

The Medical Facilities for Google Maps and Specific Information on Android

Chao-Tung Yang

Department of Computer Science
Tunghai University, Taichung
City, 40704, Taiwan (ROC)
e-mail : ctyang@thu.edu.tw

Yen-Yu Chu

Department of Computer Science
Tunghai University, Taichung
City, 40704, Taiwan (ROC)
e-mail : g97350062@thu.edu.tw

摘要

使用 Google 提供的 Android SDK 開發自己所需程式，已經成為國內的程式設計師們的目標。然而手機提供的 Google Map 這項應用程式，使用功能上相當有限，只提供了搜尋、路線、地圖模式、我的位置、加入 Google 定位等功能，但是，使用此地圖卻無法讓我們得到有目標性的目的，像是需要時卻往往令人措手不及的醫院藥局的詳細資料。像是目睹了交通事故時，常會猶豫該打 119 還是 110，但如果手機中可及時查閱距離最近的醫院電話，即可免去這一項憂慮。亦或是旅遊途中忽然想到藥品配備不足，即可到距離目前最近的藥局補充所需。本研究實作的重點在於：提供使用者基本的 Google map 功能，並隨時提供顯示所在周邊的各大綜合醫院及衛生署認可藥局的資料和明顯的地標，以及連結的網址，好讓使用者可直接進入網站查詢醫生看病時間或預約掛等。對於不擅長使用現代科技產品的長輩們使用起來更輕鬆、更安心且快速的得到所需的資訊。

關鍵詞: Google Map、Android。

1. 前言

在全球資訊發達的情況下，全世界的手持裝置產量已經大幅超越了個人電腦，儘管已目前世界上的手持裝置由於受制於不同協定或是硬體裝置的情況下，以致於並於所有手持裝置皆能夠上網，而即使能夠上網也受限於不同的協定使得速率有所不同，但相較於個人電腦的產量，全世界的手持裝置的產量仍然不斷的提昇當中，並且在其消費市場上智慧型手機已漸漸的被廣為接受，且每年的產量亦逐步提高。而目前智慧型手機其消費市場可以分為四種作業系統 Symbian、Windows、iPhoneOS、Android。而其中又以 Android 作業系統因為其

為 Google 所推動的作業系統，而 Google 集合了各業者共同組成開放式手機平台聯盟(Open Handset Alliance)，又其 Android 系統為一基於 Linux 核心的軟體平台和作業系統，由於 Linux 採 Apache 架構成為自由軟體和開放原始碼有名的例子，所以 Android 系統幾乎完整保留了 Linux 原有的特性，結合了智慧型手機的概念，且本研究專注於 Google Map 與醫療服務的結合，故採用 Android 作業系統做為我們開發的軟體平台，而我們所作的應用主要在於可以讓使用者能在手機中輕鬆查看醫院及藥局的所在地點及詳細資訊，可省下輸入關鍵字查詢及觀看資訊評價等麻煩的手續。

2. 背景

2.1 Linux

Linux 為一類 Unix 作業系統的統稱，Linux 作業系統的核心亦稱之為 Linux，實際上一般人則已習慣稱 Linux 當作 Linux 核心。絕大多數 Linux 核心的作業系統使用了大量的 GNU(GNU's Not Unix)軟體，包含了 shell 程式、工具、程式函式庫、編譯器及其它工具等軟體，目前使用 GNU 專案各種工具和資料庫的作業系統(GNU/Linux)，基於這些零件的 Linux 軟體被稱作 Linux 套件。一般而言，一個 Linux 發行套件包含大量軟體，其中包括軟體開發工具(例如 gcc)、資料函式庫(例如 PostgreSQL、MySQL)、Web 伺服器(例如 Apache)、X Window、桌面環境(例如 GNOME 和 KDE)、辦公套件(例如 OpenOffice.org)、腳本語言(例如 Perl、PHP 和 Python)等等。Linux 核心最初是為英特爾 386 微處理器設計的。現在 Linux 核心支援從個人電腦到大型主機甚至包括嵌入式系統在內的各種硬體裝置。現在，Linux 已經成為了一種受到廣泛關注和支援的作業系統。包括 IBM 和惠普、戴爾在內的一些電腦業巨頭也陸續支援 Linux，並

