

逢 甲 大 學

資訊工程學系

專題研究報告

智慧型導覽機器人

指導教授：張貴忠 老師

學生：柯權哲 D0372672

邱皇旗 D0381806

吳啟民 D0381778

郭怡均 D0308939

中 華 民 國 一 百 零 六 年 十 二 月

致謝

從專題想法的建構，到專題的完成，真的很不容易。首先我們要感謝指導教授張貴忠老師，專題的題目到內容，歷經了無數的開會研討，老師總是很有耐心的指導及建議，在專題實作的過程更是給予諸多提點，才得以讓我們在一開始確立專題方向並一步步的完成系統。同時也要特別感謝儒錦，在 Android 方面以豐富的經驗給了非常多指導，幫助我們更快完成系統；還要感謝柏邑，由於我們建構車體的部分，需要裁切厚木板，謝謝你和我們一起討論並且幫助我們分切木板；以及感謝承儒，幫助我們解決在數據處理上的問題，最後感謝各位專題組員犧牲時間一起完成這份專題。

最後還要再次感謝貴忠老師，因為有老師不斷地引導，我們的專題才得以順利完成。

摘要

導覽的定義是「引導觀眾,藉由一種有意的安排,來實行引導活動,經由導覽人員來進行,以達到某種教育計畫性的目的」,而近年來,有越來越多的大型展覽活動及校園活動,且只要有人的地方,就會有服務的需求,因此展覽及活動的工作人員需求量也會相對增加,加上科技日益進步,機器人也逐漸興起,於是本專題將以逢甲大學為例,研發一個實體的導覽機器人系統,主要目的為創新校園的導覽服務,希望能夠減少以往活動時消耗的勞力資源,也能促使使用者親自與機器人進行互動,透過導覽機器人來為使用者進行不同需求的導覽服務,如參加校園活動、參訪、來校參加研討會...等等不同活動,除能提高服務品質外,更能將多餘的人力應用於其他需求。本專題的導覽機器人之系統架構,含有行控、通訊、導航及導覽四大模組來進行開發,最後整合上述軟體、硬體完成一套完整的導覽系統,使其能提供如導引解說、導覽互動等實體導覽功能。

關鍵字:機器人、導覽、導航、系統整合、Raspberry Pi3

目錄

摘要.....	3
目錄.....	4
圖目錄.....	7
Chapter 1 緒論	10
1.1 研究動機.....	10
1.2 研究目的.....	10
1.3 研究方法.....	11
1.4 研究流程.....	11
1.5 工作分配及甘特圖.....	12
Chapter 2 系統開發平台	13
2.1 Android Studio.....	13
2.2 Raspberry Pi 3.....	13
2.2.1 Ubuntu MATE 16.04 LTS	13
2.2.2 Eclipse.....	14
2.3 VSCode	15
Chapter 3 文獻回顧及探討	16
3.1 Bluetooth.....	16
3.2 Android	16
3.2.1 Android	16
3.2.2 Google Map API.....	18
3.2.3 Webview	18
3.3 Raspberry Pi.....	19
3.3.1 Raspberry Pi3.....	19
3.3.2 PWM.....	20
3.3.3 HC-SR04.....	21
3.3.4 L298N	21
3.4 Web	22
3.4.1 Responsive Web Design	22
3.4.2 Bootstrap 4 Beta.....	22
3.4.3 SPA	23
3.4.4 JQuery	24
3.4.5 語音合成.....	24
3.4.6 Restful API.....	25
3.4.7 Nodejs	25
3.5 資料庫.....	26
3.5.1 MongoDB	26

3.5.2 No-SQL.....	27
3.5.3 導覽系統.....	27
Chapter 4 系統分析與設計	28
4.1 系統設計.....	28
4.2 系統架構圖.....	28
4.3 系統流程圖.....	29
4.4 使用者與系統互動模式.....	30
4.5 Android App	30
4.5.1 Android 架構	30
4.5.2 Singleton Design Pattern.....	31
4.5.3 Bluetooth Socket.....	32
4.5.4 Sensor Data	32
4.5.5 過濾修正.....	32
4.5.6 Webview 溝通	33
4.6 Raspberry Pi3.....	33
4.6.1 Bluetooth Socket.....	33
4.6.2 Car Control.....	34
4.7 Web.....	34
4.7.1 前端.....	35
4.7.2 後端.....	37
4.8 資料庫.....	38
4.8.1 MongoDB	38
4.8.2 校園節點.....	38
4.9 路徑規劃演算法.....	39
4.9.1 路徑規劃.....	40
4.9.2 節點判斷法.....	41
4.10 角度修正演算法.....	41
4.10.1 方向感測器修正.....	42
4.10.2 磁力及重力加速度修正.....	43
4.11 自動避障.....	45
Chapter 5 系統實作成果	46
5.1 系統功能與介紹.....	46
5.1.1 歡迎頁面.....	46
5.1.2 導覽頁面.....	46
5.1.2.1. 學院、大樓、景點介紹.....	47
5.1.2.2. 校園訊息.....	48
5.1.2.3. 導覽介面(Google Map Dialog).....	50
5.1.3 車體結構設計.....	51

Chapter 6	結論	53
6.1	結論	53
6.2	未來展望	53
相關文獻		54

圖目錄

圖 1.1 系統流程圖.....	11
圖 2.1 Android Studio.....	13
圖 2.2 Ubuntu MATE 16.04 LTS	14
圖 2.3 Eclipse.....	14
圖 2.4 VS Code	15
圖 3.1 Bluetooth.....	16
圖 3.2 Android 系統架構圖	17
圖 3.3 Google Maps APIs.....	18
圖 3.4 WebView	19
圖 3.5 Raspberry Pi3.....	19
圖 3.6 BCM43438.....	20
圖 3.7 PWM.....	20
圖 3.8 HC-SR04.....	21
圖 3.9 超音波行進/回彈示意圖	21
圖 3.10 L298N	21
圖 3.11 RWD 示意圖	22
圖 3.12 Bootstrap	23
圖 3.13 傳統型和 SPA 型網頁分析圖	23
圖 3.14 jQuery 和 JavaScript 關係圖.....	24
圖 3.15 語音合成.....	24
圖 3.16 RESTful API.....	25
圖 3.17 Nodejs	26
圖 3.18 MongoDB	26
圖 4.1 系統架構圖.....	28
圖 4.2 系統流程圖.....	29
圖 4.3 使用者與系統互動模式圖.....	30
圖 4.4 Android 架構圖	30
圖 4.5 Singleton 概念圖	31
圖 4.6 Android Bluetooth 通訊概念圖	31
圖 4.7 校園磁力分布圖.....	33
圖 4.8 前後端溝通示意圖.....	34
圖 4.9 Client 與後端溝通架構圖	35
圖 4.10 Web 前端三大語言	35
圖 4.11 網頁內容變更示意圖.....	36

圖 4.12 JQuery 運作示意圖	36
圖 4.13Mongo DB 結構設計	38
圖 4.14 校園節點分佈圖.....	39
圖 4.15 貪婪演算法.....	39
圖 4.16 路徑規劃圖.....	40
圖 4.17 角度計算及修正.....	41
圖 4.18 三軸方向感測.....	42
圖 4.19 方向感測數值修正.....	42
圖 4.20 目標節點判斷.....	43
圖 4.21 三軸加速度計感測.....	44
圖 4.22 磁力計四種不同影響之二維平面圖.....	44
圖 4.23 自動避障修正前.....	45
圖 4.24 自動避障修正前.....	45
圖 5.1 導覽系統歡迎頁面.....	46
圖 5.2 導覽系統學院介紹圖 5.3 導覽系統大樓介紹.....	47
圖 5.4 導覽系統景點介紹.....	48
圖 5.5 校園訊息畫面.....	48
圖 5.6 更多校園新聞頁面.....	49
圖 5.7 校園新聞內容頁面.....	49
圖 5.8 現在位置介面.....	50
圖 5.9 路線規劃介面.....	51
圖 5.10 導覽系統自走車側面照.....	52
圖 5.11 導覽系統自走車正面照.....	52

表目錄

表格 1.1 工作分配表.....	12
表格 1.2 甘特圖.....	12
表格 5.1 自走車材料表.....	51

Chapter 1 緒論

1.1 研究動機

多元教育環境的氛圍下，專業學術文化乃至社會創業經驗的交流與分享日益蓬勃，在很多大專院校中，每天都會有各種領域的專業人士穿梭，他們分享自己研究專業的經驗與歷程，也提點除了目前的研究領域，甚至對領域的未來所提出的一些精闢的見解和預期。除學術演講之外，尚還有許多的大型展覽活動及校園活動需要在大學舉辦，而在廣大的校區當中給人員提供服務，不論是人員在校園的引導，活動的舉行，甚至到活動結束之後人員的疏導，都是難題。

然而只要有人的地方，就有服務上的需求，改善這個問題，是我們研究的起點。而當今現有的校園導覽系統 APP，幾乎都是以靜態網頁或是手持定位 App 的模式來進行導覽，但這樣的導覽方式往往只是單方面的提供資料，使用者可能還是會不清楚該如何前往該處，其所提供的實質效果是有限的。

因此我們便思考，是否能有一個移動的實體平台來實現校園導覽，讓使用者能有更清楚的導覽與更好的體驗。我們同時也思考如何讓機器人更貼近使用者不同的服務需求，讓使用者在與機器人的互動上有更好的體驗，於是本專題致力於此問題，將導覽系統實體化，提昇使用者的體驗度。

1.2 研究目的

基於上述，本研究將以逢甲大學為實例進行探討，以提高導覽體驗度為前提，設計並開發校園導覽機器人系統，並期望未來在各活動場合能減少一定會負擔的大量勞力成本。

除了機器人之外，現今的網路蓬勃發展，IOT(Internet Of Things)的應用也愈來愈多，於是我們將自走車作為移動基底，並於上方架立平板 App 供使用者操作，將欲命令自走車行為的訊息透過 Bluetooth 傳遞，發送給 Raspberry Pi 3 對自走車進行控制，而導覽資訊及各種資料由 Android App 透過 Wifi 向 Server 發出 Http 協定請求，並透過路徑規劃及角度修正等演算法，將自走車控制至指定位置，達到實際帶領使用者到目的地的效果，以其幫助使用者能快速認識逢甲校園，及找尋所目標所在位置。

1.3 研究方法

本系統整合內容過於龐雜，故我們將其分成各種模組來釐清各自的研究方法，除能在一開始更清楚各模組之細節外，也能藉此提昇開發研究效率。故我們將系統分為行控、通訊、導航、導覽模組，首先是行控模組，以 Raspberry Pi3 來建置環境，硬體程式部分以 C++ 來撰寫並取得板子的控制，經過反覆試跑及修改，再用購買來的木板切割、打鐵釘，以有限的條件，搭建出適合本導覽系統的車體結構；通訊模組則是實作 Android 與 Pi3 的藍芽通訊和 Android WebView 與 Web 的溝通；而導航模組，為了規劃出較佳的導航路線，我們應用了 Google map API，也新增校園路徑節點，我們做了一些修正演算法及判斷演算法，主要有：路徑規劃、定位過濾、方向角度修正及節點判斷；最後是導覽模組，這個模組內有幾個部分，如：校園資料的收集與資料庫建置，網頁前端和 RWD 實作、網頁後端實作，最後則是前端、後端、資料庫的串接。

