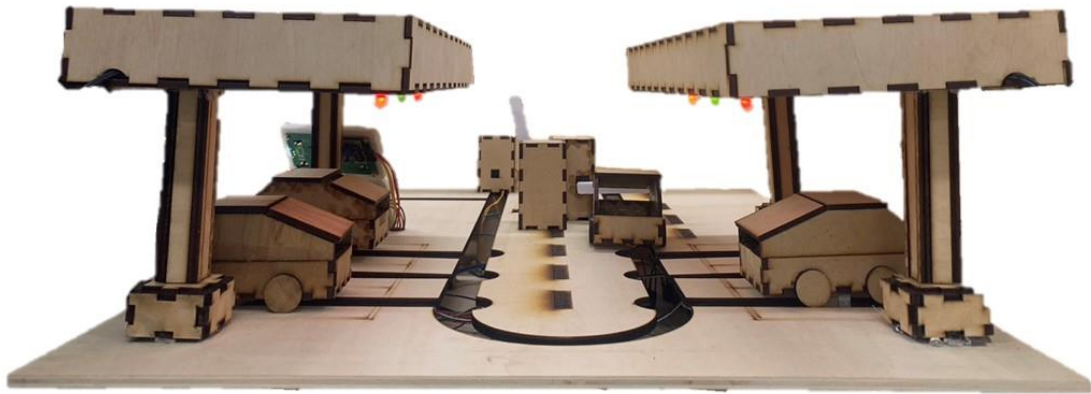


臺北市立大安高級工業職業學校專題製作競賽
「專題組」作品說明書



群別：電機與電子群

作品名稱：預停好好用

關鍵詞：預約停車 網頁伺服器
Optical Character Recognition (OCR)

目錄

| | |
|---------------------------|----|
| 壹、摘要 | 1 |
| 貳、研究動機 | 1 |
| 參、主題與課程之相關性或教學單元之說明 | 2 |
| 一、硬體製作 | 2 |
| 二、程式撰寫 | 2 |
| (一)、車牌辨識 | 2 |
| (二)、預約車位 | 3 |
| (三)、燈號狀態以及剩餘車位 | 3 |
| 三、成品外觀 | 4 |
| 肆、研究方法 | 4 |
| 一、研究流程 | 4 |
| (一)、研究步驟 | 4 |
| (二)、操作步驟 | 4 |
| 二、使用材料及工具 | 7 |
| (一)、零件介紹 | 7 |
| (二)、機構原理 | 11 |
| (三)、軟體介紹 | 11 |
| 伍、研究結果 | 15 |
| 一、硬體結構 | 15 |
| (一)、柵欄桿 | 15 |
| (二)、LCD 框架 | 16 |
| 二、軟體通訊架構 | 17 |
| (一)HTTP 伺服器 | 17 |
| (二)、Serial(UAR) | 17 |
| 三、成果展示 | 17 |
| (一)、APP 介面 | 17 |
| (二)、車牌辨識 | 18 |
| (三)、燈號更改和剩餘車位 | 19 |
| 陸、討論 | 19 |
| 一、設備性能不足 | 19 |
| 二、光敏電阻數值太敏感 | 19 |
| 三、使用極限開關和光敏電阻 | 19 |
| 柒、結論 | 20 |
| 捌、參考資料及其他 | 21 |
| 一、網路資料 | 21 |

表目錄

| | | |
|------|--------------------|----|
| 表 1 | 時間分配表..... | 5 |
| 表 2 | Mega2560 規格 | 8 |
| 表 3 | ESP32-CAM 規格 | 8 |
| 表 4 | SS-5GL 規格..... | 8 |
| 表 5 | Mg90s 伺服馬達規格 | 9 |
| 表 6 | 雙色 LED 燈規格..... | 9 |
| 表 7 | 光敏電阻規格..... | 9 |
| 表 8 | 白光 LED 燈條規格..... | 10 |
| 表 9 | UNO 規格..... | 10 |
| 表 10 | CH340G 規格 | 10 |

圖目錄

| | | |
|------|---------------------------------|----|
| 圖 1 | Arduino Esp32cam 程式撰寫..... | 2 |
| 圖 2 | Python 車牌辨識程式撰寫..... | 2 |
| 圖 3 | Arduino LCD 螢幕程式撰寫..... | 2 |
| 圖 4 | App inventor 程式撰寫..... | 3 |
| 圖 5 | Visual Studio HTTP 伺服器..... | 3 |
| 圖 6 | Arduino 接收 Python 程式撰寫..... | 3 |
| 圖 7 | Arduino 剩餘車位和讀取光敏電阻數值程式撰寫..... | 4 |
| 圖 8 | 停車場成品正視圖..... | 4 |
| 圖 9 | 停車場成品側視圖..... | 4 |
| 圖 10 | 車牌辨識流程圖..... | 5 |
| 圖 11 | Mega2560..... | 6 |
| 圖 12 | 預約車位流程圖..... | 7 |
| 圖 13 | Mega2560..... | 8 |
| 圖 14 | ESP32-CAM..... | 8 |
| 圖 15 | SS-5GL..... | 8 |
| 圖 16 | Mg90s 伺服馬達..... | 9 |
| 圖 17 | 雙色 LED 燈..... | 9 |
| 圖 18 | 光敏電阻..... | 9 |
| 圖 19 | 白光 LED 燈條..... | 10 |
| 圖 20 | UNO..... | 10 |
| 圖 21 | CH340G..... | 10 |
| 圖 22 | Autodesk inventor logo..... | 11 |
| 圖 23 | Autodesk inventor 3D 圖繪製介面..... | 11 |
| 圖 24 | Arduino logo..... | 12 |
| 圖 25 | Arduino 程式撰寫..... | 12 |
| 圖 26 | HTTP logo..... | 12 |
| 圖 27 | App Inventor logo..... | 13 |
| 圖 28 | Fritzing logo..... | 13 |
| 圖 29 | 車牌辨識接線圖..... | 13 |
| 圖 30 | 燈號和剩餘車位接線圖..... | 14 |
| 圖 31 | Vscode..... | 14 |
| 圖 32 | Jupyter..... | 15 |
| 圖 33 | Autodesk Inventor 3D 結構圖..... | 15 |
| 圖 34 | 柵欄桿..... | 16 |
| 圖 35 | LCD 框架..... | 16 |
| 圖 36 | 車牌辨識的通訊圖..... | 17 |
| 圖 37 | 預約車位的通訊圖..... | 17 |
| 圖 38 | APP 主頁..... | 18 |
| 圖 39 | APP 預約畫面..... | 18 |
| 圖 40 | APP 完成預約..... | 18 |
| 圖 41 | esp32cam 網頁..... | 18 |
| 圖 42 | LCD 顯示車牌..... | 18 |

| | | |
|------|-------------|----|
| 圖 43 | 三種燈號顯示..... | 19 |
| 圖 44 | 剩餘車位顯示..... | 19 |

