

國立台灣師範大學資訊工程研究所
碩士論文

指導教授：林均翰 博士

共同指導：葉耀明 博士

結合動態情境資料分析之鏈結開放資料應用
研究

Research on Linked Open Data Applications Using
Dynamic Context Data Analysis

研究生： 余建穎 撰

中華民國 105 年 6 月

摘要

結合動態情境資料分析之鏈結開放資料應用研究

余建穎

政府開放資料現今在各國漸漸盛行，從 2009 年美國和英國政府開放資料平
台上線後，各個國家開始發展自有的政府開放資料平台。臺灣也不例外，臺北
市政府首先推動台北市開放資料平台，給予民眾應用或資料查詢開發使用。但
綜觀目前台灣各區的開放資料，雖然蘊含了各領域的大量資料，但卻遇到許多
窘境，其中包含格式雜亂、格式不容易整合應用、資料授權複雜...等問題，使
得民眾或開發者針對這些資料不易應用與開發服務。

本研究利用鏈結資料概念來用於開放資料上形成鏈結開放資料應用，透過
統一轉換成 RDF 格式來描述資料並加以進行鏈結於雜亂資料中相互關係來增強
資料可讀性和可行性，能使資料更容易被使用，開發上也更具有彈性。本研究
利用臺北市的地理位置為情境感知的鏈結基礎，利用行動裝置定位功能，於臺
北市開放資料平台中採用交通類別的停車場、微笑單車以及捷運形成鏈結開放
資料集，再利用開放資料於時間軸上數據分析和列車到站時間計算，讓使用者
能透過當下地理位置以及自行設定條件進行查詢、計算與服務整合串聯成由開
車、停車、租借微笑單車以及搭乘捷運列車，進一步地搭配旅遊景點的動態情
境資料分析或自行設計一套專屬於自己的客製化一日遊行程的情境分析應用。

關鍵字：政府開放資料、鏈結開放資料、動態情境資料分析

Abstract

Research on Linked Open Data Applications Using Dynamic Context Data Analysis

Chien-Ying Yu

Many countries has been developed their own open government data platform after Data.gov.(England) and Data.gov.(America) platform have been lunched since 2009. Government data platform has been established in each country, including Taiwan. Taipei City Government is the first local government which promotes the Taipei open data platform first and provides people to use this platform. Although there are amounts of data about various areas in the open data of Taiwan, it faces many data format problems including inconsistency, licensing etc.

This research uses the concepts of linking open data to analyze correlated open data via semantic web technology. The linking open data are transformed to an RDF (Resource Description Framework) format to enhance the readability and systematic development. Besides, our system provides a semantic web access interface which can enhance the usability of data. We develop a location-based context-aware service based on Taipei city open data to provide various smart city services. Through the integration of parking lots, public bicycle and MRT data, users can use their mobile devices to arrange activities according to the location of dynamic context data analysis and set their own requirements to design a one-day-traveling schedules.

Keyword : open government data, linked open data, dynamic context data analysis

誌謝

歷經兩年不斷地努力學習，透過持續討論的建議以及修改實作與內容，終於完成研究和撰寫這份論文，在此由衷地感謝葉耀明及林均翰教授研究的過程中教導與鼓勵，也很開心能在研究所中參加全國大專院校資訊應用服務創新競賽中獲得佳作，增加自己的人生歷練與台上報告的技巧，也在競賽中尋找出自已研究的方向而產生動力繼續深入研究。謝謝老師在我研究中無頭緒時候，提點出問題所在且給予我許多創意想法，讓我深刻了解自己還有很多不足的地方，需要更精進地學習。

感謝葉耀明老師、林均翰老師、陳永昇老師以及許榮隆老師，能夠於當天百忙抽空前來擔任口試委員，每位老師在報告中所提出的問題及建議，讓我在論文中發現缺陷而進行修正，讓論文的內容更加完整。

感謝家人提供給我學習上的支援，讓居住在臺南的我前來台北就讀學習，其中包括房租、生活開銷以及精神上的鼓勵。大學原本念資管系提升至研究所念資工系，在課堂中可以接觸到工程方面的知識，雖然在學習上很吃力，但是老師們的熱心教導讓我可以慢慢地進入狀況，透過在學校研究時所漸漸地產生興趣，也會開始撰寫自己程式，研發出不一樣有趣的應用，也謝謝曾經幫助過我的同學，在研究所中透過健身、聚會以及共同熬夜學習增加彼此友誼，讓我可以順遂地完成研究的成果，希望各位同學未來的路上可以順順利利，感謝兩年間的陪伴可以度過美好的時光。

目錄

附表目錄.....	vi
附圖目錄.....	vi
第一章 繢論.....	1
1.1 研究背景.....	1
1.2 研究動機與目的.....	4
第二章 文獻探討.....	6
2.1 情境感知(Context Awareness)	6
2.2 開放資料(Open Data).....	8
2.3 政府開放資料(Open Government Data).....	10
2.4 鏈結開放資料(Linked Open Data)	11
2.5 資源描述框架(Resource Description Framework)	13
2.6 SPARQL.....	15
2.7 Jena API	16
2.8 Google API.....	17
2.9 PHPMyAdmin.....	18
第三章 系統規劃.....	19
3.1 系統整體規劃.....	19
3.2 鏈結資料圖(Linked Data Diagram)	21
3.3 情境感知(Context Awareness)	27
3.4 縮減範圍搜尋(Reduce Size)	28
3.5 微笑單車數據分析與推薦	30
3.6 捷運不同格式整合運用	33
3.7 捷運列車到站時間分析	35
3.8 臺北一日遊規劃行程之情境應用	36
3.9 使用案例(Use Case Diagram)	48
3.10 活動圖(Activity Diagram)	51
3.10.1 瀏覽所有微笑單車資訊.....	51
3.10.2 瀏覽合適微笑停車場資訊.....	52
3.10.3 瀏覽常用微笑單車站點.....	53
3.10.4 使用者新增常用微笑單車站點.....	54
3.10.5 使用者刪除常用微笑單車站點.....	55
3.10.6 瀏覽所有捷運行駛列車資訊.....	56
3.10.7 瀏覽最近捷運站點資訊.....	57
3.10.8 查詢捷運行駛列車到站時間資訊.....	58
3.10.9 查詢捷運站點票價以及無障礙電梯資訊.....	59
3.10.10 臺北一日遊規劃行程之情境應用	60

3.10.11 分析所處停車場周圍微笑單車或捷運站點.....	61
3.10.12 瀏覽分析及推薦資料.....	63
3.10.13 縮減搜尋範圍(Reduce Size)	64
3.10.14 動態更新政府開放資料.....	66
3.10.15 使用狀況分析.....	67
第四章 系統開發與實作.....	68
4.1 系統架構.....	68
4.2 類別圖(Class Diagram)	71
4.3 循序圖(Sequence Diagram).....	74
4.3.1 瀏覽所有微笑單車站點資訊.....	74
4.3.2 瀏覽合適微笑單車站點資訊.....	75
4.3.3 瀏覽常用站點並查看	76
4.3.4 新增常用微笑單車站點.....	78
4.3.5 刪除常用微笑單車站點.....	79
4.3.6 瀏覽所有捷運行駛列車資訊.....	80
4.3.7 瀏覽最近捷運站點資訊.....	81
4.3.8 查詢捷運列車到站時間資訊.....	82
4.3.9 查詢捷運站點票價以及無障礙電梯資訊.....	83
4.3.10 臺北一日遊規劃行程之情境應用	84
4.3.11 停車場至微笑單車或捷運跨資料集應用	85
4.3.12 縮減範圍搜尋(Reduce Size)	87
4.4 實作畫面.....	89
4.4.1 瀏覽所有微笑單車站點.....	89
4.4.2 瀏覽合適微笑單車站點.....	91
4.4.3 瀏覽、新增、刪除常用站點.....	92
4.4.4 分析所在停車場並透過停車場尋找周圍微笑單車或捷運站點(跨 資料集應用).....	94
4.4.5 瀏覽所有捷運行駛列車資訊.....	95
4.4.6 瀏覽最近捷運站點資訊.....	96
4.4.7 查詢捷運列車到站時間.....	97
4.4.8 查詢捷運站點票價、無障礙電梯以及轉車資訊.....	99
4.4.9 臺北一日遊規劃行程之情境應用	101
第五章 結論與未來發展.....	103
5.1 結論.....	103
5.2 未來發展.....	105
參考文獻.....	106

附表目錄

表 3-1 Data Schema - info	24
表 3-2 Data Schema - location.....	24
表 3-3 Data Schema - time.....	24
表 3-4 Data Schema - parking garage.....	24
表 3-5 Data Schema - ubike station	25
表 3-6 Data Schema - mrt	25
表 3-7 Data Schema - 停車場資訊.....	25
表 3-8 Data Schema - 停車場動態格位資訊.....	25
表 3-9 Data Schema - 微笑單車資訊.....	26
表 3-10 Data Schema - 常用微笑單車站點.....	26
表 3-11 Data Schema - 捷運站點.....	26
表 3-12 使用 Reduce Size 的執行時間差異(第一次執行).....	29
表 3-13 使用 Reduce Size 的執行時間差異(第二次執行).....	29
表 3-14 使用 Reduce Size 的執行時間差異(第三次執行).....	29
表 3-15 使用 Reduce Size 的執行時間差異(第四次執行).....	29
表 3-16 使用 Reduce Size 的執行時間差異(第五次執行).....	29
表 3-17 上禮拜前 3 個小時之微笑單車的使用率以及可用率	31
表 3-18 上禮拜後 3 個小時之微笑單車的使用率以及可用率	31
表 3-19 目前時間前 3 個小時之微笑單車使用率以及可用率	31

附圖目錄

圖 1-1 鏈結資料圖(Linked Data Diagram).....	3
圖 1-2 開放資料類別個數	4
圖 1-3 個別交通工具 App	5
圖 2-1 情境感知服務流程	7
圖 2-2 Linked Open Data 鑑定之標準，5 個 Level.....	13
圖 2-3 RDF 範例一	14
圖 2-4 RDF 範例二	15
圖 2-5 RDF 關係之概述圖	15
圖 2-6 Jena 應用之流程圖	16
圖 3-1 系統之互動圖	19
圖 3-2 系統之規劃圖	19

圖 3-3 臺北市政府局部的開放鏈結資料圖	22
圖 3-4 臺北市開放鏈結資料構想圖	23
圖 3-5 微笑單車時間軸分析	32
圖 3-6 捷運不同格式之應用	33
圖 3-7 不同格式統一轉換圖	34
圖 3-8 不同觀點之情境應用圖	37
圖 3-9 行事曆記錄圖	39
圖 3-10 一日遊交通搭乘圖	40
圖 3-11 一日遊規劃行程圖 - 範例一	41
圖 3-12 交通工具應用顯示圖 - 範例一	42
圖 3-13 搜尋臺北一日遊資訊	43
圖 3-14 捷運一日遊行程資訊圖	44
圖 3-15 一日遊規劃行程圖 - 範例二	46
圖 3-16 交通工具應用顯示圖 - 範例二	47
圖 3-17 使用案例圖	48
圖 3-18 活動圖 - 瀏覽所有微笑單車資訊	51
圖 3-19 活動圖 - 瀏覽合適微笑單車資訊	52
圖 3-20 活動圖 - 瀏覽常用微笑單車站點	53
圖 3-21 活動圖 - 使用者新增常用微笑單車站點	54
圖 3-22 活動圖 - 使用者刪除常用微笑單車站點	55
圖 3-23 活動圖 - 瀏覽所有捷運行駛列車資訊	56
圖 3-24 活動圖 - 瀏覽最近捷運站點資訊	57
圖 3-25 活動圖 - 查詢捷運行駛列車到站時間資訊	58
圖 3-26 活動圖 - 查詢捷運站點票價以及無障礙電梯資訊	59
圖 3-27 活動圖 - 一日遊之規劃行程安排	60
圖 3-28 活動圖 - 分析所處停車場周圍微笑單車或捷運站點	61
圖 3-29 活動圖 - 瀏覽分析及推薦資料	63
圖 3-30 活動圖 - 縮減搜尋範圍(Reduce Size)	64
圖 3-31 活動圖 - 動態更新政府開放資料	66
圖 3-32 活動圖 - 使用狀況分析	67
圖 4-1 系統架構圖	68
圖 4-2 系統類別圖	71
圖 4-3 循序圖 - 瀏覽所有微笑單車站點	74
圖 4-4 循序圖 - 瀏覽合適微笑單車站點	75
圖 4-5 循序圖 - 瀏覽常用站點並查看	76
圖 4-6 循序圖 - 新增常用站點	78
圖 4-7 循序圖 - 刪除常用站點	79
圖 4-8 循序圖 - 瀏覽所有捷運行駛列車資訊	80

圖 4-9 循序圖 - 瀏覽最近捷運站點資訊.....	81
圖 4-10 循序圖 - 查詢捷運列車到站時間資訊.....	82
圖 4-11 循序圖 - 查詢捷運站點票價以及無障礙電梯資訊.....	83
圖 4-12 循序圖 - 一日遊之規劃行程安排.....	84
圖 4-13 循序圖 - 停車場至微笑單車跨資料集應用	85
圖 4-14 循序圖 - 縮減範圍搜尋(Reduce Size)	87
圖 4-15 瀏覽所有微笑單車以及查看站點資訊	89
圖 4-16 微笑單車站點之 Google Map 導航和使用狀況與推薦	90
圖 4-17 自訂搜尋範圍頁面以及輸入需求範圍後的搜尋結果	91
圖 4-18 瀏覽常用站點以及新增常用站點	92
圖 4-19 刪除常用站點(台北市信義區松壽路 11 號).....	93
圖 4-20 瀏覽常用微笑單車站點資訊(台北市中正區汀洲路三段 203 巷 25 號).....	93
圖 4-21 停車場至微笑單車或捷運跨資料集之應用	94
圖 4-22 瀏覽所有捷運行駛列車和更新資訊	95
圖 4-23 瀏覽最近捷運站點和 Google Map 導航	96
圖 4-24 查詢捷運列車到站時間(相同路線).....	97
圖 4-25 查詢捷運列車到站時間(不同路線).....	98
圖 4-26 查詢捷運票價以及無障礙電梯資訊	99
圖 4-27 查詢捷運轉車以及 Google 導航站點資訊	100
圖 4-28 查詢所有停車場、Ubike 以及捷運資訊以及輸入行程	101
圖 4-29 規劃行程顯示圖以及檢視視覺化地圖	102

第一章 繢論

1.1 研究背景

現今智慧型手機已越來越普及，而根據統計[1]，從 2013 年開始台灣的智慧型手機普及率已高達 51%，台灣民眾若出門必定攜帶智慧型手機已高達 81%，也就是說幾乎每兩人之中就有一位隨身攜帶智慧型手機。智慧型手機對於人類來說越來越貼近生活，顯現人們漸漸離不開沒有智慧型手機的生活，大多數人們所從事的休閒、娛樂、通訊、搜尋....等大部分活動都藉由智慧型手機，甚至有不少人感覺：『寧可沒有電視，但是一定要有支智慧型手機』，凸顯某方面智慧型手機漸漸取代電視，伴隨著智慧型手機功能方面日益壯大，除了手機基本的通訊功能之外，也包含 GPS 定位、上網瀏覽、以及各式各樣的感測器應用....等等，智慧型手機所提供之加值服務越來越多元化。以上敘述智慧型手機的優點具備有：1.貼近民眾的生活、2.強大且方便的加值服務、3 各種感測晶片集於一身，綜合這三大特點使得智慧型手機亦成為情境感知(稱 Context Awareness)系統的重要發展平台。

最近幾年各國政府陸續推廣政府開放資料(稱 Open Government Data)，期許將政府資料公開化，讓一般民眾能自由取用且善加運用，讓資料能有更好的價值。台灣現今已有不少的政府單位率續類似開放資料的平台，讓一般民眾可以利用簡單的搜尋方式找到資料。在 2011 年九月時，臺北市政府開放平台(稱

Data.Taipei)是第一個上線提供給大眾做使用的平台[2]。緊接著政府資料開放平台也伴隨著上線[3]，到至今全台灣大多縣市也隨之有開放平台[15]。隨著時代演進，開放資料亦日益壯大，許多以開放資料為基礎的加值服務推陳出新，成長般的速度已超乎想像。

當網際網路迅速地發達，各式各樣種類資料遍散佈於每個小角落，但是大多數遍散佈於網際網路的資料皆以超文本(Hypertext)方式做呈現，故資料彼此間關係相較薄弱，而且不具備機器可讀性(Machine-readable)，其衍生出整合與分析上的窘境，除此之外超文本是將網頁超連結的方式互相鏈結，而不是利用資料本身來相互鏈結。在資料過度飽和的時代，於是越來越多人強調資料本身潛在力做為相關應用，期望資料能夠通用且機器可讀性的格式來儲存，並且利用資料相關特性可以相互做鏈結，在此造就出 Linked Data 所誕生的意義。

Linked Data 是把 W3C 制定協定作為基礎給予描述資料的方式，然而藉由公開自己的資料與其他開放資料做成鏈結，將資料群集越來越壯大，重要的是資料蘊含著無限潛在力，其便利性也更加強大。

以圖 1-1 為鏈結資料圖，而圖中包含有一個資料集為『DBpedia』[17]，在 2007 年是由 OpenLink Software、Freie Universität Berlin 以及 University Of Leipzig 共同發表出 DBpedia 資料集，而 Wikipedia 是具有結構的資料組合而成，其中皆有一個 Info Block，接著 DBpedia 櫄取自於 Wikipedia 的 Info Block，針對每筆不同資料做描述且加以鏈結，造就出 Linked Data。DBpedia 是

目前廣為人知的 Linked Data，最近幾年啟發許多人響應，至今已高達 50 個以上的資料集與 DBpedia 彼此連結，讓 Linked Data 迅速擴張。

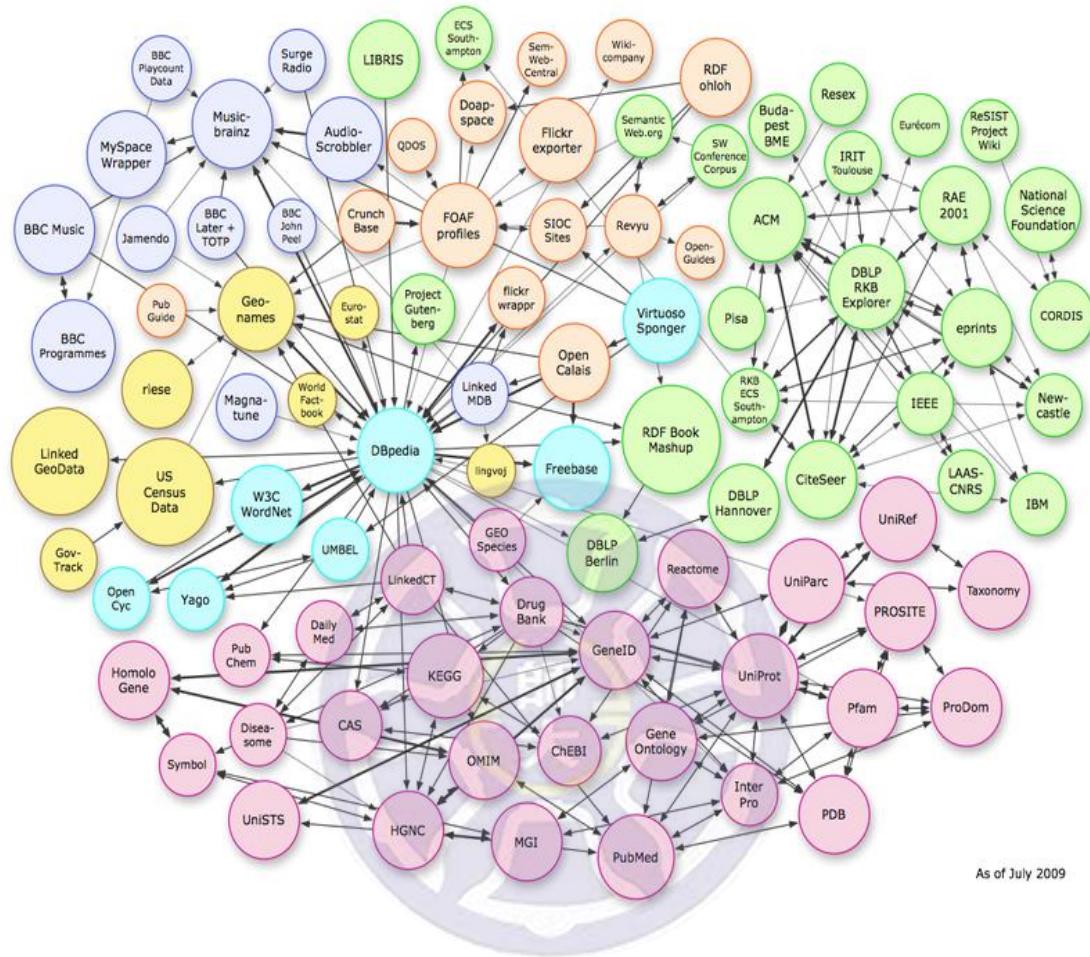


圖 1-1 鏈結資料圖(Linked Data Diagram)

1.2 研究動機與目的

近幾年政府開放資料風靡各國，連帶著網頁應用以及手機應用程式的興起。現今臺北市政府開放資料平台已高達 1828 種資料來源，其中大部分與民眾息息相關的交通資訊以及公共訊息，如圖 1-2。因此臺北市政府希望能響應開放資料的使用，每年舉辦數場次的資訊創意競賽，為了吸引開發者透過臺北市政府開放資料平台的眾多資料集來開發出各種加值服務與應用。臺北市政府也依照本身的開放資料自行研發行動 App，現今已在 Android Store 和 Google Play 中上架，再加上額外有業餘開發的 App，產品多樣到不計其數。

目前雖有許多已政府開放資料為基礎的加值服務與應用，但絕大多數僅將政府給予的資料做統整，提供給一般民眾方便查詢、瀏覽的使用，缺乏針對這些資料實質意義的分析應用，讓人有遺珠之憾的感覺，為了將這些資料能貼近人們，而手機本身能夠掌握使用者所處的時間、地點、甚至狀態資訊，結合其他加值服務，我們還能了解使用者對於未來行程規劃、對事物的特殊喜好，能進行與 Open Government Data 做結合分析與應用，能給予更貼近使用者的生活資料，而不是僅能被動的查詢。

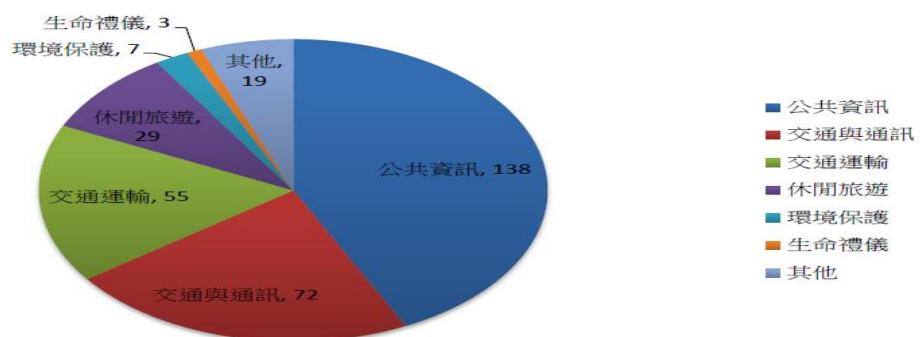


圖 1-2 開放資料類別個數

然而臺北市政府開放資料所提供的資料，在網路上下載的資料都屬於即時資料，而無法利用先前資料做查詢的應用，於是透過歷史的資料紀錄來建立出分析應用機制，讓資料更具有價值。另外市面上在行動裝置商店所下載交通工具的 App 皆僅做單一的應用服務，如圖 1-3，無法針對旅遊方面讓民眾透過一個系統完整搭配使用，於是構想出交通工具整合為一，建立一日遊行程規劃的應用服務，透過旅遊景點進行搭配呈現，可以讓民眾更清楚地了解一系列旅遊的流程。



圖 1-3 個別交通工具 App

研究中主要目的是期望能將混亂無章的 Open Government Data 利用 RDF(全名 Resource Framework，稱資源描述框架)的正規化格式進行資料統一格式，將透過使用者不同需求進一步分類，除了能提高搜尋速度的目的，另外結合情境感知(Context Awareness)應用，達到有效使用政府開放資料的意義，同時亦期望透過應用更貼近使用者的生活。上述實作交通方面的應用，例如：停車場、Ubike 以及 MRT，透過彼此不同資料集進行鏈結的動作，預期能進行不同交通工具的開放資料轉換搭乘租借互相搭配使用，加入行程的規劃能讓原本無相關聯的資料變得更加具有彈性，也能對於旅遊族或上班族更便利的生活。

第二章 文獻探討

2.1 情境感知(Context Awareness)

利用行動裝置偵測、整合使用者目前的狀態資訊及位置資訊，並且使用所得到的資訊加以分析整合，自動地提供符合使用者需求的資訊和服務。系統會整理出使用者所需要的資訊或是針對目前使用者所處情境狀態給予資訊和服務，透過分析資訊和服務可以減少冗長、不必要的資訊，讓使用者更貼近生活與實際操作應用。

最早提出情境感知的概念是在 1994 年 Schilit[24]與 Theimer[25]所提出，主要的概念是將使用者所需要可以善加利用的資訊，依據使用者的情境以及地位置，利用無線傳輸的技術傳遞至使用者。透過感測器偵測使用者的環境狀況，分析判斷服務出最適合的資訊提供給使用者，提升資訊上的效率和服務的舒適度。「情境(context)」是描述使用者在周遭環境中任何非實體或實體的資訊、狀態，非實體包括光亮、時間、風力、時間、經緯度等等，實體則包括人、建築物、地理位置、交通工具、其他物件等等。總而言之，簡單來說情境感知就是和使用者彼此互動所提供的資訊的東西皆可列入參考依據，透過互動所提供的資訊內容皆可以當作「情境資訊」。

情境感知的服務流程有四個步驟:1.確認情境資訊、2.感測器感知、3.情境資訊處理、4.提供服務[6]如圖 2-1：

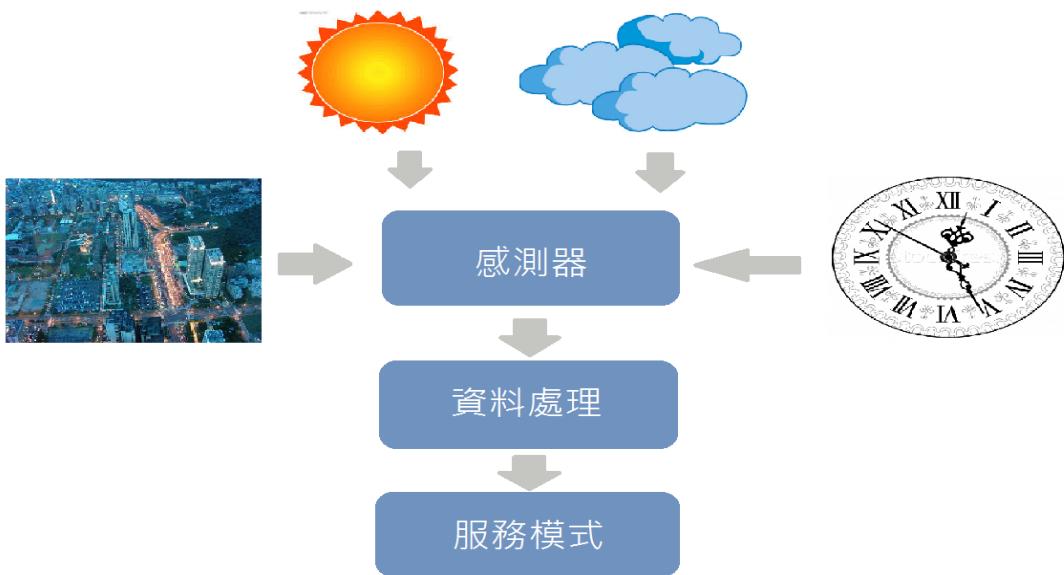


圖 2-1 情境感知服務流程

- 情境感知資訊：年齡、性別、情緒、習慣、喜好、能力等資訊，以及時間、位置、光亮、氣溫、所處環境設施等等的環境資訊。
- 感測器：行動裝置上軟硬體的使用以及傳感器所得到的資訊，例如陀螺儀、GPS、時鐘、臉部辨識、使用者喜好紀錄等等。
- 資訊處理：可分為兩種處理模式，(1)集中式：將行動裝置感測到的情境資訊，利用無線網路傳輸的技術傳送伺服器端，在從伺服器端運算的結果，最後將使用者設定條件或是情境所計算的結果傳送至使用者的行動裝置。
 (2)分散式：在使用者的行動裝置上處理情境資訊的處理加工，並直接得出所需要的結果，提供服務。
- 服務模式：David Kotz 的定義可分為主動服務模式(Active context)以及被動服務模式(Passive context)。所謂主動服務模式為系統依照接收到的情境感

知資訊，來改變系統的行為模式（例如：在重要會議時，手機會依據預先輸入在行事曆上會議的時間，會將鈴聲切換到靜音或者拒接）；而被動服務模式指的是依照接收情境的資訊，改變系統的資訊所呈現的方式（例如：當我們買票時，系統會依據使用者曾經的喜好，在購票的時候，會推薦相關的電影）。

依據情境資訊的內容，可以將資訊分為五種：

- I. 時間資訊：季節、白天夜晚、年、月、日等。
- II. 空間資訊：地理位置、速度、室內室外、海拔等。
- III. 環境資訊：交通、氣溫、濕度等。
- IV. 社會資訊：相關聯絡人資訊、個人身分等。
- V. 設備資訊：停車場格位數、無線上網設備、公車到站狀態等。

2.2 開放資料(Open Data)

指的是經過處理篩選過的資料集，現今 Open Data 檔案格式種類眾多，包含 JSON、CSV、excel、等等眾多複雜的檔案格式。在紛亂的格式中進行篩選的資料中不受任何專利權、著作權或任何管理機構的限制，任何人皆可自由地使用它，且可利用眾多資料進行開發新的服務功能以及加值服務等。因網際網路迅速火熱的發展，Web2.0 興起，慢慢出現各式各樣的「開放運動」在社會上，例如開放原始碼(Open Source)以及技術開放等等，各種領域伴隨著這股熱潮引起新興運動。目前網路上漸漸出現分享各種開放原始碼平台、Linux 的開放技術也漸漸受到矚目以及硬體界也正盛行 3D 列印等等，皆是開放運動所帶來的影響成果。

「內容開放」是目前大家所關注的議題之一，主要是開放各種專業領域所能夠清楚地掌握的資料內容，能夠增加各領域之間的技術交流，Science Common 的前執行長 John Wilbanks 曾提過：「曾經許多科學家表示，在這歷史的一刻，正當我們有足夠的技術與能力將科學資料以全球性的層次來進行發佈，增強彼此之間的合作關係且加速新科技的發展，但是很諷刺地看到我們急著把資訊封閉，並嚴格地禁止在這些知識數據上使用新的技術[13]」。有些人認為大多數資料皆是由於大眾所提供之資訊，但是大部分的使用者卻無法獲知這些資訊，故藉由內容開放的方式，將資料開放給大眾提供參考使用。綜合以上開放資料的運動逐漸興起，讓政府、學界都有 Open data 運動可以善加使用。

依據 W3C eGov Interest Group[26]所提出的定義，開放資料(Open Data)須符合以下兩項標準：

- 格式未加工(Raw Format)：不能對資料本身做額外的加工。
- 機器可讀(Machine-readable)：將資訊的格式建構出可讓電腦或者程式進行讀取且加以處理。

2.3 政府開放資料(Open Government Data)

顧名思義，是由政府所屬的機關管理，政府平台所提供之各式各樣的類型開放資料，其內容大多數皆在政府施政時匯集於一般民眾使用，不僅可提供使用者去讀取，政府施政透明也是其中大宗之一，在世界各國政府正推動的趨勢，經由政府資料的開放化，而促使了跨機關相互的資料可彼此間順利地交流，更能讓廣大民眾的滿足需求，且可增強監督政府的方式。伴隨著智慧型手機普遍的時代，各種應用服務持續推出，政府開放資料也搭上這股順風車，利用有限的資料來源，善用民間的無限想像力，進一步開發出能讓人們能更舒適更貼近的服務應用，來達到政府、業者、民間三者可全贏的局面。

依照行政院院會決議第 3322 次指示[4]，Open data 能增加政府施政透明度、提升民眾的生活品質，且滿足業界所需，對於政府各個層級或者各個部會之間決策的品質都扮演其中的重要決策數據，可想而知開放資料的重要性，各個部門皆應該把重點放在民眾的需求考量為第一優先，在規劃時也需考慮到機器的可讀性。開放資料的類型應優先考量於提升民眾生活品質，如食、衣、住、行、文化、就業、育樂等都是頗為重要的各項生活層面，期許往後透過政府開放資料，進一步促成各個政府機關以及民間彼此發展出多樣的應用與服務，造就出更加完美的生活品質。依照 W3C eGov Interest Group[27]定義，政府開放資料須以下八條原則為標準：

1. Complete：所有的開放資料皆可被任何取用，不能加設密碼或者權限。

2. Primary：所有的資料需未經修改、整合過的，皆為最原始的。
3. Accessible：資料能被任何人所使用，且能做為任何應用。
4. Timely：資料須是最即時的，不可是過時的資料。
5. Machine processable：資料是可被電腦、程式可讀的格式做為儲存，讓資料得以被自動地存取。
6. Non-discriminatory：資料能被任何取用，不會受限於權限之影響。
7. Non-proprietary：能重複使用的存取資料，不影響互斥的存取。
8. License-free：資料沒有明確版權或是法規之限制。但是若部分有合理機密的資料、特別權限之限制或是隱私，則允許能有部分之版權限制。

2.4 鏈結開放資料(Linked Open Data)

鏈結資料(Linked Data)是透過資料之間的關係做成鏈結的一種方法，利用鏈結後所產生的資料會具有結構性。而且是藉由資料彼此之間的鏈結，資料與資料間能夠互相溝通，如此就能提升出資料的價值與用途。鏈結資料(Linked Data)是建構於標準網路的技術之下，例如 RDF、HTTP、URIs 等等，其中用途是在於透過資料結構化的雜亂資料，能讓資料更具有機器可讀且有結構化的性質，更進階地能夠使用電腦進一步做規模化或者自動化邏輯的分析，將資料的數據能完整的發揮出應有的價值。

Tim Berners-Lee 在 2009 年所提出了四項核心標準[19]：

1. 使用 URI 描述出事物或者資源

2. 使用 HTTP URIs 提供所有的資源藉由網際網路能讓使用者去存取且使用
3. 對於各種 URI 的資源提供有用且正確的資訊，進一步使用 SPARQL 和 RDF 去做查詢以及描述
4. 提供鏈結到有關其他事物的 URI

Linked Open Data 基礎建構於 Linked Data 的方法，希望由政府或各種企業所能提供的多項開放資料，進一步鏈結資料的利用方式，能提供給一般民眾做使用，透過 Linked Data 的易分享性、易可讀性，增強開放資料所能帶來的好處。

Tim Berners-Lee 在 2010 所提供出一套能針對 Linked Open Data 的鑑定標準 [20]。根據圖 2-2 中所鑑定之標準，其中共有 5 個 Level，若 Level 越高所表示的是資料越貼近地符合 Linked Open Data 的精神。

Level 1. 無論資料格式，利用開放授權的此種方法開放(在此同時只能稱為 Open Data)。

Level 2. 全部資料皆以機器可讀取性且能結構化的格式發表，(例如當使用 Excel 來紀錄數據並取代紙本的掃描圖片檔。

Level 3. 符合 Level 2，進一步利用非授權資料的格式，例如 JSON 、 CSV...等等來取代 Excel 。

Level 4. 符合上述全部的規範，且利用 W3C 定義出開放資料之標準格式 (例如 SPARQL 、RDF)來描述的資料。

Level 5. 符合上述全部的規範，所提供的資料能讓資料連結至其他人，目的是將資料之間彼此鏈結。



圖 2-2 Linked Open Data 鑑定之標準，5 個 Level

2.5 資源描述框架(Resource Description Framework)

資源描述框架(Resource Description Framework，簡稱 RDF)以及 RDF Schema 是全球網際網路資訊協會(簡稱 W3C)所提出問題中的先導規範，目的是解決資源描述之問題，描述的豐富性及詳細表達出網路資源的結構與內容[17]。在 2004 年 2 月，RDF 已成為 W3C 標準，並允許讓使用者或者開發者能建立階層式的屬性和概念，讓資料呈現出本體(Ontology)的樣貌、離型，主要為用來可供閱讀再使用、資料交換、網路編碼和機器可解析且可讀的 metadata，提供架構的基礎。

