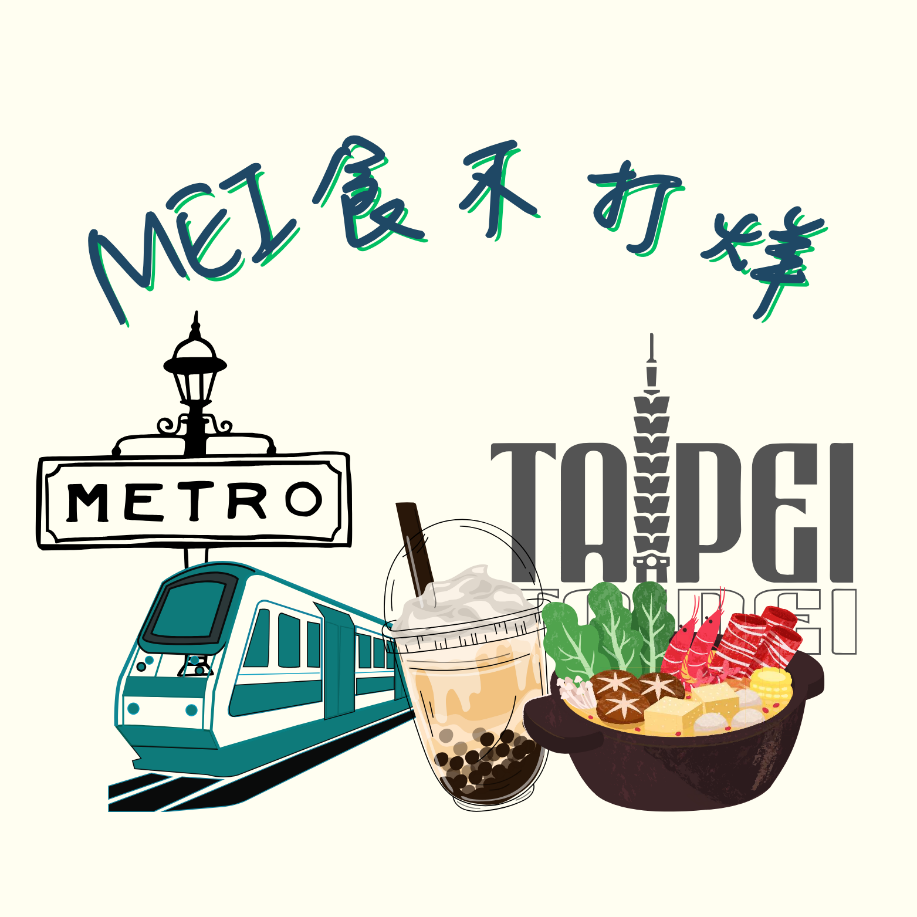
國立臺北商業大學

資 訊 管 理 系

114，資訊系統專案設計

**系統手冊**

****

**組別：第114407組**

**題目：MEI食不打烊**

**指導老師：葉明貴系主任**

**組長：11146061 郭宥妍**

**組員：11146066 簡孝宇 11146073 張楷偉  
11146083 林勝威**

**中華民國114年10月08日**

**目錄**

[第1章　前言 1](#_Toc209171541)

[1-1 背景介紹 1](#_Toc209171542)

[1-2 動6A5F 1](#_Toc209171543)

[1-3 系統目的與目標 2](#_Toc209171544)

[1-4 預期成果 2](#_Toc209171545)

[第2章　營運計畫 3](#_Toc209171546)

[2-1 可行性分析 3](#_Toc209171547)

[2-2 商業模式－Business model 4](#_Toc209171548)

[2-3 市場分析－STP 5](#_Toc209171549)

[2-4 競爭力分析SWOT-TOWS 7](#_Toc209171550)

[第3章　系統規格 9](#_Toc209171551)

[3-1 系統架構 9](#_Toc209171552)

[3-2 系統軟、硬體需求與技術平台 11](#_Toc209171553)

[3-3 使用標準與工具 11](#_Toc209171554)

[第4章　專案時程與組織分工 12](#_Toc209171555)

[4-1 專案時程 12](#_Toc209171556)

[4-2 專案組織與分工 13](#_Toc209171557)

[4-3 GitHub紀錄 15](#_Toc209171558)

[第5章　需求模型 16](#_Toc209171559)

[5-1 使用者需求 16](#_Toc209171560)

[5-2 使用個案圖(Use case diagram) 17](#_Toc209171561)

[5-3 使用個案描述 17](#_Toc209171562)

[5-4 分析類別圖(Analysis class diagram) 20](#_Toc209171563)

[第6章　設計模型 21](#_Toc209171564)

[6-1 循序圖(Sequential diagram) 21](#_Toc209171565)

[6-2 設計類別圖(Design class diagram) 23](#_Toc209171566)

[第7章　實作模型 24](#_Toc209171567)

[7-1 佈署圖(Deployment diagram) 24](#_Toc209171568)

[7-2 套件圖(Package diagram) 24](#_Toc209171569)

[7-3 元件圖(Component diagram) 26](#_Toc209171570)

[7-4 狀態機(State machine) 27](#_Toc209171571)

[第8章　資料庫設計 31](#_Toc209171572)

[8-1 資料庫關聯表 31](#_Toc209171573)

[8-2 表格及其Meta data 32](#_Toc209171574)

[第9章　程式 39](#_Toc209171575)

[9-1 元件清單及其規格描述 39](#_Toc209171576)

[9-2 其他附屬之各種元件 41](#_Toc209171577)

[第10章　測試模型 42](#_Toc209171578)

[10-1 測試計畫 42](#_Toc209171579)

[10-2 測試個案與測試結果 43](#_Toc209171580)

[第11章　操作手冊 45](#_Toc209171581)

[第12章　使用手冊 46](#_Toc209171582)

[第13章　感想 47](#_Toc209171583)

[第14章　參考資料 49](#_Toc209171584)

[附錄 50](#_Toc209171585)

**圖目錄**

[▲圖 3-1‑1：系統架構圖 9](#_Toc210668591)

[▲圖 3-1‑2：系統流程圖 10](#_Toc210668592)

[▲圖 4-3‑1：上傳GitHub紀錄－11146061 郭宥妍 15](#_Toc210668593)

[▲圖 4-3‑2：上傳GitHub紀錄－11146066 簡孝宇 15](#_Toc210668594)

[▲圖 4-3‑3：上傳GitHub紀錄－11146073 張楷偉 15](#_Toc210668595)

[▲圖 4-3‑4：上傳GitHub紀錄－11146083 林勝威 15](#_Toc210668596)

[▲圖 5-2‑1：使用個案圖 17](#_Toc210668597)

[▲圖 5-4‑1：分析類別圖 20](#_Toc210668598)

[▲圖 6-1‑1：循序圖－登入/註冊 21](#_Toc210668599)

[▲圖 6-1‑2：循序圖－餐廳查詢 21](#_Toc210668600)

[▲圖 6-1‑3：循序圖－餐廳評論 22](#_Toc210668601)

[▲圖 6-1‑4：循序圖－捷運查詢 22](#_Toc210668602)

[▲圖 6-2‑1：設計類別圖 23](#_Toc210668603)

[▲圖 7-1‑1：佈署圖 24](#_Toc210668604)

[▲圖 7-2‑1：套件圖－登入/註冊模組 24](#_Toc210668605)

[▲圖 7-2‑2：套件圖－餐廳管理模組 25](#_Toc210668606)

[▲圖 7-2‑3：套件圖－評論模組 25](#_Toc210668607)

[▲圖 7-2‑4：套件圖－捷運/站點模組 25](#_Toc210668608)

[▲圖 7-2‑5：套件圖－用戶管理模組 26](#_Toc210668609)

[▲圖 7-2‑6：套件圖－首頁/重定向模組 26](#_Toc210668610)

[▲圖 7-3‑1：元件圖 26](#_Toc210668611)

[▲圖 7-4‑1：狀態機－登入/註冊流程 27](#_Toc210668612)

[▲圖 7-4‑2：狀態機－用戶（帳號啟用/停權/管理員） 28](#_Toc210668613)

[▲圖 7-4‑3：狀態機－餐廳營業 28](#_Toc210668614)

[▲圖 7-4‑4：狀態機－評論發佈流程 29](#_Toc210668615)

[▲圖 7-4‑5：狀態機－前端頁面/表單切換 29](#_Toc210668616)

[▲圖 7-4‑6：狀態機－首頁重定向 30](#_Toc210668617)

[▲圖 8-1‑1：MySQL資料庫ER圖 31](#_Toc210668618)

[▲圖 8-1‑2：資料庫關聯圖 32](#_Toc210668619)

**表目錄**

[▼表 2-3‑1：潛在使用者區分為三大族群 5](#_Toc210668663)

[▼表 2-3‑2：平台主打特色 6](#_Toc210668664)

[▼表 2-4‑1：SWOT分析 7](#_Toc210668665)

[▼表 2-4‑2：TOWS分析 8](#_Toc210668666)

[▼表 3-2‑1：系統軟、硬體需求與技術平台 11](#_Toc210668667)

[▼表 3-3‑1：系統環境 11](#_Toc210668668)

[▼表 3-3‑2：系統開發工具 11](#_Toc210668669)

[▼表 3-3‑3：程式開發技術 11](#_Toc210668670)

[▼表 3-3‑4：文件編寫工具 11](#_Toc210668671)

[▼表 3-3‑5：管理程式平台 11](#_Toc210668672)

[▼表 4-1‑1：專案時程時程表 12](#_Toc210668673)

[▼表 4-2‑1：專案組織與分工 13](#_Toc210668674)

[▼表 4-2‑2：工作內容 14](#_Toc210668675)

[▼表 5-1‑1：使用者需求表 16](#_Toc210668676)

[▼表 5-3‑1：登入 17](#_Toc210668677)

[▼表 5-3‑2：註冊 18](#_Toc210668678)

[▼表 5-3‑3：查詢餐廳 18](#_Toc210668679)

[▼表 5-3‑4：查看餐廳詳細資訊 19](#_Toc210668680)

[▼表 5-3‑5：發表評論/打卡 19](#_Toc210668681)

[▼表 8-2‑1：資料表 32](#_Toc210668682)

[▼表 8-2‑2：user 33](#_Toc210668683)