【預停好好用】

壹、摘要

為了讓大能在停車時減少等待的時間並且更便利，我們的專題設計了一個停車場的預約系統，讓使用者能夠輕鬆的使用 App 事先預約自己需要的車位並且選擇想要預約的時間。

硬體的部分我們針對真實停車場做模擬並規劃了六個停車格，我們運用了光敏電阻來偵測是否有車輛停入，如果有車輛停入，他會觸發雙色 led 燈變色，而雙色 led 在一般狀況下會顯示綠色，有車輛停入會顯示紅色，有人預約則是顯示無色。

軟體的部分分成兩個:預約系統及車牌辨識。車牌辨識的部分，運用了 Optical Character Recognition (OCR) 的原理去讀取車牌。在預約系統的部分，我們架設了網頁伺服器、並撰寫 APP 進行車位預約，透過 APP 傳送指令給伺服器，伺服器接受到指令後再使用 UART 操控 Mega2560 使停車場狀態改變。

貳、研究動機

相信大家日常生活中一定常常會有找不到車位的問題，如果今天是一家人出遊而花了很多時間在尋找車位，是否就浪費了很多玩樂的時間？尤其是在疫情趨緩的當下，大家一定更常出門旅遊，在車位有限的情況下，找不到車位的機率又大幅地提升了。抑或是一大早要到某些地方，但到了目的地後卻找不到車位，相信一整天的好心情也會就此受到影響。

除此之外，我們也常常在新聞上看到許多人為了要爭奪同個車位而發生衝突，無論是因為一個好不容易找到的車位卻有人正擋在前方占位又或者是一個等了很久的車位卻被別人搶先一步等等。最重要的是在我們尋找車位的同時油錢正隨著時間一點一滴的流逝。

相信大家不管是去餐廳，看醫生，或者是出去玩所住的飯店都會先做預約的動作，那為什麼要事先預約呢？正因為預約能夠減少等待的時間，那我們為什麼不把預約系統運用在停車場呢？這樣不但能夠減少大家找車位的時間也能夠減少以上狀況的發生，更重要的是油錢也不會浪費在不必要的地方。因此，「預停好好用」就此誕生了。

參、主題與課程之相關性或教學單元之說明

一、硬體製作

停車場主體是使用 5mm 而車子是利用 3mm 的木板製成，利用雷射切割機切割出我們需要的大小並且進行組裝，而停車場柵欄桿及 LCD 螢幕的蓋板則是利用 3D 列印來製作。

二、程式撰寫

(一)、車牌辨識

我們使用 Arduino IDE、Python 撰寫控制程式，使用 ESP32-CAM 進行拍照並上圖片傳到網頁上(如圖 1)，再藉由 Python 使用 HTTP 負責抓下圖片並利用 OCR 將圖片上的文字讀取出來(如圖 2)，再將讀取出來的數值傳給 Arduino Uno 板(如圖 3)，讓上方 LCD 螢幕顯示。

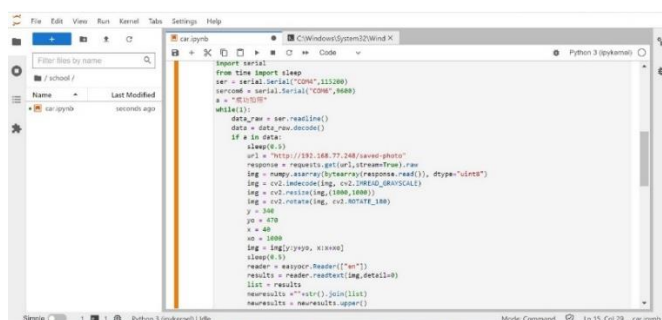


```
camera[Arduino 1.8.13]
#include <WebServer.h>
#include <ESP32-CAM.h>
WebServer server(80);
ESP32CAM cam;

void setup() {
  server.begin();
  cam.begin();
}

void loop() {
  if (server.hasRequest()) {
    server.send(200, "text/html", "Hello!");
  }
}
```

圖 1 Arduino Esp32cam 程式撰寫

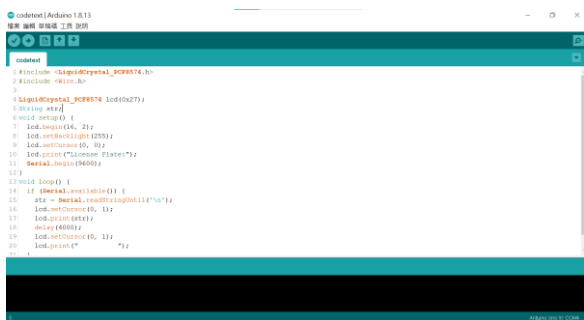


```
import cv2
import numpy as np
import pytesseract
import re
import requests
import time

url = "http://192.168.77.248/receive_photo"
ser = serial.Serial("COM4", 115200)
ser.write("start")

while True:
    data_ser = ser.readline()
    data = data_ser.decode()
    if data:
        sleep(0.5)
        url = "http://192.168.77.248/receive_photo"
        response = requests.get(url, stream=True)
        img = numpy.asarray(bytearray(response.read()), dtype="uint8")
        img = cv2.imdecode(img, cv2.IMREAD_COLOR)
        img = cv2.resize(img, (1000, 1000))
        img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        x = 40
        y = 340
        y2 = 400
        x2 = 480
        img = img[y:y2, x:x2]
        reader = pytesseract.from_image(img, "eng")
        results = reader.get_text()
        list = results
        newresults = ""
        newresults = newresults.upper()
```

圖 2 Python 車牌辨識程式撰寫



```
code[Arduino 1.8.13]
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup() {
  lcd.begin(16, 2);
  pinMode(14, OUTPUT);
  digitalWrite(14, HIGH);
  Serial.begin(9600);
}

void loop() {
  if (Serial.available()) {
    int data = Serial.read();
    lcd.setCursor(0, 1);
    lcd.print(data);
    delay(4000);
    lcd.setCursor(0, 1);
    lcd.print("");
  }
}
```

圖 3 Arduino LCD 螢幕程式撰寫

(二)、預約車位

我們使用 App inventor 撰寫 App(如圖 4)，當你使用 App 預約車位後，App 會傳送指令給伺服器(如圖 5)，伺服器再透過 UART 控制停車場的主控板來改變停車場這狀態(如圖 6)。