且成立了一些組織支援其發展，如 Open Invention Network (OIN) 成員有 IBM, SONY, NEC, Philips, Novell, Red hat 等) 購買了微軟專利，允許任何個體以開放的原則使用。很多人認為，和其他的商用 Unix 系統以及微軟 Windows 相比，作為自由軟體的 Linux 具有較低的建構成本，高安全性，更加可信賴等優勢。

2.2 Open Handset Alliance

開放手機聯盟(OHA)是由 Google 結合其它 33 家業者(Aplix、Ascender Corporation、Audience、Broadcom、China Mobile、eBay、Esmertec、HTC、Intel、KDDI、LivingImage、LG、Marvell、Motorola、NMS Communications、Noser、NTT DoCoMo、Nuance、Nvidia、PacketVideo、Qualcomm、Samsung、SiRF、SkyPop、SONiVOX、Sprint Nextel、Synaptics、TAT - The Astonishing Tribe、Telecom Italia、Telefónica、Texas Instruments、T-Mobile、Wind River)所組成，Google 期望，手機開發商不必再為專用的作業系統或手機品牌設計獨有應用程式，而能夠專注於研發工作，設計更貼心的手機上網服務

2.3 Android

Android(機器人)是 Google 以 Linux 為基礎的作業系統平台，與 Windows Mobile(Microsoft 提供)、iPhoneOS(Apple 提供)等手機作業系統處於同一級別。除了作業系統之外，它還包含中介軟體(Middleware)負責硬體與應用程式之間的溝通，並提供 Android SDK/NDK 應用軟體開發套件，允許嵌入式系統的開發人員使用它來開發 Android 平台的應用程式。Android 的所有應用程式用 Java 語言撰寫，然後透過 Android Runtime 的 Dalvik 虛擬機器來轉換執行 DX Bytecode，應用程式的使用者圖形介面(Graphic User Interface)部分則透過 XML (eXtension Markup Language)程式來設計。而 Google 2009 年六月份釋出的 Android 1.5 NDK(Native Development Kit)，讓開發人員可透過 C 或 C++ 等程式語言撰寫 Android 應用程式。更通俗的說，Android 並不是一支手機，是在手機上使用的作業系統，軟體是免費的，原始碼也完全開放

2.4 Google Map

Google Map 為 Google 公司所提供的電子地圖服務，包括局部詳細的衛星照片。目前已提供三種視圖：一是向量地圖(傳統地圖)，可提

供行政區和交通以及商業訊息；二是不同解析度的衛星照片(俯視圖，與 Google Earth 上的衛星照片一樣)可讓使用者更加確定自己的所在位置；三是後來加上的地形視圖，可以用以顯示地形和等高線。

Google Earth 為其姐妹產品，是桌面應用程式，可在三維模型上提供街景和更多的衛星照片以及 GPS 定位的功能(需付費)，但沒有前述的向量地圖及地形視圖。2005 年 6 月 20 日，Google Maps 的覆蓋範圍，從原先的美國、英國及加拿大，擴大至全球；2005 年 9 月，Google 和北京圖盟科技有限公司 (Mapabc) 合作出版中國大陸地區的 Google 本地，後來，Google 本地重新命名為 Google 地圖。台灣的圖資則是由勤崙科技所提供，並於 2007 年 10 月 11 日推出台灣版地圖。

2006 年，Google 發佈了一個 java 程式稱為 Google Maps for Mobile，可以用在 Java-based 的手機上。直至 2008 年 12 月 15 日，Google Maps for Mobile 支援以下平台：Android、iPhoneOS (iPhone/iPod Touch)、Windows Mobile、Nokia Symbian S60 3rd、Symbian OS(UIQ V3)、BlackBerry、Palm OS 5 或更新、使用 Java 平台的手機(MIDP 2.0 或更新)，如 Sony Ericsson K800i 等。

目前 Google Map 已開發了「街景視圖」服務，使用者可以通過街道上的視角查看街景(如公車站或招牌)。但是該服務卻有侵犯個人隱私的嫌疑，因為街景視圖的照片是由 Google 的甲蟲車在街上隨機拍到的，因此街上行人的動作、停車的車牌號碼都是一清二楚。因此目前僅有北美部分地方開通了「街景視圖」服務。此外 Google Map 尚有其它功能包括路線查詢，規劃路徑等功能，可以顯示兩個地點之間的距離和行車時間。