RDF 主要用處是使用所呈現出網路資源的狀態與資訊，適合使用在用來表示詮釋得資料，且載明出特定領域之框架(Schema)，方便於資源描述語彙中做

宣告該領域，利用框架中所能表達領域的概念、定義以及階層架構彼此之間的關係，能將從語法的某層級提升至讓該領域的應用中語意的層級，能將機器去理解與處理，而且不會容易喪失原本語意中的通用架構[5]。

RDF 用來建構在 XML 之上，基本的模型包含 Statement、Resource 和 Properties。Resource 有兩種能呈現的型態：「Literal Resource」和「URI Resource」。

例如圖 2-3，存在於一個是以 URI 來描述的 Resource：長框中右方的『t-shirt』；另外有兩種 Properties，分別：『size』和『color』；『size』的 Value 為上述所說的 Literal Resource：『12』；接著是『color』的 Value 則是為上述所說的 URI Resource：『white』。

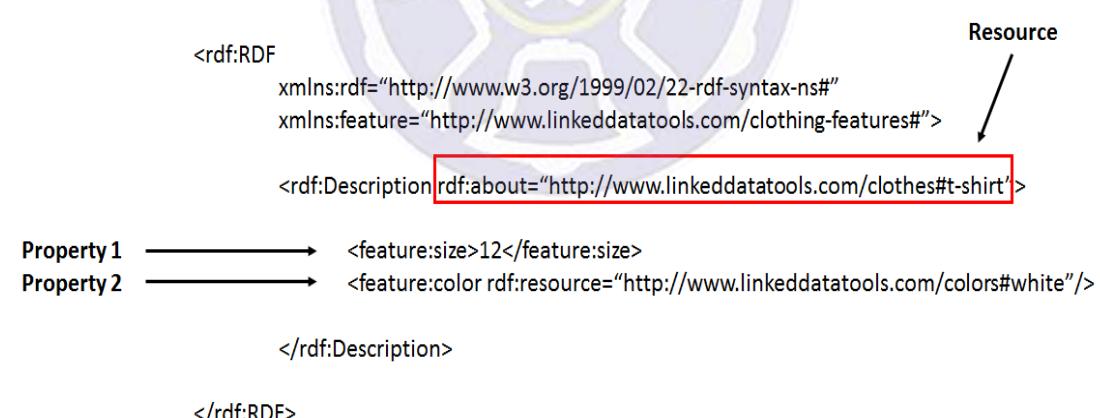


圖 2-3 RDF 範例一

RDF 所能描述物件中基本單位的是 Statement。Statement 是用來描述事情或者事物，其所使用到的三體(triple)，分別代表的是： Object、Subject 以及 Predicate(Property)。這三種所能描述出一句話：「 Subject has a Predicate, which is Object 」，於圖片 2-4 為例。

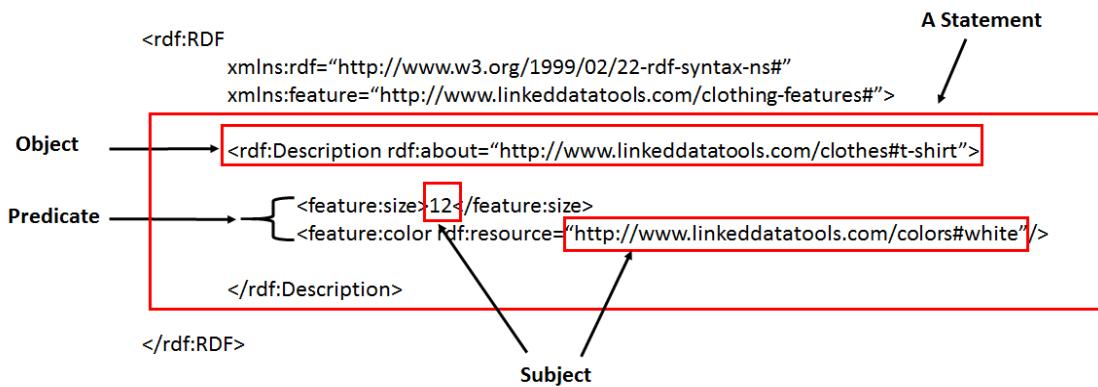


圖 2-4 RDF 範例二

- ◆ Subject : T-shirt (<http://www.linkeddatatools.com/clothes#t-shirt>是嚴格敘述)
- ◆ Predicate(property) : Color(<feature:color>是嚴格敘述)
- ◆ Object : white(<http://www.linkeddatatools.com/colors#white>是嚴格敘述)

可以用白話文來描述：『t-shirt』中有一個屬性為『size』，其中的值為『12』、也有一個屬性為『color』，其中的值為『白色』，若用圖來表示如圖 2-5。



圖 2-5 RDF 關係之概述圖

2.6 SPARQL

SPARQL 是 SPARQL Protocol and RDF Query Language 縮寫而成，目的是用在 RDF 內相關的 Resource 查詢語言。RDF 的查詢語言延伸而來就叫做 SPARQL，例如 SeRQL、RDQL、rdfDB、發展而產生出來，用於提供人們進一步語意網相關的研究與開發。SPARQL 在 2008 年已經成功成為 W3C 推薦的標準。

2.7 Jena API

伴隨著各種領域的資料量大幅地驚人成長，出現各種不同的檔案型態，由於檔案彼此之間互相獨立，這就是造成了檔案的數據相當地龐大，但是卻無法適時且有效利用的窘境。為了處理這類狀況，我們必須針對這類複雜的數據建構出正規化的格式，不但能讓機器可讀，而且可以為了應用程式開發者創造出合適的開發環境，激發出無限的創造力。Jena API 是 HP Labs 利用 Java API 所研發出來的，目的是提供將資料透過本體轉換出的應用程式介面，此用於 OWL 和 RDF 所組合而成的本體(Ontology)，所能提供一系列的查詢、創建、鏈結以及修改的動作。原本混亂的資料在本來中並沒有提供查找、搜尋的功能，另外讓資料轉換出本體完整描述後能使用 SPARQL，代表「 SPARQL Protocol and RDF Query Language 」(SPARQL 協定和 RDF 查詢的語言)能針對指定的資料迅速地完成搜尋。於 2008 年，SPARQL 已正式成為一項 W3C 推薦標準。

下圖 2-6 表示 OWL 、Jena API 、RDF 以及 SPARQL 彼此之間的關係[11] 。

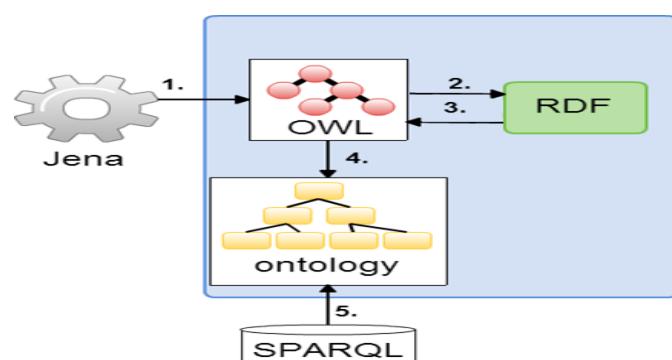


圖 2-6 Jena 應用之流程圖

根據圖片 2-6 所分別於下列五種步驟：

1. 首先使用 Jena API
2. OWL 尋找出合適的 RDF 檔案或者符合的條件
3. RDF 會根據 OWL 所規定的規則
4. 接續上述第三個步驟能產生 Ontology
5. SPARQL 搜尋的語法去 Ontology 中查詢，且拿取需要的資料

2.8 Google API

Google API 是頗盛名的谷歌公司研發出個應用程式介面。其中網際網路的應用軟體具有優良的開放性，為了滿足一般使用者的需求，所提供之豐富且多樣的產品與服務；在此同時也為滿足開發者來開發應用程式，谷歌公司則提供能大量並且能協助廠商來開發出整合的 API 介面[21]。

現今谷歌公司在市面上已開放出 60 種以上的 API，功能範圍非常廣大，每種谷歌都能提供具備可以利用對應的 API 進行呼叫的服務。然而 Google API 時常會與 JSON、XML 或者 JavaScript 等等做結合的應用技術。目的是為了能讓許多人能夠使用谷歌的服務創造新奇的應用，谷歌也提供眾多不需要編寫程式，就單單需要利用簡單的設定可以自然地使用 API 的工具。

2.9 PHPMyAdmin

PHPMyAdmin 是個以 PHP 作為基礎，利用 Web-Base 一種方式架設在網路主機上 MYSQL 的資料庫做為管理工具，管理者必須利用 Web 上介面管理 MySQL 的資料庫。PHPMyAdmin 的一個大優勢是它與其他 PHP 的程式都需在網頁伺服器中來進行執行，但是我們能在無論任何地方利用程式所產生出 HTML 的介面進一步地去遠端，且針對資料庫中進行自由地新增、刪除...之類的動作[22]。

其中主要包括的功能如以下：

- 建立、刪除、修改以及查詢資料庫。
- 可依據使用者的需求進行多國語言的使用者介面切換。
- 可將資料表中複雜的資料任意地匯入或匯出製成各種的格式檔案。
- 資料表的維護方面，基礎的功能如以下六種：
 1. 資料表的檢查
 2. 資料表的修復
 3. 資料表的分析
 4. 資料表的最佳化
 5. 資料表的強迫更新
 6. 建立和檢視出 View 之功能

第三章 系統規劃

3.1 系統整體規劃

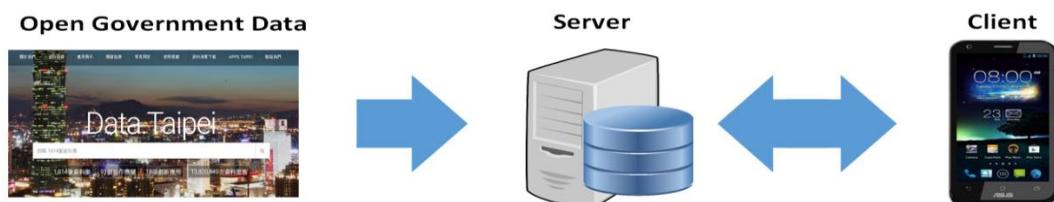


圖 3-1 系統之互動圖

系統互動中主要的對象區分為三種：有使用者(Client)、伺服器(Server)以及政府開放資料網站(Open Government Data)。政府開放資料網站所提供之需求的開放資料作為使用，而伺服器是進行一系列將資料做處理，利用使用者端所提供之情境資訊或不同條件與使用者進一步互動。

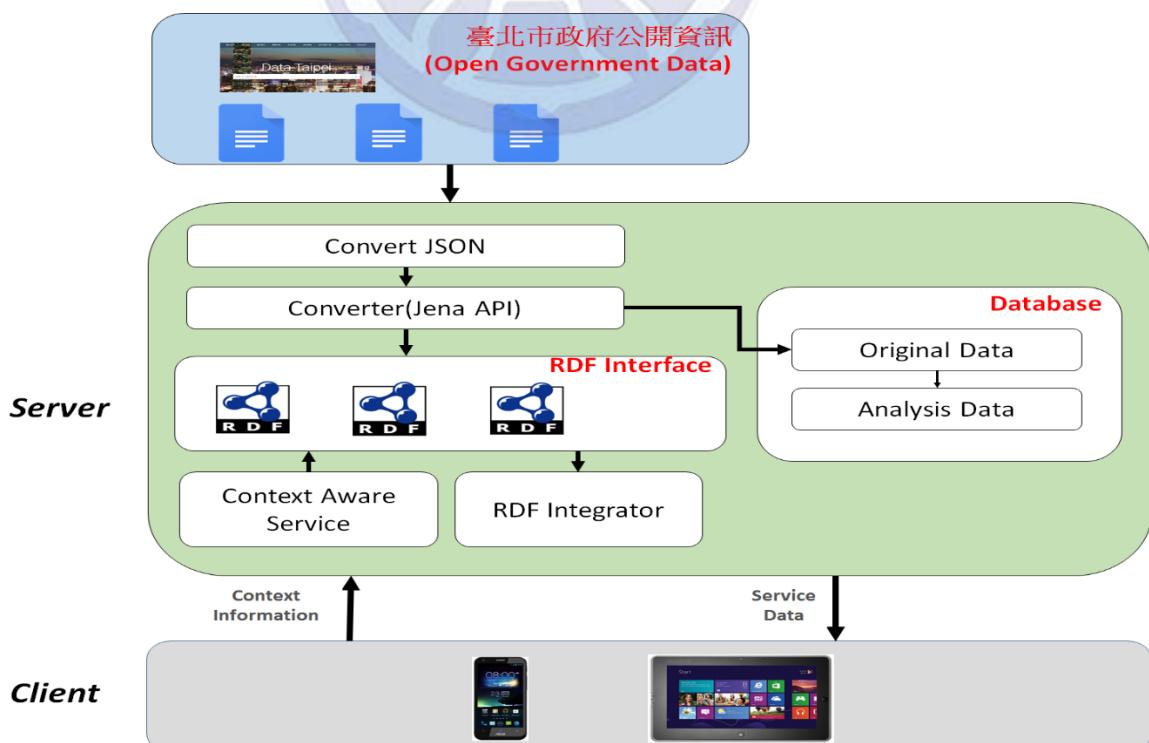


圖 3-2 系統之規劃圖

研究中主要區分為三個部分，第一部分為從臺北市政府公開資料物抓取需要的資料，將不同資料格式一律轉成 JSON 格式，再進行轉換為 RDF 格式來描述，也進行給予能存取的平台；第二部分為臺北市政府目前僅給予當下時間的資料，故不能給予民眾進行查詢歷史紀錄的資料，因此研究中不僅會將資料轉換為 RDF 格式，也同時地將每筆所更新的資料存取至資料庫中，根據使用者需求，進行分析資料且將回傳結果顯示於使用者介面上；第三部分為使用者端，當使用者利用應用程式，將根據目前的時間情境感知資訊(稱 Time Based Context Information)和所處的地理環境情境感知資訊(稱 Location Base Context Information)進行條件分析，而從 Server 端獲得相對應的資料。

臺北市政府目前提供的資料格式混雜，可讀性差，無法輕易存取利用，因此於研究建立一個 RDF Interface，將臺北市政府提供的資料一律轉換為 RDF 資料格式所儲存，並利用使用者需求，讓各式各樣的資料集做整合，建立出 Linked Open Data 來達成統一格式的目的。在統一格式時候，可讀性相對較佳、資料也能輕易地建立鏈結，而藉由將資料彼此之間的鏈結，不同類型的資料能輕易找出鏈結，以利進一步作資料分析且運用。RDF Interface 將提供 SPARQL 的端點，以便讓使用者能利用 SPARQL 做查詢、善加利用這些資料，開發許多加值的應用服務。於研究會談及相關技術和開發利用 Context Aware 應用服務做結合。

研究中將結合『停車場資訊(Park)』、『微笑單車資訊(Ubik)』以及『捷運

資訊 (MRT)』三份不同資料，讓使用者能瀏覽臺北市所有停車場、微笑單車以及捷運的相關即時資訊，而針對捷運到站時間以及車輛空位個數會隨著時間不同給予使用者即時到站時間以及空位個數。於服務中加入情境感知(稱 Context Awareness)的概念，根據使用者的情境給予符合需求的停車場、微笑單車或捷運資訊，讓服務更加貼近使用者生活，本研究所利用到的情境感知主要有 Time-Based 與 Location-Based 兩種模式做為提供服務應用。

3.2 鏈結資料圖(Linked Data Diagram)

臺北市政府現今已經提供很多種開放資料來提供民眾使用，但是由於格式相當地雜亂，使得開發者或者一般使用者困難地做應用，必須先得透過處理加以分析，再經過整理之後才可以自由地進行開發或者運用服務。研究中目的是要解決資料格式雜亂的窘境，於研究所採用的方法為：利用 RDF 來儲存格式，取而代之先前雜亂的資料格式，加強資料能可讀性，於進一步透過整理之後的資料來建立鏈結資料(Linked Data)，增強資料的擴充以及實用性。

本研究中利用 RDF 儲存格式的方式，描述出政府開放資料，且提供存取的平台(例：SPARQL)，方便一般使用者進行存取以及閱讀。臺北市的政府開放資料集現今已經提升到了 1852 筆，相較前幾年的 323 筆，可見每年成長速度相當地迅速，相信未來的一大趨勢是該如何處理相當龐大的數據來善加利用。資料集中與『地理環境』相關約莫有 7 成不同的資料，故在研究中利用地理環境的條件來做研究的核心概念，透過地理環境的條件建構出 Linked Data，嘗試地將符合

以相關條件的不同資料來彼此之間鏈結在一起，建立鏈結資料且進行創新的應用。

以下是擷取臺北市政府局部的開放資料所繪製的鏈結圖，如圖 3-3。

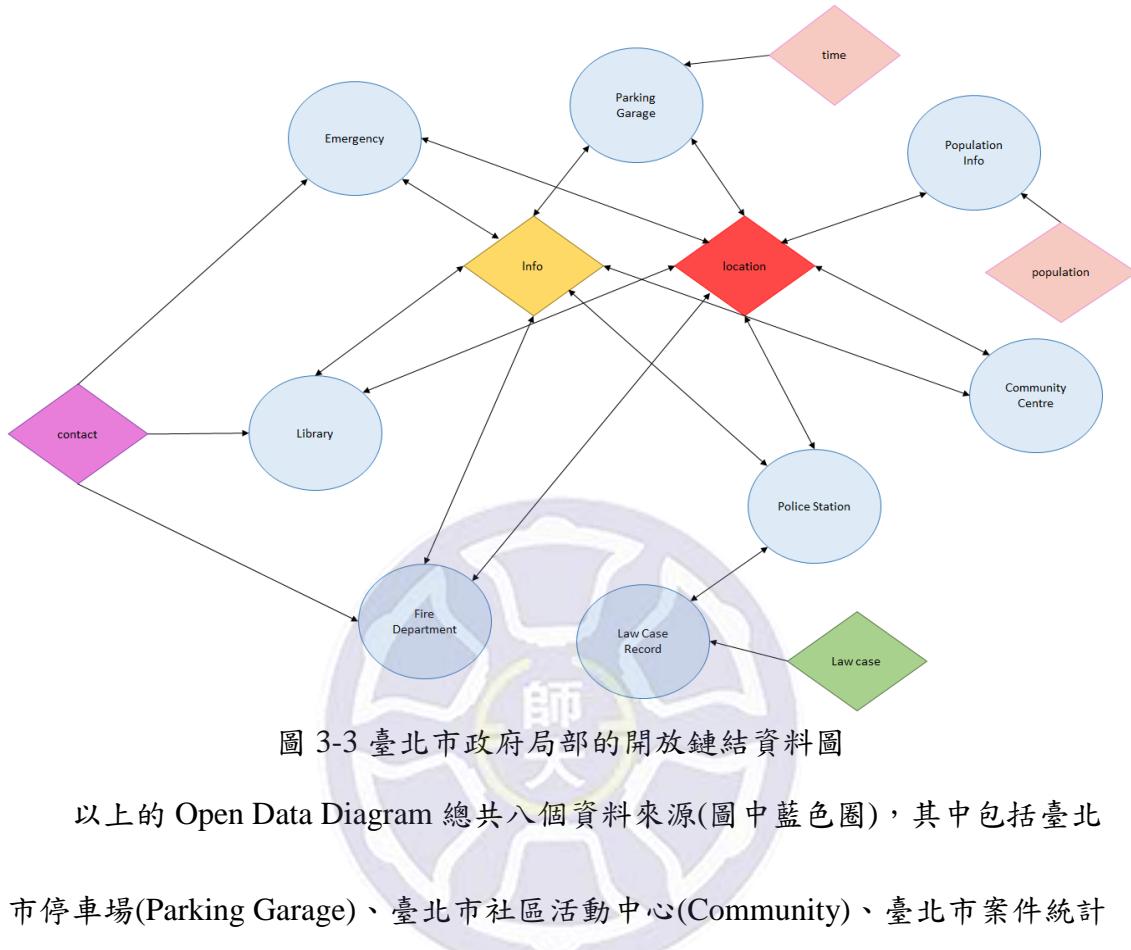


圖 3-3 臺北市政府局部的開放鏈結資料圖

以上的 Open Data Diagram 總共八個資料來源(圖中藍色圈)，其中包括臺北市停車場(Parking Garage)、臺北市社區活動中心(Community)、臺北市案件統計(Law Case Record)、臺北市人口資訊(Population Info)、臺北市警察局(Police Station)、圖書館資訊/Library)、臺北市消防局(Fire Department)、臺北市急救責任醫院(Emergency)，以上只是局部的部分。

研究中打算會利用到三種不同資料的來源且建立 RDF Interface，其中分別：臺北市停車場資訊(於台北市停車管理工程處所提供之資料，屬於交通運輸類別)、臺北市微笑單車資訊(於台北市交通局所提供之資料，屬於交通運輸類別)、臺北市捷運資訊(於台北市公共運輸處所提供之資料，屬於交通運輸類別)，三種不同資料來

源類別皆為交通運輸類，建構出 Linked Open Data Diagram，如圖 3-4。

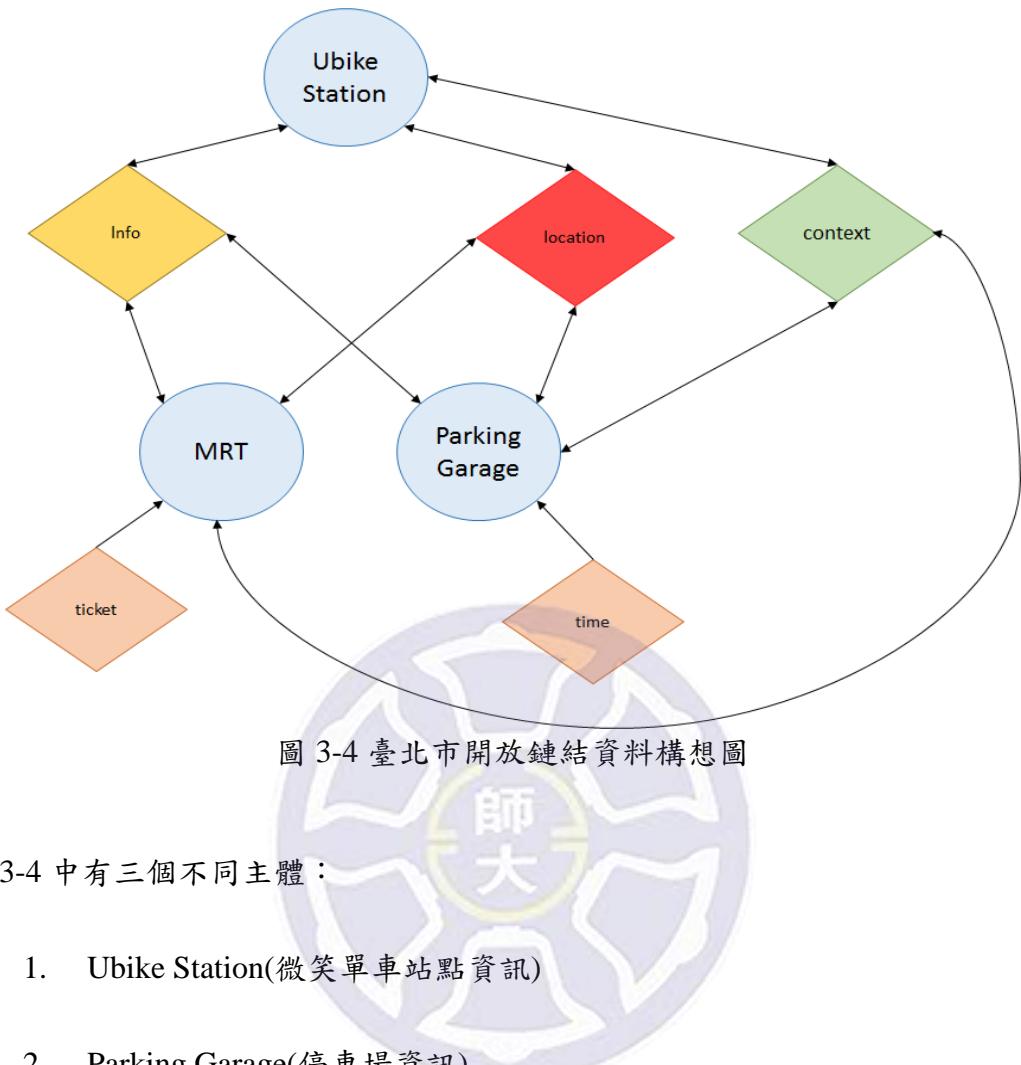


圖 3-4 臺北市開放鏈結資料構想圖

圖 3-4 中有三個不同主體：

1. Ubike Station(微笑單車站點資訊)
2. Parking Garage(停車場資訊)
3. MRT(捷運站點資訊)

分別由四個基本單位構成：

1. Info：使用來記錄基本的資料，例如 Name、Type、ID... 等等。
2. Location：使用來記錄地理環境資料，例如經緯度、地址... 等等。
3. Time：使用來記錄營業時間或者服務時間。
4. Ticket：使用來記錄車票價錢。

最後是 Context 用於情境感知的部分，根據使用者的需求，透過地理環境資料

(Location-Based) 抓取需要的資料，且將需要的資料進一步鏈結的動作，另外製成一份全新的鏈結資料。

以下是實作出各個不同資料集 Dataset 的 Schema

表 3-1 Data Schema - info

`xmlns:info="http://data.taipei.gov.tw/infomation#"`

Attribute	Data Type	Description
<code>info:id</code>	Long	該筆資料唯一的識別碼
<code>info:name</code>	String	該筆資料名字
<code>info:type</code>	String	該筆資料不同屬性。其中會利用到三種不同 type，分別是： parking、ubike 與 mrt

表 3-2 Data Schema - location

`xmlns:loc="http://data.taipei.gov.tw/location#"`

Attribute	Data Type	Description
<code>loc:address</code>	String	地址資訊
<code>loc:area</code>	String	區域資訊。其中項目為臺北市全部的行政區域
<code>loc:latitude</code>	Double	二度分帶的座標(UTM)y 軸座標之資訊
<code>loc:longitude</code>	Double	二度分帶的座標(UTM)x 軸座標之資訊
<code>loc:original_latitude</code>	Double	緯度的資訊
<code>loc:original_longitude</code>	Double	經度的資訊

表 3-3 Data Schema - time

`xmlns:time="http://data.taipei.gov.tw/time#"`

Attribute	Data Type	Description
<code>time:start</code>	String	服務開始的時間
<code>time:end</code>	String	服務結束的時間

表 3-4 Data Schema - parking garage

`xmlns:pa="http://data.taipei.gov.tw/parking#"`

Attribute	Data Type	Description
<code>pa:total_car</code>	Integer	停車場的車位總數
<code>pa:available_car</code>	Integer	停車場剩餘的空位數

表 3-5 Data Schema - ubike station

xmlns:ub="http://data.taipei.gov.tw/ubike#"

Attribute	Data Type	Description
ub:total_ubike	Integer	微笑單車的總車數
ub:available_ubike	Integer	微笑單車剩餘的可用車數

表 3-6 Data Schema - mrt

xmlns:mrt="http://data.taipei.gov.tw/mrt#"

Attribute	Data Type	Description
mrt:start_station	String	捷運發車的起點
mrt:end_station	String	捷運末班的終點
mrt:updatetime	Double	捷運發車的時間

以下為本次研究主要的 Data Schema :

表 3-7 Data Schema - 停車場資訊

停車場資訊

Attribute	Data Type	Description
info:id	Long	該筆停車場的唯一識別
info:name	String	該筆停車場資料的名字
info:type	String	該資料的類型：屬性設定為”parking”
loc:longitude	Double	二度分帶 x 軸座標資訊
loc:latitude	Double	二度分帶 y 軸座標資訊
loc:address	String	站點位置的資訊
loc:area	String	該站所屬區域
pa:total_car	Integer	該站總車位數量
time:start	String	該站開始時間
time:end	String	該站結束時間

表 3-8 Data Schema - 停車場動態格位資訊

停車場動態格位資訊

Attribute	Data Type	Description
info:id	Long	該筆停車場的唯一識別
pa:available_car	Integer	該筆停車場的動態格位

表 3-9 Data Schema - 微笑單車資訊
微笑單車(或 Ubike)資訊

Attribute	Data Type	Description
info:id	Integer	微笑單車唯一的識別碼
info:name	String	微笑單車站點的名稱
info:type	String	資料的類別：屬性設定成”ubike”
loc: longitude	Double	二度分帶 x 軸座標資訊
loc: latitude	Double	二度分帶 y 軸座標資訊
loc:original_latitude	Double	緯度的資訊
loc:original_longitude	Double	經度的資訊
loc:address	String	站點位置的資訊
ub:total_ubike	Integer	該站微笑單車的總數
ub:available_ubike	Integer	該站剩餘單車的數量

表 3-10 Data Schema - 常用微笑單車站點
常用微笑單車資訊

Attribute	Data Type	Description
info:id	String	常用微笑單車站點的唯一識別碼
loc:address	String	常用微笑單車站點的地址
loc:latitude	Double	緯度的資訊
loc:longitude	Double	經度的資訊

表 3-11 Data Schema - 捷運站點
常用微笑單車資訊

Attribute	Data Type	Description
info:id	String	捷運的唯一識別碼
loc:name	String	捷運站點的名稱
loc:type	String	資料的類別：屬性設定成”mrt”
loc:longitude	Double	二度分帶 x 軸座標資訊
loc:latitude	Double	二度分帶 y 軸座標資訊
loc:address	String	站點位置的資訊
mrt:start_station	String	該站捷運發車的起點
mrt:end_station	String	該站捷運末班的終點
mrt:updatetime	Double	該站捷運發車的時間

因為政府所提供的開放資料，停車場動態格位與停車場資訊兩種分別有彼此不同 RDF 的 DataSet，但是彼此之間仍然能利用唯一識別的 ID 的資訊進一步做資料的溝通。在實作 Query 時會利用『縮減搜尋範圍(Reduce Size)』的技術來進行，目的是減少不需要的時間以及搜尋。其中微笑單車的部分全部資訊都包括動態車輛皆於同一份的開放資料裡存在。

3.3 情境感知(Context Awareness)

使用「時間情境感知(稱為 Time-Based)」和「地理環境情境感知(稱為 Location-Based)」來偵測於環境資訊，用這兩種做為基礎，提供出能符合的情境規則資料於使用者做使用。研究中使用到三種資訊，分別為『停車場資訊』、『捷運資訊』以及『微笑單車資訊』開發出情境感知行動應用。利用使用者所使用時的應用程式獲得的時間資訊以及地理環境位置，此獲得的資訊會立即回傳至 Server 端，進一步做資料的整合與分析，產生出一份最清楚且貼近使用者的資料給使用者做參考。在此舉例一個貼近使用者的例子：使用者的位置目前於「台北市文山區汀洲路四段 88 號」，若想要搜尋鄰近的交通工具，在此提供三種可做選擇，分別是停車場資訊、捷運資訊或者微笑單車站點，接著只要觸碰一個按鍵，使用者需要的服務就會立即將鄰近且符合的相關資訊依依呈現於行動裝置的介面上，能提供使用者做不同選擇中的參考，另外使用者也能自由地輸入需要的範圍，利用輸入所需要的範圍再加上目前的地理資訊做結合，系統就會立即提供使用者需要的結果。

3.4 縮減範圍搜尋(Reduce Size)

本研究中利用三個不同的資料集，分別是：『停車場動態格位資訊(依據時間來動態更新停車格位)』、『停車場資訊(包括時間資訊、地理位置以及收費資訊)』、『微笑單車資訊(動態車輛資訊以及地理位置)』。其中專注於停車場動態格位資訊以及停車場資訊進一步做一連串的處理，當使用者欲用應用程式時，不會因每次針對最大範圍資料來做資料的鏈結動作，同時拿取局部資料做使用，其實是依據先前搜尋產生記錄的資料，接著做詳細的分析，若有偵測到地理情境資訊和使用者自行設定皆符合規則時，此時只需要對先前進行搜尋的局部範圍中得到的結果，進一步做資料的鏈結動作，接著拿取與使用者自行設定的資料來看是否符合，目的是縮小範圍內的分析做處理能有效地減少搜尋或者鏈結資料所花的時間，而且每次使用時所留下的資料，日積月累地數量是相當龐大且可觀，利用這些資料能進行許多的應用與分析，例如：自動化推薦、使用者的習慣...等等。

下表比較使用 Reduce Size 以及未使用 Reduce Size 做搜尋之時間差異。

表 3-12 使用 Reduce Size 的執行時間差異(第一次執行)

First use.定點搜尋周圍 2 公里

搜尋花費時間(milliseconds)

使用 Reduce Size	1620
未使用 Reduce Size	1605

表 3-13 使用 Reduce Size 的執行時間差異(第二次執行)

Second use.定點搜尋周圍 1.5 公里

搜尋花費時間(milliseconds)

使用 Reduce Size	1585
未使用 Reduce Size	1650

表 3-14 使用 Reduce Size 的執行時間差異(第三次執行)

Third use.定點搜尋周圍 1 公里

搜尋花費時間(milliseconds)

使用 Reduce Size	1410
未使用 Reduce Size	1499

表 3-15 使用 Reduce Size 的執行時間差異(第四次執行)

Forth use.定點搜尋周圍 0.5 公里

搜尋花費時間(milliseconds)

使用 Reduce Size	1225
未使用 Reduce Size	1452

表 3-16 使用 Reduce Size 的執行時間差異(第五次執行)

Forth use.定點搜尋周圍 0.3 公里

搜尋花費時間(milliseconds)

使用 Reduce Size	1185
未使用 Reduce Size	1422

3.5 微笑單車數據分析與推薦

臺北市政府提供微笑單車的資訊，其中包括「動態車輛資訊」以及「地理環境資訊」。「微笑單車的動態車輛資訊」是當中最具有價值的資料，遺珠之憾的是現今卻無法於特定時間點中獲知微笑單車的車輛資訊。舉例來說：若使用者目前剛於公司開完重要會議時，欲搜尋於公司附近且無需等候車位的微笑單車的站點，但此時距離開完會議需一段時間，但不能預先獲知鄰近微笑單車站點的時間點能看出還有多少車輛的機率，若是從已租借到車輛的使用者觀點著手，也不能知道鄰近微笑單車站點此時是否有空位以及該站的成功歸還機率有多少。

系統建置出一個從『真實數據』去做分析且預測推薦機制，每隔時間會利用臺北市政府開放資料去做更新，同時存取至資料庫去做分析，來呈現微笑單車可用率以及使用率的模型，能使使用者利用分析的資料獲知微笑單車的前後時間點中使用狀況，以下這四筆公司是計算微笑單車的可用率(剩下車輛的比率)以及使用率(租借的比率)。

$$\text{Availability Rate} = \frac{\text{Available ubike}}{\text{Total ubike}}$$

$$\text{Utility Rate} = \frac{(\text{Total ubike} - \text{Available ubike})}{\text{Total ubike}}$$

$$\text{Average Availability Rate} = \frac{\text{AvailabilityRate1} + \dots + \text{AvailabilityRate15}}{15}$$

$$\text{Average Utility Rate} = \frac{\text{UtilityRate1} + \text{UtilityRate2} + \dots + \text{UtilityRate15}}{15}$$

利用前一個禮拜採用的數據分析當時的時間前後三個小時之間以及目前的

時間前三個小時，來計算微笑單車可用率和使用率，如圖 3-5 之微笑單車時間軸的分析圖將表 3-17、3-18 以及 3-19 做呈現。

表 3-17 上禮拜前 3 個小時之微笑單車的使用率以及可用率

過去時間為 2016/05/4 13:14:30

	10:14	11:08	11:38	12:38	12:44	13:08
使用率	76%	70%	76%	69.5%	67%	65.5%
可用率	24%	30%	24%	30.5%	33%	34.5%

表 3-18 上禮拜後 3 個小時之微笑單車的使用率以及可用率

過去時間為 2016/05/4 13:14:30

	13:38	14:08	15:17	15:47	16:17	16:47
使用率	65%	65.5%	67%	69%	68%	65%
可用率	35%	34.5%	33%	31%	32%	35%

表 3-19 目前時間前 3 個小時之微笑單車使用率以及可用率

目前時間為 2016/05/11 13:14:30

	10:14	11:08	11:38	12:38	12:44	13:08
使用率	65%	65%	66%	65%	64.5%	64%
可用率	35%	35%	34%	35%	35.5%	36%