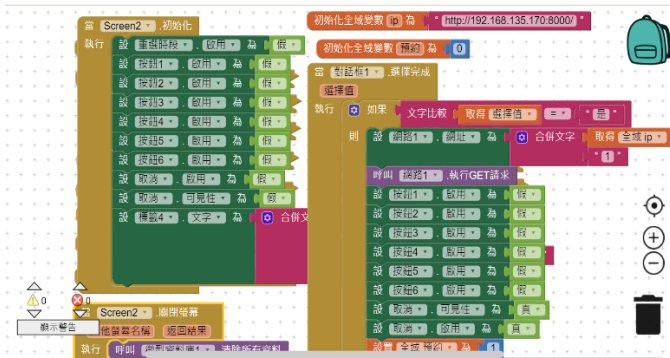


圖 4 App inventor 程式撰寫

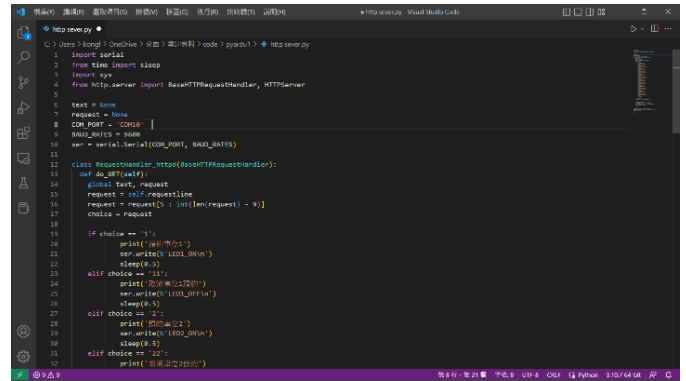


圖 5 Visual Studio HTTP 伺服器

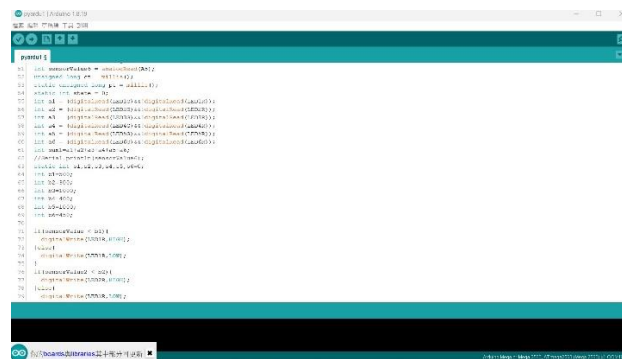


圖 6 Arduino 接收 Python 程式撰寫

(三)、燈號狀態以及剩餘車位

我們使用 Arduino IDE 撰寫控制程式，當 Mega2560 接收到光敏電阻數值低於設定值，下方的 LCD 螢幕會減少剩餘的車位，當有人預約或車位已經被使用時，剩餘車位會減少。當車位沒有被使用時車位上方的 LED 燈會亮綠色，有人預約時 LED 燈會變成無色，當車位被使用時 LED 燈會變成紅色(如圖 7)。

表 1 時間分配表

| | 7月 | 8月 | 9月 | 10月 | 11月 | 12月 | 1月 |
|--------|----|----|----|-----|-----|-----|----|
| 1.資料蒐集 | | | | | | | |
| 2.元件採購 | | | | | | | |
| 3.外殼設計 | | | | | | | |
| 4.程式撰寫 | | | | | | | |
| 5.通訊連接 | | | | | | | |
| 6.電路製作 | | | | | | | |
| 7.成品測試 | | | | | | | |

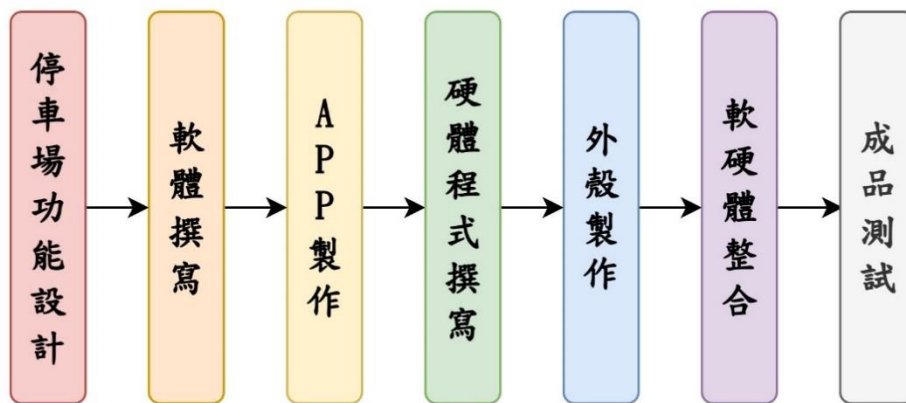


圖 10 研究步驟

(二)、操作步驟

1. 車牌辨識

當極限開關觸發後，ESP32-CAM 動作擷取圖片，接著 Python 將抓取圖片並讀出車牌號碼，再傳到 Arduino Uno 板，讓 LCD 螢幕顯示，最後柵欄桿升起在下降。(如圖 11)。