2.5 JAVA

Java，是一種可以撰寫跨平台應用軟體的物件導向的程式語言，起初目標設定在家用電器等小型系統的程式設計語言，來解決諸如電視機、電話、鬧鐘、烤麵包機等家用電器的控制和通訊問題。後來 Java 伴隨著網際網路的迅猛發展而發展，逐漸成為重要的網路程式語言。

Java 程式語言的風格十分接近 C++ 語言。繼承了 C++ 語言物件導向科技的核心，Java 捨棄了 C++ 語言中容易引起錯誤的指標(以參照取代)、運算符過載(operator overloading)、多重繼承(以介面取代)等特性，增加了垃圾回收

器功能用於回收不再被參照的物件所佔據的內部記憶體空間。Java 不同於一般的編譯執行的電腦語言，它首先將原始碼編譯成位元組碼 (Bytecode)，然後依賴各種不同平台上的虛擬機器來解釋執行位元組碼，從而實作了「一次編譯、到處執行」的跨平台特性。與傳統程式不同 Sun 公司在推出 Java 之際就將其作為一種開放的科技。全球數以萬計的 Java 開發公司被要求所設計的 Java 軟體必須相互相容。

「Java 語言靠群體的力量而非公司的力量」是 Sun 公司的口號之一，並獲得了廣大軟體開發商的認同。這與微軟公司所倡導的注重精英和封閉式的模式完全不同。

Sun 公司對 Java 程式語言的解釋是：Java 程式語言是個簡單、物件導向、分佈式、解釋性、安全與系統無關、移植性、高性能、多執行緒和動態的語言。

3. 系統設計與實作

3.1 架構

基本上 Android 系統架構中，最上層的 Application 與 Application Framework 都是使用 Java 語言寫成的，而中間層的 Dalvik 虛擬機器與 Libraries 則採用 C/C++ 語言寫成，最底層的 Linux 系統核心則採用 C 語言寫成。因此底層的 Linux 作業系統已經被 Android 上層的應用程式包裝起來，Android 使用者與程式開發人員僅會接觸到最上層的應用程式與應用架構，而 Java 語言所開發的應用程式則會透過 Google 所開發的 Dalvik 虛擬機器，讓所開發的 Java 程式可以在 C 語言寫成的 Linux 平台上正常執行。如果深入看其系統架構，總共包含四個主要層次，如 Fig. 1 所示。

- Application (應用程式): 最上層為 Java 應用程式，例如 Google Maps、Email、即時通、計算機、瀏覽器、通訊錄、MP3 播放器... 等等應用程式。
- Application Framework (應用軟體架構): 所有 Android 核心應用程式 framework API 的總集合，主要目的是讓程式開發者可以方便取用這些常用的應用程式設計架構，以便快速地開發 Android 應用程式，而這些常見的 API 例如：顯示功能(包含：訊息方塊、條列訊息、按鈕、內嵌式瀏覽器等)、訊息提供功能(圖形、地方化字串、佈局檔案等)、提示訊息(主要用來顯示警告訊息，特別是

當使用者操作錯誤時)... 等等。

- Library (函式庫): 由大多數開放原始碼的函式庫所組成，例如：標準的 C 函式庫 Libc、OpenSSL、SQLite... 等等，還有網頁函式庫 WebKit 負責 Android 網頁瀏覽器的運作，另外還 2DSGL/3D OpenGL 圖形與多媒體函式庫分別支援各種影音與圖形檔的播放，例如：MPEG4, H.264, MP3, AAC, AMR JPG, and PNG 等多媒體檔案格式。
- Android Runtime: 提供 Android 特有的 Java 核心函式庫與可轉換 Java 執行碼 Bytecode 為 Android 專屬 Dalvik Executable(.dex) 檔案格式 Dalvik 虛擬機器，讓程式設計師可以透過 Java 語言撰寫 Android 應用程式，並 Android 平台上同時執行多個 Java 應用程式。
- Linux Kernel (系統核心): 採用 Linux Kernel 2.6 版，負責硬體的驅動程式、電源管理、系統安全、記憶體管理、行程管理、網路功能(TCP/IP、UDP、WiFi)、檔案系統(yaffs2, vfat, msdos... 等)、輸入與輸出裝置... 等等，以及在軟體與硬體之間建立起一個抽象層 (abstraction layer) 溝通介面，讓軟體與硬體間可以相互溝通與運作。

