國立台北商業大學

資訊管理系

**114** 資訊系統專案設計

**一張含有 標誌, 圖形, 符號, 字型 的圖片

AI 產生的內容可能不正確。系統手冊**

**組別：第114414組**

**題目：**

**指導老師：唐震老師**

**組長：11146013 薛嘉鈴**

**組員：11146015黃煒薰 11146006 柯妤蓁**

**11146028 鍾承諺 11146030 張承佑**

**中華民國114年5月28日**

**目錄**

[**第一章 前言** 1](#_Toc196992874)

[1-1 背景介紹 1](#_Toc196992875)

[1-2 動機 1](#_Toc196992876)

[1-4 預期成果 2](#_Toc196992877)

[**第二章 營運計畫** 3](#_Toc196992878)

[2-1可行性分析 3](#_Toc196992879)

[2-2 商業模式—Business model 3](#_Toc196992880)

[2-3 市場分析-STP 4](#_Toc196992881)

[2-4 競爭力分析SWOT-TOWS 4](#_Toc196992882)

[**第三章 系統架構** 6](#_Toc196992883)

[3-1 系統架構 6](#_Toc196992884)

[3-2 系統軟、硬體需求與平台技術 6](#_Toc196992885)

[3-3 使用標準與工具 7](#_Toc196992886)

[**第四章 專案時程與組織分工** 10](#_Toc196992887)

[4-1 專案時程 10](#_Toc196992888)

[4-2 專案組織與分工 10](#_Toc196992889)

[4-3 上傳GitHub紀錄 12](#_Toc196992890)

[**第五章 需求模型** 13](#_Toc196992891)

[5-1 使用者需求 13](#_Toc196992892)

[5-2 使用個案圖 13](#_Toc196992893)

[5-3 使用個案描述：使用活動圖描述 13](#_Toc196992894)

[5-4 分析類別圖 16](#_Toc196992895)

[**第六章 設計模型** 17](#_Toc196992896)

[6-1 循序圖 17](#_Toc196992897)

[6-2 設計類別圖 17](#_Toc196992898)

[**第七章 實作模型** 18](#_Toc196992899)

[7-1 佈署圖 18](#_Toc196992900)

[7-2 套件圖 18](#_Toc196992901)

[7-3 元件圖 18](#_Toc196992902)

[7-4 狀態圖 18](#_Toc196992903)

[**第八章 資料庫設計** 19](#_Toc196992904)

[8-1 資料庫關聯表 19](#_Toc196992905)

[8-2 表格及其Mata data 19](#_Toc196992906)

**圖目錄**

[圖 1 3-2-1台灣的行動裝置（手機和平板）上網流量占比 6](#_Toc196992907)

[圖 2 4-1-1 專案時程甘特圖 10](#_Toc196992908)

[圖 3 5-2-1 使用個案圖 13](#_Toc196992909)

[圖 4 5-4-1分析類別圖 16](#_Toc196992910)

**表目錄**

[表 1 1-2-1各運動夥伴媒合平台比較 1](#_Toc196992911)

[表 2 2-2-1商業模式九宮格 3](#_Toc196992912)

[表 3 2-4-1競爭力分析SWOT-TOWS 5](#_Toc196992913)

[表 4 3-2-1 PWA的優點 7](#_Toc196992914)

[表 5 3-2-2系統硬體需求 7](#_Toc196992915)

[表 6 4-2-1專業組織與分工表 10](#_Toc196992916)

[表 7 5-1-1使用者需求表 13](#_Toc196992917)

[表 8 5-3-1 註冊與登入 13](#_Toc196992918)

[表 9 5-3-2 建立個人檔案與偏好設定 14](#_Toc196992919)

[表 10 5-3-3查看場館與地圖資訊 14](#_Toc196992920)

[表 11 5-3-4活動發起與報名參與 14](#_Toc196992921)

[表 12 5-3-5自動媒合夥伴與推播通知 15](#_Toc196992922)

[表 13 5-3-6夥伴與活動評價機制 15](#_Toc196992923)

**第一章 前言**

**1-1 背景介紹**

近年來運動風氣興盛，健康意識抬頭，越來越多人想走出家門運動。然而，忙碌的生活讓人難以找到興趣相投、實力相當的運動夥伴與合適的場館。

因此，我們以「想運動卻找不到夥伴」的族群為出發點，設計一個整合場館資訊的平台，透過配對系統與揪團功能，聚集有相同需求的使用者，讓他們能更輕鬆地找到適合的運動夥伴。我們也會分析現有平台的優缺點與運動者的真實需求，提出具體的解決方案，打造一個更便利且人性化的運動媒合體驗。

**1-2 動機**

現代人越來越重視運動與健康，但在實際執行的過程中，許多人仍面臨不少困難。想運動時卻常常找不到人一起參與，身邊的朋友、同事、同學不是時間配合不上，就是運動程度落差太大，與陌生人又容易感到不自在。這些看似小小的障礙，往往就足以讓人放棄原本的運動計畫。本組成員皆曾遇過類似的困擾—偶爾興起運動的念頭，卻因為找不到人陪同而放棄。這些經驗讓我們深刻理解，「找到適合的運動夥伴」是一件既簡單卻又不容易實現的事。