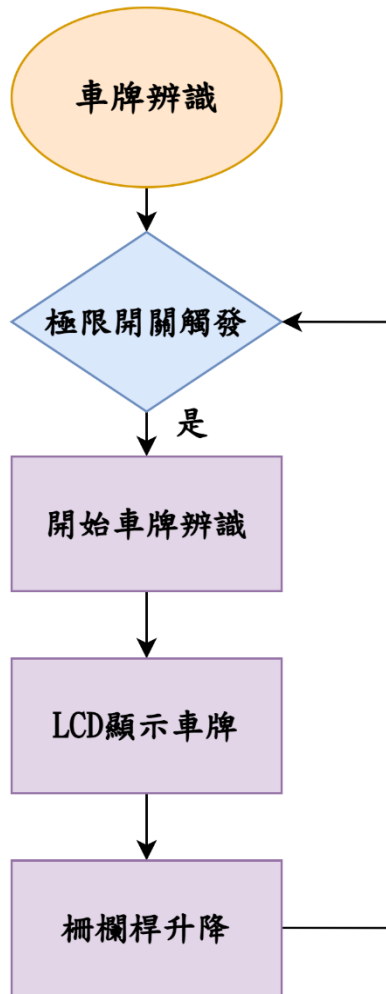


圖 11 車牌辨識流程圖

2. 預約車位

使用者輸入自己的車牌後選擇自己想要的時段和要預約的車位，如果隨時不要停車可以取消預約(如圖 12)。

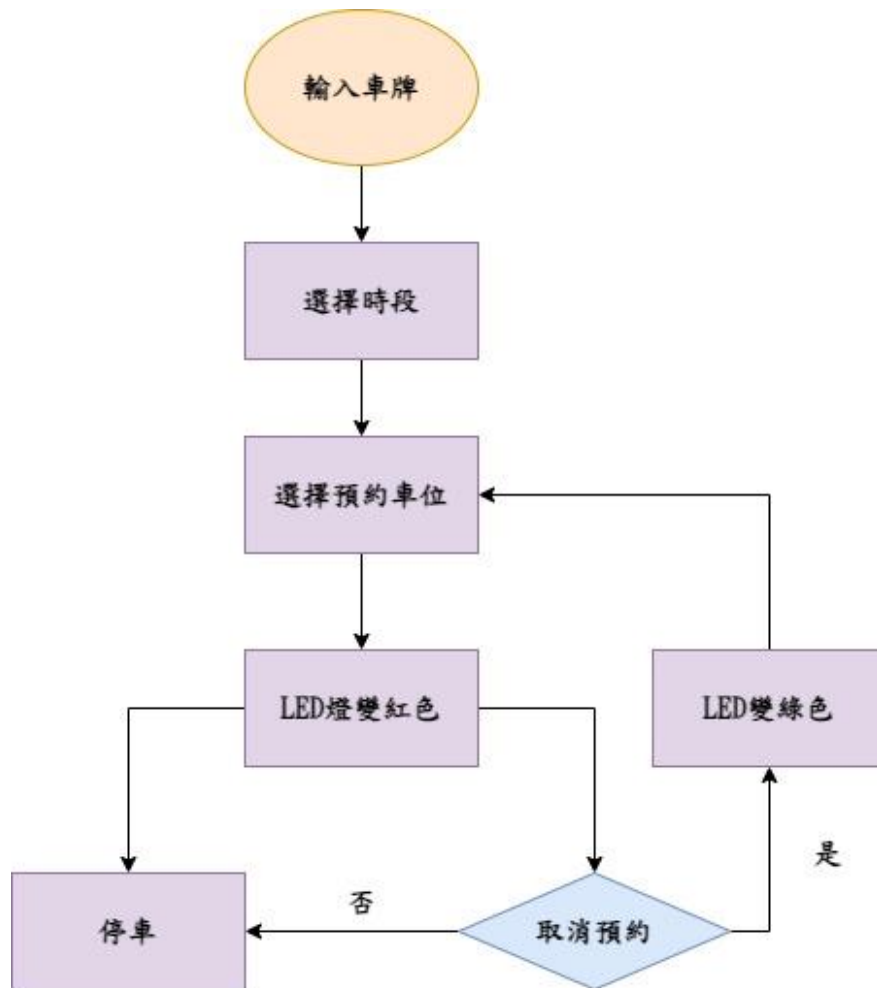


圖 12 預約車位流程圖

二、使用材料及工具

(一)、零件介紹

1. Mega2560

Arduino Mega2560 是一款功能強大的開發板，擁有 54 個數位輸出/輸入接腳（其中 15 隻支援 PWM 輸出），和 16 個類比接腳，以及 256KB 的閃存，幾乎是 Arduino UNO 的四倍，其規格、外型(如表 2、圖 13)。我們使用 Mega2560 當作控制停車場燈號狀態以及剩餘車位。

表 2 Mega2560 規格

| | |
|---------|-------------|
| 核心處理器 | Mega2560 |
| I/O 數 | 54 |
| 程式記憶體大小 | 256KB |
| 程式記憶體類型 | 閃存 |
| 電壓 - 電源 | DC 7V ~ 12V |
| 時脈速度 | 16 Hz |

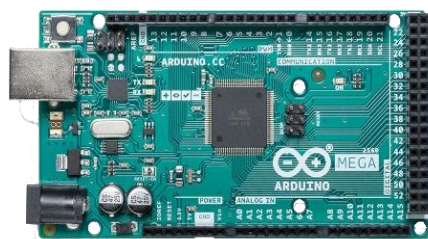


圖 13 Mega2560

2. ESP32-CAM

ESP32-CAM 是安信可科技研發的 ESP32 開發板，搭載 200 萬像素的攝影鏡頭，具備拍照、串流視訊以及人臉辨識功能，其規格、外型(如表 3、圖 14)。我們使用 ESP32-CAM 來進行拍攝車牌的圖片，以及操控柵欄桿升降。

表 3 ESP32-CAM 規格

| | |
|-------|---------------|
| 產品尺寸 | 27*40.5*4.5mm |
| CPU | ESP32 |
| ROM | 520KByte |
| 通用 IO | 9 |
| 供電電壓 | 5V |
| 額定電流 | 2A |



圖 14 ESP32-CAM

3. 極限開關 SS-5GL

我們使用極限開關進行車牌辨識開始的條件，當極限開關被壓下，就開始車牌辨識，其規格、外型(如表 4、圖 15)。

表 4 SS-5GL 規格

| | |
|------|--------------|
| 額定電流 | 5A |
| 額定電壓 | 125 V |
| 工作溫度 | -25°C ~ 85°C |



圖 15 SS-5GL

4. MG90S 伺服馬達

MG90S 是一種小型的伺服馬達，常用於模擬玩具、小型機器人或者各種控制系統中。它具有較高的轉速（可達到 55 轉/分鐘）和較強的扭矩（可達到 1.8 公斤·米），使用方便且價格實惠，其規格、外型(如表 5、圖 16)。我們使用伺服馬達連接 3D 列印的柵欄桿連接。

表 5 Mg90s 伺服馬達規格

| | |
|------|------------------|
| 尺寸 | 23 x 12.2 x 29mm |
| 重量 | 11g |
| 輸入電壓 | 4.8V~6V |
| 轉速 | 55rpm/分鐘 |
| 轉矩 | 1.8kg·m |
| 角度 | 180° |
| 接口 | 舵機模式接口 |



圖 16 Mg90s 伺服馬達

5. 雙色 LED 燈

雙色 LED 燈是一種具有兩種不同顏色的可變 LED 燈。它可以通过控制電壓來切换不同的顏色。通常情況下，一個雙色 LED 燈有一個紅色 LED 和一個綠色 LED，但也可以有其他顏色的組合，如黃色、藍色等，其規格、外型(如表 6、圖 17)。我們用雙色 LED 燈來顯示車位的狀態。

表 6 雙色 LED 燈規格

| | |
|------|------------|
| 尺寸 | 5mm |
| 工作電壓 | 2V~3V |
| 顏色 | 紅色和綠色 |
| 接口 | 標準的 LED 接口 |
| 壽命 | 可達到數千小時 |



圖 17 雙色 LED 燈

6. 光敏電阻

光敏電阻（也稱為光敏電阻器、光敏電阻器等）是一種電子元器件，具有對光照強度敏感的特性。它的電阻值會隨著光照強度的變化而變化。通常情況下，光敏電阻的電阻值越小，光照強度越大，其規格、外型(如表 7、圖 18)。我們使用光敏電阻來偵測是否有車子進入。

表 7 光敏電阻規格

| | |
|------|--------------|
| 尺寸 | 3mm |
| 工作電壓 | 0.1V~0.3V |
| 工作溫度 | -25°C ~ 85°C |
| 光敏常數 | 10~60 |
| 產生電流 | 10uA~20uA |