[▼表 8-2‑3：metroline 33](#_Toc210668684)

[▼表 8-2‑4：station 34](#_Toc210668685)

[▼表 8-2‑5：restaurant 34](#_Toc210668686)

[▼表 8-2‑6：businesshours 35](#_Toc210668687)

[▼表 8-2‑7：review 36](#_Toc210668688)

[▼表 8-2‑8：checkinreview 36](#_Toc210668689)

[▼表 8-2‑9：traininfo 37](#_Toc210668690)

[▼表 8-2‑10：stationid 37](#_Toc210668691)

[▼表 9-1‑1：元件清單與規格描述表(前端)－網頁介面撰寫 38](#_Toc210668692)

[▼表 9-1‑2：元件清單與規格描述表(前端)－網頁介面設計 38](#_Toc210668693)

[▼表 9-1‑3：元件清單與規格描述表(前端)－網頁動畫設計 39](#_Toc210668694)

[▼表 9-1‑4：元件清單與規格描述表(後端)－爬蟲 39](#_Toc210668695)

[▼表 9-1‑5：元件清單與規格描述表(後端)－讀取API 39](#_Toc210668696)

[▼表 9-1‑6：元件清單與規格描述表(後端)－Service 40](#_Toc210668697)

[▼表 9-1‑7：元件清單與規格描述表(後端)－定時更新資訊 40](#_Toc210668698)

[▼表 9-1‑8：元件清單與規格描述表(後端)－使用者資訊管理 40](#_Toc210668699)

[▼表 9-1‑9：元件清單與規格描述表(後端)－串接相關網頁 40](#_Toc210668700)

[▼表 10-2‑1：測試結果－會員註冊 43](#_Toc210668701)

[▼表 10-2‑2：測試結果－會員登入 43](#_Toc210668702)

[▼表 10-2‑3：測試結果－更改會員資料 43](#_Toc210668703)

[▼表 10-2‑4：測試結果－站點查詢 43](#_Toc210668704)

[▼表 10-2‑5：測試結果－夜貓子專區 44](#_Toc210668705)

# 第1章　前言

## 背景介紹

台北捷運系統作為都市運輸的重要骨幹，每日承載大量通勤與觀光人潮，其沿線站點涵蓋多個商圈與飲食聚落，具有高度的區域特色與美食潛力。而現有的美食資訊平台多數以行政區域或餐廳類型分類為主，缺乏以「捷運站」為導向的查詢邏輯，對依賴大眾運輸移動的使用者而言，難以快速掌握特定站點周邊的餐飲資訊。

其次，隨著社群媒體風潮興起，越來越多使用者習慣透過「打卡」、「拍照上傳」等方式分享美食與生活體驗，社群互動逐漸成為資訊擴散的重要來源。導入會員制度與互動任務機制，已成為提升平台活躍度與使用者參與的主流設計方向。

此外，夜間活動人口對於深夜餐飲選擇與交通回程規劃仍面臨資訊不足的問題，顯示夜間生活相關資訊整合的需求仍有成長空間。

## 動機

隨著智慧型手機與網路服務的快速發展，民眾在外出時越來越依賴行動裝置來查詢即時資訊，特別是在交通與美食領域的需求尤為顯著。台北捷運系統作為首都地區最重要的公共運輸網絡，每日串聯數百萬人次的通勤與觀光流動，其各站周邊聚集了大量在地美食資源。然而，目前市面上仍缺乏一個以「捷運站」為核心，整合交通、美食、社群服務的互動平台。

## 系統目的與目標

本系統旨在打造一個以台北捷運站為主軸的美食資訊平台，使用者可透過手機查詢各站周邊的推薦餐廳與深夜美食資訊。平台採用捷運站導向的查詢方式，提升資訊搜尋效率，並結合會員登入、拍照打卡、評論與任務等互動功能，強化社群參與與內容豐富度。此外，設置「夜貓子專區」，整合深夜營業店家與捷運末五班車時刻，協助夜間用戶安排行程。系統也導入推播功能，主動提醒捷運列車即將到站的時間，提升使用者的行動便利性。整體平台支援繁體中文語言，並採用響應式設計，確保行動裝置上的良好操作體驗，打造一個實用、友善且具互動性的美食導覽平台。

## 預期成果

我們預期系統可以以獨家設置「夜貓子專區」，整合深夜營業店家與末班車時刻，專為夜間行動者量身打造，貼心守護每一個夜晚的旅程。平台支援繁體中文與響應式設計，無論身處何地都能輕鬆操作，串聯每位饕客與捷運沿線美味。讓行動導覽與社群美食兼具，成就智慧城市的新型態餐飲探索體驗，進而達到「美味同行，捷運每一站都精彩！」。

# 第2章　營運計畫

## 可行性分析

* 技術可行性

1. 本系統採用成熟穩定的Web技術，包括HTML5、CSS、JavaScript、Fetch API資料處理與響應式設計，並搭配Django等後端框架進行資料串接與展示。
2. 使用者功能包含捷運站導向的美食查詢、夜貓子餐廳推薦、打卡評論與多語操作，皆為可實作功能，已分工執行中。

* 資料可行性

1. 我們實際向台北捷運官方申請了站點資料API，取得各站相關資訊與即時列車資訊作為查詢基礎。
2. 同時也向Google Places API正式申請金鑰，用以補足店家圖片、營業資訊、使用者評論等資料，確保內容即時且具參考價值。
3. 除了API資料，也可搭配必要的手動輸入測試資料與模擬使用者評論來建立測試案例與展示效果。

* 操作可行性

1. 系統導向捷運站點搜尋邏輯，搭配夜間資訊與推播提醒，符合真實情境需求。
2. 支援繁體中文語言，提供觀光友善環境，並優化手機使用體驗，易於推廣與實際使用。

* 實施可行性

1. 已完成API串接測試與資料顯示範例，分工包含前端介面設計、後端串接邏輯、資料處理與互動功能開發，具備落地實作能力。
2. 系統開發過程將持續依照專題進度規劃逐步整合並測試。

## 商業模式－Business model

* 關鍵夥伴（Key Partners）

1. 台北捷運公司（提供站點資訊API）
2. Google Places API（提供餐廳詳細資訊與圖片）
3. 開放資料平台（補足區域與地點資訊）

* 關鍵活動（Key Activities）

1. 建立捷運站導向的美食搜尋系統
2. 串接捷運與Google API資料
3. 設計互動功能（打卡、評論、任務）
4. 提供夜間美食與交通資訊整合
5. 響應式網站開發

* 價值主張（Value Proposition）

1. 台北第一個「以捷運站為中心」的美食與夜貓資訊平台
2. 解決夜間餐飲選擇難
3. 支援繁體中文與行動操作，觀光友善、通勤方便
4. 提供「打卡任務」、「夜間專區」、等創新互動功能

## 市場分析－STP

* 市場區隔Segmentation

依據使用者的行為特性與使用情境，可將潛在使用者區分為下列三大族群：

▼表 2-3‑1：潛在使用者區分為三大族群

| 區隔族群 | 特徵說明 |
| --- | --- |
| 通勤族群 | 以捷運作為每日上下班交通工具，對「午餐、晚餐」推薦有高度需求。使用時間集中於早上與傍晚。 |
| 夜間活動族 | 活動時間為夜間，常有深夜餐飲需求（如學生、夜班族、夜生活族群）。關心餐廳營業時間與回家交通。 |
| 國內自由行旅客 | 在台北捷運沿線移動，對地點不熟，習慣使用手機查詢。常有語言障礙，關心推薦、地圖、評論支援。 |

* 目標對象Targeting

平台的首波目標使用者為：

1. 以台北捷運為主要移動方式的通勤者與觀光客

* 因平台以「捷運站」為搜尋核心，特別適合此類使用者快速找到附近美食。

1. 對夜間生活有需求的年輕族群或上夜班者

* 夜間外食資訊缺乏，平台設置「夜貓子專區」與「末班車時刻」剛好填補空白。
* 產品定位Positioning

「全台第一個以捷運站為導向的多語美食地圖，深夜行動族群的最佳夥伴」

▼表 2-3‑2：平台主打特色

| 定位主軸 | 說明 |
| --- | --- |
| 捷運站導向 | 使用者以車站為起點查找周邊美食，符合實際移動邏輯。 |
| 夜貓子專區 | 整合深夜營業餐廳與末班車資訊，貼近夜間需求。 |
| 社群互動 | 提供拍照打卡、評論任務，提升參與感與內容豐富度。 |

## 競爭力分析SWOT-TOWS

▼表 2-4‑1：SWOT分析

|  |  |
| --- | --- |
| 優勢Strengths | 劣勢Weaknesses |
| * 以捷運站為導向，查詢邏輯清晰直覺 * 結合「夜間營業」與「班車提醒」，功能創新 * 提供打卡、評論、任務互動功能，增加使用體驗 * 系統介面採響應式設計，適用各種裝置操作 | * 初期資料仍須人工補充，規模有限 * 評論與使用者互動資料需時間累積 * 需仰賴外部API（捷運與Google）穩定度 * 系統需持續維護與更新，開發期工作量大 |
| 機會Opportunities | 威脅Threats |
| * 台北捷運系統龐大，站點周邊美食豐富 * 使用者對於夜間資訊與在地推薦有實際需求 * 行動裝置使用普及，適合推動即時查詢功能 * 開放資料與公共API提供資料串接支援 | * 類似平台（如Google Maps、美食社群App）功能重疊 * 多語內容需人工處理，語言品質需控管 * API數據可能改版或停止提供，存在依賴風險 * 資料更新頻率與正確性會影響使用者信任 * 多語內容維護成本高，翻譯準確度需控制 |

