以使用開發完成的電子元件例如 Switch 或 sensors 或其他控制器、LED、步進馬達或其他輸出裝置。Arduino 也可以獨立運作成為一個可以跟軟體溝通的介面。

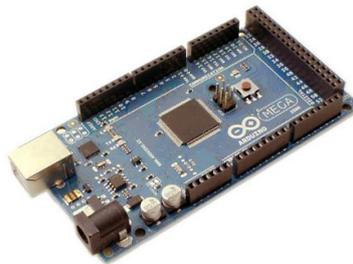


圖 3 Arduino Mega 2560

### 2、DHT11 溫溼度感測器

DHT11 是一款可以同時偵測溫度與濕度的複合感測器，其內部除了電阻式濕度感測元件及 NTC 溫度感測元件外，並與一個八位元單晶片相連接，能將測得的溫濕度資料轉為數位訊號在以 DATA PIN 送出。使用方式很簡單，以 Arduino 為例：將第一隻腳接上高電位，第四隻腳接地並將第二隻腳接上 Arduino 的輸入 PIN，就能將溫濕度資料傳至 Arduino 內在撰寫程式輸出至電腦或自己架設的網站裡，其中資料輸入 PIN 可以在撰寫 Arduino 程式時自行定義。

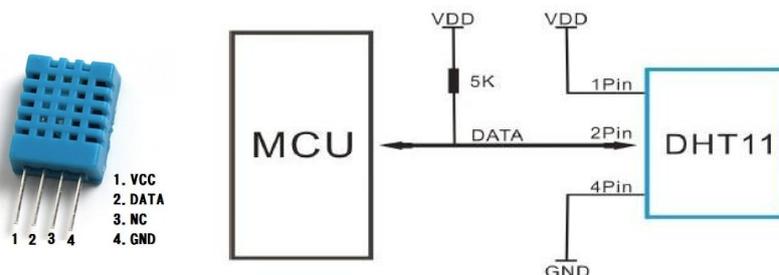


圖 4 DHT11 溫濕度感測器

表 2 DHT11 規格

項目	下限	上限	單位
濕度測量範圍	20	90	相對溼度(%RH)
濕度測量精準度	-5	+5	相對溼度(%RH)
溫度測量範圍	0	50	攝氏溫度(°C)
溫度測量精準度	-2	+2	攝氏溫度(°C)
電源供應範圍	3	5	伏特(V)
頻率	0	0.5	赫茲(Hz)

為了了解 DHT11 的穩定性與精確性是否足夠用於植物上，我們使用兩組 DHT11 接上 Arduino 並透過 Webserver 和無線基地台讓我們能從行動裝置連上溫濕度偵測的網站上觀察。



圖 5 溫濕度觀察

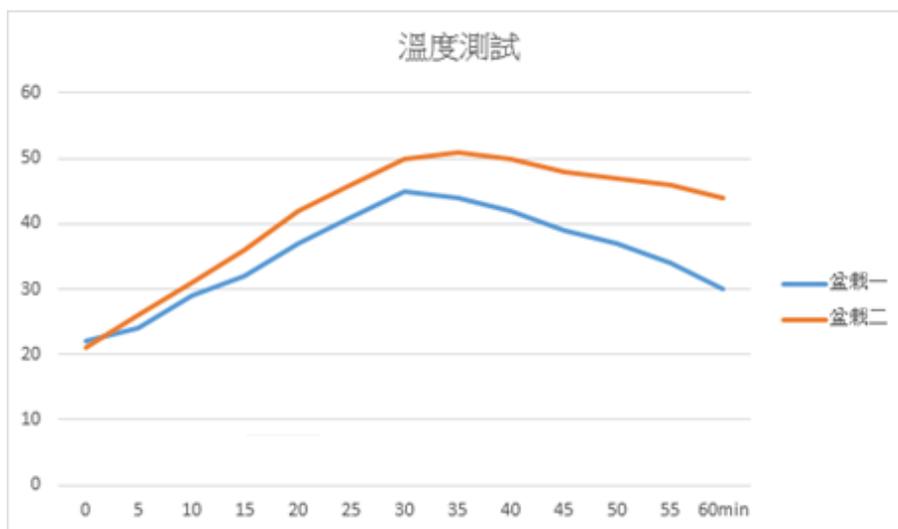


圖 6 溫度測試記錄

溫度測試數據如上圖所示，我們使用吹風機直接對 DHT11 加溫，藉由吹風機跟感測器的距離來控制加溫的幅度。當周圍環境有明顯變化時我們測得的數值也應該要有變化起伏，所以以加溫 30 分冷卻 30 分的方式測試穩定度並記錄所得到的數據。

盆栽一是規格範圍內及生活中可能遇到的數值，比如說夏天正午太陽直接照射到盆栽溫度就有可能超過 30 度的高溫。在盆栽一的曲線可以看到溫度緩緩上升感測器有穩定的持續變化，而 30 分後的冷卻時間內溫度也有明顯的下降趨勢；再來盆栽二我們以貼近感測器的距離加溫，在 20 分時已經測得 53 度的高溫突破規格所定的範圍，雖然到 30 分之前都還在對感測器加溫不過數值到了 51 度變化開始出現異常，數值變動的幅度非常小。

根據上述的觀察得知 DHT11 在正常範圍使用下的反應速度良好，不管是加溫冷卻都有穩定的數值變化；而在不正常的環境下，當突破規格範圍時反應速度緩慢甚至是停滯不動，須等待一段時間才能恢復正常運作，不過對於一般的生活環境來說 50 度已經是綽綽有餘，所以 DHT11 在溫度偵測方面是可行的。

