

國立臺北商業大學

資訊管理系

111' 資訊系統專案設計

系統手冊



MRTaipei

組別：第 111208 組

題目：MRTaipei | 台北先生

指導老師：許晉龍老師

組長：11036006 王佑強

組員：11036025 柯玄娜 11036027 錢慧蓁

11036029 李崧毓 11036041 龔渝捷

中華民國 112 年 1 月 4 日

國立臺北商業大學專題課程作品

電子文件上網授權書

本授權書所授權之作品為授權人在 國立臺北商業大學 資訊管理（科/系）

 組 學年度 第 學期專題課程之成果報告作品。

專題題目：

指導教授：

☐ 本人 ☐ 團體著作人

☐ 同意 提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

茲同意將授權人擁有著作權之上列論文全文（含摘要）無償授權本人就讀學校（國立臺北商業大學）圖書館，不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列作品收錄、重製與利用，並得將數位化之上列作品及其電子檔上載資料庫系統。於著作權法合理使用範圍內，讀者得進行線上檢索、閱覽、下載或列印。

專題課程成果報告作品全文上載網路公開之範圍及時間：

本校區域網路 ☐ 立即公開

☐ 自中華民國 年 月 日公開

校外網際網路 ☐ 立即公開

☐ 自中華民國 年 月 日公開

☐ 不同意 提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

授權人

簽 名

（作品若為團體作品，請每一位團體成員均應簽署）

中 華 民 國 年 月 日

目錄

第 1 章 背景與動機	1
1-1 簡介	1
1-2 問題與機會	2
1-3 相關系統探討	3
第 2 章 系統目標與預期成果	4
2-1 系統目標	4
2-2 預期成果	4
第 3 章 系統規格	5
3-1 系統架構	5
3-2 系統軟、硬體需求與技術平台	7
3-3 使用標準與工具	8
第 4 章 專案時程與組織分工	9
4-1 專案時程	9
4-2 專案組織與分工	10
第 5 章 需求模型	11
5-1 使用者需求	11
5-2 使用個案圖(Use case diagram)	12
5-3 使用個案描述	13
5-4 分析類別圖(Analysis class diagram)	19
第 6 章 設計模型	20
6-1 循序圖(Sequential diagram)	20
6-2 設計類別圖(Design class diagram)	27
第 7 章 實作模型	28
7-1 佈署圖(Deployment diagram)	28
7-2 套件圖(Package diagram)	28
7-3 元件圖(Component diagram)	28
7-4 狀態機(State machine)或時序圖(Timing diagram)	29
第 8 章 資料庫設計	33
8-1 資料關聯表	33
8-2 資料表及其 Meta data	34
第 9 章 程式	36
9-1 元件清單及其規格描述	36
9-2 其他附屬之各種元件	104
第 10 章 測試模型	107
10-1 測試計畫	107
10-2 測試個案與測試結果資料	108
第 11 章 操作手冊	112
第 12 章 使用手冊	113
第 13 章 感想	117

第 14 章 參考資料 120

附錄 121

圖目錄

圖 3-1-1、系統架構圖	5
圖 3-1-2、系統功能架構圖	6
圖 5-2-1、使用個案圖	12
圖 5-3-1、登入之活動圖	13
圖 5-3-2、路線規劃之活動圖	14
圖 5-3-3、下車提醒之活動圖	15
圖 5-3-4、收藏路線之活動圖	15
圖 5-3-5、回饋金即時報之活動圖	16
圖 5-3-6、優惠資訊之活動圖	16
圖 5-3-7、尋找遺失物(使用者)之活動圖	17
圖 5-3-8、尋找遺失物(管理者)之活動圖	18
圖 5-4-1、分析類別圖	19
圖 6-1-1、登入之循序圖	20
圖 6-1-2、路線規劃之循序圖	21
圖 6-1-3、下車提醒之循序圖	22
圖 6-1-4、收藏路線之循序圖	22
圖 6-1-5、回饋金即時報之循序圖	23
圖 6-1-6、優惠資訊之循序圖	24
圖 6-1-7、尋找遺失物(使用者)之循序圖	25
圖 6-1-8、尋找遺失物(管理者)之循序圖	26
圖 6-2-1、設計類別圖	27
圖 7-1-1、佈署圖	28
圖 7-2-1、套件圖	28
圖 7-3-1、元件圖	28
圖 7-4-1、登入狀態機	29
圖 7-4-2、驗證狀態機	29
圖 7-4-3、規劃路線狀態機	30
圖 7-4-4、下車提醒狀態機	30
圖 7-4-5、收藏路線狀態機	31
圖 7-4-6、查詢遺失物狀態機	31
圖 7-4-7、通報遺失物狀態機	32
圖 8-1-1、資料庫關聯表	33
附錄-1、科技志工提案證明書	121
附錄-2、資服創新競賽(資訊應用組)	121
附錄-3、資服創新競賽	121

表目錄

表 1-2- 1、SWOT 分析表	2
表 1-3- 1、相關系統比較表	3
表 3-2- 1、伺服器端規格表	7
表 3-2- 2、手機端規格表	7
表 3-3- 1、使用工具表	8
表 4-1- 1、專案時程表	9
表 4-2- 1、專案組織與分工表	10
表 5-1- 1、功能性需求表	11
表 5-1- 2、非功能性需求表	11
表 8-2- 1、users-使用者 資料表	34
表 8-2- 2、fav_routes -收藏路線 資料表	34
表 8-2- 3、lost_items -遺失物 資料表	35
表 9-1- 1、後端程式清單	36
表 9-1- 2、前端程式清單	57
表 9-2- 1、附屬程式清單	104
表 10-2- 1、網站瀏覽測試	108
表 10-2- 2、登入測試	108
表 10-2- 3、功能選單測試	108
表 10-2- 4、路線規劃測試	108
表 10-2- 5、搜尋最佳路線測試	108
表 10-2- 6、詳細路線資料測試	109
表 10-2- 7、車站內部路線測試	109
表 10-2- 8、下車提醒測試	109
表 10-2- 9、收藏路線測試	109
表 10-2- 10、收藏路線清單測試	109
表 10-2- 11、瀏覽收藏路線測試	110
表 10-2- 12、尋找遺失物測試	110
表 10-2- 13、搜尋遺失物測試	110
表 10-2- 14、瀏覽遺失物測試	110
表 10-2- 15、通報遺失物(使用者)測試	111
表 10-2- 16、回饋金即時報測試	111
表 10-2- 17、優惠資訊測試	111
表 10-2- 18、遺失物通報(管理者)測試	111
表 11-1- 1、使用者與管理者的操作頁面	112
表 12-1- 1、使用者的使用手冊	113
表 12-1- 2、管理者的使用手冊	116

第1章 背景與動機

1-1 簡介

臺北捷運是臺北都會區非常重要的一項大眾運輸工具，也是民眾日常通勤和出遊最依賴的交通工具之一，臺北捷運系統雖方便快捷，但是當站內連接著其他路線，或是擁有好幾個出口，都很容易讓人迷失方向。為了剛到臺北都會區的北漂青年，又或者是平常甚少使用捷運系統的民眾，他們時常因看不懂站內外地圖，導致繞遠路浪費不少時間，如何協助他們在搭乘捷運時能夠更有效率的進行移動，是值得深思的課題。

捷運擁有許多路線，但站點固定，民眾若想前往捷運未達之處，可利用 YouBike 代步。YouBike 服務固然便利，但不確定附近站點是否還有車輛可供租借、欲前往的站點是否有位置停放等等，都是民眾使用該服務易出現的困擾，這些問題也會影響民眾使用 YouBike 的意願。

對於不熟悉捷運系統，或是偶爾要前往陌生地點的民眾來說，路線導航是很需要的服務，但市面上應用程式品質良莠不齊，造成民眾不知道該如何選擇，需到陌生地點非經常行為，此服務使用頻率不高，民眾額外下載應用程式意願低，這時若是選擇從台灣通訊軟體市占率近 9 成的應用程式——LINE 著手進行服務，使用 LINE 提供的聊天機器人服務「LINE Bot」，透過 Line Bot 導向網頁端，讓多元服務唾手可得、完善、好操作的應用程式讓民眾接受度提升。

1-2 問題與機會

▼表 1-2- 1、SWOT 分析表

Strengths 優勢	Weaknesses 劣勢
1. 免下載 2. 操作簡單，流程方便快捷 3. 開發成本低	1. 相較於現有相關應用程式，進階功能少 2. 知名度低
Opportunities 機會	Threats 威脅
1. 低碳生活盛行 2. 大眾運輸間轉乘複雜	1. 已有相似功能應用程式 2. 疫情期間大眾運輸運量降低

環保意識的提升，政府開始宣傳低碳生活，鼓勵大家以大眾運輸作為主要交通工具，而對於大臺北民眾來說臺北捷運是生活上不可或缺的交通工具，亦是旅客來臺北旅遊逛街最常使用的交通工具之一。但對於北漂青年以及不熟悉北捷的民眾，大眾運輸間的轉乘相當複雜，所以我們決定開發此系統來協助民眾。

根據我們的分析結果，現今相關應用程式眾多，像是台北捷運 Go 和台北等公車等，由於它們的上市時間較長、進階功能多，且具有一定的知名度，以致於我們較難在市場上佔有一席之地。再加上我們目前系統功能以簡單實用的路線規劃及回饋金查詢等功能為主，額外進階功能(例如：i 郵箱查詢、置物櫃查詢等)少，較難吸引使用者，所以在應用程式上市初期使用率可能偏低，大眾接受度不高。

因此我們系統以臺灣普遍使用的通訊軟體 LINE 作為平台，採免下載的方式，不但操作簡單，流程方便快捷，亦可滿足使用者願意接受資訊卻不願意多下載程式的需求，來提升使用者的接受度，進而提高使用率。此外，系統附帶下車提醒功能，能夠降低習慣在通勤間休息的民眾的不安感，提高搭乘的舒適度。且系統提供之遺失物通報功能讓使用者在拾獲或丟失物品時，可以先一步通報，讓丟失物品的人可以更快速找回物品，減去人員的負擔以及協尋的時間等，所以不管是在開發成本或人力成本等都相對較低。

1-3 相關系統探討

以下是我們從目前較常見的台北捷運 GO、台北等公車及 Bus+去實際操作應用程式後所進行的比較。

▼表 1-3- 1、相關系統比較表

		MRTaipei 台北先生	台北捷運 GO	台北等公車	Bus+
下載方式		免下載，加入 LINE 官方帳號即可	需下載程式	需下載程式	需下載程式
註冊方式		免註冊，有 LINE 即可	部分功能需先註冊	註冊即可享雲端同步	部分功能需先註冊
路線規劃	站內	✓ (鄰近手扶梯和車廂)	✗	✗	✗
	站外	✓ 結合大眾交通工具和 YouBike	✗	✓ 公車為主	✓ 公車和捷運為主
YouBike 站點資訊提供		1.0/2.0、可借數量、可停車位與位置等。	僅提供相關連結(YouBike 官網)。	1.0/2.0、可借數量、可停車位與位置等。	可借數量、可停車位與位置
下車提醒		以 LINE 進行推播通知。	需連結藍牙。	提供公車到站通知。	提供公車到站通知。
回饋查詢		以視覺化呈現	以文字顯示	✗	✗
遺失物通報		✓ 提供遺失物丟失及拾獲的查詢與通報	✓ 提供遺失物協尋登錄	✗	✗

第2章 系統目標與預期成果

2-1 系統目標

臺北是一個以捷運作為主要交通工具的地方，但是大多數民眾或旅客可能會因看不懂捷運網路圖，而不知道該搭甚麼方向的車或該去哪裡轉車。

然而現在相關應用程式眾多，在沒有他人的推薦下很難抉擇哪個應用程式最適合自己使用，甚至是有的人不願在手機上下載各式各樣的應用程式，因此我們以 LINE 作為平台，開發了此系統。

提供使用者在免額外下載應用程式的前提下，進行最佳路線規劃，不僅包含了站內外路線導覽，更結合了公車、捷運及 YouBike 等交通工具去規劃路線，且附有以下車提醒，透過 LINE 去進行推播提醒。

還有額外提供遺失物通報的功能，假設使用者丟失物品時，可以透過本系統通報丟失的物品，也提供管理者可以將拾獲之遺失物進行通報比對，如有丟失之遺失物與拾獲之遺失物相似，將直接透過 LINE 通知丟失之使用者。

未來希望可以結合簡單的遺失物影像辨識系統快速辨識出物品的特徵，使登記流程更簡潔，幫助民眾更易找回所遺失的物品。

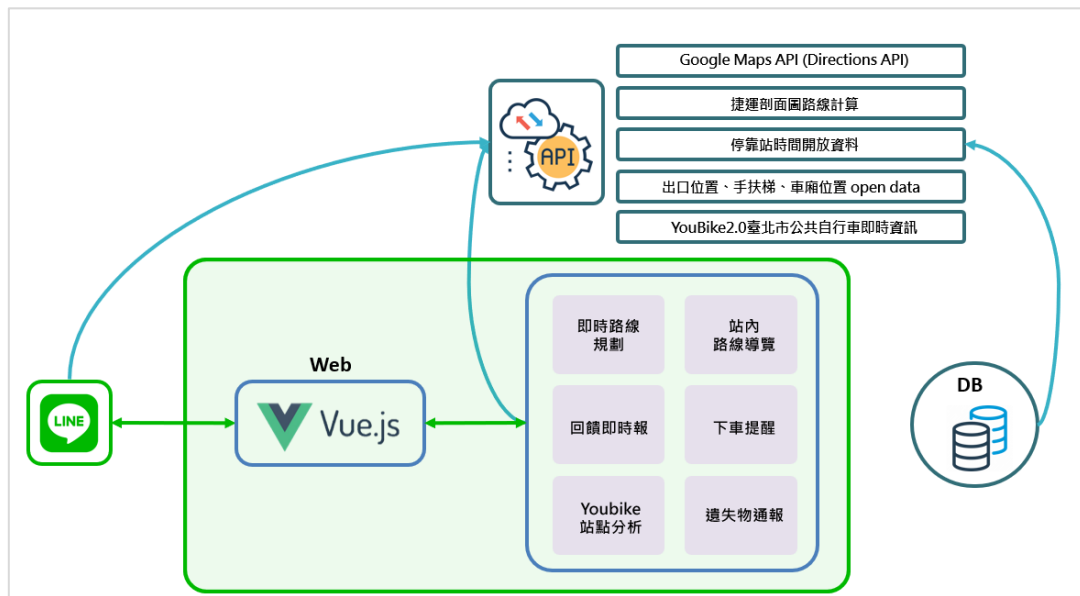
2-2 預期成果

透過完整的路線規劃，協助旅客以更有效率的方式搭乘、轉乘捷運公車和 YouBike，且透過下車提醒，也可以降低旅客搭乘捷運時的不安感。

在免額外下載應用程式的前提下，只需加入 LINE 官方帳號即可使用，因此使用者加入的門檻較低，接受度相對提高，便可有效累積使用者。

第3章 系統規格

3-1 系統架構



▲圖 3-1- 1、系統架構圖

在使用功能服務前，使用者可以選擇透過 LINE 加入官方帳號，點擊相應的圖文選單即可，如下：

1. 登入

使用者在首次加入 LINE 加入官方帳號後，如要使用提供之所有功能的使用者，則需先透過 LINE 允許 APP 存取資訊後，即可使用所有功能；反之如拒絕存取者，則只能使用部分功能(如：路線規劃及額外優惠)。

管理者則是透過管理者之 LINE 帳號進入，即可登記拾獲之遺失物。

2. 路線規劃（收藏路線）

使用者在點擊圖文選單後，只要設定「出發地」與「目的地」，系統會透過使用者所輸入的位置，與「捷運車站出入座標開放資料」結合 Geocoding API 服務，替使用者規劃路線與最近出口。同時搭配實際探訪資料，提供最近手扶梯和車廂的資訊，並根據 YouBike 站點位置與剩餘車輛、空位自動引導最佳 YouBike 站點。

如需收藏路線，可點擊收藏按鈕，日後若有需要只須到路線收藏清單選擇路線，即可直接幫使用者規劃好路線。

3. 下車提醒

在路線規劃完成後，使用者可以預先查看路線資訊，並點選出發開始行程。如需下車提醒的使用者，只需在開始行程後，點擊下車提醒，系統就會根據使用者 GPS 資訊與「捷運兩站之行駛時間、停靠站時間開放資料」中的停靠站時間、行駛時間進行計算分析，在使用者快到達目的地前兩分鐘發下車提醒推播。

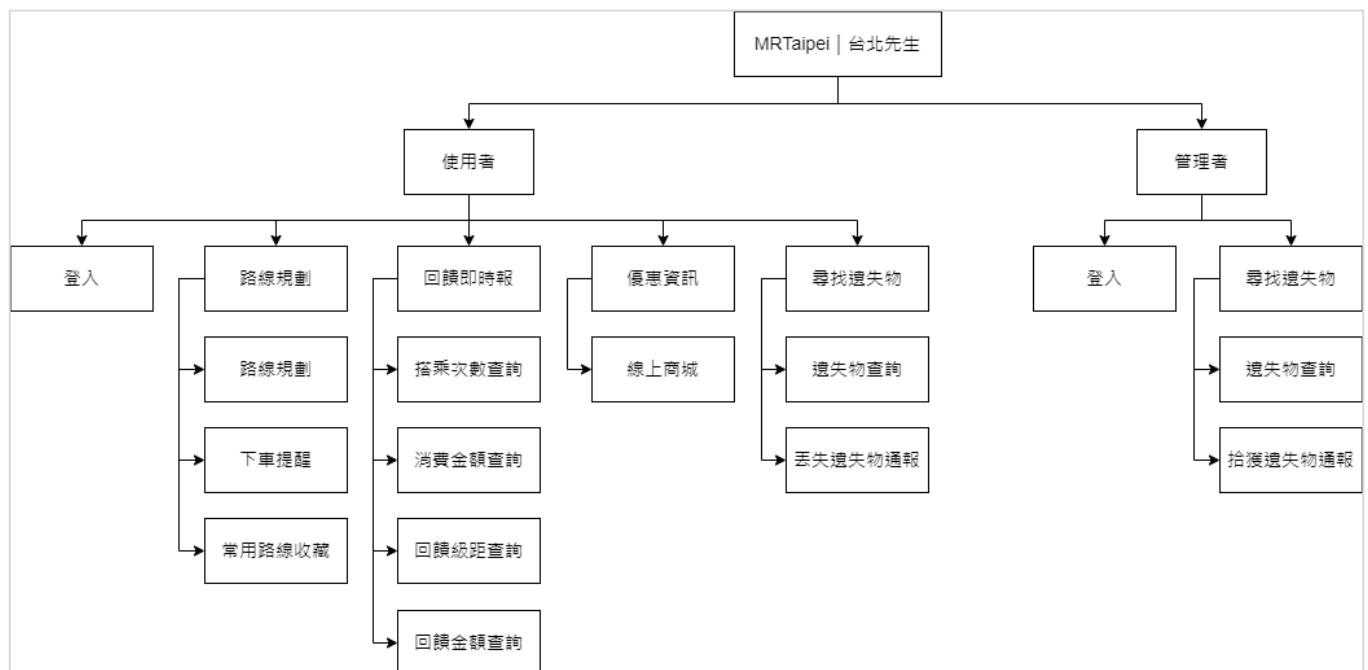
4. 回饋即時報

使用者只要點選回饋即時報，系統會根據開放資料中的「常客優惠資訊開放資料」，以視覺化的方式呈現目前搭乘次數、消費金額、回饋級距與金額。

5. 尋找遺失物

如有拾獲遺失物之民眾，只需將物品帶至捷運站服務中心，由管理者將遺失物之資料輸入系統，並與系統中現有的遺失物進行比對，如有相似物品則透過 LINE 通知該丟失之使用者，反之則進行通報。

若為遺失物品之民眾，可點擊尋找遺失物按鈕，接著點選我要通報按鈕，即可進行遺失物通報。如有拾獲該遺失物，並經系統比對後有符合項目者，則通知民眾前往領取。



▲圖 3-1-2、系統功能架構圖

3-2 系統軟、硬體需求與技術平台

▼表 3-2- 1、伺服器端規格表

伺服器端規格	
作業系統	Ubuntu 20.04
開發環境	Windows 10
整合式開發環境(IDE)	Visual Studio Code
程式語言	Python
框架	Laravel
套件管理	Composer
伺服器	Nginx
資料庫	MySQL

▼表 3-2- 2、手機端規格表

手機端規格	
作業系統	Android 10 , iOS 15.4
開發環境	Windows 10
整合式開發環境(IDE)	Visual Studio Code
程式語言	JavaScript
框架	Vue.js
套件管理	npm

3-3 使用標準與工具

▼表 3-3- 1、使用工具表

開發輔助工具	
編輯器	Visual Studio Code
資料庫管理工具	phpMyAdmin
版本控制工具	Git
API 測試工具	Postman
文件撰寫工具	
文件製作	Microsoft Word
簡報製作	Microsoft PowerPoint
UML 工具	Draw.io
介面設計工具	
介面設計工具	Figma
雛型設計	Figma
製圖工具	Adobe Ai, Ps
專案管理工具	
溝通工具	LINE、Telegram
版本控管	Git
進度追蹤	Trello
影片軟體	
影片軟體	FinalCutPro
剪輯軟體	FinalCutPro

第4章 專案時程與組織分工

4-1 專案時程

▼表 4-1-1、專案時程表

任務描述	2022											
	一月	二月	三月	四月	五月	六月	七月	八月	九月	十月	十一月	十二月
題目構想												
資料蒐集												
功能分析												
系統需求分析												
資料庫建置												
伺服器架設												
系統前端開發												
系統後端開發												
路線規劃												
LIFF Login												
LINE Notification												
系統整合												
系統測試												
文件製作												
簡報製作												
海報製作												

4-2 專案組織與分工

▼表 4-2- 1、專案組織與分工表

●：主要 ◎：協助		11036006 王佑強	11036025 柯玄娜	11036027 錢慧蓁	11036029 李崧毓	11036041 龔渝捷
企劃	主題構想	●				
	需求分析	◎	●	●	◎	●
	系統分析	◎	●	●	◎	●
	資料收集	●	●	●	●	●
美術	介面設計	●	●	●	●	●
	Logo 設計	●	●	●		●
前端	(UI/UX)前端設計	●			●	●
	前端開發				●	
後端	資料庫建置			●	◎	
	伺服器架設			●	◎	
	LINE 建置		●	●		●
	LIFF Login		◎		●	
	路線規劃	●			◎	
	路線收藏	●				
	下車提醒		●	●		●
	尋找遺失物	●			◎	
	回饋金即時報	●			◎	
報告	文件撰寫		●	●		●
	簡報製作		●	●		●
	海報製作		●	●		●
	影片製作	●				●

第5章 需求模型

5-1 使用者需求

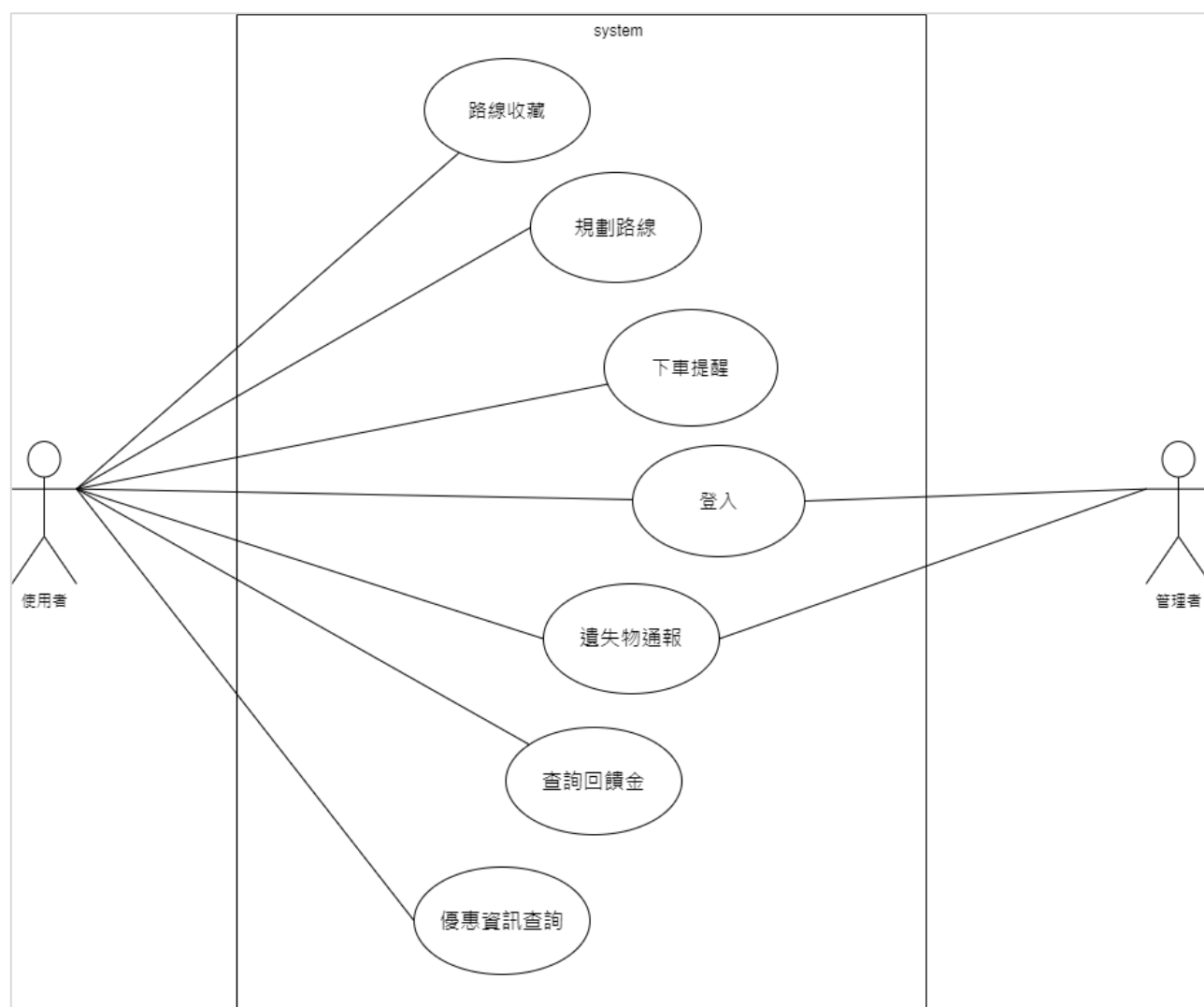
▼表 5-1- 1、功能性需求表

功能性需求	
功能項目	說明
登入	針對使用者與管理者提供 LINE 登入方式。
路線規劃	針對使用者提供站內外路線查詢規劃、常用路線收藏。
下車提醒	針對使用者提供下車提醒，以 LINE 進行推播通知。
回饋金查詢	針對使用者提供查詢目前搭乘次數、消費金額、回饋級距與金額。
遺失物通報	針對使用者查詢遺失物、遺失物通報。 針對管理者提供拾獲查詢遺失物、遺失物通報。
優惠資訊	針對使用者提供額外優惠資訊(捷客配)。