▼表 2-4‑2：TOWS分析

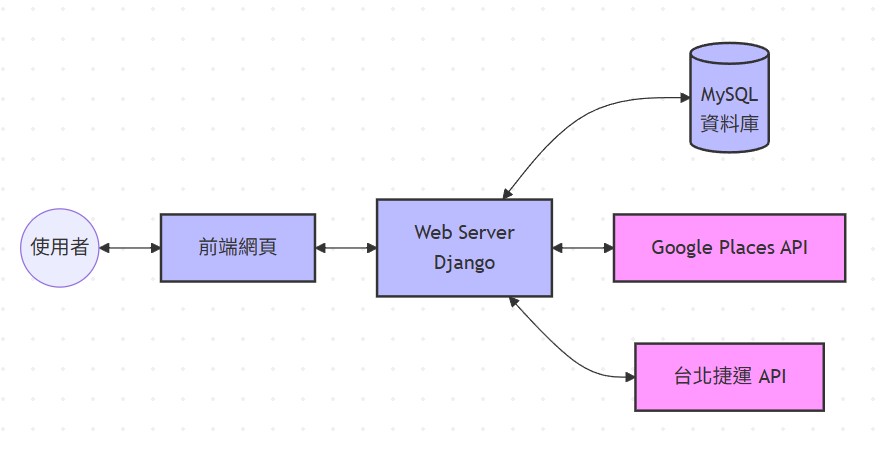
|  |  |  |
| --- | --- | --- |
| 內部因素  外在因素 | 優勢Strengths | 劣勢Weaknesses |
| 機會Opportunities | SO策略（增長性策略）   * 利用捷運導向與夜間資訊整合特色，解決使用者移動與用餐交會問題 * 運用中文介面設計，提升對國內觀光族的實用性與友善度 * 採模組化架構，未來可擴展至其他縣市捷運系統 | WO策略（移轉性策略）   * 初期聚焦重點站點，逐步擴充內容並建立測試資料 * 結合API與手動輸入方式補足資料來源，提高資料覆蓋率 * 強化錯誤回報與使用回饋機制，逐步優化功能 |
| 威脅Threats | ST策略（多角化策略）   * 強調系統特色（捷運導向＋班車提醒＋夜間美食）與現有平台做出區隔 * 建立API備援處理流程，減少依賴性造成的風險 * 設計簡潔、操作直觀，提升易用性以留住使用者 | WT策略（防禦型策略）   * 提供基礎版本測試範圍，確保資料正確與功能完整後再逐步擴展 * 強化多語翻譯管理，方便維護 * 規劃資料更新流程與開發紀錄，方便未來交接或維護 |

綜合SWOT與TOWS分析可見，本系統具備以「捷運站」為導向的查詢優勢，搭配夜間美食資訊、末班車時刻支援，成功區隔於現有以地區或餐廳類型為主的美食平台，展現明確的功能特色與實用價值。

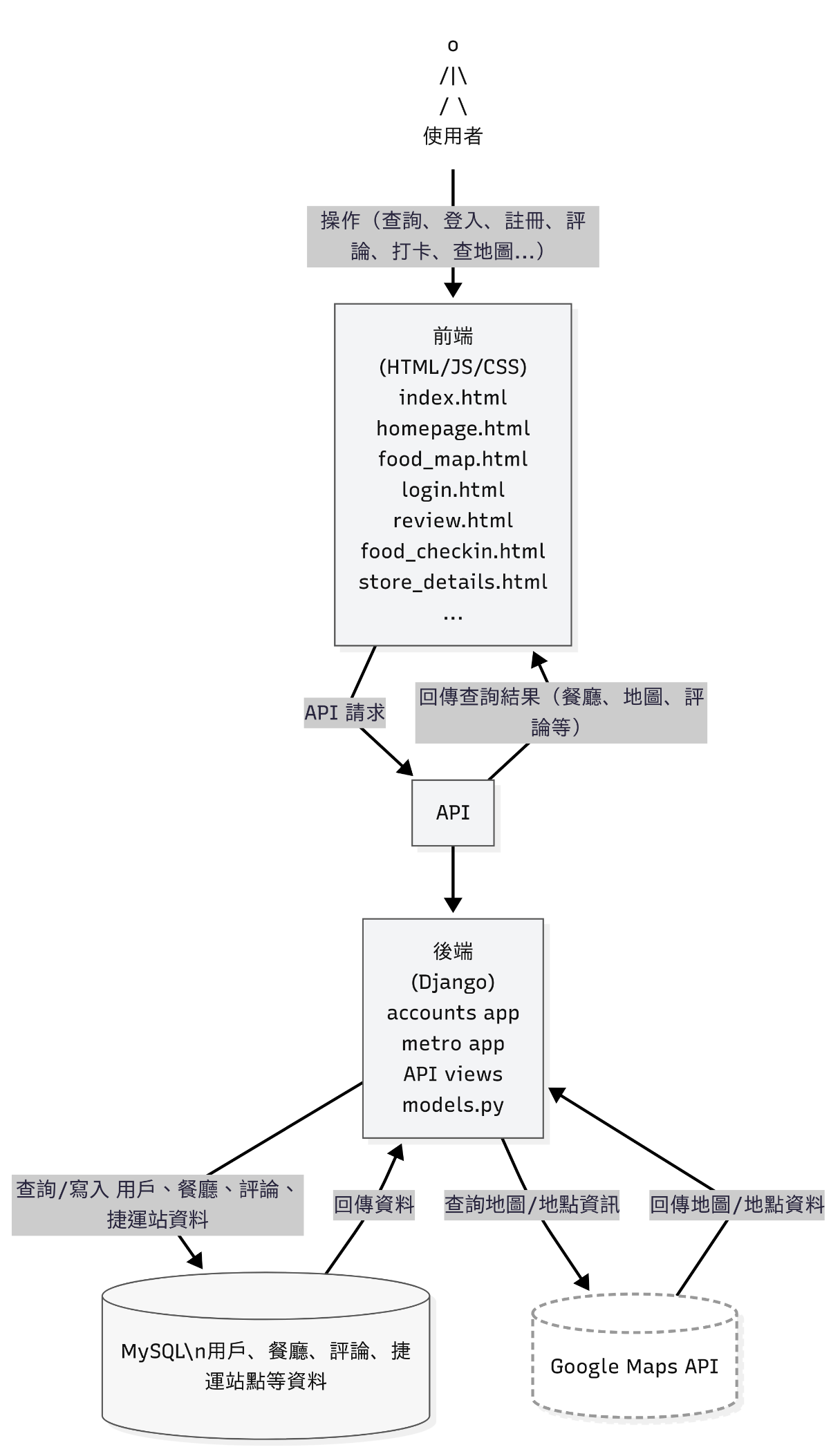
然而，系統初期仍面臨資料覆蓋率不足、API依賴度高與語言維護成本等挑戰，須透過階段性擴充、錯誤回報機制與多語翻譯策略逐步克服。透過模組化設計與漸進式開發，可有效提升平台穩定性與可維護性，並為未來推展至其他縣市或新增功能保留彈性。

# 第3章　系統規格

## 系統架構



▲圖 3-1‑1：系統架構圖



▲圖 3-1‑2：系統流程圖

## 系統軟、硬體需求與技術平台

▼表 3-2‑1：系統軟、硬體需求與技術平台

|  |  |
| --- | --- |
| 設備 | 規格 |
| 手機裝置 | |
| 作業系統 | iOS、Android |
| 版本 | iOS全系列、Android 8以上 |
| 操作介面 | Safari、Chrome |

## 使用標準與工具

▼表 3-3‑1：系統環境

|  |  |
| --- | --- |
| 作業系統 | Windows 11 |
| 網路需求 | WiFi/有線網路 |

▼表 3-3‑2：系統開發工具

|  |  |
| --- | --- |
| 網頁設計 | Visual Studio Code |
| 資料庫 | MySQL |
| 伺服器工具 | Django |

▼表 3-3‑3：程式開發技術

|  |  |
| --- | --- |
| 前端技術 | HTML、CSS、JavaScript |
| 後端技術 | Python、Django |
| API串接 | RESTful API、Google Places API、台北捷運API |

▼表 3-3‑4：文件編寫工具

|  |  |
| --- | --- |
| 繪圖工具 | Canva、Flaticon |
| 簡報製作 | Microsoft PowerPoint 2016 |
| 文件製作 | Microsoft Word 2016 |

▼表 3-3‑5：管理程式平台

|  |  |
| --- | --- |
| 專案管理 | github、fork |

# 第4章　專案時程與組織分工

## 專案時程

▼表 4-1‑1：專案時程時程表

預期進度　實際進度

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1月 | | | 2月 | | | 3月 | | | 4月 | | | 5月 | | | 6月 | | | 7月 | | | 8月 | | | 9月 | | | 10月 | | | 11月 | | | 12月 | | |
| 主題構思 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 相關資料蒐集 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統功能分析 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統模型 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| UI/UX設計 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Logo設計 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Web開發 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 資料庫設計 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 資料庫建立 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| API設定 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 後端開發 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統測試 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統整合 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 初評系統手冊 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 複評系統手冊 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統簡報 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## 專案組織與分工

▼表 4-2‑1：專案組織與分工

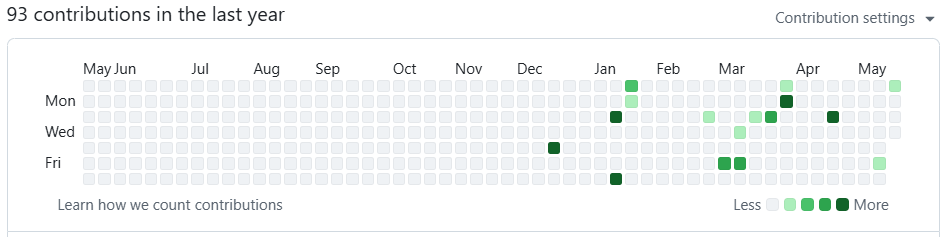
●主要負責人 ○次要負責人

| 組　員  項　目 | | 11146061 郭宥妍 | 11146066 簡孝宇 | 11146073 張楷偉 | 11146083 林勝威 |
| --- | --- | --- | --- | --- | --- |
| 後端開發 | 資料庫建置 |  |  |  | ● |
| 伺服器架設 |  |  |  | ● |
| API設定 |  |  |  | ● |
| 捷運資料轉換 |  |  |  | ● |
| 前端開發 | 登入/註冊 | ○ |  | ○ | ● |
| 首頁 | ○ | ○ | ● |  |
| 夜貓子專區 | ○ | ○ | ● |  |
| 美食打卡 | ○ | ○ | ● |  |
| 捷運時刻表 |  | ○ | ● |  |
| 美食地圖 |  | ○ | ● |  |
| 通知 |  | ○ | ● |  |
| 會員中心 |  | ○ | ● |  |
| Javascript程式 |  | ● |  |  |
| 美術設計 | UI / UX | ● |  |  |  |
| Web介面設計 | ● |  | ○ |  |
| 色彩設計 | ● |  |  |  |
| Logo設計 | ● |  |  |  |
| 素材設計 | ● |  |  |  |
| 文件撰寫 | 統整 | ● |  |  |  |
| 第1章 前言 | ● |  |  |  |
| 第2章 營運計畫 |  | ● |  |  |
| 第3章 系統規格 |  |  |  | ● |
| 第4章 專題時程與組織分工 | ● |  |  |  |
| 第5章 需求模型 |  |  |  | ● |
| 第6章 設計模型 |  |  |  | ● |
| 第7章 實作模型 |  |  |  | ● |
| 第8章 資料庫設計 |  |  |  | ● |
| 第9章 程式 | ○ | ○ |  | ● |
| 第10章 測試模型 | ● |  |  |  |
| 第11章 操作手冊 | ● |  |  |  |
| 第12章 使用手冊 | ● |  |  |  |
| 報告 | 簡報製作 |  | ● |  |  |

