

國立成功大學工程科學系

專題競賽成果報告

「**DRONK**」——結合外送平台之無人機外送

學 生：俞杉麒

陳颺

王毅暄

郭紘碩

葉芷彤

指導教授：賴謹峰

中華民國一一〇年九月

## 壹、研究動機與目標

近年來網路科技技術的普及使大眾的消費模式改變，線上購物已成為常態。惟2019年11月底爆發之新冠肺炎疫情，致使餐飲外送正式走入大眾線上購物的視野。由於臺灣地狹人稠，餐飲外送業多屬意機車做為其營業之載具。然而吾人親身體驗過其服務後，發現經常因交通堵塞而出現訂單延宕，更有甚者，亦常於新聞媒體上得知外送業者因趕時間而於外送途中發生車禍，危害人身安全。吾人便思考是否有外送方式能兼顧速度及安全，而近期興起的無人機便是解答之一。

因目的為出於餐飲需求的外送，吾人於此專題將成為一飲料業者，欲利用無人機（Drone）外送自家所售之飲料（Drink），是為「**DRONK**」系統。

## 貳、研究步驟

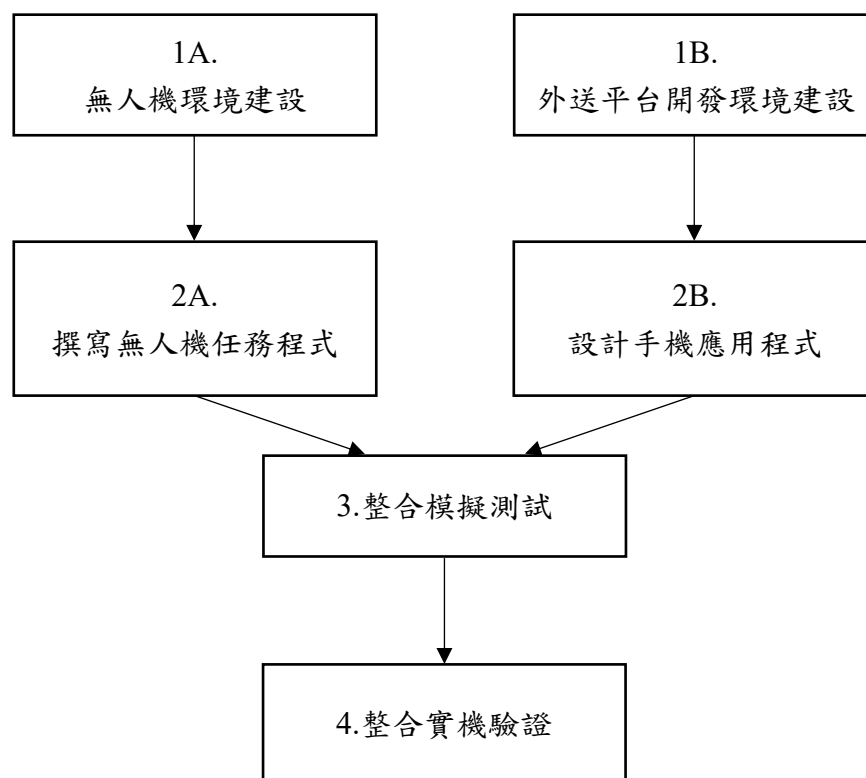


圖 2-1 研究步驟

本專題的研究步驟如圖 2-1。

本專題各項研究步驟說明如下：

1A.無人機環境建設：

搭建一架符合此專題需求之無人機，惟因實驗安全考量吾人並未實際考慮載重能力，因此該無人機並無法實際運送貨物。

2A.撰寫無人機任務程式：

利用 Dronekit-Python 提供之 API 撰寫程式以供樹莓派（Raspberry Pi 3）與飛控板（Pixhawk）溝通，藉此控制無人機完成所設計之任務。

1B. 外送平台開發環境建設：

a) Xcode：App 開發平台

Xcode 是一個由蘋果公司所開發設計的 Mac OS 及 iOS 應用程式整合開發環境（IDE）。

b) Firebase：資料庫系統

Firebase 是一款雲端資料庫，它提供了身分驗證（Authentication）、即時資料庫（Cloud Firestore）、儲存（Storage）、主機（Hosting）、儲存（Storage）、函數編寫代碼（Function）及機器學習功能（ML Kit）等後端服務。同時 Firebase 也提供了品質管理、數據分析、擴展功能…等等豐富的支援服務。

此專題主要只用到了資料庫以及儲存的部分。

## 2B.設計手機應用程式：

### a) 功能設想、手繪 App UI 介面草稿 (如圖 2-2)

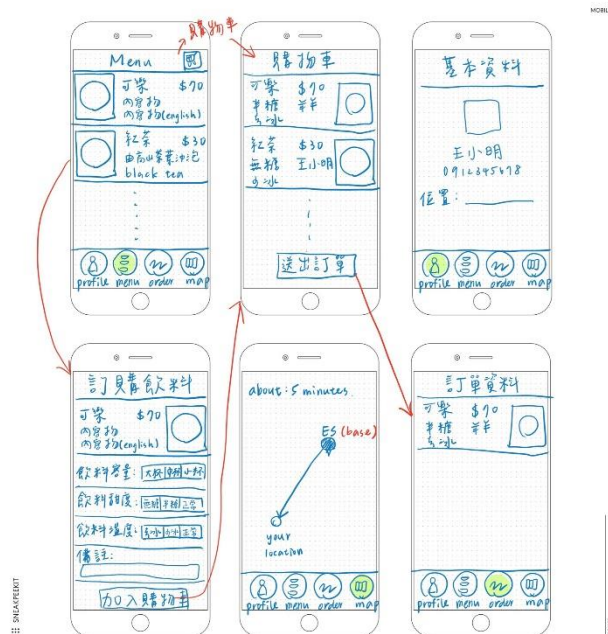


圖 2-2 功能設想、手繪 App UI 介面草稿

### b) 資料格式設計

飲料資料格式：

name : String (飲料名稱)

image : String (飲料照片)

price : Int (單杯價格)

content : String (介紹)

eng\_content : String (英文介紹)