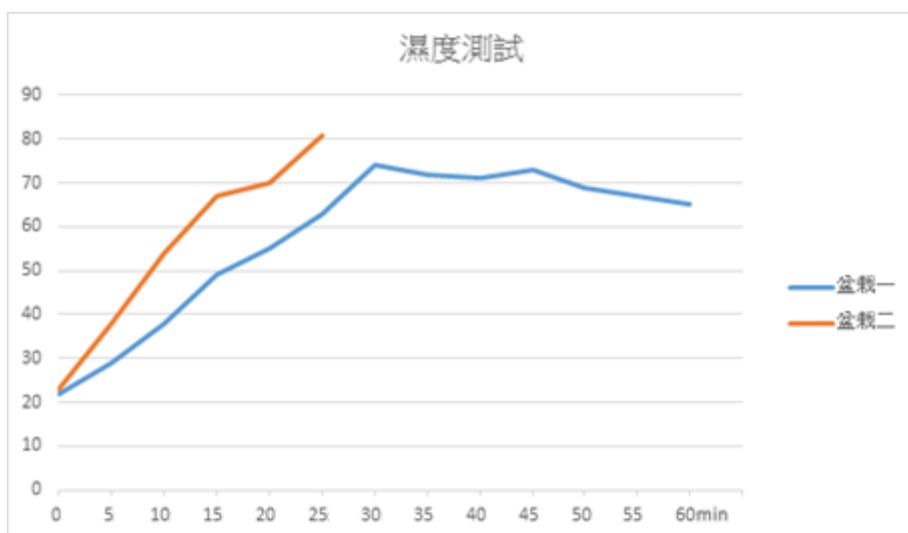


圖 7 濕度測試記錄

測試濕度的方法跟溫度差不多，我們使用噴霧器對 DHT11 噴水，以調整噴霧器集中或分散來控制 DHT11 周圍的濕度影響。

盆栽一在這次測試中一樣是以正常使用的紀錄為數據，以較分散的噴霧慢慢對 DHT11 周圍增加濕度，我們可以透過曲線得知所得的數值有明顯提升，不過停止噴水後下降的趨勢不明顯；再來是盆栽二，以集中的方式將水直接噴在 DHT11 上，濕度快速增加在 25 分的紀錄中測得 81% 的濕度，而之後在網頁上顯示的結果都是無法讀取。

依據上述觀察，DHT11 在濕度方面偵測濕度增加的反應良好，不過濕度下降的反應比溫度還緩慢，這可能是因為周圍濕氣比吹風機加熱的溫度更難散去或是水直接觸碰 DHT11 這種不正常使用所導致的結過。雖然濕度偵測會因為有水直接的碰觸而不穩定，但我們可以藉由改變澆水的量或方式來改善，所以我們認為 DHT11 在濕度偵測這方面也是可行的。

### 3、繼電器

繼電器是一種電子控制元件，當輸入電流達到額定電流後，輸出端便會從斷路狀態轉變成通路狀態，利用此特性即可讓小電流控制大電流。由於 Arduino 最多只能提供五伏特的電源造成負載選擇上的限制，因此我們利用繼電器的特性，由 Arduino 對繼電器輸入端輸出五伏特電源讓輸出通路，這樣就能讓 Arduino 控制更多負載如燈泡或風扇等。

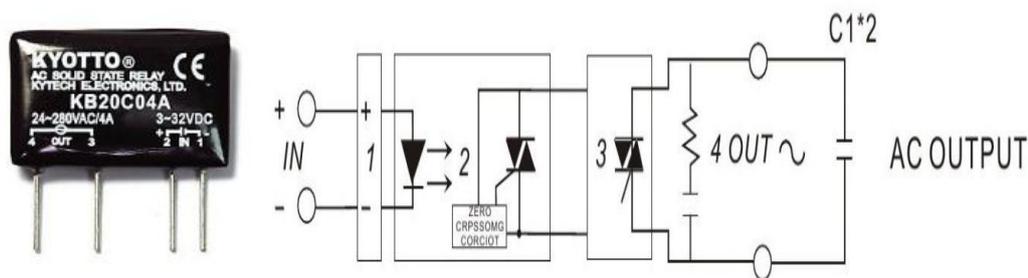


圖 8 KB20C04A 交流型固態繼電器

表 3 KB20C04A 輸出入規格

輸入		
項 目	數 值	單 位
控制電壓範圍	3~32	伏特(V)直流(DC)
最小導通電壓	3	伏特(V)
最小關斷電壓	1	伏特(V)
輸入電流	15	毫安培(mA)
輸出		
項 目	數 值	單 位
負載電壓範圍	24~280	伏特(V)交流(AC)
破壞電壓	600	伏特(V)
最大瞬間承受電流	20	安培(A)

### 三、網路設備

#### (一)WizNet W5100 網路伺服器

Arduino Ethernet 網路擴展模組以 W5100 為核心，可以使 Arduino 成為簡單的 Web 伺服器或通過網路控制讀寫 Arduino 的數位和類比介面等網路應用。Arduino Ethernet 擴充板可以讓 Arduino 控制器連接網路。它是一塊內置 WizNet W5100 TCP/IP 微處理器的擴充板。



圖 9 Arduino W5100

## (二)架設伺服器與網頁

將裝上擴充模組的 Arduino 與無線分享器用乙太網路線直接連接，使用者就可以使用行動裝置連上無線分享器進入 Arduino 架設的網站裡。Arduino 除了能把 DHT11 測得的溫濕度數值放進網站裡外，也能設計選項表單，在提交選項時依照結果對某個腳位送出 Hi 訊號或 Lo 訊號，藉此控制繼電器、NXT 等物件。

以下圖程式為例，以撰寫程式的方式設定 Webserver、建立燈泡開關選單的網頁、控制輸出腳位等功能。在開頭定義 Arduino 的實體網卡與網站位置並設定燈泡的腳位；建立網頁所需的 HTML 字串，包括網頁的開頭、結尾及選單的主體；在自定命令的區塊中撰寫處理網頁內容及控制腳位的功能；最後在參數設置(void setup)跟迴圈(void loop)的區塊中新增執行上述功能與啟動伺服器的指令。

```
sketch_mar28a | Arduino 1.0.3
File Edit Sketch Tools Help

sketch_mar28a $

#include "SPI.h"
#include "Ethernet.h"
#include "WebServer.h"
#include "Streaming.h" // 引用處理字串的程式庫
#include "dht11.h"

static byte mac[] = { 0xF0, 0x7B, 0xCB, 0x4B, 0x7C, 0x9F };
IPAddress ip(192, 168, 0, 25); // 設定首頁位置
IPAddress subnet(255, 255, 255, 0);
IPAddress gateway(192, 168, 0, 1);
WebServer webserver(" ", 80);
// 設定網卡實體與IP位置

const byte LED_PIN = 8; // 定義燈泡腳位
```