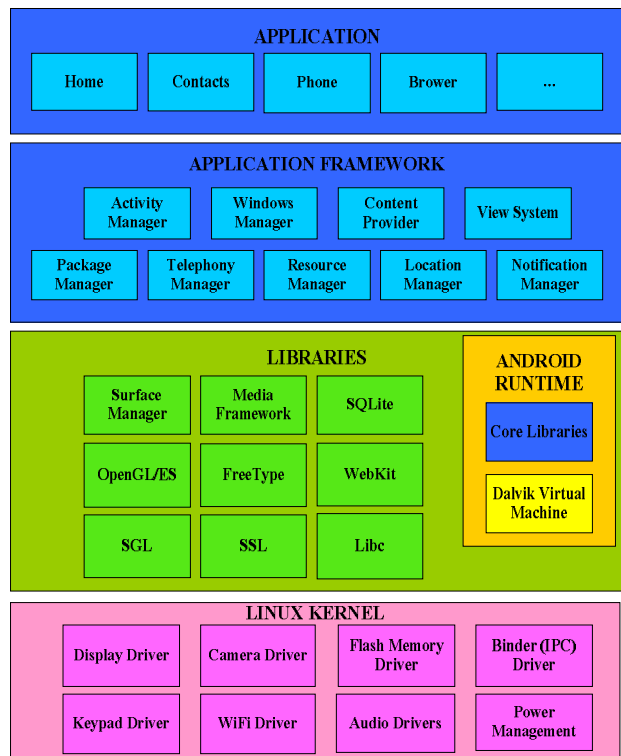


Fig1. 系統架構

3.2 開發程式架構

Android 程式設計專案的檔案架構，不管是使用 Eclipse 產生專案，或使用 activitycreator 產生一個 Android 程式設計專案，其架構如 Table.1 所示。

其 Androidmanifest.xml 檔案定義了設計應用程式是該有的功能導向，清楚描述應用程式主要功能是什麼，能處理什麼樣的資料以及甚麼時候該執行這些動作，而這些功能導向由下列四個元素所定義(並非每個程式都包含這四大元素)，並以不同的 XML 標籤名稱定義在 Androidmanifest.xml 檔案中，：

- Activity (活動) - 使用<Activity>標籤

在 Android 手機系統上，Activity 通常就代表一個應用程式的顯示畫面。例如：顯示一個 Email 通訊錄清單畫面就是一個 Activity，而編輯通訊錄畫面就是另一個 Activity。

- Intent 與 Broadcast Receiver (意圖與廣播接收) - 使用<receiver>標籤

在 Android 手機上操作任何一個應用程式都有其目的，例如：使用 FTP 檔案傳輸程式下載或上傳檔案，使用網頁瀏覽器瀏覽網頁，這種資料取得的過程在 Android 程式設計中，就將它稱為 Intent Receiver(意圖接收)。而當設計的應用程式產生 Intent 後，目的是要接收對方傳來的資料，則必須產生一個 Broadcast Receiver(廣播接收)，來處理接收到的資料。

- Service (服務功能) - 使用<Service>標籤

當 Android 應用程式並不需要顯示畫面，但卻要長時間在背景執行時，就是具備 Service 服務功能的應用程式，例如：MP3 播程式。

- Content Provider (內容提供) - 使用

<Provider>標籤應用程式所進行的動作是讓使用者可以儲存它們的資料或文件，例如：一個 SQLite 資料庫程式就是屬於 Content Provider 功能的程式。

此外，AndroidManifest.xml 檔案中還可以進一步使用<uses-permission>與<permission>標籤設定應用程式的使用者執行權限，以及使用<instrumentation>標籤設定應用程式原始碼的組成與測試功能。更完整的標籤語法，如 Table.2 所示。

Table.1 程式設計專案檔案結構表

專案目	專案子目錄	資源檔案	檔案用途
Project/			
	AndroidManifest.xml		應用程式設定檔
	src/	*.java	Java 原始碼檔案
	res/anim/	*.xml	動畫設定檔
	res/drawable/	*.png, *.jpg, *.gif	應用程式圖檔
	res/layout	main.xml	視窗佈局檔
	res/values	arrays.xml colors.xml dimens.xml string.xml styles.xml	陣列定義檔 繪圖顏色定義檔 尺寸定義檔 文字資源檔 樣式定義檔
	res/xml	*.xml	程式執行時所需讀取的 XML 檔案
	res/raw	*.raw	程式執行時所應用的資源檔，例如：音效檔