目前市面上雖然已有一些運動媒合平台，但普遍知名度不高，使用者數也相對有限。因此，我們針對現有平台進行了優缺點分析，並思考為何大眾仍偏好透過社群平台來揪團運動，而不是使用這些專門的媒合工具。同時，我們也進一步探討：要如何設計一個平台，讓原本沒有運動習慣的人也能輕鬆加入運動圈，進而培養出規律運動的習慣？

為了提升使用者體驗與媒合成功率，我們希望整合地圖、場館資訊與使用者資料，結合各平台的優點並改善不足，打造一個更被大眾接受、資源更集中、媒合更高效的運動平台，幫助每一位使用者輕鬆找到理想的運動夥伴。

▼表 1 1-2-1各運動夥伴媒合平台比較

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 平台名稱 | 使用者體驗感 | 介面簡單 | 社群功能 | 場館資訊 | 運動夥伴媒合 |
| pajo |  |  | ✓ | ✓ | ✓ |
| Strava | ✓ | ✓ | ✓ |  |  |
| 馬拉松世界 |  |  | ✓ |  |  |
| Join sport | ✓ | ✓ | ✓ |  |  |
| Fitter | ✓ |  | ✓ |  |  |
| 我們做的 | ✓ | ✓ | ✓ | ✓ | ✓ |

**1-3 系統目的與目標**

本次專題的目標，為打造一個更完善的運動夥伴媒合平台，從使用者的角度出發，滿足運動者對運動夥伴的多元需求。透過蒐集使用者資料與運動數據，我們希望建立一個具備良好使用者體驗、介面簡潔易用、整合各地場館資訊、媒合效率高的平台，最終能成為取代傳統社群揪團方式的最佳選擇。

**1-4 預期成果**

預期本系統能透過蒐集使用者資料與運動數據，有效媒合適合的運動夥伴。使用者可自行發起揪團，系統則會主動推播給符合條件的對象，提升參與率與媒合效率。註冊流程中導入 Email 驗證機制，以提升帳戶使用安全性；同時整合 Google 地圖服務，提供即時場館資訊，協助使用者解決找不到場地的問題。

**第二章 營運計畫**

**2-1可行性分析**

1. 技術可行性：目前環境中所使用的技術已足以支援本系統之開發。後端採用 Flask 框架，搭配前端 HTML/CSS 製作響應式網頁，資料庫則透過 Docker 建立容器化的開發環境，並使用 Alembic 管理資料庫遷移與版本控制，確保資料表結構維護的一致性與彈性。Google Maps API 可整合場館資訊，提供地圖互動功能，提升使用者體驗。初期以條件篩選方式推薦運動夥伴，後續可逐步引入簡易演算法或機器學習模型，進行使用者行為分析與配對優化。整體技術架構輕量靈活，具備良好的擴充性與維護性，適合持續開發與功能延伸。
2. 市場可行性：

**2-2 商業模式—Business model**

▼表 2 2-2-1商業模式九宮格

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **關鍵夥伴**  •運動場地設施業者  •運動社團 | **關鍵活動**  •平台與媒合演算法開發  •場地與地圖資訊維護  •使用者評價管理 | **價值主張**  •快速安全配對合適運動夥伴  •地圖與場地資訊整合、活動輕鬆揪團  •使用者介面簡潔  •功能使用簡單體驗感佳 | | **顧客關係**  •用戶回饋管道 | **目標客群**  •想找運動夥伴但缺乏管道的運動者  •想建立運動習慣但需要動機  •有特定運動興趣 |
| **關鍵資源**  •專題開發小組與系統架構  •用戶資料與運動數據庫 | **通路**  •網站  •社群媒體(FB、IG) |
| **成本架構**  •網站開發維護成本  •行銷推廣 | | | **收益流**  •商業廣告曝光(品牌置入) | | |

根據表 2-2-1 商業模式九宮格分析，本組的目標族群為「想建立運動習慣卻苦於找不到人一起運動」的使用者。我們以運動夥伴媒合與揪團功能為核心，吸引更多有運動需求的人使用本系統。平台將提供快速且安全的運動夥伴配對服務，結合地圖與場地資訊整合、輕鬆發起活動揪團的功能，並強調介面簡潔、操作直覺、使用體驗良好。系統目前以商業廣告曝光作為主要收入來源。

**2-3 市場分析-STP**

Segmentation 市場區隔

* 年齡區隔：涵蓋青少年、成年人至全年齡層，針對不同年齡層設計差異化的服務內容與介面友善性。
* 地區區隔：以北部地區或都市區為主，這些地區民眾對科技接受度高、生活步調快，對健康與便利工具的需求較高。
* 心理區隔：重視健康、熱愛運動者，此族群常以運動維持身心平衡，並尋求相關社交互動。
* 行為區隔：針對缺乏運動夥伴、或有意願嘗試多元運動項目的用戶，開發能配對活動與人群的平台機制