圖 10 定義腳位及網卡設定

```
P(htmlHead) =
"<doctype html><html>"
"<head><meta charset='utf-8'>"
"<meta http-equiv='refresh' content='15'>" // 燈泡選單網頁開頭
"<title>LED</title>"
"</head><body>";

P(Head) =
"<doctype html><html>"
"<head><meta charset='utf-8'>"
"<meta http-equiv='refresh' content='15'>" // 燈泡選單結果網頁開頭
"<title>LED_SW</title>"
"</head><body>";

P(htmlFoot) = "</body></html>"; // 網頁結尾

P(LED) =
"<form method='post' action='sw'>"
"燈光 : <input name='light' type='radio' value='ON'> 開 // 燈泡選單網頁
" <input name='light' type='radio' value='OFF' checked= 關
" <br><input type='submit' name='button' val='送出'"
"</form>";
```

圖 11 建立網頁的 HTML 字串

```
void LEDCmd(WebServer &server, WebServer::ConnectionType type, char *, bool)
{
  server.httpSuccess();
  if (type != WebServer::HEAD){
    server.printP(htmlHead);
    server.printP(LED);
    server.printP(htmlFoot);}
} // 顯示燈泡網頁

void postCmd(WebServer &server, WebServer::ConnectionType type, char *, bool)
{
  char name[16];
  char value[16];
  server.httpSuccess();
  if (type == WebServer::POST){
    server.printP(head);
    while (server.readPOSTparam(name, 16, value, 16)){
      if (strcmp(name, "sw")==0){
        server<<"<html><body>";
        if (strcmp(name, "light")==0){
          server << "燈光已開";
          if (strcmp(value, "ON") ==0){
            server << "打開" </p>;
            digitalWrite(LED_PIN, HIGH);}
          else{
            server << "關閉" </p>;
            digitalWrite(LED_PIN, LOW);}}
        server.printP(htmlFoot);}
    } // 燈泡選單處理 選擇一送出訊號至PIN
```

圖 12 處理網頁及控制腳位

```
void setup()
{
  pinMode(LED_PIN, OUTPUT);
  pinMode(FAN_PIN, OUTPUT);
  Ethernet.begin(mac, ip, gateway, subnet); // 啟動乙太網路連線
  webserver.addCommand("sw", &postCmd); // 處理「sw表單處理頁面」請求
  webserver.addCommand("sw1", &FANpostCmd); // 處理「sw表單處理頁面」請求
  webserver.addCommand("form.html", &formCmd); // 處理「表單頁面」請求
  webserver.addCommand("FAN.html", &FANCmd); // 處理「表單頁面」請求
  webserver.setDefaultCommand(&defaultCmd); // 處理「首頁」請求
  webserver.begin(); // 啟用伺服器
}

void loop()
{
  webserver.processConnection(); // 處理使用者連線請求
}
```

圖 13 啟動指令

### (三)網路攝影機

網路攝影機一般都有攝影、畫面捕捉的功能，目前市面上大致分為兩種類型：直接連接電腦用於視訊通話的視訊攝影機與監控專用的監控攝影機，我們把監控用的網路攝影機加入到系統中，讓使用者能隨時觀看盆栽的狀況。將網路攝影機與無線基地台直接用網路線連接，根據發配給攝影機的 IP 位置套入特定格式取得到正確的流串播放網址，透過瀏覽器連線並驅動撥放軟體即可撥放現在的影像畫面。



圖 14 D-Link DCS-1130L 網路攝影機

## 四、LEGO 結構與控制

### (一)NXT

所謂的 NXT，全名是 LEGO Mindstorms NXT，是指由樂高教育事業部門（LEGO Educational Divition）與麻省理工學院於 1998 年共同研發的樂高可程式化機器人套件，其前身是 LEGO Mindstorms RCX（目前已停產）。這套積木結合各種物理、機械原理與電腦程式設計，使組裝出的機器自動完成目標工作。

表 4 LEGO Mindstorms NXT 主要零件及功能

名稱	功能	圖片
NXT 主機	是機器人的大腦，有四個輸入端與三個輸出端，輸入端可以連接各種感應器，輸出則可連接馬達或燈泡。可以使用 USB 或藍牙跟電腦連線。而且有 100x64 像素的 LCD 螢幕，可以顯示各種文字或圖片。	
馬達	馬達是機器人動作的主要零件，由於裡面裝有角度感應器，可以非常精準的控制馬達旋轉的角度，像順時針轉動 30 度，或逆時針轉動 2 圈等等。	
觸碰感應器	主要的功能大致上跟一個按鈕差不多，以二進位來說，按下就是 1，沒按下就是 0，可用於近距離有接觸的感應，像是如果碰到牆壁，便向反方向走之類的用法。	
光源感應器	主要用於感測光的亮度，但無法辨別顏色，有點像光敏電阻。最暗與最亮之間有 100 個值，顏色或亮度越亮，其值就越高。在比賽中，就要依賴這個感應器在指定的路線上行走。	
聲音感應器	感測週圍聲音的大小。單位有 db 和 dbA，可以用聲音大小的變化來控制機器人的行為。	
超音波感應器	長得很像人眼的超音波感應器，顧名思義就是會發出超音波來感測距離。	

## (二)NXT 控制程式

透過介面模組讀取 Arduino 判斷溫濕度的結果訊號，讓 NXT 主機對正確的 Port 輸出電壓讓外接的馬達能夠運轉帶動遮罩、軌道等結構。因為 NXT 的輸出 Port 有數量限制，於是我們根據溫度跟濕度分別控制自動澆水與自動遮罩的功能。