▼表 5-1- 2、非功能性需求表

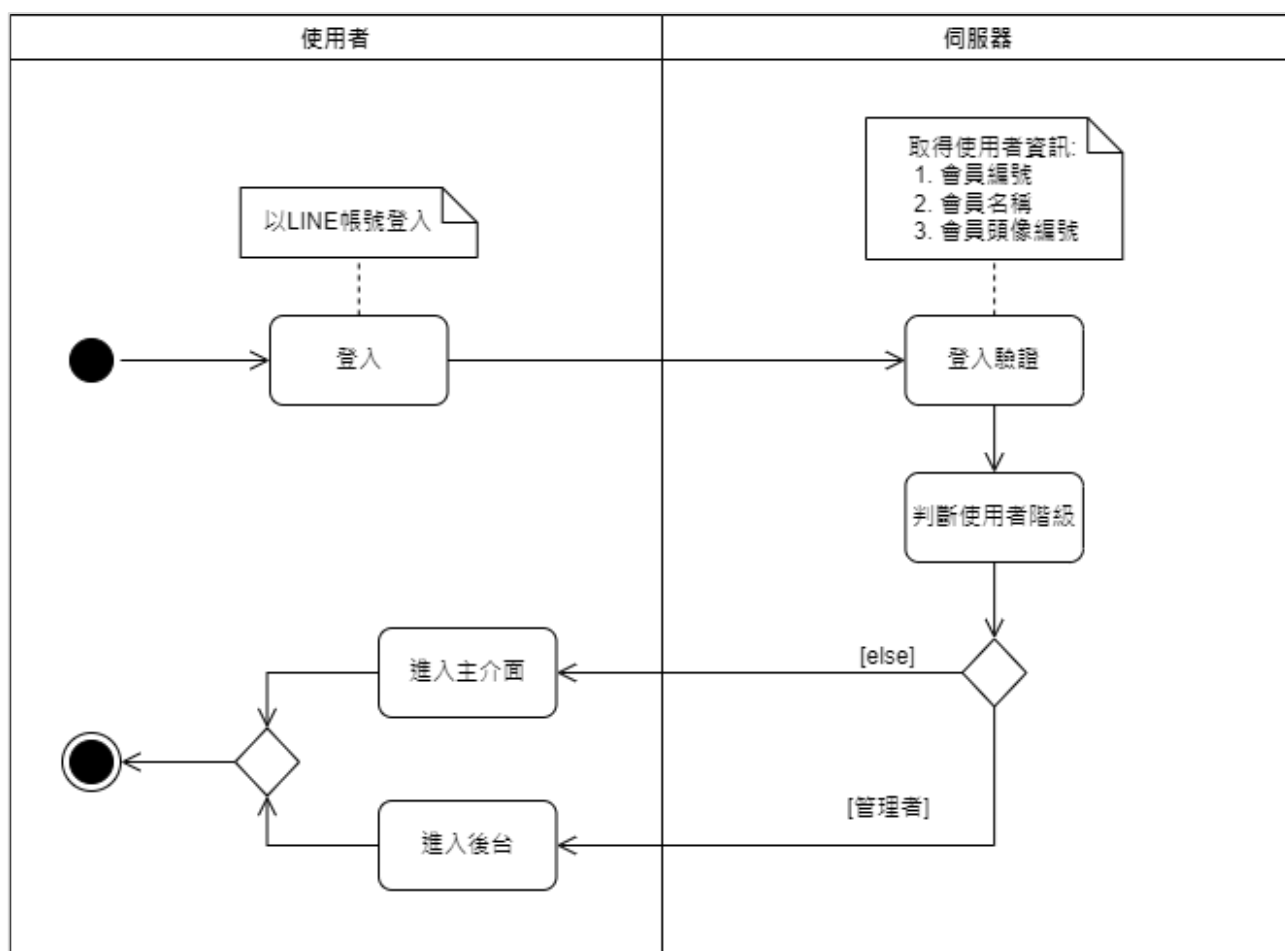
非功能性需求
裝置需求：Android 最低需求 4.1 版、iOS 最低需求 8.0 版，且手機須能上網(行動網路、Wi-Fi)及定位功能。
免安裝下載，須具備 LINE、連接網際網路，加入官方帳號即可。

5-2 使用個案圖(Use case diagram)

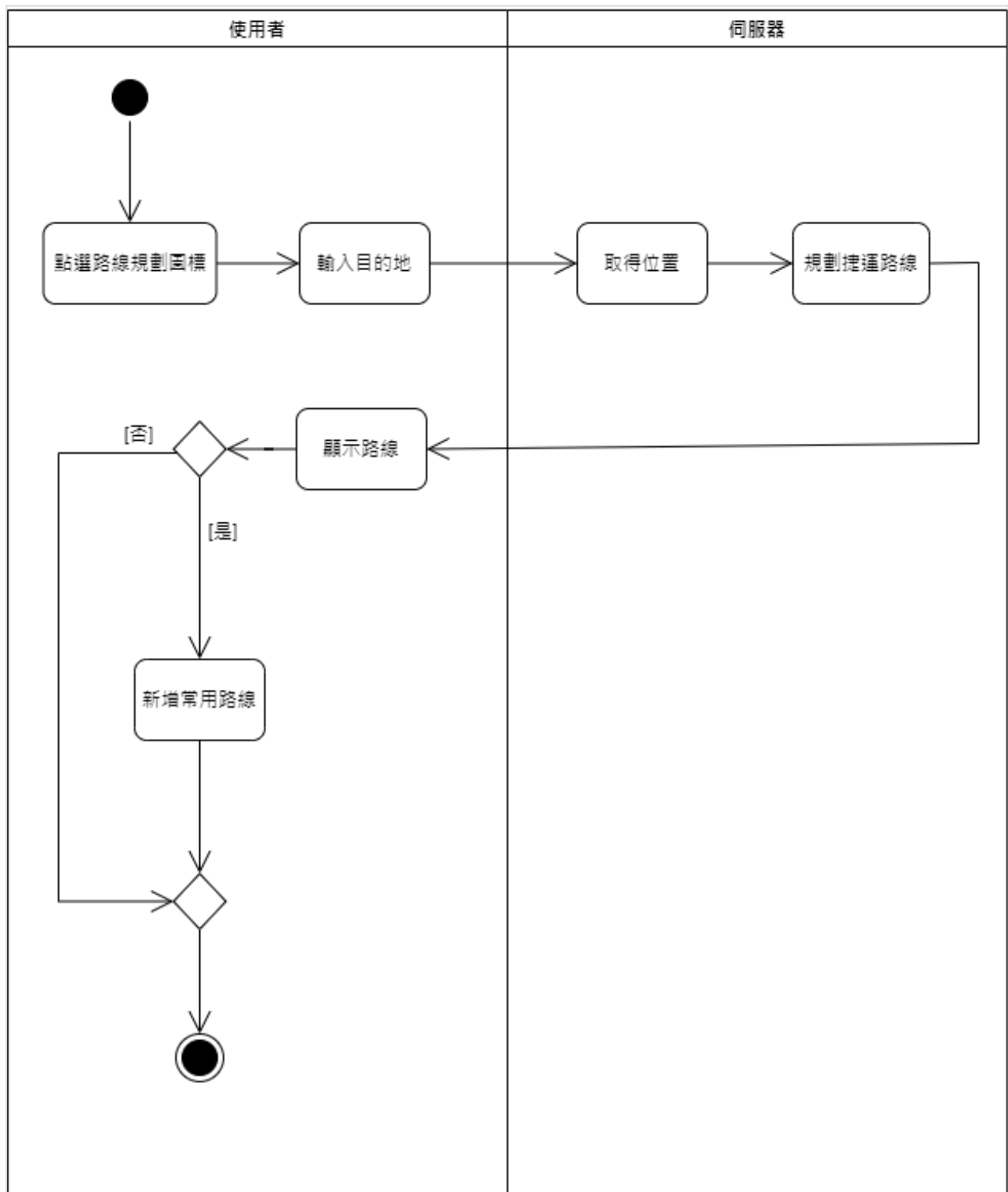


▲圖 5-2- 1、使用個案圖

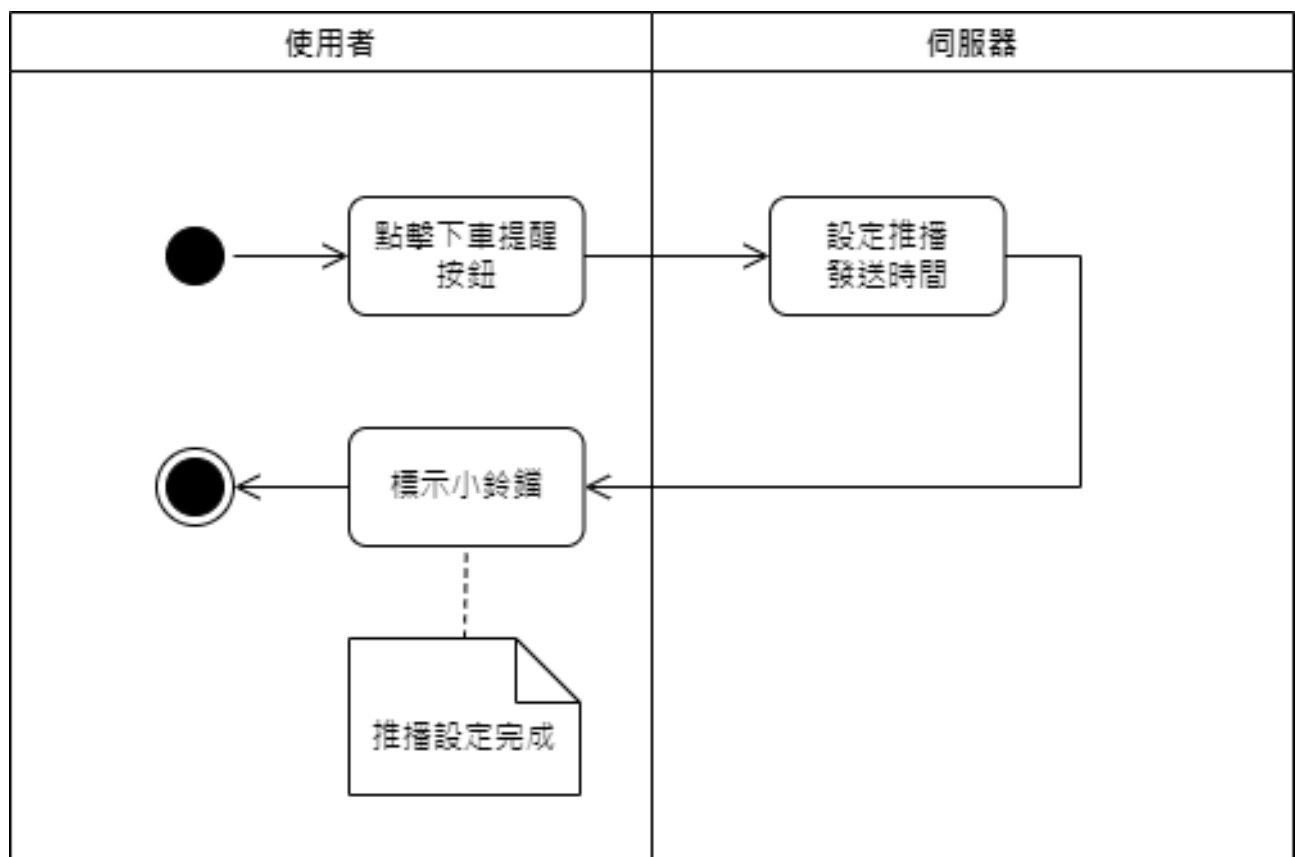
5-3 使用個案描述(Activity diagram)



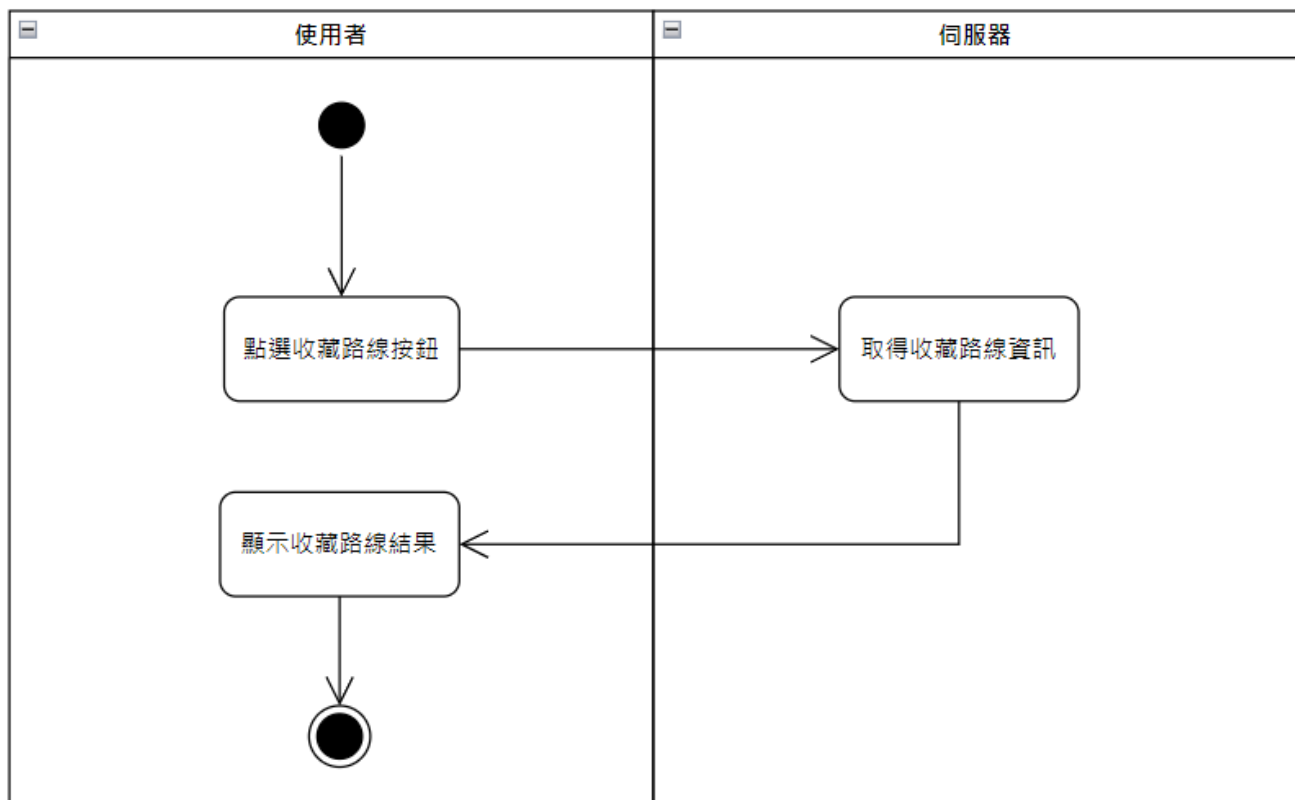
▲圖 5-3- 1、登入之活動圖



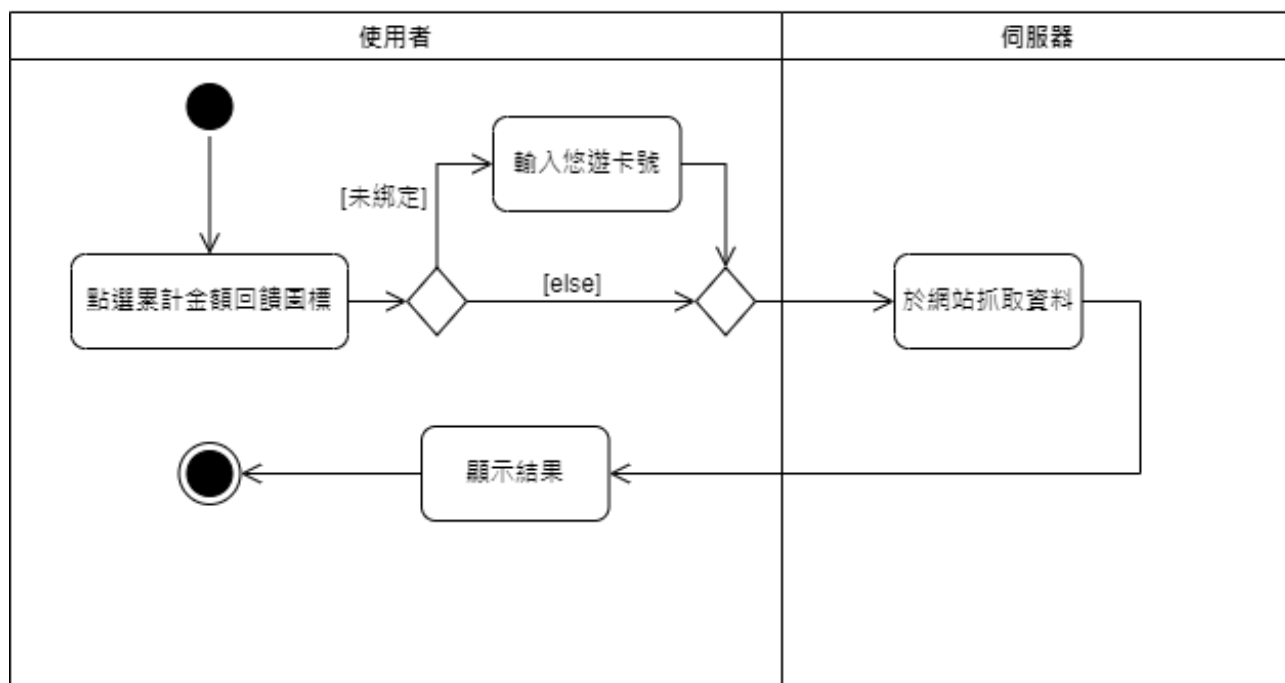
▲圖 5-3- 2、路線規劃之活動圖



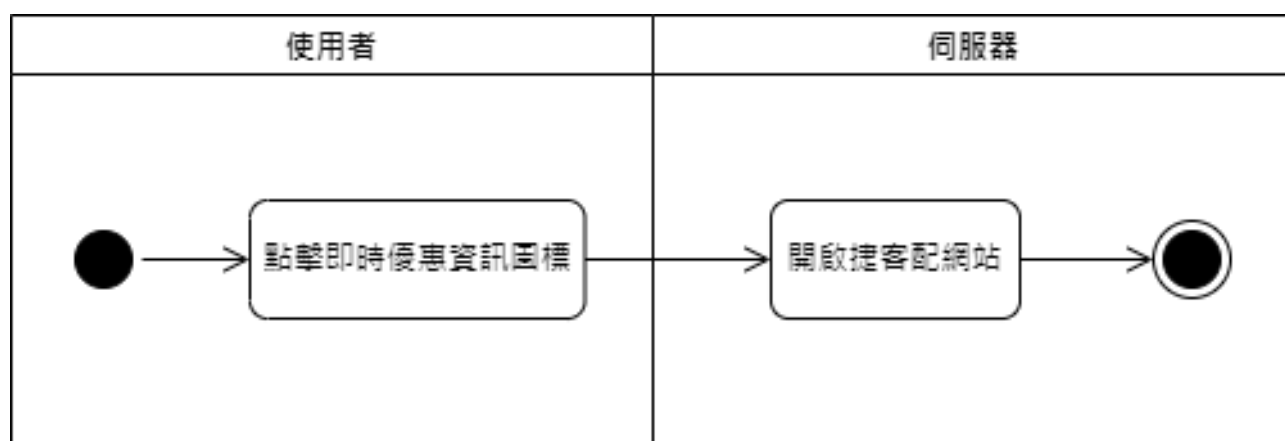
▲圖 5-3-3、下車提醒之活動圖



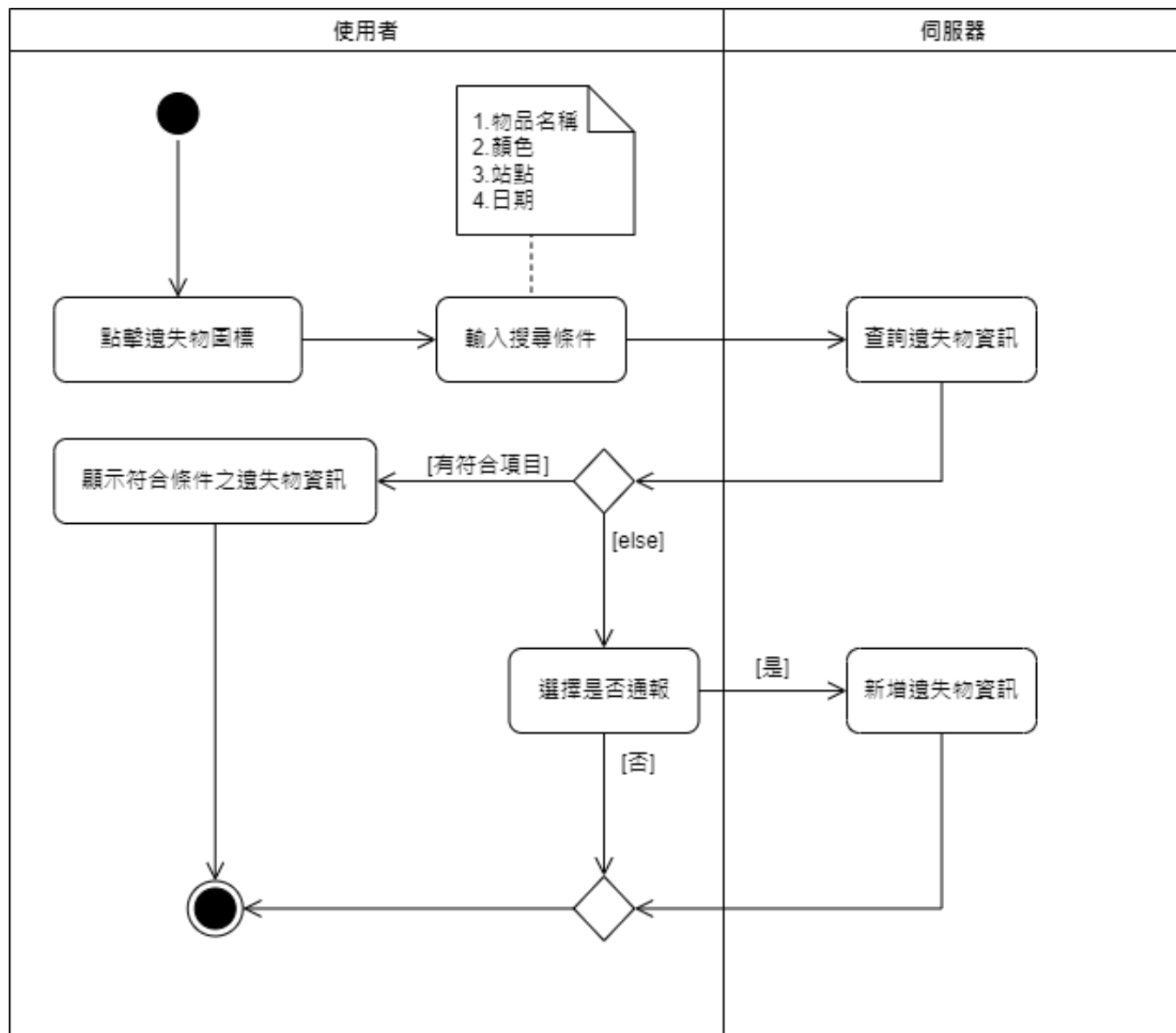
▲圖 5-3-4、收藏路線之活動圖



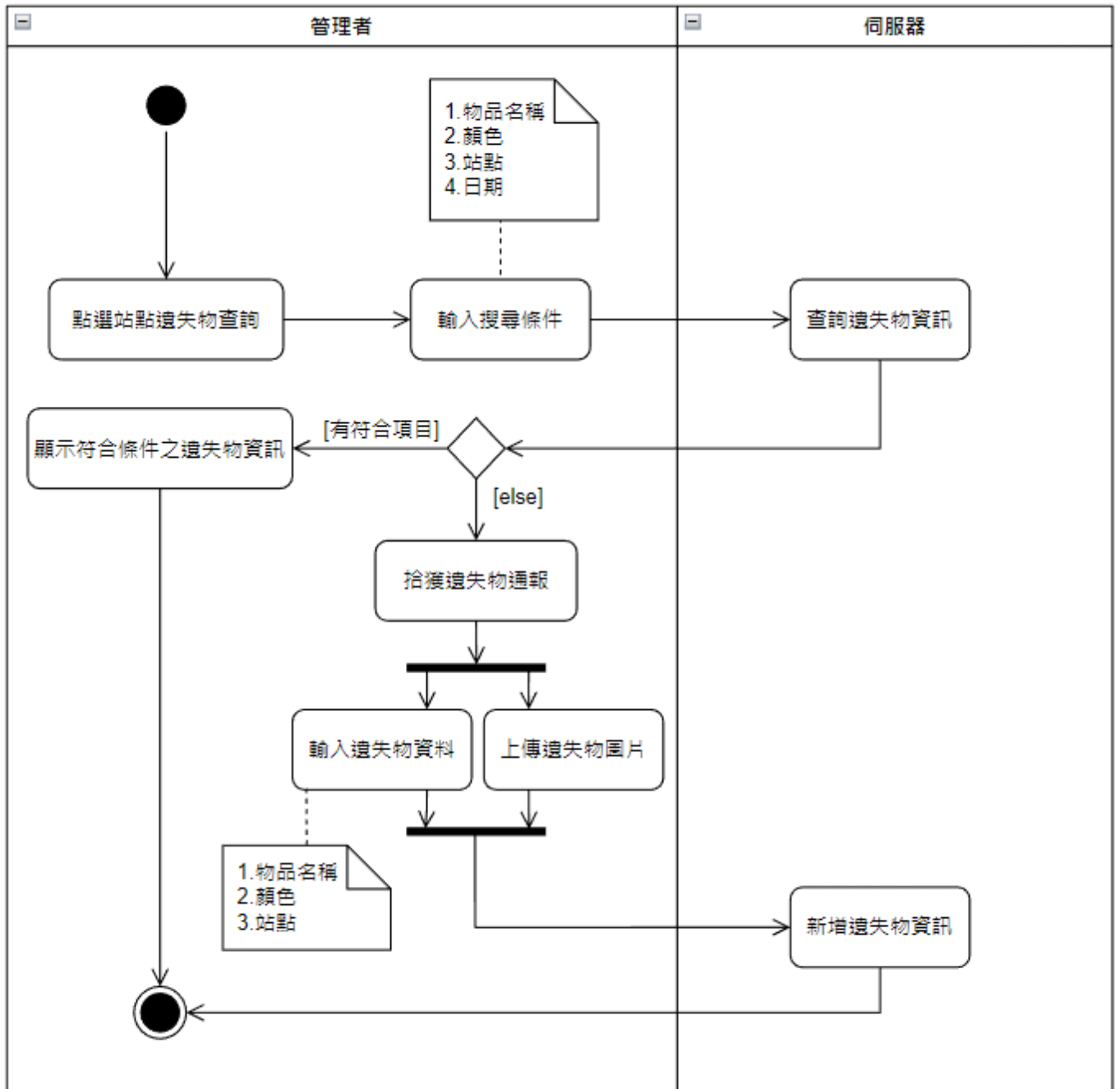
▲圖 5-3- 5、回饋金即時報之活動圖



▲圖 5-3- 6、優惠資訊之活動圖

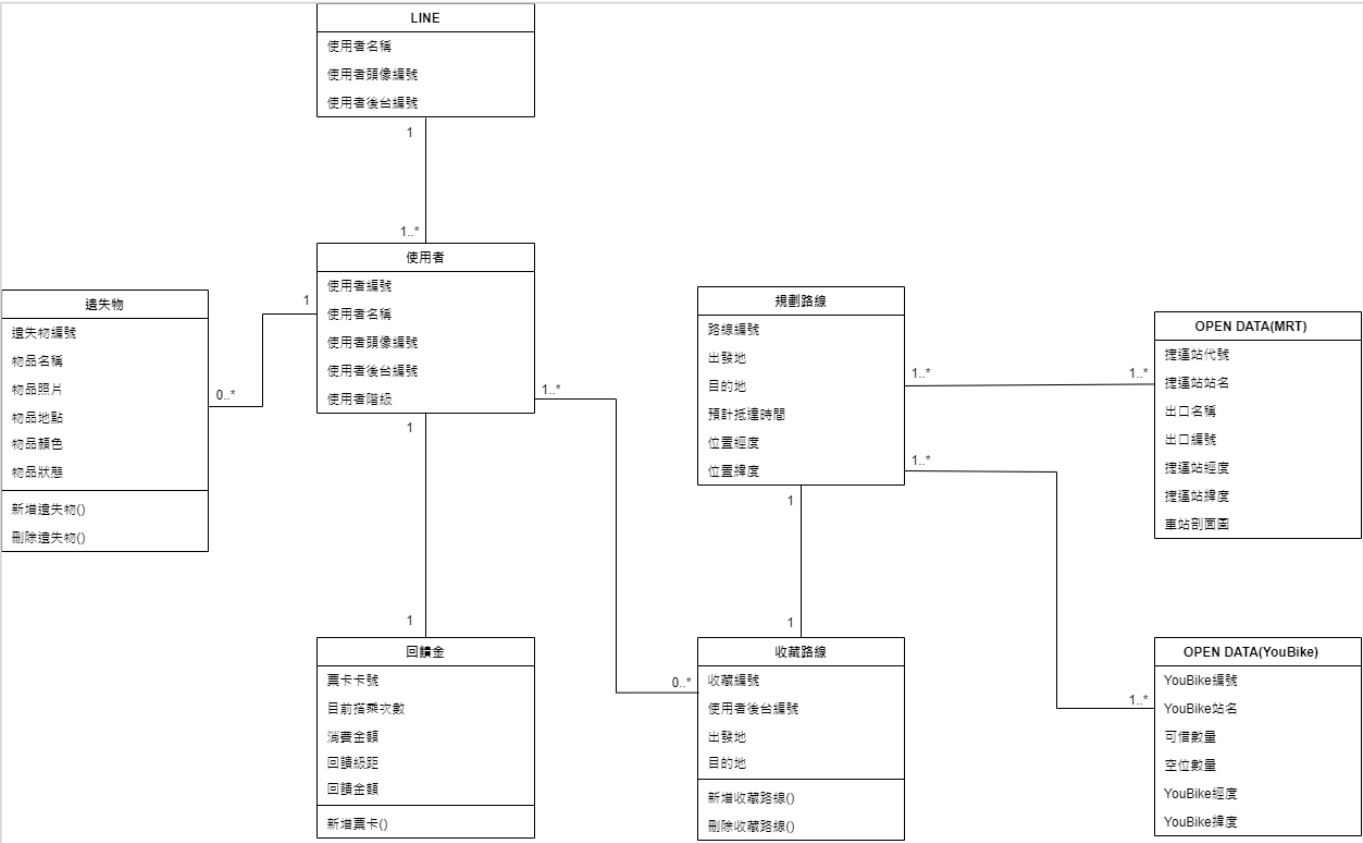


▲圖 5-3- 7、尋找遺失物(使用者)之活動圖



▲圖 5-3- 8、尋找遺失物(管理者)之活動圖

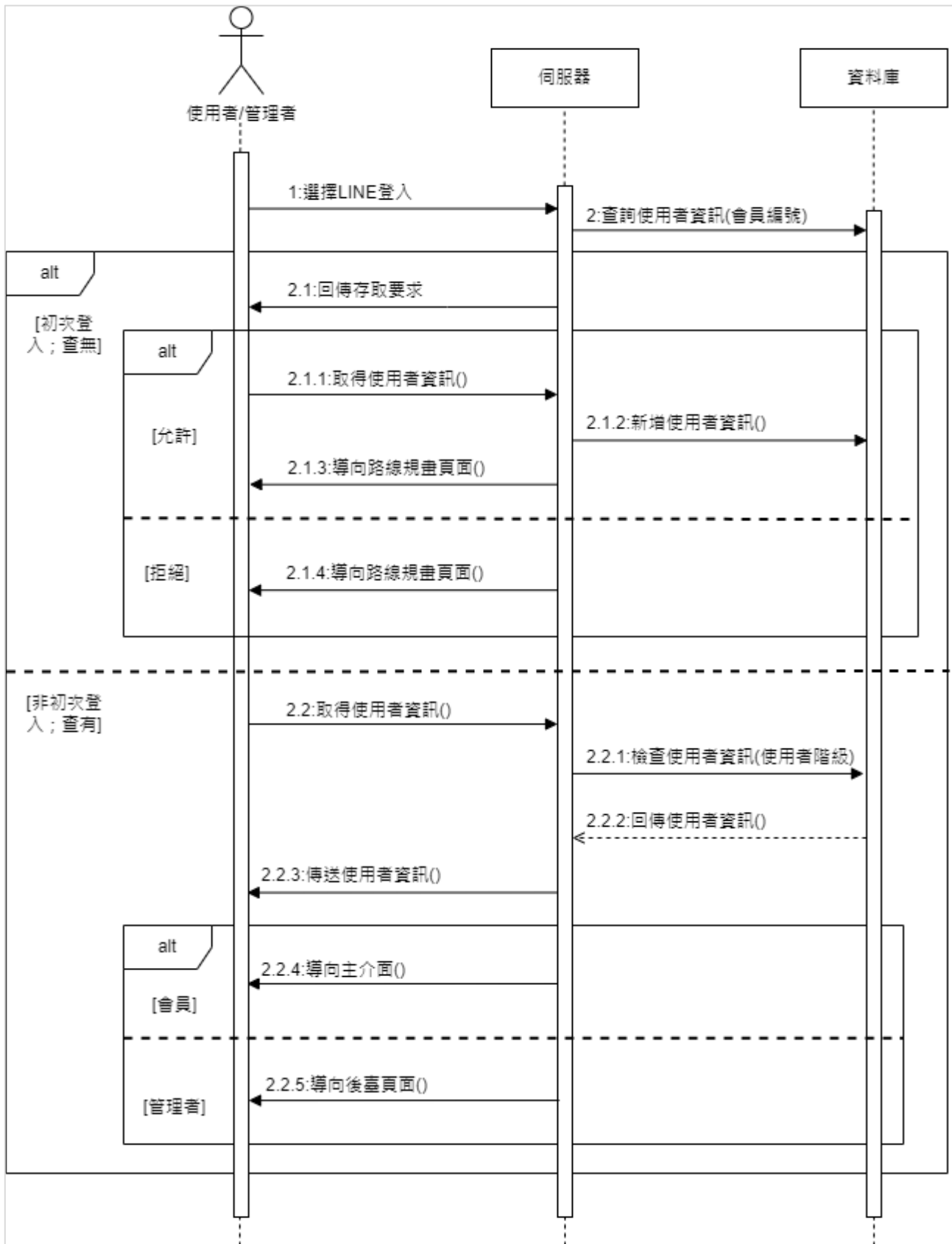
5-4 分析類別圖(Analysis class diagram)