Targeting 目標市場

* 青年社交運動族群，此族群多為大專生或上班族，重視生活與工作的平衡，習慣使用網路平台尋找活動與資源。他們傾向參與有趣、輕鬆又能兼具社交的運動，例如羽球、籃球、飛輪等團體活動。
* 充滿熱情的新手運動族群，此族群可能剛開始接觸運動，動機來自健康考量、朋友影響或個人成長需求，需要更多誘因與引導進入運動生活

Positioning 市場定位

* 便利性：整合 Google Maps，使用者可快速查找附近的運動場地與活動資訊。
* 社交性：設有評價與回饋系統，促進參與者之間的交流與互信，提升活動黏著度。
* 低門檻：操作介面清楚、步驟簡單，任何年齡層或經驗程度者皆可快速上手。
* 客製化：提供多元運動選擇，並依據用戶喜好推薦活動與夥伴，提升使用體驗與滿意度。

**2-4 競爭力分析SWOT-TOWS**

S — Strengths（優勢）

1. 市場需求明確：調查顯示，大多數受訪者在找運動夥伴時有困難，對媒合系 統存在明確需求。
2. 功能多元化潛力：可整合配對、聊天、評價、安全認證、活動揪團、場地資訊等功能，提高使用黏著度。
3. 運動趨勢持續成長：全民健身意識提升，有助推動使用者主動尋找運動夥伴。
4. 男女皆有需求：性別分布平均，能拓展更廣泛的用戶族群。

W — Weaknesses（劣勢）

1. 初期用戶基數不足：媒合平台在早期若沒有足夠使用者，配對體驗可能不佳。
2. 陌生人互動顧慮：安全性與信任機制不足，容易造成使用者卻步。
3. 時間與地點難配合：即使系統媒合成功，也可能因行程不合而無法參與。
4. 競品模仿容易：若無獨特機制或品牌力，容易被複製或取代。

O — Opportunities（機會）

1. 與運動場館、健身房、運動品牌合作：推動場地整合、優惠活動或品牌聯名，提高使用誘因。
2. 結合社群與挑戰機制：打造運動社群文化，如累積點數兌換、打卡挑戰、徽章制度等，提高參與度。
3. 結合健康科技或穿戴裝置：與手錶、APP 整合可提升精準媒合與健康追蹤。
4. 導入 AI 推薦系統：根據行為與偏好智能配對，提高媒合成功率與滿意度。

T — Threats（威脅）

1. 社群平台競爭（如 LINE 社團、Facebook）：現有平台已有一定使用者習慣。
2. 使用者黏著度低：若媒合後無持續機制或活動設計，可能淪為一次性使用。
3. 資料與隱私風險：個資安全與用戶信任為關鍵議題，任何漏洞都可能導致信任崩壞。
4. 高行銷推廣成本：需大量資源投入推廣與品牌建立，初期不易獲利。

▼表 3 2-4-1競爭力分析SWOT-TOWS

|  |  |  |
| --- | --- | --- |
| 內部  外部 | 優勢（Strengths） | 劣勢（Weaknesses） |
| 機會（Opportunities） | SO策略：利用多元功能與健康趨勢，結合場館／品牌合作迅速擴展用戶。導入 AI 與穿戴裝置強化配對功能，創造差異化優勢。 | WO策略：利用外部合作提升平台初期使用者數，改善媒合失敗問題。- 導入社群獎勵機制（打卡、點數）刺激使用者參與度，彌補平台吸引力不足。 |
| 威脅（Threats） | ST策略：透過平台特色（如活動揪團、評價機制）強化使用黏著度，降低與社群平台競爭衝擊。強調安全設計與信任機制，對抗資安與陌生人疑慮。 | WT策略：加強使用者教育與推廣活動，降低使用者流失與認知門檻。積極建置資安與客服制度，降低信任與風險成本。 |

**第三章 系統架構**

**3-1 系統架構**

分析出多個運動平台都有不同的優缺點，我們的系統旨在結合各平台優點，改善缺點，將系統功能著重於運動夥伴媒合揪團的部分，不發展太多功能模糊平台定位。先透過會員註冊蒐集資料到資料庫，使用資料分析技術，讓相同運動喜好及習慣的人能被配對到。系統架構主要分為三個部分：

1. 前端介面；使用 HTML、CSS 與 JavaScript 建構響應式網頁，提供會員註冊、登入、個人資料填寫與修改、偏好設定等功能介面，讓使用者能方便地完成個人資料建檔，並查看配對資訊。未來亦可擴充 PWA 技術，提升使用流暢度與便利性。
2. 後端服務：採用 Flask 框架開發 API，透過 SQLAlchemy 操作 MySQL 資料庫，搭配 Alembic 管理資料表遷移，實作會員資料儲存、運動偏好設定、媒合條件篩選等邏輯。未來可加入行為分析與推薦演算法以提升配對準確度。
3. 串接前端與後端：利用 Flask 所建立之 RESTful API 串接前端與 MySQL 資料庫，前端透過 fetch API 發送資料並接收回傳結果，實現資料即時更新與互動式操作，例如頭像上傳、偏好修改與配對條件儲存等功能。