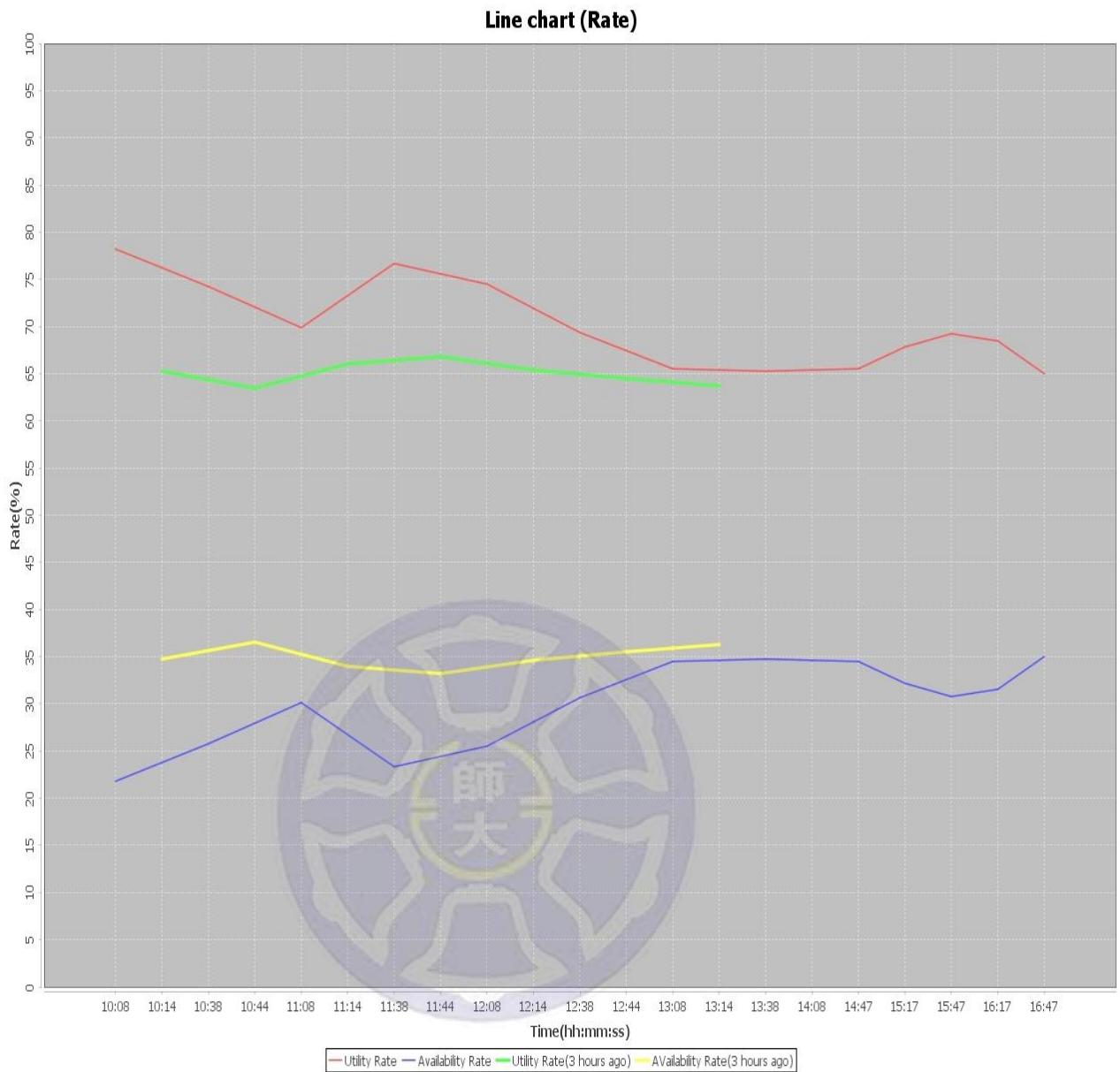


圖 3-5 微笑單車時間軸分析

在圖表中可以觀察發現當下時間之後三個小時微笑單車的使用情況，使用率幾乎皆大於 65% 的情況，代表是租借者有一定機率至該站點無法租借微笑單車；相對地可用率幾乎都在 35% 以內，代表是正在使用微笑單車的民眾至該站點有較大機率能歸還微笑單車。由真實的數據進一步分析能給予使用者最客觀且符合該站點使用情況的數據，讓使用者能根據數據判斷說是否要前往該微笑單車站點進行租借或歸還的動作，視覺化的圖表也較易讓使用者做參考基準。

3.6 捷運不同格式整合運用

臺北市政府開放資料平台目前有著大量的資料集，在如此龐大資料集中找出有用的資料在加以應用，本研究中在臺北市政府開放資料平台搜尋有關於捷運的資料集，其中找出有四種資料集可以在捷運上做搭配使用，分別是「臺北市捷運系統票價」、「臺北市捷運車站無障礙電梯位置」、「臺北捷運列車到站站名」和「臺北捷運高運量系統及文湖線相鄰兩站之行駛之間、停靠時間」，透過彼此的互相搭配，可以讓捷運功能做的更加完善，也可以透過不同資料來進行串聯使用，來達到想要的結果，但是「臺北市捷運系統票價」、「臺北市捷運車站無障礙電梯位置」、「臺北捷運列車到站站名」和「臺北捷運高運量系統及文湖線相鄰兩站之行駛之間、停靠時間」四種資料中分別都是不同的格式，不同格式對想製作加值的服務會相當的混亂，使得一般使用者或開發者難以應用，必須做善加的處理才可應用，如圖 3-6。

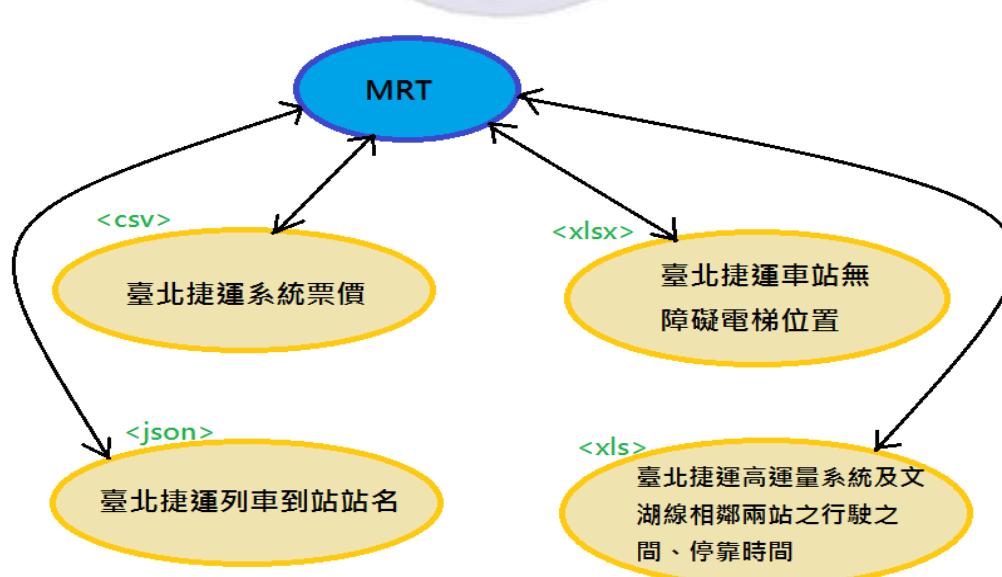


圖 3-6 捷運不同格式之應用

由於臺北市政府公開資料有著各式各樣的格式，包含 csv、xls、xml、txt... 等，其格式混雜且多種，讓開發者或一般民眾難以去做使用，甚至需要的資料為不同檔案類型，本研究為了解決雜亂的格式去做處理，利用捷運的構想運用於臺北市政府公開資料，在捷運眾多資料中採用的四筆資料去做加值服務的應用，恰巧需要的資料皆為不同格式（csv、xls、xlsx 以及 json），於是利用 php 的程式語法來統一格式，讓四筆資料格式中 csv、xls、xlsx 皆一律轉換為 json，並進一步產生 rdf 格式進行之後的應用，目的是解決資源描述，豐富的描述及表達網路資源的詳細內容與結構讓資料具有本體的離型，讓資料交換、網路編碼、可供閱讀再使用、機器可讀可解析的 metadata，提供基礎的架構，進而讓開發的應用有著更好呈現的效果，創造出更多更有價值的加值服務應用，如圖

3-7。

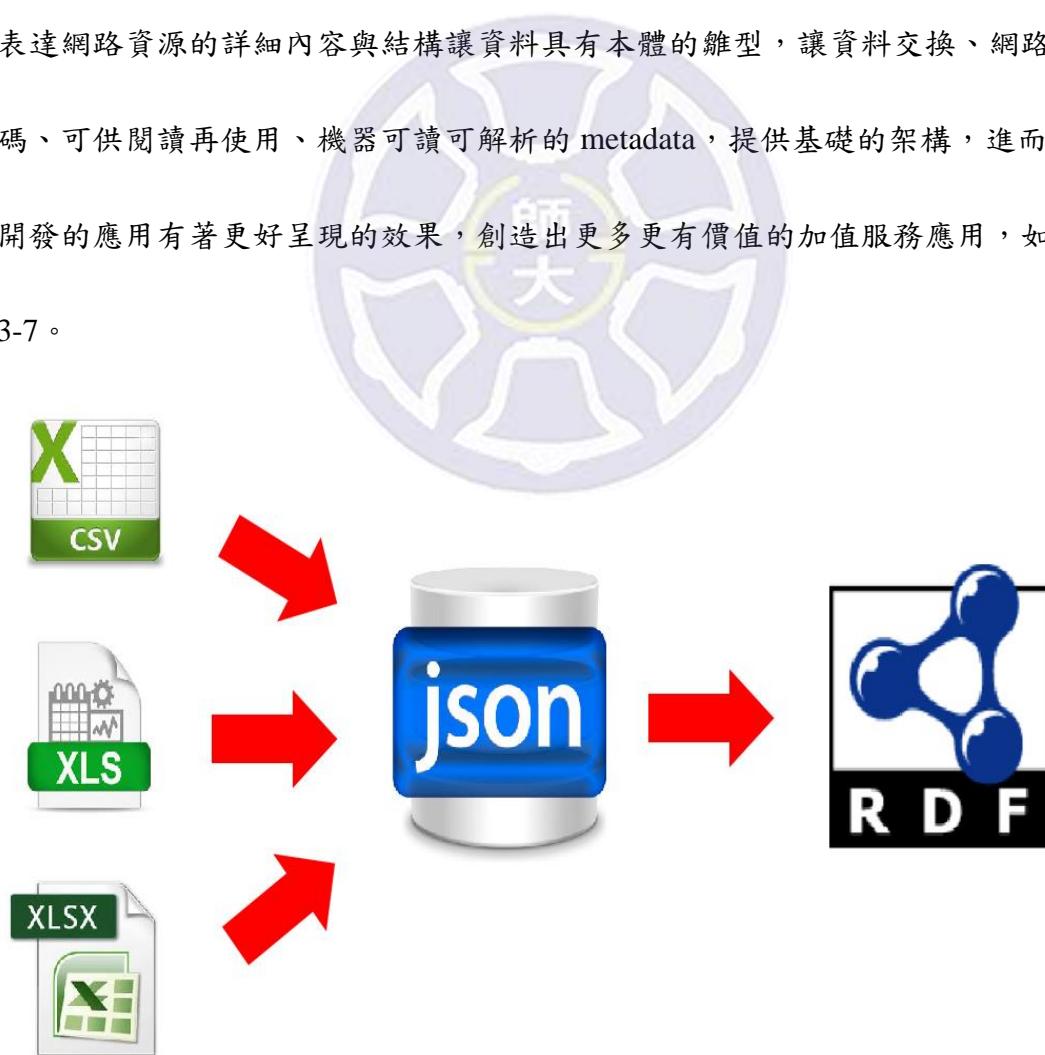


圖 3-7 不同格式統一轉換圖

3.7 捷運列車到站時間分析

臺北市政府提供捷運公開資訊，其中包括「臺北捷運列車到站站名」、「臺北捷運高運量系統及文湖線相鄰兩站間之行駛時間、停靠站時間」是其中用來計算捷運到站計算時間的應用，目前臺北市政府所提供的開放資料中沒有明確的給使用者提供預測捷運到站的時間點，只有提供此時目前行駛列車中的時間，故我們利用這兩份政府公開資料來推算說此時位於捷運上的站名，經過計算推得預計多久會到達目前位於此站的時間，方便使用者利用此資訊可以得知捷運列車中何時會到達來把握時間。

ArriveTime：列車到達此站時間。

UpdateTime：行駛中列車時間。

EstimateTime：預估多久時間會到達。

NowTime：目前使用者此刻時間。

StopTimeA : A 站停靠時間。

StationA_to_ StationB : A 站到 B 站行駛時間。

計算如下：

$$\text{ArriveTime} = \text{UpdateTime} + \text{StopTimeA} + \text{StationA_to_ StationB} + \dots$$

$$\text{EstimateTime} = \text{ArriveTime} - \text{NowTime}$$

3.8 臺北一日遊規劃行程之情境應用

臺北市是繁榮且便利的都市，在這繁榮的都市中必然有著車流量龐大的問題，因為人口過於稠密，在臺北茫茫人海中，尤其是在上下班時間路上不管機車或者汽車總是水洩不通，於是目前上班族或者學生會選擇大眾交通工具，例如捷運、Ubike、公車.....等等，為了提供更好的生活品質以及解決車位一票難求問題，所以我們營造出一個情境應用，讓使用者可以依據當時的情形並透過APP解決到達目的地的過程，其中在路途的過程中可以透過不同交通工具的轉換得知如何搭乘避開尖峰時期所帶來的車潮以及獲知即時的資訊，以便讓使用者大大節省時間且避免白跑一趟之情況。

本研究中以臺北市為一日旅遊的規劃行程為模擬，建構出情境應用的例子，首先以兩種例子做為出發點：第一種是以使用者自行安排為觀點，利用使用者自己絞盡腦汁構想的行程進行規劃，安排一套適合全家可以玩樂整天的行程，首先以地點做為優先構想，再來利用交通工具去融入原先規劃的行程，以交通工具做為搭乘方式去各個景點玩樂，接著以 Open Data 的概念利用研究所開發的功能進一步做結合應用，讓使用者可以馬上獲得即時資訊去進行一天的旅遊；第二種是以部落客熱門文章為觀點，利用網路上熱心民眾分享的文章去做規劃行程的動作，避免行程中出現瑕疵，首先從推薦行程進行規劃的藍圖，再來配合客製化的交通工具搭配出屬於自己的方式，最後結合 Open Data 所開發的功能，利用即時資訊考慮使用者目前需求做為旅遊的主軸，如圖 3-8。

	自訂行程	配合交通工具	結合Open Data
以使用者自行安排為觀點	 <p>↓</p> <p>花博公園 ↓ 臨安泰古厝 ↓ 淡水老街 ↓ 漁人碼頭</p>	<p>停車場 ↓ 花博公園 ↓ Ubike ↓ 臨安泰古厝 ↓ 捷運 ↓ 淡水老街 ↓ Ubike ↓ 漁人碼頭</p>	<p>停車場總數和剩餘車位 ↓ 花博公園 ↓ 微笑單車總數和剩餘車位資訊 ↓ 臨安泰古厝 ↓ 捷運到站時間、票價以及無障礙電梯資訊 ↓ 淡水老街 ↓ 微笑單車總數和剩餘車位資訊 ↓ 漁人碼頭</p>
以部落格熱門文章為觀點	 <p>↓</p> <p>龍山寺 ↓ 東區餐廳 Mee's Cafe ↓ 松山文創園區 ↓ 下午茶 Akuma caca Cafe ↓ VIVO 光癮時尚餐飲</p>	<p>龍山寺 ↓ 捷運 ↓ 東區餐廳 Mee's Cafe ↓ 捷運 ↓ 松山文創園區 ↓ Ubike ↓ 下午茶 Akuma caca Cafe ↓ Ubike ↓ VIVO 光癮時尚餐飲</p>	<p>龍山寺 ↓ 捷運到站時間、票價以及無障礙電梯資訊 ↓ 東區餐廳 Mee's Cafe ↓ 捷運到站時間、票價以及無障礙電梯資訊 ↓ 松山文創園區 ↓ 微笑單車總數和剩餘車位資訊 ↓ 下午茶 Akuma caca Cafe ↓ 捷運到站時間、票價以及無障礙電梯資訊 ↓ VIVO 光癮時尚餐飲</p>

圖 3-8 不同觀點之情境應用圖

以使用者自行安排的觀點為例，當一家人出發到台北遊玩時候，假設我們是台北一日遊之行程，因為網際網路的發達，首先我們可以利用各個旅遊景點的官方網站獲得玩樂的情報，其中包括交通資訊以及圖片中有關於行、育、樂方面，方便我們可以對於眾多的旅遊指南來事先做功課，再來先利用手機的行事曆來紀錄行程的安排與時間，假設安排台北一日遊之行程，早上 10 點到下午 3 點到花博公園欣賞不定時的展覽和一起吃頓美味午餐，下午 3 點到下午 5 點到林安泰古厝觀賞古蹟之美，下午 5 點到晚上 8 點在淡水逛著淡水老街買小吃以及著名的淡水阿給，接著最後晚上 8 點到 10 在漁人碼頭走著繽紛的吊橋順便欣賞美麗風景，其中包含交通時間，重要的是需要思考如何搭乘何種交通工具到達遊玩的目的地，緊密的思考可以省下不少時間且更可以愉快的過一天，如

圖 3-9。



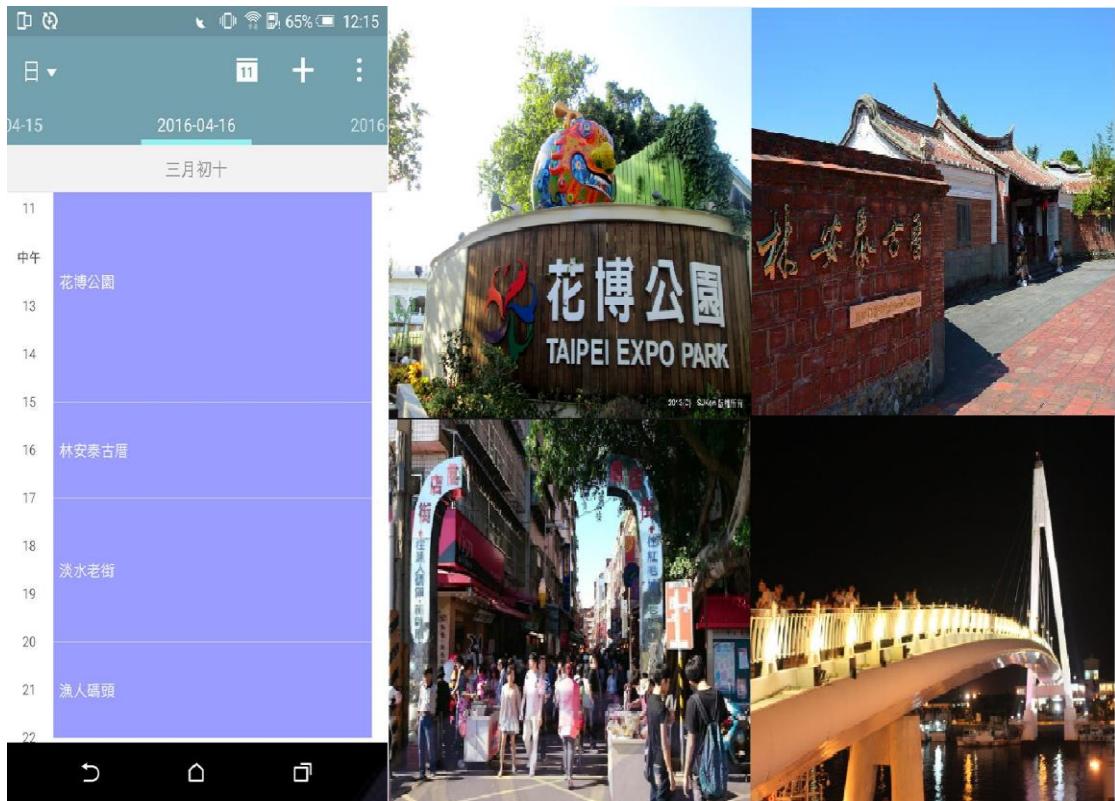


圖 3-9 行事曆記錄圖

在此我們在行事曆中的行程加入交通工具來製作圖示，模擬一些行進到達的方式，其中一開始我們有兩種方式，第一種是若我們是以汽車為代步起點出發，我們可以搜尋附近距離花博公園鄰近的停車場來停放汽車，之後以 ubike 方式遊玩花園公園，第二種是若我們是以捷運抵達，以搜尋附近鄰近的捷運站進行搭乘，再到達紅線圓山站即可到達，之後以 ubike 方式遊玩花園公園，臨安泰古厝位於花園公園附近，便可利用 ubike 到達此處，接著搭乘捷運紅線去淡水老街買名產逛街，其中捷運站附近有 ubike 可以租借，騎乘 ubike 逛淡水周圍的河堤，沿路可以欣賞街頭藝人的表演，另外漁人碼頭騎乘 ubike 可從此處到達，晚上欣賞美景吹著海風別有一番風味，如圖 3-10。

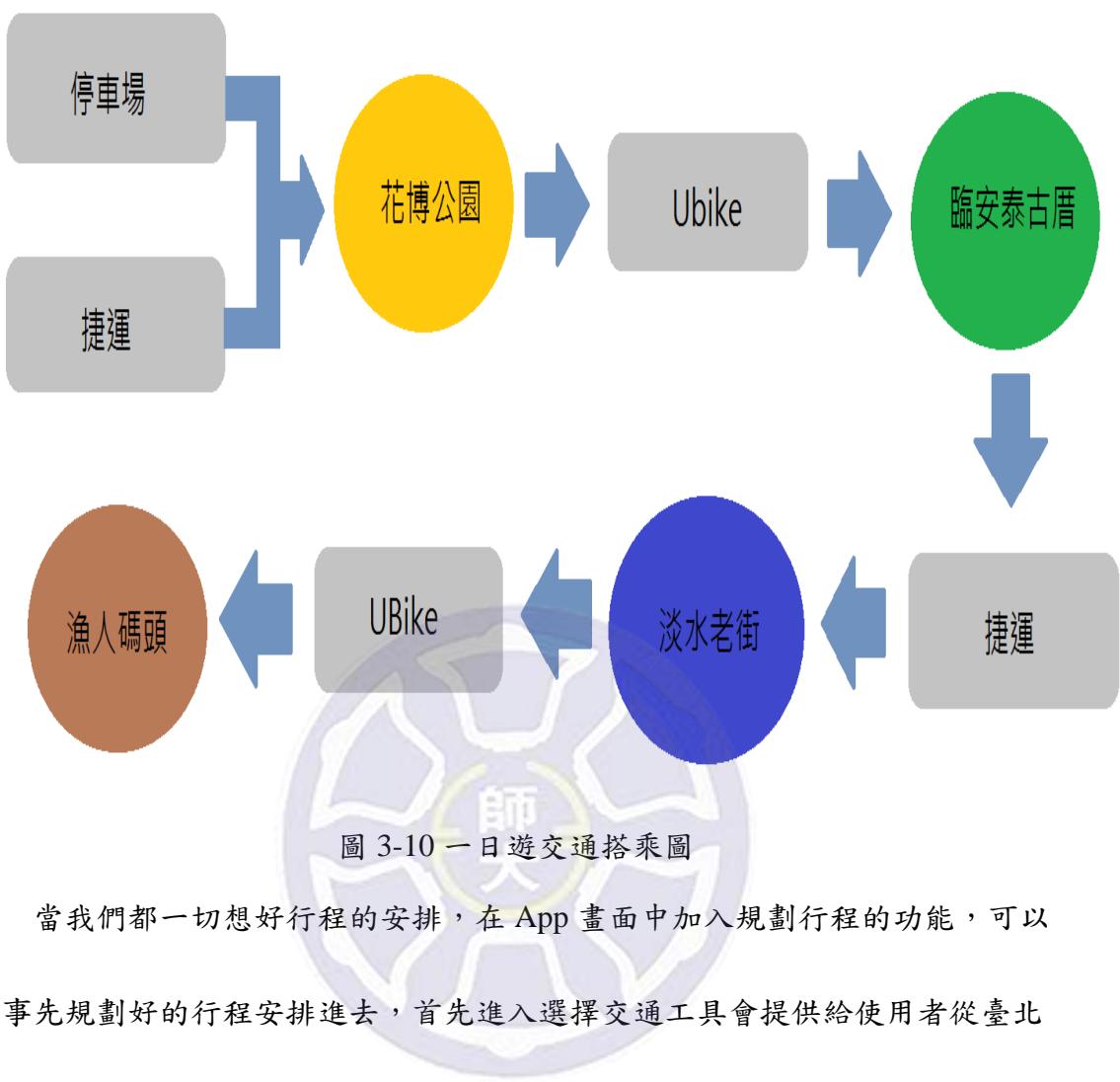


圖 3-10 一日遊交通搭乘圖

當我們都一切想好行程的安排，在App畫面中加入規劃行程的功能，可以把事先規劃好的行程安排進去，首先進入選擇交通工具會提供給使用者從臺北市政府公開資料所提供的各項交通工具給使用者選擇，其中包括：停車場、微笑單車以及捷運站點，依據想到達的地點讓使用者自行輸入，每當輸入及選擇完成一筆資料後會立即顯示於畫面中，若輸入多筆資料，顯示會提供地址、地點、搭乘何種交通工具到達目的地等相關資訊，可以更清楚明白行程規劃的流程。

另外提供顯示在地圖的畫面，當使用者按檢視地圖的按鈕，會把使用者所選擇的交通工具完全呈現且依據優先選擇來畫線，讓使用者能更直覺的檢視交

交通工具的動向，若不滿意的話，可以再次規劃更完美的搭乘站點，規劃方面能更靈活的使用，且讓使用者能有不一樣的操作感受；若滿意的話，按下完成按鈕紀錄規劃行程的資訊，可以再次參考使用，如圖 3-11。



圖 3-11 一日遊規劃行程圖 - 範例一

在台北的一日遊中，其中最怕的就是交通阻塞、車位租不到和錯過列車，研究中開發出的 APP 可以尋找各項交通資訊，其中最常見的交通工具如：捷運、ubike、停車位，透過搭乘交通工具的混合使用，讓使用者可以省下大大的時間，在旅行的過程中利用此 APP 使用的資訊可以幫助遊玩時交通工具的轉換，讓到達目的地更加順暢無阻，例如開車時，可以搜尋鄰近的停車場位置進行停車的動作，免除找車位的麻煩；租借 ubike 時可以搜尋目前位置利用自己設置的範圍去搜尋範圍內的站點，而觀看租借的情況來考慮到哪個站點租借或者歸還；搭乘捷運時，使用者可以觀看目前位於哪條捷運線路的時間且查詢多

久會到達以及票價查詢起點到達目的地的價錢和捷運站點轉站的資訊，透過位置去考慮到哪轉車，另外提供無障礙電梯資訊，幫助行動不便的人可以清楚電梯位置去搭乘，如此透過交通工具的轉換可以充實玩樂一天又可避免在美好的假期為了塞車而困擾，甚至失去興致，如圖 3-12。



圖 3-12 交通工具應用顯示圖 - 範例一

若我們不是以自行所構想的規畫行程的話，是以部落格熱門文章為例，由於網際網路的蓬勃發展，利用手機上網去輸入想要尋找的關鍵字，例如：臺北一日遊，如圖 3-13，就會顯示在網路上看到許多的熱心旅遊客在部落客分享推薦文章，在此舉例一個利用網路上推薦熱門文章來實作臺北一日遊之情境應用，部落客提供許許多多的旅遊圖片、交通方式、景點特色…等的資訊，讓想去旅遊的人可以有很多種選擇，其中本研究以此篇部落客為例，利用捷運來遊玩臺北一整天，列舉四個路線包括路線一捷運松山新店線站、路線二捷運板南線、路線三捷運中和新蘆線以及路線四捷運淡水信義線，而且詳細述說利用同條路線的捷運可以邊停邊玩邊看每個景點，避免花時間於轉車方面，當中我們採用路線二：捷運板南線來規劃此行程，如圖 3-14。

[台北一日遊行程安排規劃，捷運7條路線旅遊景點美食推薦！@ 小環妞 ...](#)
[wkitty.pixnet.net/blog/post/33207982-台北一日遊行程安排規劃，捷運7條路線旅遊 ▼](#)
搭捷運遊[台北一日遊](#)7條路線吃喝玩樂美食推薦無論是台北人還是中南部人肯定都很需要這樣子[台北一日遊](#)行程資訊也常常有讀者問我如果要去[台北一日遊](#)或二日遊 ...
【[台北旅遊景點](#)】推薦行程規劃 ... [台北象山站](#) 搭配象山美景的 ...

[台北一日遊自由行建議行程推薦分享@ 穀子自由行-台北好好玩:: 痞客邦 ...](#)
[travel2tp.pixnet.net/blog/post/27686192-台北一日遊自由行建議行程推薦分享 ▼](#)
我編列了幾條大眾交通捷運台北市區一日遊路線圖，時間為10-15小時。 ... 黃金博物園區)-九份老街-十分(放天燈)鹿福瀑布-平溪-菁桐車站-深坑豆腐老街[台北一日遊](#) ...

[台北一日遊15個好玩捷運景點+必去逛街地圖推薦圖文寫真報導\(交通導 ...](#)
[beei.pixnet.net/blog/post/36474519-台北一日遊15個好玩捷運景點+必去逛街地圖 ▼](#)
台北 15 個熱門捷運旅行開始Taipei 第一站：紅樓創意市集(The Red House Market for Artists & Designers) 推薦指數: A+++.

[台北捷運一日遊行程規劃 - 隨意窩日誌 - 隨意窩Xuite](#)
[blog.xuite.net/.../349714145-台北捷運一日遊行程規劃，精選十條台北捷運路線+~+... ▼](#)
台北 捷運有五條路線，路網分布大[台北](#)，無論是上班族、學生族、觀光客，在[台北](#) ... 201510191600[台北](#)捷運一日遊行程規劃，精選十條[台北](#)捷運路線~ 旅遊景點、美食 ...

[【台北景點】台北一日遊/台北兩天一夜/台北好吃好玩景點\(觀光客必去 ...](#)
[https://mihia.tw/taipeinote/ ▼](#)
2015年12月10日 - 我覺得[台北](#)真的是個非常棒又美的地方，所以幫大家整理了好吃好玩的景點，還附上了地圖，希望你們可以順利安排好兩天一夜的旅行喔！

圖 3-13 搜尋臺北一日遊資訊

路線一 捷運松山新店線

公館寶藏巖藝術村(公館站)→ 尖蚪探索食堂→ 公館小吃→ 台灣博物館南門分館(中正紀念堂)→ 西門町(西門站)



路線二 捷運板南線

龍山寺參拜(龍山寺站)→ 東區餐廳Mee's Café(國父紀念館站)→ 松山文創園區看展逛街(市政府站)
→ Akuma caca Café下午茶→ VIVO光ழ�時尚餐廳(忠孝敦化站)



路線三 捷運中和新蘆線

1914文化創意園區(忠孝新生站)→ 幾米餐廳→ 華山大草皮野餐→ 行天宮參拜(行天宮站)→ 三重水漾公園(三重站)



路線四 捷運淡水信義線

淡水文化園區-穀牌穀倉(淡水站)→ 淡水老街→ 搭船or騎單車前往漁人碼頭→ 漁人碼頭情人橋&愛之船飯店
→ 世界巧克力公園→ 漁人碼頭夕陽→ 淡水老街晚餐or巧克力主題餐廳



參考自**bobowin**旅行攝影生活：
Blog.xuite.net/bobowin/me/349714145

圖 3-14 捷運一日遊行程資訊圖

當我們從部落客推薦行程文章中選擇理想的臺北市一日遊時，則可以設想行程中更細部的情況，則是由交通工具做為考量，雖然部落客文章中有推薦大致上的搭乘方法，但其可以在加點自己所想法，讓行程的過程中更加完美，於是選擇的行程中首先是前往龍山寺廟宇拜拜祈求萬事平安，接著搭乘龍山寺站前往東區餐廳 Mee's Café 享受美好午餐，吃完午餐坐捷運到市政府站觀賞松山文創園區的展覽且租台臺北市政府所提供的 Ubike 來騎車順便逛街一整個下午，逛街逛到疲倦了，可以到市政府附近著名的 Akuma caca Café 下午茶，搭配入口即化的甜點，讓下午的美好行程增添些美味，晚餐搭捷運到忠孝敦化站去 VIVO 光癮時尚餐飲吃道地的西班牙料理，其環境優美、氣氛絕佳以及不可或缺的美味菜色，讓人垂涎三尺，不管帶情人或家人來是個絕佳地方。

再來就是研究中所提供的規畫行程功能，可以把事先做好的行程功課，輸入到 App 中，運用從臺北市政府公開資料所提供的各項交通工具給使用者做選擇，其中包括：停車場、微笑單車以及捷運站點，依據想到達的地點讓使用者自行輸入，每當輸入及選擇完成一筆資料後會立即顯示於畫面中，選擇時系統會根據地理情境資訊優先選擇離使用者最近的停車場、微笑單車以及捷運站點資訊，若輸入多筆資料，系統會根據所輸入資料的最末筆資料的地理位置來顯示離此位置最近的停車場、微笑單車以及捷運站點資訊，顯示的畫面提供給使用者地址、地點、搭乘何種交通工具到達目的地等相關資訊，可以更清楚明白行程規劃的流程，如圖 3-15。



圖 3-15 一日遊規劃行程圖 - 範例二

最後一切就緒就可進行行動裝置中 App 功能應用，我們可以利用各項功能

查詢所有停車場、Ubike 以及 MRT 資訊，首先以此次規劃行程中 MRT 的功能

中瀏覽所有捷運行駛列車，偵測目前使用者位置推薦最近捷運站點以及捷運行

駛列車到站資訊可以清楚列車的動向，也提供捷運無障礙資訊，讓行動不便的

民眾進行查詢動作，若不清楚捷運票價價錢，能提供依據我們所選擇的列車進

行該站的票價價錢做為查詢，另外也可以觀看轉車的資訊，每站捷運列車所在

的路線會依據顏色不同而行駛不同地方，臺北市捷運以顏色最為區分，例如：

淡水線、新店線...等，故在地圖上有配合不同路線顏色而有標誌捷運站顏色，避

免台北人生地不熟的民眾不易上錯車且可以清楚知道我們該轉什麼捷運路線。

接著此次規劃行程中 Ubike 中瀏覽所有微笑單車站點資訊，可以根據我們所選

擇任何一個微笑單車站點獲得 Ubike 的總數量以及剩餘數，而根據所選擇該微

笑單車站點的數據分析圖，讓我們可以得知該站點的曲線動向以及獲得最佳租

借和歸還時段，避免租車和還車的尖峰時段，導致租借車輛或者歸還車位一位

難求的窘境，如圖 3-16。



圖 3-16 交通工具應用顯示圖 - 範例二

3.9 使用案例(Use Case Diagram)