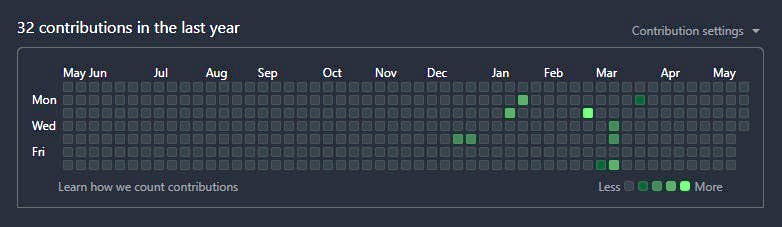
▼表 4-2‑2：工作內容

|  |  |  |  |
| --- | --- | --- | --- |
| 序號 | 姓名 | 工作內容<各限100字以內> | 貢獻度 |
| 1 | 組長  郭宥妍 | 文件撰寫、文件排版、文件統整、Logo設計、UI/UX設計 | 25% |
| 2 | 組員  張楷偉 | 前端網頁設計、前端開發 | 20% |
| 3 | 組員  簡孝宇 | 前端動畫開發、串接API、簡報製作、文件撰寫 | 25% |
| 4 | 組員  林勝威 | 前後端串接、後端開發、資料庫建置、伺服器架設、API設定 | 30% |
|  | | | 總計：100% |

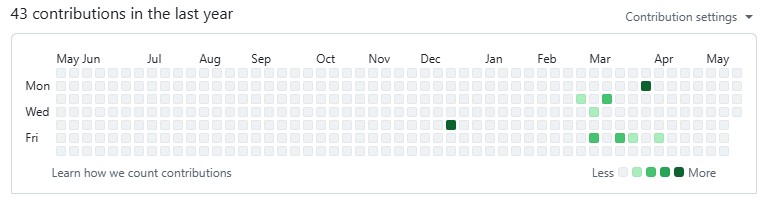
## GitHub紀錄



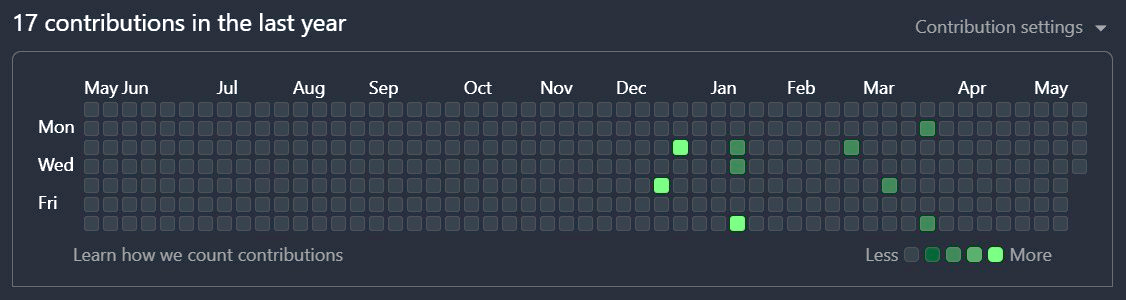
▲圖 4-3‑1：上傳GitHub紀錄－11146061 郭宥妍



▲圖 4-3‑2：上傳GitHub紀錄－11146066 簡孝宇



▲圖 4-3‑3：上傳GitHub紀錄－11146073 張楷偉



▲圖 4-3‑4：上傳GitHub紀錄－11146083 林勝威

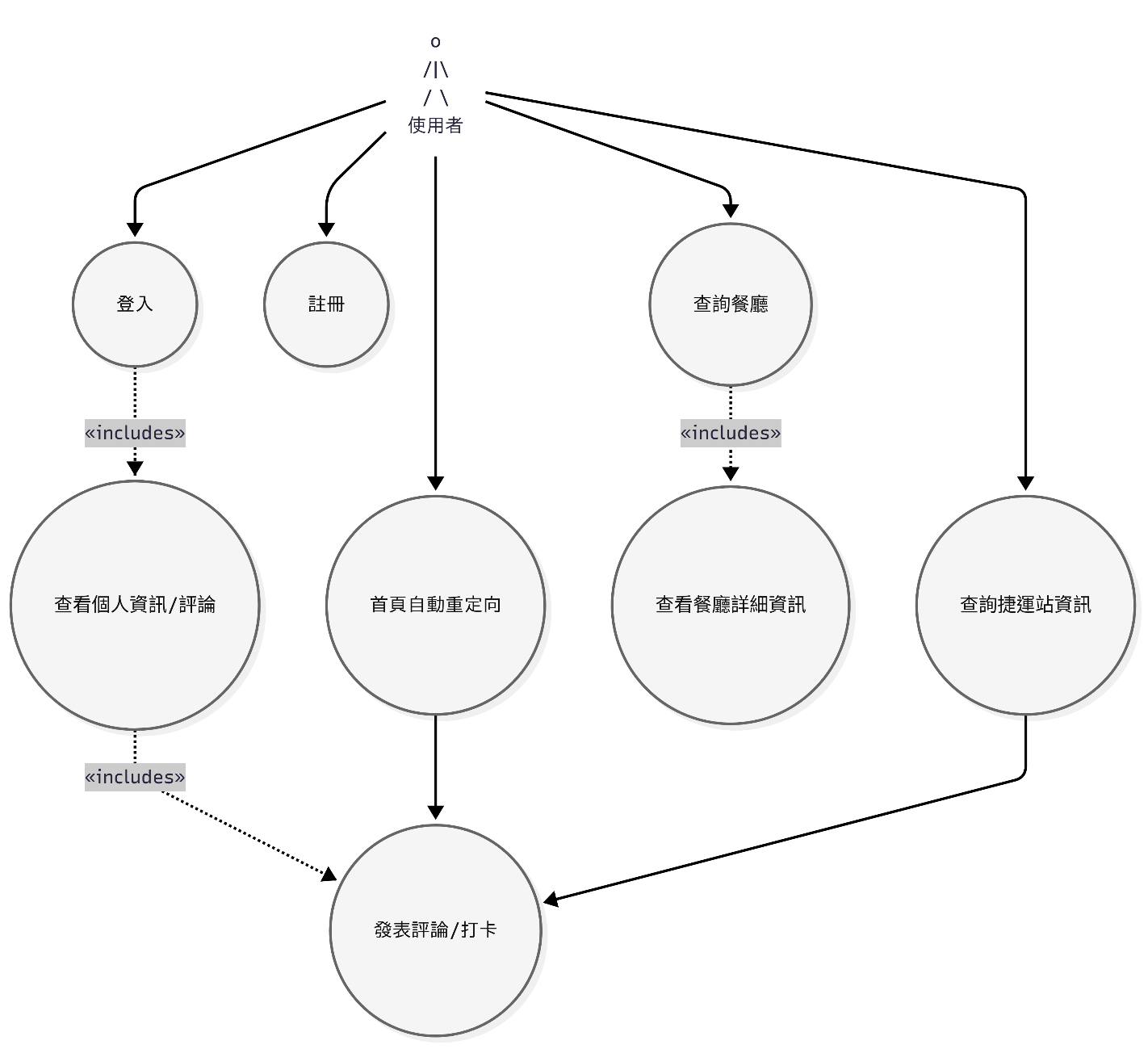
# 第5章　需求模型

## 使用者需求

▼表 5-1‑1：使用者需求表

|  |  |
| --- | --- |
| UseCase | 需求說明 |
| 登入與註冊 | 使用者能夠透過電子郵件註冊帳號並登入系統，進行個人化操作。 |
| 餐廳/捷運站查詢 | 使用者可以透過捷運站、餐廳名稱或地圖查詢附近的餐廳，快速找到想要的美食。 |
| 查看餐廳詳細資訊 | 使用者能夠查看餐廳的詳細資料（名稱、地址、評分、分類、營業時間、評論等）。 |
| 發表評論/打卡 | 使用者可以針對餐廳發表評論、評分，或進行美食打卡，分享用餐體驗。 |
| 查詢捷運站資訊 | 使用者可以查詢捷運站的基本資訊、所屬路線及附近的餐廳。 |
| 查看個人資訊/評論 | 使用者可以在個人頁面查看自己的基本資料、發表過的評論紀錄。 |
| 首頁自動重定向 | 使用者進入首頁時，系統會自動判斷登入狀態並導向對應頁面（已登入導向主頁，未登入導向登入頁）。 |

## 使用個案圖(Use case diagram)



▲圖 5-2‑1：使用個案圖

## 使用個案描述

▼表 5-3‑1：登入

|  |  |  |
| --- | --- | --- |
| 使用個案名稱：登入 | | |
| 主要參與者 | 使用者 | |
| 前置條件 | 使用者已註冊帳號 | |
| 事件路徑： | | |
| 流程 | | 系統回應 |
| 1.使用者輸入信箱與密碼 | |  |
| 2.按下登入 | |  |
| 3.系統驗證帳密 | | 顯示登入成功或失敗訊息 |
| 4.登入成功導向首頁 | |  |