Table.2 Manifest XML 標籤語法層級表

Manifest	應用程式標籤	功能標籤	Intent filter	intent 標籤
<manifest>				
	<application>			
		<activity>		
			<intent-filter>	
				<action>
				<category>
				<data>
			<meta-data>	
		<receiver>		
		<service>		
		<provider>		
	<uses-permission>			
	<permission>			
	<instrumentation>			

3.2 程式運作原理

Android 程式運作原理：幾乎所有的 Android SDK 精髓在 Application Framework 中，可參見 Figure1，它提供大量的 Java 類別供使用者開發程式。當我們在 Eclipse 中執行 Run/RunConfigurations 時，Java 程式碼不會被編譯成常見的 Java Bytecode，而是 Eclipse/Ant 編譯成 Dalvik Bytecode(*.dex)，然後再和其他資源檔打包成 APK 套件，最後在 Runtime 時，透過 Android Code Libraries 這個很大的 Java 類別總集 JAR(Java ARchive)檔案與上層 Java 程式溝通，而 Dalvik 虛擬機器則解讀 dex 檔案，與下層使用 C 語言寫成的 Linux Library 函式庫溝通，然後由最底層的 Linux Kernel 產生一個行程(process)執行該程式。

3.3 Activity Life Cycle(生命週期)

每一個 Android 應用程式執行中，對底層的 Linux Kernel 而言是每一個單獨 Process，但是對 Android 系統而言，因為侷限於手機畫面的大小與使用方便性考量，Android 使用 Activity 的觀念，將這些執行中的應用程式分為五大類，分別是：foreground process(前景模式)、visible process (可見模式)、background

process (背景模式)、empty process (空白模式) 與 service process (服務模式)。系統會依照五大類型判斷應用程式 Activity 是屬於哪一個類型，給予不同的 Activity Life Cycle，運作圖如 Figure.2。

3.4 Android 應用程式運作模型

應用程式運作過程中，首先將所開發的應用程式打包成 android package.apk 應用程式套件，安裝到 Android 系統後，將它稱為 Task(工作)，也就是 Application 應用程式；當應用程式是真正在底層的 Linux Kernel 執行時，被稱之為一個 Process(行程)；而每一個行程底下允許有一個或多個 Threads(執行緒)同時執行，但是建議所開發的程式保持 Signal Thread 單一線程的設計概念，應用程式運作模型如 Figure.3 所示。