1.4 研究流程

我們將整個系統主要劃分為四種模組來進行研究，如下圖所示：

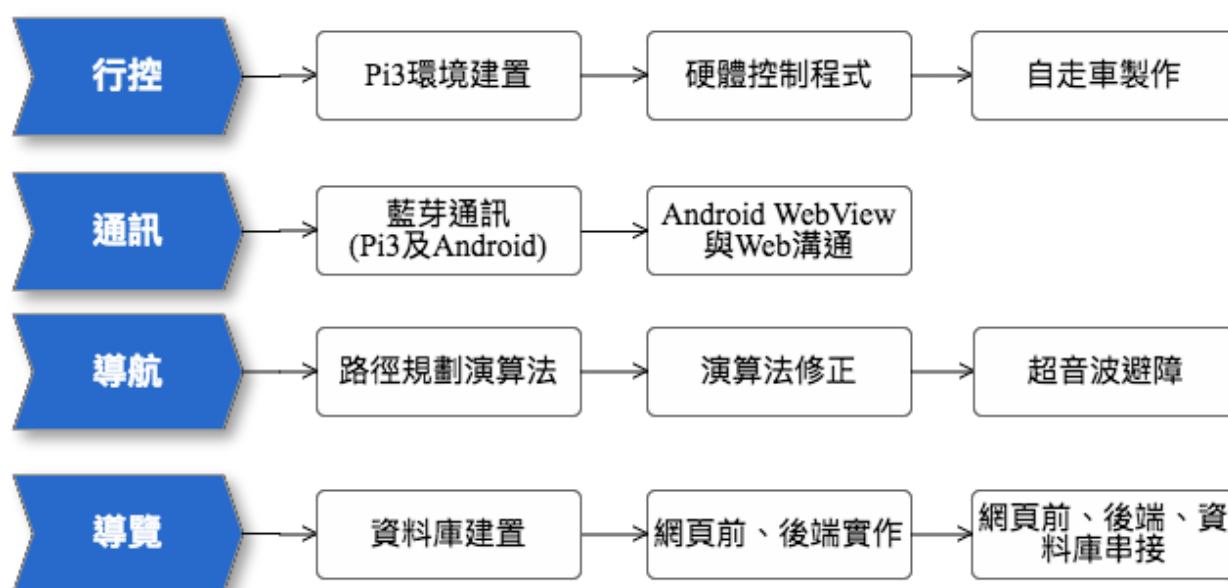


圖 1.1 系統流程圖

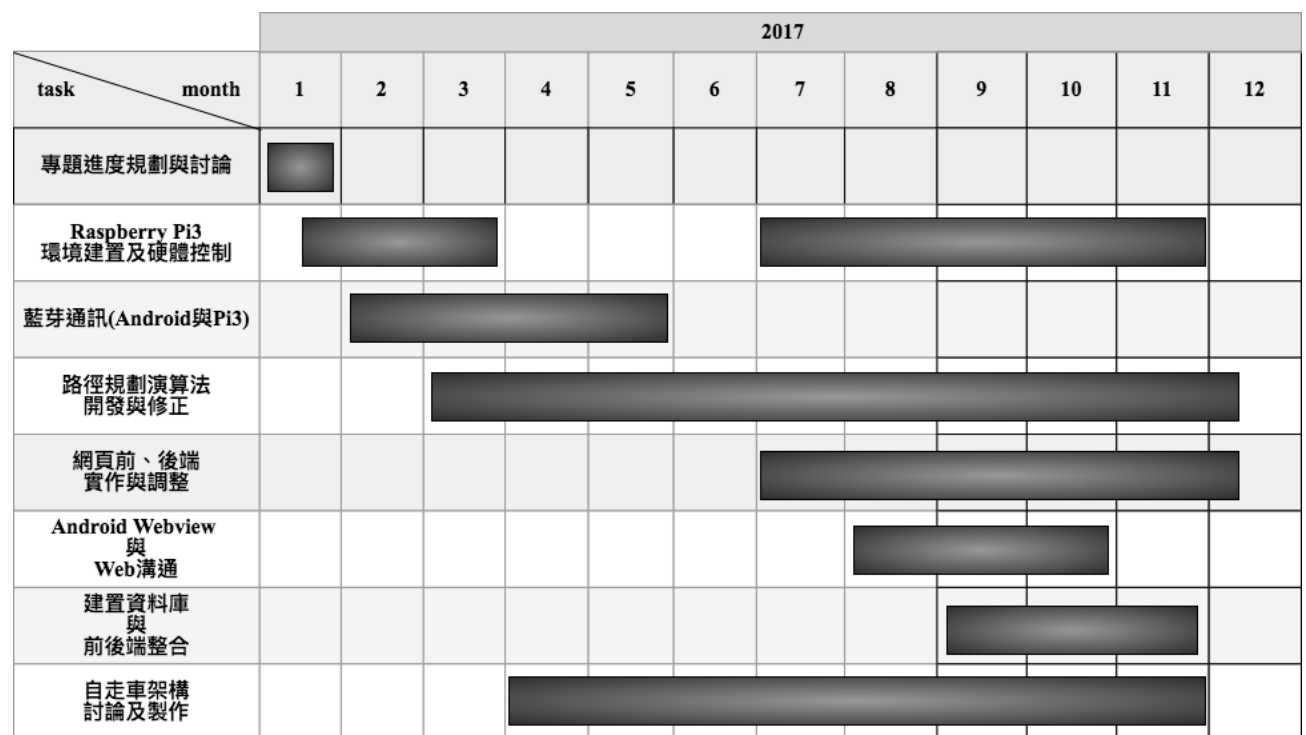
1.5 工作分配及甘特圖

以下為本專題之工作分配表及甘特圖：

表格 1.1 工作分配表

工作項目	工作分配
Android App	邱皇旗
網頁前端	柯權哲、吳啟民
網頁後端及前後端串接	柯權哲
硬體控制	柯權哲
車體製作	邱皇旗、柯權哲、吳啟民
資料庫設計	邱皇旗、柯權哲、吳啟民
系統整合	邱皇旗、柯權哲
文件撰寫	邱皇旗、柯權哲、吳啟民
影片製作	吳啟民
校園資料蒐集、海報設計	郭怡均、邱皇旗、柯權哲、吳啟民

表格 1.2 甘特圖



Chapter 2 系統開發平台

2.1 Android Studio

Android Studio 是 Google 於 2013 I/O 大會針對 Android 開發推出的新的開發工具，對於開發 Android 系統，有提升開發效率的作用。



圖 2.1 Android Studio

Android studio 的優點:

(1) 整合了 Gradle 構建工具:

Gradle 是一個新的構建工具，自從 Studio 亮相之後就支持 Gradle，可以說 Gradle 集合了 Ant 和 Maven 的優點，不管是配置、編譯、打包都有別於過去的 eclipse。

(2) 強大的 UI 編輯器:

提供拖拉式建立 UI，讓使用者可以在建立 UI 時可以更加的便利，且有效率。

(3) 內置終端:

這對於習慣操作 cmd 的開發者來說，有更便利的操作。

(4) 更完善的插件系統:

支持各種插件，如 Git...等等，你想要什麼插件，直接搜索下載。Studio 的優點不勝枚舉，在這麼多優點之下，他成為我們開發 android 最有利的利器。

2.2 Raspberry Pi 3

2.2.1 Ubuntu MATE 16.04 LTS

Ubuntu 是相當受歡迎的 Linux 發行版，具有許多分支，包括 Ubuntu MATE、Ubuntu Budgie、Kubuntu 和 Xubuntu 等，這些分支都裝載著不同的桌面環境和軟體包。

顧名思義，Ubuntu MATE 帶有 MATE 桌面環境。MATE 是堅實的 GNOME 2 的延續。憑藉著耀眼的外觀和直觀的設計，這個積極開發的桌面環境在 Linux 最佳桌面環境列表中排名第一。

Ubuntu 的 MATE 比 Ubuntu 旗艦版本要輕得多。Ubuntu MATE 的 Raspberry 版本由 Martin Wimpress 和 Rohith Madhavan 開發，基於 Ubuntu 16.04.2 LTS 版本。不言而喻，它非常輕巧。

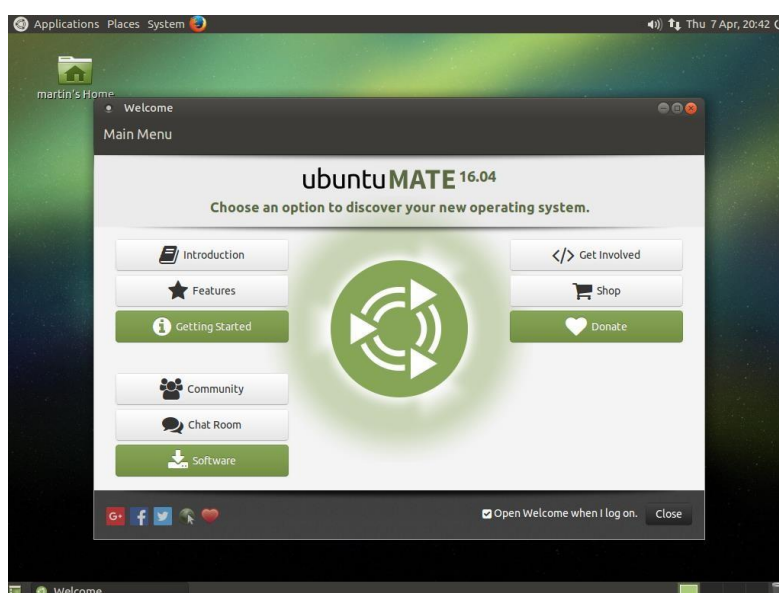


圖 2.2Ubuntu MATE 16.04 LTS

2.2.2 Eclipse

Eclipse 是十分著名的跨平台開源 IDE(Integrated Development Environment)，即整合式開發環境，常用於 Java 語言的開發，亦有許多的外掛模組，讓其搭載後也能開發 C++ 等其他語言。

Eclipse 是一個被廣泛用於許多不同領域的開發軟體，由 Open Source Community 創建，而欲採用 Eclipse 來撰寫 C++，還需要使用到 Eclipse CDT(C/C++ Development Toolkit)。

Eclipse CDT 簡單來說就是提供了一種基於 Eclipse 平台的、功能齊全的整合開發環境，許多 Java 開發人員喜愛的不少 Eclipse 優秀功能，都能提供給 C/C++ 開發人員，如：自動構建、項目管理、語法著色、代碼完成的等等，這都成為讓 CDT 非常流行於 Linux 的原因。



圖 2.3Eclipse

2.3 VSCode

Visual Studio Code (VS Code) 是一個由微軟開發的，同時支援 Windows、Linux 和 macOS 作業系統的開源文字編輯器。它支援偵錯，內建了 Git 版本控制功能，同時也具有開發環境功能，例如代碼補全（類似於 IntelliSense）、代碼片段、代碼重構等。該編輯器支援用戶自訂配置，例如改變主題顏色、鍵盤捷徑、編輯器屬性和其他參數，還支援擴充功能程式並在編輯器中內建了擴充功能程式管理的功能。

Visual Studio Code 支援多種編程語言。與其他編輯器不同的是，Code 的許多功能並不在菜單中顯示，而是通過「命令面板」來調用。用戶可以在命令面板中搜尋和輸入命令，從而實現指定功能，例如安裝外掛模組、設置屬性等。



圖 2.4VS Code

Chapter 3 文獻回顧及探討

3.1 Bluetooth

這是一種無線技術標準，用於固定或是移動的裝置，在短距離間進行交換，形成個人區域網路(PAN)。藍芽的運作原理是在 2.45GHz 的頻帶上進行傳輸作業，藍芽的傳輸範圍在 10 公尺 (0dBm) 到 100 公尺 (20dBm) 左右，採用每秒 1600 次 跳頻展頻技術。



圖 3.1. Bluetooth

3.2 Android

3.2.1 Android

Android 是一套以 Linux Kernel 為基礎的系統，是一套半開放原始碼作業系統，常見於手機平台之開發。其不僅提供了內容豐富且完整的應用程式架構，也有各種合用的函式庫給開發者做學習、使用。

綜觀 Android 系統，可將其主體分為上層的應用程式框架(Application Framework) 與應用程式 (Applications)合成的 Java 應用層，以及下層的系統函式庫 (Libraries) 與 Linux 作業系統合成的 C 語言系統層。而正因為這樣的設計方式，使得 Android 能夠同時汲取 C 語言與 Java 雙方的開放原始碼資源，讓 Android 擁有龐大的社群能量。

分析 Android 系統，能再將其分成五個階層

(1) 應用程式(Applications):

此階層是以 Java 寫成，為使用者唯一會接觸到的階層，含有一系列的應用程式：包含桌面、郵件程式、簡訊程式、瀏覽器與其他應用程式。

(2) 應用程式框架(Application Framework):

應用框架層提供了所有 Android 開發所需要的元件，讓開發者可以完整調用與核心應用程式相同的 API。這一層是編寫 Google 發佈的核心應用時所使用的 API 框架，遵循其框架的開發原則及安全限制之下，開發人員同樣可以使用這些框架來實作自己的開發，不僅簡化了程序開發的架構設計，應用程序的重用機制也讓用戶可以方便的替換程序組件。

基本上所有的應用程式皆由 services 和 systems 所組成，包含：Views system、Content Providers、Resource Manager...等。

(3) 函式庫(Libraries):

Android 有一套豐富的系統組件，這些組件皆是由 C/C++函式庫所組成，提供給開發者透過 API 呼叫進行使用、開發。

(4) 執行層(Android Runtime):

在 Android 執行層中，分有兩個核心架構，其一是 Core Libraries(核心函式庫)，一是 Dalvik Virtual Machine(Dalvik 虛擬機)。而 Android 雖然用 Java 來撰寫、開發應用程式，但卻不是以大家所用的 Java Runtime 來執行程式，而是使用自家研發的 Android Runtime 來執行程式。

(5) Linux 核心(Linux Kernel):

Android 平台所用的作業系統是 Linux，其所採用的 Linux 核心所含的服務包括：

- a. 安全性(Security)
- b. 記憶體管理(Memory Management)
- c. 行程管理(Process Management)
- d. 網路堆疊(Network Stack)
- e. 驅動程式模型(Driver Model)

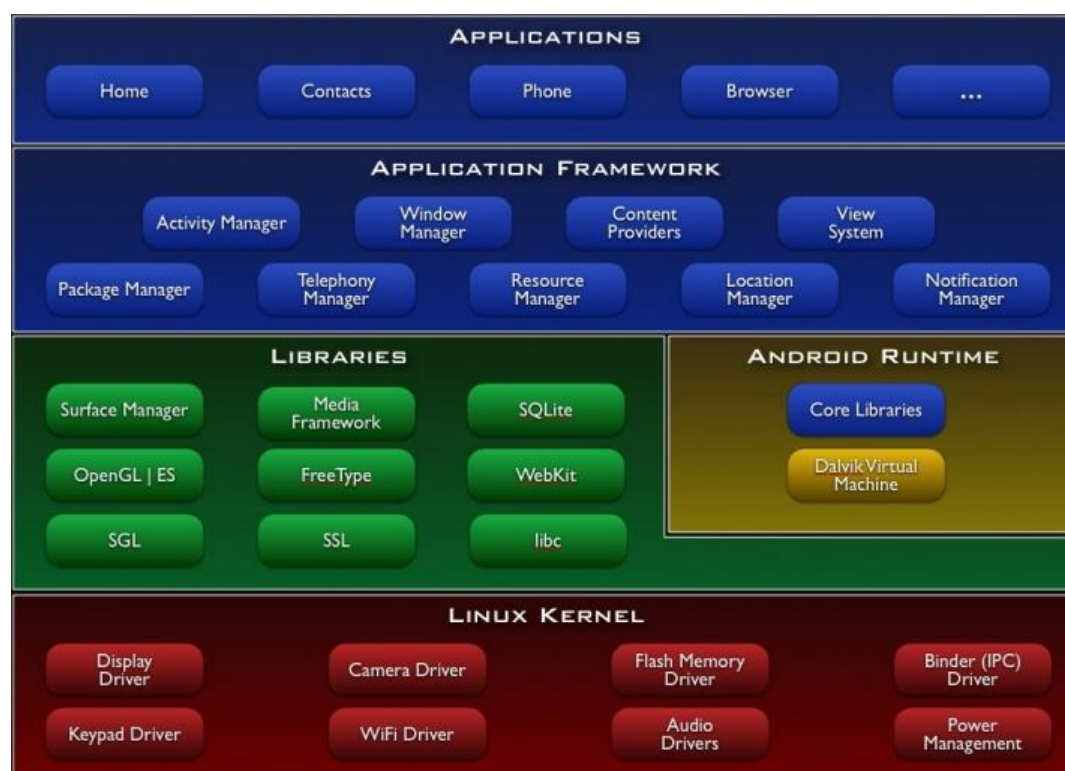


圖 3.2Android 系統架構圖

3.2.2 Google Map API

若欲自訂自身平台所需的地圖，Google 提供了各種相應平台的 API，開發者可依照需求選擇。Google 所提供的 Maps Android API 是一個給開發人員在 Android 平台上使用的原生 API，開發者能使用多種地圖功能，用於不同性質的系統開發與呈現，也有室內地圖以及樓層平面圖。如果想要使用 Google Maps Android API，必須要申請專用的 API 金鑰，才能存取 Google 地圖的伺服器，只要到 Google API Console 上註冊自己的開發專案，即能取得可新增至開發程式的 Google API 金鑰。

另有一點需注意，在 Google Maps Android API 官網中，有一個 Attribution Requirements 項目，這部分要求在撰寫 App 程式中若有使用到 Google Maps Android API，必須在 App 中有一段法律聲明，須要包含 Google Play Services attribution text，也建議將其獨立成一個 Menu 或是一個 Menu 中的關於。