**3-2 系統軟、硬體需求與平台技術**

本系統採用現今常見的 HTML 前端技術，結合 Python Flask 後端架構以及 MySQL 資料庫，建構成一個具備完整功能的網頁平台。為了提升平台的可用性與跨裝置支援，我們導入 PWA（Progressive Web App，漸進式網頁應用）技術，使本系統雖為網頁形式，卻能擁有接近原生 App 的使用體驗。

根據 StatCounter 2025 年 3 月的數據，​台灣的行動裝置（手機和平板）上網流量占比為 **59.9%**，桌上型電腦為 **40.1%** 。​這顯示出行動裝置已成為主要的上網工具，強調了開發跨平台、行動友善系統的重要性。

▼圖 1 3-2-1台灣的行動裝置（手機和平板）上網流量占比

為凸顯 PWA 作為平台核心的優勢，我們整理其三大關鍵特性如下：

▼表 4 3-2-1 PWA的優點

|  |  |
| --- | --- |
| 優點 | 說明 |
| 跨平台支援性高 | 使用者只需透過瀏覽器即可使用，不受限於 iOS 或 Android 系統。 |
| 可安裝像 App 一樣 | 使用者可將網站「加入主畫面」，一鍵啟動，介面近似原生 App。 |
| 節省開發與維護成本 | 開發一次即可同時支援手機與桌機，無需為不同系統分別維護。 |

考量網路環境差異與裝置多樣性，系統將支援各類網路連線模式（Wi-Fi / 4G / 5G），並針對低速網路進行基本快取優化。為使系統穩定運行於多數裝置上，系統硬體需求詳列如下：

▼表 5 3-2-2系統硬體需求

|  |  |
| --- | --- |
| 系統硬體需求－裝置端 | 需求規格說明 |
| 裝置類型 | 支援瀏覽器之智慧型手機 / 平板 / 桌機 |
| 作業系統版本 | Android 8.0 以上 / iOS 12 以上 / Windows 10 以上 / macOS 10.13 以上 |
| 瀏覽器建議版本 | Chrome、Safari、Edge 等主流瀏覽器（建議使用最新版） |
| 網路需求 | Wi-Fi / 4G / 5G |
| 儲存空間需求 | 至少 50MB 可用空間（用於快取與離線資料） |

**3-3 使用標準與工具**

以下列出了本組於開發運動夥伴媒合平台時所使用的主要開發工具與技術，並說明各項工具的特點與選用理由：

* Python：具備語法簡潔、易於維護的特性，搭配豐富的第三方套件，可快速建構 Web 應用與後端邏輯。團隊選擇 Python 作為後端開發語言，有效提升開發效率與可讀性。
* Flask：為 Python 撰寫的輕量級 Web 應用框架，適合中小型專案開發。Flask 提供靈活的架構與擴充性，便於快速實作 API 與處理會員資料、媒合邏輯等後端功能。
* MySQL：是一套穩定且開源的關聯式資料庫管理系統，適合用於儲存結構化資料。本系統使用 MySQL 儲存會員基本資料、運動偏好與配對條件，並透過 SQLAlchemy 操作資料表，確保資料一致性與安全性。
* DataGrip：專業的資料庫管理工具，支援多種資料庫系統。本組使用DataGrip 來查看與測試 MySQL 資料內容、編輯資料表結構，提升資料庫操作的效率與精確度。
* Alembic：作為資料庫遷移工具，搭配 SQLAlchemy 使用，可管理資料表版本與結構更新。使用 Alembic 能有效追蹤資料表異動紀錄，利於團隊協作與後續維護。
* HTML / CSS / JavaScript：作為網頁前端開發標準技術，負責呈現使用者介面與互動邏輯。前端實作響應式設計，支援手機、平板與桌機，確保各裝置皆有良好操作體驗。
* PWA（Progressive Web App）：導入漸進式網頁應用技術，使使用者可將系統加至主畫面並進行像 App 一樣的操作，同時具備離線快取、自動更新等優勢，提升行動裝置端的使用體驗。
* Visual Studio Code：輕量且功能齊全的程式開發環境，支援 Python、HTML、JavaScript 等語言的自動補全與除錯功能，亦可透過擴充套件強化開發流程，是本組主要使用的開發編輯器。
* GitHub：用於版本控制與團隊協作管理，支援多人分支開發、PR 檢視與歷程追蹤。透過 GitHub 管理專案程式碼，可同步掌握進度、避免程式碼衝突並提升團隊效率。
* Docker：為容器化部署工具，我們使用 Docker 架設 MySQL 與本地後端環境，確保開發環境與實際部署環境一致，降低環境差異導致的錯誤。
* uv：現代化的 Python 相依套件管理工具，支援快速安裝與版本鎖定功能。透過 uv.lock 檔案管理專案所需套件版本，確保團隊成員開發環境一致性，避免版本不一致導致的錯誤。
* Google Maps API：提供地圖與定位服務，協助系統呈現活動地點、查詢使用者位置及搜尋附近運動場館等功能。本組使用 Google Maps JavaScript API 將地圖嵌入網頁，並結合使用者座標達到地點視覺化與互動性，提升活動參與的便利性與真實感。