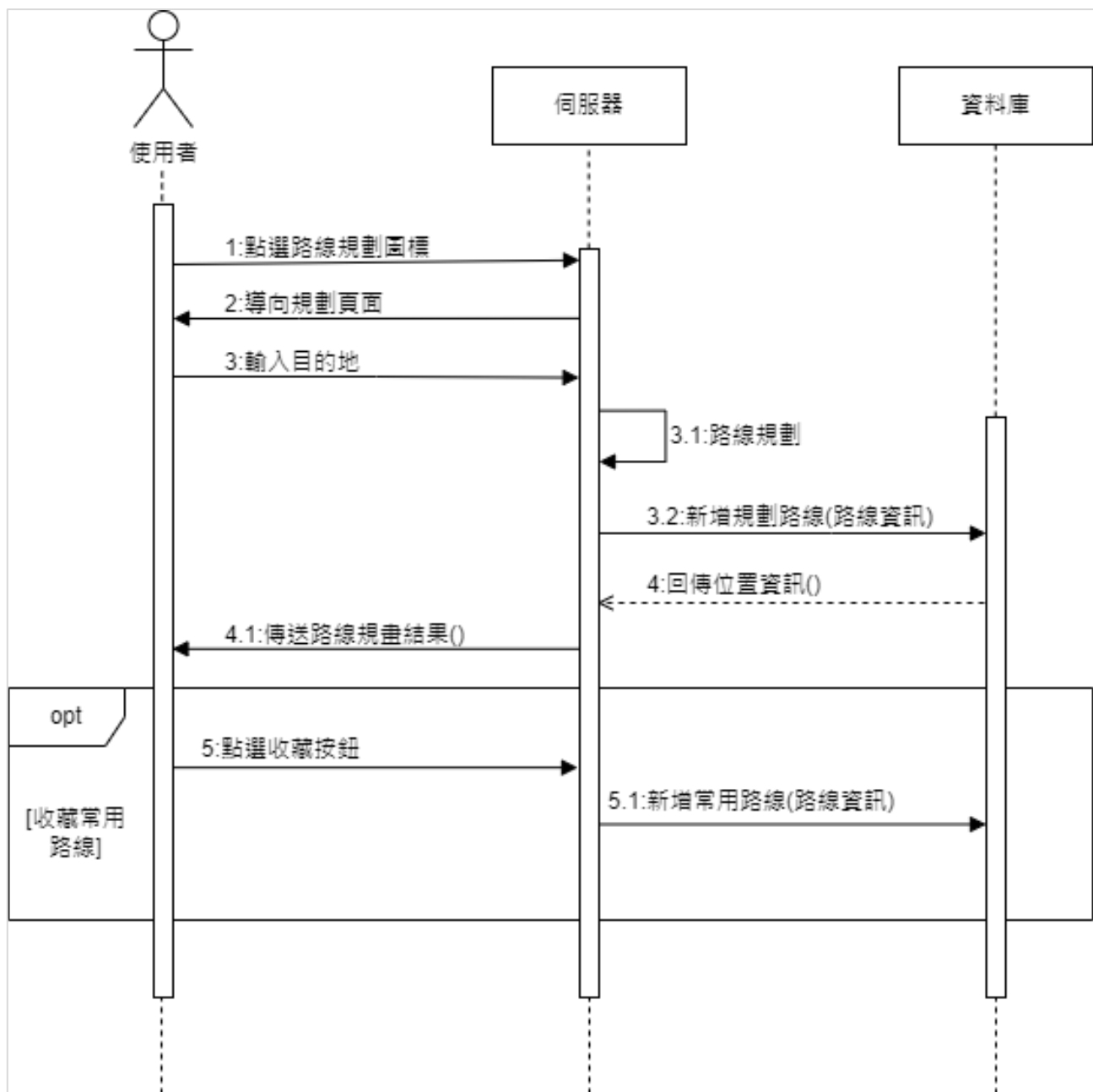
▲圖 5-4- 1、分析類別圖

第6章 設計模型

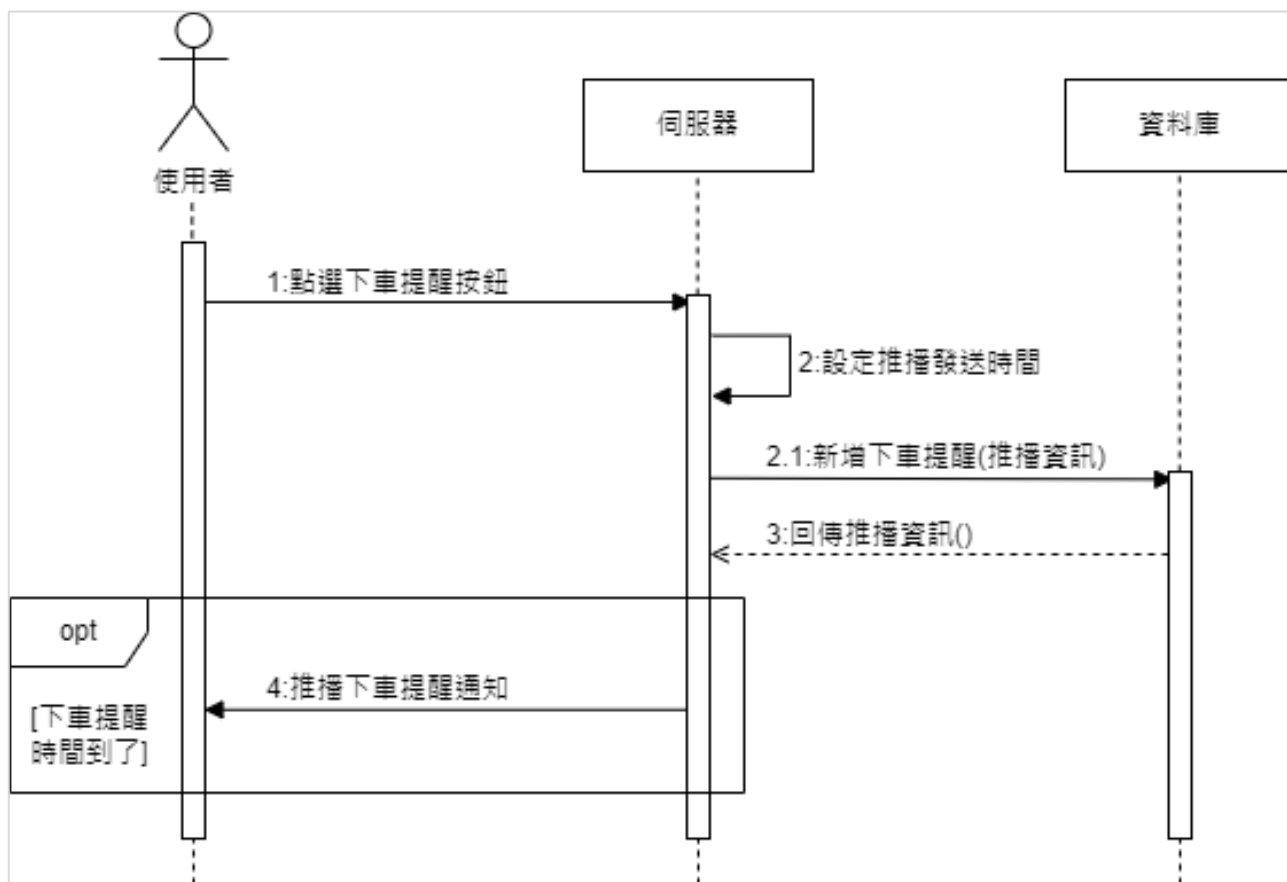
6-1 循序圖(Sequential diagram)