圖 18 光敏電阻

7. 白光 LED 燈條

白光 LED 燈條是一種使用多個白光 LED 燈組成的燈條，常用於裝飾燈光、照明、廣告等。白光 LED 燈條的規格因產品而異，其規格、外型(如表 4、圖 19)。我們使用白光 LED 燈條來穩定光敏電阻的數值，不因環境燈光不同而讀取錯誤。

表 8 白光 LED 燈條規格

| | |
|------|------|
| 尺寸 | 30cm |
| 工作電壓 | 12V |
| 顏色 | 白色 |



圖 19 白光 LED 燈條

8. UNO

Arduino Uno 是一款基於 ATmega328P 的開發板。它有 14 個數位輸入/輸出接腳(其中 6 個可用作 PWM 輸出)，6 個類比接腳，是最常見的 Arduino 開發板，其規格、外型(如表 9、圖 20)。我們使用 Arduino Uno 來接收 Python 辨識車牌的數值，並讓 LCD 顯示。

表 9 UNO 規格

| | |
|---------|-------------|
| 核心處理器 | ATmega328P |
| IO 數 | 14 |
| 程式記憶體大小 | 32KB |
| 程式記憶體類型 | 閃存 |
| 電壓 - 電源 | DC 7V ~ 12V |
| 時脈速度 | 16MHz |



圖 20 UNO

9. CH340G

CH340G 晶片是一種 USB 轉串口晶片，常用於連接電腦嵌入式系統之間的通信。它可以將 USB 信號轉換為串口信號，並且具有較高的傳輸速率和較低的功耗，其規格、外型(如表 10、圖 21)。我們使用 CH340G 來燒錄 ESP32-CAM。

表 10 CH340G 規格

| | |
|-------|-------------------|
| 尺寸 | 4mm x 4mm |
| 工作電壓 | 3.3V~5V |
| 傳輸速率 | 最高可達到 2Mbps |
| 支持的協議 | UART、SPI、I2C |
| 支持的系統 | Windows、Mac、Linux |



圖 21 CH340G

(二)、機構原理

當車子入場並壓到極限開關後，ESP32CAM 會拍照並開始辨識車牌，車牌辨識完畢之後擋桿會打開，讓車子進入。

當有車子停在車格上時，會觸發光敏電阻並使用來顯示停車狀態的 LED 變成紅色，剩餘車位也會減少。

有人預約時，會透過 APP 傳送指令給 HTTP sever，再透過服器透過 UART 操控主控版讓 LED 變成無色，剩餘車位也會減少。

軟體介紹。

(三)、軟體介紹

1. Autodesk Inventor

Autodesk Inventor 是一個電腦輔助設計(電腦輔助製圖)應用程序，用來進行 3D 機械設計、模擬、顯示、與儲存(如圖 22、23)。



圖 22 Autodesk Inventor logo

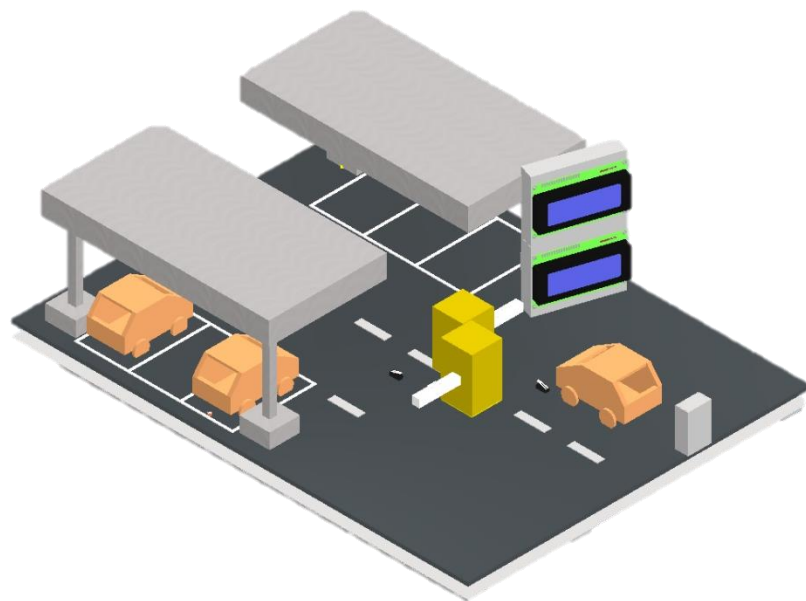


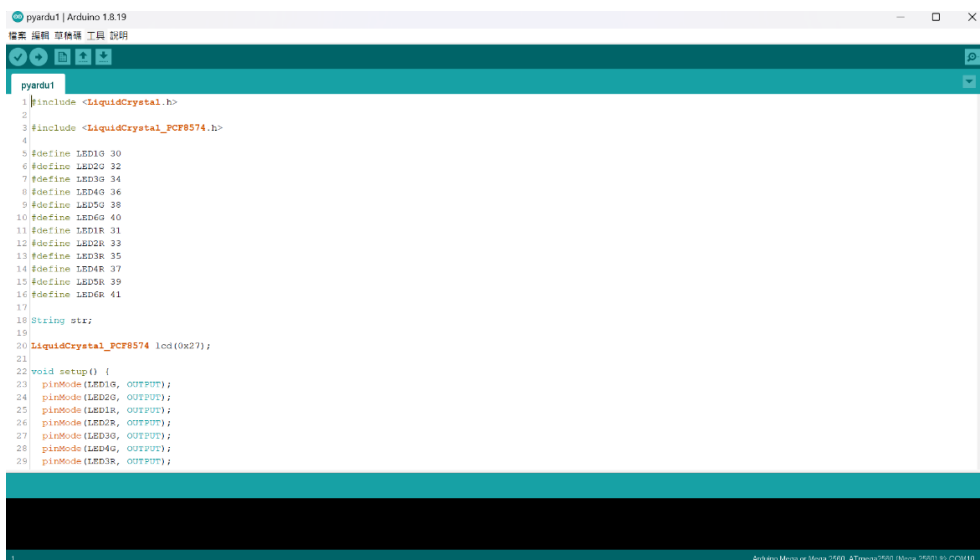
圖 23 Autodesk Inventor 3D 圖繪製介面

2. Arduino IDE

Arduino IDE 是一個開源碼的硬體程式語言編寫軟件，它兼具類似 java、C++ 等伺服器語言的開發環境，且擁有許多已模組化的套件與函式庫，提供初學者使用(如圖 24、25)。