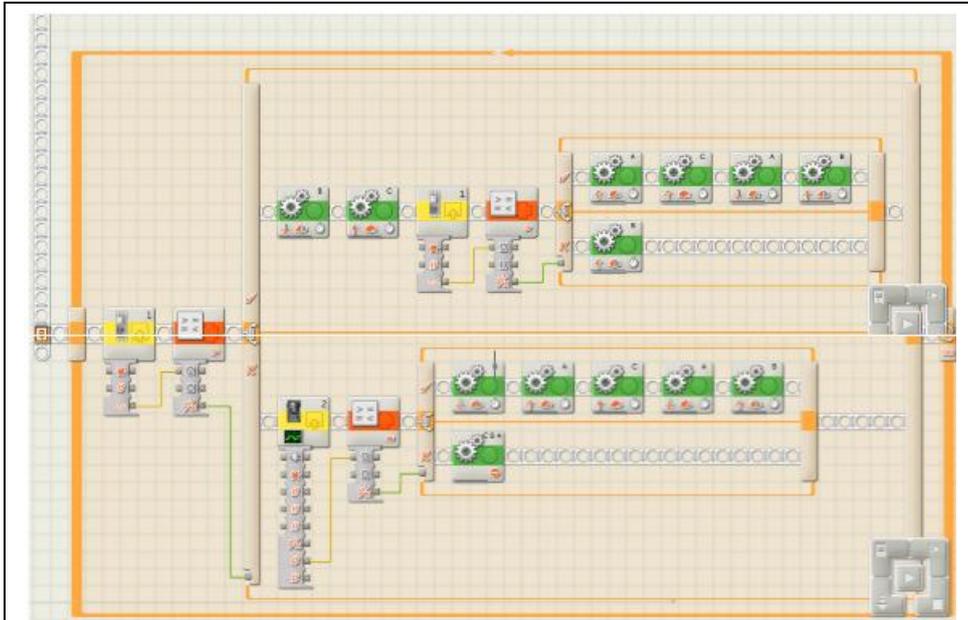


圖 15 自動澆水程式

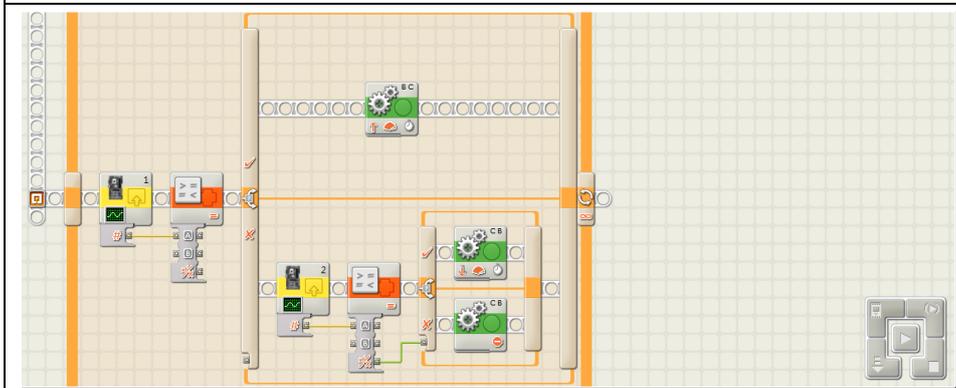


圖 16 自動遮罩程式

## 五、行動裝置操作介面

### (一)APP Inventor

APP Inventor2 是麻省理工學院(MIT)針對 Android 所推出的簡易程式開發工具。利用以圖型積木區塊代替程式碼，提供儲存資訊、重複動作或條件執行等功能，讓開發者

像堆積木一樣，堆疊程式區塊形成新的應用軟體。原名為谷歌應用程序發明家是一款由 GOOGLE 公司開發推出得應用程式開發工具。

而 APP Inventor 在經過了 GOOGLE 的開發到現在回歸到 MIT 它的目的在透過預先建立好程式模組，再讓未具備程式開發專業的一般使用者，能夠以視覺化拖放、堆疊的方式，打造出可在採用 Android 軟體平台的手持行動裝置上執行的應用程式。

## (二)APP 開發

使用 APP Inverter 開發系統的操作介面，利用 UIPickerView 物件對 NXT 進行藍芽連線並以按鈕觸發 Clock 直接控制 NXT 完成基本手動模式的動作；自動模式則是呼叫 NXT 內事先載入的程式開始讀取 Arduino 的訊號開始自動澆水與自動遮罩；而之前 Arduino 建立的網頁是利用 WebView 這個物件讓 APP Inventor 有顯示網頁的功能；最後在設置輸入框跟專用的按鈕讓介面有連線網路攝影機的功能，讓系統更加完善。