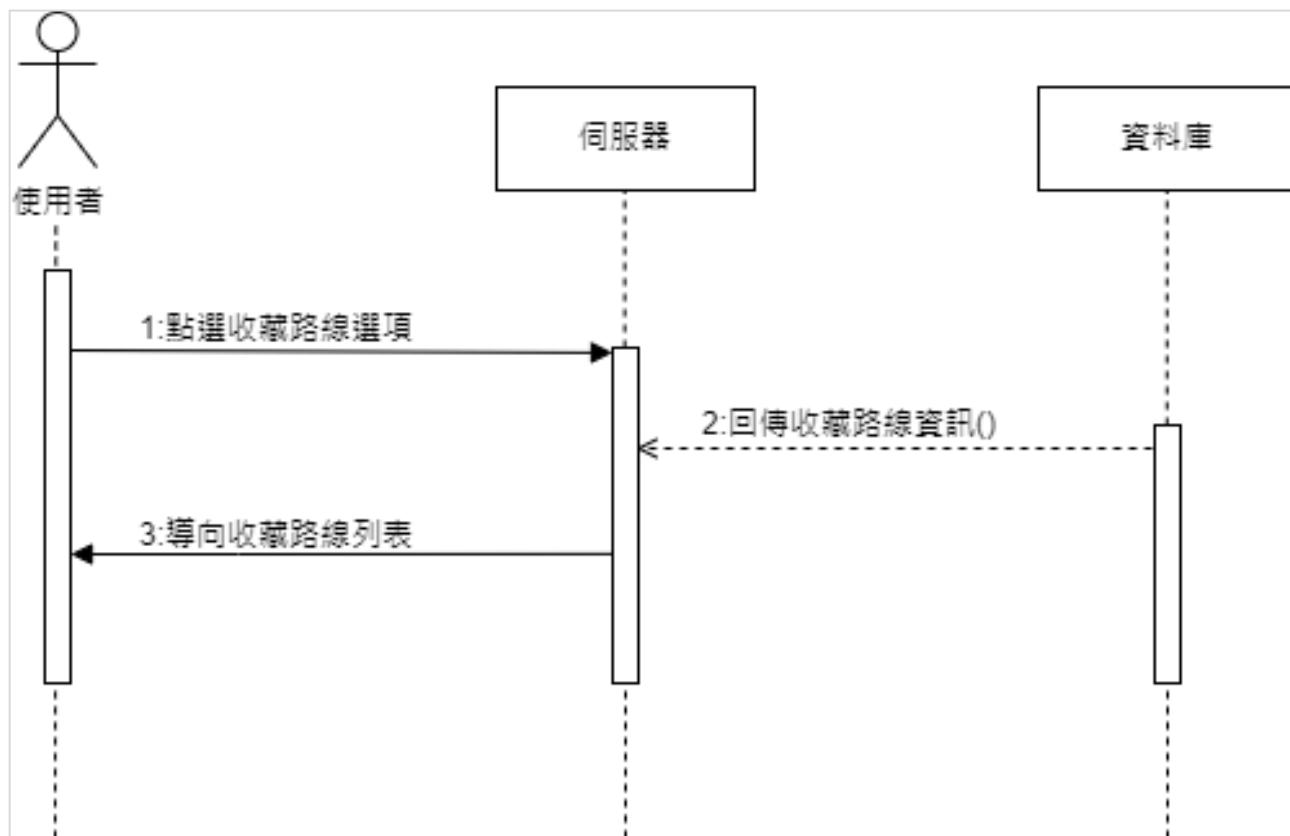
▲圖 6-1- 1、登入之循序圖



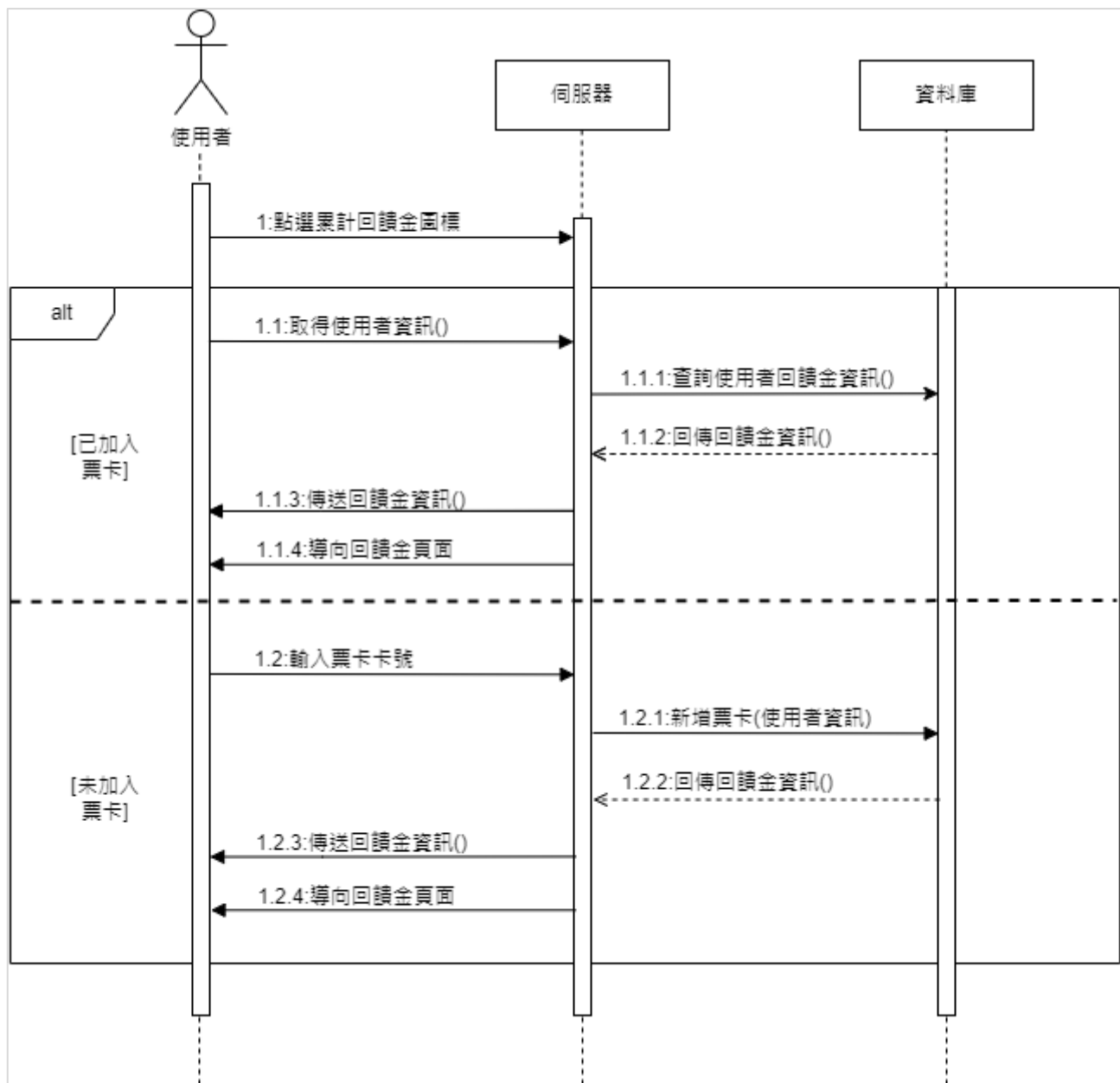
▲圖 6-1- 2、路線規劃之循序圖



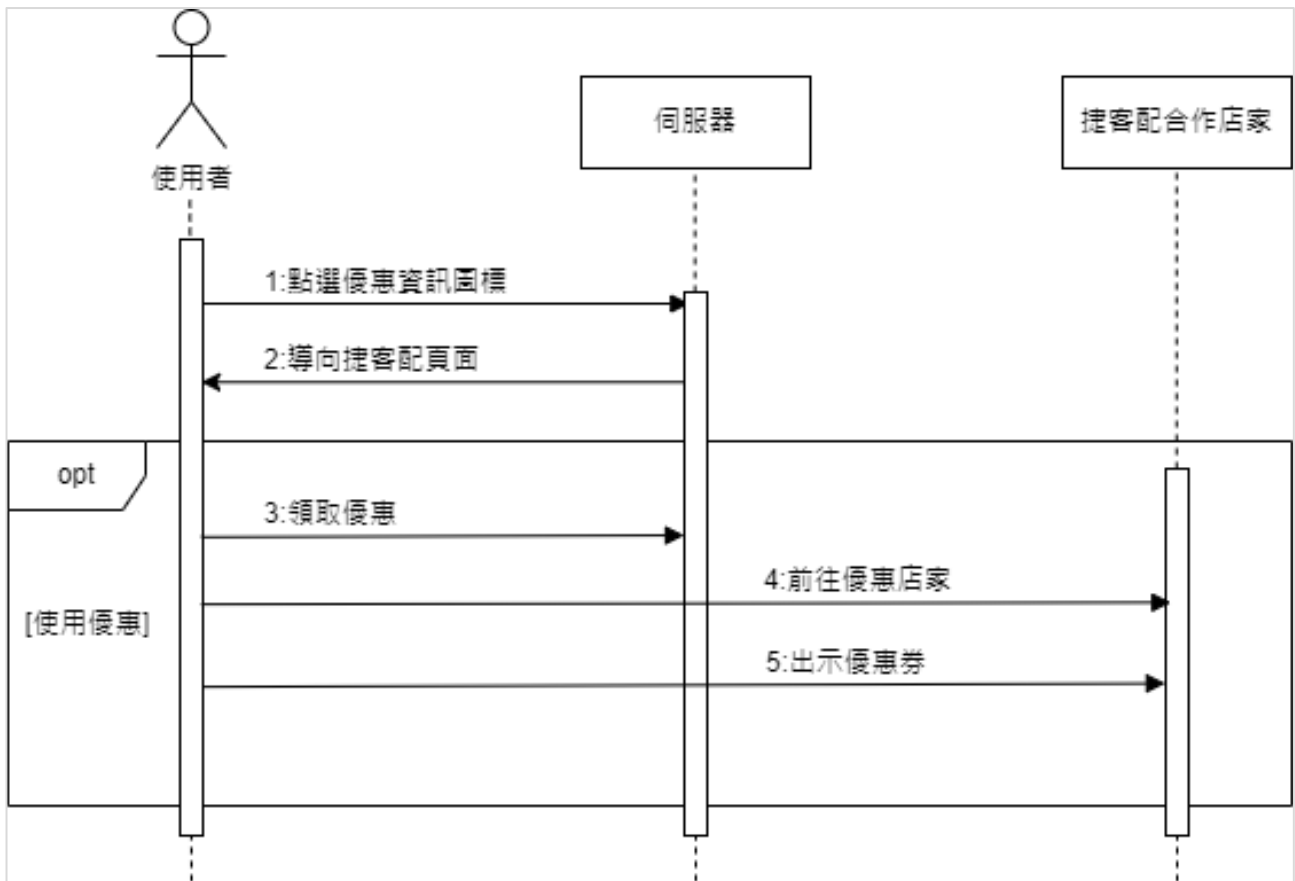
▲圖 6-1-3、下車提醒之循序圖



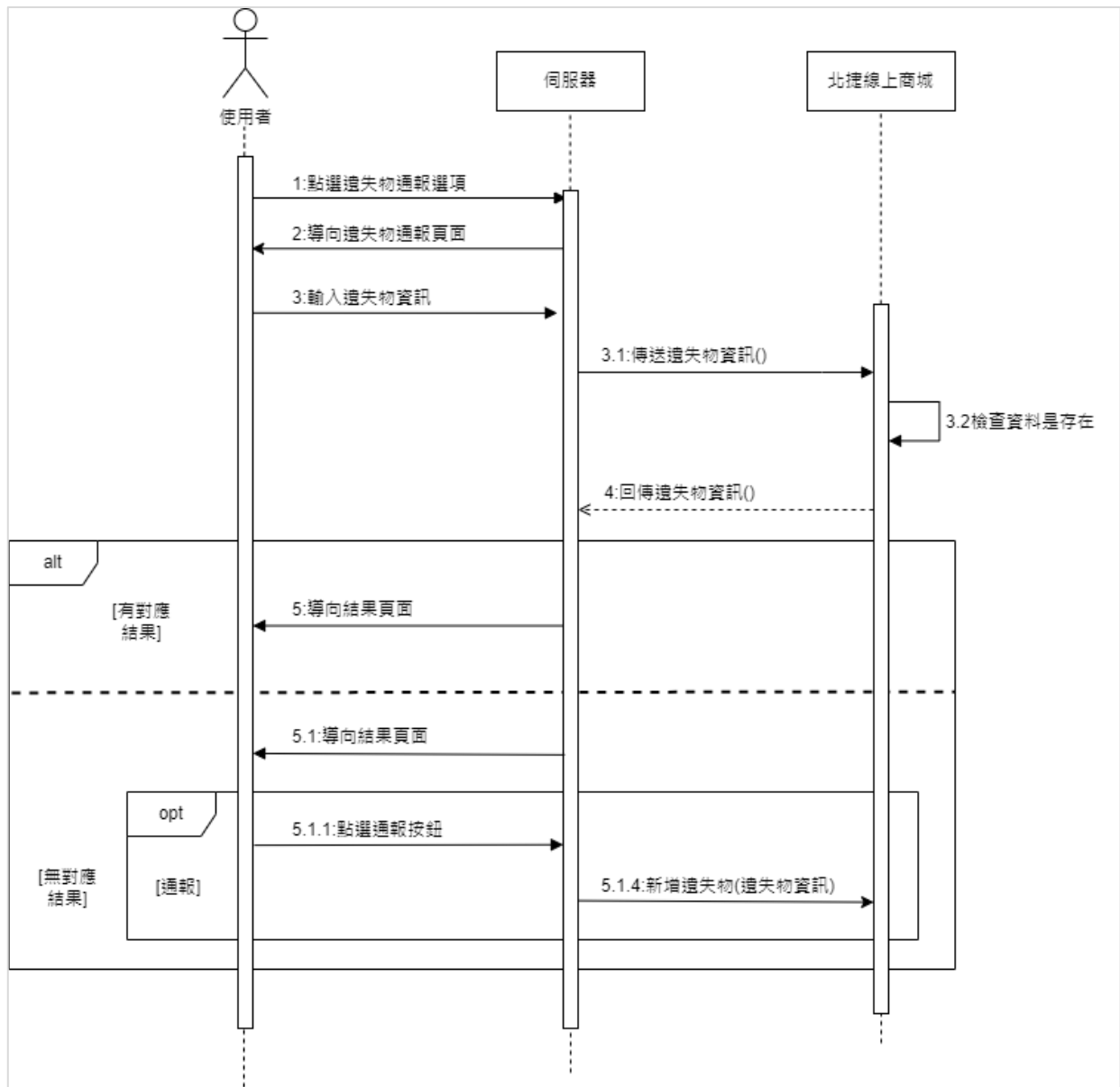
▲圖 6-1-4、收藏路線之循序圖



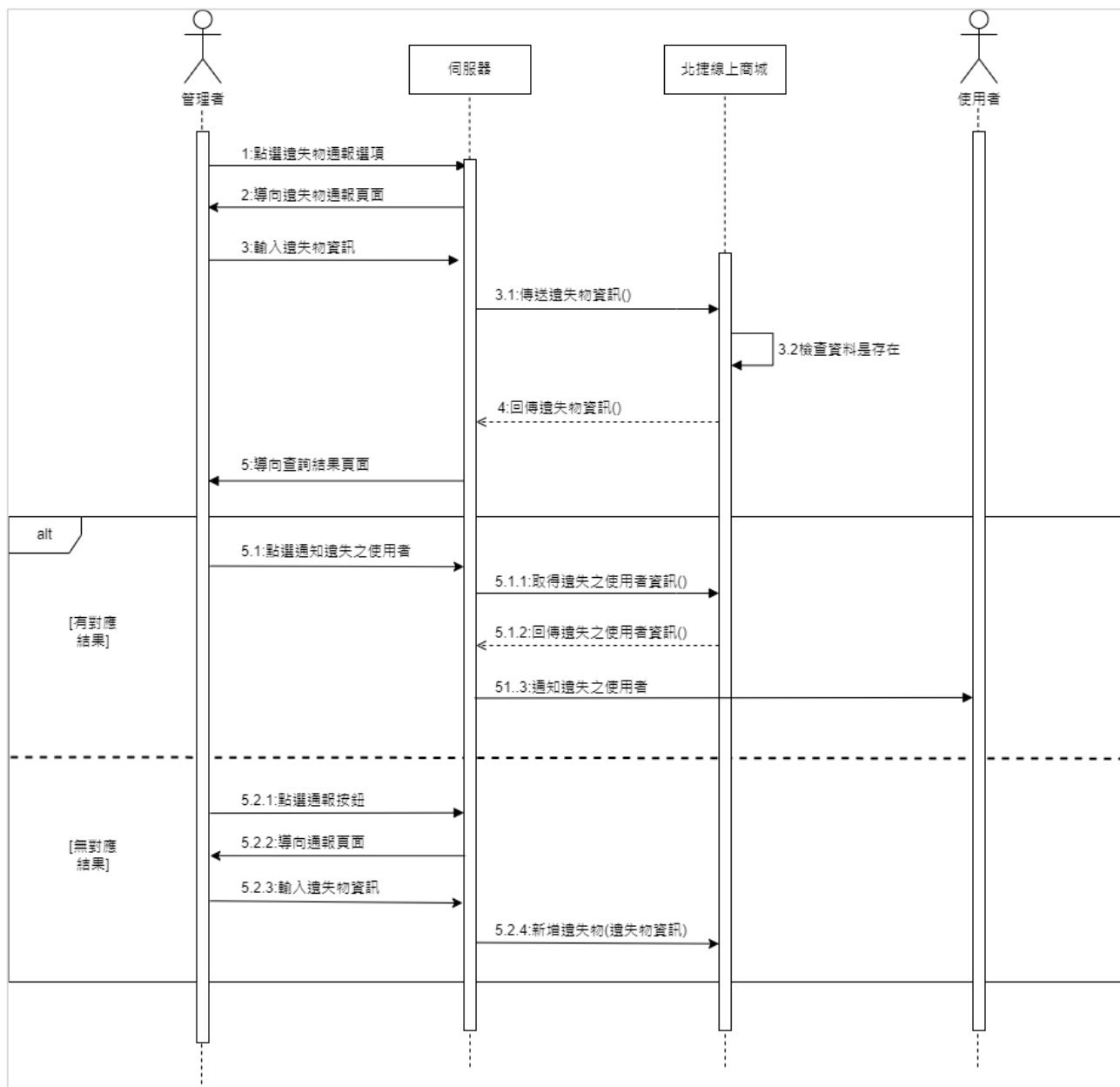
▲圖 6-1- 5、回饋金即時報之循序圖



▲圖 6-1-6、優惠資訊之循序圖

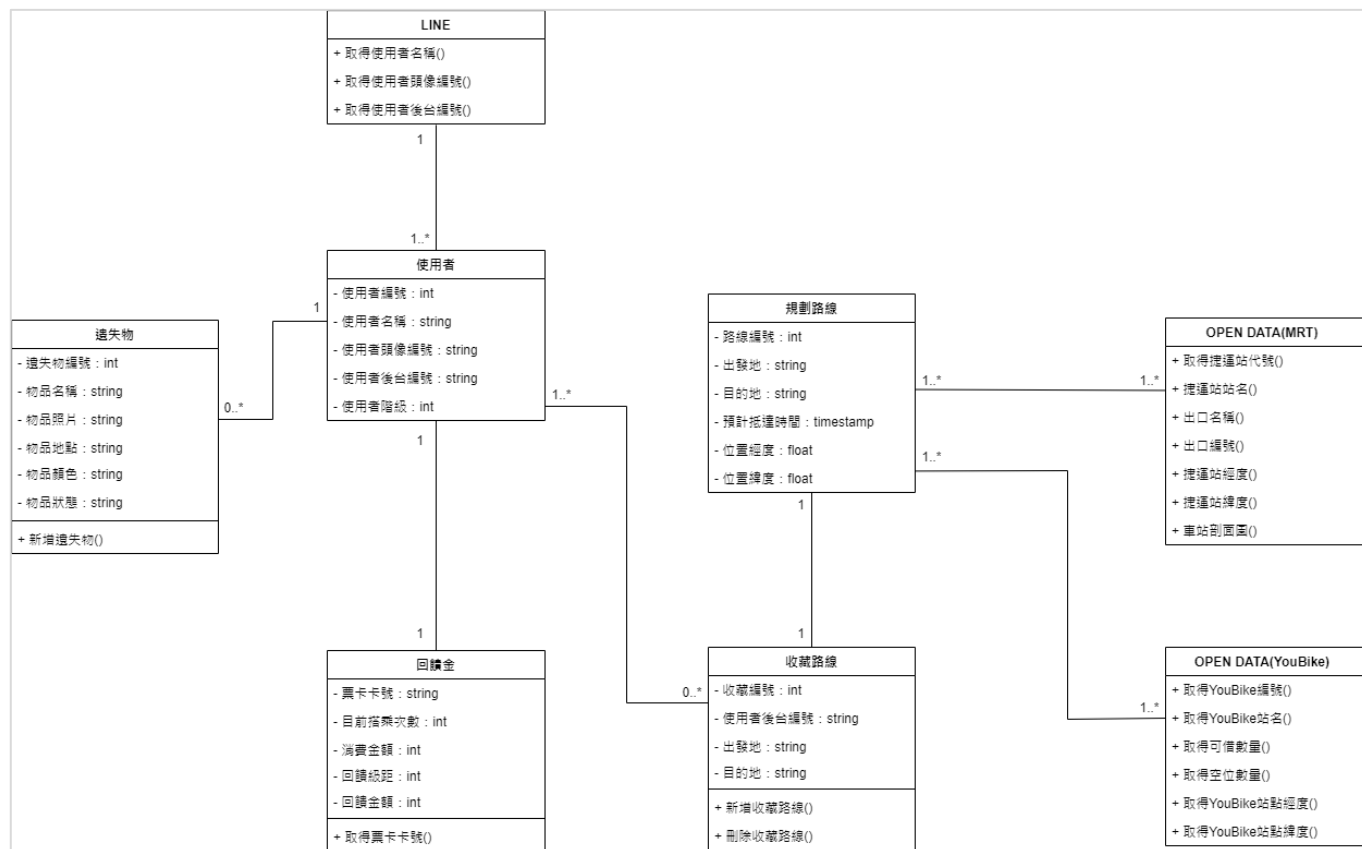


▲圖 6-1- 7、尋找遺失物(使用者)之循序圖



▲圖 6-1- 8、尋找遺失物(管理者)之循序圖

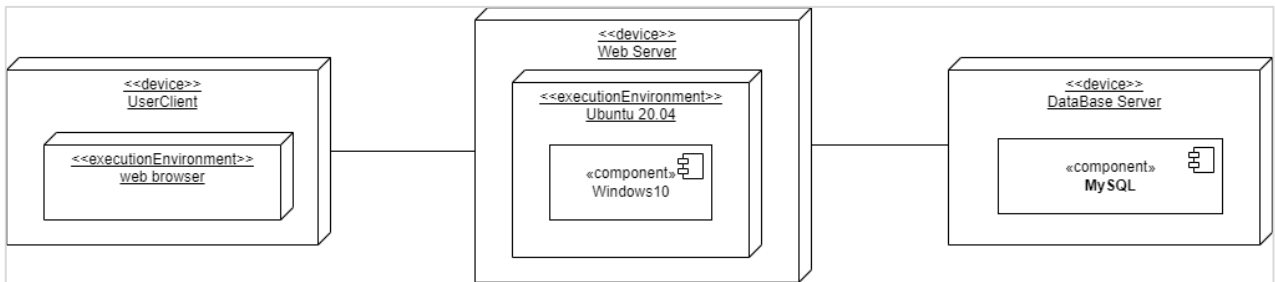
6-2 設計類別圖(Design class diagram)



▲圖 6-2- 1、設計類別圖

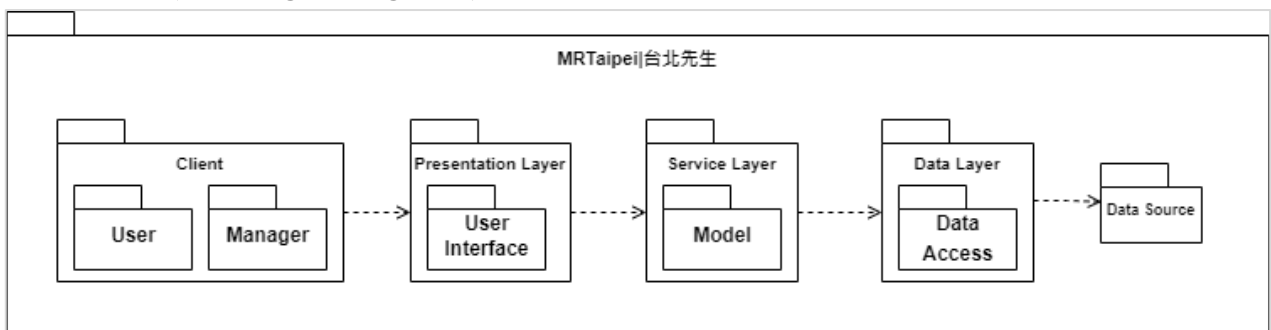
第7章 實作模型

7-1 佈署圖(Deployment diagram)



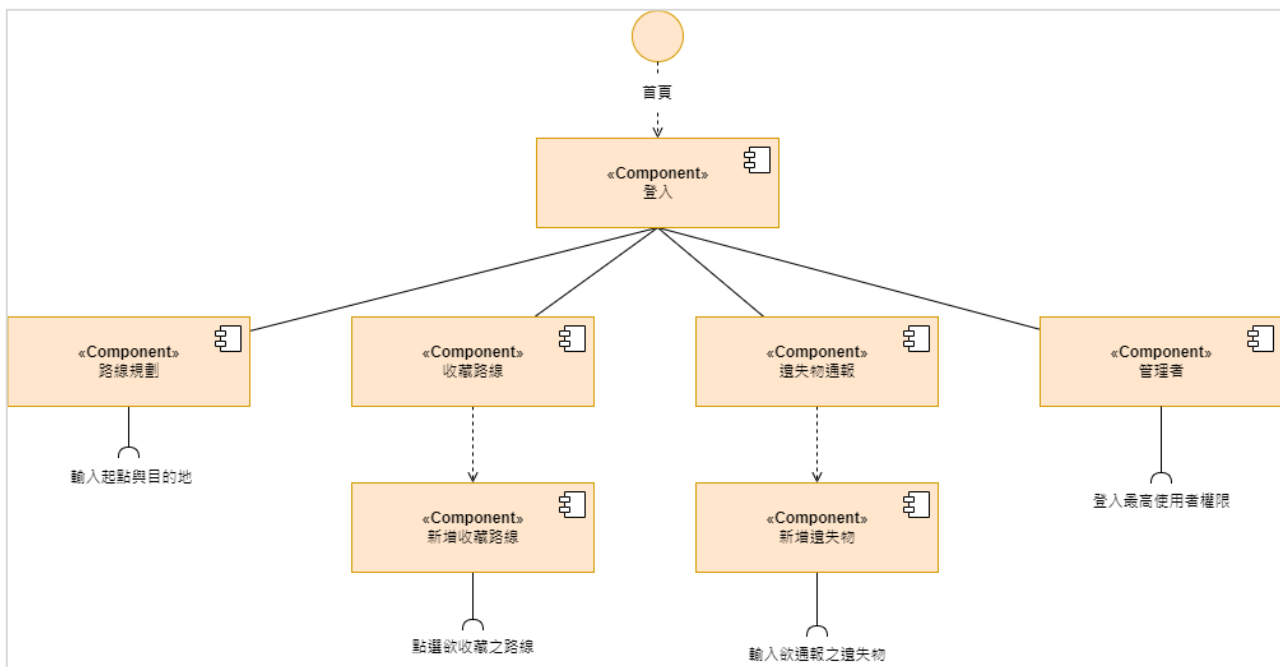
▲圖 7-1- 1、佈署圖

7-2 套件圖(Package diagram)



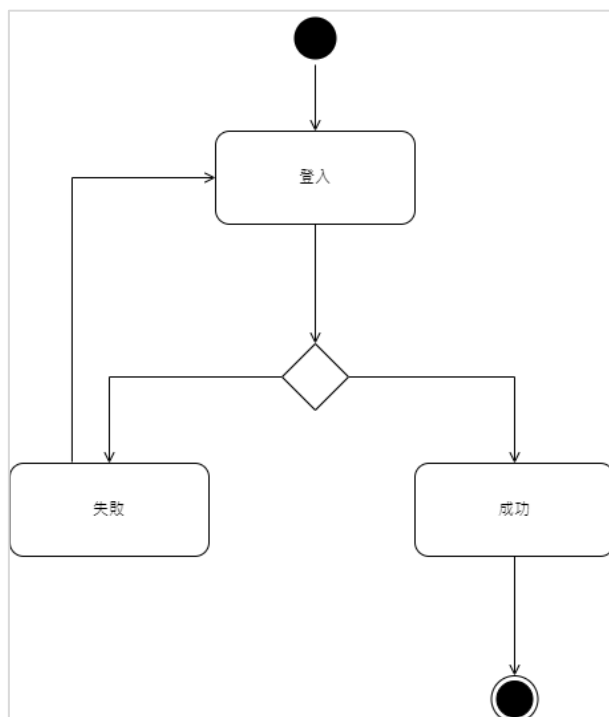
▲圖 7-2- 1、套件圖

7-3 元件圖(Component diagram)

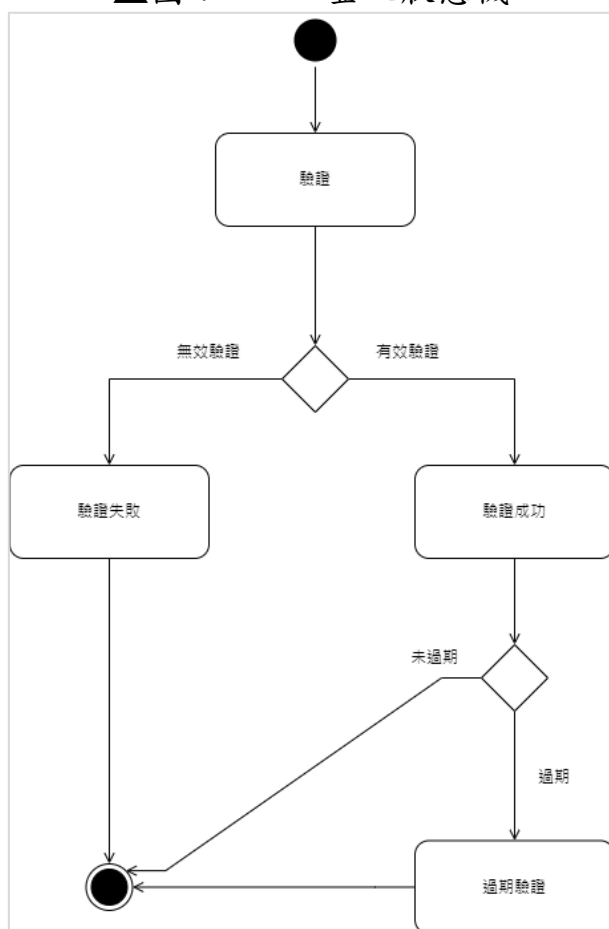


▲圖 7-3- 1、元件圖

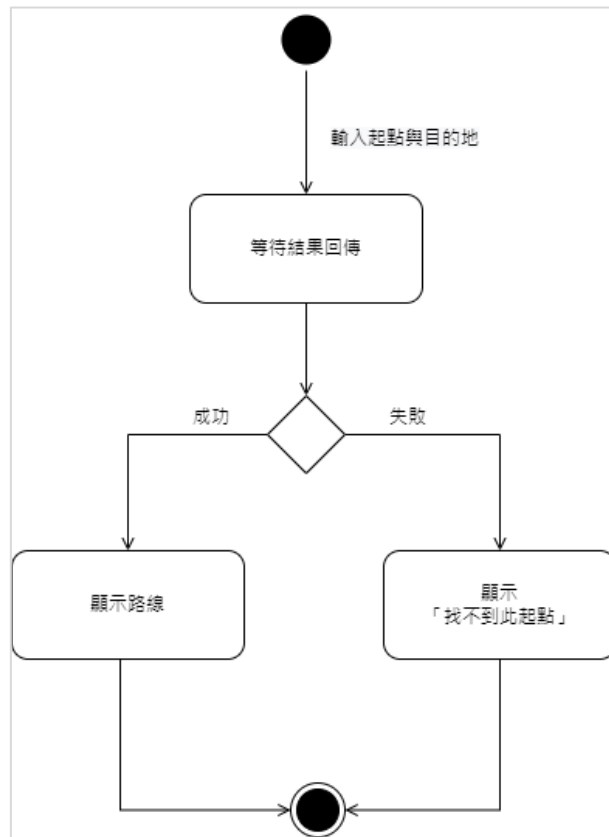
7-4 狀態機(State machine)



▲圖 7-4- 1、登入狀態機



▲圖 7-4- 2、驗證狀態機



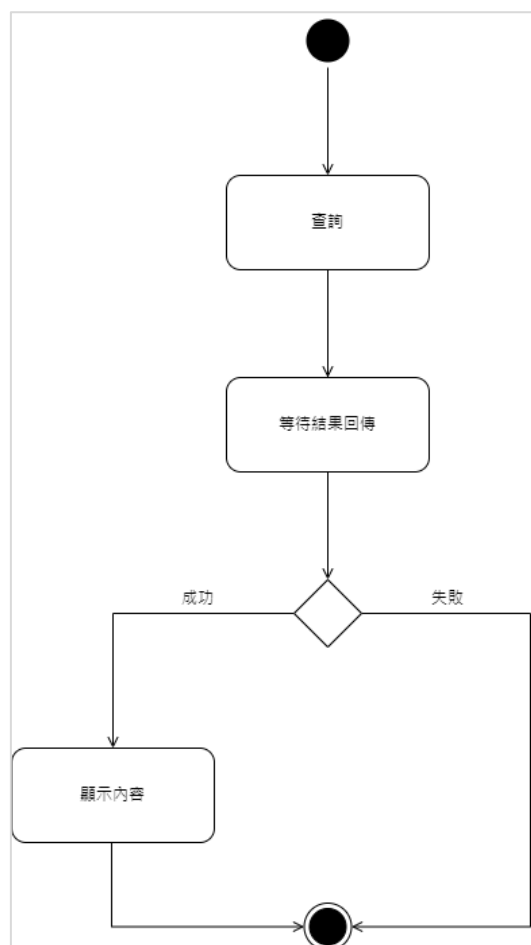
▲圖 7-4- 3、規劃路線狀態機



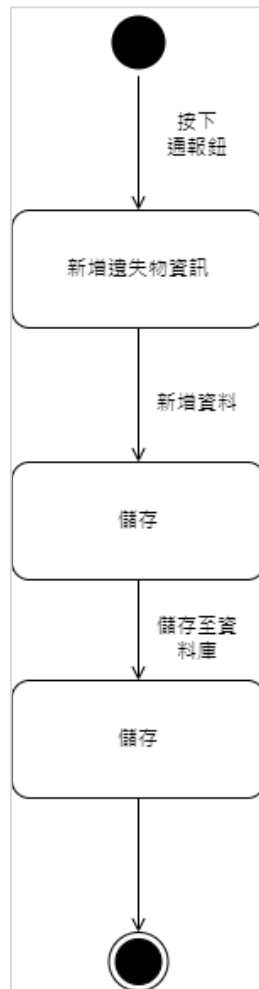
▲圖 7-4- 4、下車提醒狀態機



▲圖 7-4- 5、收藏路線狀態機



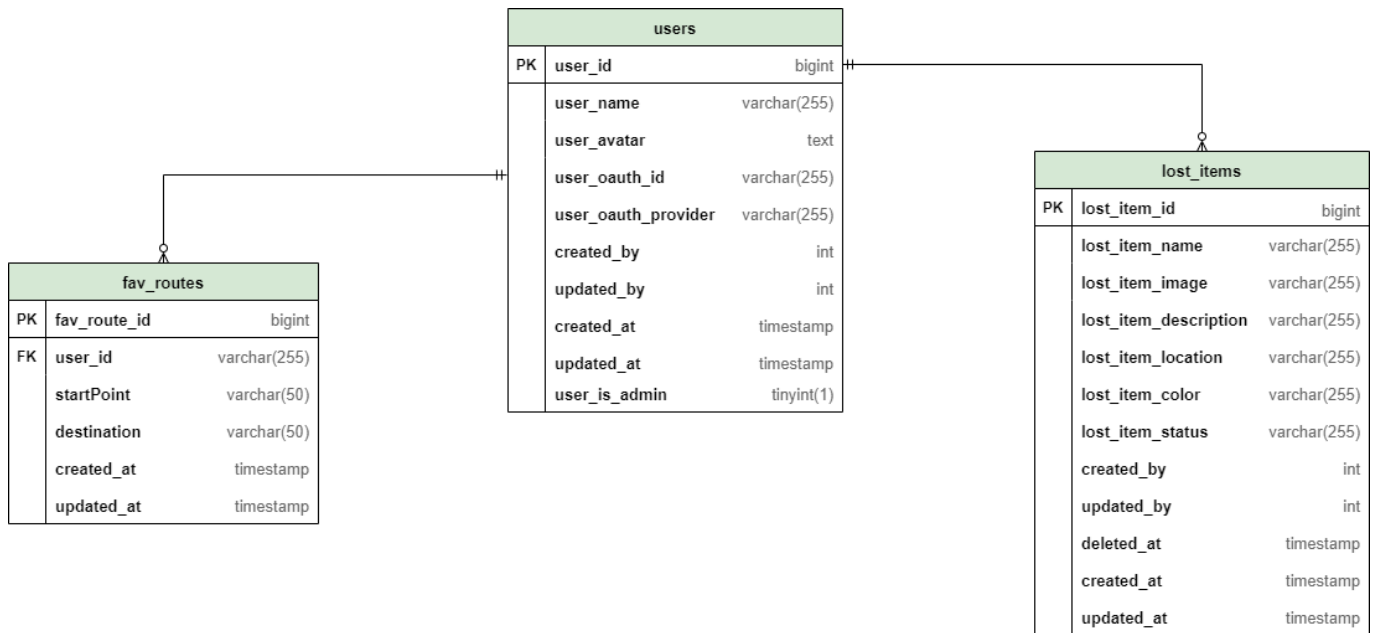
▲圖 7-4- 6、查詢遺失物狀態機



▲圖 7-4- 7、通報遺失物狀態機

第8章 資料庫設計

8-1 資料庫關聯表



▲圖 8-1- 1、資料庫關聯表

8-2 資料表及其 Meta data

▼表 8-2- 1、users-使用者 資料表

資料表英文名稱		users			
資料表中文名稱		使用者			
欄位名稱	意義	型態	空值	預設值	規則
*user_id	流水編號	bigint			AUTO_INCREMENT
user_name	使用者姓名	varchar(255)			
user_avatar	使用者頭像	text	V	NULL	
user_oauth_id	後台使用者編號	varchar(255)			
user_oauth_provider	授權提供者	varchar(255)			
created_by	建立者	int			
updated_by	更新者	int			
created_at	建立時間	timestamp			
updated_at	更新時間	timestamp			
user_is_admin	管理員權限	tinyint(1)		0	

▼表 8-2- 2、fav_routes -收藏路線 資料表

資料表英文名稱		fav_routes			
資料表中文名稱		收藏路線			
欄位名稱	意義	型態	空值	預設值	規則
*fav_route_id	流水編號	bigint			AUTO_INCREMENT
user_id	後台使用者編號	varchar(255)			
startPoint	出發地	varchar(50)			
destination	目的地	varchar(50)			
created_at	建立時間	timestamp			
updated_at	更新時間	timestamp			

▼表 8-2- 3、lost_items -遺失物 資料表

資料表英文名稱		lost_items			
資料表中文名稱		遺失物			
欄位名稱	意義	型態	空值	預設值	規則
*lost_item_id	流水編號	bigint			AUTO_INCREMENT
lost_item_name	物品名稱	varchar(255)			
lost_item_image	物品圖片	varchar(255)			
lost_item_description	物品描述	varchar(255)	V	NULL	
lost_item_location	物品地點	varchar(255)			
lost_item_color	物品顏色	varchar(255)			
lost_item_status	物品狀態	varchar(255)			
created_by	建立者	int			
updated_by	更新者	int			
deleted_at	刪除時間	timestamp			
created_at	建立時間	timestamp			
updated_at	更新時間	timestamp			

第9章 程式

9-1 元件清單及其規格描述

▼表 9-1- 1、後端程式清單

編號	檔案名稱	功能說明
1		
2	2022_03_12_000000_create_users_table.php	Laravel 建立 user 資料表腳本
3	2022_09_23_144850_create_fav_routes_table.php	Laravel 建立收藏路線資料表腳本
4	2022_10_24_025904_create_lost_items_table.php	Laravel 建立遺失物資料表腳本
5	web.php	資料庫 API, 路由 (LIFF 登入驗證、收藏路線、遺失物)
6	FavRouteController.php	收藏路線寫入、讀取 function
7	AuthController.php	LIFF Login 登入驗證、寫入頭貼
8	LostItemController.php	遺失物寫入、編修 function
9	api.php	站內路線 Dijkstra 最短路徑演算法計算、OpenCV 繪製路線 Flask API
10	objectDetectionAPI.py	讀取圖片，將其物品與背景分離，並丟入模型，取得名稱與顏色，將其翻譯成中文，回傳前端顯示

● 後端程式規格描述

路徑	database/migrations/2022_03_12_000000_create_users_table.php
用途	Laravel 建立 user 資料表腳本
部分程式碼	
<pre> 1. use Illuminate\Database\Migrations\Migration; 2. use Illuminate\Database\Schema\Blueprint; 3. use Illuminate\Support\Facades\Schema; 4. 5. return new class extends Migration 6. { 7. /** 8. * Run the migrations. 9. * 10. * @return void 11. */ 12. public function up() 13. { 14. Schema::create('users', function (Blueprint \$table) { 15. \$table->id("user_id"); 16. \$table->string('user_name'); 17. \$table->text('user_avatar')->nullable(); 18. \$table->string('user_oauth_id'); 19. \$table->string('user_oauth_provider'); 20. \$table->string('user_is_admin'); 21. \$table->tinyint("created_by"); 22. \$table->integer("updated_by"); 23. \$table->timestamps(); 24. }); 25. } 26. 27. /** 28. * Reverse the migrations. 29. * 30. * @return void 31. */ 32. public function down() 33. { 34. Schema::dropIfExists('users'); 35. } 36. };</pre>	

路徑	database/migrations/2022_09_23_144850_create_fav_routes_table.php
用途	Laravel 建立收藏路線資料表腳本
部分程式碼	
<pre> 1. use Illuminate\Database\Migrations\Migration; 2. use Illuminate\Database\Schema\Blueprint; 3. use Illuminate\Support\Facades\Schema; 4. 5. class CreateFavRoutesTable extends Migration 6. { 7. /** 8. * Run the migrations. 9. * 10. * @return void 11. */ 12. public function up() 13. { 14. Schema::create('fav_routes', function (Blueprint \$table) { 15. \$table->id('fav_route_id'); 16. \$table->string('user_id',255); 17. \$table->string('startPoint',50); 18. \$table->string('destination',50); 19. \$table->timestamps(); 20. }); 21. } 22. 23. /** 24. * Reverse the migrations. 25. * 26. * @return void 27. */ 28. public function down() 29. { 30. Schema::dropIfExists('fav_routes'); 31. } 32. }</pre>	

路徑	database/migrations/2022_10_24_025904_create_lost_items_table.php
用途	Laravel 建立遺失物資料表腳本
部分程式碼	
<pre> 1. use Illuminate\Database\Migrations\Migration; 2. use Illuminate\Database\Schema\Blueprint; 3. use Illuminate\Support\Facades\Schema; 4. 5. return new class extends Migration 6. { 7. /** 8. * Run the migrations. 9. * 10. * @return void 11. */ 12. public function up() 13. { 14. Schema::create('lost_items', function (Blueprint \$table) { 15. \$table->id('lost_item_id'); 16. \$table->string('lost_item_name')->nullable(); 17. \$table->string('lost_item_image')->nullable(); 18. \$table->string('lost_item_description')->nullable(); 19. \$table->string('lost_item_location')->nullable(); 20. \$table->string('lost_item_color')->nullable(); 21. \$table->string('lost_item_status')->nullable(); 22. \$table->integer('created_by')->default(-1); 23. \$table->integer('updated_by')->default(-1); 24. \$table->softDeletes(); 25. \$table->timestamps(); 26. }); 27. } 28. 29. /** 30. * Reverse the migrations. 31. * 32. * @return void 33. */ 34. public function down() 35. { 36. Schema::dropIfExists('lost_items'); 37. } 38. };</pre>	

路徑	routes/web.php
用途	資料庫 API, 路由(LIFF 登入驗證、收藏路線、遺失物)
部分程式碼	
<pre> 1. Route::post('/api/auth/login', [App\Http\Controllers\AuthController::class, 'login']); 2. Route::get('/api/auth/status', [App\Http\Controllers\AuthController::class, 'status']); 3. 4. Route::post('/api/favRoute/store', [App\Http\Controllers\FavRouteController::class, 'store']); 5. Route::get('/api/favRoute/read', [App\Http\Controllers\FavRouteController::class, 'read']); 6. 7. Route::get('/api/lostItems', [App\Http\Controllers\LostItemController::class, 'list']); 8. 9. Route::middleware(['auth'])->group(function () { 10. Route::post('/api/lostItem', [App\Http\Controllers\LostItemController::class, 'create']); 11. Route::get('/api/lostItem/{lostItemID}', [App\Http\Controllers\LostItemController::class, 'read']); 12. Route::put('/api/lostItem/{lostItemID}', [App\Http\Controllers\LostItemController::class, 'update']); 13. Route::delete('/api/lostItem/{lostItemID}', [App\Http\Controllers\LostItemController::class, 'delete']); 14. }); </pre>	