圖 24 Arduino logo



```
pyardu1 | Arduino 1.8.19
檔案 編輯 庫檢索 工具 說明
pyardu1
1 #include <LiquidCrystal.h>
2
3 #include <LiquidCrystal_PCF8574.h>
4
5 #define LED10 30
6 #define LED20 32
7 #define LED30 34
8 #define LED40 36
9 #define LED50 38
10 #define LED60 40
11 #define LED1R 31
12 #define LED2R 33
13 #define LED3R 35
14 #define LED4R 37
15 #define LED5R 39
16 #define LED6R 41
17
18 String str;
19
20 LiquidCrystal_PCF8574 lcd(0x27);
21
22 void setup() {
23   pinMode(LED10, OUTPUT);
24   pinMode(LED20, OUTPUT);
25   pinMode(LED1R, OUTPUT);
26   pinMode(LED2R, OUTPUT);
27   pinMode(LED30, OUTPUT);
28   pinMode(LED40, OUTPUT);
29   pinMode(LED3R, OUTPUT);

```

圖 25 Arduino 程式撰寫

3. HTTP sever

HTTP 伺服器是一種軟體，用於接收和處理來自網絡客戶端的 HTTP 請求，並通過 HTTP 協議向客戶端返回響應。HTTP 伺服器可以用於提供網頁、文件、圖像等內容，並且可以運行在各種操作系統平臺上(如圖 26)。



圖 26 HTTP logo

4. App Inventor

App Inventor 是一個由 Google 開發的可視化區塊式程式開發環境，用於創建 Android 設備的移動應用程式。它可以讓用戶使用區塊組合的方式進行程式開發，而不需要寫代碼，更易於學習和使用。App Inventor 提供了大量內建組件和函式庫，可以讓開發者快速創建與設備互動的應用程式。目前 App Inventor 維護由 MIT，並且是開源免費的，開發者可以在網上使用(如圖 27)。



圖 27 App Inventor logo

5. Fritzing

Fritzing 是個電子設計自動化軟體。它支持設計師，研究人員和愛好者參加從物理原型到進一步實際的產品。還支持用戶記錄其 Arduino 和其他電子零件為基礎的原型(如圖 28、29、30)。



圖 28 Fritzing logo

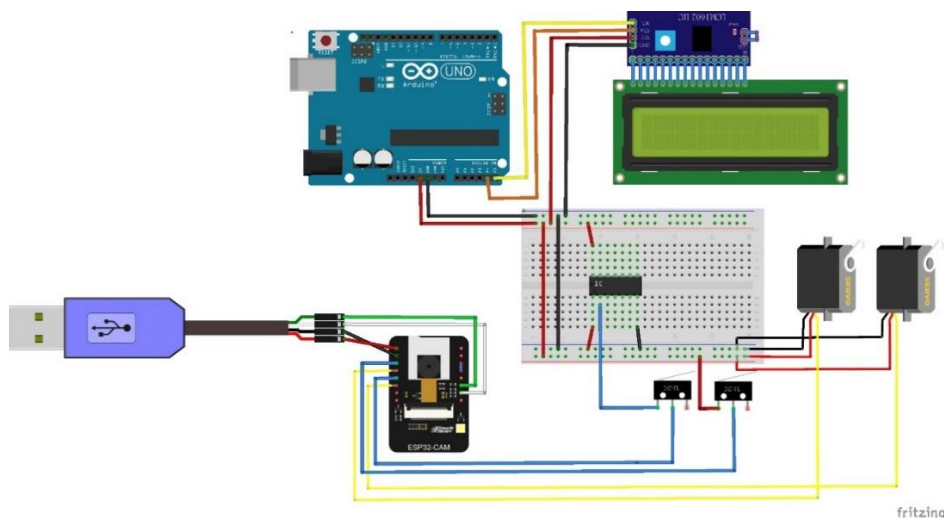


圖 29 車牌辨識接線圖

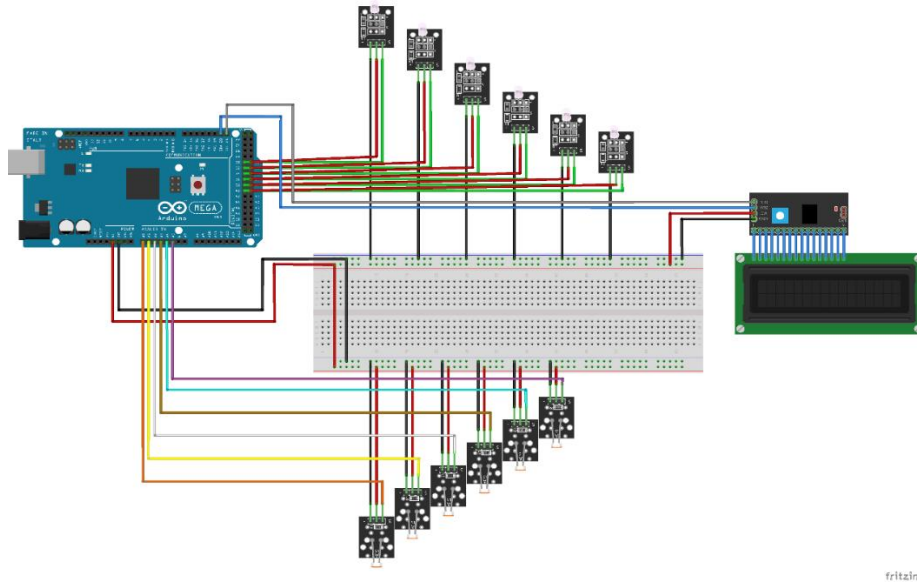


圖 30 燈號和剩餘車位接線圖

6. Vscode

Visual Studio Code (簡稱 VS Code) 是一個由微軟公司為 Windows, Linux 和 macOS 開發的免費開源碼編輯器。它旨在是一個輕量級但強大的工具，用於編輯各種程式語言的代碼。它除了常見的文本編輯器的功能外，也有許多強大的功能，例如自動完成、語法高亮、檔案瀏覽器、支援多種語言、支援擴充套件等(如圖 31)。



圖 31 Vscode

7. Jupyter

Jupyter 是一個免費的開源互動計算環境，它允許您創建和共享包含實時程式碼、方程式、可視化和敘述文字的文件。Jupyter 是一個從 IPython 分離出來的項目，IPython 是 Python 程式語言的互動式外殼。Jupyter Notebook 是用於在瀏覽器中創建和共享附帶註釋和可視化元素的程式碼的工具。Jupyter Notebook 支持多種語言，如 Python、R、Julia、Scala 和