訂單資料格式：

user : String (訂購人)

long : Int (經度)

lat : Int (緯度)

name : String (飲料名稱)

image : String (飲料照片)

price : Int (總金額)

content : String (甜度冰塊等等)

remark : String (備註)

c) 採取 MVC 架構為應用程式基礎（如圖 2-3）

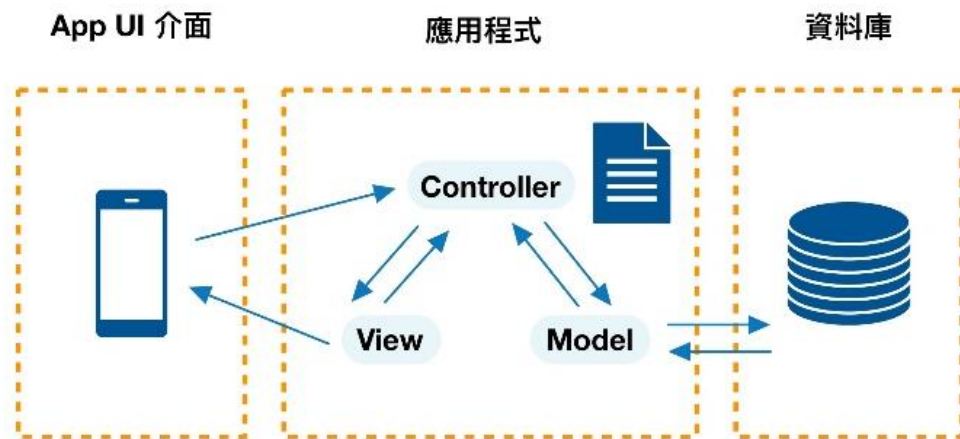


圖 2-3 MVC 架構示意圖

d) 使用 Xcode storyboard 設計 App UI 介面（如圖 2-4）

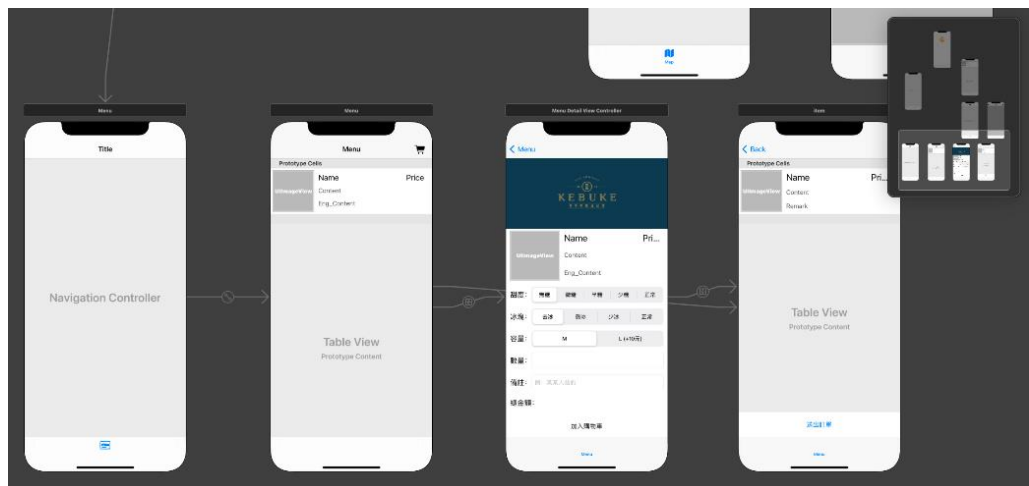


圖 2-4 App 介面設計示意圖

e) 從 Firebase 後台資料庫建立資料庫內之動態資料(Menu)

f) 取得應用程式金鑰和 ID 的設定檔，加入專案中以連接應用程式和 Firebase 資料庫

### 3. 整合模擬測試

透過 Dronekit-SITL (Software In The Loop) 以及 Xcode 的 simulator 模擬任務情境，並透過 Mission Planner 測試無人機在模擬情境中是否完成任務目標。

### 4. 整合實機驗證

利用實體手機，使用設計完成的 app 發送一訂單。

將任務匯入樹莓派，透過遠端桌面 (VNC Viewer) 執行任務檔案，藉由樹莓派與飛控板溝通，使其控制無人機，後驗證是否成功完成訂單。

### 參、DRONK 系統架構

本專題規劃 **DRONK** 系統架構分為前端（客戶端）、後端（資料庫工作人員）及無人機三大部分（如圖 3-1）。

前端（客戶端）使用手機 App 下單選購飲料，將訂單內容上傳至雲端資料庫。後端（資料庫工作人員）設置飲料菜單資訊，並管理訂單。於接收到新訂單時對無人機提出需求，執行任務檔（dronk.py）。