圖 3.3 Google Maps APIs

在此列出 Google Map Android API 所提供之常見功能：

- (1) 地圖類型：提供了四種地圖類型做選擇，標準、混合、衛星、地形，以及不載入任何地圖方塊的空網格。
- (2) 市內地圖：特定建築物有提供，也可新增自己的樓層平面圖，待審核後即可新增至 Google 地圖。
- (3) 地圖縮放控制、羅盤、我的位置按鈕、手勢功能
- (4) 地圖事件，如地圖點選、資訊視窗、縮放地圖、CameraUpdate 等。
- (5) 繪製地圖：標記、資訊視窗、圖層疊加。
- (6) 路徑規劃及導航。

3.2.3 Webview

Android Webview 這個系統元件採用 Chrome 的核心技術，它亦是 Android View 的延伸，可讓 Android 應用程式內顯示網頁內容。Webview 是 Android 中一個非常實用的套件，和 Chrome、Safari 一樣都是基於以網頁渲染引擎，可以透過加載 html 數據迅捷地呈現網頁，可不必跳出 App 就在 App 內檢視網頁。

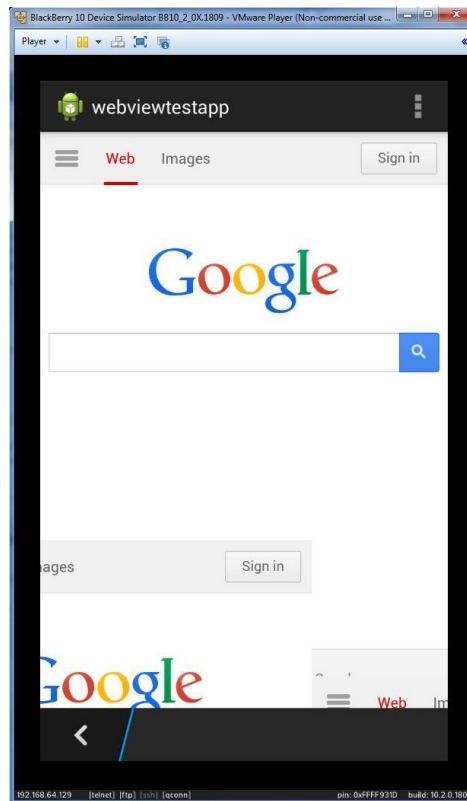


圖 3.4WebView

3.3 Raspberry Pi

3.3.1 Raspberry Pi3

樹莓派（Raspberry Pi）如下圖,是一款基於 Linux 系統的單板機電腦。它由樹莓派基金會所開發，其特色是以低價的硬體及自由開源的軟體支援，來促進基本電腦科學教育，只要 35 美金就能擁有一台簡易電腦。

雖然硬體成本低廉，但是功能卻相當齊全，部分常用功能如下：

- (1) 操作方面以開放原始碼的 Linux 系統為基礎
- (2) 硬體方面具有基本的乙太網路孔可供網路線連接
- (3) USB 介面可擴充周邊裝置
- (4) 影像方面具有高解析度的 HDMI 介面輸出



圖 3.5 Raspberry Pi3

在官方網站上有提供多種開放的作業系統及軟體可以使用，滿足一般使用者上網、玩遊戲、播放高畫質影片、音樂...等娛樂用途，而進階使用者能在上面架設網頁伺服器、檔案伺服器或以 Python 進行撰寫透過 GPIO 來控制電子設備等應用，是一個滿足基本網路瀏覽、文字處理以及學習計算機科學的小型電腦。

本系統所採用的是 Raspberry Pi3，相較於前面幾代版本相差並無太大，在官方的定價仍為\$35，而 Pi3 最大的變化則是加入 BCM43438 此一複合式晶片，含有 Wi-Fi 802.11 b/g/n (2.4GHz) 與藍牙 4.1 功能，因為是獨立晶片，不會與 USB 和乙太網路搶頻寬。



圖 3.6 BCM43438

3.3.2 PWM

PWM 全名為 Pulse Width Modulation(脈衝寬度調變)，它利用數位訊號模擬類比訊號，來調整諸多部分的控制如馬達的轉速、燈光的亮度...等，憑藉數位訊號高頻率的轉換，來調整開與關訊號的比例，就能模擬我們需要的類比訊號。

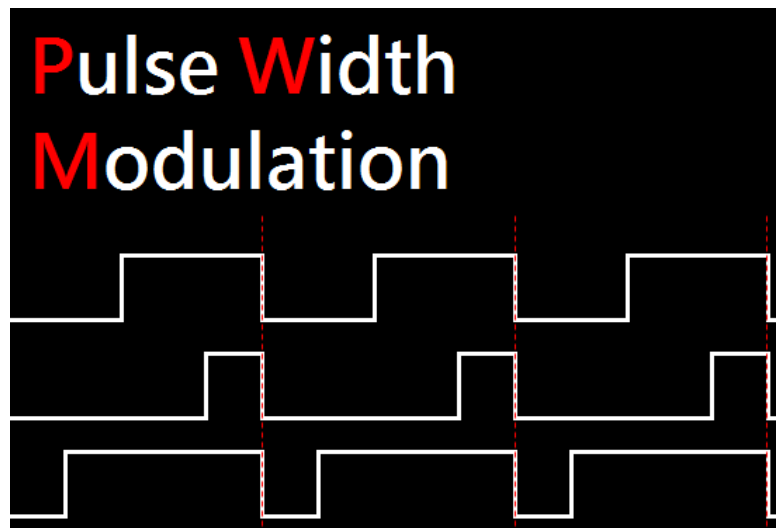


圖 3.7 PWM

3.3.3 HC-SR04

超音波感測器為超音波發射器、接收器及控制電路組成。當它被觸發的時候，會發射一連串頻率為 40kHz 的波，其波在行進時接觸到最近的物體後就會回彈，並由接收器接收其回音。

HC-SR04 是一個價格 100 元左右的超音波感測器，他可以探測的距離為 2cm~400cm，精度為 0.3 cm，感應角度為 15 度，故我們採用其作為車體系統行進的感測器。

以下為超音波行進/回彈示意圖：



圖 3.8 HC-SR04

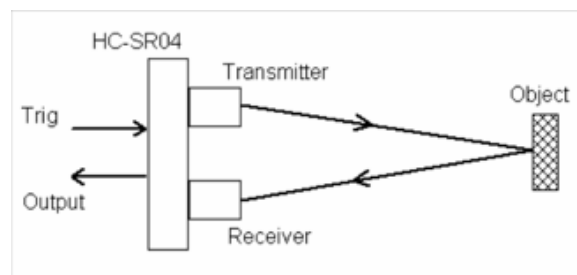


圖 3.9 超音波行進/回彈示意圖

3.3.4 L298N

L298N 是 ST 公司生產的一種能承載高電壓、大電流電機驅動晶片。其特點是工作電壓高，輸出電流也大，瞬間峰值電流可達 3A，故我們選擇它來作為驅動前輪及後輪馬達的驅動板。

其有以下特點：

- (1) L298N 作為主要驅動晶片，發熱量低，驅動能力大。
- (2) 當使用大過於 12V 驅動電壓時，可以再引出 5V 電壓供使用。
- (3) 使用了大容量的濾波電容保護二極管以提高可靠性。



圖 3.10 L298N

3.4 Web

3.4.1 Responsive Web Design

響應式網頁設計(Responsive Web Design, RWD)，或稱自適應網頁設計，此概念於 2010 年 5 月由國外網頁設計師 Ethan Marcotte 提出。RWD 是一種基於網頁的設計方式，從桌電顯示器到行動裝置，該方法可以使網站在不同的裝置上、不同的解析度時都能有適當地呈現，能效率地減少使用者體驗當下的縮放、平移和捲動。而對於網頁工程師和前端設計師而言，相對於過去需要對不同裝置的瀏覽進行開發，這樣的設計模式更利易於網頁的維護。



圖 3.11 RWD 示意圖

3.4.2 Bootstrap 4 Beta

Bootstrap 是一種用於快速開發 Web 和網站的前端框架，是一種包含 HTML、CSS、JavaScript 的框架，提供有許多類別如，字體編排、導航(Navigation)、按鈕(Button)、表單(Form)及其他各種元件和 JavaScript 擴充套件，這些都是為了達到快速開發動態網頁和 Web 應用的宗旨。

Bootstrap 是目前最主流的 CSS 前端框架，我們系統採用了在 2017/8/10 釋出的 Bootstrap 4 beta，最大的特色是改變了網格系統，以及如下幾個新元素：

- (1) Flexbox 以及強化的網格系統:網格系統不使用 float，改採 Flexbox，讓整個裝置的畫面有更具邏輯及彈性的排版。
- (2) 卡片(card)元件：事實上許多元件其本質相似度高，如 wells、panels、thumbnails 等的元件都被捨棄掉，取代之的則是 card 元件，不僅可以達到上述元件功能，運作得也更好。
- (3) 彈性更高的變數:以往樣式皆是固定的名稱、色彩，透過變數 Sass \$map 可以更有彈性的調整不同的視覺效果。
- (4) 從 Less 到 Sass:自 Sass 不再只能使用 Ruby 編譯後，Libsass 有更快的編譯速度，而且 Sass 本身具有龐大的開發社群。



圖 3.12 Bootstrap

3.4.3 SPA

SPA 即為 Single Page Application，對 SPA 而言，HTML DOM 是極其重要的元素。由於大部分的畫面都不會有任何換頁行為，程式裡多是在控制 DOM 元素及 Container。通常都是由 Client 端向 Server 端提出頁面的要求，而 Server 便會回傳相對應的 HTML 頁面給 Client，當 Client 收到了 Server 的 HTML 便會進行頁面的重新加載。

而 SPA 的優點在於使用者第一次連上網站時，瀏覽的內容都已被下載好，所有頁面、資源不需要再重載，對於使用者而言，這樣的體驗感無疑地更好。

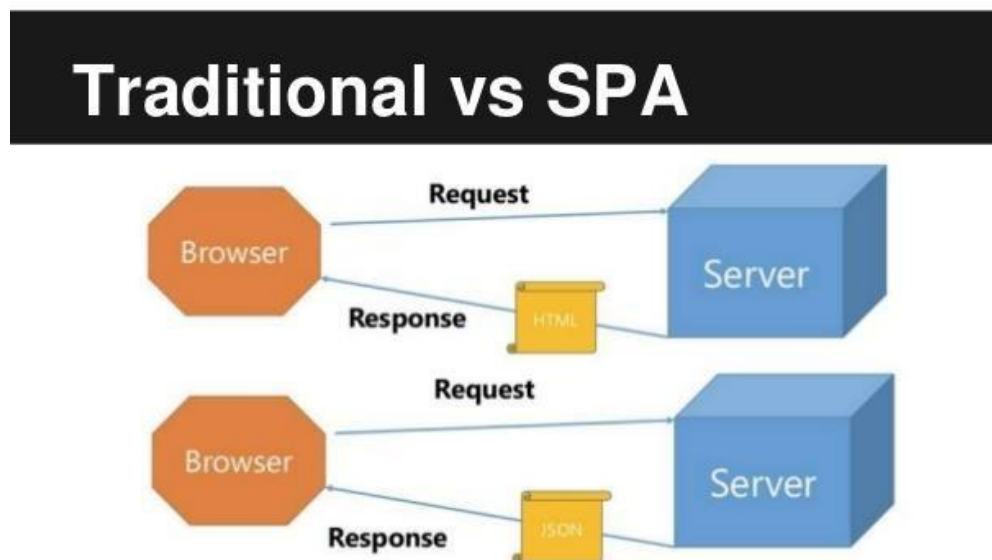


圖 3.13 傳統型和 SPA 型網頁分析圖

3.4.4 JQuery

jQuery 就是一套 JavaScript 的開源函式庫，jQuery 主要著重 DOM 文件的操作，其宗旨包含「快速選取元素」和「做一些事情」。快速選取元素是讓開發者可以一次選取單一甚至多個元素，接著開發者可以將選取的元素做一些動態的改變，諸如常見的顯示、隱藏，以及銜接等等。

簡言之，jQuery 是針對 JavaScript 所缺失的地方加以改善、增進，開發者在撰寫自己需要的功能時，正確、精準地使用 jQuery 可以令程式碼更有效率及整潔地表達。



圖 3.14jQuery 和 JavaScript 關係圖

3.4.5 語音合成

語音合成是以人工的方式產生人類語音，即文字轉語音(text-to-speech)。欲在網頁上實現語音合成，透過 Speech Synthesis 的相關 API 達成。

主要步驟有二：

- (1) 建立 SpeechSynthesisUtterance，表示欲說的內容。
- (2) 將 utterance 物件交給 window.speechSynthesis 使用 speak 的方法念出來。



圖 3.15 語音合成

3.4.6 Restful API

在網路應用程式裡，可以分成前端和後端兩個部分。而以現今的發展而言，瀏覽前端的設備蓬勃發展(手機、桌上型電腦、平板)，所以在前端設備與後端之間，勢必要有一個統一機制，方便兩者來做通信，也因此使得 API 框架流行了起來。

而 Restful API 是目前相較較為成熟的一套互聯網 APP 的 API 設計技術，主要有其五大準則：

- a.GET(SELECT)：從 Server 取出資源(一項或多項)。
- b.POST(CREATE)：在 Server 新建一個資源。
- c.PUT(UPDATE)：在 Server 更新資源 (Client 端提供改變後的完整資源)。
- d.PATCH(UPDATE)：在 Server 更新資源(Client 端提供改變的屬性)。
- e.DELETE(DELETE)：從 Server 刪除資源。

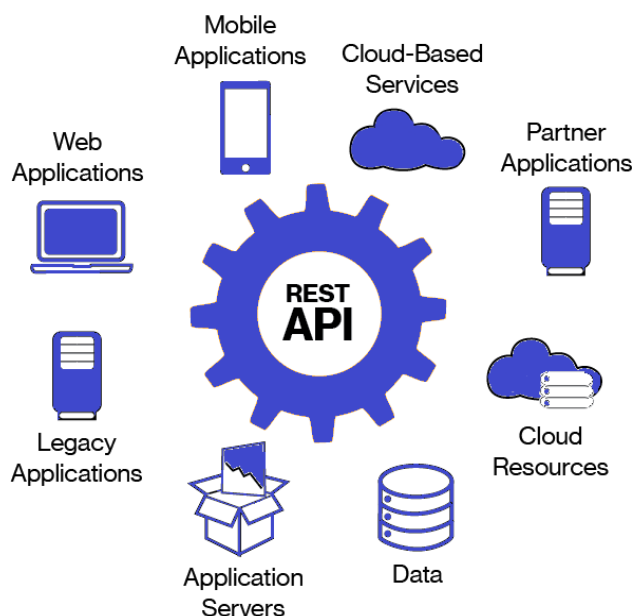


圖 3.16 RESTful API

3.4.7 Nodejs

Nodejs 是一個 JavaScript 的後端語言，是使用 V8 JavaScript 編譯器所開發的產品。性能面而言，Nodejs 是一個高效能且易擴充的網站應用程式開發框架，Nodejs 的優異之處在於資源的調校，程式接收到一筆連線(connection)時，會聯繫作業系統將其保留到 heap 中並讓連線進入休眠(sleep)，它會等到系統進行通知時，才會觸發 Callback。如此的處理方式只會佔用記憶體，並不會使用到 CPU 的資源。