Haskell，並與其他工具集成，如 Matplotlib、NumPy 和 pandas(如圖 32)。



圖 32 Jupyter

伍、研究結果

本專題結構可以分成硬體結構與軟體通訊架構，以下所示。

一、硬體結構

本次專題結構，如圖 33。

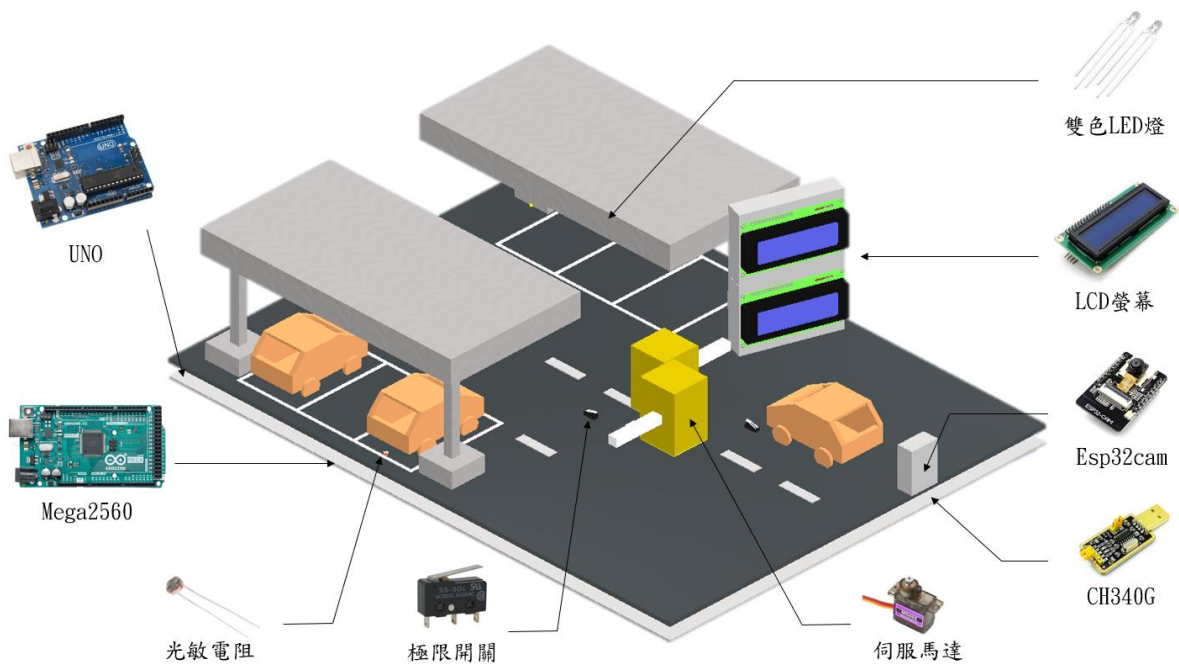


圖 33 Autodesk Inventor 3D 結構圖

(一)、柵欄桿

透過 3D 列印製作的擋桿(圖 34)，將 Mg90s 伺服馬達裝置上去。

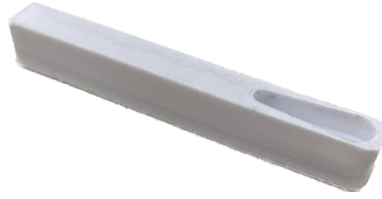


圖 34 柵欄桿

(二)、LCD 框架

透過 3D 列印製作 LCD 外框(圖 35)，用來遮擋 LCD 螢幕外圍。



圖 35 LCD 框架

二、軟體通訊架構

我們在車牌辨識中使用 Python 運用 HTTP 抓取 ESP32-CAM 網頁上的圖片，再用 UART 傳車牌數值(如圖 36)。而預約車位使用 APP 連接 HTTP 伺服器來達到遠端預約車位的功能。本專題通訊關係(如圖 37)。

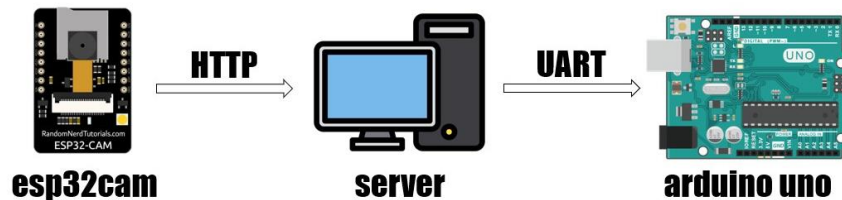


圖 36 車牌辨識的通訊圖

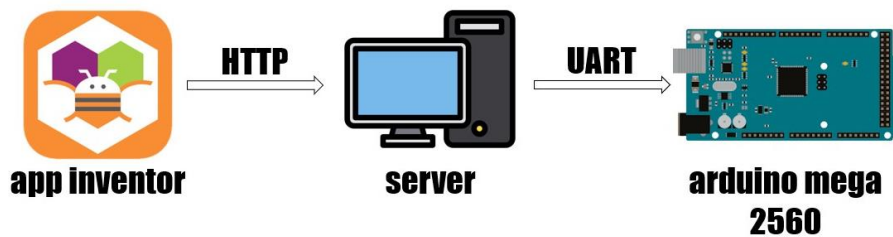


圖 37 預約車位的通訊圖

(一)、HTTP 伺服器

我們使用 Python 撰寫 HTTP 伺服器。APP 能透過 WI-FI 連接伺服器並傳送指令。

(二)、Serial(UART)

Serial 為 Arduino 元件互相溝通的方式之一，藉由 serial 讓電腦辨識出的車牌顯示在 LCD 螢幕上，HTTP 伺服器的指令也會透過 Serial 跟主控版溝通。

三、成果展示

(一)、APP 介面

APP 分為兩個部分，第一個畫面要輸入車牌號碼才能開啟預約畫面(如圖 38)。在預約畫面要先選擇你要停車的時間，選擇完時段後選擇你要的車位，如果你突然不想要停車可以取消預約(如圖 39、40)。



圖 38 APP 主頁



圖 39 APP 預約畫面



圖 40 APP 完成預約

(二)、車牌辨識

ESP32-CAM 將擷取到的圖片放在網頁上(如圖 41)，再由 python 抓下圖片，並且執行車牌辨識，接著再將讀取出來的數值傳到 arduino 再由 LCD 顯示(如圖 42)。

ESP32-CAM Last Photo

It might take more than 5 seconds to capture a photo.