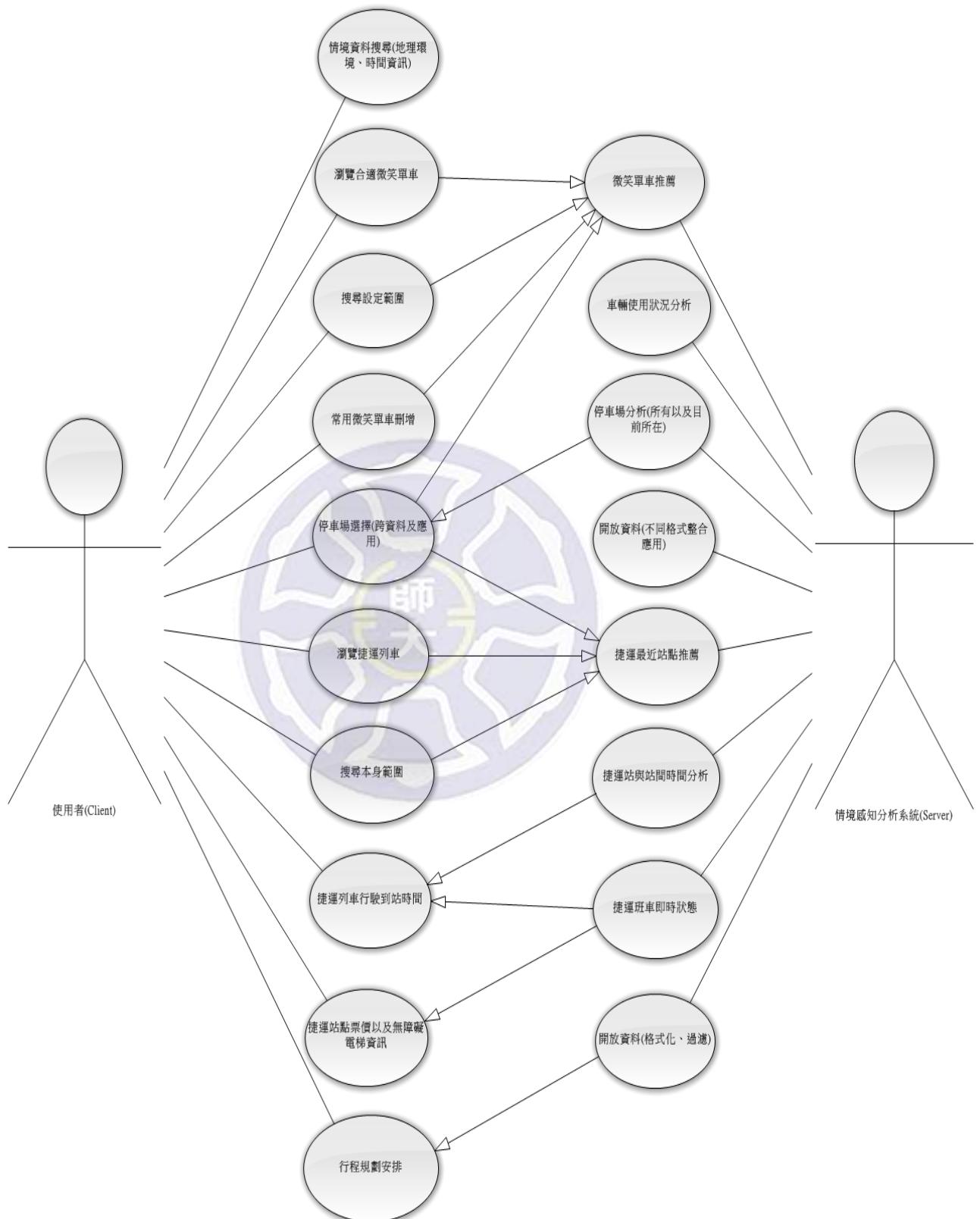


圖 3-17 使用案例圖

在系統中有分成兩個 Actor，為「一般使用者」及「情境感知分析系統」。

其中以下有十七種不同的使用案例：

1. 瀏覽合適微笑單車資訊：在系統中提供基本瀏覽的功能，讓使用者依據自行選擇的任一個微笑單車站點進行相關的瀏覽資訊。
2. 情境資料搜尋：針對使用者周圍的相關情境進行資料的搜尋，主要是蒐集「地理環境資訊」、「時間資訊」進行分析。
3. 搜尋設定範圍：讓使用者可以根據自己需求設搜尋的條件。
4. 常用微笑單車刪增：能依據使用者的期望，設定自己需求的常用站點，利用常用站點能迅速地瀏覽周圍微笑單車的站點。使用者能隨時隨地針對站點進行新增、瀏覽以及刪除。
5. 停車場選擇：根據使用者自行地選擇想要的停車場，針對所選擇的停車場所提供的使用者其相對應於停車場最近的捷運站點資訊或微笑單車站點資訊，到達可以跨資料集的相互配合應用。
6. 開放資料格式化及過濾：在系統中固定時間去政府開放資料進一步來格式化且也能資料過濾。
7. 微笑單車推薦：在系統中依據使用者所設定的推薦以及情境的狀況，來推薦適合使用者微笑單車站點的資訊。
8. 車輛使用狀況分析：根據當時時間點前後使用微笑單車站點車輛的狀況，進一步產生出適合的推薦資訊。

9. 停車場分析：根據使用者目前的地理位置來分析目前的停車場資訊。
10. 瀏覽合適捷運列車：在系統中提供基本瀏覽的功能，讓使用者依據自行選擇的任一個捷運列車進行相關的瀏覽資訊。
11. 搜尋本身範圍：依據使用者的位置把範圍資訊進行蒐集。
12. 捷運列車行駛到站時間：依據政府開放資料的資料，其中兩筆資料進行做處理分析，得出結果提供給使用者做查詢。
13. 捷運站點票價以及無障礙電梯資訊：依據使用者所選擇捷運列車到達的站點，提供此捷運站點的票價以及無障礙電梯資訊。
14. 開放資料不同格式整合應用：在系統中固定時間去抓取資料，進行不同的資料格式做處理且相互的配合應用。
15. 捷運最近站點推薦：在系統中依據使用者的情境狀況，來推薦適合使用者最近的捷運站點資訊。
16. 捷運站與站時間分析：依據政府開放資料所提供的資料，在系統中抓取資料並分析每站的站點行駛時間，另外進行各站點的路線規劃統整。
17. 捷運列車即時狀態：依據政府開放資料所提供的資料，在系統中抓取即時捷運行駛的列車資料，以便做捷運到站時間分析運用。
18. 行程規劃安排：依據政府開放資料所提供的資料，在系統中抓取停車場、微笑單車、捷運資料，再利用設定的地點加以整合一系列的旅遊行程。

3.10 活動圖(Activity Diagram)

依照 Use Case Diagram，歸納出的 Activity Diagram，如以下介紹。

3.10.1 瀏覽所有微笑單車資訊

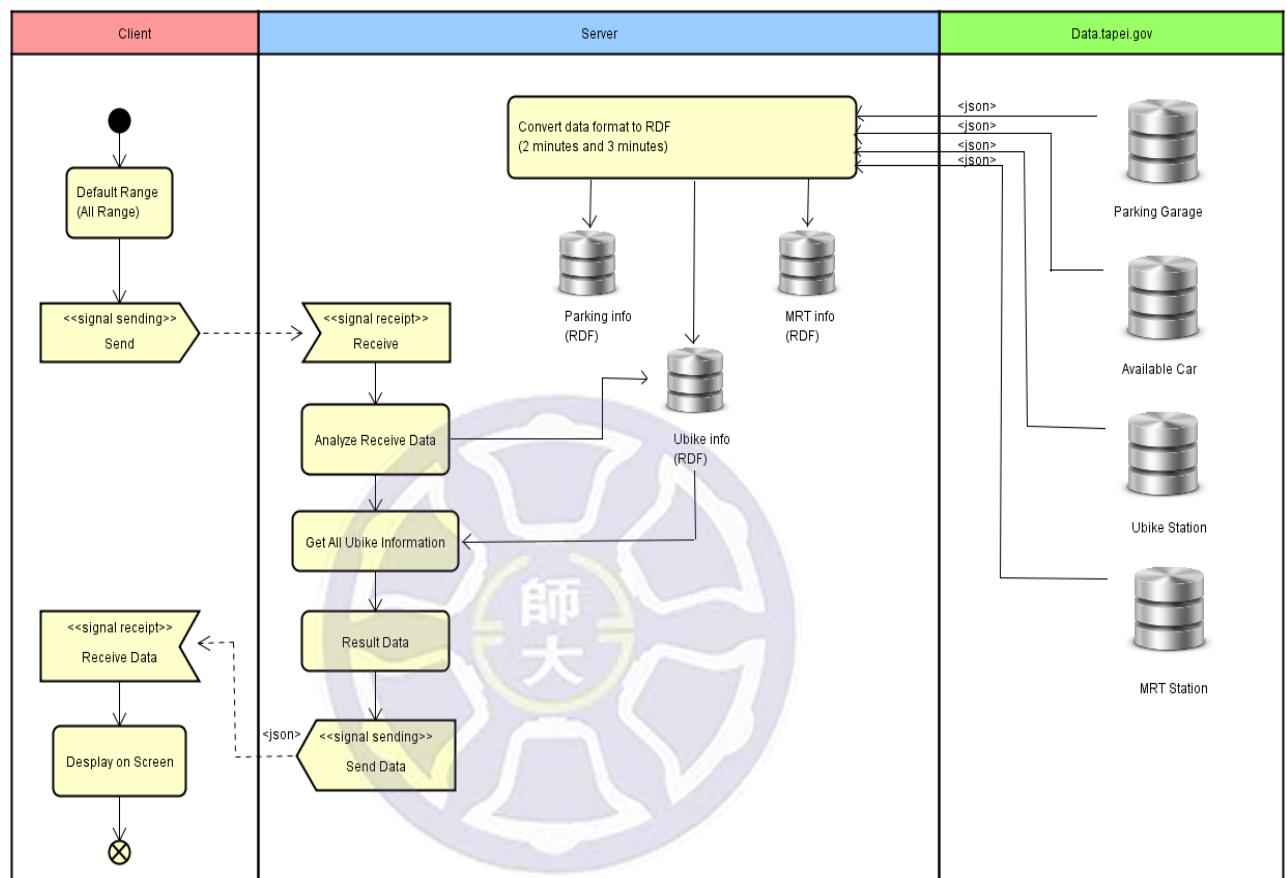


圖 3-18 活動圖 - 瀏覽所有微笑單車資訊

- 使用者首先進入系統後，將預設的條件發送到伺服器做處理
- 伺服器固定從臺北市政府開放資料兩分鐘更新一次後，將資料格式進一步轉換為 RDF 的正規化之格式
- 伺服器會從使用者端傳送過來的條件進行分析且處理，之後抓取從相互對應的 RDF 中所有的微笑單車資訊
- 將所有微笑單車資訊回傳至使用者的行動裝置並顯示於螢幕上呈現

3.10.2 瀏覽合適微笑停車場資訊

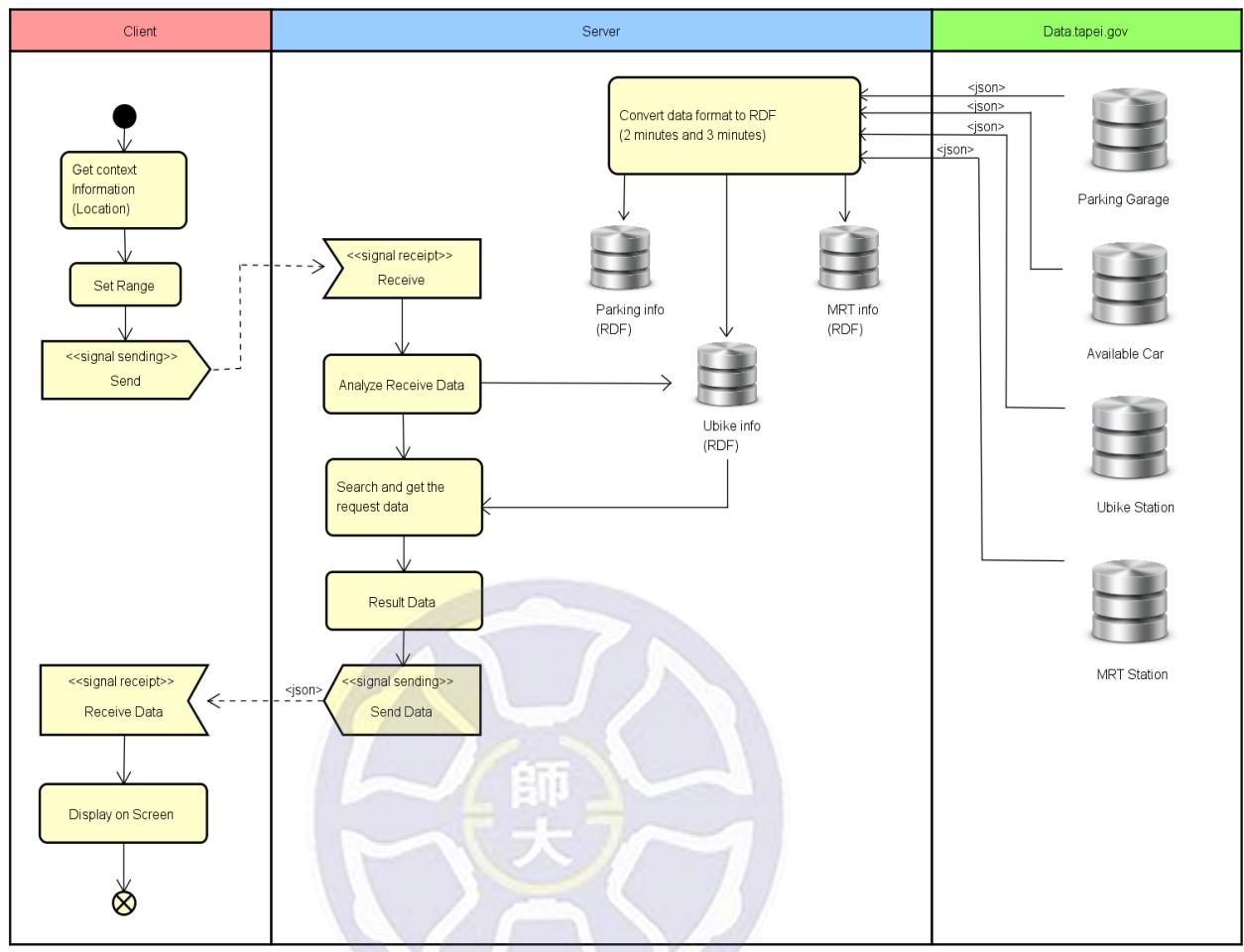
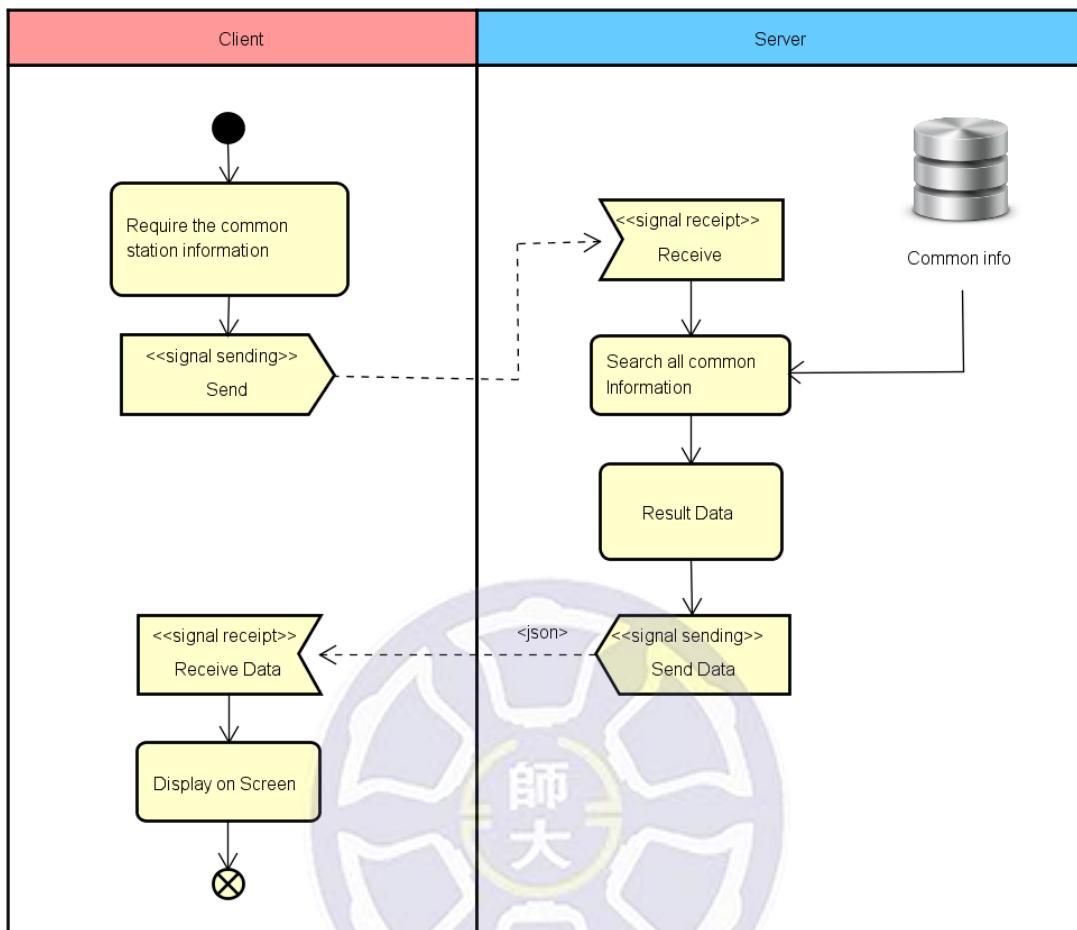


圖 3-19 活動圖 - 瀏覽合適微笑單車資訊

1. 行動裝置偵測周圍的地理情境資訊且自行設定範圍所需的條件(範圍公尺)
2. 將偵測到的使用者設定範圍和地理情境資訊的結果發送到伺服器做處理
3. 伺服器固定從臺北市政府開放資料兩分鐘更新一次後，將資料格式進一步轉換為 RDF 的正規化之格式
4. 伺服器會從使用者端傳送過來的條件進行分析且處理，之後抓取從相互對應且符合條件的 RDF 中微笑單車資訊
5. 將符合條件的微笑單車資訊回傳至使用者的行動裝置並顯示於螢幕上呈現

3.10.3 瀏覽常用微笑單車站點



powered by Astah

圖 3-20 活動圖 - 瀏覽常用微笑單車站點

1. 使用者首先進入系統
2. 對伺服器傳送要求訊息來拿取常用微笑單車站點資訊
3. 在伺服器中讀取所需要常用站點的文件並進行拿取
4. 將所讀取的結果回傳到使用者的行動裝置
5. 將結果顯示到螢幕上呈現

3.10.4 使用者新增常用微笑單車站點

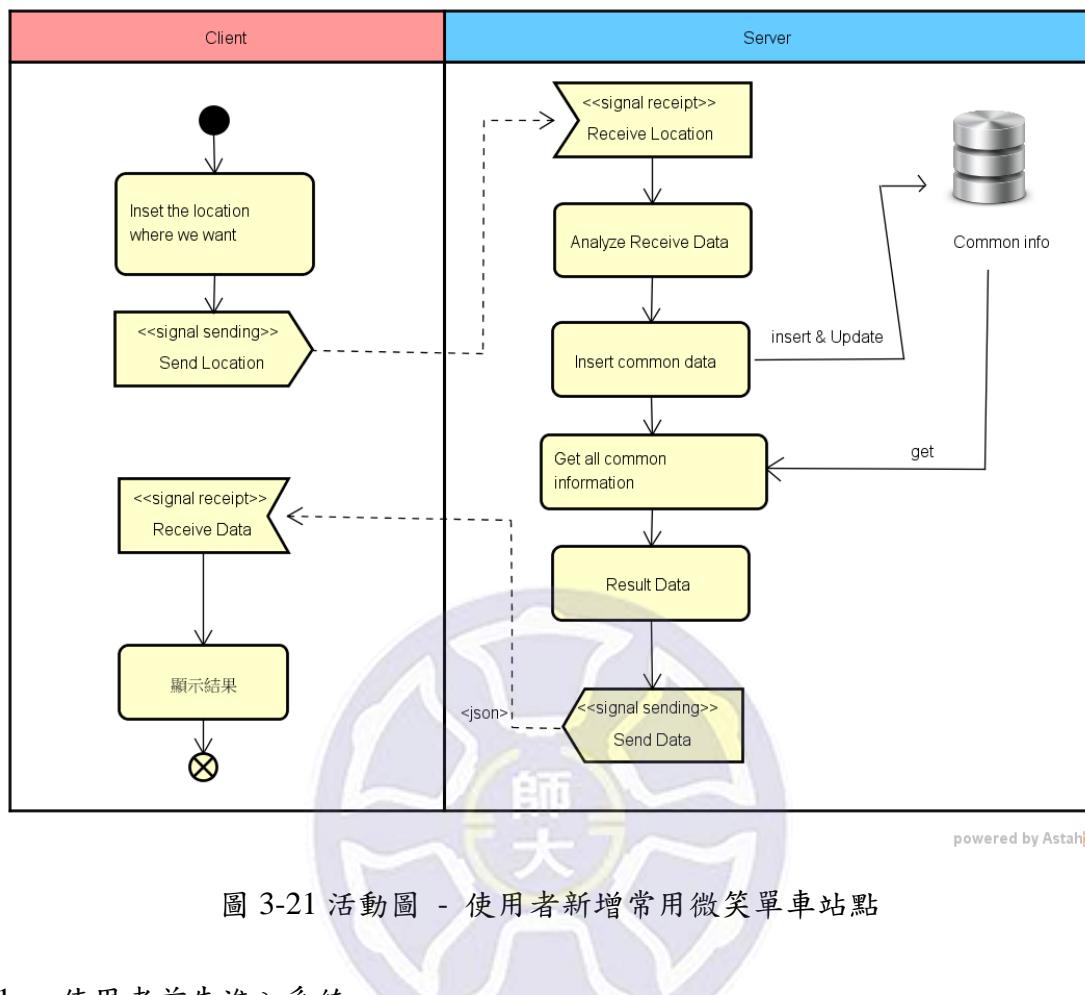
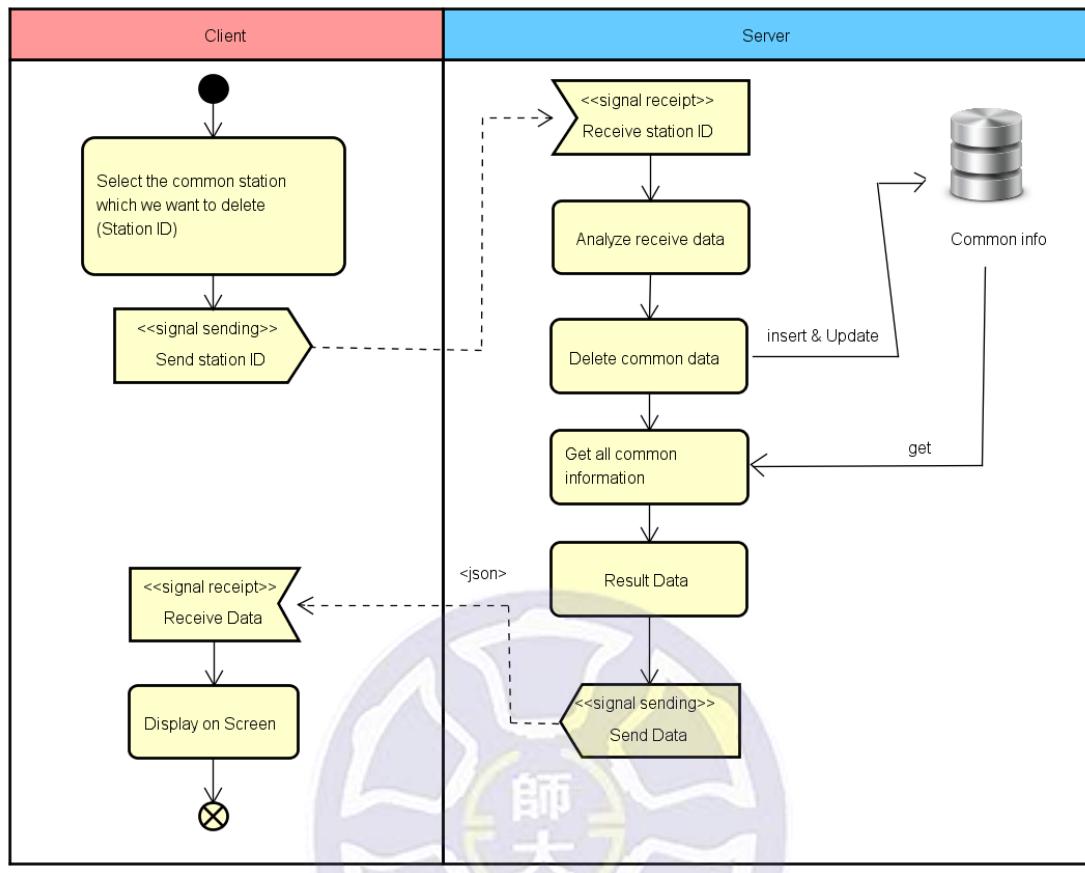


圖 3-21 活動圖 - 使用者新增常用微笑單車站點

1. 使用者首先進入系統
2. 於新增頁面中的 Google map 上選取欲新增站點的位置
3. 將選擇的站點中經緯度傳送到伺服器後，進行位置分析
4. 將新增該站點的相關資訊至常用站點的 RDF 文件檔
5. Server 抓取新增之後的常用站點資訊傳送到使用者行動裝置上
6. 使用者的行動裝置接收回傳的結果後，將結果呈現到螢幕上

3.10.5 使用者刪除常用微笑單車站點



powered by Astah

圖 3-22 活動圖 - 使用者刪除常用微笑單車站點

1. 使用者首先進入系統
2. 挑選使用者想刪除的微笑單車站點
3. 將該微笑單車的站點 ID 傳送到伺服器進行分析動作
4. 從常用站點的文件中把相同 ID 的資料進行刪除
5. 伺服器拿取刪除之後的常用站點資訊傳送到使用者行動裝置上
6. 使用者的行動裝置接收回傳的結果後，將結果呈現到螢幕上

3.10.6 瀏覽所有捷運行駛列車資訊

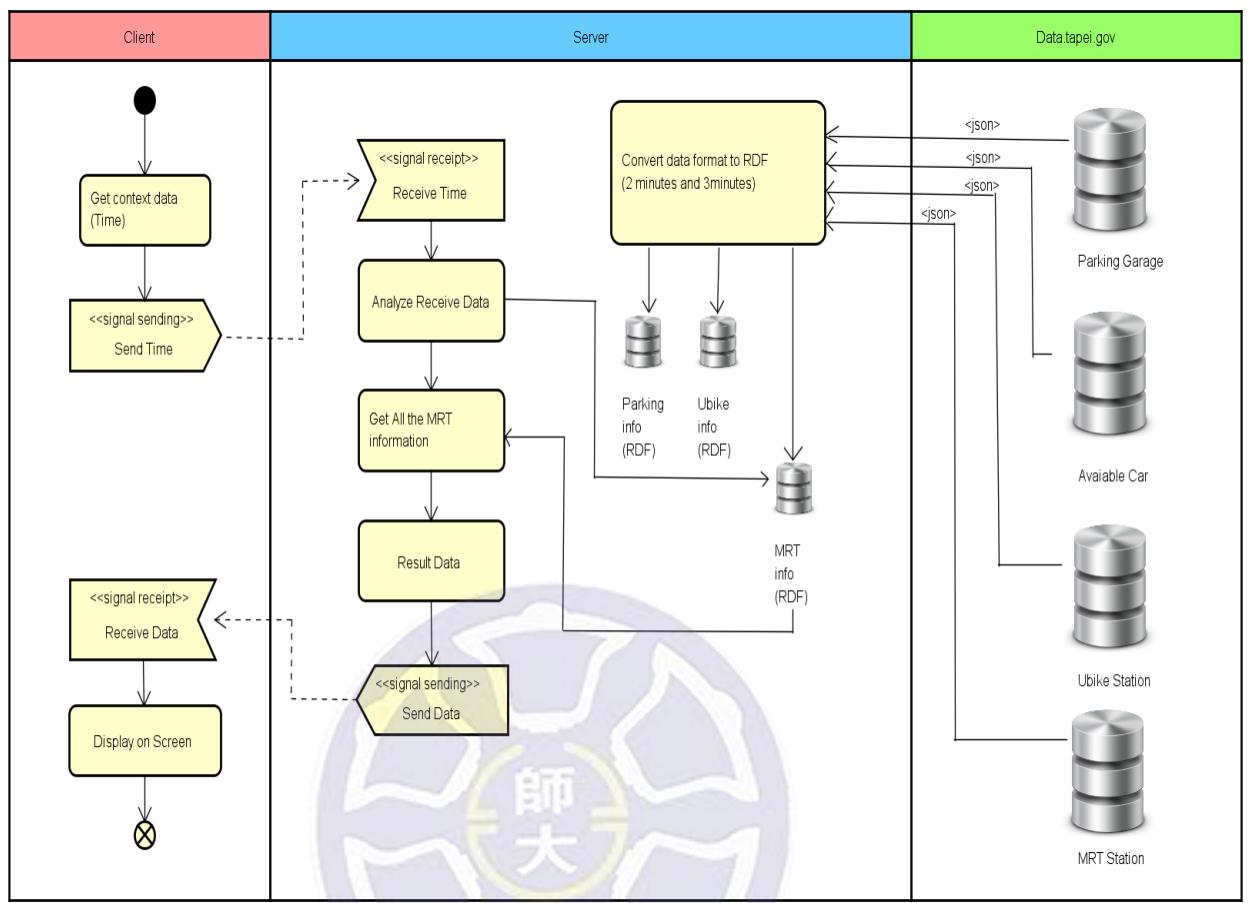


圖 3-23 活動圖 - 瀏覽所有捷運行駛列車資訊

- 使用者首先進入系統時，將時間情境資訊發送到伺服器做處理
- 伺服器固定從臺北市政府開放資料分別每兩分鐘更新停車場和微笑單車資訊
和每三分鐘更新捷運資訊後，將資料格式進一步轉換為 RDF 的正規化之格
式
- 伺服器會從使用者端傳送過來的條件進行分析且處理，之後抓取相互對應的
RDF 中所有的微笑單車資訊
- 將所有微笑單車資訊回傳至使用者的行動裝置

3.10.7 瀏覽最近捷運站點資訊

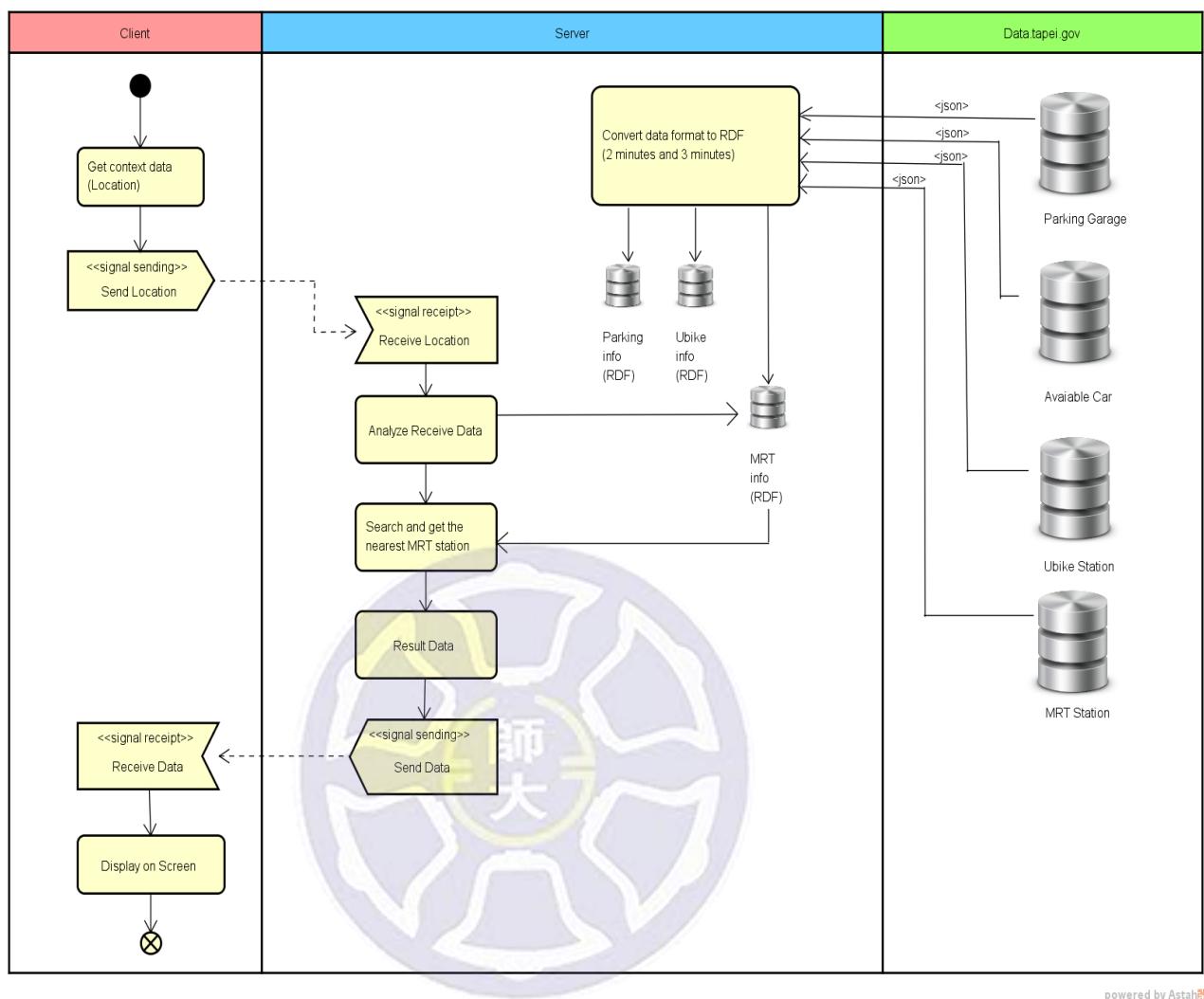


圖 3-24 活動圖 - 瀏覽最近捷運站點資訊

1. 使用者首先進入系統時，偵測所處的地理環境資訊並傳送到伺服器做處理
2. 伺服器固定從臺北市政府開放資料中抓取需要的資料，將資料格式進一步轉換為 RDF 正規化之格式
3. 伺服器會從使用者端傳送過來的資訊進行分析且處理，之後拿取相互對應的 RDF 中捷運站點資訊且根據使用者位置篩選最近的捷運站點
4. 將篩選後的捷運站點資訊回傳至使用者行動裝置並顯示於螢幕上呈現