路徑	app/Http/Controllers/FavRouteController.php
用途	收藏路線寫入、讀取 function
部分程式碼	
<pre> 1. namespace App\Http\Controllers; 2. 3. use App\Models\FavRoute; 4. use Illuminate\Http\Request; 5. 6. class FavRouteController extends Controller 7. { 8. /** 9. * Display a listing of the resource. 10. * 11. * @return \Illuminate\Http\Response 12. */ 13. public function index() 14. { 15. \$favRoute = FavRoute::all(); 16. return \$favRoute; 17. } </pre>	

```

18. /**
19. * Store a newly created resource in storage.
20. *
21. * @param \Illuminate\Http\Request $request
22. * @return \Illuminate\Http\Response
23. */
24. public function store(Request $request)
25. {
26. // $body = $request->getBody();
27. $body = json_decode($request);
28. $user = FavRoute::where('user_id', auth()->user()->user_oauth_id)->where('startPoint',
    $request->json()->get("startPoint"))->where('destination', $request->json()-
    >get("destination"))->first();
29. if ($user) {
30. $user->user_id = auth()->user()->user_oauth_id;
31. $user->startPoint = $request->json()->get("startPoint");
32. $user->destination = $request->json()->get("destination");
33. $user->delete();
34. return response()->json(['message' => '從收藏路線移除', 'data'=> $user]);
35. } else {
36. $favRoute = new FavRoute();
37. $favRoute ->user_id =auth()->user()->user_oauth_id;
38. // $favRoute ->user_id =$request->json()->auth get("user_id");
39. $favRoute -> startPoint = $request->json()->get("startPoint");
40. $favRoute -> destination =$request->json()->get("destination");
41. $favRoute -> save();
42. return response()->json(['message' => '加入收藏路線', 'data'=> $favRoute]);
43. }
44.
45. }
46.
47. /**
48. * Display the specified resource.
49. *
50. * @param \App\Models\FavRoute $favRoute
51. * @return \Illuminate\Http\Response
52. */
53. public function read(Request $request)
54. {
55. $select=FavRoute::where('user_id', '=', auth()->user()->user_oauth_id)->get();
56. return response()->json(
57. ['success' => true, 'FavRoute' => $select]
58. );
59. }

```

路徑	app/Http/Controllers/AuthController.php
用途	LIFF Login 登入驗證、寫入頭貼
部分程式碼	
<pre> 1. namespace App\Http\Controllers; 2. 3. use Illuminate\Http\Request; 4. use Illuminate\Support\Facades\Auth; 5. use App\Models\User; 6. 7. class AuthController extends Controller 8. { 9. // receive idToken from client, then get user info from line login api, using oauth id to check if it in database, then update user data, if not then create new user 10. public function login(Request \$request) 11. { 12. \$idToken = \$request->input('idToken'); 13. 14. if (!\$idToken) { 15. return response()->json([16. 'status' => 'error', 17. 'message' => 'idToken is required' 18.], 400); 19. } 20. 21. try { 22. \$client = new \GuzzleHttp\Client(); 23. \$response = \$client->request('POST', 'https://api.line.me/oauth2/v2.1/verify', [24. 'form_params' => [25. 'id_token' => \$idToken, 26. 'client_id' => env('LINE_CLIENT_ID', false), 27.], 28.]); 29. } catch (\GuzzleHttp\Exception\ClientException \$e) { 30. \$response = \$e->getResponse(); 31. \$responseBodyAsString = \$response->getBody()->getContents(); 32. return response()->json([33. 'error' => \$responseBodyAsString, 34. 'status' => 'error', 35. 'message' => 'idToken is invalid' 36.], 400); 37. } </pre>	


```

38.
39. $body = $response->getBody();
40. $body = json_decode($body);
41. $user = User::where('user_oauth_id', $body->sub)->where('user_oauth_provider', 'line')->first();
42. if ($user) {
43.     $user->user_name = $body->name;
44.     $user->user_avatar = isset($body->picture) ? $body->picture : null;
45.     $user->user_oauth_provider = 'line';
46.     $user->updated_by = $user->user_id;
47.     $user->save();
48. } else {
49.     $user = new User();
50.     $user->user_name = $body->name;
51.     $user->user_avatar = isset($body->picture) ? $body->picture : null;
52.     $user->user_oauth_id = $body->sub;
53.     $user->user_oauth_provider = 'line';
54.     $user->created_by = -1;
55.     $user->updated_by = -1;
56.     $user->save();
57. }
58.
59. Auth::login($user);
60.
61. return response()->json([
62.     'success' => true,
63.     'user' => $user,
64. ]);
65. }
66.
67. public function status(Request $request)
68. {
69.     if (Auth::check()) {
70.         return response()->json([
71.             'success' => true,
72.             'user' => Auth::user()
73.         ]);
74.     } else {
75.         return response()->json([
76.             'success' => false
77.         ], 401);
78.     }
79. }
80. }

```

路徑	app/Http/Controllers/LostItemController.php
用途	遺失物寫入、編修 function
部分程式碼	
<pre> 1. <namespace App\Http\Controllers; 2. 3. use App\Models\LostItem; 4. use Illuminate\Http\Request; 5. use Illuminate\Support\Facades\Auth; 6. use Illuminate\Support\Facades\Validator; 7. 8. class LostItemController extends Controller 9. { 10. static \$RULES = [11. 'lost_item_name' => 'required string max:255', 12. 'lost_item_image' => 'required string max:255', 13. 'lost_item_description' => 'nullable string max:255', 14. 'lost_item_location' => 'required string max:255', 15. 'lost_item_color' => 'required string max:255', 16. 'lost_item_status' => 'nullable string max:255', 17.]; 18. 19. public function list() 20. { 21. \$items = []; 22. if (Auth::check() && Auth::user()->user_is_admin) { 23. \$items = LostItem::all(); 24. } else { 25. \$items = LostItem::where('lost_item_status', '已通報')->get(); 26. } 27. return response()->json(28. ['items' => \$items] 29.); 30. } 31. 32. public function create(Request \$request) 33. { 34. \$data = \$request->json(); 35. Validator::make(\$data->all(), self::\$RULES)->validate(); 36. 37. \$item = LostItem::create([38. 'lost_item_name' => \$request->input('lost_item_name'), 39. 'lost_item_image' => \$request->input('lost_item_image'), </pre>	

```

40. 'lost_item_description' => $request->input('lost_item_description'),
41. 'lost_item_location' => $request->input('lost_item_location'),
42. 'lost_item_color' => $request->input('lost_item_color'),
43. 'lost_item_status' => "已通報",
44. 'created_by' => auth()->user()->user_id,
45. 'updated_by' => auth()->user()->user_id,
46.
47. ]];
48. return response()->json(
49. ['success' => true, 'item' => $item]
50. );
51. }
52.
53. public function read(Request $request, $lostItemID)
54. {
55. $item = LostItem::find($lostItemID);
56. if ($item) {
57. return response()->json(
58. ['success' => true, 'item' => $item]
59. );
60. } else {
61. return response()->json(
62. ['success' => false, 'message' => 'Item not found.']
63. );
64. }
65. }
66.
67. public function update(Request $request, $lostItemID)
68. {
69. $data = $request->json();
70. Validator::make($data->all(), self::$RULES)->validate();
71.
72. $item = LostItem::find($lostItemID);
73. if (!$item) {
74. return response()->json(
75. ['success' => false, 'message' => '找不到物品']
76. );
77. }
78.
79. if (Auth::user()->user_is_admin) {
80. $item->update([
81. 'lost_item_name' => $request->input('lost_item_name'),
82. 'lost_item_image' => $request->input('lost_item_image'),
83. 'lost_item_description' => $request->input('lost_item_description'),
84. 'lost_item_location' => $request->input('lost_item_location'),

```

```

85. 'lost_item_color' => $request->input('lost_item_color'),
86. 'lost_item_status' => $request->input('lost_item_status'),
87. 'updated_by' => auth()->user()->user_id,
88. ]);
89. } else {
90. if ($item->created_by == Auth::user()->user_id) {
91. $item->update([
92. 'lost_item_name' => $request->input('lost_item_name'),
93. 'lost_item_image' => $request->input('lost_item_image'),
94. 'lost_item_description' => $request->input('lost_item_description'),
95. 'lost_item_location' => $request->input('lost_item_location'),
96. 'lost_item_color' => $request->input('lost_item_color'),
97. 'updated_by' => auth()->user()->user_id,
98. ]);
99. } else {
100. return response()->json(
101. ['success' => false, 'message' => '你沒有權限更新此物品']
102. );
103. }
104. }
105. return response()->json(
106. ['success' => true, 'item' => $item]
107. );
108. }
109.
110. public function delete($lostItemID)
111. {
112. $lostItem = LostItem::find($lostItemID);
113. if (!$lostItem) {
114. return response()->json([
115. 'status' => false,
116. 'message' => '找不到物品'
117. ], 404);
118. }
119.
120. $user = Auth::user();
121. if ($user->user_is_admin) {
122. $lostItem->delete();
123. } else {
124. if ($lostItem->created_by == $user->user_id) {
125. $lostItem->delete();
126. } else {
127. return response()->json([
128. 'status' => false,

```

```

129. 'message' => '你沒有權限刪除此物品'
130. ], 403);
131. }
132. }
133.
134. return response()->json([
135. 'status' => true,
136. 'message' => '物品已刪除'
137. ], 200);
138. }
139. }

```

路徑	Dijkstra_python/api.php
用途	站內路線 Dijkstra 最短路徑演算法計算、OpenCV 繪製路線 Flask API
部分程式碼	
<pre> 1. import flask 2. from flask import jsonify, request, make_response 3. # from flask_cors import CORS 4. import numpy as np 5. import pandas as pd 6. import cv2 7. x = float('inf') 8. app = flask.Flask(__name__) 9. app.config["DEBUG"] = True 10. app.config["JSON_AS_ASCII"] = True 11. 12. #讀取站內剖面圖檔、節點資料 13. 14. def getSpotPosition(stationName): 15. spotPosition = pd.read_csv("stationSpotPosition/spotPosition_"+ stationName + ".csv", 16. encoding="Big5") 17. spotFull_list = [] 18. spotPosition_list = [] 19. nrows = spotPosition.shape[0] 20. for i in range(nrows): 21. ser = spotPosition.loc[i, :] 22. row_dict = [] 23. for idx, val in zip(ser.index[1:3], ser.values[1:3]): 24. if type(val) is str: 25. row_dict.append(val) 26. elif type(val) is np.int64: </pre>	

```

26. row_dict.append(int(val))
27. elif type(val) is np.float64:
28. row_dict.append(float(val))
29. spotPosition_list.append(row_dict)
30. row_dict2 = {}
31. for idx, val in zip(ser.index[:3], ser.values[:3]):
32. if type(val) is str:
33. row_dict2[idx] = val
34. elif type(val) is np.int64:
35. row_dict2[idx] = int(val)
36. elif type(val) is np.float64:
37. row_dict2[idx] = float(val)
38. spotFull_list.append(row_dict2)
39. # print(spotFull_list)
40. spotBranch_list = []
41. for i in range(nrows):
42. ser = spotPosition.loc[i]
43. row_dict = {}
44. # count = 0
45. for idx, val in zip(ser.index[3:], ser.values[3:]):
46. # count += 1
47. if val == 'x':
48. row_dict.append(float('inf'))
49. else :
50. row_dict.append(int(val))
51. spotBranch_list.append(row_dict)
52. # print(spotBranch_list)
53. return spotPosition_list, spotFull_list, spotBranch_list
54.
55. def dijkstra(mat,begin,end):
56. n = len(mat)
57. parent = [] #紀錄每個結點的父輩結點
58. collected = [] #紀錄是否經過該結點
59. distTo = mat[begin] #紀錄該點到 begin 結點路徑長度,初始值存所有點到起始點距離
60. path = [] #紀錄路徑
61. for i in range(0,n): #初始化工作
62. if i == begin:
63. collected.append(True) #所有結點均未被收集
64. else:
65. collected.append(False)
66. parent.append(-1) #均不存在父輩結點
67. while True:
68. if collected[end] == True:

```

```

69. break
70. min_n = x
71. for i in range(0,n):
72. if collected[i]==False:
73. if distTo[i] < min_n: #代表頭結點
74. min_n = distTo[i]
75. v = i
76. collected[v] = True
77. for i in range (0,n):
78. if (collected[i]==False) and (distTo[v] + mat[v][i] <= distTo[i]): #更新最短距離 ?? GET 重複值時
    進不去該判斷
79. parent[i] = v
80. distTo[i] = distTo[v] + mat[v][i]
81. e = end
82. while e != -1: #利用 parent-v 繼承關係, 循環回溯更新 path 並輸出
83. path.append(e)
84. e = parent[e]
85. path.append(begin)
86. path.reverse()
87.
88. # print("path: ",path)
89. # print("distance: ",distTo[end])
90. path.append(distTo[end])
91. return path
92. def getStartStationPath(startPoint, startTrainHeadto, destiTrainHeadto, spotFull_list,
    spotBranch_list, getPathCount):
93. # 入口是在矩陣中第幾個
94. startPoint_order = 0
95. if(getPathCount == 0):
96. destination = startTrainHeadto
97. for startSpot in spotFull_list:
98. if startSpot['position'] == startPoint:
99. break
100. startPoint_order+=1
101. elif(getPathCount == 1):
102. destination = destiTrainHeadto
103. for startSpot in spotFull_list:
104. if startTrainHeadto in startSpot['position']:
105. if startPoint in startSpot['position']:
106. break
107. startPoint_order+=1
108.
109. endPoint_order = []
110. count = 0

```

```

111. # 在正確月台的列車有哪些
112. for endSpot in spotFull_list:
113. if destination in endSpot['position']:
114. endPoint_order.append(count)
115. count +=1
116. shortPath = []
117. endPoint_results = []
118. path = []
119. # 計算入口雨正確列車哪節車廂最近，並輸出路徑
120. for i in endPoint_order:
121. endPoint_results.append(dijkstra(mat = spotBranch_list,begin = startPoint_order,end = i))
122. for j in range(len(endPoint_results)):
123. path.append(endPoint_results[j][-1])
124. shortPath.append(endPoint_results[j][-1])
125. indexMinPath = shortPath.index(min(shortPath))
126. return path[indexMinPath], spotFull_list[indexMinPath]['position']
127. def getDestiStationPath(spotFull_list, destiTrainHeadto, inCarNumber, endPoint):
128. inCarNumberCount = 0
129. # 在正確月台的列車有哪些
130. for endSpot in spotFull_list:
131. if destiTrainHeadto in endSpot['position']:
132. if inCarNumber[0:4] in endSpot['position']:
133. break
134. inCarNumberCount += 1
135. endPointOrder = 0
136. for Spot in spotFull_list:
137. if Spot['position'] == endPoint:
138. break
139. endPointOrder +=1
140. return inCarNumberCount, endPointOrder
141. def drawPath(spotPosition, path, stationName): #繪製路徑
142. spotpath = []
143. for i in range(len(path)):
144. spotpath.append(spotPosition[path[i]])
145. finalPath = np.array(spotpath, np.int32)
146. # print(finalPath[-1])
147. stationMap_img = cv2.imread('stationMap/stationMap_' + stationName + '.jpg')
148. red_color = (0, 0, 255)
149. cv2.polylines(stationMap_img, pts=[finalPath], isClosed=False, color=red_color, thickness=20)
150. cv2.circle(img=stationMap_img, center = (finalPath[0][0],finalPath[0][1]), radius =70, color
    =(64, 172, 53), thickness=-1)
151. cv2.circle(img=stationMap_img, center = (finalPath[-1][0],finalPath[-1][1]), radius =70, color
    =(164, 172, 53), thickness=-1)

```



```

152. # cv2.imshow('image',stationMap_img)
153. cv2.imwrite('stationPath.png', stationMap_img) #儲存圖片
154. etval, buffer = cv2.imencode('.png', stationMap_img)
155. response = make_response(buffer.tobytes())
156. response.headers['Content-Type'] = 'image/png'
157. return response
158. @app.route('/spotPositionALL', methods=['GET'])
159. def spotPosition_ALL():
160. if 'startStation' in request.args:
161. startStation = request.args.get('startStation', None)
162. else:
163. return "Error: No position provided. Please specify another."
164. spotFull_list = getSpotPosition(startStation)
165. return jsonify(spotFull_list)
166.
167. @app.route('/spotBranchALL', methods=['GET'])
168. def spotBranch_ALL():
169. if 'startStation' in request.args:
170. startStation = request.args.get('startStation', None)
171. else:
172. return "Error: No position provided. Please specify another."
173. spotBranch_list = getSpotPosition(startStation)
174. return jsonify(spotBranch_list)
175.
176.
177. @app.route('/python/pathdraw', methods=['GET'])
178. def pathdraw():
179. if 'startPoint' in request.args:
180. startPoint = request.args.get('startPoint', None)
181. if 'startStation' in request.args:
182. startStation = request.args.get('startStation', None)
183. if 'endPoint' in request.args:
184. endPoint = request.args.get('endPoint', None)
185. if 'transferStation' in request.args:
186. transferStation = request.args.get('transferStation', None)
187. if 'destiStation' in request.args:
188. destiStation = request.args.get('destiStation', None)
189. if 'startTrainHeadto' in request.args:
190. startTrainHeadto = request.args.get('startTrainHeadto', None)
191. if 'destiTrainHeadto' in request.args:
192. destiTrainHeadto = request.args.get('destiTrainHeadto', None)
193. if 'imgNum' in request.args:
194. imgNum = request.args.get('imgNum', None)
195. else:

```

```

196. return "Error: No position provided. Please specify another."
197.
198. spotPosition_list, spotFull_list, spotBranch_list = getSpotPosition(startStation)
199. pathStartStation, inCarNumber = getStartStationPath(startPoint, startTrainHeadto,
    destiTrainHeadto, spotFull_list, spotBranch_list, getPathCount=0)
200. response = drawPath(spotPosition_list, pathStartStation, startStation)
201. imgList = []
202. imgList.append(response)
203.
204. spotPosition_list, spotFull_list, spotBranch_list = getSpotPosition(transferStation)
205. pathStartStation, inCarNumber = getStartStationPath(inCarNumber[0:4], startTrainHeadto,
    destiTrainHeadto, spotFull_list, spotBranch_list, getPathCount=1)
206. response = drawPath(spotPosition_list, pathStartStation, transferStation)
207. imgList.append(response)
208. spotPosition_list, spotFull_list, spotBranch_list = getSpotPosition(destiStation)
209. inCarNumberCount, endPointOrder = getDestiStationPath(spotFull_list, destiTrainHeadto,
    inCarNumber, endPoint)
210. pathDestiStation = dijkstra(mat = spotBranch_list, begin = inCarNumberCount, end =
    endPointOrder)
211. response = drawPath(spotPosition_list, pathDestiStation[:-1], destiStation)
212. imgList.append(response)
213. print(type(imgNum))
214. return imgList[int(imgNum)]
215. # inCarNumberCount, endPointOrder = getDestiStationPath(spotFull_list, destiTrainHeadto,
    inCarNumber, endPoint)
216. # pathDestiStation = dijkstra(mat = spotBranch_list, begin = inCarNumberCount, end =
    endPointOrder)
217. # response = drawPath(spotPosition_list, pathDestiStation[:-1], destiStation)
218. # response = drawPath(spotPosition_list, pathStartStation, destiStation)
219. # imgList.append(response)
220.
221.
222.
223.
224. #app.run(debug=True, host='127.0.0.1', port=3000)

```

路徑	objectDetection_python/objectDetectionAPI.py
用途	讀取圖片，將其物品與背景分離，並丟入模型，取得名稱與顏色，將其翻譯成中文，回傳前端顯示
部分程式碼	
<pre> 1. # %% 2. from __future__ import print_function 3. from flask import Flask,jsonify,request 4. from turtle import color 5. import cv2 6. import urllib.request 7. import matplotlib.pyplot as plt 8. import numpy as np 9. from opencv.runtime import Core 10. import googletrans 11. from pprint import pprint 12. import cvzone 13. from cvzone.SelfiSegmentationModule import SelfiSegmentation 14. import binascii 15. import struct 16. from PIL import Image 17. import scipy 18. import scipy.misc 19. import scipy.cluster 20. import webcolors 21. from translate import Translator 22. NUM_CLUSTERS = 5 23. 24. # Initial 25. app = Flask(__name__) 26. app.config["DEBUG"] = True 27. app.config["JSON_AS_ASCII"] = False 28. 29. # 讀取圖片，將其物品與背景分離 30. def objectDetection(img_url): 31. img_req = urllib.request.urlopen(img_url) 32. 33. ie = Core() 34. segmentor = SelfiSegmentation() 35. 36. model = ie.read_model(model="model/v3-small_224_1.0_float.xml") 37. compiled_model = ie.compile_model(model=model, device_name="CPU") 38. </pre>	

```

39. output_layer = compiled_model.output(0)
40. arr = np.asarray(bytearray(img_req.read()), dtype=np.uint8)
41. img = cv2.imdecode(arr, -1) # Load it as it is
42. input_image = cv2.resize(src=img, dsize=(224, 224))
43.
44. # The MobileNet model expects images in RGB format
45. image = cv2.cvtColor(input_image, code=cv2.COLOR_BGR2RGB)
46.
47. # resize to MobileNet image shape
48. img_Out = segmentor.removeBG(image, (255,255,255), threshold=0.99)
49.
50. # reshape to model input shape
51. input_image = np.expand_dims(input_image, 0)
52.
53. result_infer = compiled_model([input_image])[output_layer]
54. result_index = np.argmax(result_infer)
55. return result_index, img_Out
56.
57. # 將物品丟入模型，取得名稱與顏色，將其翻譯成中文
58. def ConvertInferenceResult(result_index, img_Out):
59.     translator = googletrans.Translator()
60.
61.     # Convert the inference result to a class name.
62.     imagenet_classes = open("utils/imagenet_2012.txt", encoding="utf-8").read().splitlines()
63.
64.     # The model description states that for this model, class 0 is background,
65.     # so we add background at the beginning of imagenet_classes
66.     imagenet_classes = ['background'] + imagenet_classes
67.
68.     imgResultsNum = imagenet_classes[result_index][:9]
69.
70.     ar = np.asarray(img_Out)
71.     shape = ar.shape
72.     ar = ar.reshape(np.product(shape[:2]), shape[2]).astype(float)
73.
74.     # print('finding clusters')
75.     codes, dist = scipy.cluster.vq.kmeans(ar, NUM_CLUSTERS)
76.     # print('cluster centres:\n', codes)
77.
78.     vecs, dist = scipy.cluster.vq.vq(ar, codes) # assign codes
79.     counts, bins = np.histogram(vecs, len(codes)) # count occurrences
80.     # index_max = np.argmax(counts) # find most frequent
81.     try:
82.         index_max = np.argsort(counts, axis=0)[-2]
83.     except IndexError:
84.         index_max = 0

```

```

85. peak = codes[index_max]
86. colour = binascii.hexlify(bytearray(int(c) for c in peak)).decode('ascii')
87. print('most frequent is %s (%s)' % (peak, colour))
88.
89. #讀取 hexToColor 資料存 2d arrays 到 list
90. hexToColor_classes = open("utils/hexToColorName.txt", encoding="utf-8")
91. colorDict = []
92. for line in hexToColor_classes.read().splitlines():
93.     colorDict.append(line.split(' '))
94.
95. requested_colour = (peak[0], peak[1], peak[2])
96. actual_name, closest_name = get_colour_name(requested_colour)
97.
98. # 找 closest name 對應的 中文
99. for i in colorDict:
100.     color_name = i[0].lower()
101.     if(color_name == closest_name):
102.         closest_name = i[1]
103. break
104. # print("Actual colour name:", actual_name, ", closest colour name:", closest_name)
105. results = translator.translate(imgResults, dest='zh-tw')
106. print(imgResultsNum)
107. print(results.text + ' ' + closest_name)
108. data = {
109.     "item_name": results.text,
110.     "item_color": closest_name,
111.     "img_result_num": imgResultsNum,
112.     "most_frequent_color": colour
113. }
114. return jsonify(data)
115.
116. def closest_colour(requested_colour):
117.     min_colours = {}
118.     for key, name in webcolors.CSS3_HEX_TO_NAMES.items():
119.         r_c, g_c, b_c = webcolors.hex_to_rgb(key)
120.         rd = (r_c - requested_colour[0]) ** 2
121.         gd = (g_c - requested_colour[1]) ** 2
122.         bd = (b_c - requested_colour[2]) ** 2
123.         min_colours[(rd + gd + bd)] = name
124.     return min_colours[min(min_colours.keys())]
125.
126. def get_colour_name(requested_colour):
127.     try:
128.         closest_name = actual_name = webcolors.rgb_to_name(requested_colour)
129.     except ValueError:

```

```

130. closest_name = closest_colour(requested_colour)
131. actual_name = None
132. return actual_name, closest_name
133.
134. @app.route('/objectDetection', methods=['GET'])
135. def startPoint():
136.     if 'img_url' in request.args:
137.         img_url = request.args.get('img_url', None)
138.         result_index, img_Out = objectDetection(img_url)
139.         response = ConvertInferenceResult(result_index, img_Out)
140.         print(response)
141.         return response
142.     else:
143.         return "Error: No img url provided. Please specify another."
144.
145.
146. # app.run(debug=True, host='127.0.0.1', port=3001)
147.
148. # plt.show()
149.
150.
151. # install
152. # https://clay-atlas.com/blog/2020/05/05/python-cn-note-package-googletrans-google-  
translate/
153. # https://blog.csdn.net/xfyuanjun520/article/details/115465873
154. # https://www.delftstack.com/zh-tw/howto/python/opencv-background-subtraction/
155. # https://stackoverflow.com/questions/3241929/python-find-dominant-most-common-color-  
in-an-image

```

▼表 9-1- 2、前端程式清單

編號	檔案名稱	功能說明
1	index.vue	登入畫面
2	find.vue	遺失物畫面
3	navigation.vue	路線搜尋畫面
4	1-Info.vue	個人清單畫面
5	2-Route.vue	路線摘要畫面
6	3-Detail.vue	詳細路線畫面
7	4-StationMap.vue	站內剖面圖畫面
8	Z-FavoriteRoutes.vue	收藏路線畫面
9	VerticalMetroRoute.vue	詳細路線中，捷運路線顯示
10	VerticalUBikeRoute.vue	詳細路線中，Youbike 路線顯示
11	map.vue	direction API
12	favRoute.js	寫入收藏路線，如路線已被收藏過會被刪除，並再彈出 response 信息
13	liff.js	LIFF 登入、到站提醒

● 前端程式規格描述

路徑	pages/index.vue
用途	登入畫面
部分程式碼	
<pre> 1. <template> 2. <div class="menu"> 3. <div class="title">MR.Taipei</div> 4. <Button @click="lineLogin"> LINE Login </Button> 5. <Button color="lightgray" @click="\$router.push('/navigation')"> 6. 跳過不進行登入 7. </Button> 8. </div> 9. </template> 10. 11. <script> 12. import liff from '~/tools/liff' 13. 14. export default { 15. name: 'IndexPage', 16. async mounted() { 17. await liff.getBackendAuthStatus() 18. 19. if (this.\$store.state.auth.user) { 20. this.\$router.push('/navigation') 21. } else { 22. if (await liff.checkLogin()) { 23. await liff.authWithBackend() 24. this.\$router.push('/navigation') 25. } 26. } 27. }, 28. methods: { 29. async lineLogin() { 30. liff.login() 31. }, 32. }, 33. } 34. </script> 35. 36. <style scoped> 37. .menu { 38. display: flex; </pre>	


```

39.flex-direction: column;
40.width: 100%;
41.box-sizing: border-box;
42.}
43..menu > * {
44.margin-top: 1rem;
45.width: 100%;
46.
47.box-sizing: border-box;
48.}
49..title {
50.font-size: 2rem;
51.text-align: center;
52.color: white;
53.}
54.</style>

```

路徑	pages/find.vue
用途	遺失物畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Loading v-model="loading"></Loading> 4. <div class="header p-4 w-100"> 5. <div class="cols justify-content-space-between w-100 mb-4"> 6. <div class="cols"> 7. 16. <div class="flex justify-content-center rows pl-3"> 17. <div class="color-text font-size-regular mb-1"> 18. Hello, 19. {{ 20. \$store.state.auth.user </pre>	

```

21. ? $store.state.auth.user.user_name
22. : 'guest'
23. }}
24. </div>
25. <div class="color-text">
26. {{
27. $store.state.auth.user
28. ? $store.state.auth.user.user_is_admin
29. ? '管理員'
30. : '一般訪客'
31. : '一般訪客'
32. }}
33. </div>
34. <Button
35. v-if="
36. $store.state.auth.user &&
37. $store.state.auth.user.user_is_admin &&
38. mode == 'list'
39. "
40. @click="mode = 'adminList'"
41. style="margin-top: 8px"
42. size="mini"
43. >切換至管理者介面</Button
44. >
45. <Button
46. v-if="
47. $store.state.auth.user &&
48. $store.state.auth.user.user_is_admin &&
49. mode == 'adminList'
50. "
51. @click="mode = 'list'"
52. style="margin-top: 8px"
53. size="mini"
54. >切換至一般使用者介面</Button
55. >
56. </div>
57. </div>
58. <div class="flex cols align-items-center justify-content-center">
59. <div class="home" @click="$router.push('/')">
60. <Icon icon="home"> </Icon>
61. </div>
62. </div>
63. </div>