|  |  |
| --- | --- |
| 分類 | 工具 / 技術名稱 |
| 系統開發環境 | Windows 11、macOS、Visual Studio Code |
| 開發平台 | HTML、CSS、JavaScript、Python、Flask、MySQL、Docker |
| 程式開發工具 | Flask、SQLAlchemy、Alembic、Google Maps API、DataGrip |
| 前端技術 | HTML、CSS、JavaScript、PWA（Progressive Web App） |
| 後端技術 | Python、Flask、MySQL、Alembic、Docker |
| 資料表設計與版本管理 | SQLAlchemy（ORM 模型定義）、Alembic（資料表遷移與版本控制） |
| 資料庫管理工具 | DataGrip |
| 圖表與設計工具 | Diagrams.net |
| 文件與簡報工具 | Microsoft Word、Canva |
| 版本與專案管理 | Git、GitHub |
| 檔案儲存平台 | GitHub、Google Drive |
| 相依套件管理 | uv（Python dependency lock tool） |

**第四章 專案時程與組織分工**

**4-1 專案時程**

根據專案工作流程,我們把專案分成 16 個項目,將專案彙整成甘特圖如下:

▼圖 2 4-1-1 專案時程甘特圖

**4-2 專案組織與分工**

▼表 6 4-2-1專業組織與分工表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 項目/組員 | | 11146013  薛嘉鈴 | 11146015  黃煒薰 | 11146006  柯妤蓁 | 11146028  鍾承諺 | 11146030  張承佑 |
| 系統規劃 | 功能樹與流程設計 |  |  |  |  |  |
| 使用者需求分析 |  |  |  |  |  |
| 前端開發 | 使用者登入頁面 |  |  |  |  |  |
| 使用者註冊頁面 |  |  |  |  |  |
| 偏好運動表單頁面 |  |  |  |  |  |
| 基本資料頁面 |  |  |  |  |  |
| 建立活動頁面 |  |  |  |  |  |
| 查看活動頁面 |  |  |  |  |  |
| 查看會員資料頁面 |  |  |  |  |  |
| Figma 系統頁面雛形 |  |  |  |  |  |
| UI/UX 設計 |  |  |  |  |  |
| 後端開發 | 資料庫設計 |  |  |  |  |  |
| MySQL 架構設計 |  |  |  |  |  |
| SQLAlchemy 模型建立 |  |  |  |  |  |
| Alembic 資料表遷移 |  |  |  |  |  |
| 初始資料建立與測試 |  |  |  |  |  |
| 會員註冊\登入\顯示\編輯資料API |  |  |  |  |  |
| 頭像上傳 / 裁切功能 |  |  |  |  |  |
| 建立活動/查看活動API |  |  |  |  |  |
| Google Maps API |  |  |  |  |  |
| 美術設計 | Logo 設計 |  |  |  |  |  |
| 色彩設計 |  |  |  |  |  |
| 名牌製作 |  |  |  |  |  |
| 文件撰寫 | 統整 |  |  |  |  |  |
| 第一章 前言 |  |  |  |  |  |
| 第二章 營運計畫 |  |  |  |  |  |
| 第三章 系統架構 |  |  |  |  |  |
| 第四章 專案時程與組織分工 |  |  |  |  |  |
| 第五章 需求模型 |  |  |  |  |  |
| 第六章 設計模型 |  |  |  |  |  |
| 第七章 實作模型 |  |  |  |  |  |
| 第八章 資料庫設計 |  |  |  |  |  |
| 報告 | 簡報製作 |  |  |  |  |  |