無人機透過遠端遙控的樹莓派啟動，樹莓派藉由機載 4G 行動網路從後台更新訂單資料，記錄下新訂單的客戶資訊及客戶指定之目的地。無人機飛往目的地遞交飲料後返航基地。

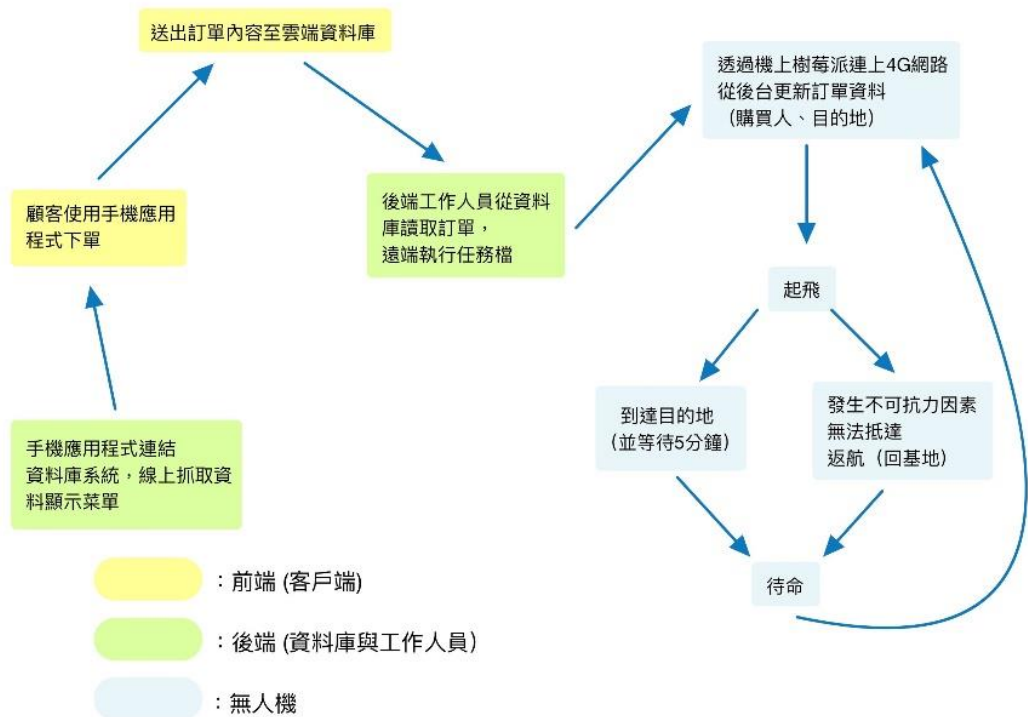


圖 3-1 DRONK 系統架構

#### 肆、無人機

##### 1. 無人機搭建：



圖 4-1 無人機

無人機（如圖 4-1）包含無人機本體、PIXHAWK4 飛控板、Raspberry Pi 3、華為 E3372h 4G 行動網卡。



圖 4-2 試飛

初始試飛時（如圖 4-2）僅含有無人機本體與 PIXHAWK4 飛控板。



## 2.環境架設：

利用 MissionPlanner 實時監控無人機狀態（如圖 4-3）

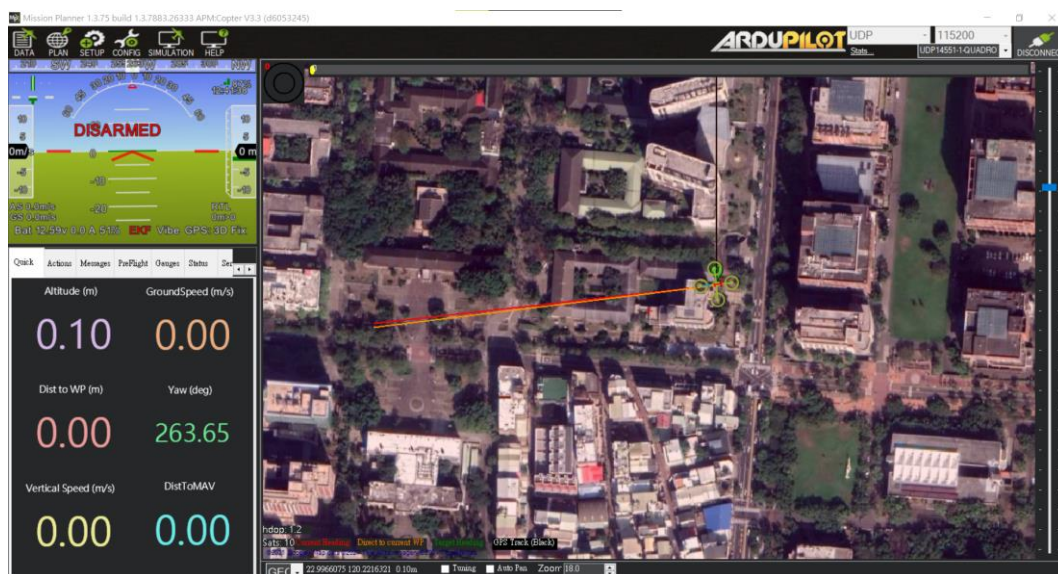


圖 4-3 MissionPlanner 監控無人機

以 4G 行動網卡於樹莓派上模擬乙太網路，使無人機於任何地方皆有網路訊號與後端進行交流，更貼近實際使用情境，亦便於監控無人機狀態。

透過 Dronekit 函式庫撰寫無人機任務程式（如圖 4-4）

```
65 arm_and_takeoff(50)
66
67 print("Set default/target airspeed to 10")
68 vehicle.airspeed = 10
69
70 print("Going towards first point for 40 sec...")
71 fp = LocationGlobalRelative(22.999219, 120.221206, 100)
72 vehicle.simple_goto(fp)
73
74 # sleep so we can see the change in map
75 time.sleep(40)
```

圖 4-4 基礎飛行程式碼式意圖

藉由各個函式組合可以撰寫出符合任務需求之飛行計畫。

```

7  #連接資料庫(由於設定任何人都能存取，所以不需要設定其他的API Key)
8  db_url = 'https://dronk-f75d9-default-rtdb.firebaseio.com'
9  fdb = firebase.FirebaseApplication(db_url, None)
10
11 #在user下查詢新增的資料(.get讀取)
12 user = fdb.get('/Order/user', None) #None全部讀取，1代表讀取第一筆，以此類推
13 latitude = fdb.get('/Order/latitude', None)
14 longitude = fdb.get('/Order/longitude', None)
15
16 print(user)
17 print(latitude)
18 print(longitude)