綜觀 Nodejs 有幾點優點:

- (1) 入手容易:只要曾經接觸過 PHP、Ruby 或是 Python 多少對 JavaScript 有一些基礎觀念，要入門 Nodejs 都能迅速很多
- (2) 高性能、低耗能:非同步 I/O、Event-Driven
- (3) 強大社群支援:NPM
- (4) Realtime 應用的選擇:Socket.IO



圖 3.17 Nodejs

3.5 資料庫

3.5.1 MongoDB

MongoDB 是 10gen 這家公司開發的一個 NoSQL Database，屬於 Document-Oriented Database 這一類型，希望能夠結合 Relational Database 與 Key/Value Database 雙方的優點，很適合用在 Web 應用程式、Internet 架構的環境底下。

MongoDB 是一個高性能，開源，無模式的文檔型資料庫，是當前 NoSql 資料庫中比較熱門的一種。它在許多場景下可用於替代傳統的關聯式資料庫或鍵/值存儲方式。MongoDB 使用 C++開發。



圖 3.18 MongoDB

3.5.2 No-SQL

NoSql，全稱是 Not Only Sql,指的是非關聯式的資料庫。下一代資料庫主要解決幾個要點：非關聯式的、分散式的、開源的、水準可擴展的。原始的目的是為了大規模 web 應用，這場運動開始于 2009 年初，通常特性應用如：模式自由、支援簡易複製、簡單的 API、最終的一致性（非 ACID）、大容量資料等。NoSQL 被我們用得最多的當數 key-value 存儲，當然還有其他的文檔型的、列存儲、圖型資料庫、xml 資料庫等。

3.5.3 導覽系統

一般來說，大部分大學校園都具有非常遼闊的校區，在校園資訊方面更是龐雜，在校園的位置資訊提供方面，早期多是使用靜態的網頁、或是在校園中規劃立牌...等等。而幾年下來，行動裝置的普及讓行動導覽的出現愈發普及。

在此我們找出一款行動導覽 App 為例作為校園導覽系統的範例：

(1) NTHUAPP 清華大學

NTHUAPP 是清華大學學生自主開發的 App 應用程式，提供部分資訊如：校園巴士、遺失物及行事曆。而校園地圖的實作是使用清華大學校園地圖影像，以供使用者查看，以該系統而言不僅無法提供 GPS 定位，亦無法提供路徑導航。

(2) NCKU 成功大學

NCKU 校園導覽系統，是由成大工科所學生所開發的程式，提供部分資訊如：課程資訊、系館聯絡資訊。而校園地圖導覽方面是由 Google Map 下載的地圖圖資疊加自行繪製的地圖影像，而定位是使用 GPS 來取得經緯度，再藉經緯度/座標轉換，得到校園地圖的相對座標位置。此系統的目的主要在於得到目的地最短路徑，及地標資訊的獲取。

綜觀許多導覽 App，皆是相近於上述所提到的系統。而本系統組成含有校園導覽系統，除了即時更新系統網頁上資訊以外，更有實體的行動導覽，在實體的導覽車上架有作為使用者介面的平板，供使用者瀏覽、點擊觸控，期望能提供更齊全且更不一樣的使用者體驗。

Chapter 4 系統分析與設計

4.1 系統設計

本系統的設計主要是提供使用者在進入逢甲校園後，能夠更快速的認識逢甲大學，並減少使用者對校園的不熟悉，以及花費在尋找大樓或學院等時間，因此本系統將即時顯示導覽機器人當下位置，並能在確定路線後主動帶領使用者進行導覽。

4.2 系統架構圖

下圖為本專題之系統架構圖，而所有的模組都是由 Android 當作橋樑來進行溝通，最後並透過 Raspberry Pi3 來對自走車進行操控。

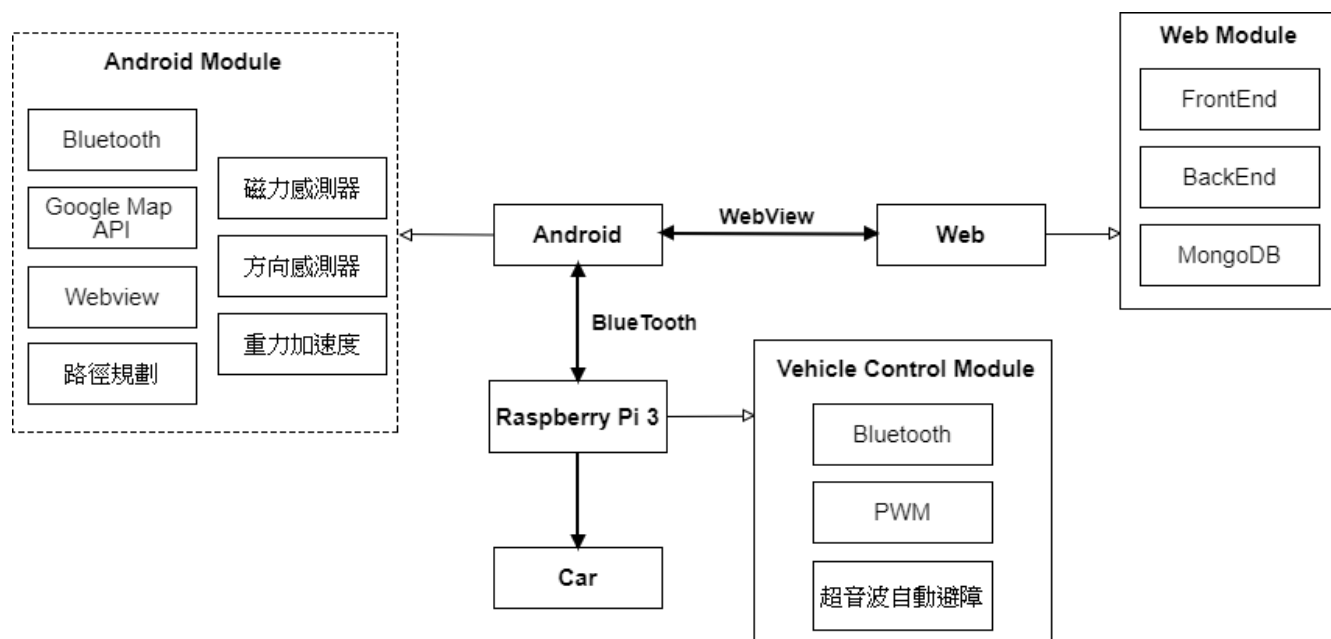


圖 4.1 系統架構圖

4.3 系統流程圖

下圖為本專題之系統流程圖，整個系統的運作主要是靠下圖右三區塊之 Android 模組來傳遞命令，再由 Raspberry Pi3 來執行動作，而系統 Client 端部分則是由 Android Webview 透過 Http 向 Server 取的 Web 端的畫面並輸出。

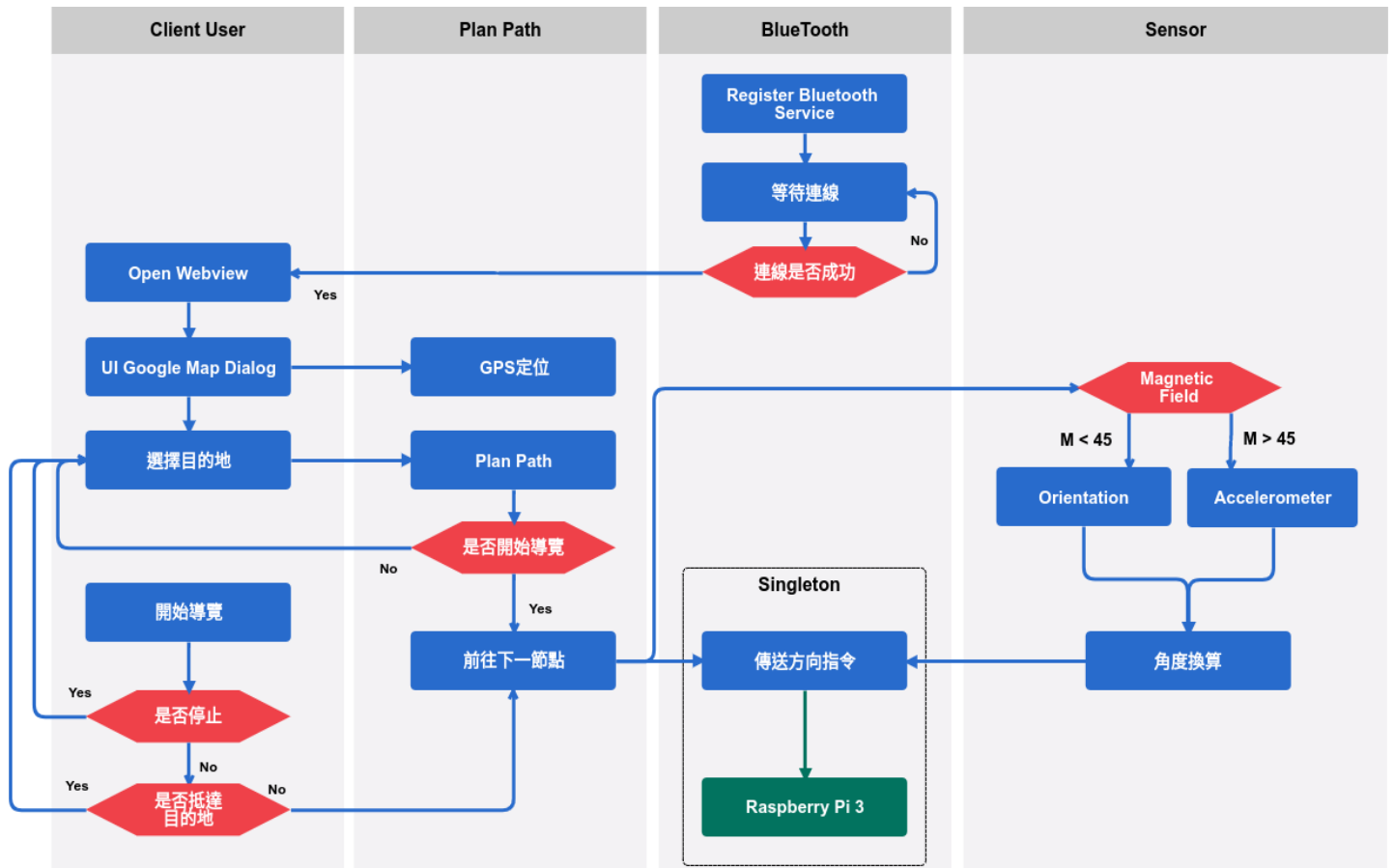


圖 4.2 系統流程圖

4.4 使用者與系統互動模式

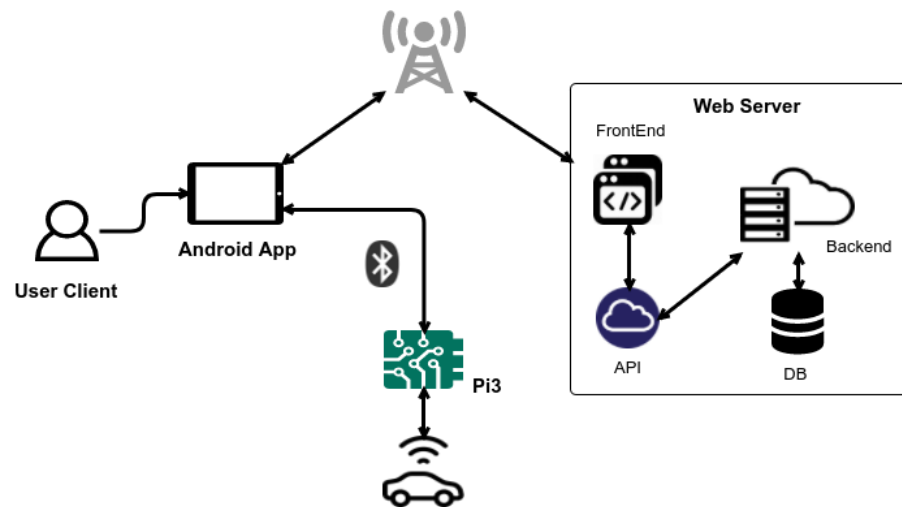


圖 4.3 使用者與系統互動模式圖

4.5 Android App

4.5.1 Android 架構

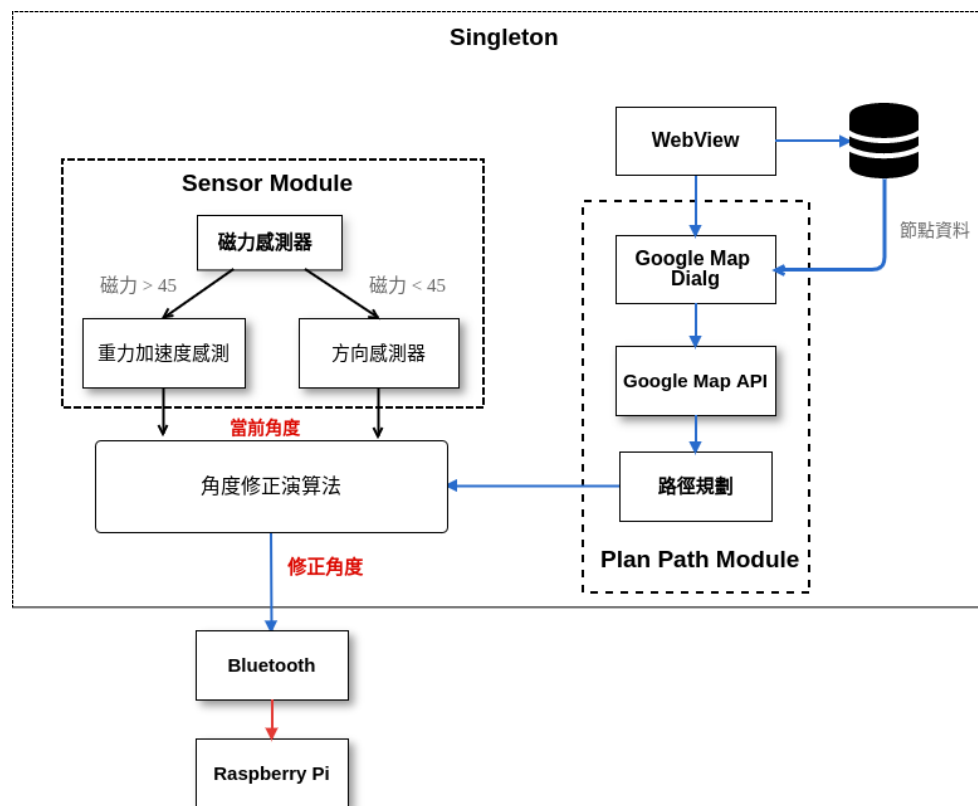


圖 4.4 Android 架構圖

4.5.2 Singleton Design Pattern

Singleton 是讓有需要某項 Service 的 Activity，透過 Method 的宣告，即可得到此 Service，在 Singleton 的概念中，分成三個區塊