3.10.8 查詢捷運行駛列車到站時間資訊

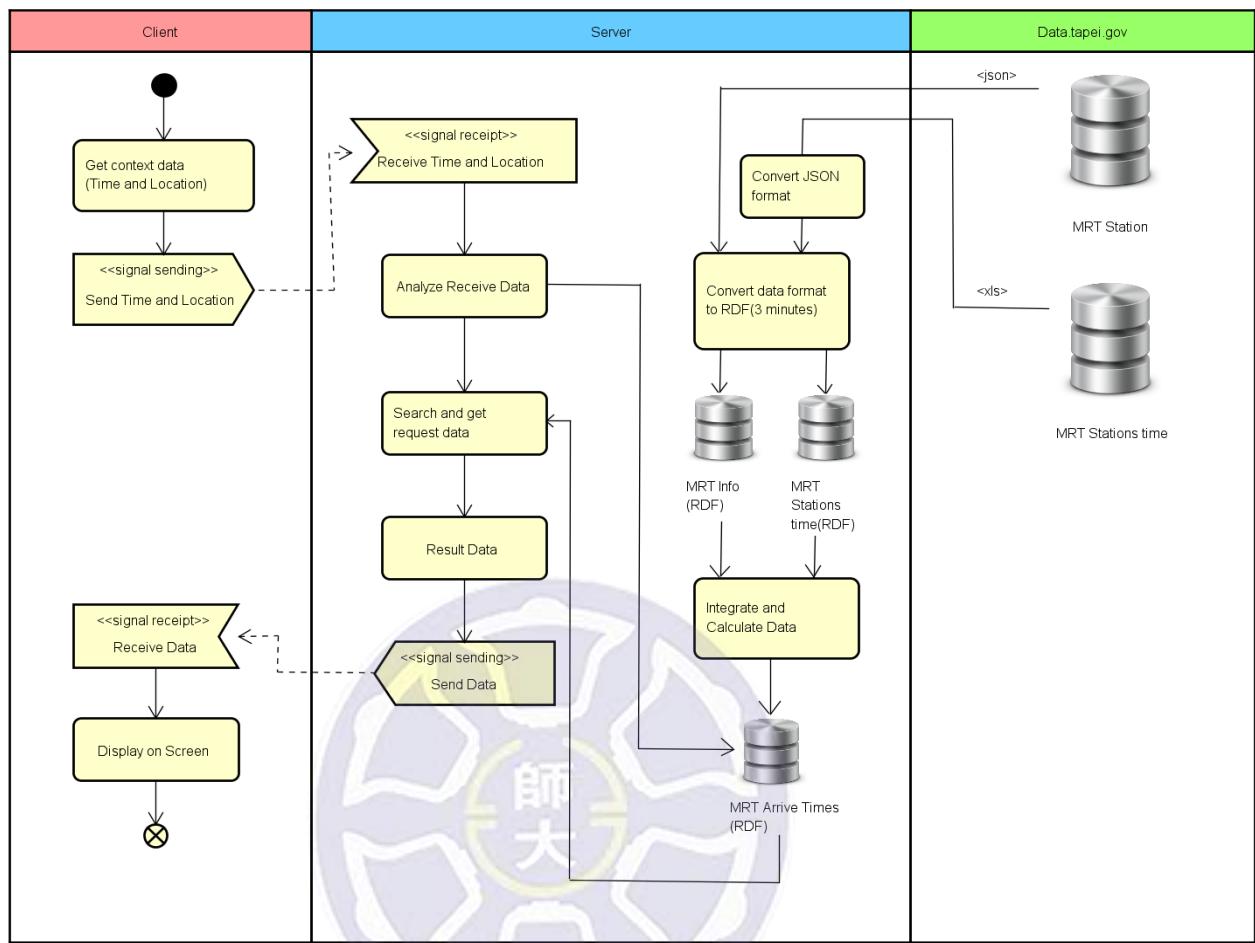
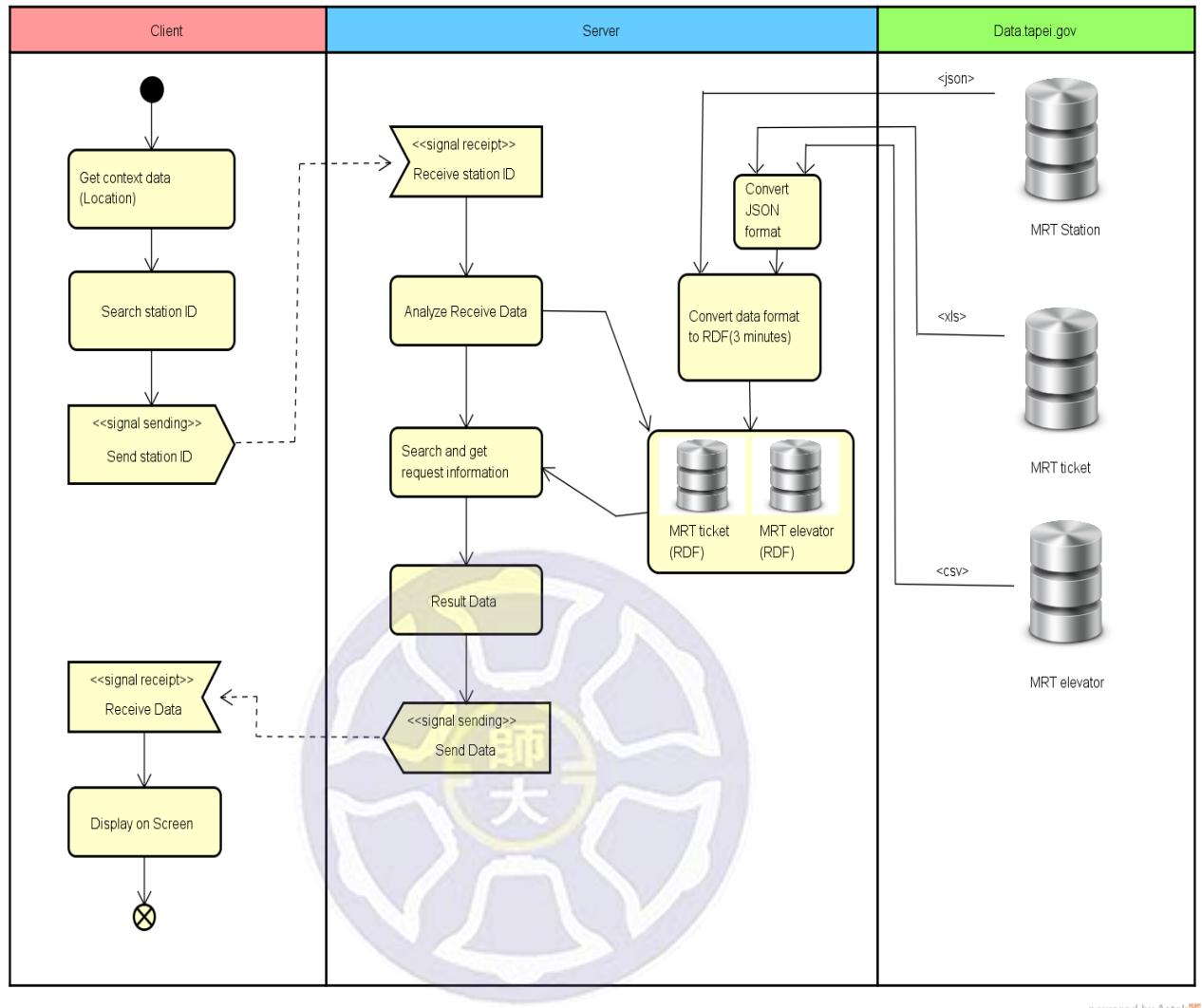


圖 3-25 活動圖 - 查詢捷運行駛列車到站時間資訊

1. 使用者首先進入系統時，偵測所處的地理環境資訊並傳送到伺服器做處理
2. 伺服器固定從臺北市政府開放資料中抓取需要的資料，將不同資料檔案格式進一步統一轉換為個別的 RDF 正規化格式
3. 再將個別的 RDF 正規化透過整合且計算分析後，產生一份整合的 RDF 檔案
4. 伺服器會從使用者端傳送過來的資訊進行處理，之後拿取整合的 RDF 檔案中
有符合使用者目前所在線路上捷運行駛列車的相關到站資料
5. 將所得到的捷運到站資訊回傳至使用者行動裝置並顯示於螢幕上呈現

3.10.9 查詢捷運站點票價以及無障礙電梯資訊

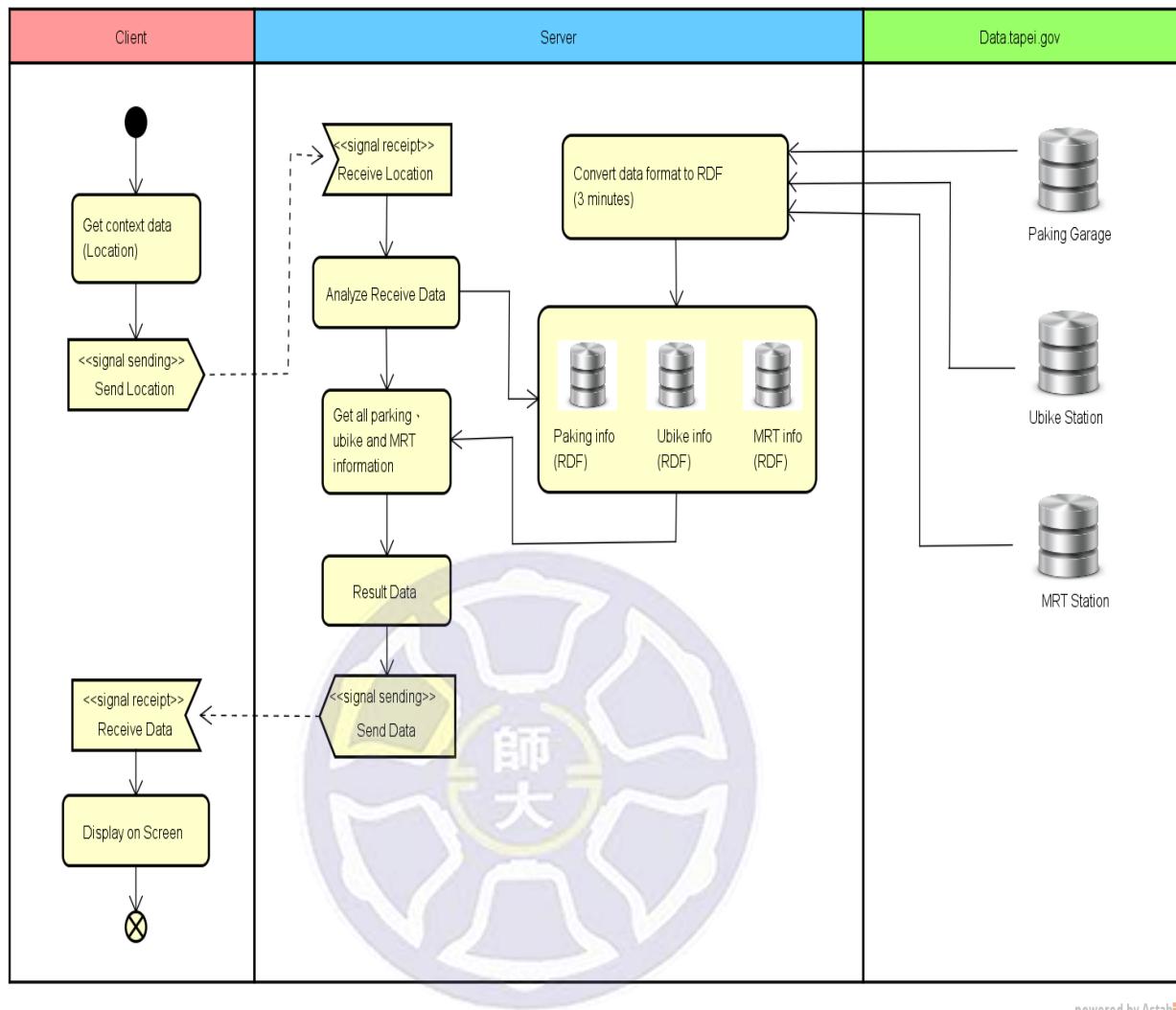


powered by Astah

圖 3-26 活動圖 - 查詢捷運站點票價以及無障礙電梯資訊

- 使用者首先進入系統時，偵測所處的地理環境資訊並傳送到伺服器做處理
- 伺服器從臺北市政府開放資料中抓取需要的資料，將不同資料檔案格式進一步統一轉換為個別的 RDF 正規化格式
- 伺服器會從使用者端傳送過來的捷運站點資訊進行處理，之後拿取捷運票價和無障礙電梯 RDF 檔案中搜尋相關的資料
- 將所得到的結果回傳至使用者行動裝置並顯示於螢幕上呈現

3.10.10 臺北一日遊規劃行程之情境應用



powered by Astah

圖 3-27 活動圖 - 一日遊之規劃行程安排

1. 使用者首先進入系統時，偵測所處的地理環境資訊並傳送到伺服器做處理
2. 伺服器固定從臺北市政府開放資料中抓取停車場、微笑單車和捷運資訊，將不同資料檔案格式進一步統一轉換為個別的 RDF 正規化格式
3. 伺服器會從使用者端傳送過來的資訊進行處理，並優先挑選符合離使用者最近的站點進行優先排序出所有各項交通工具列表
4. 將所得到的各項交通工具站點資訊回傳至使用者行動裝置並呈現於螢幕中

3.10.11 分析所處停車場周圍微笑單車或捷運站點

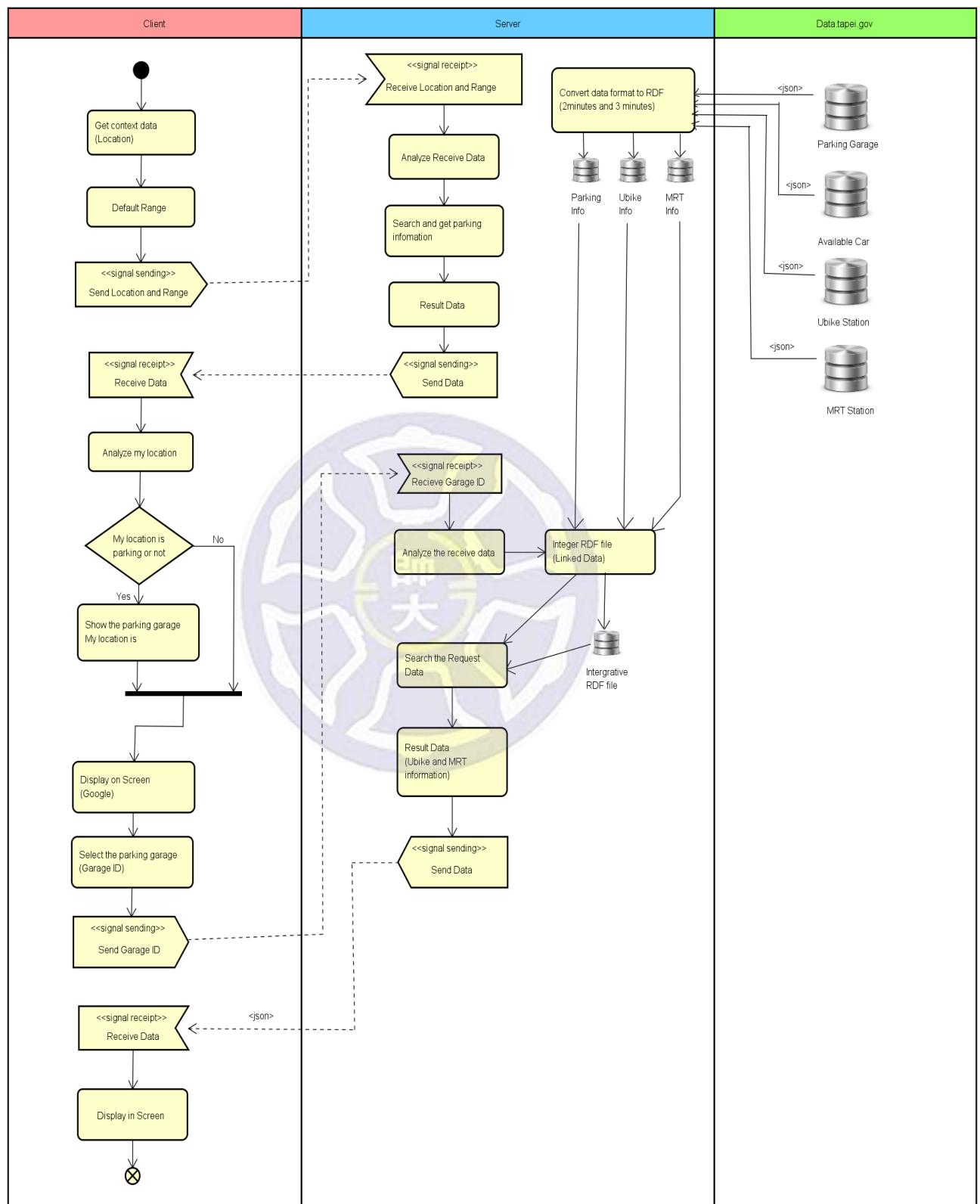


圖 3-28 活動圖 - 分析所處停車場周圍微笑單車或捷運站點

1. 使用者首先進入系統
2. 裝置立即偵測目前周圍的地理環境資訊
3. 從偵測地理環境情境資訊和預設範圍資訊傳送到 Server 端分析且處理
4. 在 Server 端搜尋相互對應的 RDF 檔案中抓取符合需求條件的停車場資訊。

Server 固定每兩和三分鐘去更新不同資料，進一步把資料格式去轉換 RDF 的正規化格式

5. 將所讀取的結果回傳到使用者的行動裝置上
6. 使用者如選擇呈現的停車場站點資訊
7. 行動裝置會把所選擇的停車場相關資訊傳送到 Server 進行分析
8. Server 會把停車場 RDF 資訊檔案、捷運 RDF 資訊檔案以及微笑單車 RDF 資訊檔案去做整合處理，且抓取符合條件的捷運資訊和微笑單車資訊
9. 將所搜尋的結果回傳到行動裝置的螢幕上

3.10.12 瀏覽分析及推薦資料

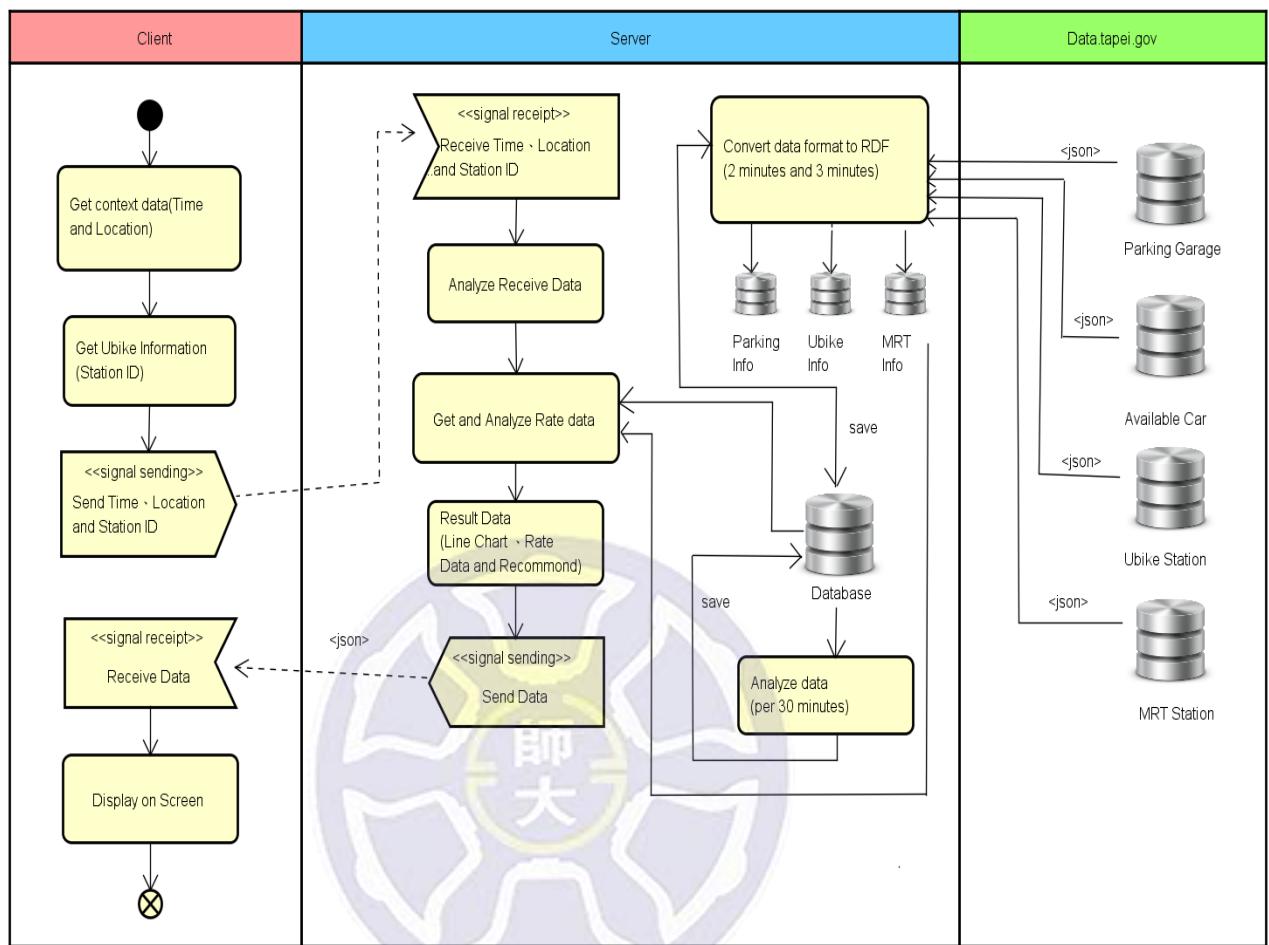
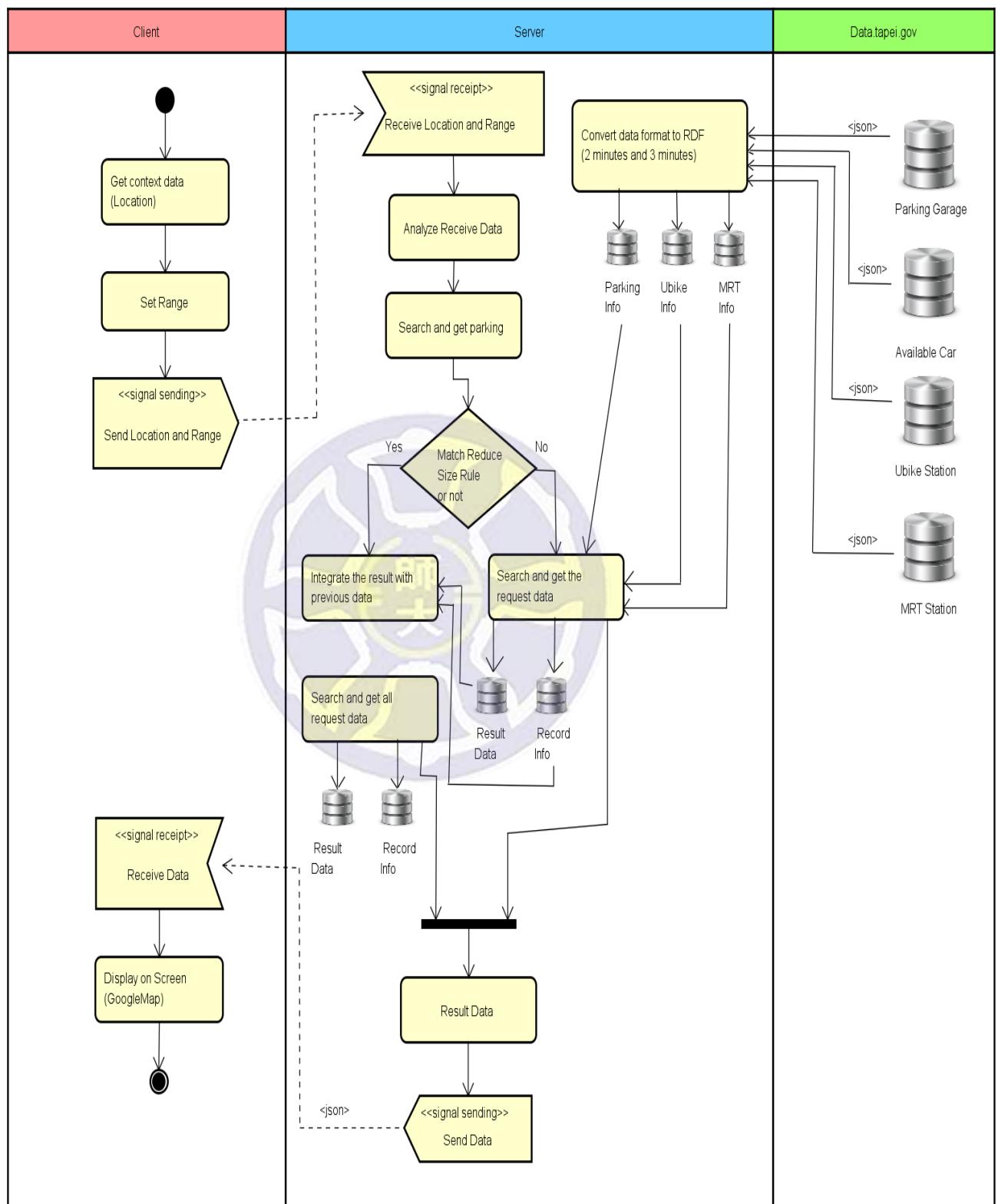


圖 3-29 活動圖 - 瀏覽分析及推薦資料

- 首先進入系統，抓取目前的時間情境資訊，分析所選的微笑單車站點資訊
- 把微笑單車站點資訊和時間情境資訊傳送到 Server 做分析及處理
- 在 Server 端中 Database 會以每兩分鐘記錄一份從臺北市政府開放資料的數據，而且每 30 分鐘分析一次數據
- 從 Database 中拿取符合條件的數據並加以分析處理
- 獲得一份具備推薦資訊結果以及分析的數據圖
- 將分析的結果回傳到使用者行動裝置並顯示於螢幕上呈現

3.10.13 縮減搜尋範圍(Reduce Size)



powered by Astah

圖 3-30 活動圖 - 縮減搜尋範圍(Reduce Size)

1. 首先進入系統，抓取目前的地理環境情境資訊且設定範圍的條件
2. 把地理環境情境資訊和設定範圍傳送到伺服器進行處理
3. 分析後判斷是否有符合 Reduce Size 中的條件
4. 若不符合 Reduce Size 的規則，會從最大範圍中 RDF 資料檔案抓取有符合條件的資料，且產生一筆相關紀錄檔以及結果檔案
5. 若符合 Reduce Size 規則，會從先前搜尋所產生的相關紀錄檔以及結果檔案來分析且抓取有符合條件的資料，同步地產生一筆相關紀錄檔以及結果檔案
6. 將所搜尋的結果傳送回使用者行動裝置並呈現於螢幕中



3.10.14 動態更新政府開放資料

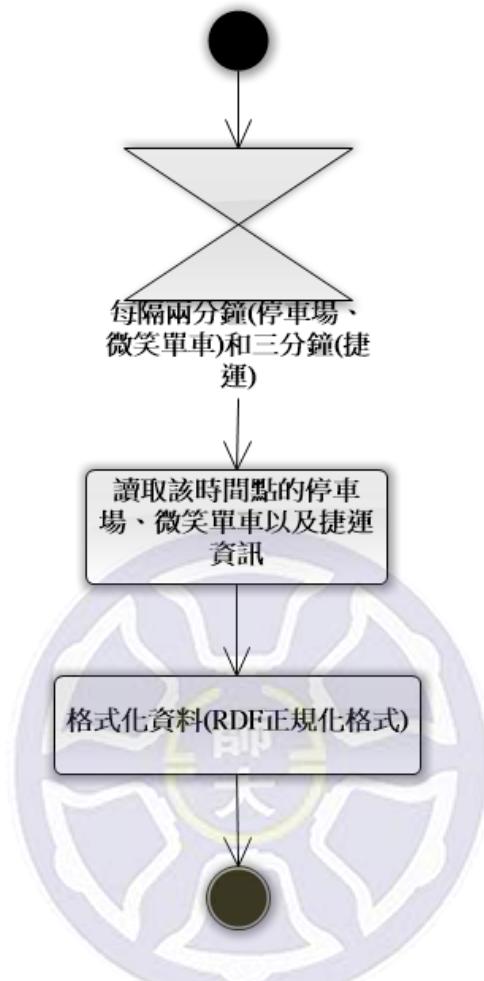


圖 3-31 活動圖 - 動態更新政府開放資料

1. 每隔兩分鐘以及三分鐘分別執行此 Process
2. 對臺北市政府開放資料的平台抓取停車場即時的空位資訊、微笑單車即時的車輛資訊以及捷運即時行駛資訊
3. 進一步地把所獲得資料轉換成 RDF 正規化格式

3.10.15 使用狀況分析

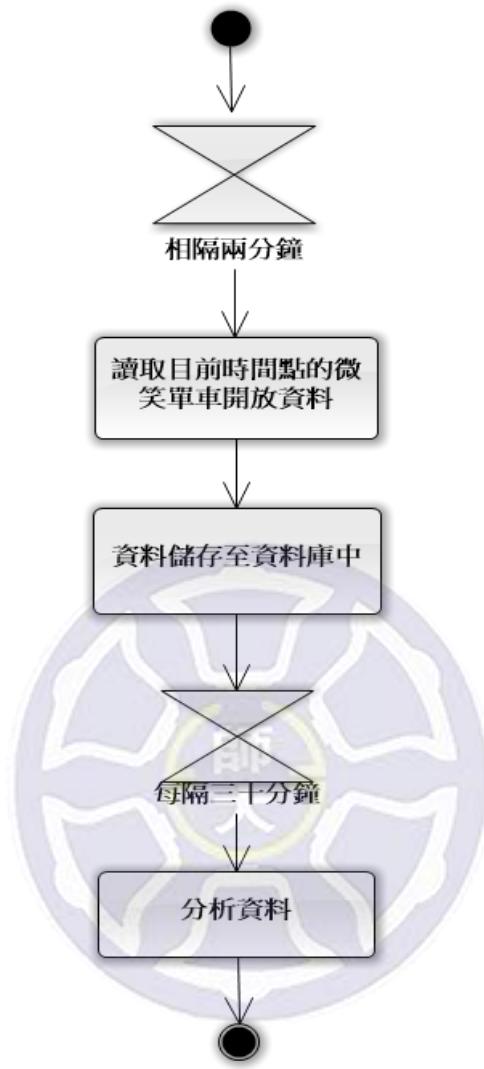


圖 3-32 活動圖 - 使用狀況分析

1. 固定每隔兩分鐘從臺北市政府開放資料的平台去更新資料，並且把即時資料存取到資料庫中
2. 固定每隔三十分鐘針對資料庫所蒐集的資料處理使用狀況的分析，因為每隔兩分鐘去臺北市政府開放資料將即時資訊更新且儲存一次，以此計算，三十分鐘能有十五筆的資料，針對這三十分鐘內所獲得的十五筆資料去做分析。

第四章 系統開發與實作

4.1 系統架構

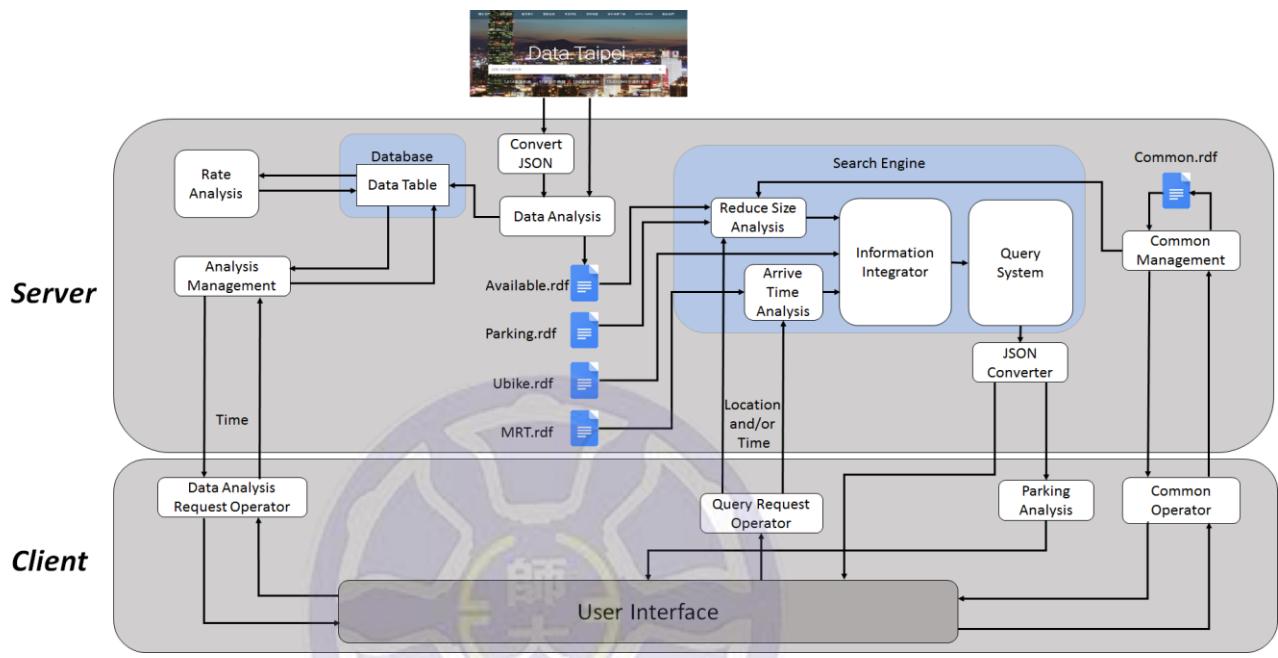


圖 4-1 系統架構圖

如圖 4-1 是實際所開發出的系統架構。此系統主要分為兩部分：Server 端和 Client 端。而 Server 端為主要為 Search Engine、Data Analysis、Analysis Management 以及 Common Management。

I. Data Analysis：於臺北市政府公開資料進行更新且把獲得的檔案轉換為 RDF 正規化格式，根據資料不同進行每兩或三分鐘執行抓取動作，且將當下所得到之即時資訊儲存到資料庫中，進一步處理且分析。

II. Search Engine：其中細分為四大部分。

i. Reduce Size Analysis：首先從 Client 端發出請求和情境資訊進行

分析，判斷有無符合觸發規則。同時 Reduce Size Analysis 會將 Common Management 整合發出要求資訊進一步條件分析且判斷。

ii. Arrive Time Analysis：從 Client 端發出請求和地理環境及時間情境資訊進行分析，利用捷運行駛即時資訊進行計算，判斷捷運多久會到達以及此路線是否為使用者目前會到達的列車。

iii. Information Integrator：把 Reduce Size 所分析出的結果和 Arrive Time Analysis 所計算出的結果進行四份檔案的相互整合鏈結，四份檔案分別為：「停車場地理環境資訊」、「停車場動態車位」、「微笑單車站點資訊」以及「捷運資訊」，四份檔案已事先處理為

RDF 格式不同檔案。

iv. Query System：此為 Linked Open Data 之 Query 介面，能把上述的步驟經過處理的檔案進一步做存取和搜尋的動作，由於所有的檔案已事先處理轉換為 RDF 格式檔案，Query System 可以利用 SPARQL 做資料存取的功能。

III. Common Management：此為進行管理常用站點，也可以透過使用者新增刪除的需求進一步做資料存取的動作。

IV. Analysis Management：根據從使用者操作時所接收的時間及站點的資訊進行分析，接著至資料庫中抓取與其對應的數據進一步做處理，給予合適推薦。

V. Data Table：為了存取從臺北市政府開放資料和經過分析的數據所做的資料庫欄位。

VI. Rate Analysis：此分析為將存取至資料庫中的微笑單車使用數據以每三十分鐘為基準，把三十分鐘以內所獲得的數據進一步分析且將運算得出的結果儲存至資料庫中，提供之後參考使用。

VII. JSON Converter：為了開發上的方便和降低流量之因素，在進行傳輸資料時候把資料包裝為 JSON 送出。

Client 端，指的是使用者操作的 Android 行動裝置，其主要功能是瀏覽停車場、微笑單車以及捷運資訊、偵測所處停車場與拿取分析的數據並推薦以及一日遊規劃行程的安排，利用四個單元處理以上的事件：Common Operator 、Parking Analysis 、Query Request 、Data Analysis Request Operator 。

I. Common Operator：為常用站點操作單元，透過此單元能讓使用者針對常用站點進行一系列的查詢、新增以及刪除的動作，主要為整合使用者的操作行為和相關站點的資訊傳送至 Server 端給予合適處理。

II. Parking Analysis：使用者若操作有關停車場和微笑單車跨資料集之應用時，此時系統將分析出使用者目前的位置是否為停車場。

III. Query Request Operator：為搜尋請求操作單元，透過此單元能整合使用者目前地理情境資訊，也包括預設條件或者使用者自行的範圍條件，把整合出的結果資訊回傳至 Server 端給予適當的搜尋做處理。

IV. Data Analysis Request Operator：為數據分析請求運算單元，若使用者預期

至 Server 端請求指定微笑單車的站點目前使用狀況以及合適的推薦，此時

會蒐集時間情境資訊和從行動裝置要求站點的資訊，將兩筆資訊整合並發

送至 Server 端來分析數據以及進行合適的推薦資訊，接著從 Server 端接受

到的結果呈現在使用者的螢幕中。

4.2 類別圖(Class Diagram)