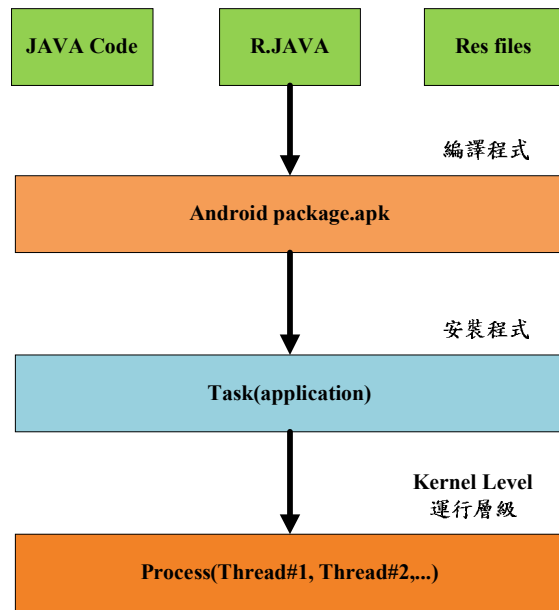


Fig.3 Android 應用程式運作模型

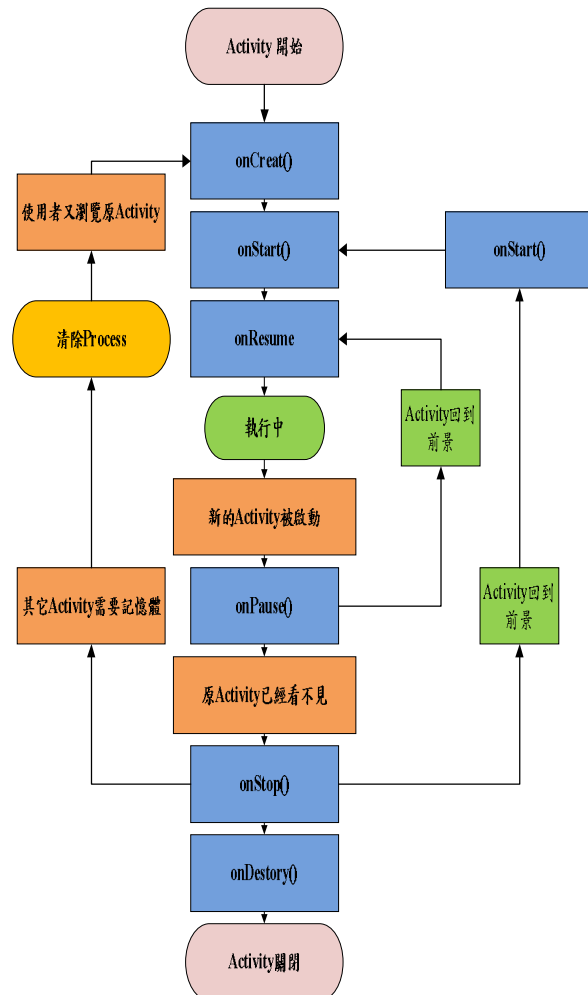


Fig.2 Activity Life Cycle 運作圖

4. 研究方法與實作

4.1 地圖控制

Google Map 是 Android 平台標準功能之一，使用 MapView 物件類別 com.google.android.maps.MapView 就可以將 Google Map 嵌入 Android 應用程式中。但是因為授權限制，在使用 MapView 開發應用程式前，必須先向 Google 申請一組經過驗證的 Google Map API key，才能正常在手機上使用 Google Map 服務。

● 查詢 debug.keystore 位置

打開「Eclipse」開發環境選單，選擇「Windows > Preferences」選項，開啟「Preferences」選單，在視窗左側點選「Andorid > Build」選項。此時右側畫面中「Default debug keystore」所填的路徑就是「debug.keystore」所在的位置，如 Fig. 4。

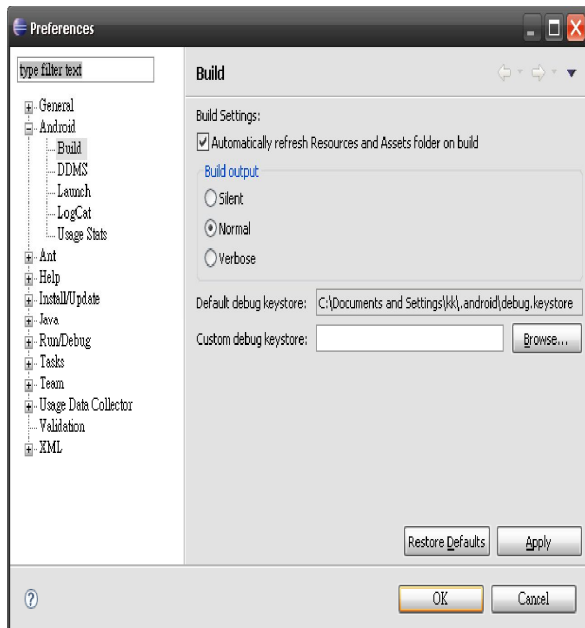


Fig. 4 debug.keystore 位置

● 取得 MD5 指紋訊息

開啟命令提示字元，使用 keytool 指令，經由「debug.keystore」檔案來產生一組「MD5」格式的指紋(Fingerprint)訊息，作為申請開發金鑰時所需的證書(certificates)，如 Fig. 5 所示。

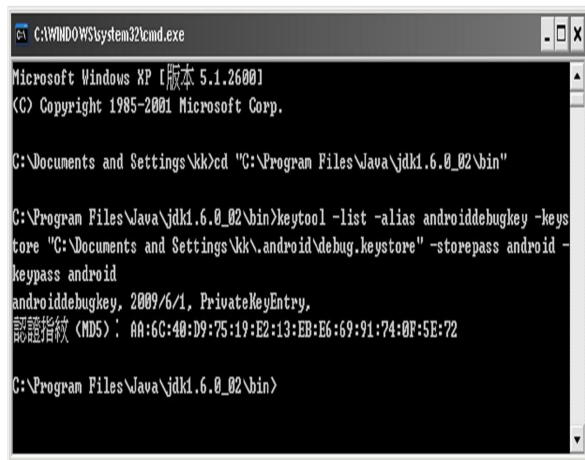


Fig. 5 MD5 指紋訊息

● 驗證取得 API key

開啟「Google 地圖」網頁申請開發金鑰。輸入取得的 MD5，得到金鑰後，加入要開發專案的 xml 檔案中，如 Fig. 6 所示。



Fig. 6 Google 地圖 API Key 申請

4.2 建立選單(MENU)