●主要負責人/○次要負責人

**4-3 上傳GitHub紀錄**

**第五章 需求模型**

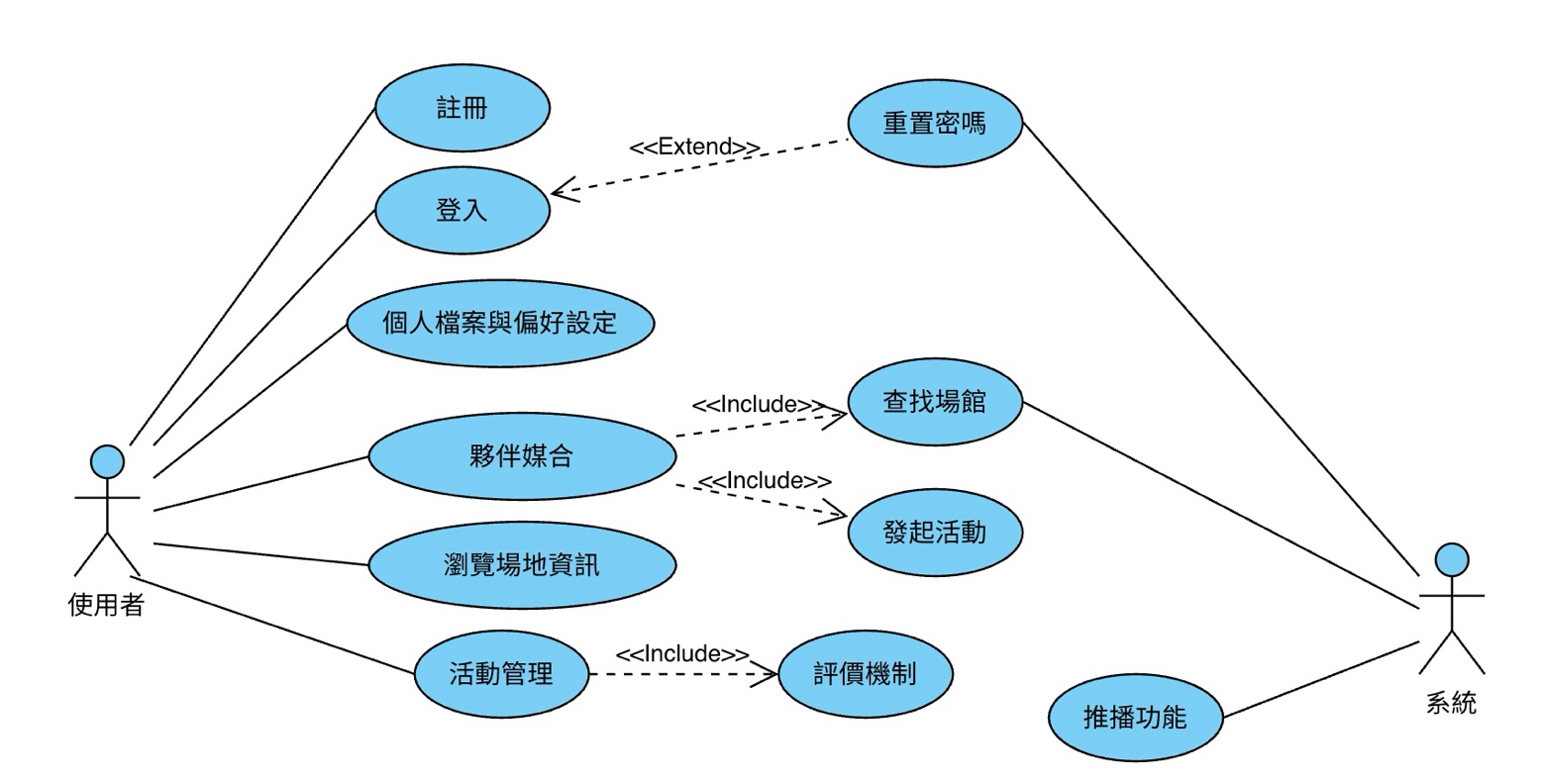
**5-1 使用者需求**

▼表 7 5-1-1使用者需求表

|  |  |
| --- | --- |
| Use Case | 需求 |
| 註冊與登入功能 | 使用者可註冊帳號並透過Email驗證完成註冊、登入。 |
| 建立個人檔案與偏好設定 | 使用者可填寫基本資料與選擇喜好運動項目、經常運動時間等作為配對依據。 |
| 查看場館與地圖資訊 | 使用者可透過整合Google地圖查找附近運動場地，並顯示場地基本資訊。 |
| 活動發起與報名參與 | 使用者可主動建立運動活動，也可瀏覽與報名參與他人發起的活動。 |
| 自動媒合夥伴與推播通知 | 系統會依據偏好自動媒合潛在運動夥伴，並推播活動邀請給合適對象。 |
| 夥伴與活動評價機制 | 使用者可對活動品質與運動夥伴進行評價，作為平台信任機制之一。 |

**5-2 使用個案圖**

▼圖 3 5-2-1 使用個案圖



**5-3 使用個案描述：使用活動圖描述**

▼表 8 5-3-1 註冊與登入

|  |  |
| --- | --- |
| 使用個案名稱：註冊與登入 | |
| 行為者 | 使用者 |
| 前提 | 使用者尚未擁有帳號或尚未登入系統。 |
| 結束狀態 | 使用者成功註冊帳號並完成登入，進入平台主畫面。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 輸入 Email 與密碼 | 檢查欄位格式是否正確 |
| 點選「註冊」按鈕 | 寄送驗證信至使用者 Email |

▼表 9 5-3-2 建立個人檔案與偏好設定

|  |  |
| --- | --- |
| 使用個案名稱：建立個人檔案與偏好設定 | |
| 行為者 | 使用者 |
| 前提 | 使用者已成功註冊並登入平台。 |
| 結束狀態 | 使用者成功儲存個人資料與運動偏好，作為媒合依據。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 輸入基本資料（暱稱、性別、生日、地區等） | 即時驗證格式正確性 |
| 選擇偏好運動項目（可複選） | 顯示運動類別圖示選單，支援多選 |
| 選擇常運動的時間時段 | 顯示選項（早上、下午、晚上、彈性）供勾選 |