- (1) Private Constructor: 為了防止 Class 實作其內容
- (2) Static Method: 建立 Instance
- (3) 透過 getInstance()，獲得在 Singleton Service

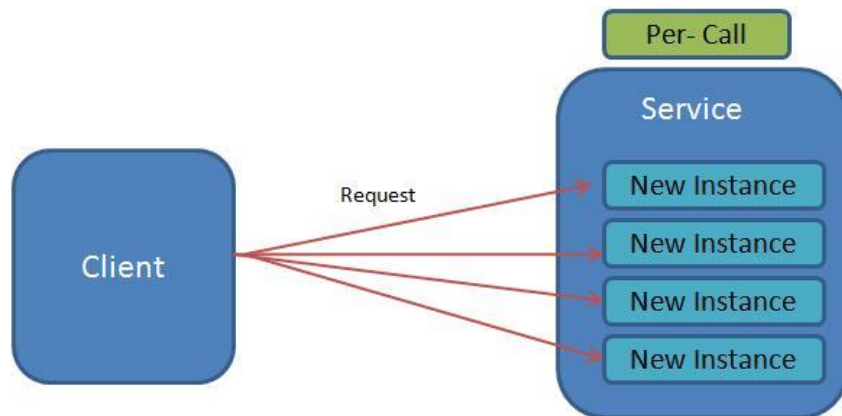


圖 4.5 Singleton 概念圖

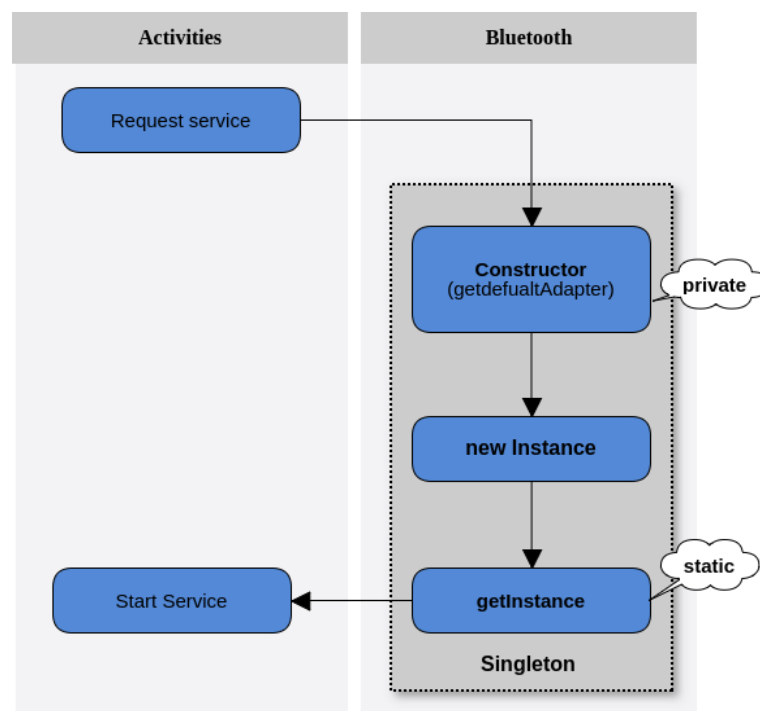


圖 4.6 Android Bluetooth 通訊概念圖

4.5.3 Bluetooth Socket

在 Bluetooth 的模組上，使用 Android 上 BluetoothSocket 的 Library，藉由 Socket 的方式，讓我們的行動裝置可以成功的連結上 Raspberry Pi3，進行資料的交換，在程式執行時，使用 Handler 實作 Run 的 Method，讓 Bluetooth 可以執行在背景程式中，當需要傳遞訊息時，可以使用 Write() 傳遞出去。

再利用 Singleton 的 Design Pattern，讓多個 Activity 都能使用到這項服務，Bluetooth 實作部分寫入 Singleton 的 Private Class 中，讓需要的時候才被呼叫，不會像 Static Class 會在每個 Activity 裡產生。

4.5.4 Sensor Data

宣告 SensorManager 型別的參數，再利用 getSensorList() 註冊手機的 sensor 服務，registerListener 可以設定 data 傳遞的頻率，頻率分成四種，分別是 **SENSOR_DELAY_FASTEST**、**SENSOR_DELAY_GAME**、**SENSOR_DELAY_NORMAL**、**SENSOR_DELAY_UI**，可以依據系統需求做調整，在使用過程中，當 sensor 數值改變時，會呼叫 onSensorChanged()，獲得當下的數值。

4.5.5 過濾修正

過濾修正是將不可能，或是不應該發生的極端因素進行過濾，而我們主要針對以下兩種問題進行修正。

- (1) **GPS 定位**：由於 GPS 容易受到天氣、建築物等外在影響，使定位會不定時偏移，於是我們利用兩種方式來修正此問題。
 - (a) 當 GPS 在定位時，所得到的定位與前一個定位距離超過 10M，代表此定位已造成嚴重偏差，則判定這個點是錯誤點，並予以排除。而我們可將定位點套入直角座標兩點距離公式算得其距離，並依經緯度每度約為 111.1 公里，換算後即可得知實際距離。
 - (b) 程式剛執行階段，若在定位時得到 10 個以上的相近定位點，取最後一點作為定位點，便能在導覽開始前確認當下定位沒有偏差，即可代表本次 GPS 的定位是具有可信度的。
- (2) **方向感測器**：

在方向感測器的部分，由於方向感測器會受到磁力的影響，而在逢甲大學校園中，各種因素如機房...等等，導致每處磁力分佈都不一樣，故我們蒐集了逢甲大學的磁力分佈數據。而地球上的磁場平均是 50，所以我們分類出：數據大於 55 以上的點標示為紅色、數據小於 55 的點標示為藍色(如下圖所示)。而當磁力感測器偵測到 55 以上數值時，我們將會採用加速度感測計來做方向修正，如此可免除磁力過大而影響修正角度。



圖 4.7 校園磁力分布圖

若超過基準值，系統將會切換到使用加速度感測器去判斷方位，利用 sensorManager 的 getRotationMatrix function，將加速度感測器的值，進行旋轉矩陣，得到的值，單位是弧度，再藉由 Math.toDegrees() 轉成角度，因為這個方法將會利用磁力的值進行修正，所以比方向感測器更能減少誤差，但是也因為要透過修正，所以會比較沒有效率。

4.5.6 Webview 溝通

在使用 webview 前，必須先 Import WebView，WebSettig，WebViewClient，宣告 WebView 的物件，由於我們 Web 是利用 JavaScript 來溝通，所以為了要在 App 使用 javaScript，所以要 call setJavaScriptEnable() 去開啟，再透過 loadUrl()，指定 Web 的位址。

4.6 Raspberry Pi3

4.6.1 Bluetooth Socket

Raspberry Pi3 上的藍芽通訊主要是為了要與 Android App 溝通，接收從 Android 端所發送過來的指令，並透過程式執行前進或是修正角度等動作。

而本專題主要是利用 C 來撰寫 Bluetooth Socket 程式，並利用 thread 來將兩端的藍芽通訊建立起來，如此一來便能一方面接收 Android 端的指令，也可以透過自身感測器同時對自走車修正角度及操控，才不會被其中一方給 Block 住。

4.6.2 Car Control

依據 Bluetooth 接收到由 APP 端所發送的訊息來對自走車進行控制，當開始導覽後將會進入前進模式，接收到角度時將會進行轉向（利用後退並將前輪轉向的方式進行角度修正）。

而自走車的控制主要是利用 PWM 來對 GPIO 腳位傳送類比訊號。

4.7 Web

為了提高設計上的彈性，以前提高導覽系統的使用者體驗，於是我們採取前後端分離的概念來設計，而所謂的前後端分離，即是前端與後端的程式邏輯互不影響，只是單純的雙向溝通，我們前端是 SPA (Single Page Application) 的方式實作，後端則是一個純粹提供 API (Application Programming Interface) Endpoint 的 API Serve，此種方式可以取代過去後端都是回傳 html 的方式，也能提高網頁前後端的溝通彈性及速度。

前後端的溝通方式如下圖所示，兩者之前是透過 Http Request 的方式來溝通，而前端主要透過 UI 觸發事件向後端提出請求，並可依後端回傳的 JSON(JavaScript Object Notation) 隨時更改 UI 內容。

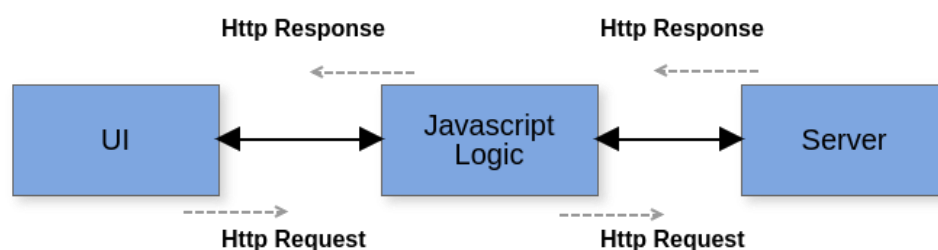


圖 4.8 前後端溝通示意圖

下圖為 Web 架構示意圖

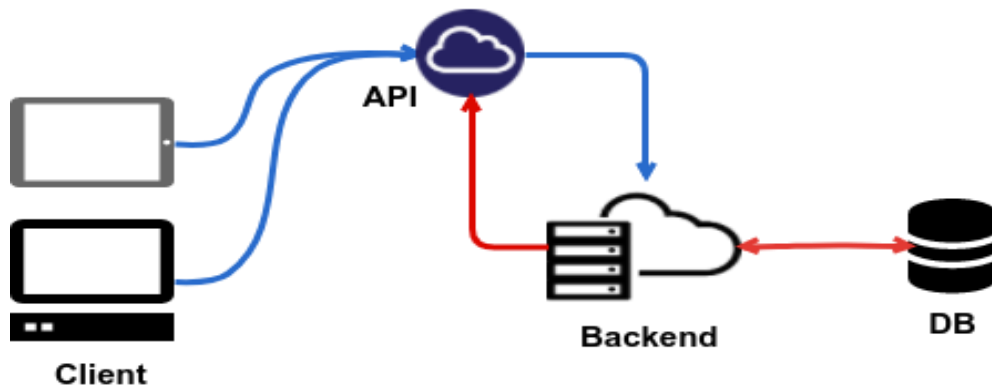


圖 4.9 Client 與後端溝通架構圖

4.7.1 前端

以往的網站都是以多頁式的設計方式呈現，當按鈕有觸發事件時則會換到新的頁面，但這樣的設計方式會造成過多時間的資源載入，每次都需要重新載入全部的資源。而我們為了提高使用者的體驗效果，我們的網頁則是採用 SPA (Single Page Application) 技術來設計。

SPA 技術能在第一次載入網頁時，就將所有資源載入完成，且不需要在因換頁而重新載入，減少不必要的資源載入時間，而單一頁面即可看完所有資訊，對使用者來說資訊傳達會更直接快速，且也讓網頁更像一個 Desktop Application。

前端部份主要實現了以下功能：現在位置、導覽現況、學院介紹、大樓介紹及景點介紹，而其功能皆有所對應的 API(Application Programming Interface) Endpoint 可以呼叫。

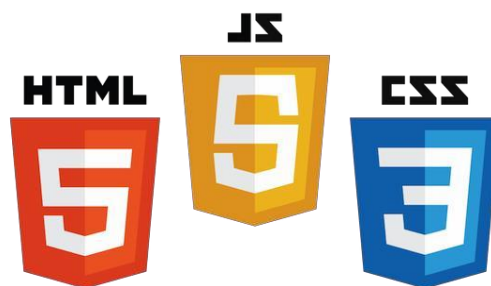


圖 4.10 Web 前端三大語言

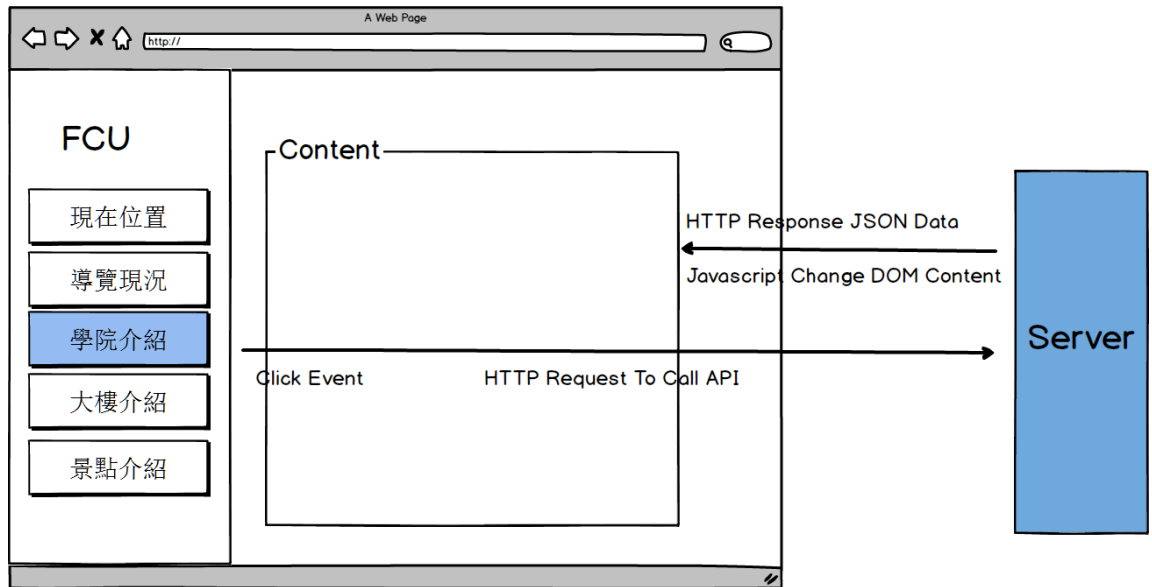


圖 4.11 網頁內容變更示意圖

上圖為網頁內容變更示意圖，前端受到觸發事件後向 Server 端提出資料請求，並使用 Javascript 中的 JQuery Library 將所回傳的 JSON 資料來直接變更 Html DOM 屬性。