Android 每個頁面對應到一個 Activity，每個 Activity 都會有一個獨立的選單，而每個選單的選項都包含建立選單與處理選單動作兩個部份。

在 Android 手機或模擬器上，按下硬體的「Menu」鍵時，所彈出的選單列是由「onCreateOptionsMenu」函式來定義。而在 Android 應用程式的選單列上選擇相應的選項後，則是依賴「onOptionsItemSelected」函式，來負責處理選單列中各選項所個別對應的動作。

增加選單選項的格式為「menu.add(0, identifier, 0, string)」，identifier 為定義的名稱，以及判斷對應動作時的選項名稱。還可使用「.setIcon(R.drawable.icon_name)」為每個選項加上各自的圖片，如 Fig. 7 所示。



Fig. 7 主要選單設計

4.3 醫院藥局端設計

4.3.1 設計對話框

撰寫負責處理對話框的

「openOptionsDialog」函式，建立一個「AlertDialog」對話框類別實體，再以「AlertDialog」呼叫以下方法如 Table. 3。

- 資訊對話框- 在對話框程式中加入「setMessage」顯示資訊。專案中用來顯示醫院和藥局的定址電話，以及現在位址的經緯度。
- 陣列對話框- 在對話框程式中，使用「setItems」加入陣列內容及其對應的動作，以供使用者選擇不同的功能。專案中用於顯示「醫院/藥局」、「衛星/街道圖」以及選擇資訊傳送方式「信箱/簡訊」等等。

Table.3 設定對話框

方法	功能
Builder	預備對應的介面元件
show	顯示對話框
setIcon	設定對話框圖像
setTitle	設定對話框標題
setMessage	設定對話框的主要內容
setItems	設定陣列對話框內容及對應的動作

4.3.2 定義座標與設定圖標層

使用 GeoPoint 函式定位地圖座標，其參數分別為「緯度」和「經度」值，需要以整數來表示。「1000000」也可以寫作科學表示法的「1E6」，程式中使用一個符點數乘上數字表是目的，其輸入的座標數值與查詢得到的「KML」值保持一致。

設定圖標層：網路地圖的定位功能透過 ItemizedOverlay 類別可以將標誌疊加在 Google Map 上，由衛星定位點類別的 GeoPoint() 函式，透過指定經緯度來定位。其方法為宣告一個「items」列表，負責儲存圖標列表，其中參數分別是：Point-地點座標、title-標題字串、snippet-說明字串。最後建立 creatItem() 以及 size() 函式，回傳地圖標誌物件數目與大小。

加入圖標層：要在 Google Map 上放置標誌，必需將標誌圖檔放到 res/drawable 目錄底下。這裡以圖檔 androidlogo.png 表示。「setBounds」接受 left、top、right、bottom 四個參數。

4.4 搜尋功能

- 輸入實際地址搜尋- Android 手機內建地圖搜尋的功能，但除了內建的地圖程式外，透過 Geocoder 物件，同樣可利用程式控制的方式，輸入地址字串，反查 GPS 地理座標(GeoPoint)的服務。

要使用 Google 提供的 Geocoder 服務，必須要有 Google Map API Key，在非商用的情況下，Geocoder 反查 Address 物件的服務是較沒有限制的。不過，以地址反查地標的回傳結果，有筆數的限制，如果反查過多筆數時，可能發生無回傳地標的情況。所以，在輸入地址查詢時，要盡可能地輸入完整地址或常用的地標敘述。

專案中以自訂函數 getGeoByAddress() 傳入使用者輸入的地址，再利用 Google 提供的 Geocoder 服務 Geocoder.getFromLocationName() 來取回自 Google 伺服器找到的搜尋結果，以 List<Address> 存放。再以 List.get() 取出 Address 的物件，使用 Address.getLatitude() 及 Address.getLongitude() 進一步取得經緯度，最後以 GeoPoint 的物件作為回傳值使用。得到 GeoPoint 後，就可以利用自訂函 refreshMapViewByGeoPoint() 更新 MapView 的內容。