▼表 10 5-3-3查看場館與地圖資訊

|  |  |
| --- | --- |
| 使用個案名稱：查看場館與地圖資訊 | |
| 行為者 | 使用者 |
| 前提 | 使用者已登入系統並進入主頁或地圖頁。 |
| 結束狀態 | 使用者查詢到所需的運動場館資訊並可作進一步操作（如報名、揪團等）。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 點選「場館地圖」或「找場地」 | 顯示整合 Google 地圖的場地頁面 |
| 點選任一場館圖示 | 彈出場館詳細資訊卡片（如場地名稱、地址、開放時段） |

▼表 11 5-3-4活動發起與報名參與

|  |  |
| --- | --- |
| 使用個案名稱：活動發起與報名參與 | |
| 行為者 | 使用者 |
| 前提 | 使用者已登入系統並完成基本資料與偏好設定。 |
| 結束狀態 | 使用者成功建立活動或完成活動報名，活動資訊更新於平台中。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 點選「發起活動」 | 顯示活動建立表單（含運動種類、時間、地點、限制人數等欄位） |
| 輸入活動資訊並提交 | 系統儲存活動至資料庫並顯示「發起成功」提示 |
| 點選「報名參與」 | 系統檢查資格與人數上限，符合條件則報名成功並顯示確認畫面 |

▼表 12 5-3-5自動媒合夥伴與推播通知

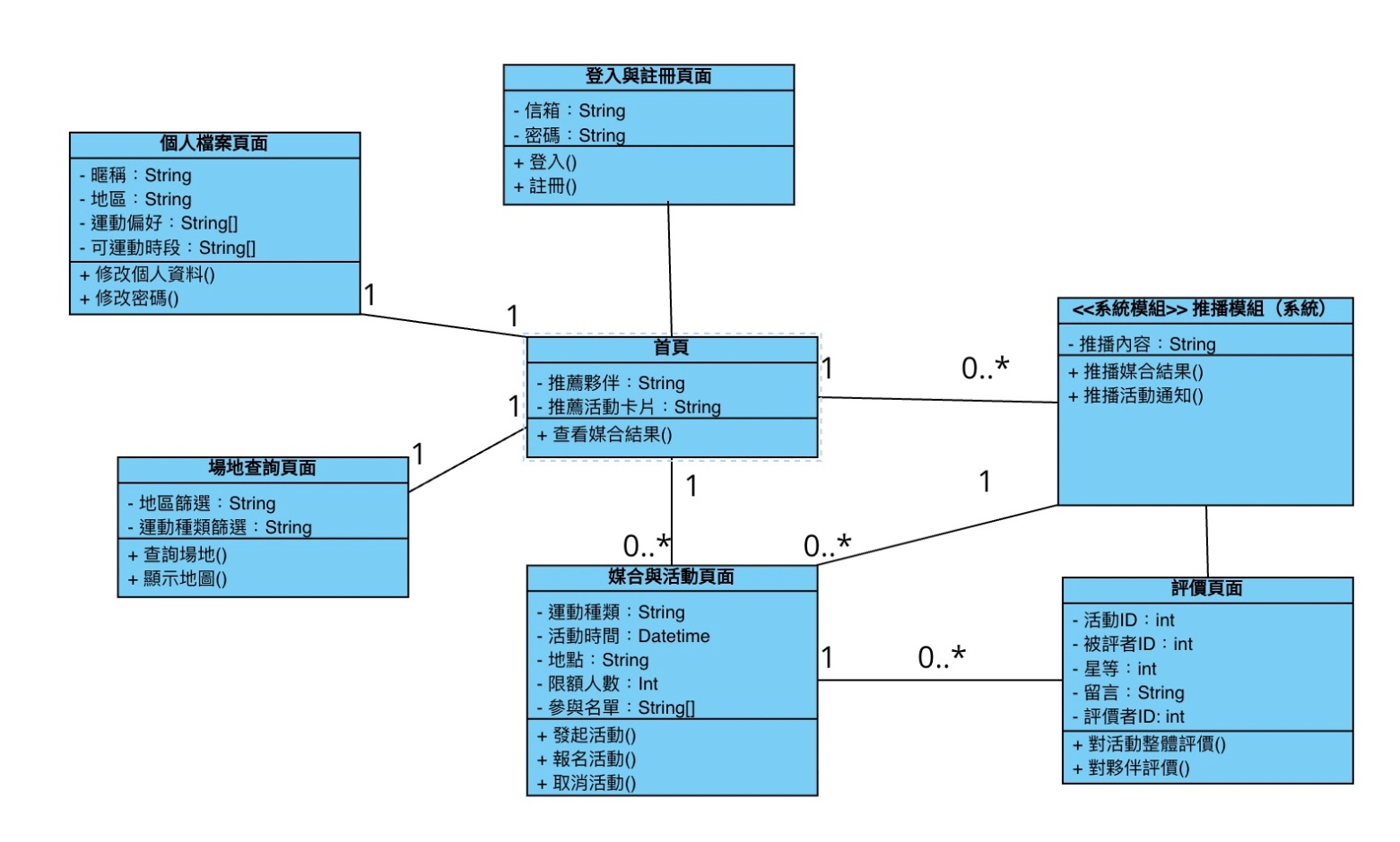
|  |  |
| --- | --- |
| 使用個案名稱：自動媒合夥伴與推播通知 | |
| 行為者 | 使用者 |
| 前提 | 使用者已完成個人偏好設定，並開啟通知功能。 |
| 結束狀態 | 使用者收到媒合夥伴或活動的推播訊息，並可進行進一步參與。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 無操作時等待系統通知 | 系統定期執行媒合演算法，尋找相符夥伴與活動 |
| 點選推播訊息 | 導向活動詳情頁或對方個人簡介頁 |
| 參與活動或邀請對方聯絡 | 系統更新報名狀態 |