```

圖 4-5 與資料庫溝程式碼式意圖

**DRONK** 系統旨在由資料庫獲取飛行必要資訊（經緯度及訂購人）。

藉由圖 4-5 所示之程式碼於任務開始時自動更新目標地址。

由樹莓派執行任務檔（飛行計畫）後，向飛控板發出訊號，並交由飛控板控制無人機飛行。

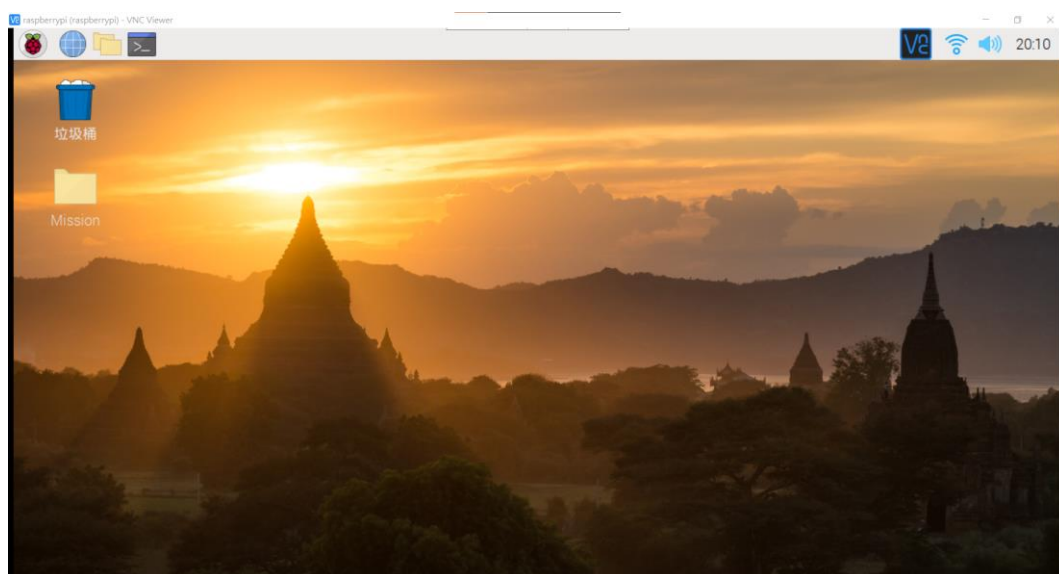


圖 4-6 遠端樹莓派桌面

利用 VNC Viewer 實踐遠端桌面（如圖 4-6），形同建立一地面站，可遠端直接對樹莓派下達指令，進而控制無人機的行為。

## 伍、外送平台與應用程式

### 1. 應用程式架構：

MVC 架構採用（如圖 5-1）

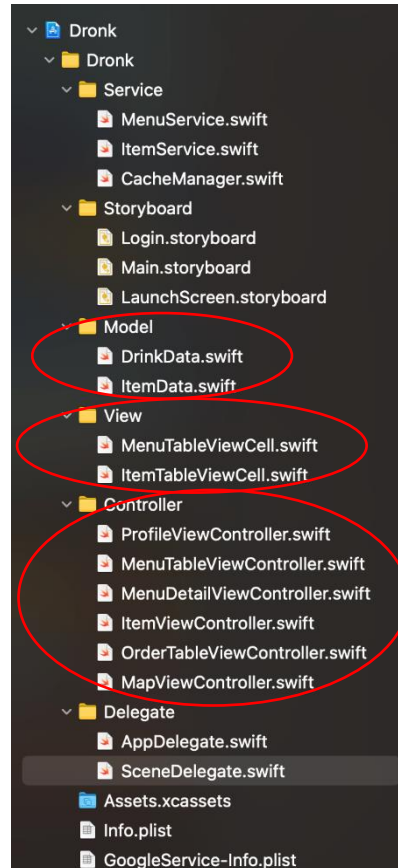


圖 5-1 MVC 架構

圖 5-1 最下面之.plist 檔為連結 firebase 資料庫的金鑰檔。

此應用程式使用了 MVC 架構（Model-View-Controller 軟體設計模式），因為在應用程式開發中，非常重視關注點分離（separation of concerns, SOC）此一設計原則，以便於 App 各種不同部分的分工。