```

```

64. <Notice
65. v-if="lastReportedItemByMe"
66. class="mb-4"
67. title="你最近一次通報的遺失物"
68. icon="android-studio"
69. :message="
70. lastReportedItemByMe.lost_item_name +
71. ' | ' +
72. lastReportedItemByMe.lost_item_color
73. "
74. :tips="
75. lastReportedItemByMe.lost_item_location +
76. ' | ' +
77. new Date(lastReportedItemByMe.created_at).toLocaleString() +
78. ' | ' +
79. lastReportedItemByMe.lost_item_status
80. "
81. >
82. </Notice>
83. </div>
84.
85. <div class="items p-4">
86. <div class="rows" v-if="mode == 'list' || mode == 'adminList'">
87. <div class="cols">
88. <div class="col-6">
89. <Button
90. class="itemsButton"
91. size="small"
92. @click="( ) => (asc = !asc)"
93. ><Icon class="font-size-huge mr-2" icon="calendar"></Icon> 排序
94. </Button>
95. </div>
96. <div class="col-6">
97. <Button
98. class="itemsButton"
99. size="small"
100. @click="( ) => (showSearch = !showSearch)"
101. ><Icon class="font-size-huge mr-2" icon="magnify"></Icon> 搜尋
102. </Button>
103. </div>
104. </div>
105. <Input title="搜尋" v-model="search" v-if="showSearch"></Input>
106. <div

```

```

107. class="flex cols justify-content-space-between align-items-center mt-3"
108. >
109. <span class="font-size-regular">所有通報遺失物</span>
110. <span v-if="false" class="font-size-mini color-secondary"
111. >查看更多</span>
112. >
113. </div>
114.
115. <div
116. class="item flex justify-content-space-between align-items-center mt-3"
117. v-for="item in items
118. .filter((item) =>
119. search
120. ? (
121. item.lost_item_name +
122. item.lost_item_color +
123. item.lost_item_location
124. ).includes(search)
125. : true
126. )
127. .sort((a, b) => (asc ? (a > b ? -1 : 1) : a > b ? 1 : -1))"
128. :key="item.lost_item_id"
129. @click="view(item)"
130. >
131. <div class="cols align-items-center">
132. <div class="itemIcon mr-3">
133. <Icon icon="android-studio"></Icon>
134. </div>
135. <div>
136. <div class="itemTitle">
137. {{ item.lost_item_name }} | {{ item.lost_item_color }}
138. </div>
139. <div class="itemSubTitle">
140. {{ item.lost_item_location }} |
141. {{ new Date(item.created_at).toLocaleString() }} |
142. {{ item.lost_item_status }}
143. </div>
144. </div>
145. </div>
146. <div>
147. <Icon class="arrowIcon" icon="chevron-right"></Icon>
148. </div>
149. </div>

```

```

150. <div style="padding-top: 5rem"></div>
151. </div>
152. <div class="" v-if="mode == 'report' || mode == 'edit'">
153. <div class="reportUpload" @click="$refs.fileInput.click()">
154. 
155. <span v-else>上傳遺失物圖片</span>
156. </div>
157. <input
158. type="file"
159. style="display: none"
160. @change="fileSelected"
161. ref="fileInput"
162. />
163. <Input
164. title="物品名稱"
165. titleColor="#000"
166. backgroundColor="#fff"
167. color="#000"
168. class="mb-3 reportInput"
169. v-model="report.lost_item_name"
170. ></Input>
171. <Input
172. title="物品站點"
173. titleColor="#000"
174. backgroundColor="#fff"
175. color="#000"
176. class="mb-3 reportInput"
177. v-model="report.lost_item_location"
178. ></Input>
179. <Input
180. title="物品顏色"
181. titleColor="#000"
182. backgroundColor="#fff"
183. color="#000"
184. class="mb-3 reportInput"
185. v-model="report.lost_item_color"
186. ></Input>
187. <Input
188. title="通報時間 (依伺服器時間為準)"
189. titleColor="#000"
190. backgroundColor="#fff"
191. color="#000"
192. class="mb-3 reportInput"

```

```

193. type="datetime-local"
194. v-model="report.reportTime"
195. v-if="mode == 'report'"
196. > </Input>
197. <div class="viewItemContent" v-else>
198. <div class="viewItemContentTitle">通報時間</div>
199. <div class="viewItemContentText">
200. {{ new Date(viewItem.created_at).toLocaleString() }}
201. </div>
202. </div>
203.
204. <Select
205. title="通報状態"
206. titleColor="#000"
207. backgroundColor="#fff"
208. color="#000"
209. class="mb-3 reportInput mt-3"
210. type="datetime-local"
211. v-model="report.lost_item_status"
212. v-if="mode == 'edit'"
213. >
214. <option>已通報</option>
215. <option>已領回</option>
216. <option>已丢弃</option>
217. </Select>
218.
219. <Button textColor="#fff" color="#eca468" @click="submit">
220. {{ mode == 'report' ? '確認通報' : '編輯物品' }}</Button>
221. >
222. <Button
223. textColor="#fff"
224. color="#eca468"
225. style="margin-top: 8px"
226. @click="( ) => (mode = 'list')"
227. >
228. 返回</Button>
229. >
230. </div>
231. <div v-if="mode == 'view'">
232. 
233. <div class="viewItemContent">
234. <div class="viewItemContentTitle">物品名稱</div>
235. <div class="viewItemContentText">{{ viewItem.lost_item_name }}</div>
236. </div>

```

```

237. <div class="viewItemContent">
238. <div class="viewItemContentTitle">物品顏色</div>
239. <div class="viewItemContentText">{{ viewItem.lost_item_color }}</div>
240. </div>
241. <div class="viewItemContent">
242. <div class="viewItemContentTitle">物品站點</div>
243. <div class="viewItemContentText">
244. {{ viewItem.lost_item_location }}
245. </div>
246. </div>
247. <div class="viewItemContent">
248. <div class="viewItemContentTitle">通報時間</div>
249. <div class="viewItemContentText">
250. {{ new Date(viewItem.created_at).toLocaleString() }}
251. </div>
252. </div>
253. <div class="viewItemContent">
254. <div class="viewItemContentTitle">物品狀態</div>
255. <div class="viewItemContentText">
256. {{ viewItem.lost_item_status }}
257. </div>
258. </div>
259.
260. <Button
261. textColor="#fff"
262. color="#eca468"
263. style="margin-top: 8px"
264. @click="( ) => (mode = 'list')"
265. >
266. 返回</Button>
267. >
268. <Button
269. textColor="#fff"
270. color="#eca468"
271. style="margin-top: 8px"
272. @click="editItem(viewItem)"
273. v-if="$store.state.auth.user && $store.state.auth.user.user_is_admin"
274. >
275. 編輯</Button>
276. >
277. </div>
278. </div>
279.
280. <Button

```

```

281. class="reportButton"
282. textColor="#fff"
283. color="#eca468"
284. v-if="mode == 'list' && this.$store.state.auth.user"
285. @click="doReport"
286. >
287. <Icon class="reportIcon" icon="bullhorn"></Icon>
288. 我要通報</Button>
289. >
290. </div>
291. </template>
292.
293. <script>
294. import config from './config'
295.
296. import Icon from '~/components/Elements/Icon.vue'
297. import Loading from '~/components/Elements/Loading.vue'
298. export default {
299. components: { Icon, Loading },
300. data() {
301. return {
302. mode: 'list',
303. items: [],
304. loading: false,
305. asc: false,
306. showSearch: false,
307. search: '',
308.
309. viewItem: null,
310. lastReportedItemByMe: null,
311. lastMode: 'list',
312. report: {
313. lost_item_name: '',
314. lost_item_image: null,
315. lost_item_location: '',
316. lost_item_color: '',
317. itemImageFile: null,
318. reportTime: new Date(
319. new Date().getTime() + new Date().getTimezoneOffset() * -1 * 60 * 1000
320. )
321. .toISOString()
322. .slice(0, 16),
323. },
324. }

```



```

325. },
326. mounted() {
327. this.getItems()
328. },
329. methods: {
330. async getItems() {
331. this.loading = true
332. await this.$axios
333. .get(`/api/lostItems`)
334. .then((response) => {
335. this.$set(this, 'items', response.data.items)
336. if (this.$store.state.auth.user) {
337. this.$set(
338. this,
339. 'lastReportedItemByMe',
340. this.items.find(
341. (item) => item.created_by == this.$store.state.auth.user.user_id
342. )
343. )
344. }
345. })
346. .catch((err) => {
347. console.log(err)
348. })
349. this.loading = false
350. },
351. async fileSelected(event) {
352. if (event.target.files[0] == undefined) return
353. this.loading = true
354. this.report.itemImageFile = event.target.files[0]
355. this.report.lost_item_image = await this.uploadImage(
356. this.report.itemImageFile
357. )
358. let itemInfo = await this.getItemInfo(this.report.lost_item_image)
359. this.report.lost_item_color = itemInfo.item_color
360. this.report.lost_item_name = itemInfo.item_name
361. this.loading = false
362. },
363. async uploadImage(file) {
364. var formData = new FormData()
365. formData.append('image', file)
366.
367. return this.$axios

```

```

368. .post('https://api.imgur.com/3/image', formData, {
369. headers: {
370. Authorization: 'Client-ID ' + config.imgur.clientID,
371. },
372. })
373. .then((response) => {
374. return response.data.data.link
375. })
376. .catch((error) => {
377. alert('上傳失敗，請稍後再試')
378. console.log(error)
379. return null
380. })
381. },
382. async getItemInfo(url) {
383. return this.$axios
384. .get('/objectDetection?img_url=' + url)
385. .then((response) => {
386. return response.data
387. })
388. .catch((error) => {
389. return {
390. item_name: "",
391. item_color: "",
392. }
393. })
394. },
395. async submit() {
396. this.loading = true
397. let url =
398. this.mode == 'report'
399. ? '/api/lostItem'
400. : '/api/lostItem/' + this.report.lost_item_id
401. let method = this.mode == 'report' ? 'post' : 'put'
402. await this.$axios[method](url, this.report)
403. .then((response) => {
404. this.$snackbar({
405. message: this.mode == 'report' ? '通報成功' : '編輯成功',
406. })
407. this.mode = this.lastMode
408. this.getItems()
409. })
410. .catch((err) => {

```

```

411. this.$snackbar({
412. message: '送出失敗 錯誤:' + err.response.data.message,
413. })
414. })
415. this.loading = false
416. },
417. view(item) {
418. if (this.mode == 'adminList') {
419. this.lastMode = 'adminList'
420. this.editItem(item)
421. return
422. }
423. this.mode = 'view'
424. this.lastMode = 'list'
425. this.$set(this, 'viewItem', item)
426. },
427. doReport() {
428. this.report.lost_item_name = ""
429. this.report.lost_item_image = ""
430. this.report.lost_item_location = ""
431. this.report.lost_item_color = ""
432. this.report.reportTime = new Date(
433. new Date().getTime() + new Date().getTimezoneOffset() * -1 * 60 * 1000
434. )
435. .toISOString()
436. .slice(0, 16)
437. this.mode = 'report'
438. },
439. editItem(item) {
440. this.report.lost_item_id = item.lost_item_id
441. this.report.lost_item_name = item.lost_item_name
442. this.report.lost_item_image = item.lost_item_image
443. this.report.lost_item_location = item.lost_item_location
444. this.report.lost_item_color = item.lost_item_color
445. this.report.lost_item_status = item.lost_item_status
446. this.report.reportTime = item.reportTime
447. this.mode = 'edit'
448. },
449. },
450. layout: 'default-noPadding',
451. }
452. </script>
453.

```

```
454. <style scoped>
455. .wrapper {
456. display: flex;
457. flex-direction: column;
458. justify-content: flex-start;
459. width: 100%;
460. box-sizing: border-box;
461. }
462. .avatar {
463. width: 5rem;
464. height: 5rem;
465. object-fit: cover;
466. border-radius: 50%;
467. }
468. .home {
469. width: 2.5rem;
470. height: 2.5rem;
471. border-radius: 50%;
472. font-size: 2rem;
473. background: #fff;
474. color: #00e88e;
475. display: flex;
476. justify-content: center;
477. align-items: center;
478. }
479. .items {
480. height: 100%;
481. width: 100%;
482. background: #fff;
483. border-radius: 32px 32px 0px 0px;
484. overflow: auto;
485. }
486. .itemsButton {
487. justify-content: flex-start !important;
488. align-items: center;
489. margin: 0.5rem;
490. color: #fff !important ;
491. border-radius: 32px;
492. }
493.
494. .item {
495. padding: 1rem;
496. border: 2px solid #444444;
```

```
497 .border-radius: 24px;
498 .}
499 .itemIcon {
500 .width: 3rem;
501 .height: 3rem;
502 .border-radius: 4px;
503 .font-size: 2rem;
504 .background: #00e88e;
505 .color: #fff;
506 .display: flex;
507 .justify-content: center;
508 .align-items: center;
509 .}
510 .itemTitle {
511 .font-size: 1.2rem;
512 .font-weight: bold;
513 .}
514 .itemSubTitle {
515 .margin-top: 0.2rem;
516 .font-size: 0.8rem;
517 .}
518 .arrowIcon {
519 .font-size: 2rem;
520 .}
521 .reportIcon {
522 .font-size: 2.4rem;
523 .}
524 .reportButton {
525 .width: 90%;
526 .bottom: 16px;
527 .left: 5%;
528 .
529 .box-sizing: border-box;
530 .position: fixed;
531 .padding: 0;
532 .
533 .display: flex;
534 .align-items: center;
535 .}
536 .reportInput >>> input {
537 .margin-top: 1rem;
538 .border: 1px #000 solid;
539 .}
540 .reportUpload {
```

```
541 .width: 100%;
542 .height: 8rem;
543 .background: #66f2bb;
544 .color: #fff;
545 .border-radius: 16px;
546 .
547 .margin-bottom: 2rem;
548 .
549 .display: flex;
550 .justify-content: center;
551 .align-items: center;
552 .}
553 .reportUpload > img {
554 .width: 100%;
555 .height: 100%;
556 .object-fit: cover;
557 .}
558 .
559 .viewItemImage {
560 .width: 100%;
561 .height: 100%;
562 .object-fit: cover;
563 .border-radius: 16px;
564 .}
565 .viewItemContent {
566 .margin-top: 1rem;
567 .display: flex;
568 .flex-direction: column;
569 .justify-content: flex-start;
570 .align-items: flex-start;
571 .}
572 .viewItemContentTitle {
573 .font-size: 1rem;
574 .font-weight: bold;
575 .margin: 8px 0px;
576 .}
577 .viewItemContentText {
578 .margin-top: 0.2rem;
579 .font-size: 1.6rem;
580 .}
581 </style>
```

路徑	pages/navigation.vue
用途	路線搜尋畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Icon 4. v-if="panelHistory.length > 1" 5. @click.native="prev" 6. icon="arrow-left" 7. class="prev" 8. ></Icon> 9. <component 10. :is="currentPanel" 11. :data="data" 12. :map="map" 13. @next="nextPressed" 14. ></component> 15. </div> 16. </template> 17. 18. <script> 19. import InfoPanel from '~/components/Navigation/Panels/1-Info.vue' 20. import RoutePanel from '~/components/Navigation/Panels/2-Route.vue' 21. import DetailPanel from '~/components/Navigation/Panels/3-Detail.vue' 22. import StationMapPanel from '~/components/Navigation/Panels/4-StationMap.vue' 23. import FavoriteRotuesPanel from '~/components/Navigation/Panels/Z-FavoriteRotues.vue' 24. 25. export default { 26. name: 'NavigatoinPage', 27. layout: 'map', 28. 29. data() { 30. return { 31. // 目前頁面及頁面瀏覽紀錄 32. currentPanel: InfoPanel, 33. panelHistory: [], 34. 35. // 傳進各個頁面的資料 36. data: { 37. leaveTime: new Date(), 38. tokenResult: {}, 39. idTokenDecode: {}, </pre>	

```

40.},
41.
42.map: null,
43.}
44.},
45.
46.mounted() {
47.let queryParams = this.$route.query
48.this.map = this.$nuxt.$map
49.},
50.
51.methods: {
52.nextPressed(event) {
53.if (event == 'getInfoPanel') {
54.this.currentPanel = InfoPanel
55.}
56.if (event == 'getRoute') {
57.if (this.panelHistory.length <= 1) {
58.this.panelHistory.push(InfoPanel)
59.}
60.this.panelHistory.push(RoutePanel)
61.this.currentPanel = RoutePanel
62.}
63.if (event == 'getDetail') {
64.this.panelHistory.push(DetailPanel)
65.this.currentPanel = DetailPanel
66.}
67.if (event == 'getStationMap') {
68.this.panelHistory.push(StationMapPanel)
69.this.currentPanel = StationMapPanel
70.}
71.if (event == 'getFavoriteRotues') {
72.this.panelHistory.push(FavoriteRotuesPanel)
73.this.currentPanel = FavoriteRotuesPanel
74.}
75.},
76.prev() {
77.if (this.panelHistory.length > 1) {
78.this.panelHistory.pop()
79.this.currentPanel = this.panelHistory[this.panelHistory.length - 1]
80.if (this.currentPanel == InfoPanel) {
81.this.panelHistory = []
82.}
83.}

```



```

84. },
85. },
86. }
87. </script>
88.
89. <style scoped>
90. .wrapper {
91. width: 100%;
92. box-sizing: border-box;
93. }
94. .prev {
95. position: fixed;
96. bottom: 1.5rem;
97. left: 1rem;
98. /* background-color: #f3f5f8; */
99. width: 20%;
100. font-size: 2rem;
101. color: #17e68e;
102. /* border-radius: 10px; */
103. display: flex;
104. justify-content: center;
105. align-items: center;
106. }
107. </style>

```

路徑	Component/Navigation/Panels/1-Info.vue
用途	個人清單畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Panel v-if="!menu"> 4. <div class="cols"> 5. <div class="col-12 p-2"> 6. <Input v-model="data.fromLocation" title="起點" /> 7. </div> 8. </div> 9. <div class="cols"> 10. <div class="col-12 p-2"> 11. <Input v-model="data.toLocation" title="目的地" /> 12. </div> </pre>	

```

13. </div>
14. <div class="rows align-items-center mt-4 mb-2">
15. <span class="color-secondary font-size-small">出發時間</span>
16. <span class="color-text font-size-huge">{{
17. data.leaveTime
18. .toLocaleTimeString(undefined, {
19. hour12: false,
20. hour: '2-digit',
21. minute: '2-digit',
22. }}
23. .replace('24:', '00:')
24. }}</span>
25. </div>
26. </Panel>
27. <Panel v-else>
28. <div class="cols">
29. <div class="col-5 rows align-items-center p-3">
30. 
39. <span class="color-text font-size-regular mt-2">{{
40. this.$store.state.auth.user
41. ? this.$store.state.auth.user.user_name
42. : null
43. }}</span>
44. </div>
45. <div class="col-7">
46. <div class="rows">
47. <Button
48. size="small"
49. color="#eca468"
50. class="mb-3"
51. @click="$router.push('/find')"
52. >尋找遺失物</Button>
53. >
54. <Button
55. size="small"

```

```

56. color="#eca468"
57. class="mb-3"
58. @click="$emit('next', 'getFavoriteRoutes')"
59. >路線收藏</Button>
60. >
61. <Button size="small" @click="menu = !menu">路線規劃</Button>
62. </div>
63. </div>
64. </div>
65. </Panel>
66. <div>
67. <div class="cols">
68. <div class="col-3 p-2">
69. <Button color="#f3f5f8" textColor="#00e88e" @click="menu = !menu">
70. <Icon v-if="menu" icon="close"></Icon>
71. <Icon v-else icon="menu"></Icon>
72. </Button>
73. </div>
74. <div class="col-9 p-2">
75. <Button v-if="!menu" @click="$emit('next', 'getRoute')">
76. >搜尋最佳路線</Button>
77. >
78. </div>
79. </div>
80. </div>
81. </div>
82. </template>
83.
84. <script>
85. import ubike from '~/tools/ubike'
86. export default {
87.   props: {
88.     data: {
89.       type: Object,
90.     },
91.   },
92.   async mounted() {
93.     ubike.getUbikeApi() //取 ubike 資料
94.   },
95.   data() {
96.     return {
97.       menu: false,
98.     }

```

```

99. },
100. }
101. </script>
102.
103. <style scoped>
104. .wrapper {
105. width: 100%;
106. box-sizing: border-box;
107. }
108.
109. .wrapper > div {
110. margin-top: 1rem;
111. }
112.
113. .avatar {
114. box-sizing: border-box;
115. width: 100%;
116. height: auto;
117. object-fit: cover;
118. border-radius: 50%;
119. }
120. </style>

```

路徑	Component/Navigation/Panels/2-Route.vue
用途	路線摘要畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Panel> 4. <div class="cols"> 5. <div class="col-12 p-2"> 6. <div class="rows flex-cols-flex-start"> 7. 起點 8. 9. {{ 10. data.fromLocation 11. }} 12. </div> 13. </div> 14. </div> 15. <div class="cols"> </pre>	

```

16. <div class="col-12 p-2">
17. <div class="rows flex-cols-flex-start">
18. <span class="font-size-small color-secondary">目的地</span>
19. <span class="font-size-large color-text">{{
20. data.toLocation
21. }}</span>
22. </div>
23. </div>
24. </div>
25. <div>
26. <HorizontalMetroRoute
27. v-if="data.route"
28. :route="data.route"
29. ></HorizontalMetroRoute>
30. </div>
31. <div class="cols mt-4 mb-2">
32. <div class="col-4 rows">
33. <span class="color-secondary font-size-small">出發時間</span>
34. <span class="color-text font-size-regular">{{ departure_time }}</span>
35. </div>
36. <div class="col-4 rows">
37. <span class="color-secondary font-size-small">抵達時間</span>
38. <span class="color-text font-size-regular">{{ arrival_time }}</span>
39. </div>
40. <div class="col-4 rows">
41. <span class="color-secondary font-size-small">&nbsp;</span>
42. <div class="cols ml-3">
43. <span class="color-focus font-size-regular">{{ duration }}</span>
44. </div>
45. </div>
46. </div>
47. </Panel>
48. <div class="cols">
49. <div class="col-3 p-2">
50. <Button color="#f3f5f8" textColor="#00e88e"> &nbsp;</Button>
51. </div>
52. <div class="col-9 p-2">
53. <Button @click="$emit('next', 'getDetail')">詳細路線資料</Button>
54. </div>
55. </div>
56. </div>
57. </template>
58.

```

```

59. <script>
60. export default {
61. props: {
62. data: {
63. type: Object,
64. }
65. },
66.
67. async mounted() {
68.
69. let directionsService = new google.maps.DirectionsService()
70. let directionsRenderer = new google.maps.DirectionsRenderer()
71.
72. let request = {
73. origin: this.data.fromLocation,
74. destination: this.data.toLocation,
75. travelMode: 'TRANSIT',
76. transitOptions: {
77. modes: ['SUBWAY'],
78. },
79. }
80.
81. directionsRenderer.setMap(this.$nuxt.$map)
82.
83. let directions = await new Promise((r) => {
84. directionsService.route(request, function (result, status) {
85. r({
86. result,
87. status,
88. })
89. })
90. })
91.
92. if (directions.status === 'OK' && directions.result.routes[0].legs[0].departure_time) {
93. console.log('[Navigation]', '[DirectionsService]', directions)
94.
95. let steps = directions.result.routes[0].legs[0].steps
96. let routeTime = directions.result.routes[0].legs[0]
97.
98. localStorage.setItem('Navigation:RouteTime', JSON.stringify(routeTime))
99.
100. this.departure_time = routeTime.departure_time.text
101. this.arrival_time = routeTime.arrival_time.text
102. this.duration = routeTime.duration.text
103.
104. localStorage.setItem('Routes', JSON.stringify(steps))

```

```

105.steps.forEach((res, key) => {
106.new google.maps.Marker({
107.position: {
108.lat: res.start_location.lat(),
109.lng: res.start_location.lng(),
110.},
111.label: { text: key + ", color: '#fff' },
112.map: this.$nuxt.$map,
113.})
114.})
115.
116.directionsRenderer.setDirections(directions.result)
117.} else {
118.window.$nuxt.$snackbar({
119.message: '找不到此起點',
120.})
121.}
122.},
123.data() {
124.return {
125.arrival_time: null,
126.departure_time: null,
127.duration: null,
128.}
129.},
130.}
131.</script>
132.
133.<style scoped>
134.wrapper {
135.width: 100%;
136.box-sizing: border-box;
137.}
138.
139.wrapper > div {
140.margin-top: 1rem;
141.}
142.</style>

```

路徑	Component/Navigation/Panels/3-Detail.vue
用途	詳細路線畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Panel> 4. <div class="justify-content-space-between align-items-flex-end"> 5. 路線資訊 6. NT 45 7. </div> 8. 9. <div class="col-12 mt-3"> 10. <div class="rows flex-cols-flex-start"> 11. 起點 12. </div> 13. </div> 14. 15. <div class="cols mt-1"> 16. <div class="col-10 flex-cols-flex-start"> 17. {{ 18. data.fromLocation 19. }} 20. </div> 21. <div class="col-2 flex-cols-flex-start arrival-time"> 22. {{ 23. departure_time 24. }} 25. </div> 26. </div> 27. 28. <div class="col-12 mt-3"> 29. <div class="rows flex-cols-flex-start"> 30. 目的地 31. </div> 32. </div> 33. <div class="cols mt-1"> 34. <div class="col-10 flex-cols-flex-start"> 35. {{ data.toLocation }} 36. </div> 37. <div class="col-2 flex-cols-flex-start arrival-time"> 38. {{ 39. arrival_time </pre>	


```

40. }}</span>
41. </div>
42. </div>
43. <div v-if="data.detailRoutes">
44. <div v-for="(detail, index) in data.detailRoutes" :key="index">
45. <hr v-if="data.fromLocation === '信愛公園' || detail.type == 'metro'"/>
46. <VerticalMetroRoute
47. v-if="detail.type == 'metro'"
48. :route="detail.route"
49. ></VerticalMetroRoute>
50. <VerticalUBikeRoute
51. v-if="detail.type == 'ubike' && data.fromLocation === '信愛公園'"
52. :route="detail.route"
53. ></VerticalUBikeRoute>
54. </div>
55. </div>
56. </Panel>
57. <div class="cols mb-1">
58. <div class="col-3 p-1">
59. <Button color="#f3f5f8" textColor="#00e88e"> &nbsp;</Button>
60. </div>
61. <div class="col-2 p-1">
62. <Button @click="favorite" color="#e68181" textColor="#ffffff" size="small">
63. <Icon icon="heart"></Icon>
64. </Button>
65. </div>
66. <div class="col-2 p-1">
67. <Button
68. @click="setNotice"
69. color="#e9a668"
70. size="small"
71. >
72. <Icon icon="bell-ring"></Icon>
73. </Button>
74. </div>
75. <div class="col-5 p-1">
76. <Button @click="$emit('next', 'getStationMap')" size="small"
77. >車站内部路線</Button>
78. >
79. </div>
80. </div>
81. </div>
82. </template>

```

```

83.
84. <script>
85. import favRoute from '~/tools/favRoute'
86. import liff from '~/tools/liff'
87. let isSaved = false
88. export default {
89.   props: {
90.     data: {
91.       type: Object,
92.     },
93.   },
94.   methods: {
95.     favorite() {
96.       favRoute.writeFavRouteToDB(this.data.fromLocation, this.data.toLocation)
97.     },
98.   },
99.   async setNotice() {
100.     await liff.init()
101.     await liff.setNotice()
102.   }
103. },
104. mounted() {
105.   let routeTime = JSON.parse(localStorage.getItem('Navigation:RouteTime'))
106.   this.departure_time = routeTime.departure_time.text
107.   this.arrival_time = routeTime.arrival_time.text
108.   this.duration = routeTime.duration.text
109.
110.   let quantity = localStorage.getItem('愛國金山路口-sbi')
111.   let vacancy = localStorage.getItem('東門站出口-bemp')
112.
113.   const routes = JSON.parse(localStorage.getItem('Routes'))
114.   this.$set(this.data, 'detailRoutes', [
115.     {
116.       type: 'ubike',
117.       route: [
118.         {
119.           version: '1.0',
120.           name: '金山愛國路口',
121.           distance: '',
122.           quantity: quantity,
123.           vacancy: null,
124.         },
125.         {
126.           version: '1.0',