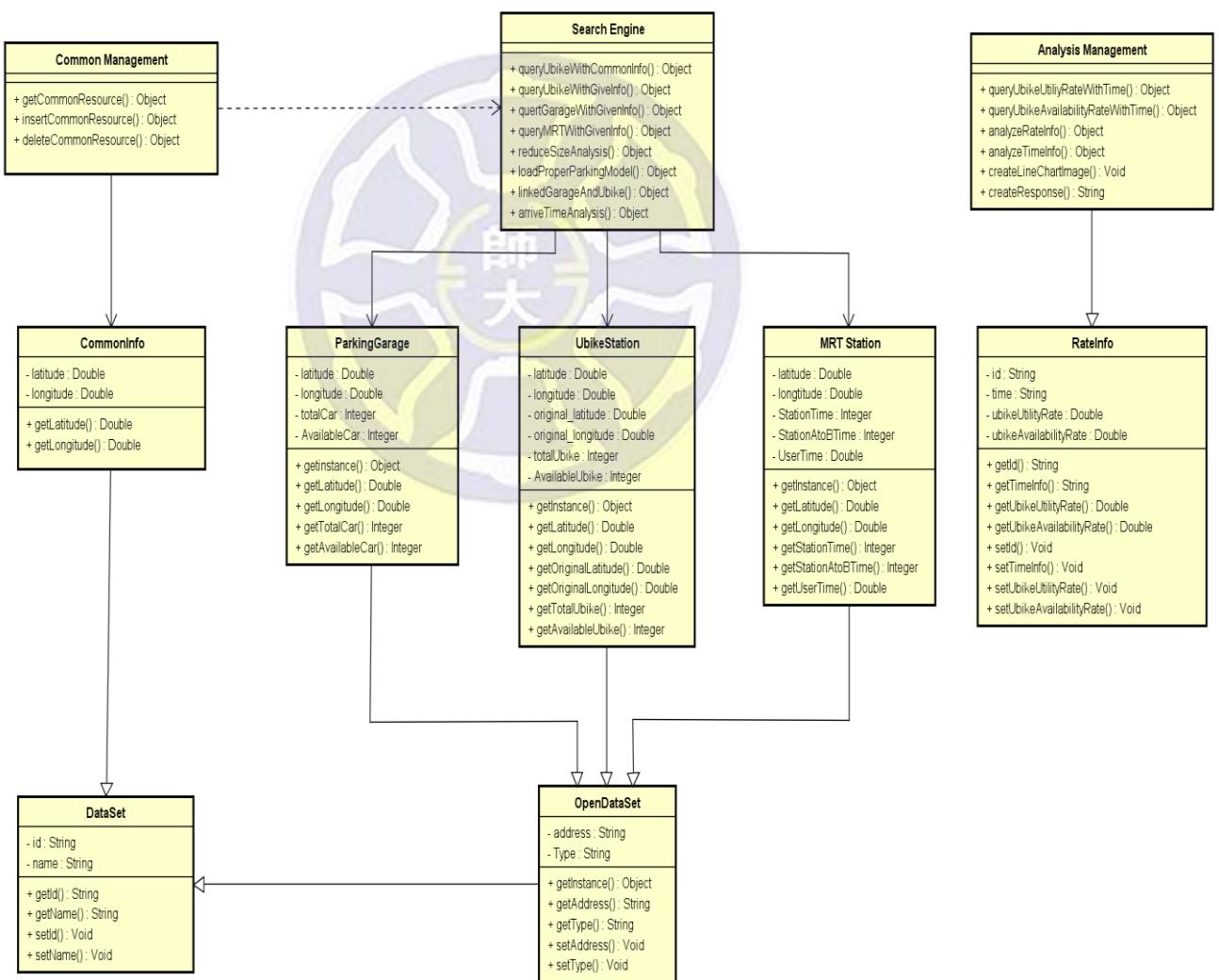


圖 4-2 系統類別圖

如圖 4-2 是 Server 端之 Class Diagram，把所有使用到的資料封裝成模組化。

1. DataSet：利用臺北市政府開放資料和常用站點資訊加以模組化為基本的資料結構，將基本的資訊儲存至 DataSet，例如有 ID、Name。其功能為最基本的 Setter 以及 Getter。
2. OpenDataSet：此繼承於 DataSet 的資料結構，在研究所利用的臺北市政府開放資料，提供不同的地理環境資訊，於研究中所利用三筆不同類型的資料，故在此資料結構中將新增兩個參數，分別為 Type 以及 Address。
3. 為了將 SPARQL 搜尋的結果做橋接管道，故新增一個 getInstance 的操作，目的是物件將具備 Query 所得到結果且建立相互對應的物件，全部的開放資料中子類別皆繼承這項操作。
4. ParkingGarage：此繼承於 OpenDataSet，額外新增停車場資訊的參數加以記錄，例如有 Longitude 、Latitude 、availableCar(剩餘空位個數)、totalCar(總單位數量)。
5. UbikeStation：此繼承於 OpenDataSet，額外新增微笑單車資訊的參數加以記錄，例如有 Longitude(二度分帶座標 x 軸)、Latitude(二帶分度座標 y 軸)、original_longitude(經度)、original_latitude(緯度)、availableUbike(剩餘單車數量)、totalUbike(總微笑單車數量)。
6. MRTStation：此繼承於 OpenDataSet，額外新增微笑單車資訊的參數加以記錄，例如有 Longitude 、Latitude 、StationTime(站點停靠時間)、StationAtoBTime(相鄰站與站間行駛時間)、UserTime(目前時間情境資訊)。

7. CommonInfo：此繼承於 DataSet，在常用站點文件中有包括每個站點的獨有識別 ID 和該站點名稱，額外新增站點中環境資訊的參數加以記錄，例如有 Latitude 以及 Longitude，方便系統處理的好處。
8. RateInfo：此是將分析數據的資料結構加以儲存，基本的參數有 ubikeAvailabilityRate(微笑單車之可用率)、ubikeUtilityRate(微笑單車之使用率)、ID(獨有識別碼)、time(時間情境資訊)。
9. Search Engine：用來執行 Reduce Size 分析、Arrive Time 分析以及 SPARQL Query 的處理類別，不進行額外資料的紀錄。利用 SPARQL 的分析，於 Search Engine 中可能會有零至多個 UbikeStation、ParkingGarage 以及 MRTStation 物件中。
10. Common Management：用來執行常用站點的操作類別，不進行額外資料的紀錄。Common Management 和 Search Engine 具備相依性的關係，由於 Search Engine 會利用到 Common Management 的操作類別，故 Common Management 可能存在零至多個 CommonInfo 物件中。
11. Analysis Management：用來執行處理數據分析的操作類別，不進行額外資料的紀錄。於 Analysis Management 中可能存在零至多個 RateInfo 物件，故利用相互對應的應用功能可以產生出推薦資訊和分析數據圖。
- 在 Client 端中所有 Data 基本上皆由 Server 獲得，其資料架構是沿襲 Server 的設計概念。

4.3 循序圖(Sequence Diagram)

以下為各種實作的 Sequence Diagram 進行詳細解說。

4.3.1 瀏覽所有微笑單車站點資訊

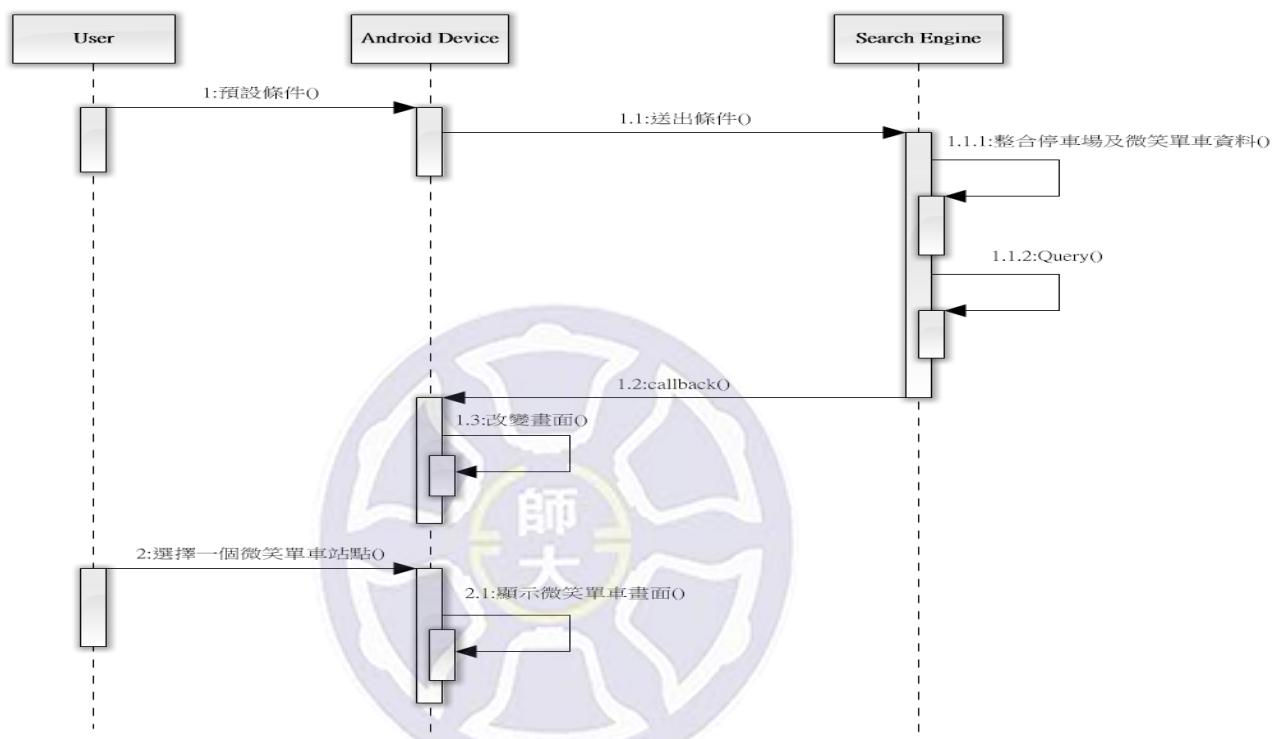


圖 4-3 循序圖 - 瀏覽所有微笑單車站點

使用者首先進入至瀏覽所有微笑單車站點的畫面，系統將採用預設條件，此條件是『所有範圍』的查詢訊息，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。系統將所查詢所有範圍的微笑單車站點回傳至 Android Device，顯示在行動裝置的螢幕上。接著使用者可以任意地選擇畫面中任何一個微笑單車站點進行瀏覽詳細內容的地圖資訊和微笑單車站點資訊，且能開啟 Google Map 進一步導航和使用狀況推薦分析來做使用。

4.3.2 瀏覽合適微笑單車站點資訊

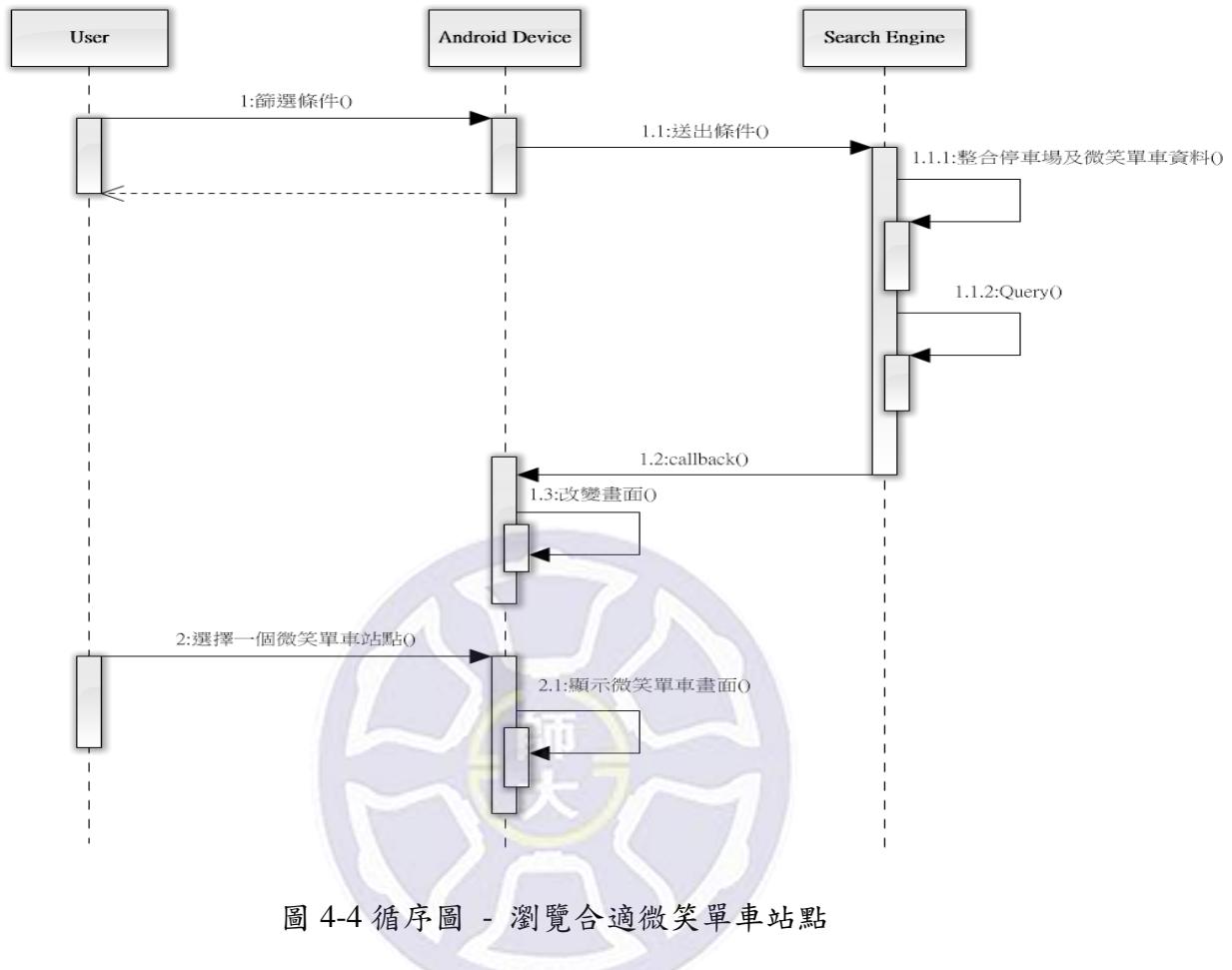


圖 4-4 循序圖 - 瀏覽合適微笑單車站點

此活動讓使用者進行驅動。使用者利用應用程式介面來自行設定範圍參數的需求，Android Device 將使用者設定參數及偵測的地理環境資訊做為查詢訊息而送至 Search Engine。為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。Search Engine 將根據「使用者自行設定的範圍」以及「地理環境資訊」，整合一份符合此兩種條件之微笑單車站點，之後將其結果回傳到 Android Device 更新畫面，接著使用者可以任意地選擇畫面中任何一個微笑單車站點進行瀏覽詳細內容的地圖資訊和微笑單車站點資訊，且能開啟 Google Map 進一步導航和使用狀況推薦分析來做使用。

4.3.3 瀏覽常用站點並查看

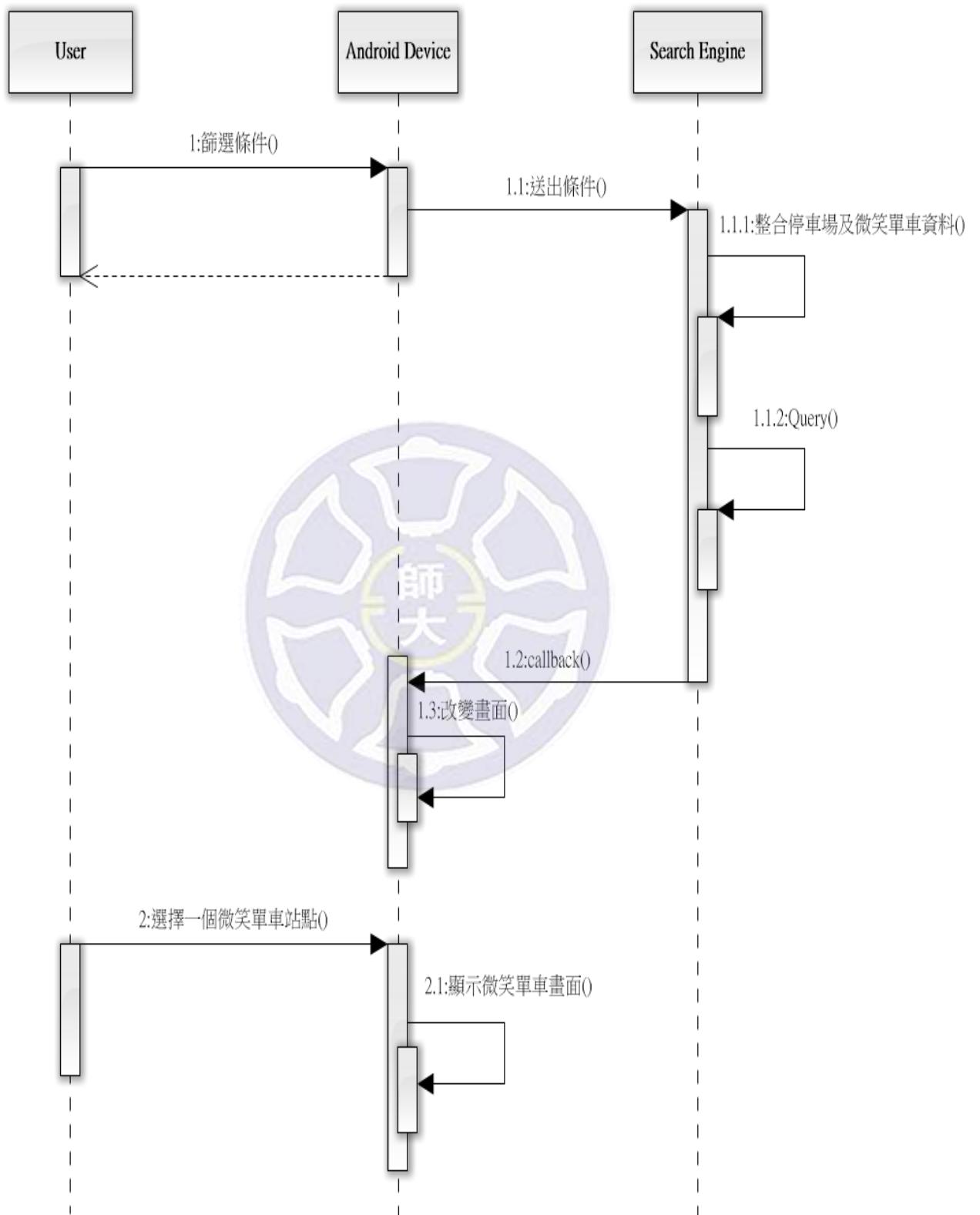


圖 4-5 循序圖 - 瀏覽常用站點並查看

此活動讓使用者進行驅動。每當使用者使用瀏覽常用站點介面時，此時 Android Device 會主動發送出「要求目前已設定的常用站點」的訊息至 Common Management。為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。當 Common Management 收到此要求訊息時，會到儲存於分析常用站點的文件為 Common file，並且讓所有常用站點的資訊當作結果回傳到 Android Device。Android Device 收到結果之後立即更新於畫面，顯示所有收到的常用站點資訊，能讓使用者來瀏覽且操作。

當使用者選擇 Android Device 螢幕的常用者點時，Android Device 將偵測所選擇的站點 ID 且發送出「要求該 ID 站點相關資訊」的訊息至 Common Management，上述也是利用「非同步」的方式進一步查詢。Common Management 若收到要求之後，會到儲存於常用站點的文件為 Common file 來進行相對應 ID 中相關地理位置資訊且會發送出「要求該地理位置周圍中微笑單車站點資訊」的訊息至 Search Engine，Search Engine 收到此訊息會整合一份停車場以及微笑單車資訊，並且拿取符合所要求的資訊當作結果。此結果回傳至 Android Device 中更新畫面，顯示符合該微笑單車常用站點的結果。接著使用者可以任意地選擇畫面中任何一個微笑單車站點進行瀏覽詳細內容的地圖資訊和微笑單車站點資訊，且能開啟 Google Map 進一步導航和使用狀況推薦分析來做使用。

4.3.4 新增常用微笑單車站點

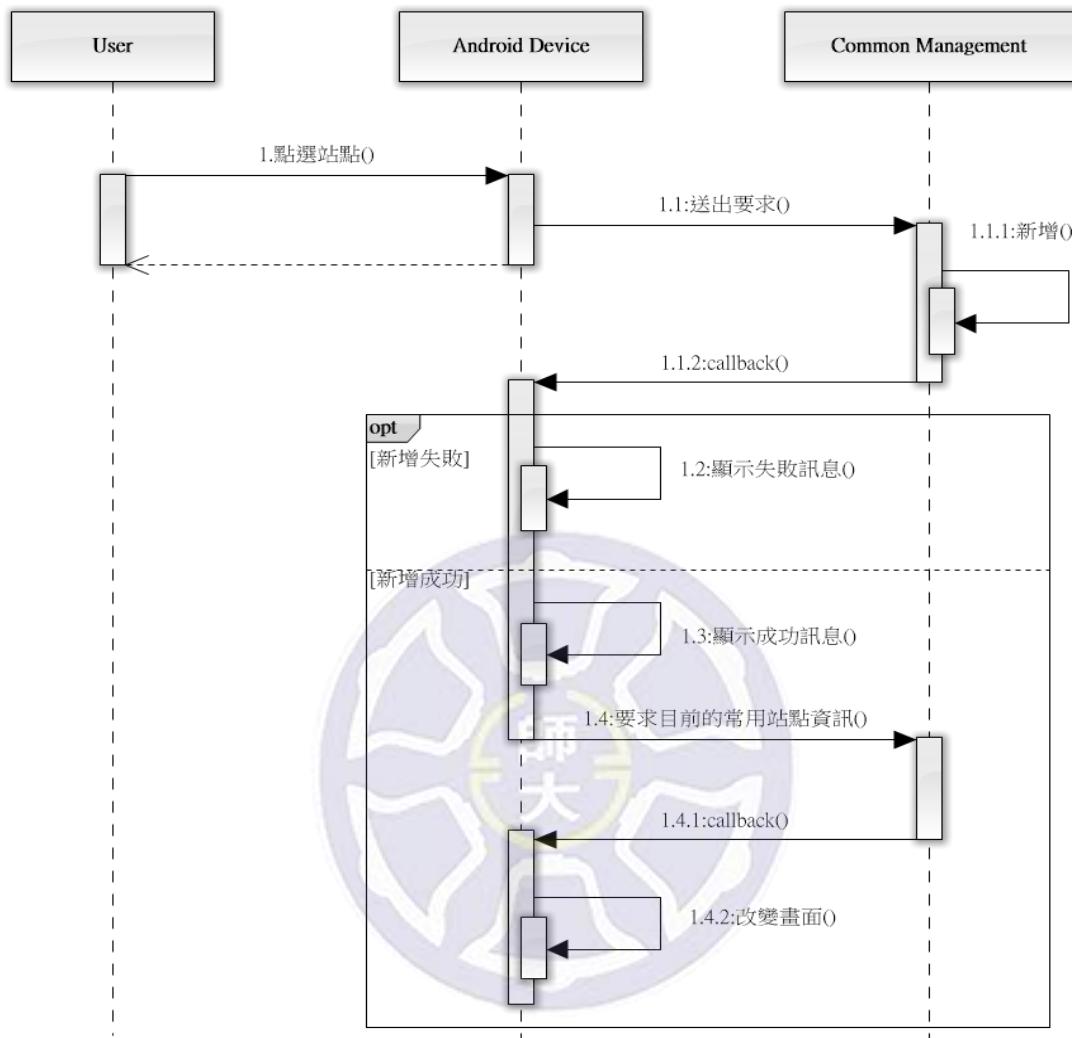


圖 4-6 循序圖 - 新增常用站點

此活動讓使用者進行驅動。使用者首先使用新增站點介面時，選擇 Google Map 中任何一個地點來按下新增按鍵時，Android Device 會發送出「新增的要 求」到 Common Management。Common Management 會讓站點資訊新增至常用 站點文件之後，會跳出「確認訊息」來確認新增是否成功，將此訊息呈現畫面上，若新增成功時，Android Device 會到 Common Management 提出目前所有已 設定的常用站點的要求且顯示在螢幕中。若新增失敗時，會提醒失敗訊息。

4.3.5 刪除常用微笑單車站點

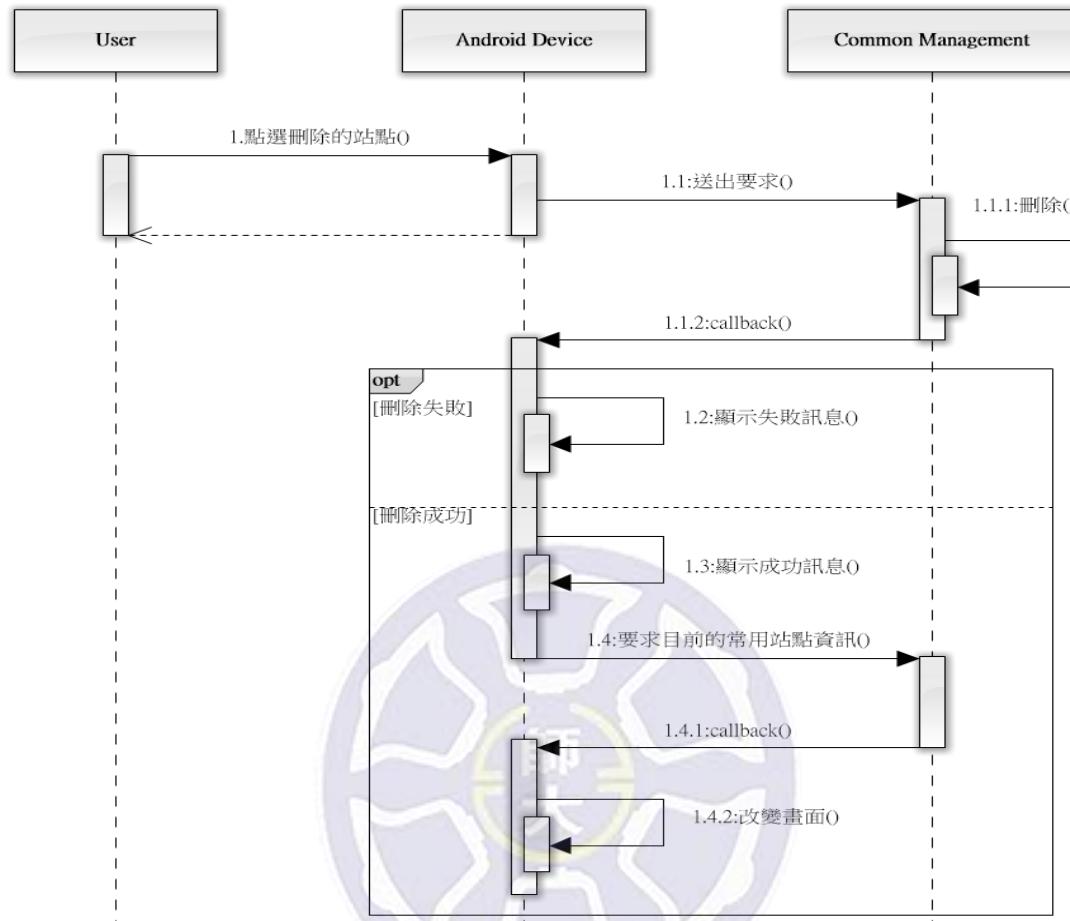


圖 4-7 循序圖 - 刪除常用站點

此活動讓使用者進行驅動。當使用者於常用站點的瀏覽畫面中進行長按住欲刪除的常用站點，此時跳出詢問是否刪除的視窗。若選擇「否」，會毫無動作。若選擇「是」，Android Device 會發送出「刪除的要求」到 Common Management，收到要求之後會立即到儲存於常用站點的文件刪除相對應的站點資訊且回傳「確認訊息」。Android Device 確定成功刪除會提醒成功訊息，之後到 Common Management 提出目前所有已設定的常用站點的要求且顯示在螢幕中。當刪除失敗，會提醒失敗訊息。

4.3.6 瀏覽所有捷運行駛列車資訊

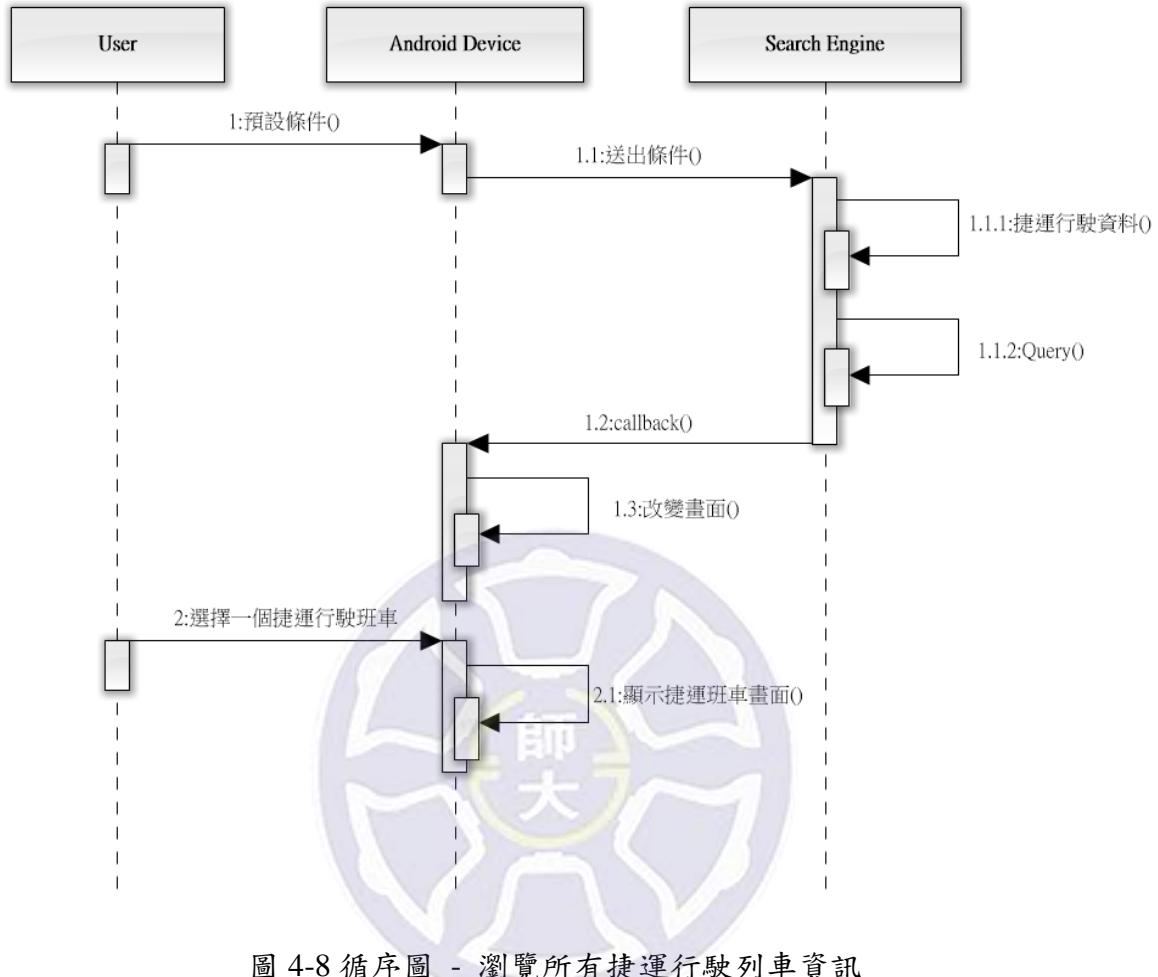


圖 4-8 循序圖 - 瀏覽所有捷運行駛列車資訊

此活動讓使用者進行驅動。使用者首先進入至瀏覽所有捷運行駛列車的畫面，系統將採用預設條件，此條件是『目前時間』的查詢訊息，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。系統將所查詢所有範圍的捷運行駛列車回傳至 Android Device，顯示在行動裝置的螢幕上。接著使用者可以任意地選擇畫面中任何一個捷運行駛列車資訊進行瀏覽詳細內容的最近捷運站點、列車到站時間、無障礙電梯資訊以及地圖資訊，且能開啟 Google Map 進一步導航使用。

4.3.7 瀏覽最近捷運站點資訊

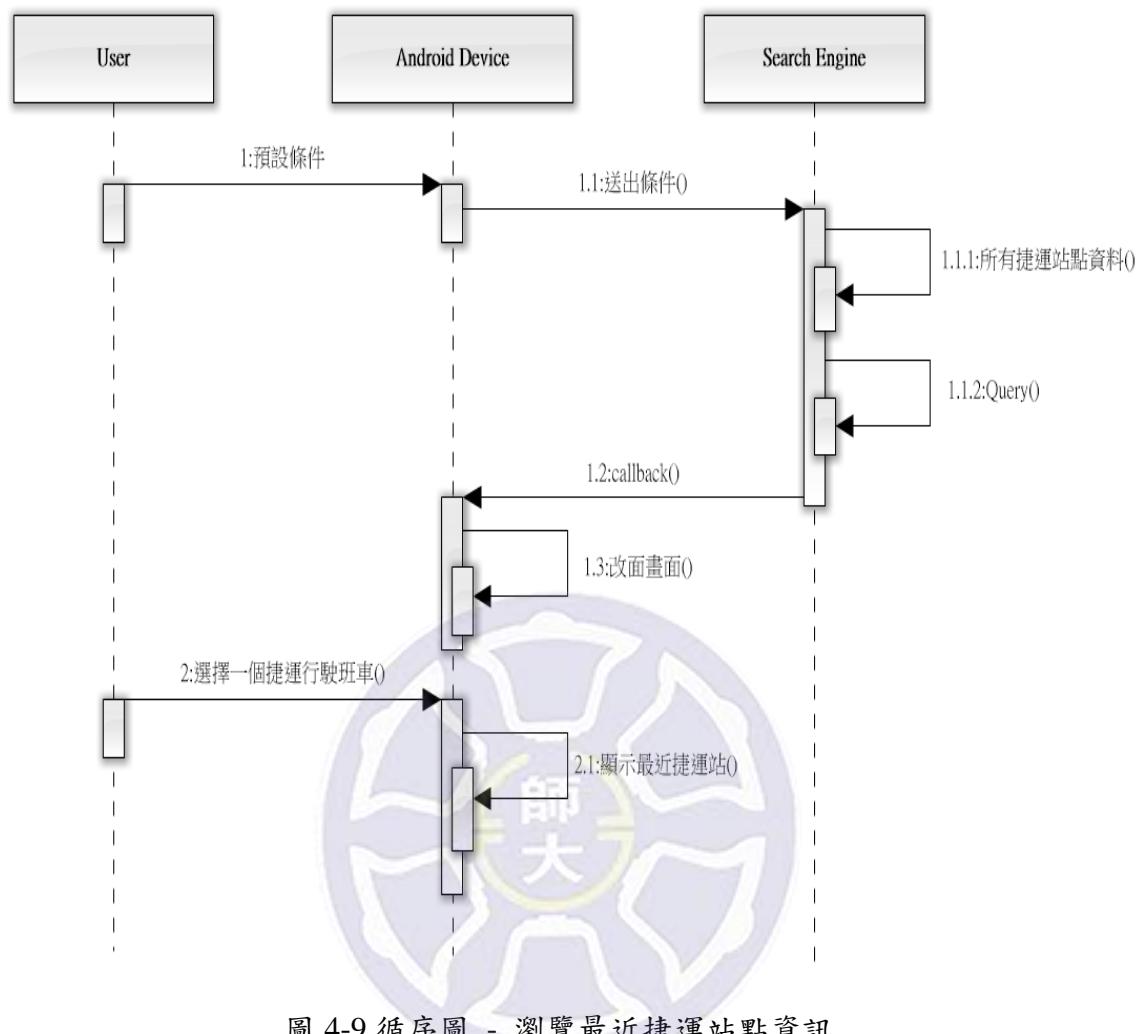


圖 4-9 循序圖 - 瀏覽最近捷運站點資訊

此活動讓使用者進行驅動。使用者首先進入至瀏覽所有捷運行駛列車的畫面，系統將採用預設條件，此條件是『所有範圍』的查詢訊息，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。系統將所查詢所有範圍的捷運站點回傳至 Android Device，顯示在行動裝置的螢幕上。接著使用者可以任意地選擇畫面中任何一個捷運行駛列車資訊進行瀏覽詳細內容的最近捷運站點、列車到站時間、無障礙電梯資訊以及地圖資訊，且能開啟 Google Map 進一步導航使用。