▼表 5-3‑2：註冊

|  |  |  |
| --- | --- | --- |
| 使用個案名稱：註冊 | | |
| 主要參與者 | 使用者 | |
| 前置條件 | 無（新用戶） | |
| 事件路徑： | | |
| 流程 | | 系統回應 |
| 1.使用者輸入信箱、密碼、確認密碼 | |  |
| 2.按下註冊 | |  |
| 3.系統驗證資料 | | 顯示註冊成功或失敗訊息 |
| 4.註冊成功導向登入頁 | |  |

▼表 5-3‑3：查詢餐廳

|  |  |  |
| --- | --- | --- |
| 使用個案名稱：查詢餐廳 | | |
| 主要參與者 | 使用者 | |
| 前置條件 | 無 | |
| 事件路徑： | | |
| 流程 | | 系統回應 |
| 1.用者於首頁、地圖頁、夜市頁等選擇捷運站或輸入關鍵字 | |  |
| 2.系統顯示符合條件的餐廳列表或地圖標記 | |  |
| 3.點擊餐廳可查看詳細資訊 | | 顯示餐廳列表、地圖標記與詳細資訊 |

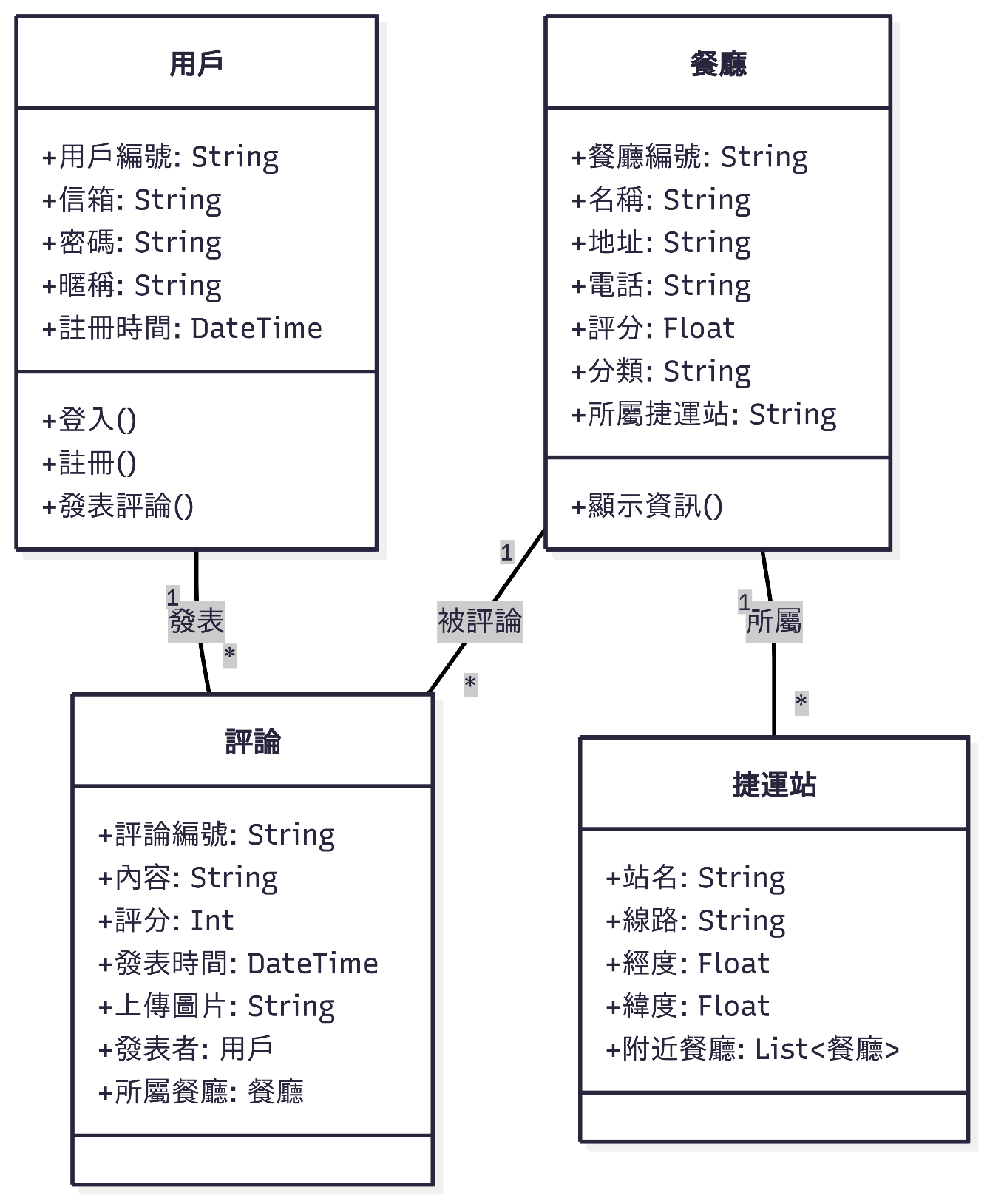
▼表 5-3‑4：查看餐廳詳細資訊

|  |  |  |
| --- | --- | --- |
| 使用個案名稱：查看餐廳詳細資訊 | | |
| 主要參與者 | 使用者 | |
| 前置條件 | 無 | |
| 事件路徑： | | |
| 流程 | | 系統回應 |
| 1.使用者於餐廳列表或地圖點擊餐廳 | |  |
| 2.系統顯示餐廳詳細頁 | | 顯示餐廳詳細資訊 |

▼表 5-3‑5：發表評論/打卡

|  |  |  |
| --- | --- | --- |
| 使用個案名稱：發表評論/打卡 | | |
| 主要參與者 | 使用者 | |
| 前置條件 | 使用者已登入 | |
| 事件路徑： | | |
| 流程 | | 系統回應 |
| 1.進入餐廳頁面或打卡頁 | |  |
| 2.輸入評論內容、評分 | |  |
| 3.按下送出 | | 顯示評論/打卡成功或失敗訊息 |
| 4.系統儲存評論/打卡資料 | |  |