而其他常用之功能，也寫成一個套件放在「Service」中，方便多次呼叫以及簡化程式碼，增加程式碼的可讀性。

## 2. App：

點飲料 App 的四個基礎畫面（如下圖 5-2、圖 5-3、圖 5-4、圖 5-5）

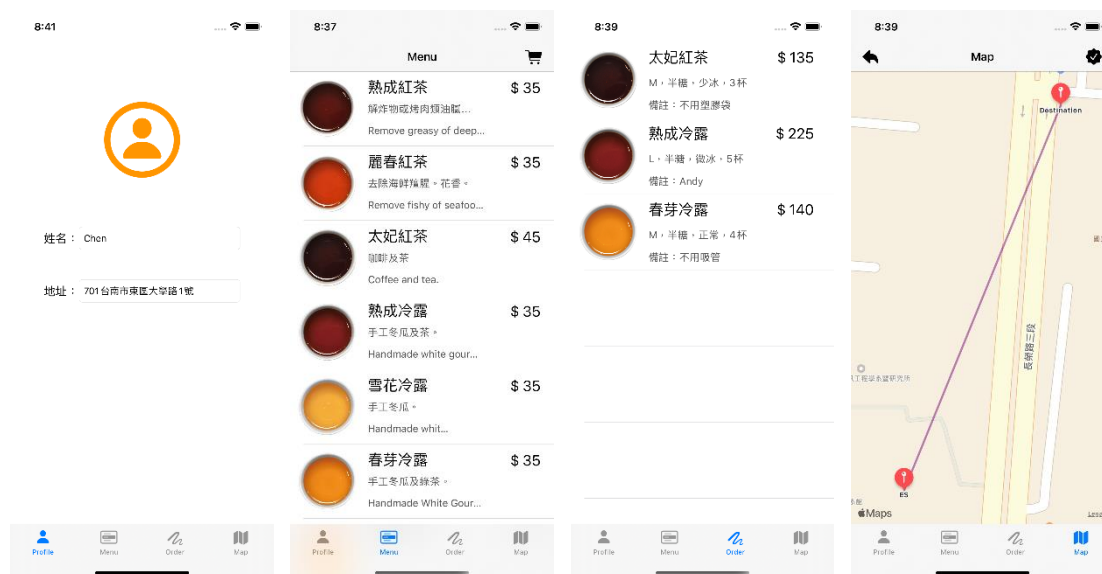


圖 5-2

圖 5-3

圖 5-4

圖 5-5

圖 5-2 為「個人資料設定」。圖 5-3 為「Menu 列表」。

圖 5-4 為「訂單紀錄」。圖 5-5 為「無人機運送路線」。

其中可於圖 5-2 之地址欄位填詳細地址，或由圖 5-5 直接釘選地圖地點來確定無人機之運送目的地座標。



圖 5-6 點飲料畫面

點飲料內容畫面（如圖 5-6）

在此畫面可以選擇飲料的甜度冰塊等內容，並且會在此預覽總金額為多少，確認無誤後即可加入購物車，接著送出訂單。

此 App 使用之素材（圖片以及內容）均來自參考文獻[5]。

### 3. 資料庫：

Menu 資料庫（如圖 5-7）



圖 5-7 Menu 資料庫

此為 Menu 資料庫，商家可由後台隨時動態調整 Menu 資訊（包含新增品項、更改價格等等）。

而非直接將所有資料寫死在應用程式內，變成靜態表格的概念，此法不符合實際使用之可行性。

其中飲料圖片是存在資料庫的另一個空間，並且是以網址的形式呼叫。

訂單資料庫（如圖 5-8）



圖 5-8 訂單資料庫

此為訂單資料庫，可以看到訂單的詳細資訊，以及經轉換過後的目的地座標，當有待命之無人機時，備好訂單內容後即可讓無人機載貨飛往目的地。

## 陸、系統整合成果

### 1. 模擬測試

#### a) 透過 Xcode 模擬機進行下單（如圖 6-1）

於此模擬中我們設成大工科系館為基地、成大博物館為我們的訂單地址



圖 6-1 模擬點餐

#### b) 後台工作人員讀取新訂單（如圖 6-2）

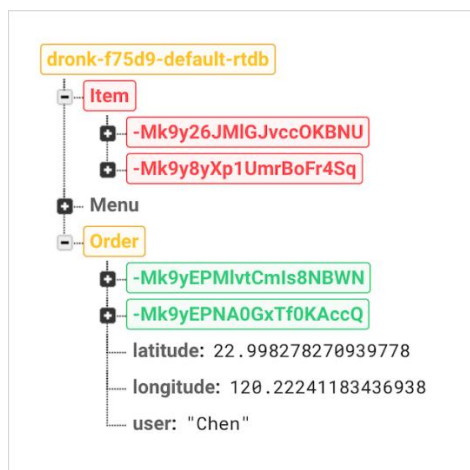


圖 6-2 後台訂單

紅色：剛刪除的資料    黃色：剛更改的資料    綠色：剛新增的資料



後端工作人員可由資料庫顯示之顏色變化來確認是否有新的訂單

圖 6-2 代表「刪除購物車中之內容 (Item)，並且送出訂單 (Order)」。

c) 虛擬機執行任務檔並獲取訂單資訊 (如圖 6-3)

```
dhiptmc@dhiptmc-VirtualBox:~/Desktop/mission5$ python dronk.py --connect udp:127.0.0.1:14550
Chen
22.9963307
120.2195393
Connecting to vehicle on: udp:127.0.0.1:14550
```

圖 6-3 虛擬機獲取訂單資訊

d) 飛行流程圖 (如圖 6-4、6-5、6-6、6-7)