```

```

127. name: '捷運東門站',
128. distance: '',
129. quantity: null,
130. vacancy: vacancy,
131. },
132. ],
133. },
134. {
135. type: 'metro',
136. route: routes,
137. },
138. ])
139. },
140. data() {
141. return {
142. arrival_time: null,
143. departure_time: null,
144. duration: null,
145. }
146. },
147. }
148. </script>
149.
150. <style scoped>
151. .wrapper {
152. width: 100%;
153. box-sizing: border-box;
154. }
155.
156. .wrapper > div {
157. margin-top: 1rem;
158. }
159. .arrival-time {
160. position: relative;
161. }
162. .arrival-time > span {
163. position: absolute;
164. bottom: 0;
165. }
166. </style>

```

路徑	Component/Navigation/Panels/4-StationMap.vue
用途	站內剖面圖畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Panel style="overflow: auto; height: 650px"> 4. <div class="justify-content-space-between align-items-flex-end"> 5. 車站內部路線 6. </div> 7. 8. <div 9. v-for="(station, index) in data.sationMaps" 10. :key="index" 11. class="station" 12. > 13. 14. {{ station.type }} 15. 16. 17. {{ station.station }} 18. 19. 20. <hr /> 21. </div> 22. </Panel> 23. <div class="cols mb-1"> 24. <div class="col-3 p-1"> 25. <Button color="#f3f5f8" textColor="#00e88e"> &nbsp; </Button> 26. </div> 27. <div class="col-4 p-1"> 28. <Button 29. @click="\$emit('next', 'setNotice')" 30. color="#e9a668" 31. size="small" 32. > 33. <Icon icon="bell-ring"> </Icon> 34. </Button> 35. </div> 36. 37. <div class="col-5 p-1"> 38. <Button @click="\$emit('next', 'getDetail')" size="small" </pre>	

```

39.>詳細路線資料</Button
40.>
41.</div>
42.</div>
43.</div>
44.</template>
45.
46.<script>
47.export default {
48.props: {
49.data: {
50.type: Object,
51.},
52.},
53.mounted() {
54.this.$set(this.data, 'sationMaps', [
55.{
56.type: '起點',
57.station: '東門站',
58.image:
  'http://140.131.114.159:3000/pathdraw?startPoint=Exit2&endPoint=Exit2&startStation=Dongmen&t
ransferStation=Daan&destiStation=Dazhi&startTrainHeadto=%E8%B1%A1%E5%B1%B1&destiTrain
Headto=%E5%8D%97%E6%B8%AF%E5%B1%95%E8%A6%BD%E9%A4%A8&imgNum=0',
59.},
60.{
61.type: '轉乘站',
62.station: '大安站',
63.image:
  'http://140.131.114.159:3000/pathdraw?startPoint=Exit2&endPoint=Exit2&startStation=Dongmen&t
ransferStation=Daan&destiStation=Dazhi&startTrainHeadto=%E8%B1%A1%E5%B1%B1&destiTrain
Headto=%E5%8D%97%E6%B8%AF%E5%B1%95%E8%A6%BD%E9%A4%A8&imgNum=1',
64.},
65.{
66.type: '終點站',
67.station: '大直站',
68.image:
  'http://140.131.114.159:3000/pathdraw?startPoint=Exit2&endPoint=Exit2&startStation=Dongmen&t
ransferStation=Daan&destiStation=Dazhi&startTrainHeadto=%E8%B1%A1%E5%B1%B1&destiTrain
Headto=%E5%8D%97%E6%B8%AF%E5%B1%95%E8%A6%BD%E9%A4%A8&imgNum=2',
69.},
70.])
71.},
72.}

```

```

73. </script>
74.
75. <style scoped>
76. .wrapper {
77. width: 100%;
78. /* height: 100%;
79. overflow: scroll; */
80. box-sizing: border-box;
81. }
82.
83. .wrapper > div {
84. margin-top: 1rem;
85. }
86.
87. .station {
88. margin-top: 0.01rem;
89. display: flex;
90. flex-direction: column;
91. }
92.
93. .station > hr {
94. width: 100%;
95. }
96.
97. .station > * {
98. margin-top: 0.2rem;
99. }
100.
101. img {
102. width: 100%;
103. height: auto;
104. }
105. </style>

```

路徑	Component/Navigation/Panels/Z-FavoriteRoutes.vue
用途	收藏路線畫面
部分程式碼	
<pre> 1. <template> 2. <div class="wrapper"> 3. <Panel> 4. <div class="justify-content-space-between align-items-flex-end"> 5. 路線收藏 </pre>	

```

6.
7. </div>
8.
9. <div
10. v-for="(route, index) in routes"
11. :key="index"
12. @click="$emit('next', 'getInfoPanel'),data.fromLocation =
    route.startPoint ,data.toLocation=route.destination"
13. :class="{ route: true, active: selectedRoute == route }"
14. >
15.
16. {{ route.startPoint }} - {{ route.destination }}
17. </div>
18.
19. <div class="cols mt-4 p-1">
20. <!-- <Button @click="$emit('next', 'getDetail')" size="small" class="w-100"
21. >搜尋最佳路線</Button
22. > -->
23. <Button @click="$emit('next', 'getInfoPanel')" size="small" class="w-100"
24. >返回規劃路線</Button
25. >
26. </div>
27. </Panel>
28. </div>
29. </template>
30.
31. <script>
32.
33. import axios from 'axios'
34. export default {
35. props: {
36. data: {
37. type: Object,
38. },
39. },
40. data() {
41. return {
42. selectedRoute: null,
43. routes: [],
44. }
45. },
46. async mounted() {
47. await axios
48. .get('/api/favRoute/read')
49. .then((response) => {

```

```

50. this.routes = response.data.FavRoute
51.
52. })
53. .catch((error) => {
54. return null
55. }
56. },
57. }
58. </script>
59.
60. <style scoped>
61. .wrapper {
62. width: 100%;
63. box-sizing: border-box;
64. }
65.
66. .route {
67. width: 100%;
68. box-sizing: border-box;
69.
70. margin-top: 1rem;
71. background: white;
72. border: 4px solid #eca468;
73. border-radius: 8px;
74.
75. display: flex;
76. justify-content: center;
77. align-items: center;
78.
79. font-size: 1rem;
80. font-weight: 500;
81.
82. padding: 1rem 0.2rem;
83. }
84.
85. .active {
86. background: #eca468;
87. }
88. </style>

```


路徑	Component/Elements/VerticalMetroRoute.vue
用途	詳細路線中，捷運路線顯示
部分程式碼	
<pre> 1. <template> 2. <div class="elements" v-if="displayElements"> 3. <div v-for="(element, index) in displayElements" :key="index" :class="element.type"> 4. <div class="icon"> 5. <div v-for="(mrtitem, i) in mrtData" :key="i"> 6. <MetroStation 7. v-if=" 8. element.type == 'station' && 9. element.station.transit != null && 10. (mrtitem.StationName.Zh_tw === 11. element.station.transit.departure_stop.name 12. mrtitem.StationName.En === element.station.transit.departure_stop.name) 13. " 14. :color="element.station.transit.line.color" 15. :code="mrtitem.StationID" 16. exit="" 17. ></MetroStation> 18. </div> 19. 20. <div 21. v-if="element.type == 'line'" 22. class="lineBackground" 23. :style="{ 24. background: element.color, 25. }" 26. ></div> 27. </div> 28. 29. <div 30. class="content" 31. v-if="element.type == 'station' && element.station.transit != null" 32. > 33. <div class="title font-size-regular color-text"> 34. 捷運{{ element.station.transit.departure_stop.name }} 35. <Icon v-if="element.station.notice" icon="bell-ring"></Icon> 36. </div> 37. <div class="tips font-size-small color-text"> 38. 往{{ element.station.transit.headsign }} </pre>	

```

39. <br />
40. <span v-if="element.index !== element.lastTransitIndex" style="color: #17e68e">
41. 出口 2</span>
42. >
43. </div>
44. </div>
45.
46. <div
47. class="datetime font-size-small color-text"
48. v-if="element.type == 'station' && element.station.transit != null"
49. >
50. {{ element.station.transit.departure_time.text }}
51. </div>
52. </div>
53. <div
54. v-for="(element, index) in displayElements"
55. :key="index == element.lastTransitIndex"
56. :class="element.type"
57. >
58. <div v-for="(mrtitem, i) in mrtData" :key="i">
59. <MetroStation
60. v-if="
61. element.type == 'station' &&
62. element.index == element.lastTransitIndex &&
63. (mrtitem.StationName.Zh_tw === element.station.transit.arrival_stop.name ||
64. mrtitem.StationName.En === element.station.transit.arrival_stop.name)
65. "
66. :color="element.station.transit.line.color"
67. :code="mrtitem.StationID"
68. exit=""
69. ></MetroStation>
70. </div>
71. <div
72. v-if="element.type == 'line'"
73. class="lineBackground"
74. :style="{
75. background: element.color,
76. }"
77. ></div>
78. <div
79. class="content"
80. v-if="element.type == 'station' && element.index == element.lastTransitIndex"
81. >

```

```

82. <div class="title font-size-regular color-text">
83. <!-- {{ element.station.title }} -->
84. 捷運{{ element.station.transit.arrival_stop.name }} 下車
85. <Icon v-if="element.station.notice" icon="bell-ring"></Icon>
86. </div>
87. <div class="tips font-size-small color-text">
88. <span style="color: #17e68e"> 出口 3</span>
89. </div>
90. </div>
91. <div
92. class="datetime font-size-small color-text"
93. v-if="
94. element.type == 'station' &&
95. element.station.transit != null &&
96. element.index == element.lastTransitIndex
97. "
98. >
99. <!-- {{ element.station.datetime }} -->
100. {{ element.station.transit.arrival_time.text }}
101. </div>
102. </div>
103. </div>
104. </template>
105.
106. <script>
107. import mrtDataJson from "~/tools/mrtData.json"; //取得捷運代碼 json
108. export default {
109. props: {
110. route: {
111. type: Array,
112. },
113. },
114. data() {
115. return {
116. mrtData: mrtDataJson,
117. };
118. },
119.
120. computed: {
121. displayElements() {
122. let elements = [];
123. let lastStation = null;
124. let lastTransitIndex;
125. for (let [index, station] of this.route.entries()) {

```

```

126. if (station.transit !== null) {
127. lastTransitIndex = index;
128. }
129. }
130. for (let [index, station] of this.route.entries()) {
131. if (lastStation !== null && station.transit !== null) {
132. elements.push({
133. type: "line",
134. color: lastStation.color,
135. });
136. elements.push({
137. type: "line",
138. color: station.color,
139. });
140. }
141. if (station.transit !== null) {
142. elements.push({
143. type: "station",
144. station: station,
145. lastTransitIndex: lastTransitIndex,
146. index: index,
147. });
148. lastStation = station;
149. }
150. }
151. console.log(elements);
152. return elements;
153. },
154. },
155. };
156. </script>
157.
158. <style scoped>
159. .route {
160. display: flex;
161. flex-direction: column;
162. }
163.
164. .icon {
165. display: flex;
166. flex-direction: column;
167.
168. align-items: flex-start;
169. }

```

```

170.
171. .line {
172. flex: 1;
173. transform: translateX(0.8rem);
174. }
175. .lineBackground {
176. min-height: 8px;
177. width: 4px;
178. }
179.
180. .station {
181. display: flex;
182. }
183.
184. .content {
185. padding: 0rem 1rem;
186. flex: 1;
187. }
188. </style>

```

路徑	Component/Elements/VerticalUBikeRoute.vue
用途	詳細路線中，Youbike 路線顯示
部分程式碼	
<pre> 1. <template> 2. <div class="elements" v-if="displayElements"> 3. <div 4. v-for="(element, index) in displayElements" 5. :key="index" 6. :class="element.type" 7. > 8. <div class="icon"> 9. <Icon v-if="element.type == 'station'" icon="bike"></Icon> 10. <div v-if="element.type == 'line'" class="lineBackground"></div> 11. </div> 12. <div class="content" v-if="element.type == 'station'"> 13. <div class="version font-size-regular color-focus"> 14. {{ element.station.version }} 15. </div> 16. <div class="name font-size-regular color-text"> 17. {{ element.station.name }} 18. </div> </pre>	

```

19. </div>
20. <div
21. class="distance font-size-mini color-text"
22. v-if="element.type == 'station' && element.station.distance"
23. >
24. {{ element.station.distance }}
25. </div>
26. <div
27. class="quantity font-size-mini color-text"
28. v-if="element.type == 'station' && element.station.quantity"
29. >
30. 可租 <span class="color-focus"> {{ element.station.quantity }} </span>
31. </div>
32. <div
33. class="vacancy font-size-mini color-text"
34. v-if="element.type == 'station' && element.station.vacancy"
35. >
36. 空位 {{ element.station.vacancy }}
37. </div>
38. </div>
39. </div>
40. </template>
41.
42. <script>
43. export default {
44.   props: {
45.     route: {
46.       type: Array,
47.     },
48.   },
49.   computed: {
50.     displayElements() {
51.       let elements = []
52.       let lastStation = null
53.       for (let [index, station] of this.route.entries()) {
54.         if (lastStation !== null) {
55.           elements.push({
56.             type: 'line',
57.             color: lastStation.color,
58.           })
59.           elements.push({
60.             type: 'line',
61.             color: station.color,

```

```

62. })
63. }
64. elements.push({
65.   type: 'station',
66.   station: station,
67.   index: index,
68. })
69. lastStation = station
70. }
71. return elements
72. },
73. },
74. }
75. </script>
76.
77. <style scoped>
78. .route {
79.   display: flex;
80.   flex-direction: column;
81. }
82.
83. .icon {
84.   display: flex;
85.   flex-direction: column;
86.
87.   align-items: flex-start;
88. }
89.
90. .mdi-bike {
91.   color: yellow;
92.   font-size: 2rem;
93. }
94.
95. .line {
96.   flex: 1;
97.   transform: translateX(0.8rem);
98. }
99. .lineBackground {
100.   min-height: 8px;
101.   width: 4px;
102.   background: yellow;
103. }
104.
105. .station {
106.   display: flex;

```

```

107. }
108.
109. .content {
110. padding-left: 1rem;
111. height: auto;
112. display: flex;
113. flex: 1;
114. align-items: center;
115. }
116. .version {
117. padding-right: 0.6rem;
118. }
119. .distance,
120. .quantity,
121. .vacancy {
122. display: flex;
123. align-items: center;
124. padding-left: 0.4rem;
125. }
126. </style>

```

路徑	layouts/map.vue
用途	direction API
部分程式碼	
<pre> 1. <template> 2. <div> 3. <section ref="map" id="map"></section> 4. <Loading v-model="loading"></Loading> 5. <div class="wrapper"> 6. <nuxt v-if="map" /> 7. </div> 8. </div> 9. </template> 10. 11. <script> 12. export default { 13. data() { 14. return { 15. map: null, 16. loading: true, 17. position: { </pre>	


```

18. latitude: 25.0487269,
19. longitude: 121.5181263,
20. },
21. }
22. },
23.
24. async mounted() {
25.   if (navigator.geolocation) {
26.     await new Promise((resolve) => {
27.       navigator.geolocation.getCurrentPosition((position) => {
28.         this.position.latitude = position.coords.latitude
29.         this.position.longitude = position.coords.longitude
30.         resolve()
31.       })
32.       setTimeout(resolve, 10000)
33.     })
34.   } else {
35.     console.warn('Browser does not support geolocation API.')
36.   }
37.
38.   this.map = new google.maps.Map(this.$refs.map, {
39.     zoom: 15,
40.     center: new google.maps.LatLng(
41.       this.position.latitude,
42.       this.position.longitude
43.     ),
44.     mapTypeId: google.maps.MapTypeId.ROADMAP,
45.   })
46.   this.$nuxt.$map = this.map
47.   this.loading = false
48.
49.   new google.maps.Marker({
50.     position: new google.maps.LatLng(
51.       this.position.latitude,
52.       this.position.longitude
53.     ),
54.     map: this.map,
55.   })
56. },
57. }
58. </script>
59.
60. <style scoped>
61. #map {

```

```

62. position: absolute;
63. left: 0;
64. bottom: 0;
65. right: 0;
66. top: 0;
67. }
68. wrapper {
69. position: fixed;
70. top: 0;
71. left: 0;
72. right: 0;
73. bottom: 0;
74. padding: 0.5rem;
75.
76. z-index: 1000;
77.
78. width: 100%;
79. height: 100%;
80. box-sizing: border-box;
81.
82. display: flex;
83. justify-content: center;
84. align-items: flex-end;
85. }
86. </style>
1.

```

路徑	tools/favRoute.js
用途	寫入收藏路線，如路線已被收藏過會被刪除，並再彈出 response 信息
部分程式碼	
<pre> 1. import axios from 'axios' 2. import jwtDecode from 'jwt-decode' 3. import config from '~/config' 4. 5. export default { 6. async writeFavRouteToDB(fromLocation, destination) { 7. axios.post(`/api/favRoute/store`, { 8. startPoint: fromLocation, 9. destination: destination 10. }) 11. .then((response) => { 12. // let authData = response.data </pre>	

```

13. // localStorage.setItem('User', JSON.stringify(authData))
14. console.log(response)
15. window.$nuxt.$snackbar({
16. message: response.data.message,
17. })
18. })
19. .catch((error) => {
20. console.error('[favRoute]', 'favRoute insert error.', error)
21. return null
22. })
23. },
24. }

```

路徑	tools/liff.js
用途	LIFF 登入、到站提醒
部分程式碼	
<pre> 1. const liff = require('@line/liff') 2. const config = require('../config.js').default 3. const axios = require('axios') 4. 5. export default { 6. async init() { 7. await liff.init({ 8. liffId: config.liff.liffID, 9. }) 10. }, 11. async login() { 12. await liff.login() 13. }, 14. async checkLogin() { 15. return liff.isLoggedIn() 16. }, 17. async logout() { 18. await liff.logout() 19. await liff.disconnect() 20. }, 21. async getUser() { 22. return liff.getProfile() 23. }, 24. async getAccessToken() { </pre>	

```

25. return liff.getAccessToken()
26. },
27. async getIDToken() {
28. return new Promise((r) => {
29. let interval = setInterval(async () => {
30. if (liff.isLoggedIn()) {
31. let token = await liff.getIDToken()
32. if (token) {
33. clearInterval(interval)
34. r(token)
35. }
36. } else {
37. return null
38. }
39. }, 200)
40. })
41. },
42. async authWithBackend() {
43. return axios
44. .post('/api/auth/login', {
45. idToken: await this.getIDToken(),
46. })
47. .then((response) => {
48. let user = response.data.user
49. window.$nuxt.$store.commit('auth/setUser', user)
50. setTimeout(liff.logout, 5000)
51. return user
52. })
53. .catch((error) => {
54. console.error('[User Login]', 'Login error.', error)
55. return null
56. })
57. },
58. async getBackendAuthStatus() {
59. return axios
60. .get('/api/auth/status')
61. .then((response) => {
62. let user = response.data.user
63. window.$nuxt.$store.commit('auth/setUser', user)
64. return user
65. })
66. .catch((error) => {
67. return null

```

```
68.})
69.},
70.async setNotice() {
71.await liff.init({
72.liffId: config.liff.liffID,
73.})
74.liff.sendMessage([
75.{
76.type: 'text',
77.text: '到站提醒'
78.},
79.])
80.then(function (res) {
81.console.log(res)
82.window.$nuxt.$snackbar({
83.message: '已設定 到站提醒',
84.})
85.return res
86.})
87.catch(function (error) {
88.console.log(error)
89.return error
90.})
91.},
92. }
```

9-2 其他附屬之各種元件

▼表 9-2- 1、附屬程式清單

編號	檔案名稱	功能說明
1	default	Nginx 設定檔

路徑	/etc/nginx/sites-enabled/default
用途	Nginx 設定檔
部分程式碼	
<pre>1. server { 2. listen 80; 3. server_name ntub.fr.rs; 4. 5. charset utf-8; 6. 7. location / { 8. proxy_pass http://127.0.0.1:1002; 9. } 10. 11. location /phpmyadmin { 12. proxy_pass http://127.0.0.1:1000; 13. } 14. 15. location /api { 16. proxy_pass http://127.0.0.1:1001; 17. } 18. 19. location /python { 20. proxy_pass http://127.0.0.1:3000; 21. } 22. 23. location /objectDetection { 24. proxy_pass http://127.0.0.1:3001; 25. } 26. 27. location ~ /\.(!well-known).* { 28. deny all; 29. } 30. } 31. 32. server { 33. listen 1000;</pre>	

```

34.
35. root /usr/share;
36. index index.php;
37.
38. add_header X-Frame-Options "SAMEORIGIN";
39. add_header X-Content-Type-Options "nosniff";
40.
41. charset utf-8;
42.
43. location ~ \.php$ {
44.     fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;
45.     fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;
46.     include fastcgi_params;
47. }
48. }
49.
50.
51. server {
52.     listen 1001;
53.
54.     root /var/www/backend/public;
55.     index index.php;
56.
57.     add_header X-Frame-Options "SAMEORIGIN";
58.     add_header X-Content-Type-Options "nosniff";
59.
60.     charset utf-8;
61.
62.     location / {
63.         try_files $uri $uri/ /index.php?$query_string;
64.     }
65.
66.     location = /favicon.ico { access_log off; log_not_found off; }
67.     location = /robots.txt { access_log off; log_not_found off; }
68.
69.     error_page 404 /index.php;
70.
71.     location ~ \.php$ {
72.         fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;
73.         fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;
74.         include fastcgi_params;
75.     }
76.
77.     location ~ /\.(!well-known).* {
78.         deny all;
79.     }
80. }
81.

```

```
82. server {
83. listen 1002;
84.
85. root /var/www/frontend/dist;
86.
87. add_header X-Frame-Options "SAMEORIGIN";
88. add_header X-XSS-Protection "1; mode=block";
89. add_header X-Content-Type-Options "nosniff";
90.
91. index index.html;
92. charset utf-8;
93.
94. location = /favicon.ico { access_log off; log_not_found off; }
95. location = /robots.txt { access_log off; log_not_found off; }
96.
97. error_page 404 /index.html;
98. }
```


第10章 測試模型

10-1 測試計畫

我們以使用者和管理者角度測試各個功能是否能夠正常運作，主要測試功能如下：

使用者、管理者

1. 網站瀏覽：確認使用者跟管理者能夠正常瀏覽網站
2. 登入：確認使用者跟管理者能正常使用 LINE 帳號登入
3. 功能選單：確認使用者跟管理者能正常瀏覽功能選單

使用者

1. 路線規劃：確認使用者夠正常連線至路線規劃介面
 - 1-1. 搜尋最佳路線：確認使用者能夠正常搜尋最佳路線
 - 1-2. 詳細路線資料：確認使用者能夠正常瀏覽路線資訊
 - 1-3. 車站內部路線：確認使用者能夠正常瀏覽車站內部路線，並正常返回詳細路線資訊
 - 1-4. 下車提醒：確認使用者能夠正常新增下車提醒，並正常接收到提醒通知
 - 1-5. 收藏路線：確認使用者能夠正常新增收藏路線
2. 收藏路線清單：確認使用者能夠連線至路線收藏介面
 - 2-1. 瀏覽收藏路線：確認使用者能夠正常前往相對的收藏之路線
3. 尋找遺失物：確認使用者能夠連線至遺失物介面
 - 3-1. 搜尋遺失物：確認使用者能夠正常搜尋遺失物
 - 3-2. 瀏覽遺失物：確認使用者能夠正常前往相對的遺失物資訊介面，並正常瀏覽遺失物資訊
 - 3-3. 通報遺失物：確認使用者能夠正常登錄丟失之遺失物
4. 回饋金即時報：確認使用者能夠連線至回饋金即時報介面
5. 優惠資訊：確認使用者能夠連線至優惠資訊查詢介面

管理者

1. 尋找遺失物：確認管理者能夠連線至遺失物介面
 - 1-1. 瀏覽遺失物：確認管理者能夠正常瀏覽所有通報遺失物
 - 1-2. 排序遺失物：確認管理者能夠正常排序遺失物
 - 1-3. 搜尋遺失物：確認管理者能夠正常搜尋遺失物
 - 1-4. 遺失物列表：確認管理者能夠正常前往相對的遺失物資訊介面，並正常瀏覽遺失物資訊
 - 1-5. 通報遺失物：確認管理者能夠正常登錄丟失之遺失物的相關資訊

10-2 測試個案與測試結果資料

▼表 10-2- 1、網站瀏覽測試

測試編號	1
功能名稱	網站瀏覽
測試目的	確認使用者跟管理者能夠正常瀏覽到網站
測試流程	連線至網站
預期成果	網站正常顯示
測試結果	正常

▼表 10-2- 2、登入測試

測試編號	2
功能名稱	登入
測試目的	確認使用者跟管理者能夠正常登入
測試流程	點擊 LINE 登入，並允許通過驗證
預期成果	進入登入狀態後連線至網站
測試結果	正常

▼表 10-2- 3、功能選單測試

測試編號	3
功能名稱	功能選單
測試目的	確認使用者跟管理者能正常瀏覽功能選單
測試流程	點選漢堡條，連線至功能選單
預期成果	網站正常顯示
測試結果	正常

▼表 10-2- 4、路線規劃測試

測試編號	4
功能名稱	路線規劃
測試目的	確認使用者能夠正常連線至路線規劃介面
測試流程	點選路線規劃，連線至路線規劃
預期成果	網站正常顯示
測試結果	正常

▼表 10-2- 5、搜尋最佳路線測試

測試編號	5
功能名稱	搜尋最佳路線
測試目的	確認使用者能夠正常搜尋最佳路線
測試流程	輸入起點與目的地，正常連線至路線資訊
預期成果	正常顯示起點、目的地及路線
測試結果	正常

▼表 10-2- 6、詳細路線資料測試

測試編號	6
功能名稱	詳細路線資料
測試目的	確認使用者能夠正常搜尋最佳路線
測試流程	點擊詳細路線資料，正常連線至詳細路線資料
預期成果	正常顯示花費價錢、起點、目的地、轉乘資訊、時間
測試結果	正常

▼表 10-2- 7、車站內部路線測試

測試編號	7
功能名稱	車站內部路線
測試目的	確認使用者能夠正常瀏覽車站內部路線
測試流程	點擊車站內部路線，正常連線至車站內部路線
預期成果	正常顯示車站內部路線
測試結果	正常

▼表 10-2- 8、下車提醒測試

測試編號	8
功能名稱	下車提醒
測試目的	確認使用者能夠正常新增下車提醒，並正常接收到提醒通知
測試流程	在路線規劃完成後，點擊下車提醒
預期成果	在正確的時間點收到提醒推播通知
測試結果	正常

▼表 10-2- 9、收藏路線測試

測試編號	9
功能名稱	收藏路線
測試目的	確認使用者能夠正常新增收藏路線
測試流程	在路線規劃完成後，點擊收藏路線
預期成果	新增成功，且後台有辦法看到
測試結果	正常

▼表 10-2- 10、收藏路線清單測試

測試編號	10
功能名稱	收藏路線清單
測試目的	確認使用者能夠連線至路線收藏介面
測試流程	打開功能選單，點擊收藏路線
預期成果	正常連線至收藏路線
測試結果	正常

▼表 10-2- 11、瀏覽收藏路線測試

測試編號	11
功能名稱	瀏覽收藏路線
測試目的	確認使用者能夠正常前往相對的收藏之路線
測試流程	於收藏路線列表點擊欲察看之收藏路線項目
預期成果	正常連線至所選之收藏路線之路線規劃頁面
測試結果	正常

▼表 10-2- 12、尋找遺失物測試

測試編號	12
功能名稱	尋找遺失物
測試目的	確認使用者與管理者能夠連線至遺失物介面
測試流程	打開功能選單，點擊尋找遺失物
預期成果	正常連線至尋找遺失物
測試結果	正常

▼表 10-2- 13、搜尋遺失物測試

測試編號	13
功能名稱	搜尋遺失物
測試目的	確認使用者與管理者能夠正常搜尋遺失物
測試流程	於尋找遺失物上方點擊搜尋，並輸入關鍵字
預期成果	依照輸入之關鍵字，列出相關之遺失物列表
測試結果	正常

▼表 10-2- 14、瀏覽遺失物測試

測試編號	14
功能名稱	瀏覽遺失物
測試目的	確認使用者能與管理者夠正常前往相對的遺失物資訊介面，並正常瀏覽遺失物資訊
測試流程	點擊欲查看之遺失物
預期成果	正常顯示所選之遺失物資訊的照片、物品名稱、物品顏色、物品站點、通報時間
測試結果	正常

▼表 10-2- 15、通報遺失物(使用者)測試

測試編號	15
功能名稱	通報遺失物
測試目的	確認使用者能夠正常通報丟失之遺失物
測試流程	點擊通報，並輸入遺失物資訊(照片、物品名稱、物品顏色、物品站點、通報時間)
預期成果	新增成功，且後台有辦法看到
測試結果	正常