## 分析類別圖(Analysis class diagram)



▲圖 5-4‑1：分析類別圖

# 第6章　設計模型

## 循序圖(Sequential diagram)

一張含有 文字, 螢幕擷取畫面, 圖表, 字型 的圖片

AI 產生的內容可能不正確。

▲圖 6-1‑1：循序圖－登入/註冊

一張含有 文字, 螢幕擷取畫面, 圖表, 字型 的圖片

AI 產生的內容可能不正確。

▲圖 6-1‑2：循序圖－餐廳查詢

一張含有 文字, 螢幕擷取畫面, 圖表, 字型 的圖片

AI 產生的內容可能不正確。

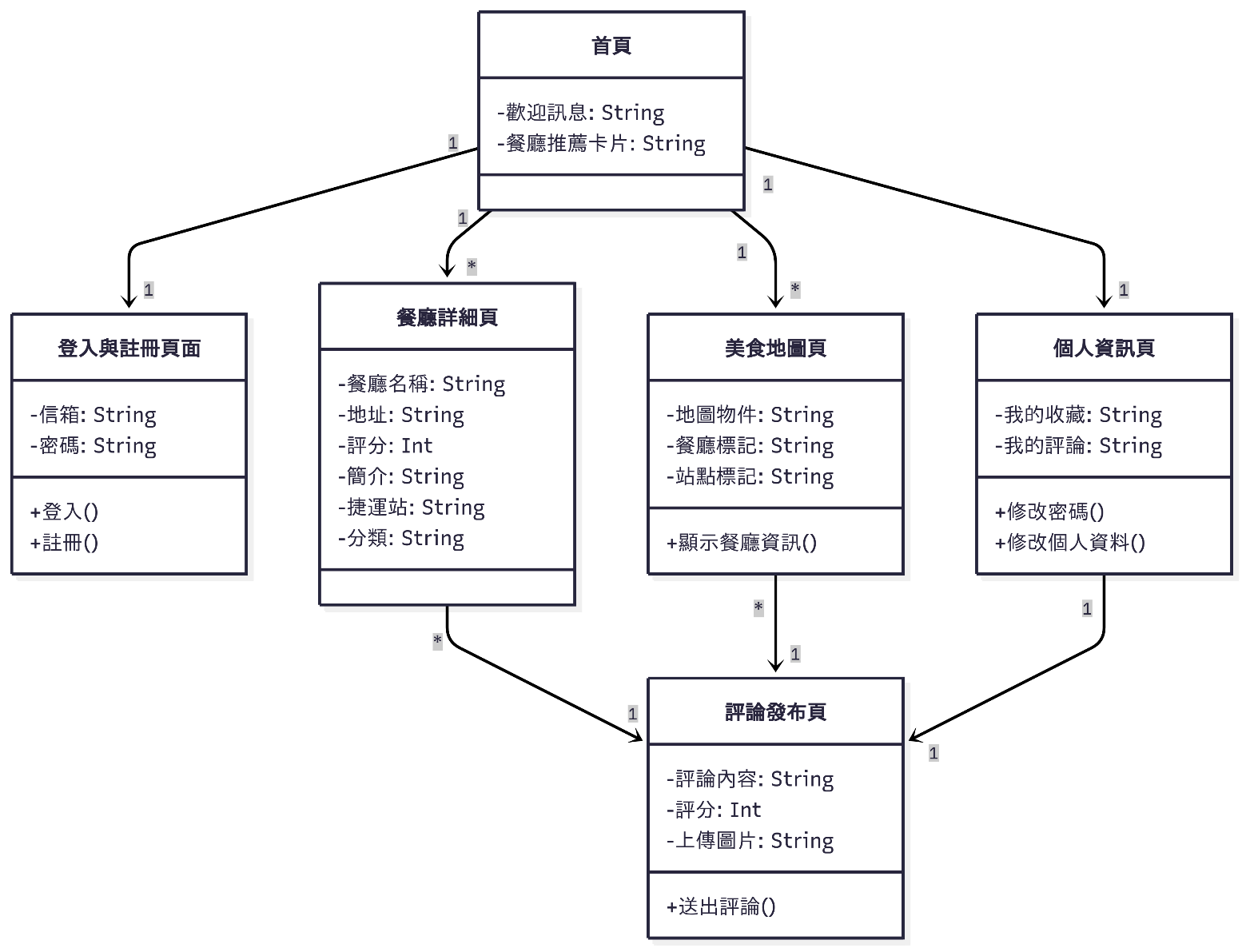
▲圖 6-1‑3：循序圖－餐廳評論

一張含有 文字, 螢幕擷取畫面, 圖表, 字型 的圖片

AI 產生的內容可能不正確。

▲圖 6-1‑4：循序圖－捷運查詢

## 設計類別圖(Design class diagram)



▲圖 6-2‑1：設計類別圖

# 第7章　實作模型

## 佈署圖(Deployment diagram)

一張含有 文字, 圖表, 螢幕擷取畫面, 方案 的圖片

AI 產生的內容可能不正確。

▲圖 7-1‑1：佈署圖

## 套件圖(Package diagram)

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

▲圖 7-2‑1：套件圖－登入/註冊模組

一張含有 文字, 螢幕擷取畫面, 字型, Rectangle 的圖片

AI 產生的內容可能不正確。

▲圖 7-2‑2：套件圖－餐廳管理模組

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

▲圖 7-2‑3：套件圖－評論模組

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

▲圖 7-2‑4：套件圖－捷運/站點模組

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

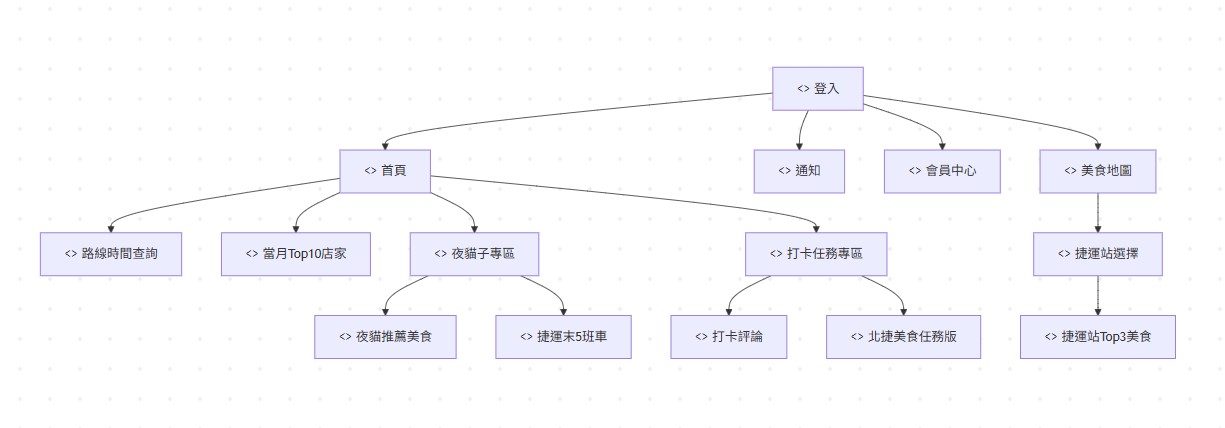
▲圖 7-2‑5：套件圖－用戶管理模組

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

▲圖 7-2‑6：套件圖－首頁/重定向模組

## 元件圖(Component diagram)



▲圖 7-3‑1：元件圖

## 狀態機(State machine)

一張含有 圖表, 文字, 方案, 寫生 的圖片

AI 產生的內容可能不正確。

▲圖 7-4‑1：狀態機－登入/註冊流程

一張含有 圖表, 文字, 圖畫, 行 的圖片

AI 產生的內容可能不正確。

▲圖 7-4‑2：狀態機－用戶（帳號啟用/停權/管理員）

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

▲圖 7-4‑3：狀態機－餐廳營業

一張含有 文字, 圖表, 寫生, 圖畫 的圖片

AI 產生的內容可能不正確。

▲圖 7-4‑4：狀態機－評論發佈流程

**一張含有 文字, 字型, 圖表, 設計 的圖片

AI 產生的內容可能不正確。**

▲圖 7-4‑5：狀態機－前端頁面/表單切換

**一張含有 文字, 字型, 螢幕擷取畫面, 圖表 的圖片

AI 產生的內容可能不正確。**

▲圖 7-4‑6：狀態機－首頁重定向

# 第8章　資料庫設計

## 資料庫關聯表



▲圖 8-1‑1：MySQL資料庫ER圖

一張含有 文字, 螢幕擷取畫面, 圖表, 行 的圖片

AI 產生的內容可能不正確。

▲圖 8-1‑2：資料庫關聯圖

## 表格及其Meta data

▼表 8-2‑1：資料表

|  |  |  |
| --- | --- | --- |
| 編號 | 資料表英文名稱 | 中文名稱/用途 |
|  | user | 使用者資料 |
|  | metroline | 捷運線路 |
|  | station | 捷運站 |
|  | restaurant | 餐廳 |
|  | businesshours | 餐廳營業時間 |
|  | review | 餐廳評論 |
|  | checkinreview | 打卡評論 |
|  | traininfo | 列車到站資訊 |
|  | stationid | 車站ID對照表 |

1. user（使用者資料）

▼表 8-2‑2：user

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 中文名稱 | 使用者資料 | | 資料表編號 | | 01 |
| 英文名稱 | user | | 主索引 | | id |
| 資料檔陳述 | 紀錄每一位使用者的個人資料及帳號、密碼 | | | | |
| 欄位名稱 | 型態 | 長度/限制 | | 說明 | |
| id | AutoField |  | | 主鍵，自動遞增 | |
| password | CharField | 128 | | 密碼（加密儲存） | |
| last\_login | DateTime | 可為空 | | 上次登入時間 | |
| is\_superuser | BooleanField |  | | 是否為超級管理員 | |
| username | CharField | 150,唯一 | | 使用者名稱 | |
| first\_name | CharField | 150 | | 名字 | |
| last\_name | CharField | 150 | | 姓氏 | |
| email | EmailField | 唯一 | | 電子郵件 | |
| is\_staff | BooleanField |  | | 是否為後台管理員 | |
| is\_active | BooleanField |  | | 帳號是否啟用 | |
| date\_joined | DateTime |  | | 註冊時間 | |
| created\_at | DateTime |  | | 創建時間（你自訂的欄位） | |
| updated\_at | DateTime |  | | 更新時間（你自訂的欄位） | |
| groups | ManyToMany |  | | 所屬群組（權限用） | |
| user\_permissions | ManyToMany |  | | 個別權限 | |

1. metroline（捷運路線）

▼表 8-2‑3：metroline

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 中文名稱 | 捷運路線 | 資料表編號 | | 02 | |
| 英文名稱 | metroline | 主索引 | | id | |
| 資料檔陳述 | 儲存台北捷運的各路線 | | | | |
| 欄位名稱 | 型態 | | 長度/限制 | | 說明 |
| id | AutoField | |  | | 主鍵 |
| name | CharField | | 100,唯一 | | 捷運線名稱 |
| color | CharField | | 20 | | 捷運線顏色 |
| created\_at | DateTime | |  | | 創建時間 |
| updated\_at | DateTime | |  | | 更新時間 |

1. station（捷運站）

▼表 8-2‑4：station

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 捷運站 | 資料表編號 | 03 |
| 英文名稱 | station | 主索引 | id |
| 資料檔陳述 | 儲存台北捷運的各站站名 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| name | CharField | 100 | 站名 |
| metro\_line\_id | ForeignKey | MetroLine | 所屬捷運線 |
| latitude | DecimalField | 15,12 | 緯度 |
| longitude | DecimalField | 15,12 | 經度 |
| station\_code | CharField | 10 | 站點代碼 |
| created\_at | DateTime |  | 創建時間 |
| updated\_at | DateTime |  | 更新時間 |

1. restaurant（餐廳）

▼表 8-2‑5：restaurant

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 中文名稱 | 餐廳 | | 資料表編號 | | 04 |
| 英文名稱 | restaurant | | 主索引 | | id |
| 資料檔陳述 | 儲存餐廳名稱及詳細資料 | | | | |
| 欄位名稱 | 型態 | 長度/限制 | | 說明 | |
| id | AutoField |  | | 主鍵 | |
| name | CharField | 200 | | 店家名稱 | |
| address | CharField | 500 | | 地址 | |
| latitude | DecimalField | 15,12 | | 緯度 | |
| longitude | DecimalField | 15,12 | | 經度 | |
| phone | CharField | 20,可為空 | | 電話 | |
| google\_place\_id | CharField | 100,唯一 | | GooglePlacesID | |
| rating | DecimalField | 3,1,可為空 | | Google評分 | |
| price\_level | IntegerField | 可為空 | | 價格等級 | |
| website | URLField | 可為空 | | 網站 | |
| image | CharField | 500,可為空 | | 照片參考 | |
| created\_at | DateTime |  | | 創建時間 | |
| updated\_at | DateTime |  | | 更新時間 | |
| last\_review\_update | DateTime | 可為空 | | 最後評論更新時間 | |
| ... | ManyToMany | Station,FoodCategory | | 關聯捷運站/美食分類 | |

1. businesshours（餐廳營業時間）

▼表 8-2‑6：businesshours

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 餐廳營業時間 | 資料表編號 | 05 |
| 英文名稱 | businesshours | 主索引 | id |
| 資料檔陳述 | 儲存餐廳營業資訊 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| restaurant\_id | ForeignKey | Restaurant | 餐廳 |
| day\_of\_week | IntegerField |  | 星期幾 |
| open\_time | TimeField |  | 開店時間 |
| close\_time | TimeField |  | 關店時間 |
| is\_closed | BooleanField |  | 是否休息 |
| created\_at | DateTime |  | 創建時間 |
| updated\_at | DateTime |  | 更新時間 |

1. review（餐廳評論）

▼表 8-2‑7：review

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 餐廳評論 | 資料表編號 | 06 |
| 英文名稱 | review | 主索引 | id |
| 資料檔陳述 | 儲存Google的評論 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| restaurant\_id | ForeignKey | Restaurant | 餐廳 |
| user\_id | ForeignKey | User,可為空 | 使用者 |
| reviewer\_name | CharField | 100,可為空 | 評論者名稱 |
| rating | IntegerField |  | 評分 |
| comment | TextField | 可為空 | 評論內容 |
| created\_at | DateTime |  | 創建時間 |
| updated\_at | DateTime |  | 更新時間 |