起飛→到達目的地上空→降落→等待 5 分鐘→起飛返回基地

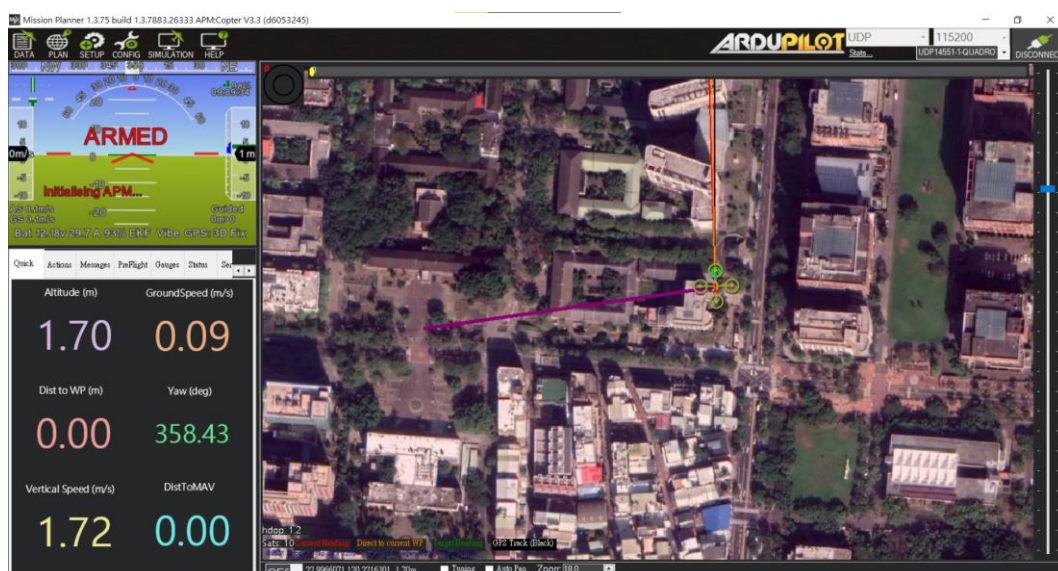


圖 6-4 起飛





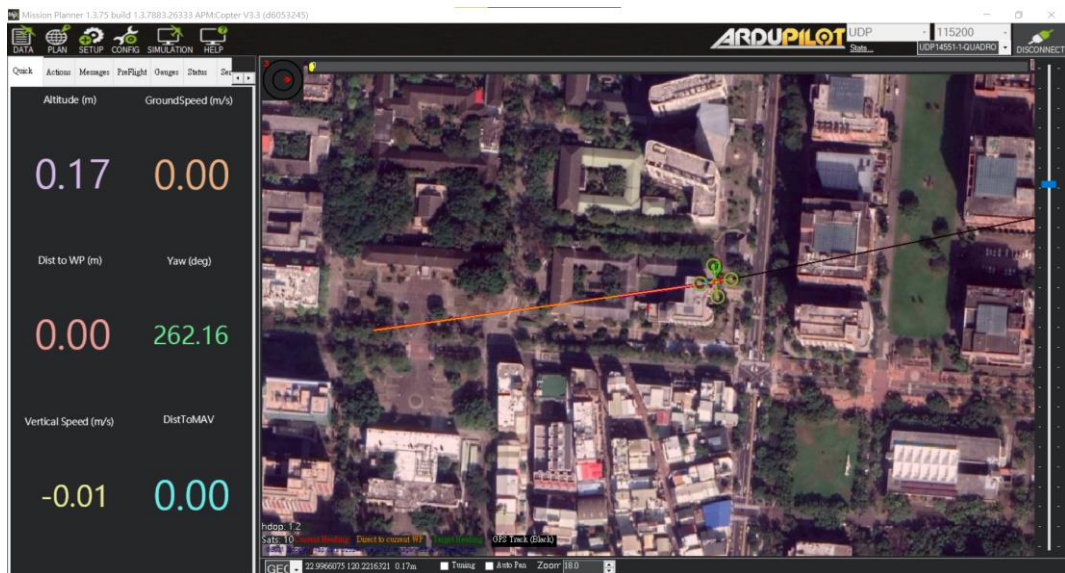


圖 6-7 返回基地

於模擬中順利完成指定之任務要求。

## 2.實機驗證

### a) 透過實體手機進行下單（如圖 6-8）



圖 6-8 手機下單示意圖

b) 後台工作人員讀取新訂單（如圖 6-2）

並遠端遙控樹莓派執行任務檔開始飛行任務（如圖 6-9）

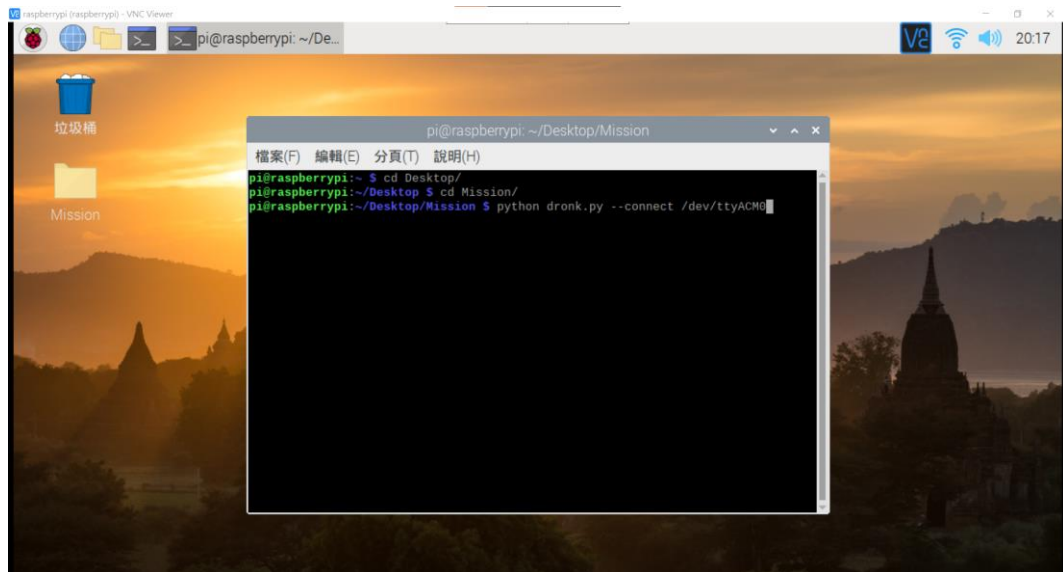


圖 6-9 透過 VNC Viewer 對樹莓派遠端下達指令

c) 驗證無人機是否可以成功完成訂單要求

本專題經過多次測試，無人機均無法順利起飛，起飛失敗原因如下：

一、吾人搭建之無人機無法額外載重，初始搭建時並未留意樹莓派及 4G 網卡的「額外」重量，導致飛機於起飛時失去平衡，無法如同初始測試時順利起飛。

二、電流不足，雖樹莓派算是功耗低的協同電腦（Companion Computer），但其與無人機共用一顆電源仍可能導致無人機無法獲得足夠電流，進而導致起飛失敗。另外，共用一顆電源也使無人機續航力下降。

三：GPS 訊號太弱，吾人使用之 GPS 模組於無人機初始搭建時就時有問題，且本專題於成大工科系系館一樓實驗，有可能因 GPS 訊號不夠強而導致無人機無法順利完成定位並成功起飛。

## 柒、結論與建議

在此專題中，我們於模擬中成功實現吾人所欲搭建之基本外送飲料平台。透過此平台即可達成 0 接觸且迅速安全的餐飲運送。

惟無人機部分有許多可改進部分，包括搭建更大台的無人機以實現實際的載貨功能，使用更大容量的電池以增加續航力、購置更精良的 GPS 天線以維持訊號穩定性。

而 App 的部分亦仍有相當大的進步空間，目前市面上「Uber EATS」和「foodpanda」即是此類型 App 最好的例子。它們功能完備、可以即時追蹤訂單位置、以及擁有更多個性化服務。

以目前情況而言，吾人認為能力範圍內有辦法增進的新功能應是：

- 一、提供「單趟行程，多筆訂單」運送。
- 二、訂單更新自動通知無人機，毋須人工派遣無人機。以儲存於任務列表之形式，完成物流全自動化目標。
- 三、顯示即時無人機位置，並且以無人機的飛行速度和基地與訂單地址兩地間的距離推算出預計抵達時間。
- 四、於無人機上增加一台相機，並藉由人臉、手勢辨識進一步增加與顧客交易之安全性。

## 捌、參考文獻

- [1] iOS 13 App 程式設計實戰心法 - Simon Ng (APPCODA)
- [2] iOS 13 App 程式設計進階攻略：不可不學的開發技巧 - Simon Ng (APPCODA)
- [3] iOS 13 程式設計實戰 - 朱克剛
- [4] iPhone APP 實作開發 - 黃士嘉/張力元
- [5] [Swift 教學] Swift 訂飲料 App  
<https://www.maxlist.xyz/2019/09/08/swift-drink-app/>
- [6] Firebase 是什麼？集 APP 後端開發與分析於一身的強大工具！  
<https://tw.alphacamp.co/blog/2016-07-22-firebase>
- [7] 【Firebase】不用錢的 NoSQL 資料庫？  
<https://william-weng.github.io/2019/06/10/firebase-helloworld/>
- [8] Firebase 電子使用手冊  
<https://firebase.google.com/docs/database/ios/start>
- [9] MVC 架構是什麼？  
<https://tw.alphacamp.co/blog/mvc-model-view-controller>
- [10] 使用 Python 操作 Firebase 資料庫(CRUD)  
<https://gist.github.com/e96031413/5c0f6a7402fe07bc58de6b172218f954>
- [11] DroneKit-Python's documentation  
<https://dronekit-python.readthedocs.io/en/latest/>
- [12] ArduPilot  
<https://ardupilot.org/ardupilot/index.html>
- [13] Dronekit setup, simulation, and visualization on Mission Planner  
[https://www.youtube.com/watch?v=TFDWs\\_DG2QY&t=430s&ab\\_channel=TizianoFiorenzani](https://www.youtube.com/watch?v=TFDWs_DG2QY&t=430s&ab_channel=TizianoFiorenzani)



[14] Connect a Raspberry Pi to a Pixhawk running Ardupilot/PX4

[https://www.youtube.com/watch?v=cZVNndOaYCE&ab\\_channel=TizianoFiorenzani](https://www.youtube.com/watch?v=cZVNndOaYCE&ab_channel=TizianoFiorenzani)