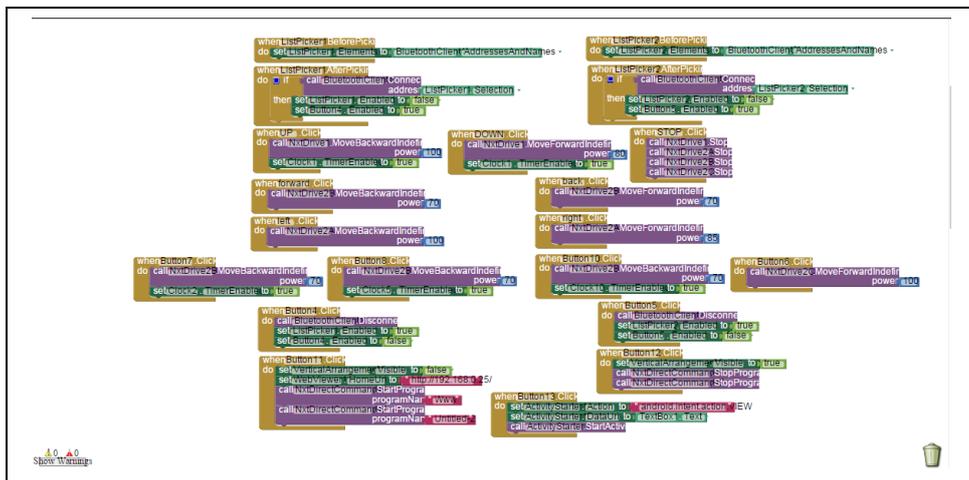


圖 17 介面按鈕控制程式

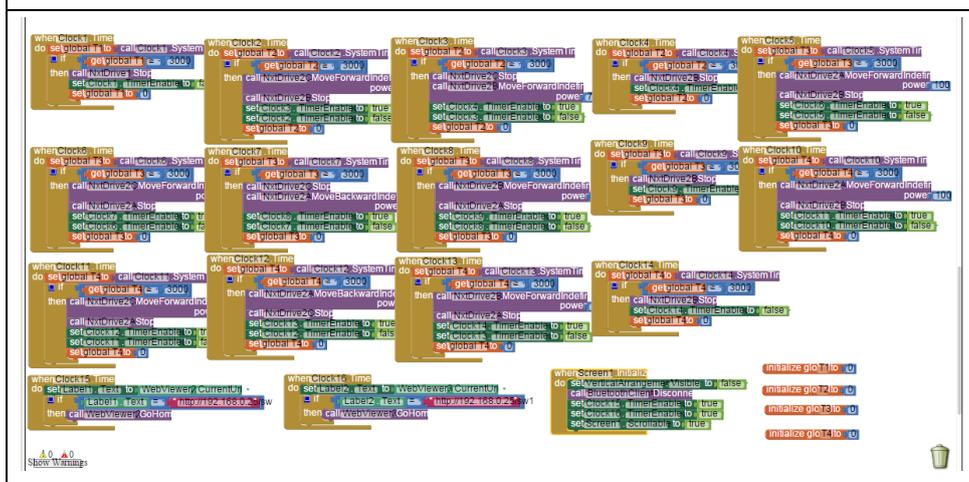
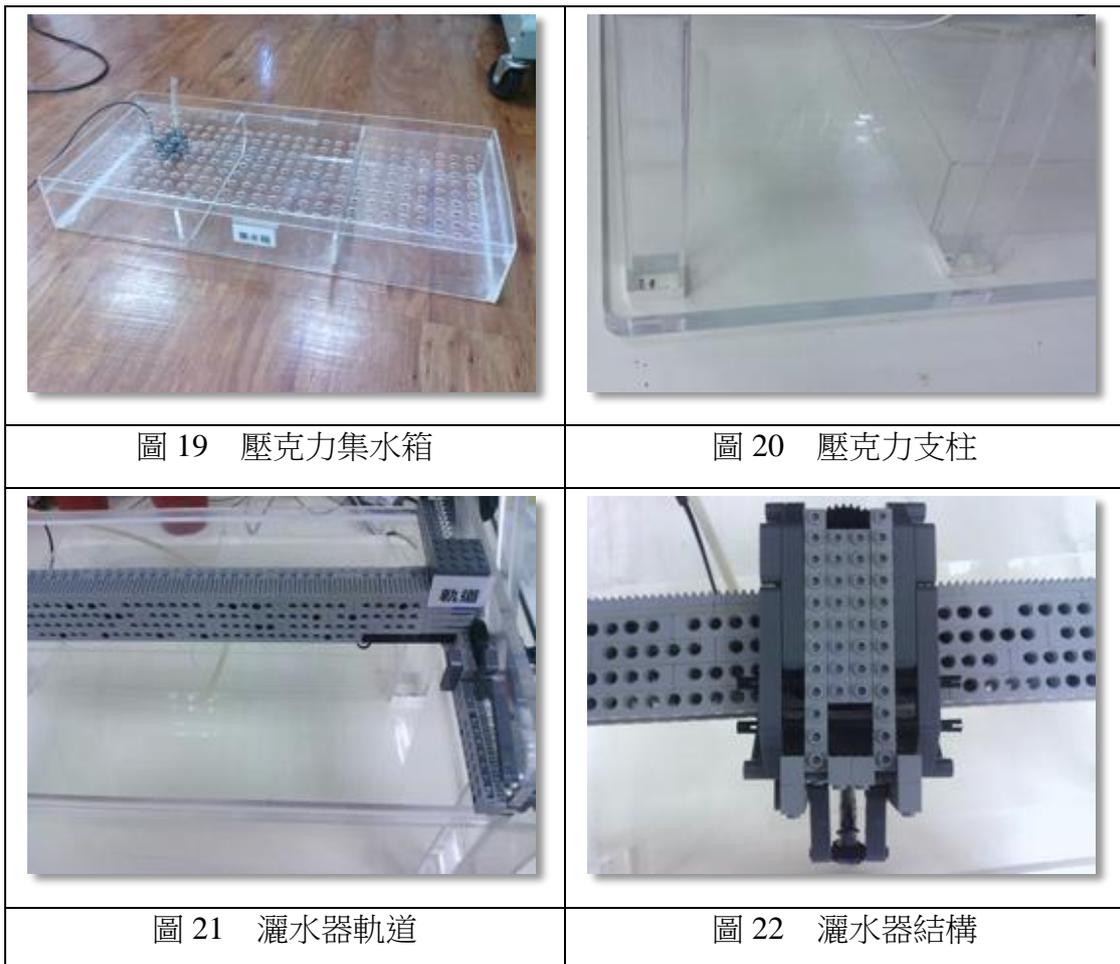


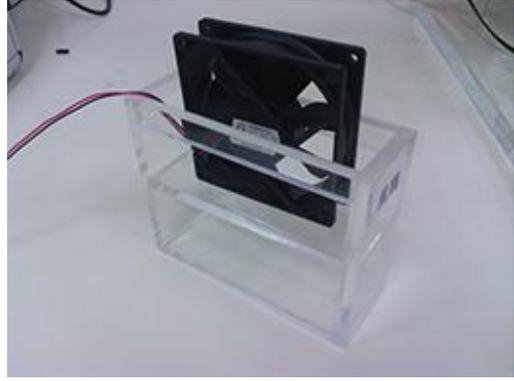
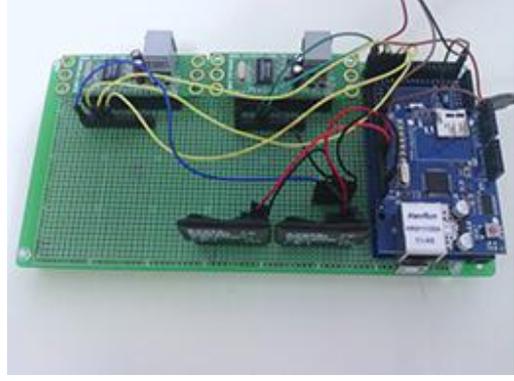
圖 18 Clock 程式

## 六、模型結構

模型的主要結構除了軌道必須搭配馬達組合必須使用樂高外，其餘部分皆以壓克力材質製作，在模型後方還有放置電路整合版、NXT 主機及線路的後台區域。軌道及澆水器的移動模式以天梯式的概念設計，可使澆水裝置在個盆栽之間可以更靈活移動以及大幅增加盆栽的數量，一舉數得。另外在頂端的遮罩，可更靈活應付各種天氣型態。最後在盆栽下方增加一個壓克力集水箱，可收集澆水時多餘的水分，利用抽水馬達收集再次利用，達到資源再利用的環保概念。