▼表 10-2- 16、回饋金即時報測試

測試編號	16
功能名稱	回饋金即時報
測試目的	確認使用者能夠連線至回饋金即時報介面
測試流程	點擊回饋金即時報
預期成果	正常顯示使用者之回饋金資訊(目前搭乘次數、消費金額、回饋級距、回饋金額)
測試結果	正常

▼表 10-2- 17、優惠資訊測試

測試編號	17
功能名稱	優惠資訊
測試目的	確認使用者能夠連線至優惠資訊查詢介面
測試流程	點擊優惠資訊查詢
預期成果	正常連線至優惠資訊介面
測試結果	正常

▼表 10-2- 18、遺失物通報(管理者)測試

測試編號	18
功能名稱	通報遺失
測試目的	確認管理者能夠正常通報拾獲之遺失物
測試流程	點擊通報，並輸入遺失物資訊(照片、物品名稱、物品顏色、物品站點、通報時間)
預期成果	新增成功，且後台有辦法看到
測試結果	正常

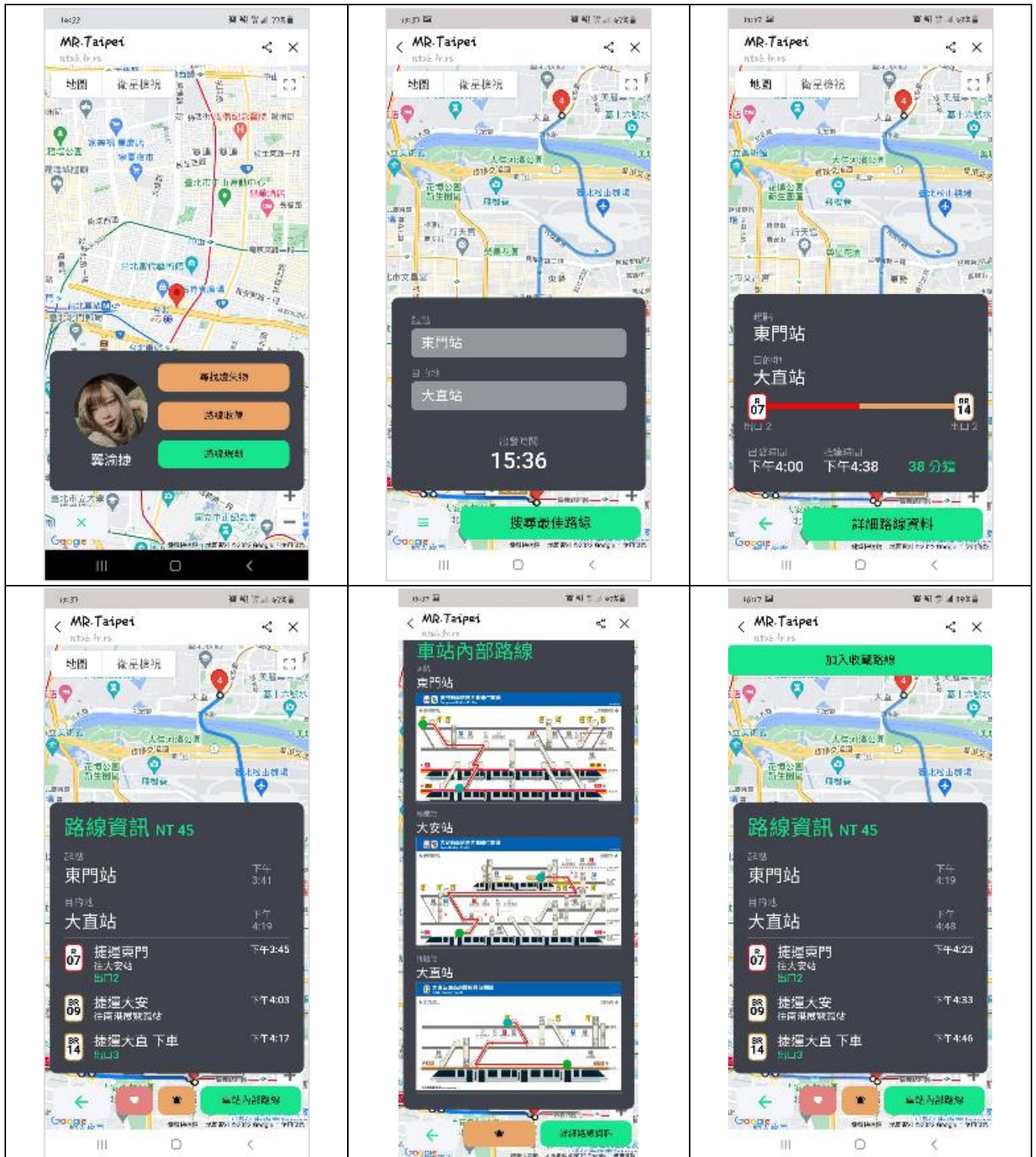
第11章 操作手冊

▼表 11-1- 1、操作頁面

		
<ol style="list-style-type: none">1. 使用者與管理者加入 LINE 官方帳號前(尚未加入好友)的加入好友畫面2. 使用者與管理者加入 LINE 後(含底下各功能選項)的畫面3. 使用者與管理者登入的畫面		

第12章 使用手冊

▼表 12-1- 1、使用者的使用手冊



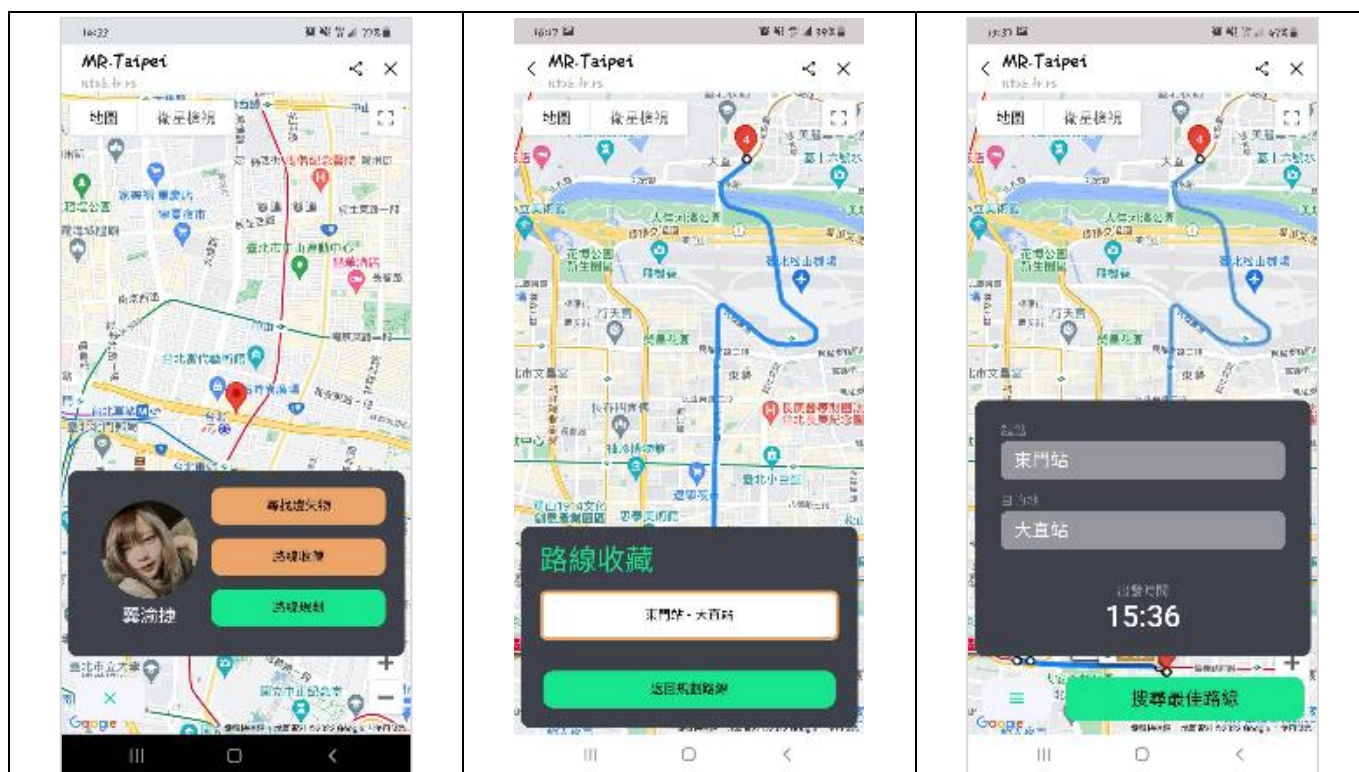
規劃路線：

- 點擊規劃路線後，輸入起點與目的地，點擊尋最佳路線，即可查看路線摘要。
- 點擊詳細路線資料，可查看詳細路線資訊。
- 點擊車站內路線，可查看內部路線圖。
- 點擊愛心圖示，即可收藏路線。
- 點擊鈴鐺圖示，設定到站通知。



下車提醒：

- 點擊鈴鐺圖示，設定到站通知。
- 下車提醒通知的畫面。



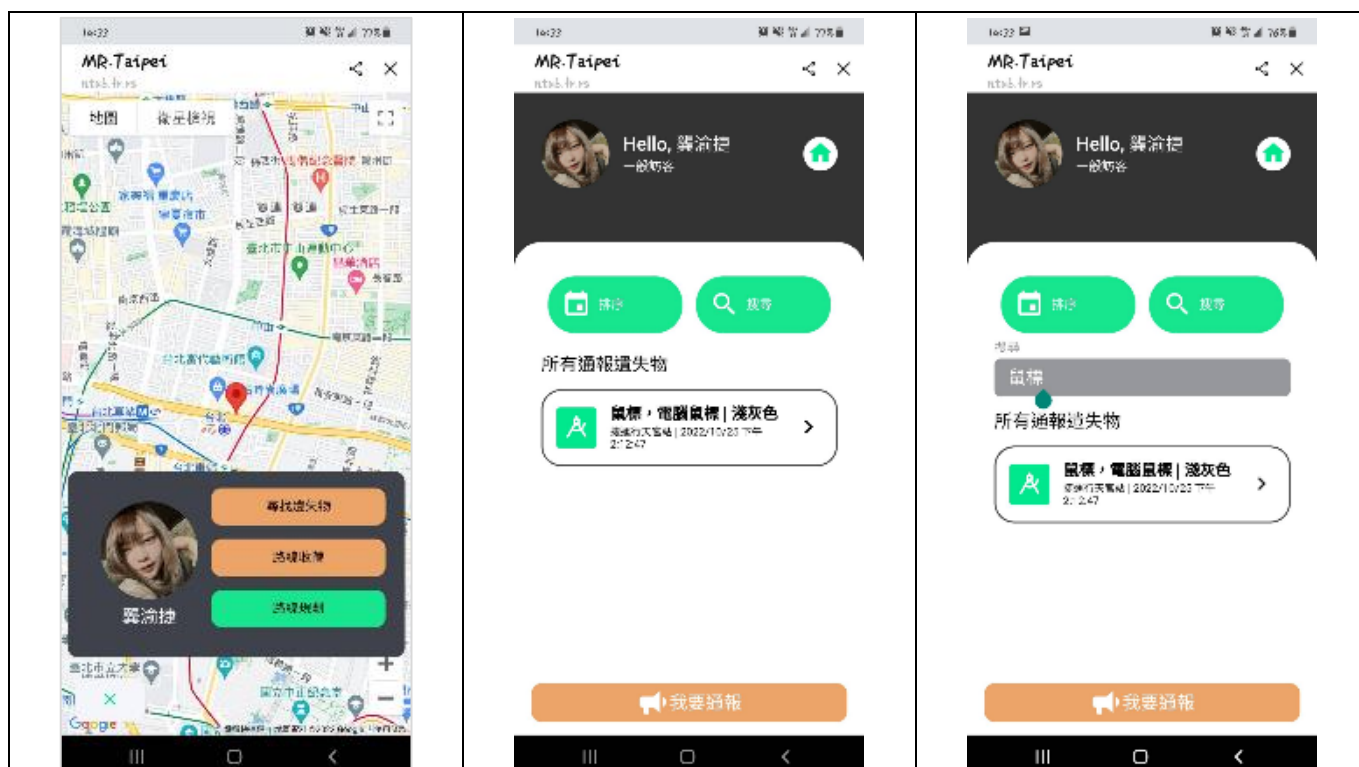
收藏路線：

- 點擊收藏路線，即可查看所收藏之路線清單。
- 點擊相對的收藏項目，即可導向該收藏之路線規劃。



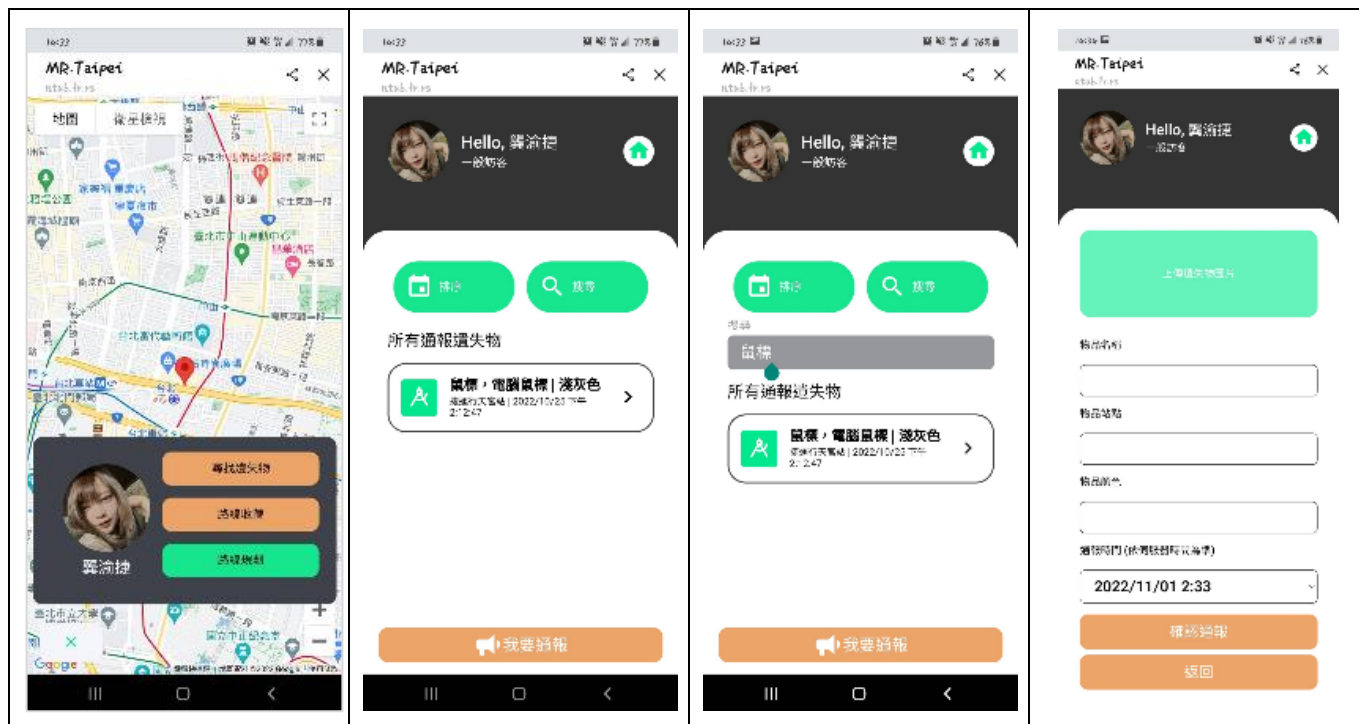
回饋金即時報與優惠資訊查詢：

- 點擊回饋金即時報，可查詢當前累積回饋資訊。
- 點擊優惠資訊查詢，可查詢捷運站附近優惠。



尋找遺失物(使用者)：

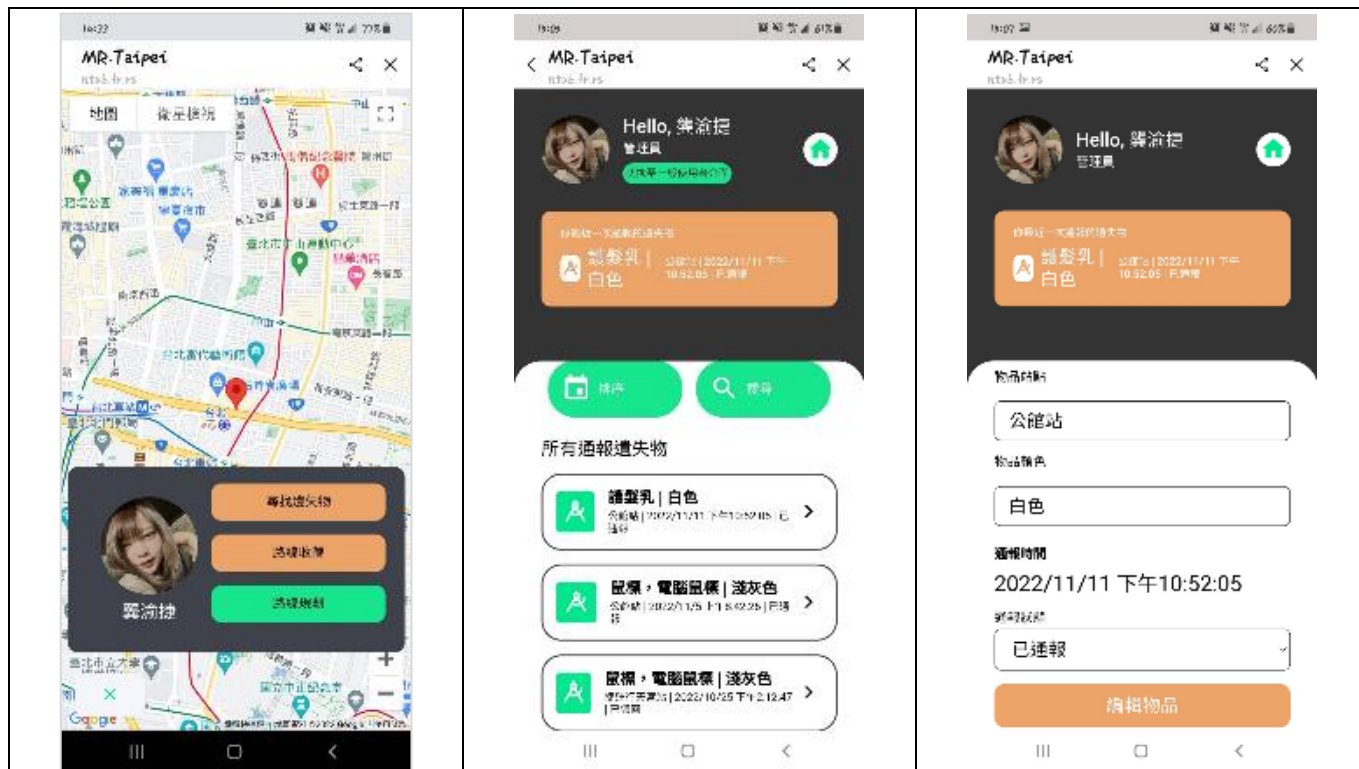
- 點擊尋找遺失物，可瀏覽所有遺失物。
- 點擊搜尋，輸入關鍵字，即可查詢相關遺失物。
- 點擊我要通報，輸入遺失物資訊或是透過相機掃描，點選確認即可完成通報。



通報遺失物(使用者)：

- 點擊尋找尋找遺失物，可瀏覽所有遺失物。
- 點擊搜尋，輸入關鍵字，即可查詢相關遺失物。
- 點擊我要通報，可透過手動輸入，亦可使用相機拍攝掃描輸入，點選確認後完成通報。

▼表 12-1- 2、管理者的使用手冊



通報遺失物(管理者)：

- 點擊相對應的遺失物項目，可編輯遺失物資訊，包含名稱、顏色、站點、狀態。

第13章 感想

王佑強：

這次的專題受眾設定為北漂青年，剛好大部分成員皆為第一次北上念書，在一開始搭乘時也感到陌生與害怕，於是先將題目在臺北捷運盃黑客松進行提案，在獲取建議與方向後，開始著手進行開發。

開發其中雖然遇到過技術上的困難，但在上網自學並與組員討論過後，我變得更上手了，一開始從功能可行性評估到發想如何實現，一路走來完成了許多功能，如：使用最短路徑演算法，再捷運站內剖面圖進行運算後，使用 OpenCV 進行繪製，並且搭配大量的政府公開資料，減少資料庫負擔，讓資料更即時且有更好的讀取效率。在完成路徑規劃後，我也學習使用 OpenVINO Toolkit 進行影像辨識，為了取得物品名稱與顏色，先進行影像前處理，將物品與背景分離後，進行運算並帶出結果。

製作專題期間也很感謝組員們的配合，雖然遇到疫情遠距上課，但大家都有把時程安排好，每個人也都學習開發技術，更讓我們再 27 屆的資服競賽中獲得兩個佳作的佳績。

柯玄娜：

起初，沒有相關經驗的我，一直很擔心自己無法幫上團隊的忙，更怕會拖累團隊，所以我更加努力地學習，無論是在網路上搜尋各方面的資訊，盡量吸收相關知識，或者直接向組員們討教，都帶給我莫大的幫助，讓我能夠繼續為團隊貢獻微小的力量。

在這次的專題中我們實作了網頁系統，配合 LINE 提供之相關 API 技術。其中，我覺得最有趣的部分是能夠接觸到 LINE 的不同面向，從使用者變成開發者，是我沒有想像過的事！在進行開發時我學習了像是串接 Messaging API 以及研究 LIFF 應用等，要熟悉這些技術並不是件容易的事，過程中透過不斷的嘗試和改進，慢慢地實現心中的各種想法，是最開心的事情！

非常感謝組長和組員們的協助，每當我碰到不理解的事情時，大家總是耐心地指導、引導我，讓我的工作得以順利進行，謝謝大家！！

錢慧蓁：

這次的專題讓我學到了很多以前沒接觸過的 Python 或 JavaScript 這些語言，甚至是一些更進階的專業知識。透過這次專題我看到了 LINE 不只是通訊軟體，透過 LINE 連結至網頁與各式各樣 API 合作達到了其它我們想要呈現的各種結果。

最一開始我其實對於製作一個完整的網頁、系統沒有太多的經驗，不過我非常感謝我的組員們和組長在這過程中給了我許多的幫助，讓我慢慢學習到了要如何蒐集資料。從最初連從哪裡開始做都不懂，一直到現在能自己去尋找資料並解決問題，也學會使用許多相關的專業工具，更讓我進一步的認識到了人與人之間的溝通合作有多麼重要，除了讓我們可以發揮各自的所長外，大家還可以一起學習進步，還能彼此互相督促，在遇到挫折時，給予彼此鼓勵。

除了學習到了製作網頁系統的方法、技巧和經驗外，更讓我體會到了跟大家一起努力的成就，謝謝我辛苦的組員們及組長，以及老師給我們的建議，讓我們可以順利完成這次的專題。

李崧毓：

這次專題讓我學到了如何從一個小小的想法經過團隊合作後變成一個完整的專案，讓許多人都可以用我們的專案，我覺得這是一個很有成就感的事情，而且在開發過程中，與他人的合作也可以讓我在未來的各種事項中比其他人擁有更多的合作經驗，程式方面則是學習到了沒有接觸過的 PHP 以及 Vue.js 前後端的程式語言，讓我擁有更多技能，可以面對更多未來的挑戰。

龔渝捷：

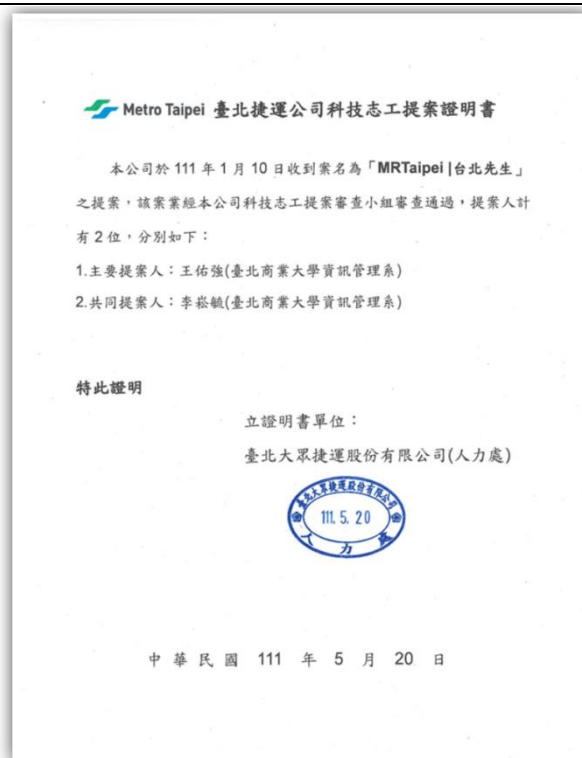
這次的專題，對我來說是很棒的經歷，從高中到二技，第一次做如此完整的專題，從開始的分工，到後面實作，到現在已經要結尾了，從來沒想過自己也能完成這項專題。最開始的心情其實是很焦慮與徬徨，不知道這專題要有怎麼樣的開頭，也不知道自己是否能滿足組員的期望，做好一切的工作，還好有組員們的幫忙，在我需要幫忙時，來支援我，在我碰到瓶頸時，給我靈感，很榮幸能與他們同組。

專題真的是一個很好的經驗，期間需要很多的討論、尋找資料、撰寫文件等，每一項都是要經過無數次的改正才能得到結果，偶爾也會有一些碰撞，但每次的碰撞都是為了能讓專題變得更好，也是讓我們成長的考驗。此次專題，讓我印象深刻的就是 LINE Notification，以前都不知道原來 LINE 有這麼多功能，而且原比想像的還要困難、複雜，最開始寫程式到結合 LINE 進行運作，都是一件大工程，過程中，遇到了許多挫折，像是照著影片做，卻失敗，幸好有隊友們強力教學，才得以讓我完成。很開心能接觸到 LINE，讓我更加了解到 LINE 的整體運作，也藉此機會讓我認識到 PYTHON 這個語言，今後，我也會不斷精進自己，努力朝著專業的目標前進。

第14章 參考資料

- Open data
 - <https://data.gov.tw/dataset/137993>
 - <https://data.gov.tw/dataset/73233>
 - <https://ptx.transportdata.tw/MOTC/v2/Rail/Metro/StationExit/TRTC>
 - <https://ptx.transportdata.tw/MOTC/v2/Rail/Metro/LineTransfer/TRTC>
 - <https://data.taipei/api/v1/dataset/893c2f2a-dcfd-407b-b871-394a14105532?scope=resourceAquire>
 - <https://data.taipei/api/v1/dataset/c32998fa-c9b2-4324-9016-e19fbe0815f1?scope=resourceAquire>
 - <https://data.taipei/api/v1/dataset/42694bb8-fc9d-4d82-af7a-0ad6f4c9ad57?scope=resourceAquire>
- 其他參考資料
 - <https://developers.line.biz/en/docs/line-login/>
 - <https://developers.line.biz/en/docs/messaging-api/>

附錄



▲附錄-1、科技志工提案證明書



▲附錄-2、資服創新競賽(資訊應用組)



▲附錄-3、資服創新競賽
(臺北生活好便利創新應用組)

評審建議事項	修正情形
<ol style="list-style-type: none"> 1. 站區內迷路，要怎麼解決是重點，請說明站內地圖如何開發？ 2. 若在捷運站內，訊號被干擾要怎麼解決？有什麼備案？ 3. YouBike 站點資訊如何與系統串接？ 4. 使用北捷 YouBike 的 OPEN DATA 遺失物照片問題（物品不見了怎麼拍照）？回饋金資料如何與系統串接？ 5. 計畫書可以將獲獎紀錄以及捷運計畫入選部分進行補強。 6. 即時資訊應該是定期更新至自己的資料庫之後再進行提供。 7. 相關系統比較表，也許我們不需要取代其他競爭者(因為其他的競爭者很強大)，而我們具有某種程度的互補性。 8. 最短路徑也許可以更加完整，或著透過不同的方式來進行(也許可以不一定要透過剖面圖來進行)。 	<ol style="list-style-type: none"> 1. 透過傳遞使用者從哪個出入口進出、搭乘方向、經過站點給 API 進行運算，取得最佳路徑後回傳路徑與剖面圖。 2. 先將地圖儲存在 LINE 聊天記錄中，事先預載後就算沒網路也可以正常讀取。 3. 使用北捷 YouBike 的 OPEN DATA。 4. 遺失物照片主要由站務員上傳，如旅客剛好有遺失物照片，可以提升比對效率。回饋金資料仍在與北捷公司爭取 API 相關資料。 5. 已修正，在附錄中。 6. 正在修正中。 7. 已修正。 8. 會加入考量之中。