如下圖所示，我們可以直接撰寫 Javascript 邏輯來接收資料後，並直接針對我們想要更改的 Html 屬性來讓網頁達到換頁的效果。

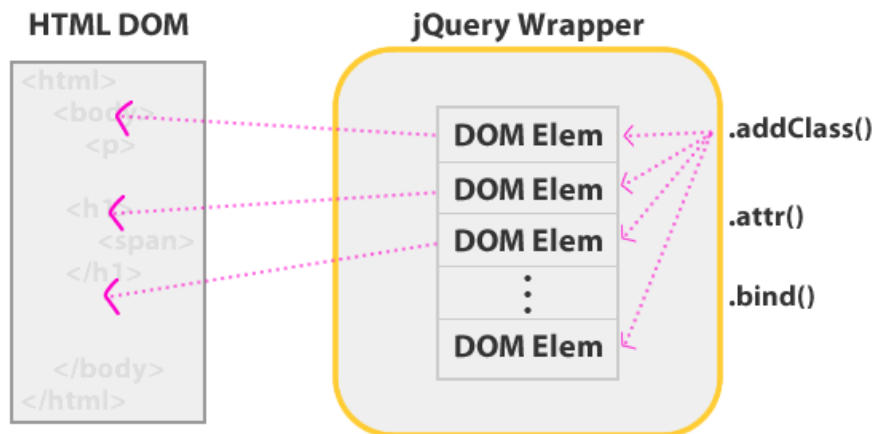


圖 4.12 JQuery 運作示意圖

4.7.2 後端

本專題中是使用 Nodejs 的 Express 框架來撰寫，利用 Express 所提供的 Router 機制可以自訂適合系統的 Router 名稱，並可以依照其名稱向資料庫取得對應的資料，在將資料透過 Http Response 回傳給呼叫此 Router 的對象。

目前後端主要提供了以下幾種 API Endpoint 給前端或使用，以下大致分為學院資訊、大樓資訊、景點資訊三種資料型態的 API，且都支援 GET、POST、DELETE 功能，而這些資訊也能在 URL 最後加上 {id} 來詢問單一特定的資料。

學院資訊： GET /api/fcu/college
POST /api/fcu/college
GET /api/fcu/college/{id}
DELETE /api/fcu/college/{id}

景點資訊： GET /api/fcu/landscape
GET /api/fcu/landscape/{id}
DELETE /api/fcu/landscape/{id}
POST /api/fcu/landscape

大樓資訊： GET /api/fcu/building
GET /api/fcu/building/{id}
DELETE /api/fcu/building/{id}
POST /api/fcu/building

而校園資訊的 API 在此系統中較為特殊，校園訊息其中包括了校園新聞、校園活動、校園演講及校園公告，而這些資訊每日都會更新，是屬於動態資訊，所以 Server 端會每 3 個小時以網路爬蟲的方式向逢甲大學網站拿資料，並將資料寫進資料庫中，以達到隨時更新的效果。

校園訊息： GET /api/fcu/bulletin
POST /api/fcu/bulletin

4.8 資料庫

4.8.1 MongoDB

MongoDB 有 Schema-less 的特徵，採用類似「Table」的結構，但不需預先定義 Schema，每筆紀錄的欄位數量和結構也能不一樣。我們分成四種 Table：學院介紹(CollegeList)、大樓介紹(BuildingList)、景點介紹(LandscapeList)、校園訊息(BulletinList)。如下圖 4.12 所示。

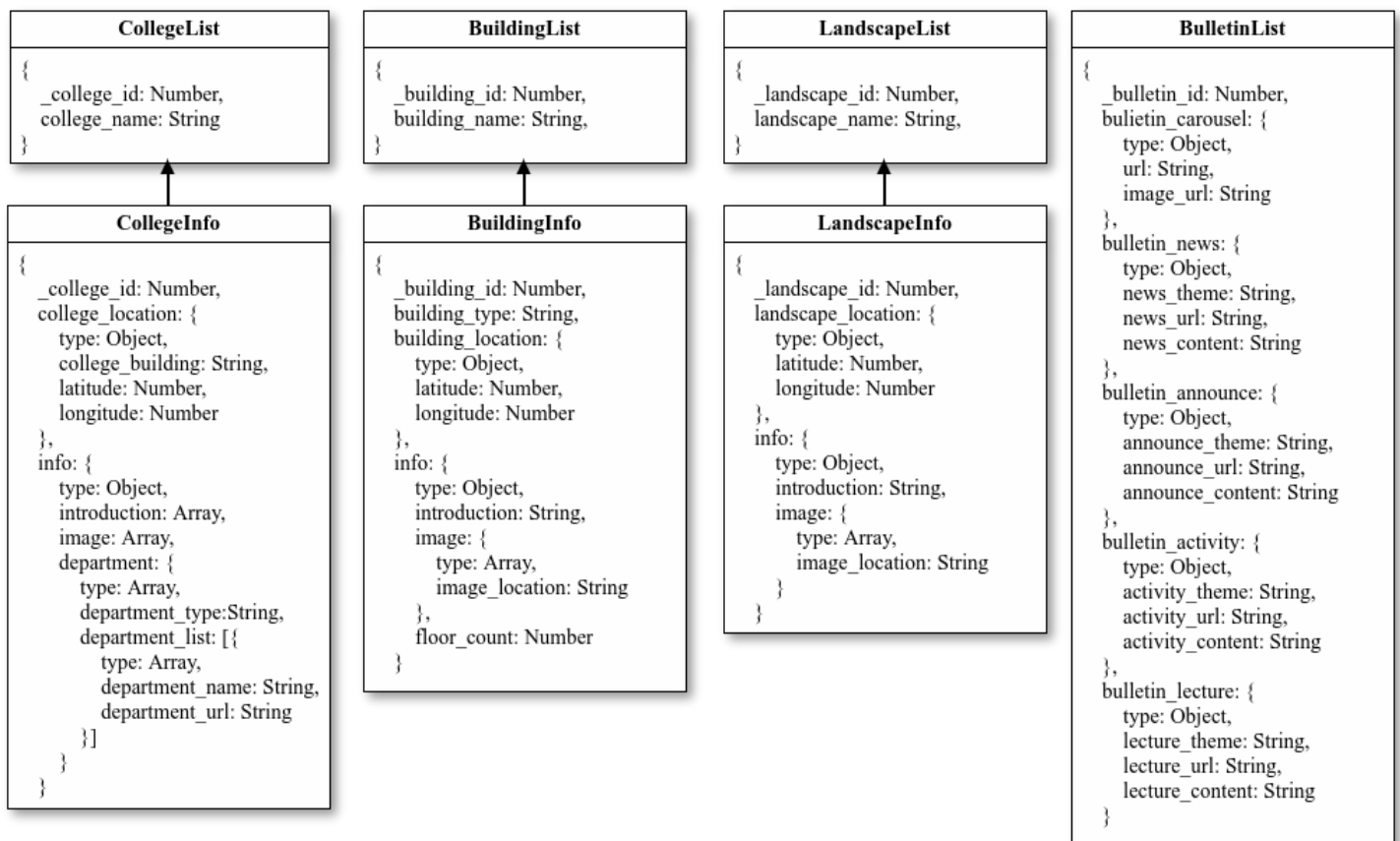


圖 4.13Mongo DB 結構設計

4.8.2 校園節點

在 Google Map 中，並非所有的路徑節點都涵蓋整個逢甲校園，而使用 Google Map API 所提供的路徑規劃時，甚至會規劃到校外的路線，因此本專題是使用 Google Map 已有的節點，搭配自行設置的校園節點（如圖 4.13 所示）再使用本系統所設計的路徑規劃演算法，使逢甲大學地圖更加完整，彌補整個校園節點不足之處。

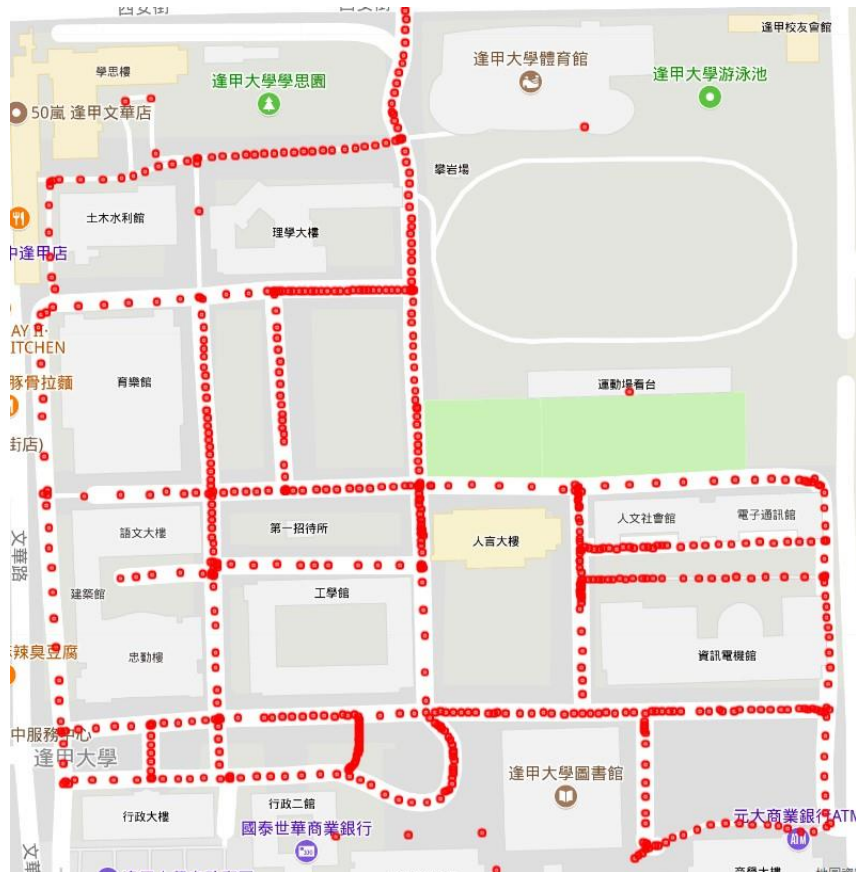


圖 4.14 校園節點分佈圖

4.9 路徑規劃演算法

在 2016 年發表的 “4-second rule” 中，統計出使用者在等待一個應用程式時，平均等待時間是 4 秒，基於我們是導覽系統，互動性更顯得重要，所以我們使用貪婪演算法(Greedy algorithm)，讓使用者在合理的時間內，獲得回饋感。

而貪婪演算法不是一種尋找最佳解的方法，其尋找方式為每一步都希望採取當前最好的狀態，能讓其演算效率提高，從起點開始，不斷的更新、改進每一步，直到無法改進為止。下圖為貪婪演算法的虛擬碼。

```
Algorithm GreedyAlgorithm
  solution.initialize
  while not isGoodEnough
    solution.improve()
  end
End of Algorithm
```

圖 4.15 貪婪演算法

4.9.1 路徑規劃

基於上述，我們取得每個參考點間的最短距離，以求出最合適的解，雖然不一定是最佳解，但是對於我們的系統而言，回饋是最重要的要素，因為本系統的目的在于讓使用者花合理的時間來走最適合的路，故使用貪婪演算法便能達到。

為了增加 UI 操作的彈性，我們讓使用者能夠自由點擊校內任何位置，即可藉由點擊的座標，計算出距離此座標最近的節點。

而路徑規劃的規則主要分成以下三大部份：

- (1) 針對目的地的座標，與使用者當下的座標，計算出當時的相對位置，符合正確方向的節點，將可能成為候選點。
- (2) 從所有有可能成為候選點中，選取距離目前座標最短的點。
- (3) 如果被選到的候選點，與使用者當下的點，相差距離超過 10m 時，將會被排除，改用另一種選取標準，選取方法為：將原本方向標準，改成只要有某一個方向成立，即可採用，雖然這個方式無法表現在大範圍的選取，卻可以解決小範圍的問題，例如對於要繞路才能抵達的目標，採用這個方式，可以消除跨越不可進入的錯誤點。



圖 4.16 路徑規劃圖

4.9.2 節點判斷法

利用目標點的座標，與當下座標，透過兩點距離公式，計算出經緯差的數值，再利用經緯度換算成距離的關係，再將單位轉換成公尺即可得知實際距離，而我們主要有以下兩種節點判斷的方式。

(1) 候選點抵達判定：

當下一候選選點的座標，與當下座標，距離小於 5m 時，候選點將會被換成再下一個候選點。

(2) 目的地判定：

當目的地的座標，與當下座標，距離小於 10m 時，將會判斷抵達目的地。

4.10 角度修正演算法

首先，假設自走車面向北方，我們即可利用以下三個點的座標做為餘弦公式的代入值

悉說明如下：

點 D：利用 3.7.1 路徑規劃演算法所計算出之節點，並利用 3.7.2 節點判斷法取得下一個節點座標。

點 S：利用 GPS 定位取得當下的座標。

點 P：點 S 向上延伸之比較點，點 P 之 Y 座標 K 為點 S 緯度向上延伸

藉由以上這三個點就可得知對於目前方向所要修正的角度，就能正確轉向下一節點，參考如下圖 4.16。

公式 3.8

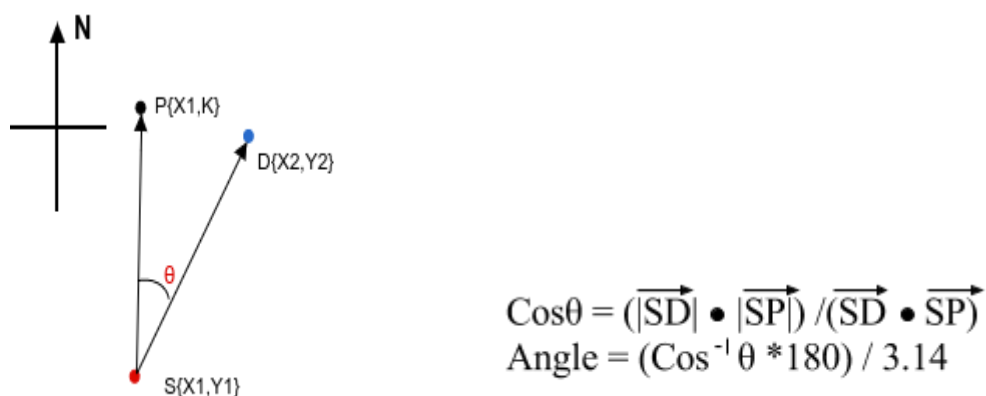


圖 4.17 角度計算及修正

4.10.1 方向感測器修正

方向感測器是一種三軸感測器(參考圖 4.17)，它使用指北針的原理，將北方設置為 0° ，以順時鐘方向遞增，所得範圍為 $0^\circ \sim 360^\circ$ ，而為了方便得知方位(即東半邊或西半邊)，所以將大於 180° 的角度值都減掉 360° ，順時針方向為正，逆時針方向為負，則修正範圍將介於 $-180^\circ \sim 180^\circ$ ，參考下圖 4.18。