4.3.8 查詢捷運列車到站時間資訊

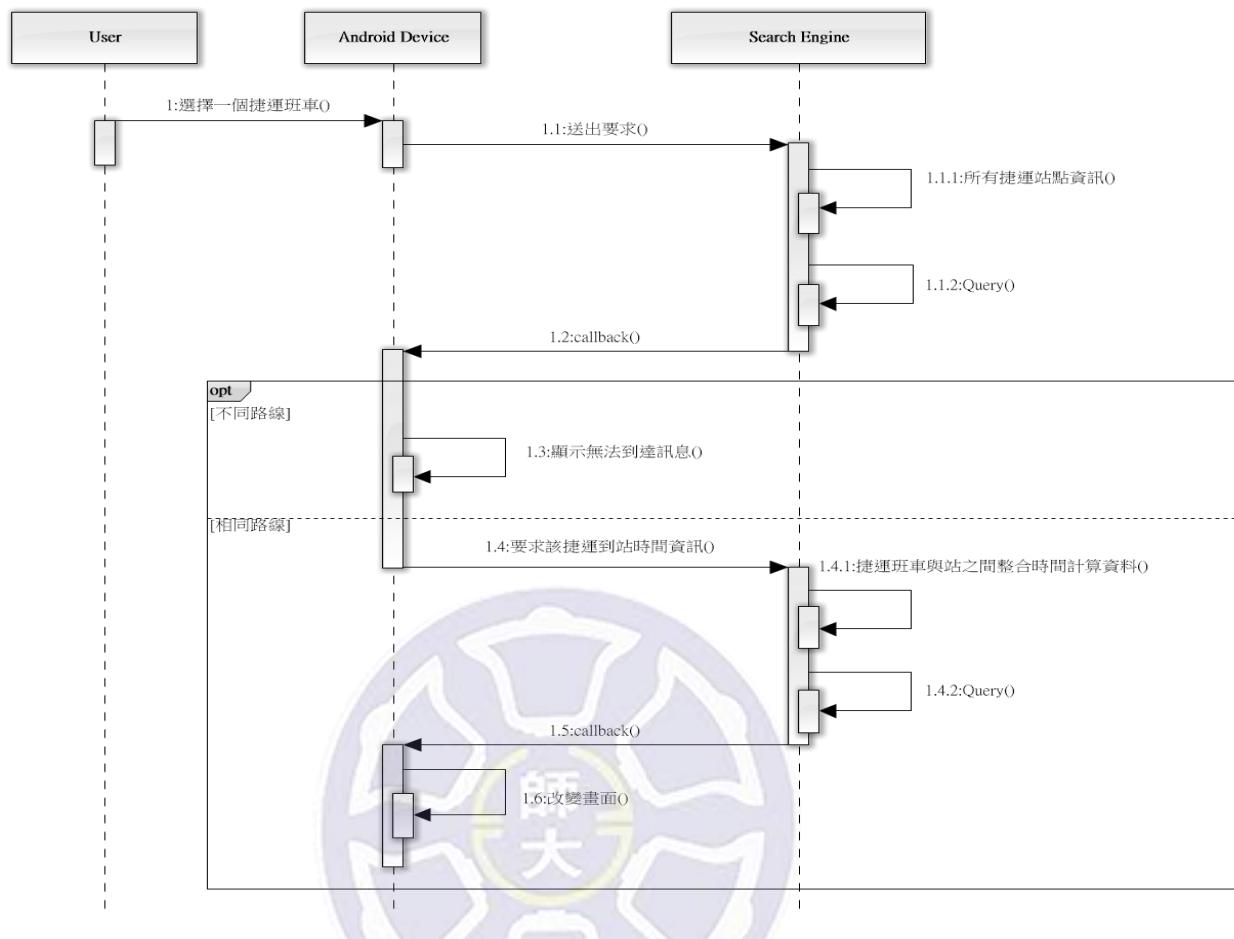


圖 4-10 循序圖 - 查詢捷運列車到站時間資訊

此活動讓使用者進行驅動。使用者首先瀏覽捷運行駛列車介面時，選擇捷運列車列表中任何一個行駛列車來按下時，會先發出要求至 Search Engine，獲得所點選的捷運站點資訊，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。接著判斷使用者目前站點與所點選的捷運列車是否為相同路線，若不同條路線，則會顯示無法到達的訊息，若相同條路線，則會發送「要求該捷運站到站時間資訊」至 Search Engine，拿取此列車經過處理計算的資料回傳至 Android Device，顯示捷運列車到站時間的訊息於螢幕上。

4.3.9 查詢捷運站點票價以及無障礙電梯資訊

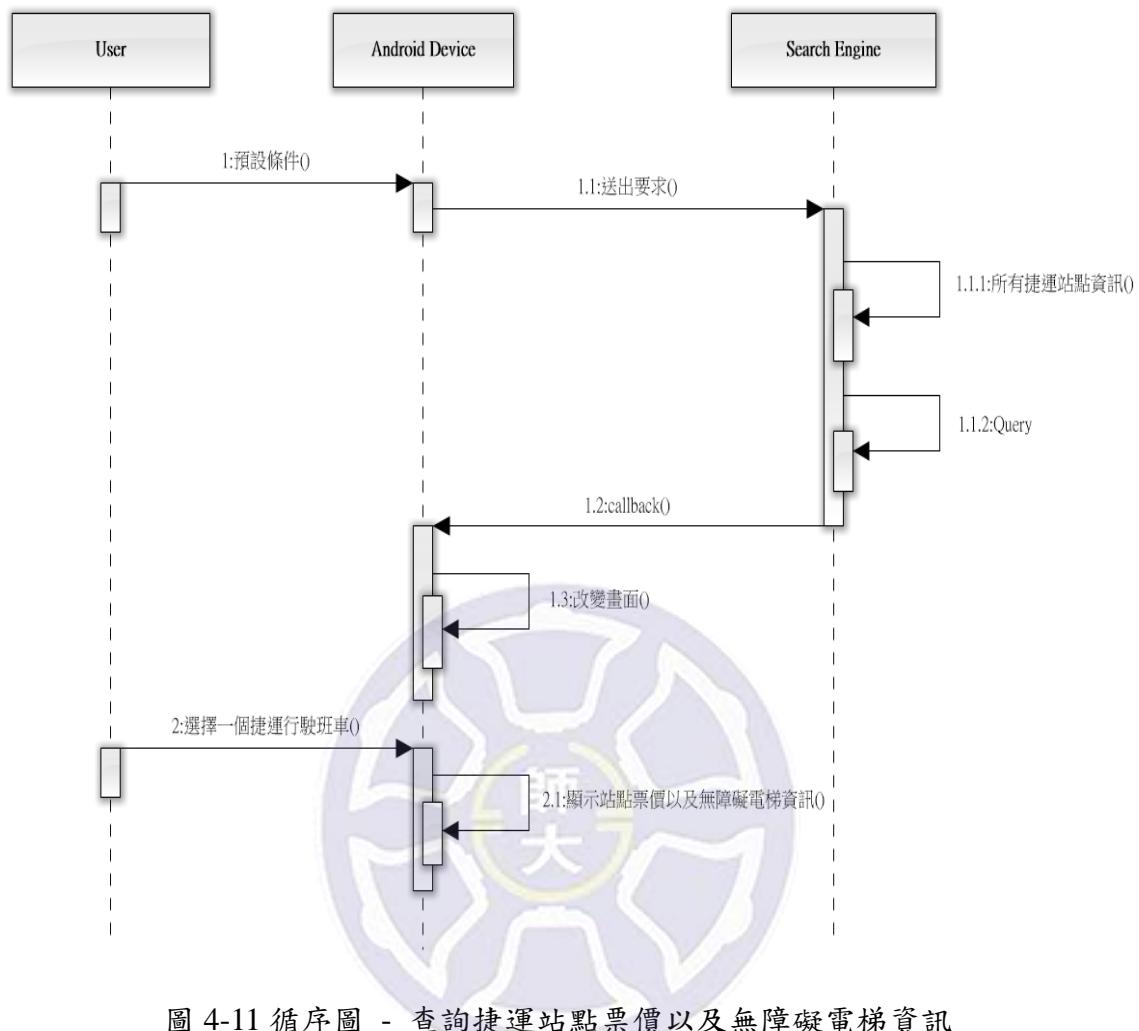


圖 4-11 循序圖 - 查詢捷運站點票價以及無障礙電梯資訊

此活動讓使用者進行驅動。使用者首先進入至瀏覽所有捷運行駛列車的畫面後，系統將採用預設條件，此條件是『所有範圍』的查詢訊息，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。系統將所查詢所有範圍的捷運站點回傳至 Android Device，顯示在行動裝置的螢幕上。接著使用者可以任意地選擇畫面中任何一個捷運行駛列車資訊進行瀏覽該站的詳細內容，其中包含最近捷運站點、列車到站時間、票價查詢、無障礙電梯資訊以及地圖資訊，且能開啟 Google Map 進一步導航使用。

4.3.10 臺北一日遊規劃行程之情境應用

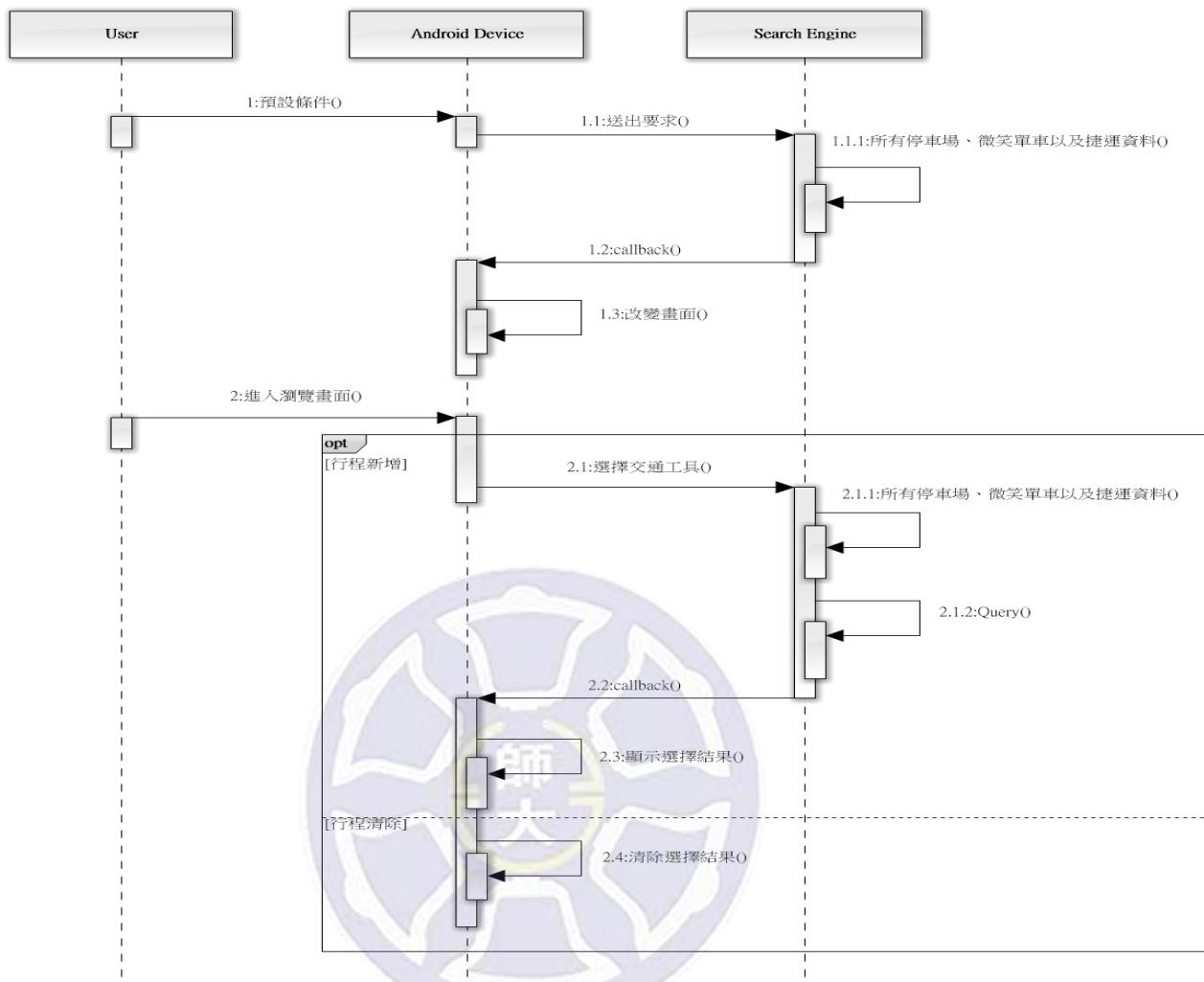


圖 4-12 循序圖 - 一日遊之規劃行程安排

使用者首先進入至瀏覽行程規劃的畫面並選擇交通工具以及輸入地點，當選擇交通工具時系統會採用預設條件，此條件是『所有範圍』的查詢訊息，為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。系統將所查詢所有範圍的停車場、微笑單車以及捷運所有站點資訊回傳至 Android Device 的螢幕上。接著使用者可以任意地選擇畫面中任何一個停車場、微笑單車以及捷運所有站點資訊，地點可以讓使用者自行輸入，完成後會立即顯示一筆資料，此可以顯示多筆資訊，讓行程完整規劃呈現。

4.3.11 停車場至微笑單車或捷運跨資料集應用

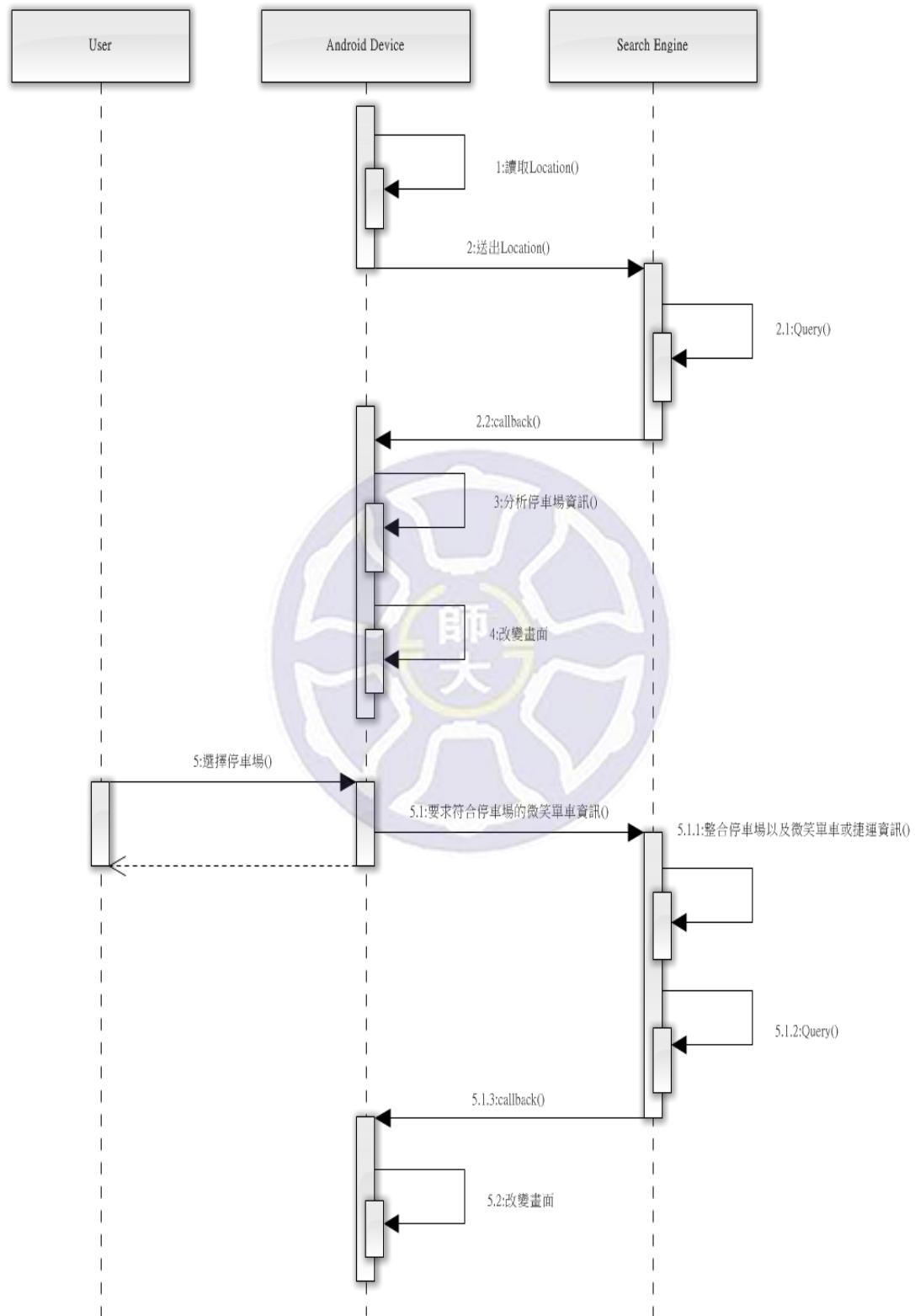


圖 4-13 循序圖 - 停車場至微笑單車跨資料集應用

此活動讓使用者進行驅動。剛開始使用者進入為停車場至微笑單車或捷運跨資料集的應用畫面時，Android Device 將自動搜尋使用者目前的地理環境資訊，Android Device 會把地理環境資訊進行整合且發出「要求所有停車場和所處停車場資訊」的訊息至 Search Engine。為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。當 Search Engine 獲得要求訊息之後，會立即搜尋目前所有停車場和所處停車場資訊，將其結果回傳至 Android Device。收到結果之後，會把所有停車場全部顯示在 Google Map 上，此時 Android Device 也將進一步分析所在停車場，會確認所得到的結果訊息來判斷是否為停車場資訊。若「是」的話，將於行動裝置的畫面呈現所處停車場名稱；若「不是」的話，將統一於行動裝置畫面中呈現「您所處的位置不在停車場」。



使用者能挑選於 Google Map 中任意的停車場或者呈現所處的停車場按鈕，Android Device 將發出「要求符合停車場位置的鄰近的微笑單車或捷運站點」的訊息至 Search Engine。上述也是利用「非同步」的方式來進行查詢。當 Search Engine 接獲要求訊息之後，會把停車場和微笑單車資訊或捷運整合且搜尋符合微笑單車或捷運的站點，將其結果回傳至 Android Device 上，顯示在螢幕上。

4.3.12 縮減範圍搜尋(Reduce Size)

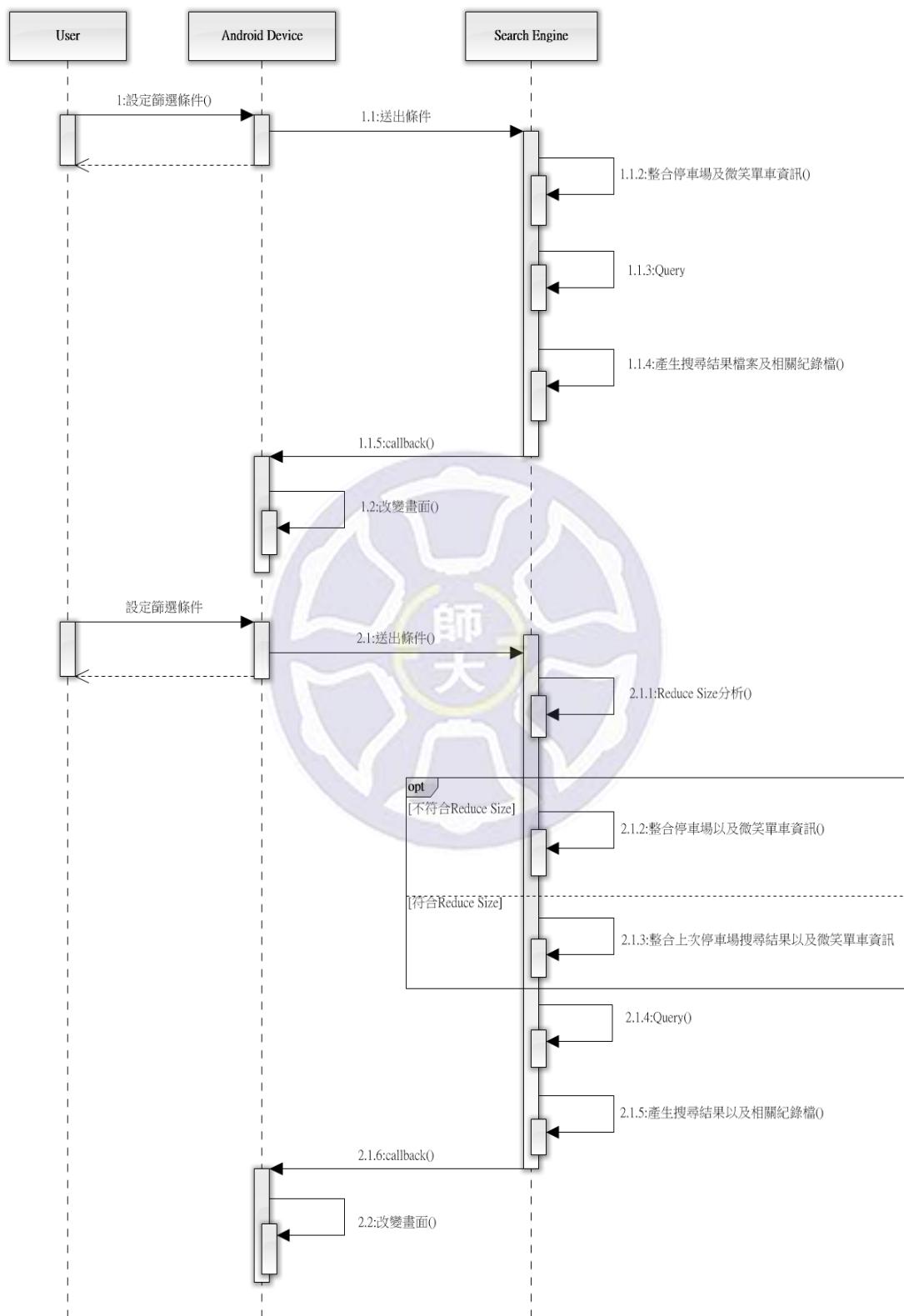


圖 4-14 循序圖 - 縮減範圍搜尋(Reduce Size)

當每次於 Search Engine 搜尋資料時，皆會進行額外 Reduce Size 來分析且整合的動作。使用者首先須自行設定或利用本身預設的條件進一步搜尋相關資料時，Android Device 會發出「要求符合設定微笑單車的站點」訊息至 Search Engine。為了不讓使用者造成不便以及畫面維護的流暢度，故使用「非同步」的方式進一步做資料查詢。當 Search Engine 接獲至要求訊息之後，能進行整合停車場和微笑單車的資訊且搜尋符合條件的資料，而每次搜尋時會同時產生一份該次所搜尋相關紀錄的檔案，目的是提供下次搜尋時候，進行迅速的分析。接著搜尋的結果回傳至 Android Device 來顯示於螢幕上。

若之後再進行搜尋的動作時，Search Engine 皆優先進一步做 Reduce Size 判斷。若有符合 Reduce Size 的規則，在搜尋時會利用先前留下來的相關紀錄檔案來做整合，而不是用預設條件的最大範圍停車場和微笑單車記錄檔案，當整合小範圍的相關紀錄檔案之後，從檔案中挑選其中符合條件的資料來當作搜尋結果。若不符合 Reduce Size 的規則，在搜尋時會利用預設條件的最大範圍停車場和微笑單車記錄檔案做整合，從檔案中挑選其中符合條件的資料來當作搜尋結果。接著搜尋的結果回傳至 Android Device 來顯示於螢幕上。

4.4 實作畫面

4.4.1 瀏覽所有微笑單車站點

當使用者進行瀏覽所有微笑單站點的頁面時，可以選擇任何一個在畫面中所有微笑單車站點的資訊。若使用者按下其中任一個微笑單車站點，能查詢該所選的微笑單車站點呈現的訊息。

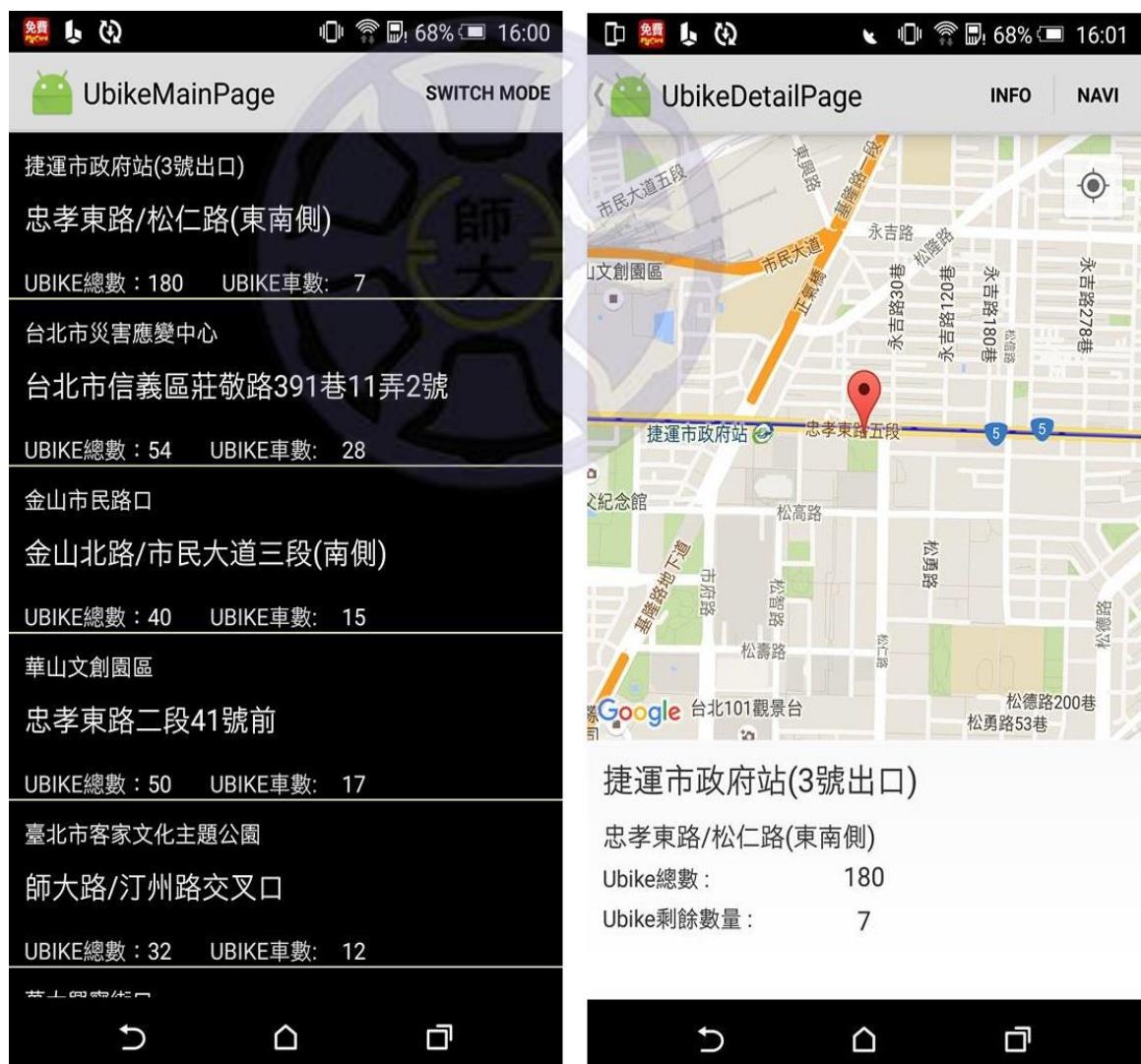


圖 4-15 瀏覽所有微笑單車以及查看站點資訊

若按下右上方的『NAVI』，系統將利用 Google Map 進一步導航。若按下右上方的『Info』，能顯示利用數據所分析的時間軸以及租借或歸還推薦。

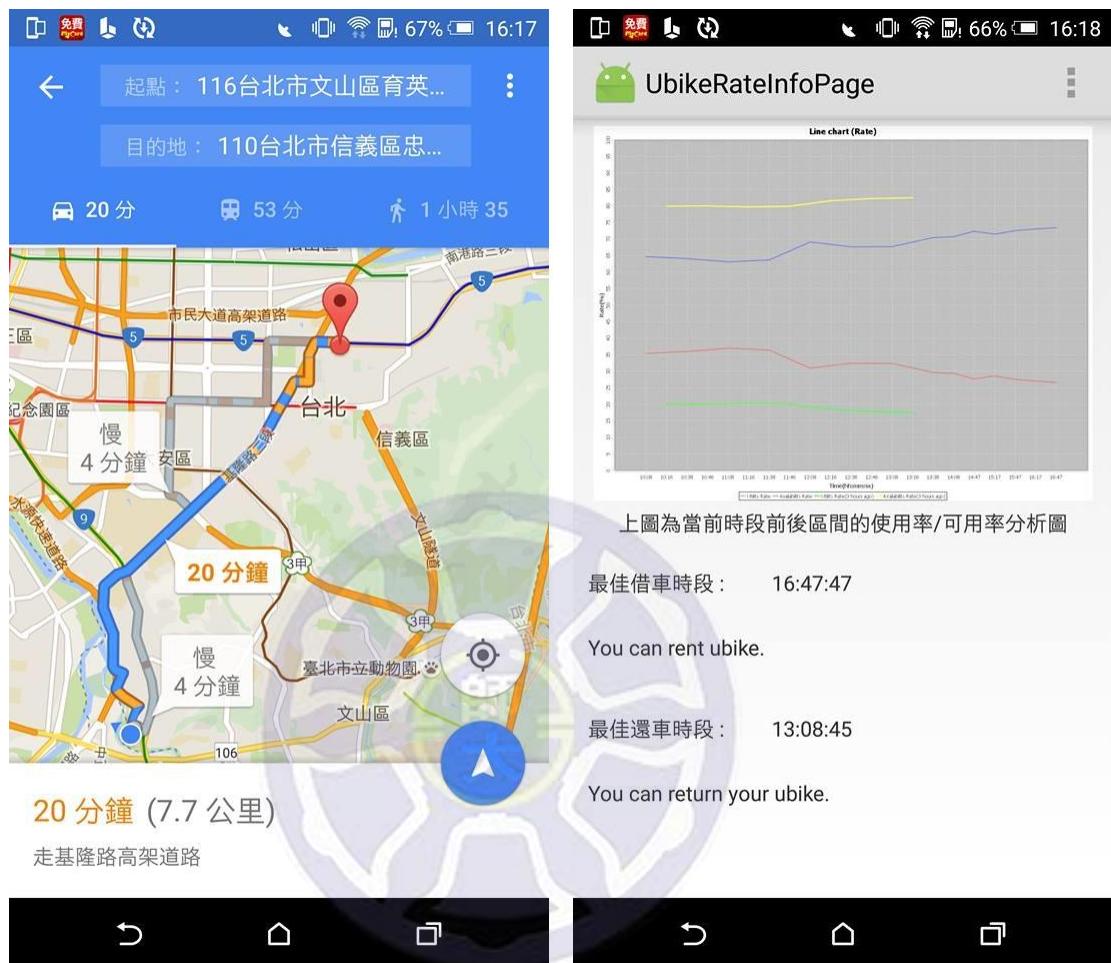


圖 4-16 微笑單車站點之 Google Map 導航和使用狀況與推薦

4.4.2 瀏覽合適微笑單車站點

當使用者進行自訂搜尋範圍的頁面時，輸入所需求的範圍值後，按下『Search』按鈕就能搜尋依照使用者所處的地理位置中周圍的微笑單車站點資訊。

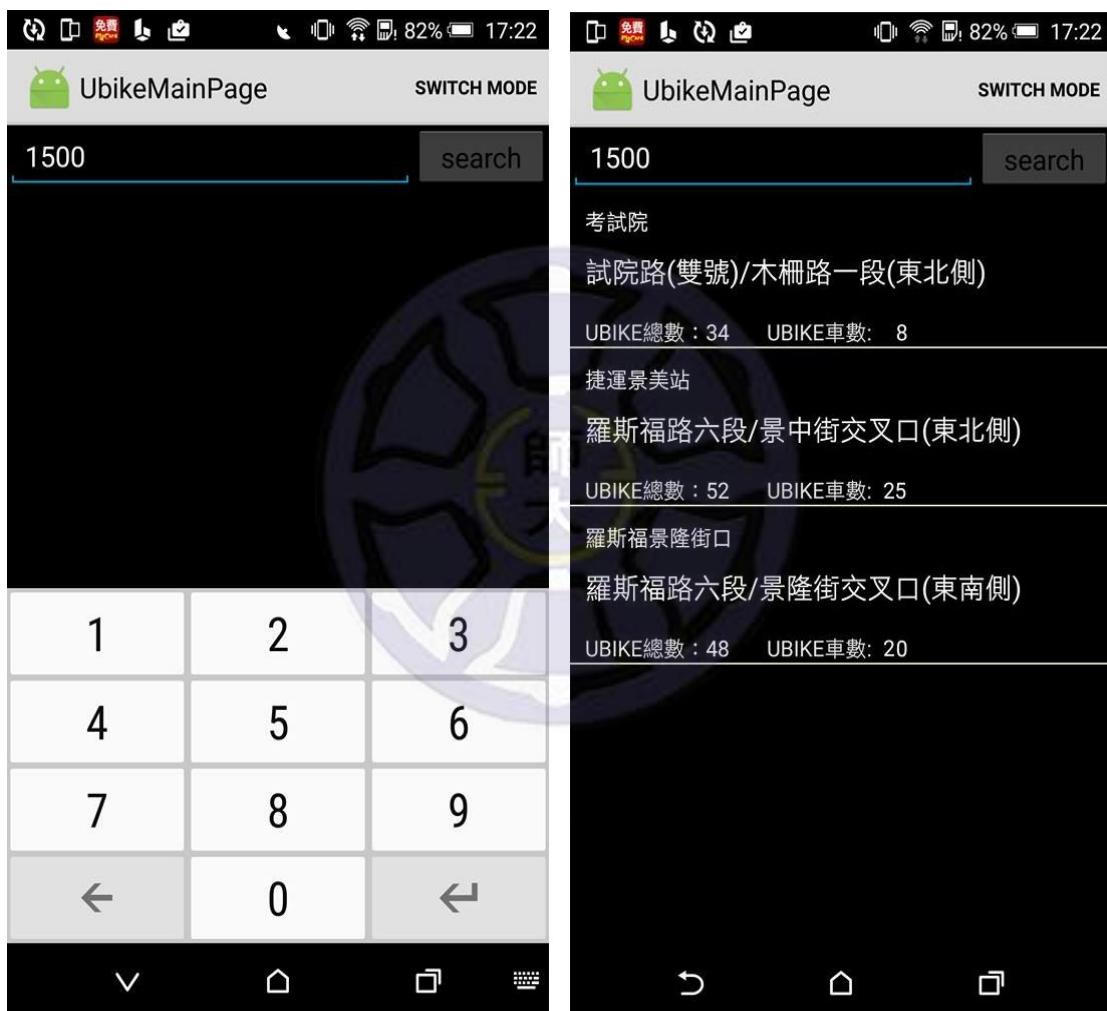


圖 4-17 自訂搜尋範圍頁面以及輸入需求範圍後的搜尋結果

4.4.3 瀏覽、新增、刪除常用站點

當使用者進行瀏覽目前所自行設定常用站點的資訊時，按下右上方的『SWITCH』的按鈕後，會進入 Google Map 中的新增頁面，若按下想要的新增位置並按下右上方『ADD』的按鈕，就能成功新增所欲加入的常用站點。若於常用站點的瀏覽頁面中，只需長按住欲刪除的常用站點，此時會跳出是否刪除的訊息，選擇「是」即可成功刪除常用站點。

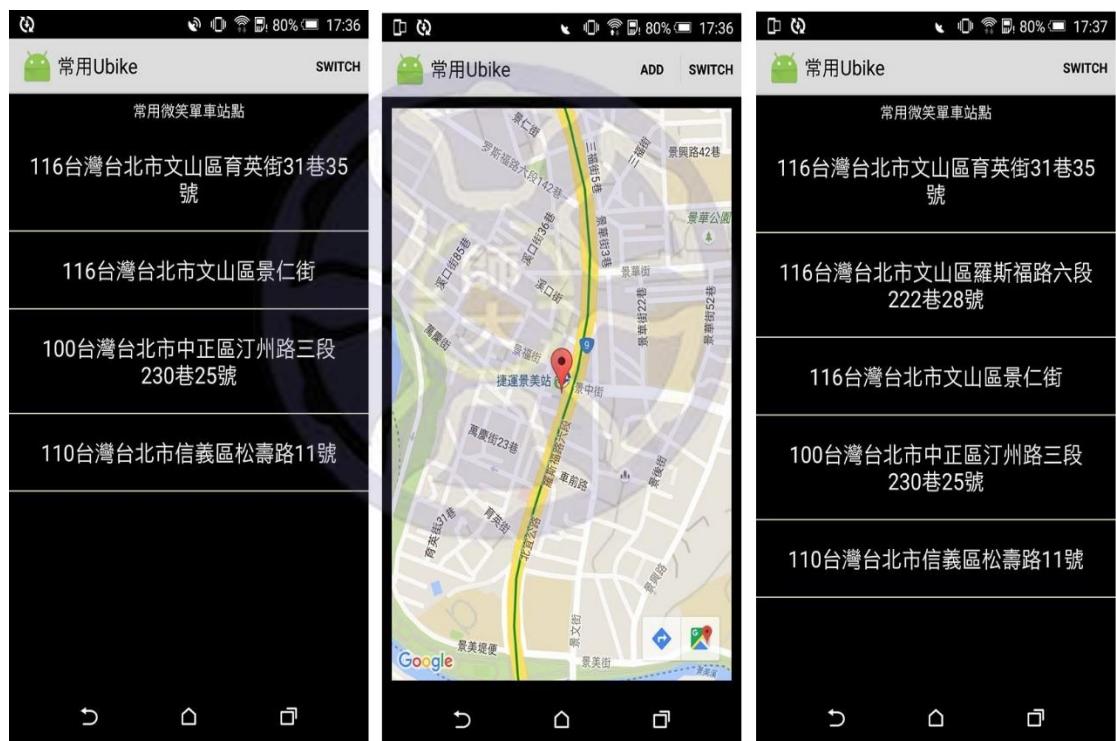


圖 4-18 瀏覽常用站點以及新增常用站點



圖 4-19 刪除常用站點(台北市信義區松壽路 11 號)