▼表 13 5-3-6夥伴與活動評價機制

|  |  |
| --- | --- |
| 使用個案名稱：夥伴與活動評價機制 | |
| 行為者 | 使用者 |
| 前提 | 使用者已參加完一場活動，並符合評價資格。 |
| 結束狀態 | 評價內容成功儲存至系統，作為後續媒合與信任參考依據。 |
| 事件路徑： | |
| Actor 動作 | 系統回應 |
| 點選某場已完成活動 | 顯示活動詳情與評價按鈕 |
| 送出評價 | 儲存評價至資料庫，並更新夥伴信譽指標 |
| 回顧歷史評價 | 可查看自己過去對他人或活動的評價記錄 |

**5-4 分析類別圖**

▼圖 4 5-4-1分析類別圖



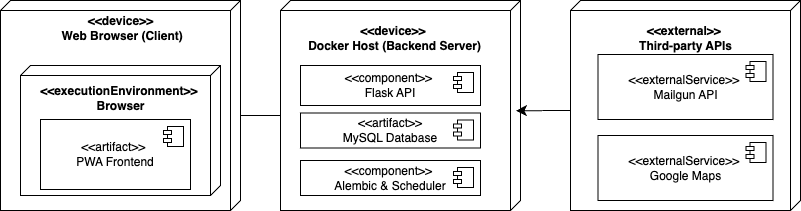
**第六章 設計模型**

**6-1 循序圖**

**6-2 設計類別圖**

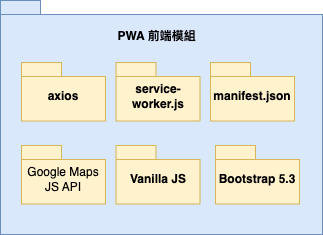
**第七章 實作模型**

**7-1 佈署圖**

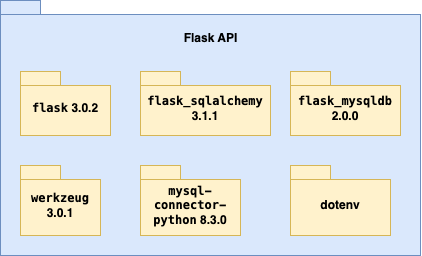
****

**7-1-1佈署圖**

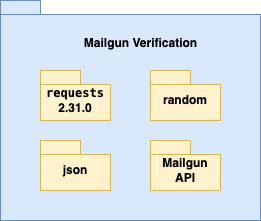
**7-2 套件圖**

****

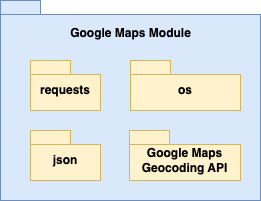
套件圖-PWA前端模型



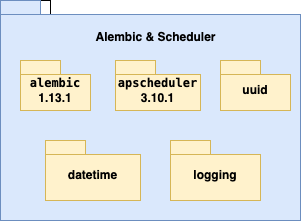
套件圖-Flask API 串接模組



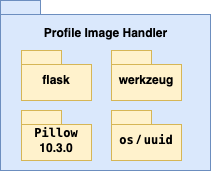
套件圖-Email 驗證模組



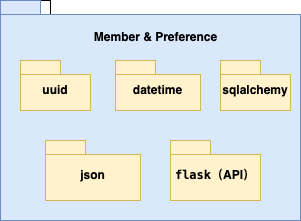
套件圖-Google Maps 地理位置模組



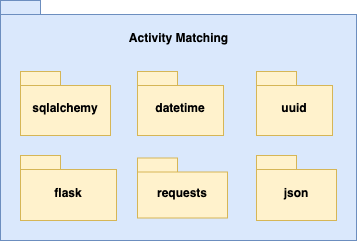
套件圖-Alembic \_ Scheduler 模組



套件圖-圖片上傳與裁切模組



套件圖-會員註冊與偏好模組



套件圖-活動配對與參與模組

**7-3 元件圖**

**7-4 狀態圖**

**第八章 資料庫設計**

**8-1 資料庫關聯表**

**8-2 表格及其Mata data**