[15] MissionPlanner

<https://ardupilot.org/planner/>

[16] [基礎] 以 VNC 和 Raspberry Pi 連線

<https://www.raspberrypi.com.tw/586/setting-up-vnc/>

[17] 遠端控制樹莓派

<https://ithelp.ithome.com.tw/articles/10235452>





# 專題名稱：Dronk——結合外送平台之無人機運輸

指導教授：賴瑾峰 專題生：俞杉麒 陳颺 王毅暄 郭紘碩 葉芷彤

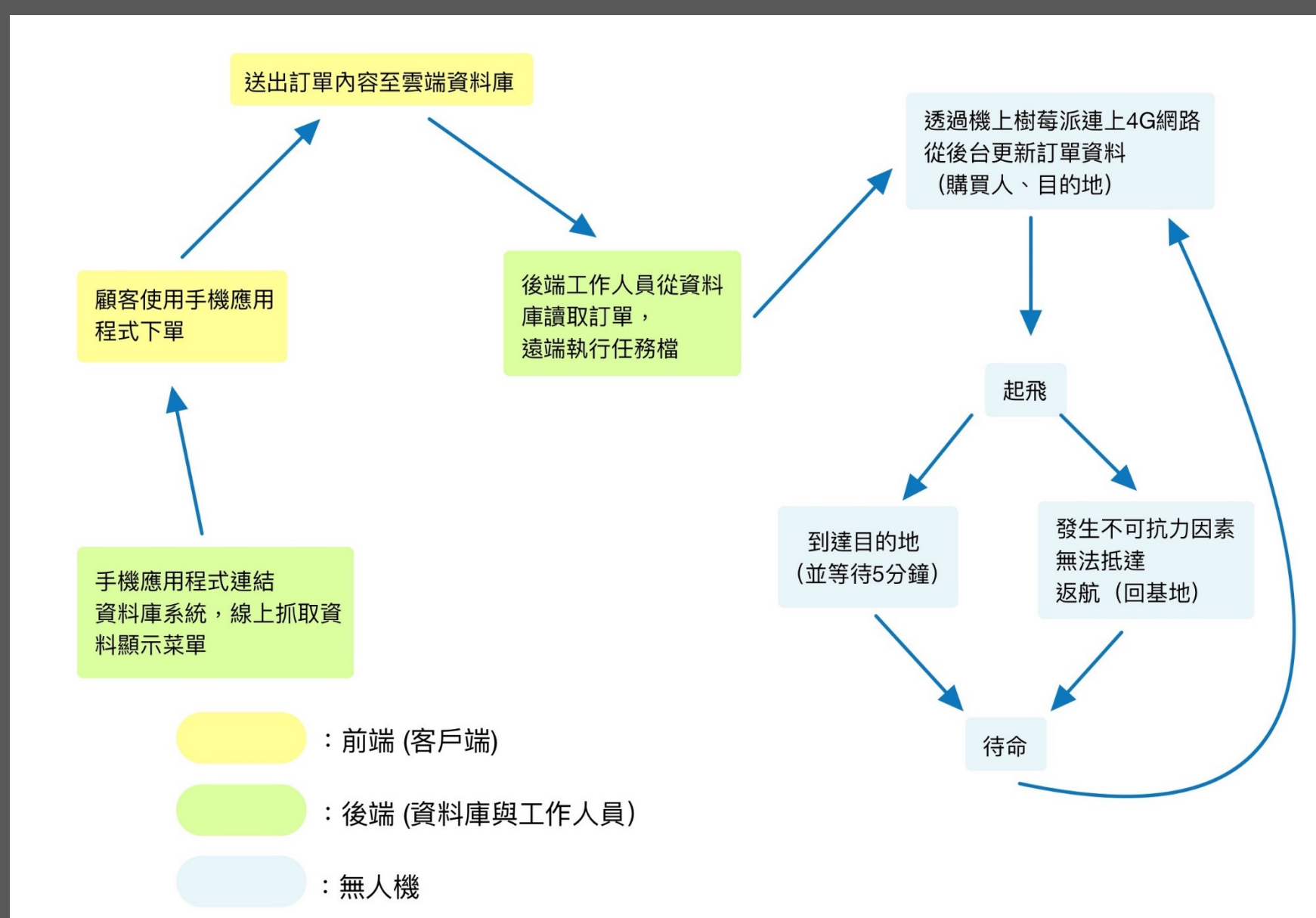


## 壹、研究動機與目標

近年來網路科技技術的普及使大眾的消費模式改變，線上購物已成為常態。2019年11月底爆發之新冠肺炎疫情，致使餐飲外送正式走入大眾線上購物的視野。由於臺灣地狹人稠，餐飲外送業多屬意機車做為其營業之載具。然而我們親身體驗過其服務後，發現經常因交通堵塞而出現訂單延宕，更有甚者，亦常於新聞媒體上得知外送業者因趕時間而於外送途中發生車禍，危害人身安全。我們便思考是否有外送方式能兼顧速度及安全，而近期興起的無人機便是解答之一。

因目的為出於餐飲需求的外送，我們於此專題將成為一飲料業者，欲利用無人機（Drone）外送自家所售之飲料（Drink），是為「DRONK」系統。

## 貳、系統架構



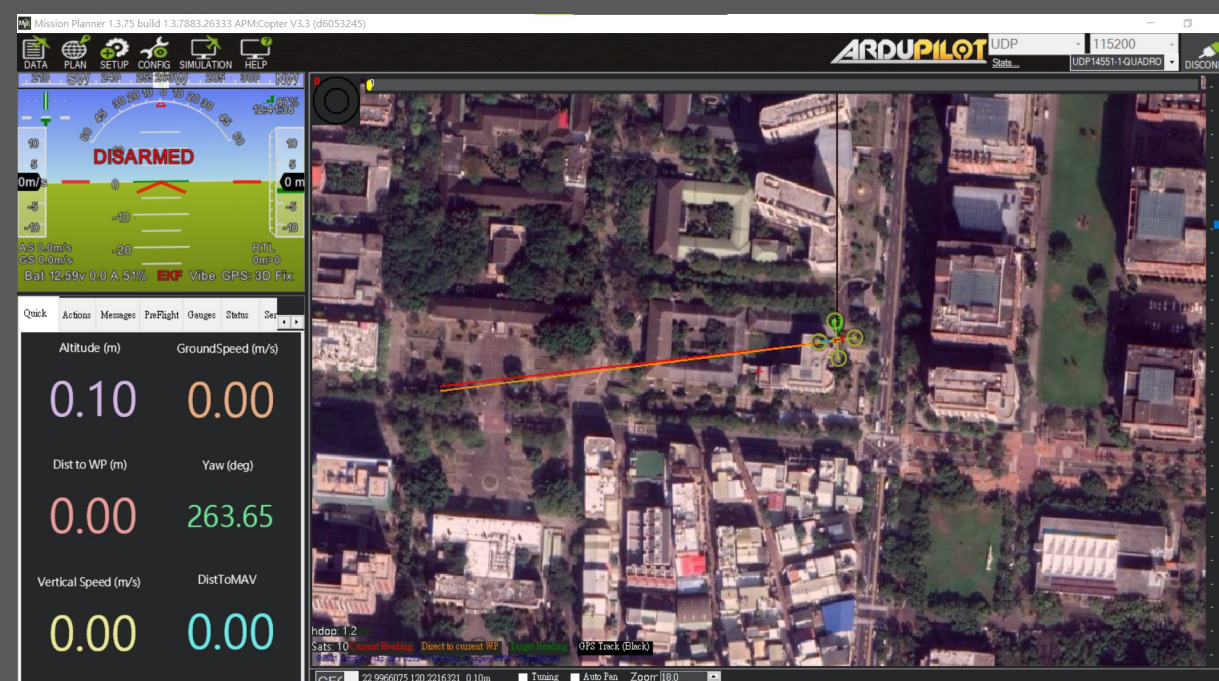
架設後台資料庫，建立資料庫內之動態資料，並設計手機應用程式，讓客戶下訂單。訂單生成後使其傳送資料至無人機上的樹莓派，無人機接到訂單資料後便會飛往指定地點送貨，確認收到後再折返回基地。