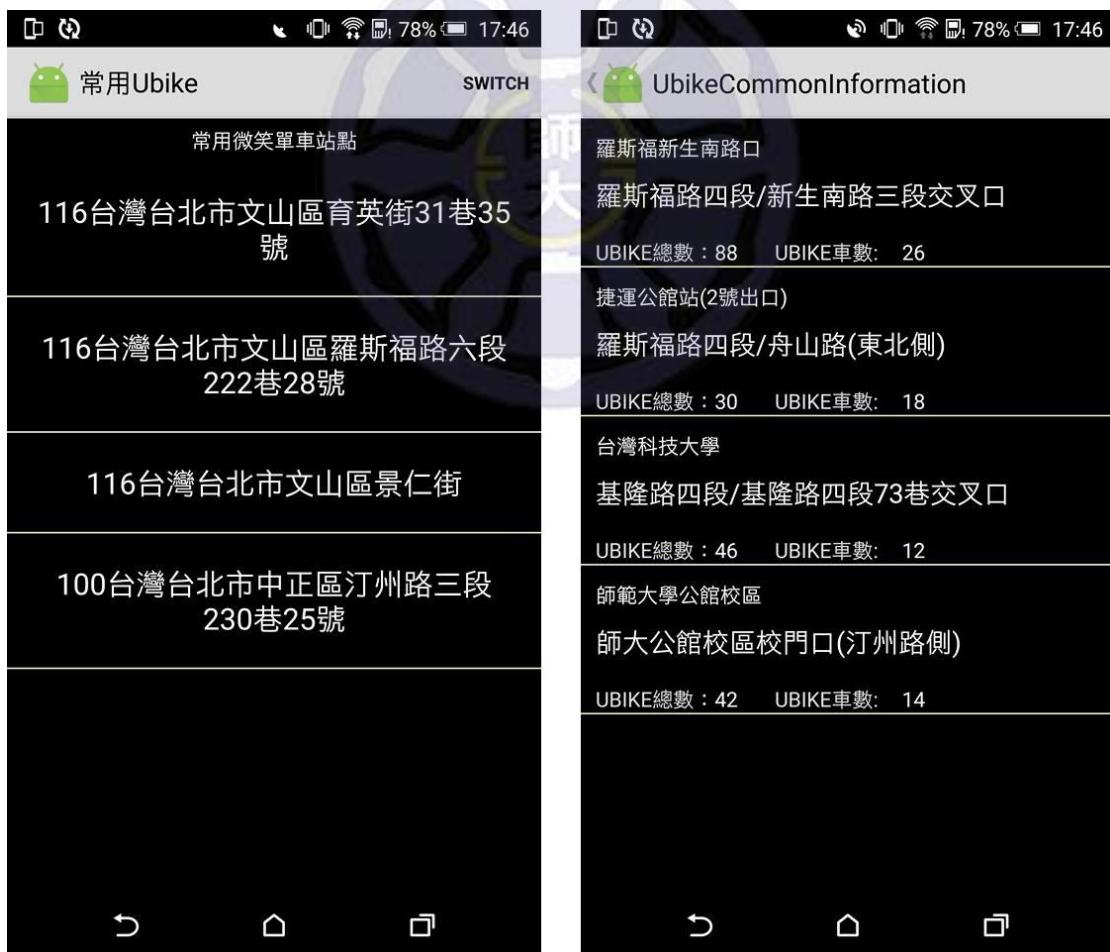


圖 4-20 瀏覽常用微笑單車站點資訊(台北市中正區汀洲路三段 203 巷 25 號)

4.4.4 分析所在停車場並透過停車場尋找周圍微笑單

車或捷運站點(跨資料集應用)

當使用者進行停車場到微笑單車跨資料的頁面時，系統會主動地分析使用者所處地理位置是否為停車場，且將所有臺北市停車場資訊呈現在 Google Map 中，若使用者按下正下方所呈現的停車場按鈕或者呈現在地圖中停車場的 Pin Marker，就能瀏覽以停車場所偵測的範圍內的微笑單車站點資訊；若使用者按下最近捷運站點，則會瀏覽以停車場所偵測的範圍內的捷運站點資訊。



圖 4-21 停車場至微笑單車或捷運跨資料集之應用

4.4.5 瀏覽所有捷運行駛列車資訊

當使用者進行瀏覽所有捷運行駛列車頁面時，可以選擇任何一個在畫面中所有捷運行駛列車的資訊。若欲按下右上方『Update』，則可以每三分鐘更新一次的最即時捷運列車資訊。若使用者按下其中任一個微笑單車站點，能查詢該所選的捷運行駛列車呈現的訊息。



圖 4-22 瀏覽所有捷運行駛列車和更新資訊

4.4.6 瀏覽最近捷運站點資訊

當使用者欲尋找最近捷運站點時，於瀏覽捷運行駛列車中右上方『Near』按鈕，按下去後會根據使用者位置尋找最鄰近的捷運站點，在頁面中正中央的Google Map 會顯示最鄰近捷運站點位置，以下圖範例為：景美，當按下去時會自動幫使用者導航最近站點的路線資訊。

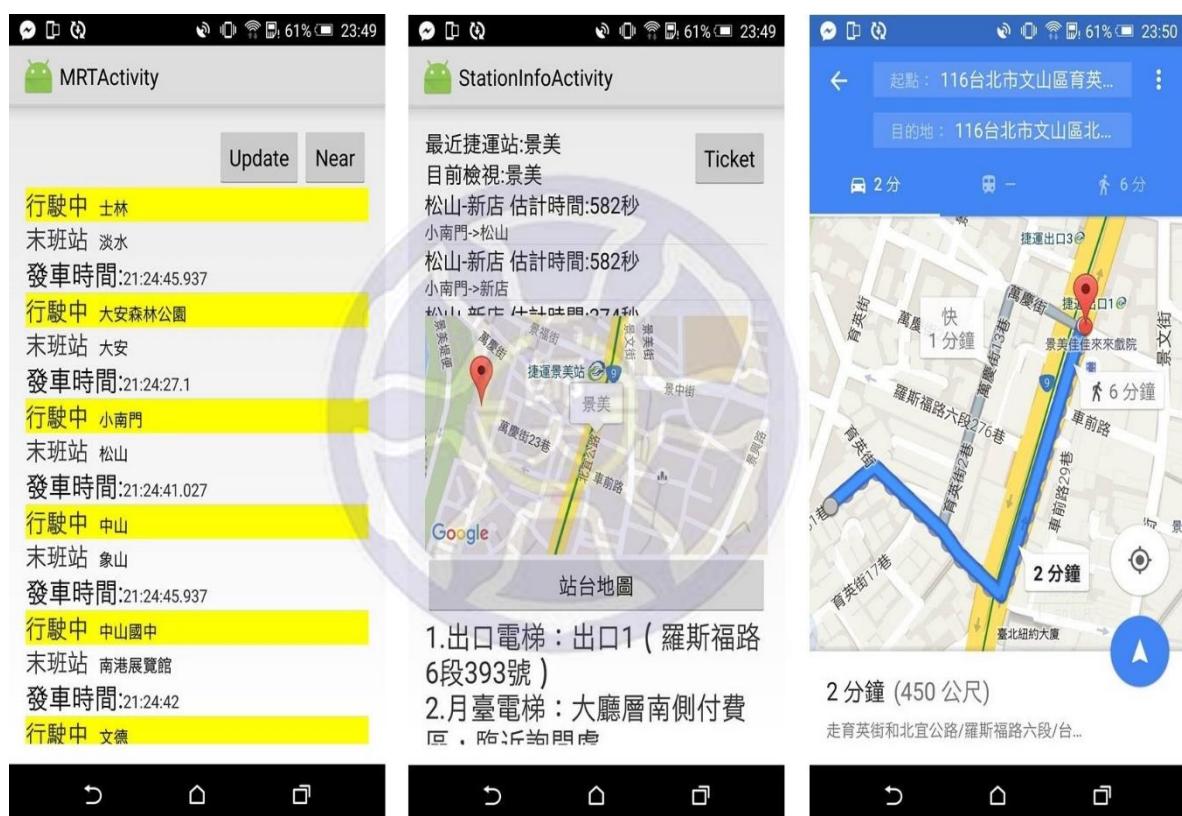


圖 4-23 瀏覽最近捷運站點和 Google Map 導航

4.4.7 查詢捷運列車到站時間

當使用者進行瀏覽所有捷運行駛列車的頁面時，可以選擇任何一個在行駛列車目錄畫面中目前所有的捷運行駛列車資訊。若使用者按下其中任一個捷運行駛列車，則會顯示該列車的到站資訊。另外有分為相同線路以及不同線路，相同線路為使用者所選擇該列車是會到達最近捷運站點的話，會顯示相同線路列車到站時間，而不會顯示不相關的列車；不同線路為使用者所選擇該列車是不會到達最近捷運站點的話，會顯示無法到達的訊息，提醒使用者該列車是不會抵達該鄰近站點。



圖 4-24 查詢捷運列車到站時間(相同路線)

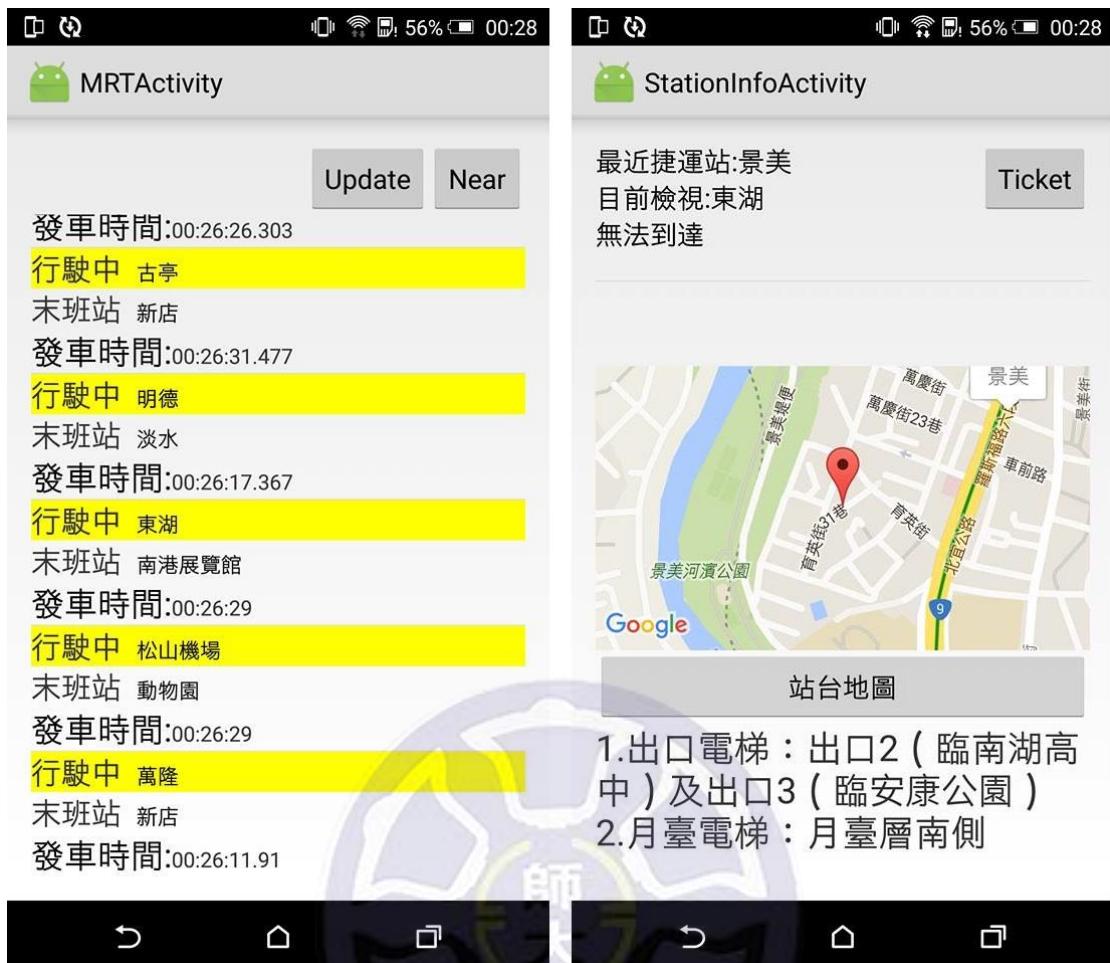


圖 4-25 查詢捷運列車到站時間(不同路線)

4.4.8 查詢捷運站點票價、無障礙電梯以及轉車資訊

當使用者選擇任何一個捷運行駛列車的頁面後，會進入到該捷運行駛列車的相關資訊，其中有顯示該捷運列車的無障礙電梯資訊，方便使用者進行尋找電梯的位置，若欲按下右上方『Ticket』，則可查詢使用者所選擇該捷運列車的票價資訊，提供使用者做為價錢參考。

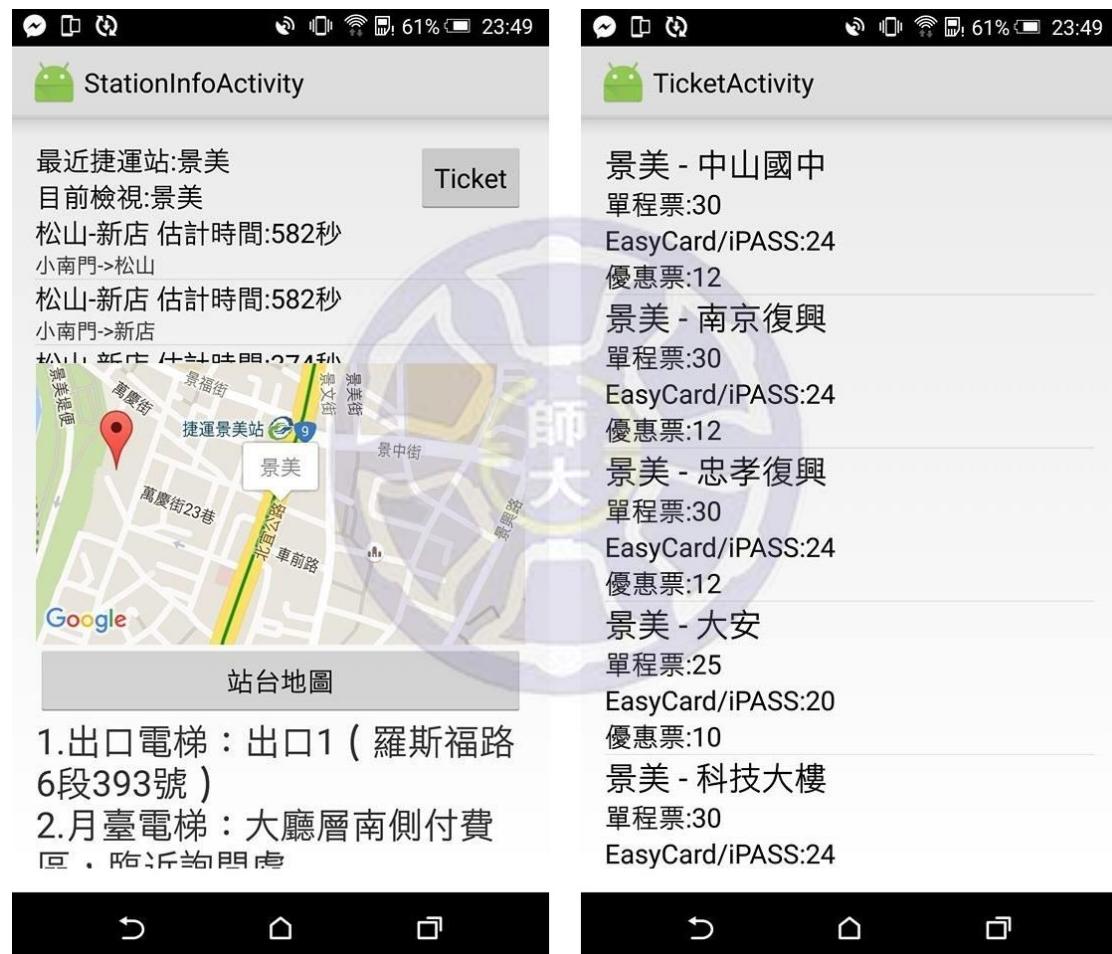


圖 4-26 查詢捷運票價以及無障礙電梯資訊

若欲按下正中央的『站台地圖』按鈕，則會顯示該目前地理位置，其圖中有進行每個捷運站點路線顏色的區別，例如景美站為綠色的線路顏色，則顯示綠色景美站的 mark，能提供使用者做為轉車需要轉到哪條線的依據。另外可以依據使用者所欲到達的捷運站點進行導航路線。

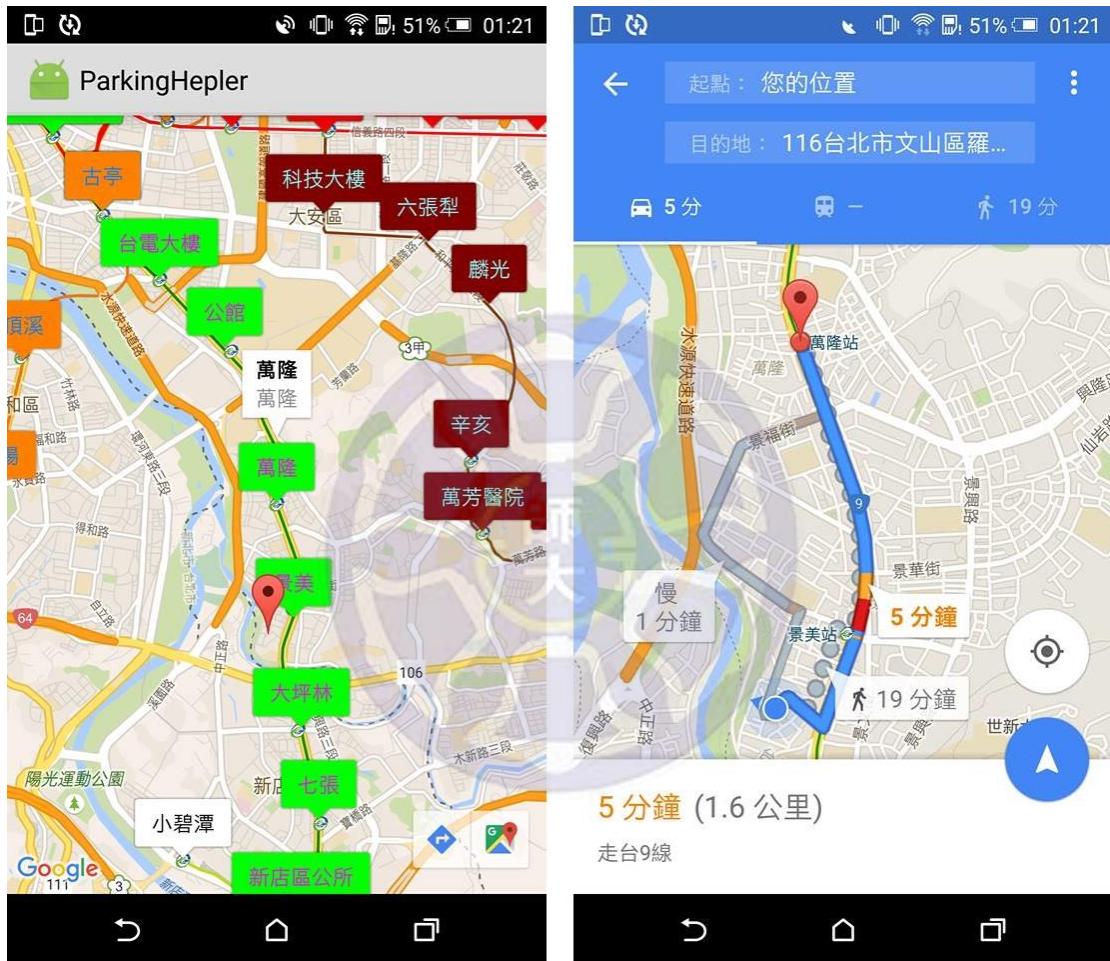


圖 4-27 查詢捷運轉車以及 Google 導航站點資訊

4.4.9 臺北一日遊規劃行程之情境應用

當使用者進行瀏覽規劃行程的頁面時，輸入兩筆資料進行規劃分為：地點、交通工具，地點為讓使用者自行輸入文字想去的目的地，而交通工具會呈現出所有停車場、Ubike 以及 MRT，讓使用者做選擇規劃，則會根據使用者位置進行最近的停車場、Ubike 以及 MRT 各個站點資訊做優先排序，若輸入非單一筆，則由最末筆資料進行再次優先排序。



圖 4-28 查詢所有停車場、Ubike 以及捷運資訊以及輸入行程

若輸入多筆資料完成後，系統會顯示完整規劃路線流程於螢幕中，並搭配運用何種交通工具到達下個目的地，同時會給予使用者所選擇停車場、Ubike 以及 MRT 的地點資訊；若按下左上方『檢視地圖』按鈕，則會顯示所選擇停車場、Ubike 以及 MRT 的標誌並根據選擇的優先順序劃上直線於 Google Map 上；若按下正上方『清除』按鈕，則會清除規劃行程的資訊做重新的規劃；若按下右上方『完成』按鈕，則系統會紀錄於 App 中做下次的查詢依據並回到首頁。

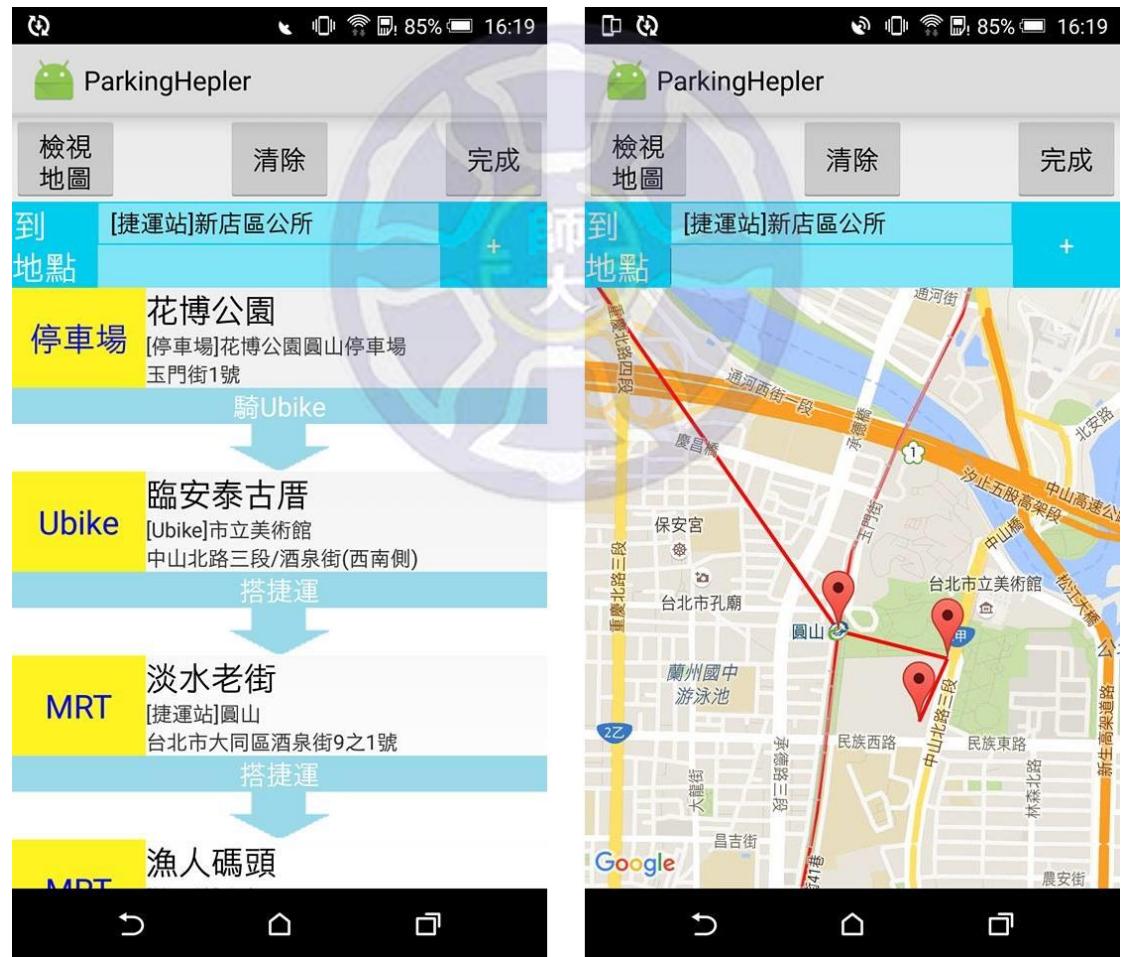


圖 4-29 規劃行程顯示圖以及檢視視覺化地圖

第五章 結論與未來發展

5.1 結論

本研究利用基於 W3C 所規定的鏈結開放資料做為分類鑑定準則的標準，讓雜亂無章的臺北市政府開放資料具備統一且知識化的檔案格式，使得資料在使用上更具靈活性。從不同種類以及性質的開放資料利用 RDF 進行描述，並建構出 Linked Open Data Diagram，好處是將原本雜亂的資料統一整合格式，開發者或使用者能容易從 RDF 格式檔案中拿取或者觀看所需求的資訊，進一步利用彼此鏈結特性，讓資料應用更加廣闊。於研究中依據『地理資訊』做為樞紐，讓停車場、微笑單車以及捷運的資訊彼此互相鏈結關係，實現跨資料集搜尋的應用。根據鏈結開放資料所分類鑑定準則，利用 RDF 知識正規化格式進一步描述於開放資料，將臺北市開放資料的水準由原本的三星等級(使用非受權資料格式，例：CSV、JSON...等)提升至四星等級(利用 W3C 制定標準格式描述資料，例：RDF)。研究中將停車場、微笑單車以及捷運資料透過 RDF 技術提升至四星等級，再把其資料利用地理環境資訊進行資料彼此鏈結，以此開放資料的水準更提升至五星等級(透過資料鏈結方式將資料彼此鏈結)。

過去在網路中所獲得的資料焦點皆放在資料呈現上，較少著重於「資料本身」，但隨著在近幾年 Big Data 議題盛行，大家漸漸重視資料本身中所蘊含的潛在力，而卻遇到「格式不一」和「資料散亂」的窘境。Linked Data 所給予的概

念是一個既有效又能解決資料雜亂格式且重新正規化的方法，讓資料格式統一，並利用鏈結特性能把資料與其他更多的資訊進行連接。現今全世界的資料處於爆炸的局面，但藉由資料的統整，可以讓開發者或民眾提供更方便查閱、挖掘資料，激發出資料本身的潛在力。開發者更可以利用現有的數據創造出各種加值的應用及服務，民眾也能利用簡單的文件明白政府目前的施政數據來達到監督效果，讓社會可以更加進步。

此外研究中也探討情境感知應用以及情境應用。情境感知應用以行動裝置作為載具，給予 Time-based 和 Location-based 的情境感知應用。現今科技不斷的創新求變，已逐漸達到『人手一機』的環境，行動裝置本身具有的運算功能也越做越強大，而開發者也應該隨著使用者角度做為延伸思考，研究中藉由了解使用者的情境資訊，提供對應符合情境的服務應用，讓應用更加智慧與貼近人們。而情境應用以模擬真實作為情境，給予臺北市開放資料的大眾工具做為輔助，配合使用者安排一套行程規劃的旅遊行程加以融入，可以根據獲得的即時資訊，讓使用者在選擇方面上能更加靈活與客製化，讓人們安排行程更加完美。

5.2 未來發展

本論文利用臺北市政府公開資料所提供的部分資料(包括停車場資訊、停車場動態格位、微笑單車資訊以及捷運資訊)，使用鏈結資料的方法建立 Linked Data Diagram，於系統中開發出 RDF Interface，提供應用程式能有效地利用此 Interface 與資料進一步互動作為開發平台。現今所使用的資料種類與資料個數雖然不是特別多種，但是已經足夠能充分展現出 RDF 知識正規化以及對資料彼此間建立 Linked Open Data 所帶出來的力量和潛在力，例如：資料存取更加方便、資料格式統一原本雜亂無章的現況...等等。未來期許能將更多臺北市政府開放平台所提供的資料進一步使用正規化，並建立出更完整且龐大的臺北市 Linked Data Diagram，甚至可以達到對現今 Wikipedia 所提供龐大的「DBpedia」進行彼此之間的鏈結，增強資料的潛在力。研究中目前只針對臺北市政府開放平台中部分資料進行 RDF 知識正規化且做鏈結的操作，未來希望將臺北市政府能與其他縣市政府所提供的開放資料進行彼此之間鏈結，造就出臺灣政府的鏈結開放資料集，達成一個全國開放資料的資料網路環境。

在研究中所提到的情境感知的應用也是未來幾年發展重點之一，論文中利用到的種情境感知功能，其實智慧型裝置內所蘊含的情境感知功能並不僅僅如此，還有許多以使用者活動中情境感知為基礎的應用尚未開發完全，期許藉由科技推陳出新的智慧型穿戴以及智慧型手提裝置的強大性能，讓開發發揮無限想像力和創造力，提供更多元的面向來貼近使用者的情境感知應用服務。

參考文獻

- [1] 蘇文彬, “Google：台灣智慧型手機普及率已達 51%,” iThome, 2013.
- [2] 臺北市政府, “臺北市政府資料開放平台,” 8 2014. [線上]. Available:
<http://data.taipei/>. [存取日期: 28 2 2016].
- [3] 國家發展委員會, “政府資料開放平台,” 29 4 2013. [線上]. Available:
<http://data.gov.tw/>. [存取日期: 28 2 2016].
- [4] 國家發展委員會, “政府資料開放平台 - 關於我們,” [線上]. Available:
<http://data.gov.tw/about>. [存取日期: 20 12 2016].
- [5] 王梅玲, “資源描述架構/資源描述綱要,” 4 2011. [線上]. Available:
<http://techserviceslibrary.blogspot.tw/2011/04/rdf-resource-description.html>. [存
取日期 10 4 2016].
- [6] 郭子瑜, “無所不在個人化情境感知服務,” 數位典藏與學習電子報, 2011.
- [7] 張書銘, “基於開放資訊之情境感知系統,” 國立師範大學碩士論文, 2013.
- [8] 馬聖儒, “連結開放資料與其情境感知應用,” 國立師範大學碩士論文, 2014.
- [9] 江凱逸 “連結開放資料結合情境感知及分析應用之研究,” 國立師範大學碩
士論文, 2015.
- [10] 彭彥瑋, “情境感知 (環境感知 Context-Awareness),” 7 11 2010. [線上].
Available: http://e-learning-101.blogspot.tw/2010/11/blog-post_07.html. [存取日
期: 25 12 2015].
- [11] 國立虎尾科技大學資工, 雲端軟體實驗室[線上]. Available:
<http://nfuxml.blogspot.tw/2012/11/jena.html> [存取日期: 4 3 2016]

- [12] M. So, “調查指出三分之二的美國人都有智慧型手機,” Engadget, 2014.
- [27] Ipsos MediaCT, Mobile Marketing Association & Interactive Advertising Bureau, "Data Sets and Country Reports," Our Mobile Planet, 2013.
- [13] Wikipedia, “開放資料,” [線上]. Available:
<http://zh.wikipedia.org/wiki/%E9%96%8B%E6%94%BE%E8%B3%87%E6%96%99>. [存取日期: 25 12 2015].
- [14] Wikipedia, “台北市公共自行車租賃系統,” 6 2015. [線上]. Available:
<http://zh.wikipedia.org/wiki/%E8%87%BA%E5%8C%97%E5%B8%82%E5%85%AC%E5%85%B1%E8%87%AA%E8%A1%8C%E8%BB%8A%E7%A7%9F%E8%B3%83%E7%B3%BB%E7%B5%B1>. [存取日期: 20 12 2015]
- [15] Wikipedia, “開放資料,” [線上]. Available:
<http://zh.wikipedia.org/wiki/%E9%96%8B%E6%94%BE%E8%B3%87%E6%96%99>. [存取日期: 25 12 2015].
- [16] Wikipedia, “網路本體語言,” 7 2015. [線上]. Available:
<https://zh.wikipedia.org/wiki/%E7%BD%91%E7%BB%9C%E6%9C%AC%E4%BD%93%E8%AF%AD%E8%A8%80>. [存取日期: 21 2 2016].
- [17] Wikipedia, “資源描述框架,” 2 2016. [線上]. Available:
<http://zh.wikipedia.org/wiki/%E8%B3%87%E6%BA%90%E6%8F%8F%E8%BF%B0%E6%A1%86%E6%9E%B6>. [存取日期 11 3 2016].
- [18] Wikipedia, “DBpedia,” 12 2015. [線上]. Available:
<http://zh.wikipedia.org/wiki/DBpedia>. [存取日期: 25 12 2015].
- [19] Wikipedia, “Linked Data,” 3 2016. [線上]. Available:
http://en.wikipedia.org/wiki/Linked_data. [存取日期: 20 3 2016].

- [20] “Design Issues-Linked Data,” 6 2009. [線上]. Available:
<http://www.w3.org/DesignIssues/LinkedData.html>. [存取日期: 3 2 2016].
- [21] Wikipedia, “Google API,” [線上]. Available:
http://zh.wikipedia.org/wiki/Google_API. [存取日期: 3 3 2016].
- [22] Wikipedia, “PhpMyAdmin” [線上]. Available:
<https://zh.wikipedia.org/wiki/PhpMyAdmin>. [存取日期: 15 3 2016]
- [23] Linked Data-Connect Distribute Data across the Web.
[線上]. Available: <http://linkeddata.org/>. [存取日期: 4 4 2016].
- [24] Five Star Scheme for linked Open Data[線上]. Available:
https://www.researchgate.net/figure/281349915_fig4_Fig-6-Five-Star-Scheme-for-Linked-Open-Data-Source-5stardatainfo. [存取日期: 10 3 2016]
- [25] B. Schilit, N. Adams, and R. Want., “Context-aware computing Applications,” from IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, US., 1994.
- [26] Schilit, B.N. and Theimer, M.M., “Disseminating Active Map Information to Mobile Hosts,” form IEEE Network 8, 1994.
- [27] W3C(e-Gov), “eGovernment at W3C - Better Government Through Better Use of the Web.,” [線上]. Available: <http://www.w3.org/egov/>. [存取日期: 10 3 2016].
- [28] T. R. Gruber, “Toward Principles for the Design of Ontologies,” 於 International Journal Human-Computer Studies., 1993.
- [29] JRG Pulido, SBF Flores, RCM Ramírez and RA Díaz, "Eliciting ontology components from semantic specific-domain maps: towards the next generation

- web," in Web Congress, 2009. LA-WEB '09. Latin American, 2009.
- [30] Alexandre Lopes Machado and Jos é Maria Parente de Oliveira, "DIGO: An Open Data Architecture for e-Government," in Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International, 2011.
- [31] Julia Hoxha and Armand Brahaj, "Open Government Data on the Web: A Semantic Approach," in Emerging Intelligent Data and Web Technologies (EIDWT), 2011 International Conference on, 2011.
- [32] Ralph Löwe, Peter Mandl and Michael Weber, "Context Directory: A Context-Aware Service for Mobile Context-Aware Computing Applications by the Example of Google Android," in Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on, 2012.
- [33] C. M. Chan, "From Open Data to Open Innovation Strategies: Creating E-Services Using Open Government Data," in System Sciences (HICSS), 2013 46th Hawaii International Conference on, 2013.
- [34] T. W. W. W. Consortium, “OWL Web Ontology Language,” 10 2 2004. [線上]. Available: <http://www.w3.org/TR/2004/REC-owl-features-20040210/>. [存取日期: 25 12 2015].