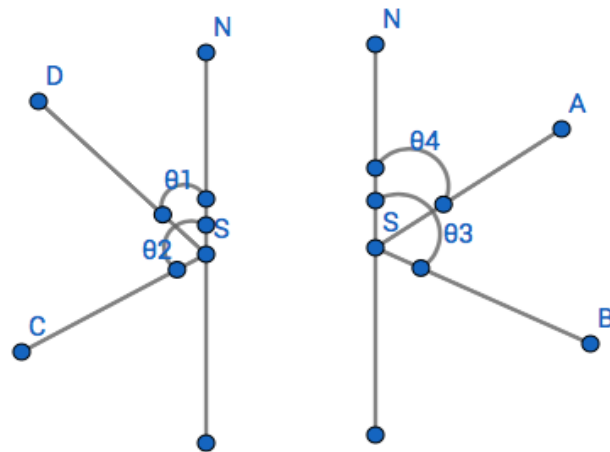


圖 4.18 三軸方向感測

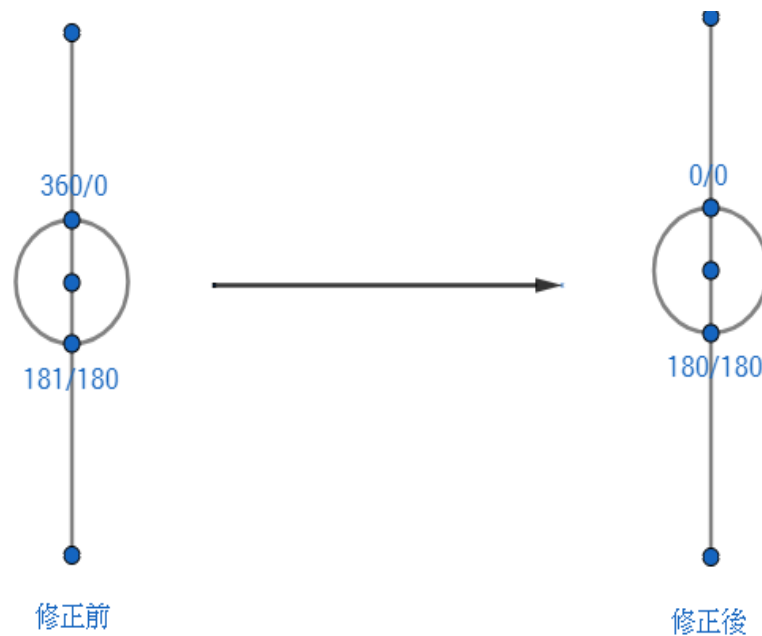


圖 4.19 方向感測數值修正

透過公式 3.8，將得到目的點與正北方的夾角，並將此夾角套入以下規則做判斷：

規則 3.8.1

當目標節點在東方時，若大於自身角度時（即 $0^\circ \sim 180^\circ$ ），將給正值，若小於的話，則給負值，可參考圖 4.19 右半部。

當目標節點在西方時，若大於自身角度的絕對值時，將給負值，反之給正值，可參考圖 4.19 左半部。

總結以上規則，即可得知轉向角度，正值為順時針轉，負值則反之，由此即可修正自走車方位。

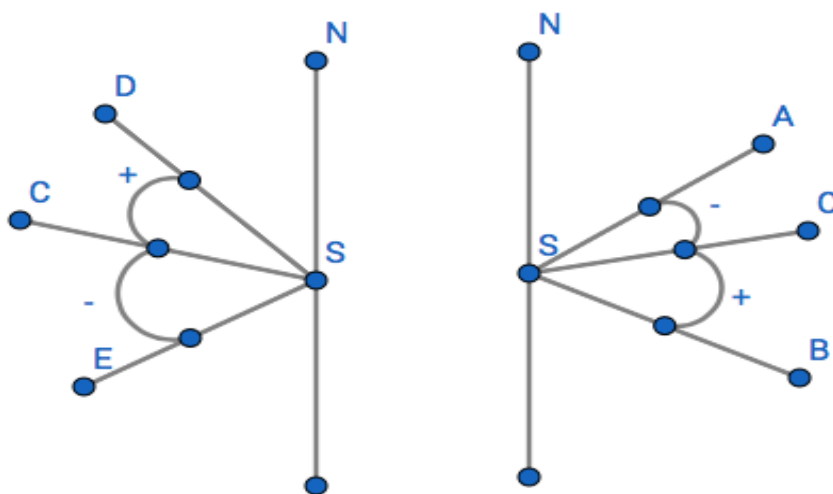


圖 4.20 目標節點判斷

4.10.2 磁力及重力加速度修正

此修正方式與方向感測器雷同，而其中不同的是，藉由磁力及加速度感測器，所獲得的值原本就是 $-180^\circ \sim 180^\circ$ ，所以在應用規則 3.8.1 時，不需要額外做處理；加速度感測器，是依據重力線的多寡，分別取分量，即可得到數值，範圍在 $9.8 \sim -9.8$ 。

(1) 加速度計感測原理：

加速度計的機制並非直接檢測加速度，而是檢測作用力所產生的加速度，故加速度計是檢測作用力施加在某一個軸向所產生的加速度。而加速度計也可以感測到重力所產生於物體的重力加速度，而地球上於同一地區所產生的重力並不會隨時間快速變化，故加速度計長時間具有可靠性。

而加速度計檢測的常見方法是電容檢測，而加速度與物體運動產生的電容變化有關。這種檢測技術精準度高、穩定度高、功耗低，也不容易受雜訊和溫度影響。

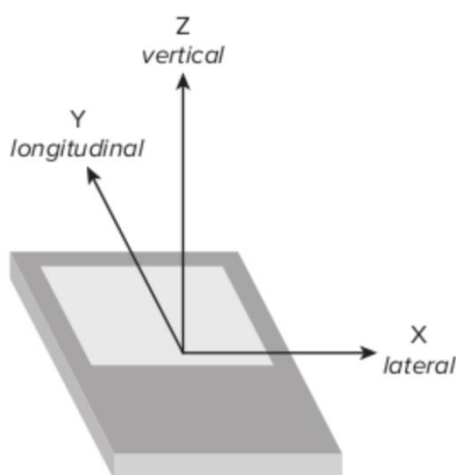


圖 4.21 三軸加速度計感測

(2) 磁力計感測原理

磁力計為用來測量環境磁力大小的元件，其磁場大小通常以高斯(Gauss)或特斯拉(Tesla)來表示，而地球本身有從地球南極到地球北極的磁場，故可以透過磁場投影到三軸來計算水平方向角度。而環境中還包含許多的干擾源，分類成：硬磁干擾(Hard iron)、軟磁干擾(Soft iron)，硬磁干擾如：永久磁鐵、電池這些固定強度的干擾；軟磁干擾如：電磁鐵、電量的變化等。

下面列舉四種不同影響之下，磁力計 X 軸與 Y 軸經歸一後畫出來的二維平面圖作說明：

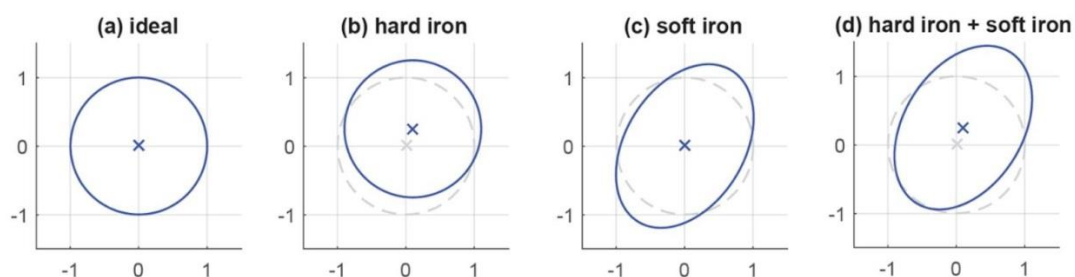


圖 4.22 磁力計四種不同影響之二維平面圖

圖(a)水平方向角度沒有誤差，即圓心在原點，為理想的圖形，圖(b)為受到硬磁干擾，整個圖形都有一個偏移量，圖(c)為受到軟磁干擾，整個磁場都產生了形變，使得整個圖形變成了橢圓形、圖(d)則是同時受到(b)、(c)兩種干擾的圖形，這也是最容易會檢測到的圖形。

4.11 自動避障

利用兩顆 hc-sr04 超音波感測器進行超音波避障，每秒發送一次超音聲波並回傳音波所花時間，依下列公式將其轉換為距離，若連續 3 次收到距離值為 1 公尺內，則視為障礙物，並立即進行角度修正直至距離超過 1 公尺。

超音波測距的方式是發射一個超聲波，當波遇到物體反射回來，再被測距儀偵測到反射波，利用來回時間與超聲波的速度算出距離，計算公式如下：

$$\text{距離} = (\text{聲波發射與接收時間差} * \text{聲音速度 (340M/S)}) / 2$$

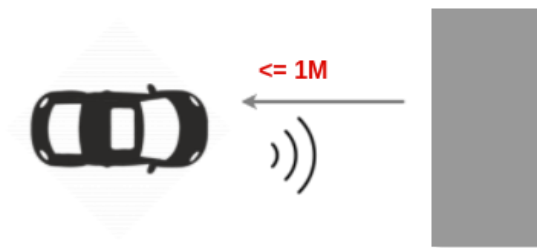


圖 4.23 自動避障修正前

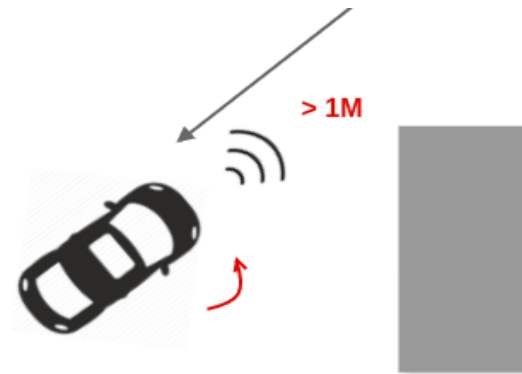


圖 4.24 自動避障修正前

Chapter 5 系統實作成果

5.1 系統功能與介紹

5.1.1 歡迎頁面

此頁面為本專題之系統歡迎頁面，使用者能透過頁面中間的五個按鈕分別前往不同功能的頁面。



圖 5.1 導覽系統歡迎頁面

5.1.2 導覽頁面

本導覽系統將校園資訊分成四個部分：學院介紹、大樓介紹、景點介紹、校園訊息。

學院介紹可瀏覽各個學院的所在位置及介紹，大樓介紹可瀏覽大樓介紹及大樓性質，景點介紹可瀏覽逢甲大學多處景點相關資訊，而校園訊息則可瀏覽逢甲大學每日相關新聞、活動...等等。