另外，將零件分散的 Arduino 電路整合成一個控制板，所有線路、繼電器、NXT 控制板都集中在這裡，在 NXT 控制板與繼電器方面並不只是把元件直接焊接上去而是中間有隔著插孔，如果之後有需要跟換元件的情況更加方便，如此一來就不用為了檢查或整理電路花費許多時間。



	
<p>圖 23 日曬遮罩結構</p>	<p>圖 24 抽水馬達</p>
	
<p>圖 25 燈泡</p>	<p>圖 26 風扇</p>
	
<p>圖 27 模型後台</p>	<p>圖 28 Arduino 整合版</p>

## 肆、討論及應用

### 一、澆水系統

對盆栽而言，澆水可說是一項必要的功能也是照顧植物最重要的一環。由於 LEGO 官方沒有相關套件，其他公司開發的元件的成本過高，因上述幾點因素，決定自製能以 NXT 驅動的抽水馬達。NXT 的每個輸出埠最多能輸出 9 伏特，依照此規格大概猜測推動

用在小魚缸的抽水馬達。因此我們將魚缸的抽水馬達插座剪去，再使用電源供應器測試在多少的電壓下才可驅動馬達，經過測試，大約 8.8 伏特就能驅動此抽水馬達。接下來就是將它接上 NXT 的輸出埠。我們使用 RCX 轉 NXT 的连接線，剛好是 6pin 兩線的，再將它與抽水馬達連接，使用來做錫固定，再用熱縮套與絕緣膠帶將其固定。因天梯式的設計，模型內的盆栽一定會在澆水裝置的活動範圍之內，讓使用者不用為了澆水而移動盆栽使便利性大幅提升。



圖 29 澆水裝置啟動

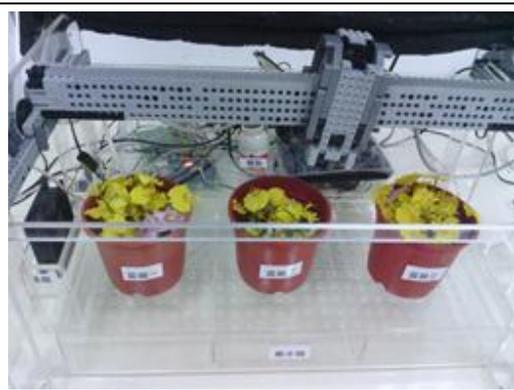


圖 30 澆水裝置天梯式移動

## 二、遮陽系統

在模型的上方設有遮陽功能的透氣黑網，能阻擋陽光避免植物因為過度日照而枯死。切換成手動模式後將可以看到升遮罩與降遮罩的控制按鈕，分別能啟動遮罩與關閉遮罩並且能依照使用者的選擇隨意切換。



圖 31 遮陽系統關閉



圖 32 遮陽系統開啟

### 三、溫室照明系統

在模型後方設有一盞溫室照明燈泡，啟動後能提供植物行光合作用所需的光線，周圍的溫度也會慢慢提升，即使在冬天也能維持適合植物生長的溫度。在控制介面的下方設有溫式照明的表單，當使用者在表單中點選是提交後，會傳送 Hi 訊號至 Arduino 中，在驅動繼電器通路使燈泡亮起；反之點選否提交燈泡將會暗下來。



圖 33 溫室照明關閉

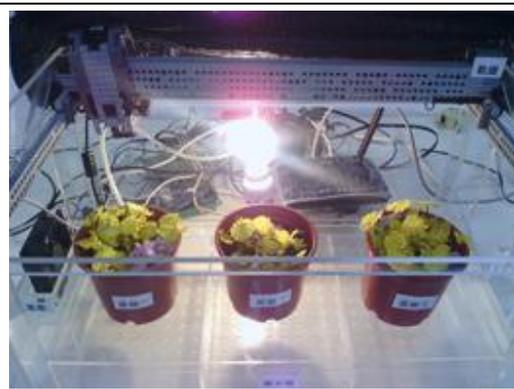


圖 34 溫室照明開啟

### 四、通風循環系統

在模型的兩側設有通風散熱用的風扇，讓盆栽周圍的空氣流通使環境更適合生長，也能解決夏天空氣悶熱的問題。通風設施的工作原理與溫室照明相同，由使用者點選是或否提交後將訊號傳至 Arduino 決定風扇的開關。

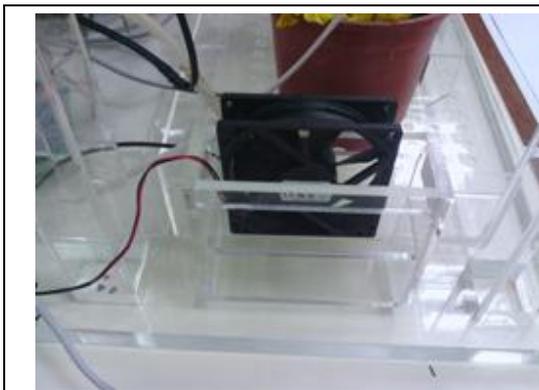


圖 35 通風 裝置關閉

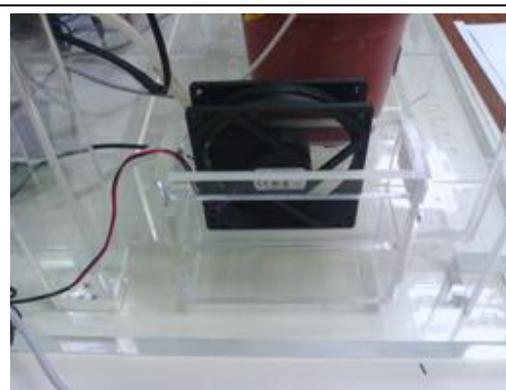


圖 36 通風裝置開啟

### 五、水循環結構

因為澆水裝置使用的水源自於設置在盆栽底下的儲水箱水箱，因此在盆栽與水箱之間有的數個排水孔，讓澆水時撒出盆栽外的水能再次流入儲水箱避免積水造成清潔上的麻煩，

流回儲水箱的水也能透過抽水馬達再次利用，能降低換水的頻率也可以達到綠色環保的目的。



圖 37 水循環



圖 38 排水孔

## 六、雲端系統

行動多功能智慧盆栽系統的模式有兩種，分別為自動澆花與手動澆花，手動澆花就是讓使用者依照上述的各種功能操作系統；而自動澆花模式就是與雲端控制的結合應用。雲端可以遠端操作及接收資料，是利用溫溼度感測器讀取盆栽的溫溼度值再傳回 ArduinoW5100 網路伺服器上，再用 Arduino 撰寫程式自動判斷盆栽的溫度或濕度，當數值到達一定程度便會傳送訊號至控制 I/O 版再轉換成 NXT 可讀取的訊號，再用 NXT 主機內預先撰寫好的程式判斷該進行何種動作，維持植物生長的良好環境。