1. checkinreview（打卡評論）

▼表 8-2‑8：checkinreview

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 打卡評論 | 資料表編號 | 07 |
| 英文名稱 | checkinreview | 主索引 | id |
| 資料檔陳述 | 儲存打卡任務的評論內容 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| restaurant\_name | CharField | 100 | 餐廳名稱 |
| metro\_line | CharField | 50 | 捷運線 |
| reviewer\_name | CharField | 50,預設匿名 | 評論者名稱 |
| rating | IntegerField |  | 評分 |
| comment | TextField | 可為空 | 評論內容 |
| created\_at | DateTime |  | 創建時間 |
| updated\_at | DateTime |  | 更新時間 |

1. traininfo（列車到站資訊）

▼表 8-2‑9：traininfo

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 列車到站資訊 | 資料表編號 | 08 |
| 英文名稱 | traininfo | 主索引 | id |
| 資料檔陳述 | 儲存捷運詳細到站資訊 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| train\_number | CharField | 50,可為空 | 列車號碼 |
| station\_name | CharField | 100 | 站名 |
| destination\_name | CharField | 100 | 目的地 |
| count\_down | CharField | 50 | 倒數時間 |
| now\_date\_time | DateTime |  | 當前時間 |
| created\_at | DateTime |  | 創建時間 |

1. stationid（車站ID對照表）

▼表 8-2‑10：stationid

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名稱 | 車站ID對照表 | 資料表編號 | 09 |
| 英文名稱 | stationid | 主索引 | id |
| 資料檔陳述 | 儲存捷運路線名稱及站點資訊 | | |
| 欄位名稱 | 型態 | 長度/限制 | 說明 |
| id | AutoField |  | 主鍵 |
| station\_name | CharField | 100 | 車站名稱 |
| line\_name | CharField | 100 | 線路名稱 |
| station\_id | CharField | 20,唯一 | 車站ID |

# 第9章　程式

## 元件清單及其規格描述

▼表 9-1‑1：元件清單與規格描述表(前端)－網頁介面撰寫

|  |  |  |
| --- | --- | --- |
| HTML－網頁介面撰寫 | | |
| 編號 | 檔案名稱 | 功能 |
| 1-1-1 | login.html | 註冊及登入 |
| 1-1-2 | homepage.html | 系統首頁 |
| 1-1-3 | night.html | 夜貓子專區 |
| 1-1-4 | drinks.html | 喝酒專區 |
| 1-1-5 | bars.html | 酒吧&夜店 |
| 1-1-6 | food\_checkin.html | 打卡任務 |
| 1-1-7 | food\_mission.html | 美食任務版 |
| 1-1-8 | store\_details.html | 美食任務版－店家詳細資訊 |
| 1-1-9 | food\_map.html | 美食地圖 |
| 1-1-10 | notify.html | 通知 |
| 1-1-11 | Member\_center.html | 會員中心 |
| 1-1-12 | edit\_profile.html | 修改會員資料 |
| 1-1-13 | favorites\_food.html | 收藏美食 |

▼表 9-1‑2：元件清單與規格描述表(前端)－網頁介面設計

|  |  |  |
| --- | --- | --- |
| CSS－網頁介面設計 | | |
| 編號 | 檔案名稱 | 功能 |
| 1-2-1 | login.css | 註冊及登入 |
| 1-2-2 | homepage.css | 系統首頁 |
| 1-2-3 | night.css | 夜貓子專區 |
| 1-2-4 | bars.css | 酒吧&夜店 |
| 1-2-5 | food\_checkin.css | 打卡任務 |
| 1-2-6 | food\_mission.css | 美食任務版 |
| 1-2-7 | store\_details.css | 美食任務版－店家詳細資訊 |
| 1-2-8 | food\_map.css | 美食地圖 |
| 1-2-9 | notify.css | 通知 |
| 1-2-10 | Member\_center.css | 會員中心 |
| 1-2-11 | edit\_profile.css | 修改會員資料 |
| 1-2-12 | favorites\_food.css | 收藏美食 |

▼表 9-1‑3：元件清單與規格描述表(前端)－網頁動畫設計

|  |  |  |
| --- | --- | --- |
| JavaScript－網頁動畫設計 | | |
| 編號 | 檔案名稱 | 功能 |
| 1-3-1 |  |  |
| 1-3-2 |  |  |
| 1-3-3 |  |  |
| 1-3-4 |  |  |
| 1-3-5 |  |  |
| 1-3-6 |  |  |

▼表 9-1‑4：元件清單與規格描述表(後端)－爬蟲

|  |  |  |
| --- | --- | --- |
| Python－爬蟲 | | |
| 編號 | 檔案名稱 | 功能 |
| 2-1-1 | import\_restaurants.py | 爬取餐廳資訊 |
| 2-1-2 | google\_maps\_reviews\_crawler.py | 爬取餐廳評論 |

▼表 9-1‑5：元件清單與規格描述表(後端)－讀取API

|  |  |  |
| --- | --- | --- |
| Python－讀取API | | |
| 編號 | 檔案名稱 | 功能 |
| 2-2-1 | urls.py | 定義捷運、餐廳、評論、酒吧、通知等API路由對應至views |
| 2-2-2 | views.py | 提供主要API，包含餐廳查詢、評論CRUD、打卡評論、捷運、路線、酒吧、通知等 |
| 2-2-3 | models.py | 定義資料模型，包含Restaurant、Review、CheckinReview、MetroFirstLastTrain等 |
| 2-2-4 | serializers.py | 序列化模型資料（部分API使用） |

▼表 9-1‑6：元件清單與規格描述表(後端)－Service

|  |  |  |
| --- | --- | --- |
| Python－Service | | |
| 編號 | 檔案名稱 | 功能 |
| 2-3-1 | review\_service.py | 建立餐廳、打卡評論、回算餐廳評分、同步Google Places評分與評論 |
| 2-3-2 | metro\_train\_service.py | 以TDX OAuth2取token，同步台北、新北首末班車資料並寫入資料庫 |

▼表 9-1‑7：元件清單與規格描述表(後端)－定時更新資訊

|  |  |  |
| --- | --- | --- |
| Python－定時更新資訊 | | |
| 編號 | 檔案名稱 | 功能 |
| 2-4-1 | update\_reviews.py | 批次更新Google Places餐廳評論與評分 |
| 2-4-2 | tasks.py | 背景任務、排程（若啟用APScheduler/Crontab） |

▼表 9-1‑8：元件清單與規格描述表(後端)－使用者資訊管理

|  |  |  |
| --- | --- | --- |
| Python－使用者資訊管理 | | |
| 編號 | 檔案名稱 | 功能 |
| 2-5-1 | urls.py | 帳號相關路由（登入、註冊、個資） |
| 2-5-2 | views.py | 帳號API，包含登入、註冊、取得使用者資訊 |
| 2-5-3 | models.py | 自訂使用者欄位（暱稱、生日、頭像等） |

▼表 9-1‑9：元件清單與規格描述表(後端)－串接相關網頁

|  |  |  |
| --- | --- | --- |
| Python－使用者資訊管理 | | |
| 編號 | 檔案名稱 | 功能 |
| 2-6-1 | urls.py | 專案全域路由入口，掛載各app的urls |
| 2-6-2 | settings.py | 專案設定：資料庫、APP、CORS、REST Framework、金鑰等 |

## 其他附屬之各種元件

▼表 9-2‑1：虛擬碼－homepage.html、homepage.css

|  |  |  |  |
| --- | --- | --- | --- |
| 編號 | 1-1-2、1-2-2 | 程式名稱 | homepage.html、homepage.css |
| 目的 | 建立首頁頁面，提供捷運站搜尋、美食地圖導向、熱門店家顯示，以及RWD響應式版面設計。 | | |
| 虛擬碼 | | | |
| 1. 頁面基本結構設定  * 採用HTML 5標準結構，語系設定為zh-TW * head區： * 設定UTF-8編碼與viewport響應式設定 * 引入favicon（圖片/logo.png）、CSS樣式表static/css/homepage.css * 載入Awesomplete外部套件（自動完成搜尋）  1. 主要區塊說明  * header區塊 * 顯示網站Logo與標題「MEI食不打烊Taipei Metro」 * 使用.title-with-logo排版，支援窄螢幕自動換行 * 搜尋列 * 兩個輸入框（出發站、目的站）＋「交換」按鈕＋「查詢」按鈕 * 使用Awesomplete套件實現捷運站自動完成功能 * 採用.search-bar容器，支援RWD * 功能按鈕區 * 包含三個主要功能按鈕：夜貓子專區、打卡任務、捷運時刻表 * 使用.features容器，按鈕有icon與文字說明 * TOP 10店家區 * 使用.top-shops容器顯示當月熱門店家 * .shop-list為橫向可滑動區域 * .shop-item顯示單一店家資訊，包含圖片與文字 * 固定底部導覽列 * 使用<nav class="footer">固定在頁面底部 * 提供「首頁、美食地圖、通知、會員中心」4個導覽按鈕 * 自動根據網址高亮目前頁面  1. CSS設定重點  * RWD響應式版心 * 使用:root { --app-max: ... }控制版心寬度 * 根據螢幕寬度分別調整手機、平板、桌機版心最大寬度 * 全域樣式 * 使用box-sizing: border-box;統一計算方式 * body採用置中排列，背景為淺灰色 * 功能按鈕區 * display: flex; justify-content: space-around; * 小螢幕時按鈕允許換行 * TOP 10店家 * 使用橫向scroll-snap技術，使滑動更順暢 * 移除scrollbar以提升外觀 * Footer導覽列 * 使用position: fixed;固定在畫面底部 * 寬度與版心同步，支援iPhone safe area * 導覽按鈕在當前頁面時會加上.active樣式變白底  1. JavaScript功能  * 監聽DOMContentLoaded事件，自動為當前頁面對應的導覽按鈕加上active 類別 * 使用Awesomplete載入stationdata.js提供站名資料 * 載入Google Maps Places API（key已設定）  1. 頁面行為流程  * 使用者進入首頁 → 顯示標題、搜尋區、功能區 * 使用者輸入出發站、目的站 → Awesomplete自動完成捷運站名 * 點擊「查詢」按鈕 → 顯示查詢結果區域 * 下方固定導覽列可切換其他功能頁面（地圖、通知、會員中心） | | | |