位於頁面左上角的逢甲大學 Logo 則可以返回歡迎頁面。

5.1.2.1. 學院、大樓、景點介紹



圖 5.2 導覽系統學院介紹

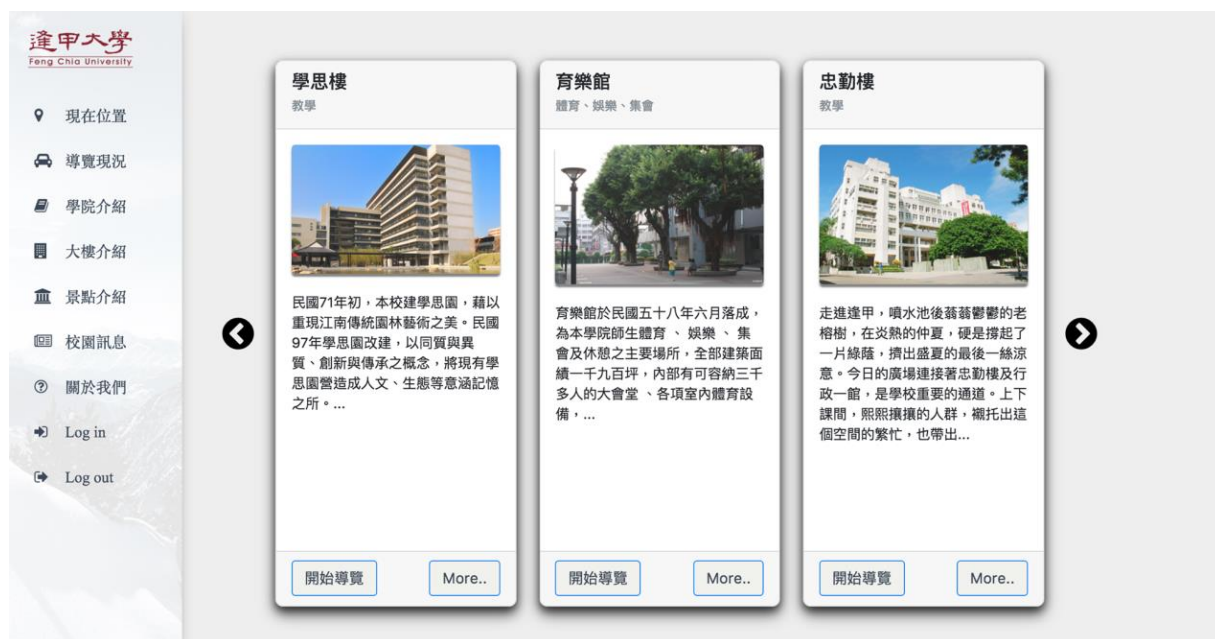


圖 5.3 導覽系統大樓介紹



圖 5.4 導覽系統景點介紹

5.1.2.2. 校園訊息

校園訊息主要是讓使用者能夠透過本系統得知逢甲校園的資訊，如校園新聞、校園活動、校園演講及校園公告。



圖 5.5 校園訊息畫面

使用者透過圖 5.5 中的 More 按鈕便能顯示如下圖所示資訊，能夠得知更多條訊息標題，點擊標題後，便會將頁面導至此篇訊息內容。

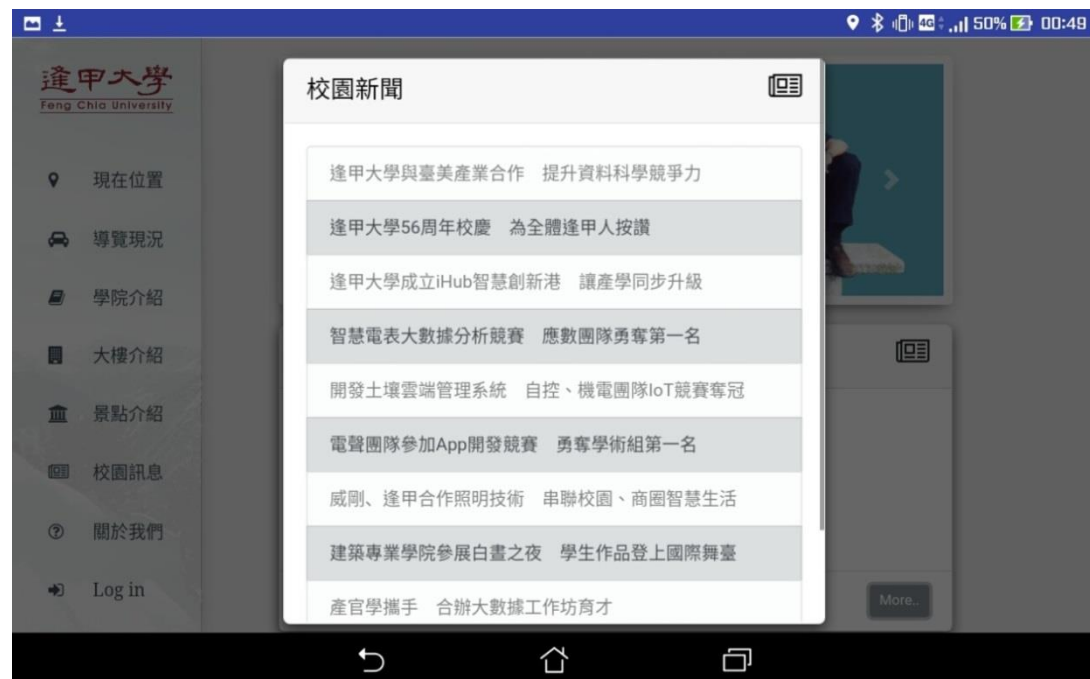


圖 5.6 更多校園新聞頁面

使用者點擊上圖 5.6 中的任一標題後，便能將頁面轉至資訊內容，而使用者只要點擊返回鍵即可回到主系統。



圖 5.7 校園新聞內容頁面

5.1.2.3. 導覽介面(Google Map Dialog)

當使用者點選「現在位置」時，會進入到一個導覽的區塊，區塊上會顯示 Google Map 逢甲校園地圖及使用者當下位置，使用者可以點選校園的建築物，點選建築物之後，會規劃一條可以抵達目的地的路線。

若使用者改變心意點選其他的建築物，地圖會即時清除先前的路線，並且依據剛才點擊的座標，規劃出新的路線，當使用者點擊「開始」標誌的按鈕，地圖就不會再接受任何的觸發，除非自走車抵達目的地，假設使用者想提早結束導覽，可以點擊“停止”標誌的按鈕，自走車將會停止，回歸到最初的狀態。

下圖為當下位置介面：

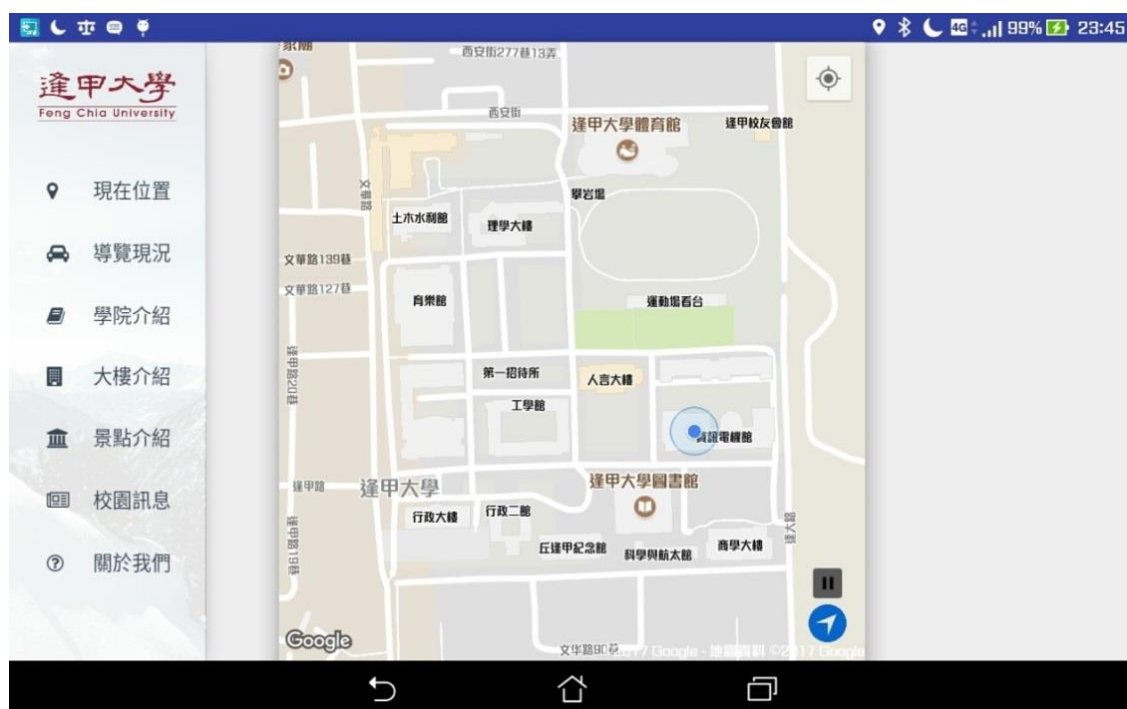


圖 5.8 現在位置介面

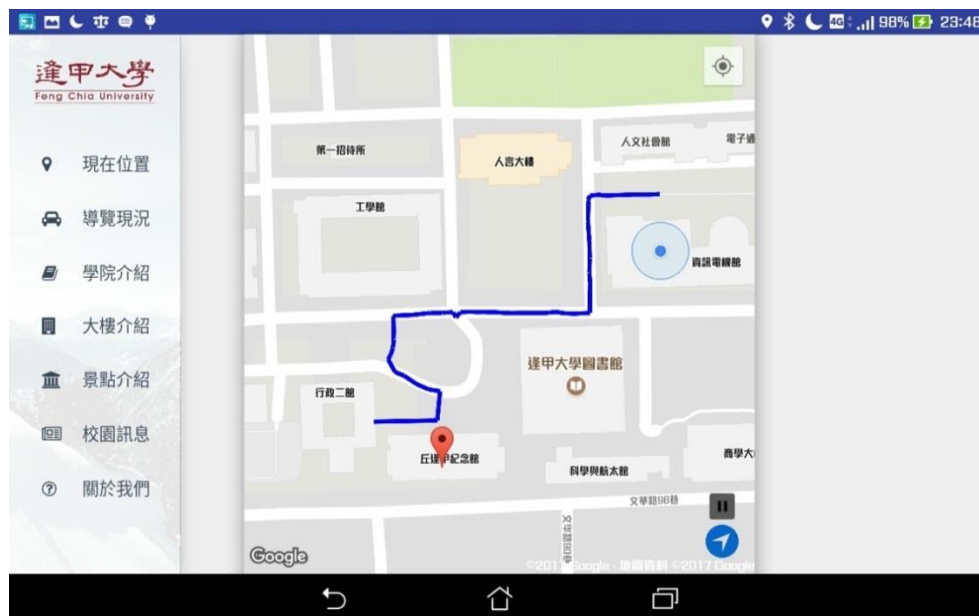


圖 5.9 路線規劃介面

5.1.3 車體結構設計

自走車的製作我們是購買簡單的遙控車，並拆去除了車體結構的部分來改裝，例如：車體上蓋、電子變速器，留下基本的底盤。

而底盤結構的部分，有電源供應器，基本的轉彎結構如：輪軸、輪座...等等，輪子的前後轉與左右轉，分別會透過三個直流馬達與齒輪結構來達到控制，而我們以 Raspberry Pi3 控制板來取代源有的電子變速器，達到更彈性以及可依我們所需來操作。

中盤結構的部分，我們購買了三塊木板，並透過工具切割成所需的大小，以長螺絲釘將其底盤的延伸結構固定，再將其以更為長的兩片木板作延伸，上方則架設一個橫板及平板固定座，以固定我們的平板。

而下表為本專題自走車所需材料表。

表格 5.1 自走車材料表

材料項目	數量
Raspberry Pi 3 控制板	1
超音波感測器	2
馬達驅動板	2
DC Motor	3
10cm輪胎	4
紅外線感測器	1

自走車的結構與實品依下圖所示。



圖 5.10 導覽系統自走車側面照



圖 5.11 導覽系統自走車正面照

Chapter 6 結論

6.1 結論

本專題提出的智慧導覽自走車系統，硬體方面，車體為四輪驅動車，是從一般遙控車改裝的車體，內部主要含有：控制板 Raspberry Pi3 作為系統中樞、馬達控制板 L298N 兩塊用以控制三顆直流馬達來操作前後輪及轉彎、超音波感測器 HC-SR04 兩顆用以閃避前方障礙物，全部都已建置完成。軟體方面，本導覽系統前端 RWD 導覽網頁、Single Page Application 呈現，後端 Nodejs 伺服器定時更新資料並可與前端串接，資料庫 MongoDB 儲存抓取回來的資料，上述實作也均已完成，完整的導覽體驗已可以透過平板進行。

6.2 未來展望

本專題系統在上述所提到軟體與硬體方面均實作完成，盼望能夠試著與深度學習做結合。深度學習在許多方面的不同應用上，其高效能令人驚嘆，常見如影像辨識，於本系統搭載上攝像頭後，便可嘗試做影像辨識的訓練相關研究，可以應用在校園內部辨識人物等用途，或也可嘗試做語音辨識方面的研究，只要於系統上搭建麥克風即可，期待往後能往這一步做相關深究。

相關文獻

- [1] “每日頭條,”[Online].Available:
<https://kknews.cc/zh-tw/digital/zxg98x1.html>
- [2] “維基百科,”[Online].Available:
<https://zh.wikipedia.org/wiki/Wikipedia>
- [3] “Baidu 百科,”[Online].Available:
<https://baike.baidu.com/item/%E5%9C%B0%E7%90%83%E7%A3%81%E5%9C%B>
- [4] “CodeData,”[Online].Available:
<http://www.codedata.com.tw/database/mongodb-tutorial-1-setting-up-cloud-env/>
- [5] “泡在網上的日子,”[Online].Available:
<http://www.jcodecraeer.com/a/anzhuokaifa/androidkaifa/2012/1009/424.html>
- [6] “Share! 樂讀分享,”[Online].Available:
<http://thats-worth.blogspot.tw/2014/04/arduino-pwm-arduino-pulse-width.html>
- [7] “傑瑞窩在這,”[Online].Available:
<https://jerrynest.io/app-android-webview/>
- [8] “Developer,”[Online].Available:
<https://developer.android.com/reference/android/hardware/SensorManager.html>
- [9] “KitSprout,”[Online].Available:
<http://kitsprout.logdown.com/posts/335386>
- [10] “RUNOOB.COM,”[Online].Available:
<http://www.runoob.com/bootstrap/bootstrap-intro.html>
- [11] “阮一峰的網路日誌,”[Online].Available:
http://www.ruanyifeng.com/blog/2014/05/restful_api.html
- [12] “iT 邦幫忙,”[Online].Available:
<https://ithelp.ithome.com.tw/articles/10160709>
- [13] “IBM,”[Online].Available:
<https://www.ibm.com/developerworks/cn/opensource/os-eclipse-stlcdt/>
- [14] “Eclipse,”[Online].Available:
<https://www.eclipse.org/cdt/>
- [15] “EET 電子工程專輯,”[Online].Available:
http://archive.eettaiwan.com/www.eettaiwan.com/ART_8800701519_480502_TA_e52a9104.HTM
- [16] “史丹利部落格,”[Online].Available:
<http://stenlyho.blogspot.tw/2008/09/android-architecture.html>
- [17] “凌群電子報,”[Online].Available:

http://www.syscom.com.tw/ePaper_Content_EPArticleDetail.aspx?id=247&EPID=179&j=3&HeaderName=%E7%A0%94%E7%99%BC%E6%96%B0%E8%A6%96%E7%95%8C

[18] “陳鍾誠的網站,”[Online].Available:

<http://ccckmit.wikidot.com/ga:architecture>

[19] “Google Maps API,”[Online].Available:

<https://developers.google.com/maps/documentation/android-api/signup?hl=zh-tw>

[20] “智慧生活科技專業社群,”[Online].Available:

<http://cheng-min-i-taiwan.blogspot.tw/2013/04/google-maps-android-api-v2-android.html>

[21] “R2 的資訊筆記,”[Online].Available:

<https://sites.google.com/site/r2internotes/android02/webview01>

[22] “Cooper Maa,”[Online].Available:

<http://coopermaa2nd.blogspot.tw/2012/09/hc-sr04.html>

[23] “廣華電子,”[Online].Available:

<http://shop.cpu.com.tw/product/46920/info/>

[24] “四處流浪的阿基,”[Online].Available:

<http://expect7.pixnet.net/blog/post/37789922->

[25] “The Front Row,”[Online].Available:

<http://blog.zhusee.in/post/56286985943/web-speech-api-part-i-using-speech-synthesis-to-make-browser>

[26] “Slide Share:NodeJS 基礎教學&簡介,”[Online].Available:

<https://www.slideshare.net/xdxie/nodejs-15251110>

[27] “MongoDB,”[Online].Available:

<http://mongodbcanred.blogspot.tw/2015/01/mongodb.html>

[28] 石豐瑞,“校園導覽系統以國立成功大學為例,”[Online].Available:

<http://handle.ncl.edu.tw/11296/ndltd/12872712994554634416>