圖 39 自動模式介面



圖 40 手動模式介面

## 七、溫濕度控制系統

透過盆栽上的感測器取得溫濕度的數值傳回網路伺服器後，事先在 Arduino 裡撰寫好的程式將會隨溫濕度變化選擇正確的動作來對應，將溫度與濕度控制在適當的範圍中。在感測器取得的濕度低於 70% 將會啟動自動澆水；溫度超過攝氏 30 度時遮罩將會自動升起，讓使用者在外出時也能讓植物有良好的生長環境。



### (一) 溫度控制

當使用者人不在家中，時間又剛好在太陽最大的中午容易造成植物枯死，但藉由我們的溫度控制就能防止此情況發生。如下圖所示，我們使用吹風機持續對著感測器加溫模擬接近正午時溫度開始升高的環境，在 10 分與 15 分之間數值就超過了程式設定的條件，也就自動遮罩起動。雖然到 15 分後仍然繼續升高但之後的數值開始下降，代表遮罩確實有隔絕熱的功能。到了 35 分後趨於穩定，所得的數據一直在 25 度與 28 度之間徘徊，讓溫度不會繼續上升。

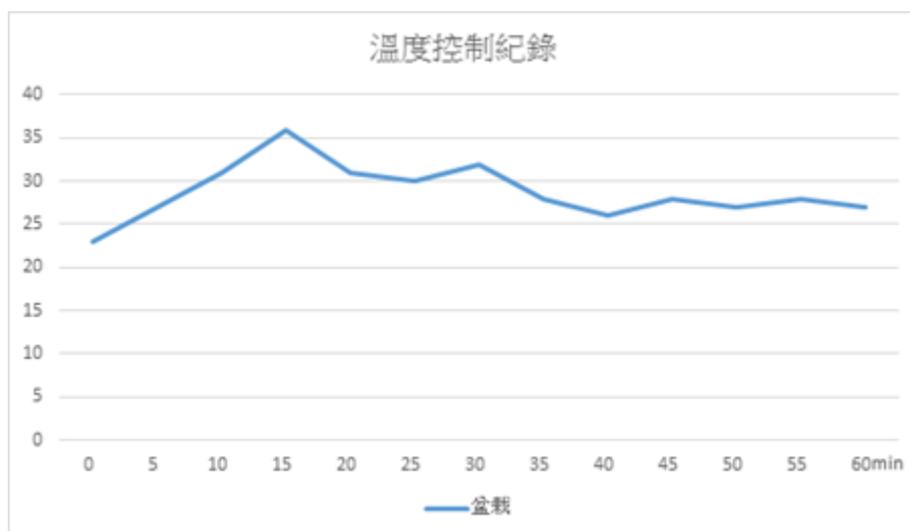


圖 43 溫度控制記錄

## (二) 濕度控制

與溫度控制相同，天氣乾燥、炎熱的天氣也會造成濕度快速下降產生不利於植物生長的环境，我們把系統的自動模式開啟後讓盆栽靜置觀察長期的濕度變化。如下圖所示，因為濕度的短期變化較不明顯，所以以每隔一小時的方式觀察濕度值，一開始時感測器剛起動數值較低，但經過第一次自動澆水後濕度就維持在 60% 到 80% 之間，並不會讓乾燥的环境發生，有利維持良好的植物生長環境。