▼表 9-2‑1：虛擬碼－metro\_train\_service.py

|  |  |  |  |
| --- | --- | --- | --- |
| 編號 | 2-3-2 | 程式名稱 | metro\_train\_service.py |
| 目的 | 以TDX OAuth2取token，同步台北、新北首末班車資料並寫入資料庫 | | |
| 虛擬碼 | | | |
| 1. API 認證（TDX OAuth2）  * get\_access\_token： * token\_url＝TDX認證端點 * headers.Content-Type＝application/x-www-form-urlencoded * data＝{ grant\_type=client\_credentials, client\_id/secret ← settings } * resp＝POST(token\_url, headers, data) * resp.status !＝200 → log.error並回傳None * 回傳resp.json().access\_token  1. 抓取、合併資料  * fetch\_metro\_first\_last\_trains： * token ← get\_access\_token(); 若None則結束 * headers.Authorization＝Bearer token * trtc＝GET(TRTC\_URL, headers).json() * ntmc＝GET(NTMC\_URL, headers).json() * data\_all＝merge(trtc, ntmc)  1. 寫入資料庫  * 逐筆item in data\_all： * 解析鍵值：line\_no、line\_id、station\_id、destination\_station\_id、train\_type * 轉時間：FirstTrainTime/LastTrainTime → TimeField；SrcUpdateTime → DateTime * update\_or\_create(唯一鍵, defaults) * 計數updated\_count++；例外 → 記錄並跳過 | | | |

# 第10章　測試模型

## 測試計畫

本系統所需的硬體為瀏覽器iOS全系列及Chrome 140.0.7339以上的版本。在系統程式開發設計完成後，我們將程式存在虛擬機後端後於手機上網頁進行測試，確認每項功能是否能夠執行成功。

* 1. 會員註冊：確認使用者能夠註冊。
  2. 會員登入：確認使用者能夠登入到首頁。
  3. 更改會員資料：修改使用者儲存的最新個人資料。
  4. 站點查詢：查詢站點之間的路線資訊。
  5. 夜貓子專區：夜間使用者以捷運站名尋找美食及捷運末班車資訊。
  6. 喝酒專區：酒吧和酒店營業詳細資訊及代駕服務。
  7. 打卡任務評論：透過打卡任務來蒐集使用者的評論內容。
  8. 美食任務版：查詢所有店家的評論及營業詳細資訊。
  9. 捷運時刻表：提供北捷各站點詳細到站時間。
  10. 美食地圖：以捷運站名查詢各站周邊美食資訊。
  11. 收藏及通知：收藏美食後會通知收藏的相關訊息。

## 測試個案與測試結果

▼表 10-2‑1：測試結果－會員註冊

|  |  |  |  |
| --- | --- | --- | --- |
| 功能編號 | (1) | 功能名稱 | 會員註冊 |
| 測試目標 | 確認使用者能夠註冊 | | |
| 測試作業 | 1. 輸入信箱與密碼 2. 驗證資料庫是否有重複的會員資料 3. 若無重複的會員資料則直接註冊該筆會員資料 | | |
| 測試結果 | 成功 | | |

▼表 10-2‑2：測試結果－會員登入

|  |  |  |  |
| --- | --- | --- | --- |
| 功能編號 | (2) | 功能名稱 | 會員登入 |
| 測試目標 | 確認使用者能夠登入到首頁 | | |
| 測試作業 | 1. 輸入信箱與密碼 2. 驗證資料庫是否有重複的會員資料 3. 若查無符合的會員資料無法登入，需先註冊帳號 | | |
| 測試結果 | 成功 | | |

▼表 10-2‑3：測試結果－更改會員資料

|  |  |  |  |
| --- | --- | --- | --- |
| 功能編號 | (3) | 功能名稱 | 更改會員資料 |
| 測試目標 | 修改使用者儲存的最新個人資料 | | |
| 測試作業 | 1. 點選【會員中心】頁面中的編輯資訊 2. 是否能夠更改會員資料並儲存 3. 再次查看會員資料是否為更改過的資料 | | |
| 測試結果 | 成功 | | |

▼表 10-2‑4：測試結果－站點查詢

|  |  |  |  |
| --- | --- | --- | --- |
| 功能編號 | (4) | 功能名稱 | 站點查詢 |
| 測試目標 | 查詢站點之間的路線資訊 | | |
| 測試作業 | 1. 點選【首頁】頁面 2. 站點選單點選後按下查詢 3. 查看站點與路線的結果 | | |
| 測試結果 | 成功 | | |

▼表 10-2‑5：測試結果－夜貓子專區

|  |  |  |  |
| --- | --- | --- | --- |
| 功能編號 | (5) | 功能名稱 | 夜貓子專區 |
| 測試目標 | 夜間使用者以捷運站名尋找美食及捷運末班車資訊 | | |
| 測試作業 | 1. 點選【夜貓子專區】頁面 2. 搜尋欄點選欲查詢站名 3. 查看站點周邊夜間美食詳細資訊 4. 選擇欲查詢的站名末五班車 5. 查看末五班車資訊 | | |
| 測試結果 | 成功 | | |

# 第11章　操作手冊

## 系統元件

▼表 11-1‑1：系統之元件資訊

|  |  |
| --- | --- |
| 元件名稱 | MEI食不打烊 |
| 費用 | 免費 |
| 版本需求 | iOS全系列、Android 8以上 |
| 網路需求 | Wifi / 4G以上網路 |
| 支援語言 | 繁體中文 |

# 第12章　使用手冊

# 第13章　感想

* 11146061 郭宥妍

在本次專題之前，我曾在高中參與過類似大學專題的模式，當時同樣擔任組長，負責統籌並帶領團隊進行系統的開發與製作，整體規模皆比照大學專題進行。這段經驗讓我在面對此次專題時更加從容，自信地面對各種流程與挑戰，不僅不會感到緊張，也能協助組員更快熟悉進度，讓整體製作過程更加順利與高效。與以往不同的是，這次我首次接觸UI/UX設計。在剛開始使用Figma進行介面設計時，由於系統介面全為英文，加上毫無使用經驗，讓我一開始完全不知從何著手。為了克服這個障礙，我積極查閱教學資源與範例，並透過不斷的練習與嘗試，逐漸熟悉操作方式，也對設計的流程與原則有了初步的理解，這是我過去未曾接觸的領域，帶給我全新的挑戰與收穫。

此外，我在團隊領導與溝通協調方面也有明顯成長。我學會了如何更有效地分配工作內容，妥善安排會議時間，並在與老師討論時清楚表達進度與問題，同時也能處理各種突發的瑣碎事項。這次專題不僅提升了我的技術與設計能力，也讓我在組織、溝通與領導方面更加成熟，為未來面對更大專案奠定了穩固的基礎。

* 11146066 簡孝宇
* 11146073 張楷偉

在這次的專題製作過程中，我學到了許多課堂上接觸不到的實作經驗。從一開始構思主題、規劃架構，到實際撰寫前端畫面與資料呈現的過程中，一路上也遇到很多的難關，例如在設計網頁版面時，要兼顧排版、美觀與響應式設計（RWD），常常一個小細節就得反覆修改；在處理資料與互動功能時，也要考慮使用者體驗與操作流程，這些都讓我更加了解網頁開發的細節與難處。

過程中也經歷過不少挫折，像是版面跑版、功能不如預期、檔案路徑錯誤等問題，常常需要花許多時間除錯。不過透過不斷嘗試、查資料與請教同學，逐漸把問題一一解決。看到網頁從草稿一步步變成可實際操作的成品，心中充滿成就感。這次專題不僅讓我提升了網頁設計的能力，也讓我學會面對困難時保持耐心、解決問題的態度，這將成為我未來學習與成長的重要養分。

* 11146083 林勝威

本次專題負責資料庫、後端、虛擬機的整合，在資料庫的部分，前期建置商家資料有遇到許多問題，像是要如何完整的取出商家的留言，原本想說使用google place api就可以將留言都抓下，後來發現是可以抓留言，但一次只有五筆的資料，後來將爬蟲技能用上來，才將大筆大筆的資料抓下，最後帶入到資料庫當中，後端在建置api時也有許多的困難，要先預想前端會需要那些資料，怎麼樣把資料傳遞會速度較快，當前端接收有問題時要幫忙解決哪裡出現了bug，虛擬機的部份為了要一直update餐廳頻論，所以有做了一些檔案讓虛擬機也能更新資料，本地端移到虛擬機上又會是新的考驗。

# 第14章　參考資料

1. 餐廳評論爬蟲<https://ithelp.ithome.com.tw/m/articles/10267131>
2. 圖片存放位置<https://ithelp.ithome.com.tw/articles/10330627?sc=pt>
3. 北捷api <https://www.metro.taipei/cp.aspx?n=BDEB860F2BE3E249>
4. Django <https://ithelp.ithome.com.tw/articles/10206355>
5. 批次檔定期更新 <https://ithelp.ithome.com.tw/articles/10369742>
6. Django資料庫 <https://ithelp.ithome.com.tw/m/articles/10301639>

# 附錄

## 評審問題回覆

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 宜補上定時撈取餐廳資料的介紹。 | 我們有在虛擬機設置一天一次抓取餐廳的資料，並且在每天凌晨3點都會更新。 |

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 檢索排序納入餐廳打烊時間、走路距離、人潮、留言的考量。 | 我們有新增餐廳從幾點營業到打烊時間，以及利用Google map計算從捷運站步行至餐廳的時間，排序方式為步行時間近到遠排序。 |

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 「美食打卡」可導入Google評論資料。 |  |

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 結合代駕服務、深夜便利商店及喝酒專區，強化夜貓子功能。 | 我們有新增「喝酒專區」，此專區從夜貓子專區進入，喝酒專區內有酒吧和夜店的店家資訊及代駕專區，代駕專區我們直接連結到Uber的頁面，另外也有加上捷運末五班車的資訊在喝酒專區，可讓用戶不需再跳轉到上一頁搜尋捷運末班車。 |

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 站點搜尋不要用下拉式選單，可以用地圖呈現比較直觀。 | 我們考量到製作時間且下拉式選單較比直接地圖點選方式更直觀，所以我們決定維持原樣使用下拉式選單來搜尋站點。 |

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形 |
| 未來可建議讓民眾導入行程，提供交通資訊和推薦美食功能，可參考MaaS融合食宿遊購行概念。 |  |