## 參、無人機



包含無人機本體、PIXHAWK4飛控板、Raspberry Pi 3、華為E3372h 4G行動網卡。

## 無人機系統架構



利用 MissionPlanner 實時監控無人機狀態，並以4G行動網卡於樹莓派上模擬乙太網路，使無人機於任何地方

皆有網路訊號與後端進行交流，更貼近實際使用情境，亦便於監控無人機狀態。

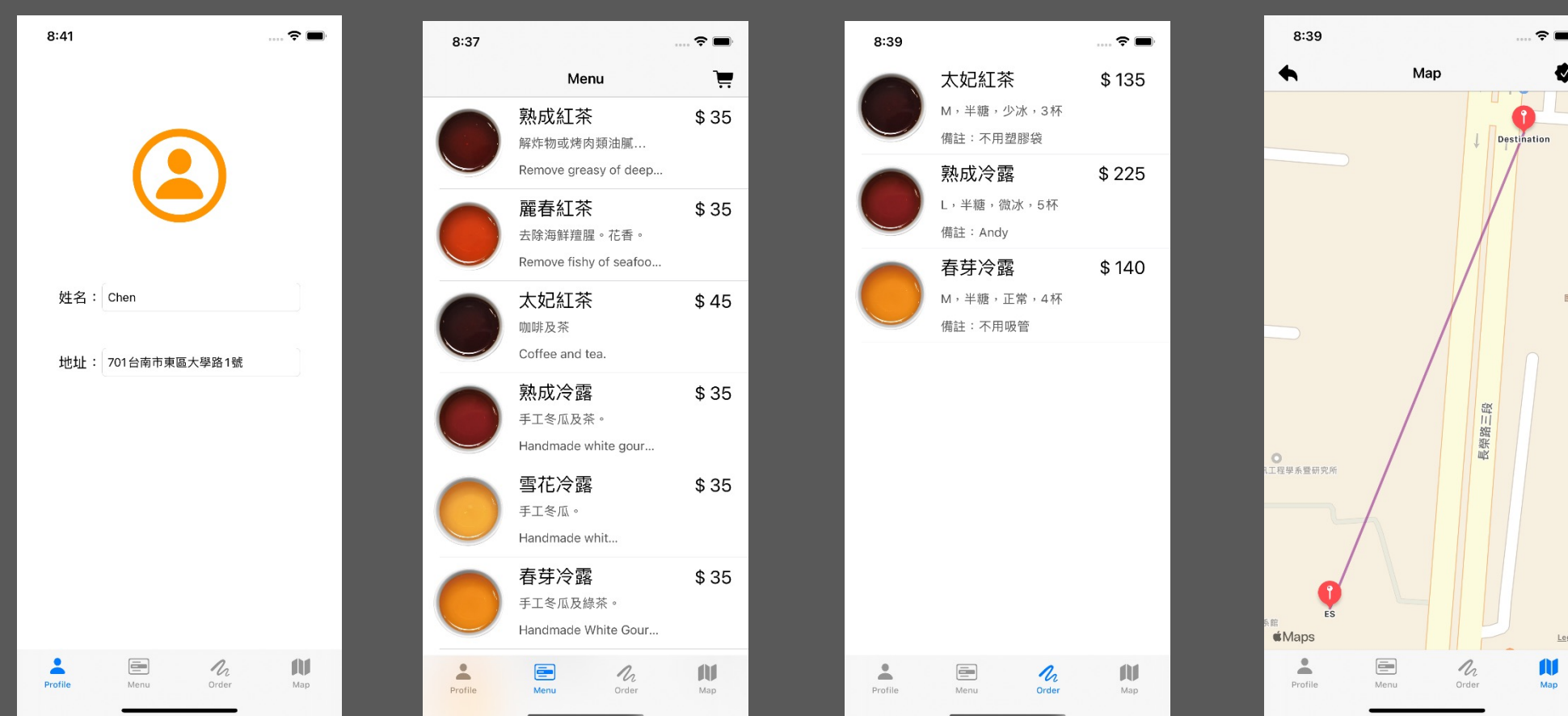
透過 Dronekit 函式庫撰寫無人機任務程式，使其符合飛行任務的要求，當有新訂單時，後端工作人員就會執行程式使其飛往指定地點。

## 肆、外送平台與應用程式

### 1. 應用程式架構

此應用程式使用了MVC架構（Model-View-Controller 軟體設計模式），因為在應用程式開發中，非常重視關注點分離（separation of concerns, SOC）此一設計原則，以便於App各種不同部分的分工。

### 2. App



### 3. 資料庫



分為Menu及訂單資料庫：商家可由Menu資料庫後台隨時動態調整Menu資訊；訂單資料庫可看到完整訂單資訊(包含座標)，

當有待命之無人機時，備好訂單內容後即可讓無人機載貨飛往目的地。

## 伍、未來展望

目前無人機已成功與手機app結合，可以接收及完成0接觸送飲料任務，未來可以以此為基礎增加更多實用的功能，例如實現單趟多筆訂單、在app上顯示無人機的飛行資訊、安裝鏡頭藉由人臉、手勢辨識增加安全性等等，以提供更全面的外送服務。