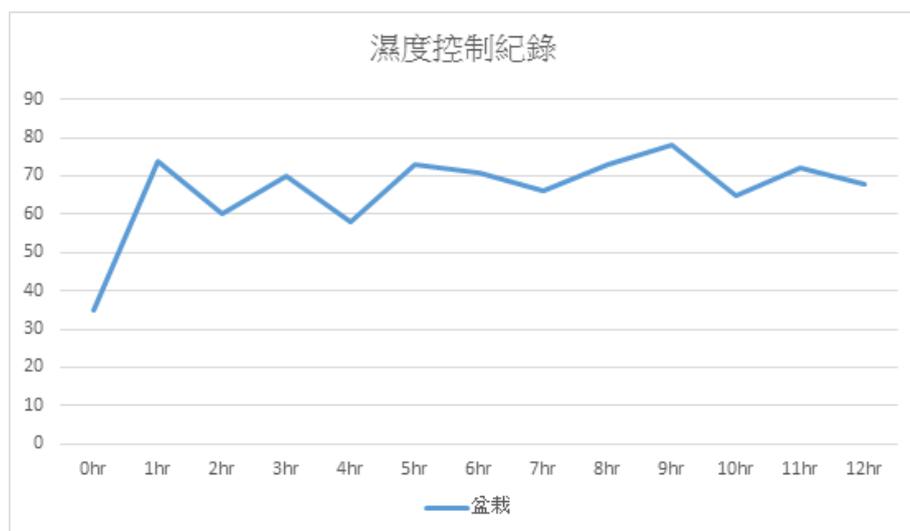


圖 44 濕度控制紀錄

## 八、網路監控系統

將網路攝影機與無線 AP 接上並架設在盆栽旁，在程式下方的輸入框裡輸入正確的網址，按下「即時影像」的按鈕後會呼叫行動裝置的播放軟體觀看家中盆栽的實際影像。



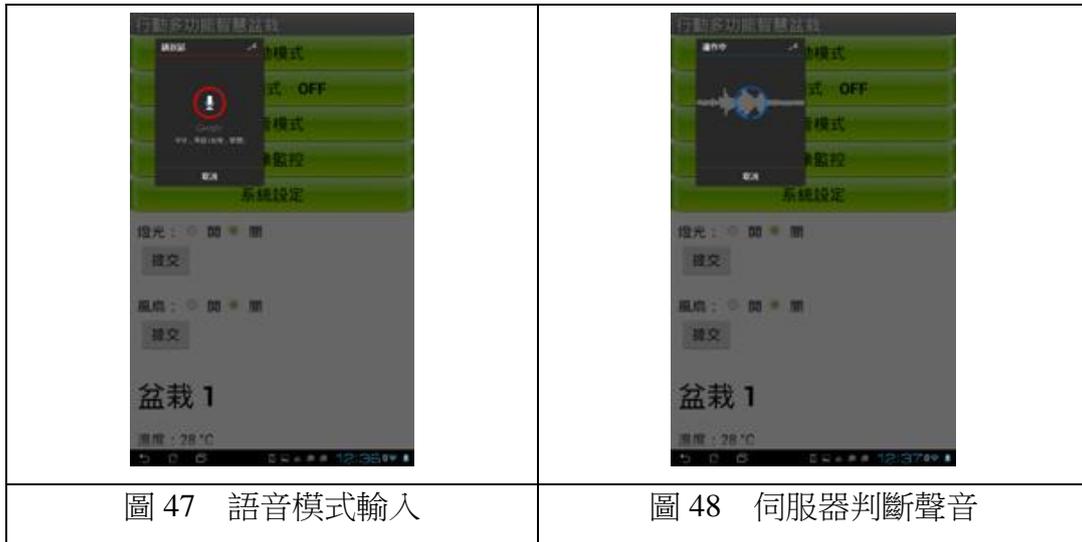
圖 45 IP 輸入與影像按鈕



圖 46 即時影像

## 九、語音操作

利用 AI2 提供的聲音轉文字程式方塊，這個功能能透過自動呼叫 Google 語音將使用者的聲音傳至伺服器作處理再將文字資料傳回來，後續再將這個文字資料以多個條件逐漸比對出最相近的關鍵字進行動作的執行。



## 十、問題討論

- 一、由於 DHT11 溫濕度感測器高於某個數值就會使 Arduino 得到的數據異常，間接影響提供給使用者的資料準確度跟自動模式的功能。但是實際測試後，DHT11 的溫度極限其實可以高於規格的 50°C，且因為實際上盆栽可能測得的數值不會這麼高，所以並不會因為感測器使系統異常。
- 二、我們在手動模式中有定位澆水的功能，但如果遇上地震之類的非人為因素造成盆栽移位使定位不正確，造成澆水器沒有對準盆栽。我們可以藉由修改下方集水箱的結構使其能夠固定盆栽避免此問題發生。
- 三、APP Inventor、NXT、Arduino 之間都有支援的程式碼或介面模組使用，並不用擔心三者會因為相容性的問題造成執行動作不正確。

## 伍、結論

### 一、總結

經過重重的討論，最後討論出製作以雲端系統控制自動澆水的系統，用了大量時間的修改及設計，也利用到了實習課所學到電子學和 APP 程式設計的東西，並用樂高做出灑水器與遮陽棚，最主要用了與以往不同的操作、製造技術、網路技術及藍牙技術，像是利用 App Inventor2 控制此機體來完成所有的動作以及使用 Arduino 當作溫溼度感測器的伺服器回傳當下的數值到系統上，並配合 NXT 主機做灑水的裝置，有了這次的研究與開發，也希望利用將來的技術能使此系統更加便利。

### 二、作品實用性

將來運用此系統能使照顧盆栽更加便利，以生活忙碌的族群為例，就算沒辦法每天替盆栽澆水也能利用自動化的功能幫使用者完成，對一般人來說也可以藉由網頁上的溫濕度數據更準確的判斷該不該澆水或是升降遮罩使照顧盆栽變得更加順手。除此之外系統還有風扇與燈泡設備來對應春夏秋冬可能出現的問題；語音操作與影像監控的功能則能滿足各種使用者的需求，讓系統更加完善。放置盆栽的模型也有針對水資源的問題加以設計，集水箱的位置就在盆栽正下方，澆水時多餘的水會透過特製的壓克力板在流回集水箱以抽水馬達進行循環達到資源的永續利用。

## 陸、參考資料及其他

- 1.CAVE 教育團隊(2012)。Android 手機應用程式超簡單!! (入門卷)。台北市。馥林文化股份有限公司。
- 2.CAVE 教育團隊(2012)。Android 手機應用程式超簡單!! (機器人卷)。台北市。馥林文化股份有限公司。
- 3.John Baichtal 著莊啟晃,黃藤毅,莊雯,林可凡譯。Make：樂高機器人與 Arduino 專題製作。台北市。碁峰資訊股份有限公司。
- 4.王培坤(2013)。MIT App Inventor 開發手冊 for Android。台北市。上奇資訊股份公司。
- 5.文淵閣工作室(2014)。手機應用程式設計超簡單：App Inventor 專題特訓班。台北市。碁

峰資訊股份有限公司。

6.文淵閣工作室(2014)。手機應用程式設計超簡單：App Inventor 初學特訓班。台北市。碁峰資訊股份有限公司。

7.如宏齋(2010)。NXT 機器人之介紹。痞客邦。2014 年 9 月 20 日，取自

<http://chunyou0830.pixnet.net/blog/post/61870649-%E3%80%90%E6%A8%82%E9%AB%98%E3%80%91nxt%E6%A9%9F%E5%99%A8%E4%BA%BA%E4%B9%8B%E5%9F%BA%E6%9C%AC%E4%BB%8B%E7%B4%B9>

8.邱信仁(2011)。NXT 樂高機器人：創意樂趣，隨心所欲！（第二版）。新北市。藍海文化事業股份有限公司。

9.柯博文(2014)。Arduino 互動設計專題與實戰。台北市。碁峰資訊股份有限公司。

10 蕭季萱，植物自我照護系統，民國 103 年 2 月 19 號，取自

<http://activity.ntsec.gov.tw/activity/race-1/50/pdf/080819.pdf>

11.趙英傑(民 102)。超圖解 Arduino 互動設計入門。旗標出版社。