ROTATE CAPTURE PHOTO REFRESH PAGE



圖 41 esp32cam 網頁



圖 42 LCD 顯示車牌

(三)、燈號更改和剩餘車位

我們使用 LED 燈用來當車位狀態的表示，當一開始沒人停車時是以綠色表示，讓使用者明顯的看出哪裡可以停車，有人預約後會變成無色，有車子停入之後那一格的燈號會變成紅色(如圖 43)。而當有車位停入或有預約，則剩餘車位減少(如圖 44)。

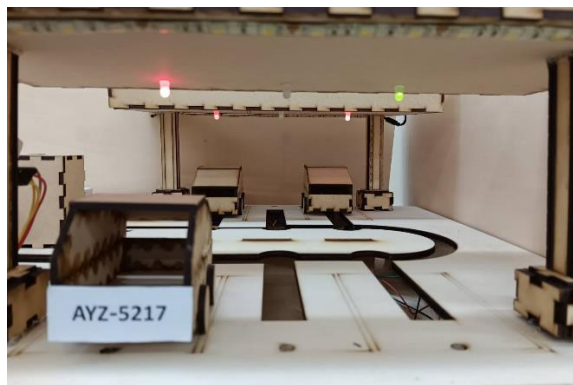


圖 43 三種燈號顯示

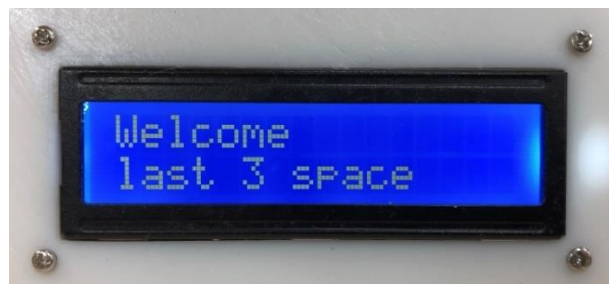


圖 44 剩餘車位顯示

陸、討論

一、設備性能不足

一開始辨識車牌的功能和預約系統是打算在同一台電腦裡寫，但後來發現有些函示庫跟電腦不吻合，而且電腦進行車牌辨識時的成效不佳，常常會有無法辨識的問題，於是我們便決定使用兩台電腦，一台用來辨識車牌，另一台用來作主系統。

二、光敏電阻數值太敏感

當我們機電整合時，常常會搬運到不同地方進行影片的拍攝，每次搬運到不同地方時，都會遇到光敏電阻數值更變，造成無法正確更改車位狀態，必須調整感測數值，後來使用白光 LED 燈調來提供穩定的光源，使光敏電阻數值不會因為外在環境變動。

三、使用極限開關和光敏電阻

我們使用極限開關和光敏電阻而不是紅外線感測器，是因為當 ESP32-CAM 連接到 WI-FI 時，無法接收到類比訊號，而選用光敏電阻是因為無法找到適合我們模擬停車場的距離，紅外線感測器有 4 公分以下的視野盲區，而且價錢較為昂貴。

柒、結論

經過組員們討論過後我們決定要做一個停車場的預約系統，在一開始，我們的停車場模型是設計十二個停車格，但基於製作成本以及整體接線的種種因素後，我們決定將它縮小至六個停車格的大小來做模擬。停車場的整體結構我們是利用木板去做切割並且組裝，而至於擋桿和 LCD 螢幕之蓋板則是利用 3D 列印製作，為了讓整體外觀看起來較整齊，我們將柱子，屋簷以及底板做成中空，能夠讓線路放置在裡面，除此之外，我們的底板設計了軌道，能使車輛更順利的移動。

我們原本想利用紅外線來當作偵測是否有車停入的原件，但是沒有適合偵測這個距離的紅外線感測器只好使用光敏電阻來替代。而為了讓光敏電阻的偵測更為穩定我們在上方加裝了 LED 燈條使亮度充足降低偵測錯誤的機率。

然而我們的作品仍有需要改進的地方像是我們希望能夠使用其他元件來代替極限開關當作觸發 ESP32CAM 拍照的條件，因為如果進場時輪子沒壓到就無法進行車牌辨識的動作。

停車場內部的燈號有三種顏色，無人使用時是綠色，有人預約時是無色，有人停入時則是紅色，而為了讓使用者不要搞混顏色，我們在 APP 的界面下方有輔助圖示說明，讓使用者能夠輕輕鬆鬆地預約。

我們的專題從零開始，每一位組員都非常地認真且努力，每一位組員都非常地認真且努力，不論是程式的撰寫也好，硬體的設計和製作也好，每個組員缺一不可，在製作的過程中，我們各自去學習自己負責的部分，並且在學習的過程中成長，每個人負責的部分不一樣，得到的收穫也不盡相同，進而將學到的專業能力運用在專題上，我們的專題就是這樣一點一滴拼湊起來的。過程中，我們不但學到了專業能力，也體悟到團隊合作的重要性，當遇到困難時，同心協力突破難關是專題成功的一大關鍵。當遇到問題不知如何解可以先上網查詢，網路上的資訊應有盡有，我們也可在查詢料的過程中增加自己的知識，除此之外，也可透過老師以及各方面的專業人士的協助讓問題更有效率的被解決，然而，組員間互相提出各自的想法並將優點記錄下來或是其他人給出意見將他們的想法記錄下來，都能讓日後在製作專題的過程中多一點想法並且將整體的完整度提高。最後，希望預停將來能夠越來越好，讓大家減少停車時的煩惱。

捌、參考資料及其他

一、網路資料

1. Arduino 與 Python 通訊(三)Arduino 傳送資料給 Python。取自：
<http://blog.3dgowl.com/arduino-arduino%E8%88%87python%E9%80%9A%E8%A8%A%E4%B8%89arduino%E5%82%B3%E9%80%81%E8%B3%87%E6%96%99%E7%B5%A6python/>
2. Arduino 筆記(89)：ESP32CAM 初次設定並透過網頁瀏覽照片。取自：
<https://atceiling.blogspot.com/2020/11/arduino89esp32cam.html>
3. 其他元件- Wi-Fi 通訊。取自：
<http://ai2.npust.edu.tw/PDF/%E5%85%B6%E4%BB%96%E5%85%83%E4%BB%B6%20-%20WiFi%E9%80%9A%E8%A8%A.pdf>
4. Coding 起來 — Python — 一行指令就能輕鬆建立網頁伺服器 — SimpleHTTPServer 套件 — http.server 使用教學。取自：
<https://chwang12341.medium.com/coding%E8%B5%B7%E4%BE%86-python-%E4%B8%80%E8%A1%8C%E6%8C%87%E4%BB%A4%E5%B0%B1%E8%83%BD%E8%BC%95%E9%AC%86%E5%BB%BA%E7%AB%8B%E7%B6%B2%E9%A0%81%E4%BC%BA%E6%9C%8D%E5%99%A8-simplehttpserver%E5%A5%97%E4%BB%B6-http-server%E4%BD%BF%E7%94%A8%E6%95%99%E5%AD%B8-34c30b81c26> USART HMI 智慧串列屏介紹。
5. 使用 OpenCV 及 Tesseract 進行 OCR 辨識(2) – Medium。取自：
<https://medium.com/ntt-data-idi-platform/%E4%BD%BF%E7%94%A8-opencv-%E5%8F%8A-tesseract-%E9%80%B2%E8%A1%8C-ocr-%E8%BE%A8%E8%AD%98-2-%E4%BD%BF%E7%94%A8-opencv-%E9%80%B2%E8%A1%8C%E5%BD%B1%E5%83%8F%E5%89%8D%E8%99%95%E7%90%86-7c81120a4608>