- 輸入經緯度搜尋-使用者輸入目的地的座標(經度與緯度)，透過 GeoPoint 的定位，就可以在地圖上顯示目的地的正確位置。專案中使用 EditText 來輸入座標的經緯度，接著將座標轉成 GeoPoint 物件後，再利用地圖控制(MapController)的 animateTo() 將地圖中心移至 GeoPoint 的座標上。

4.5 路徑規劃

在正式版本的 Android SDK 中，移除原有版本的 DrivingDirection package，所以無法透過程式來規劃導航路徑，雖然無法自行設計導航路徑，但卻可以呼叫手機內建的地圖程式來傳入規劃導航路徑。

程式執行結果會判斷當下的位置在哪裡，在輸入要前往的地址之後，就會呼叫內建的地圖程式來讓使用者選擇是否使用路徑規劃的功能，但是使用前，需要先開啟手機的 GPS 服務，程式才能取得當下的地理座標。

專案中主要以 Uri.parse() 傳入 Google Map 路徑規劃的網址 (http://maps.google.com/maps?f=d), 以利用內建的規劃路徑功能。但由於 maps.google.com 可接受的經緯度需要以「緯度,經度」的字串格式傳遞, 因此, 使用 GeoPointToString() 來重組 GeoPoint 的經緯度。實作中設 saddr 為起點的經緯度, daddr 為終點的經緯度。

4.6 傳送現在位置之經緯度

- 發送一般手機簡訊 SMS-使用發送簡訊要新增權限

「android.permission.SEND_SMS」, 而抓取手機聯絡人資料需要新增權限「android.permission.READ_CONTACTS」。

發送簡訊的關鍵程式是透過 SmsManager 物件的 sendMessage() 方法來寄出簡訊, 此方法需要傳入五個值, 依序為收件人地址、發送地址、內文、發送服務與送達服務, 其中收件人與內文兩個參數都不可為 null, 所以在專案中使用抓取手機的聯絡人作為收件人, 而內文則是直接傳入現在位置的經緯度。最後, 以 Toast.makeText() 顯示快顯資訊, 以確定成功發送簡訊。

專案中使用的 sendMessage 方法為 SmsManager 類別中, 可使用的三種傳送簡訊方式之一, Table.4 為 SmsManager 的三種發送方法。

Table.4 SMS 發送方式

方法名稱	使用時機
sendDataMessage	發送 data 格式及傳送到特定程式的 port
sendMultipartTextMessage	發送多則文字簡訊
sendTextMessage	發送文字簡訊

5. 實驗結果

我們的研究應用可達到上述之所有服務而流程如 Fig. 8 所示

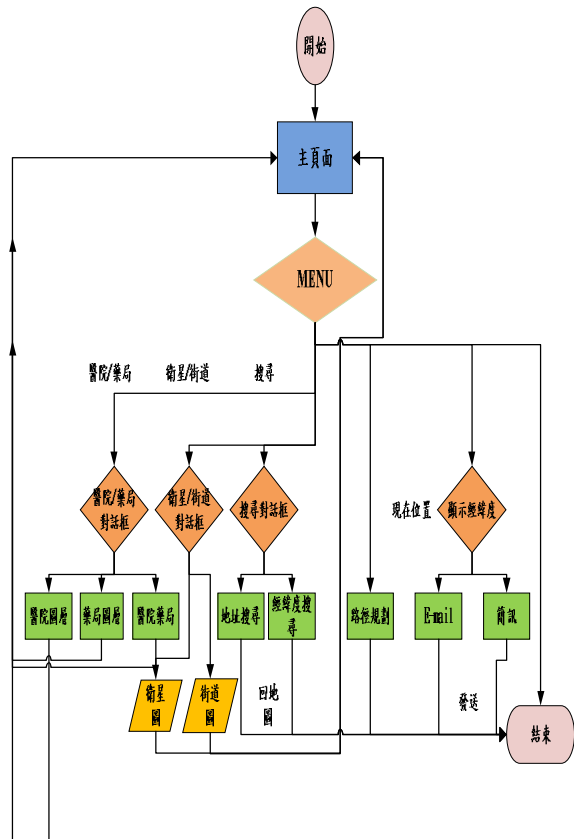


Fig. 8 功能流程圖

包含以下服務:

- GPS 定位。
- 個別顯示醫院及藥局端服務。
- 綜合顯示醫院及藥局。
- 顯示醫院及藥局資訊可直接連結首頁和撥打電話。
- 變更地圖顯示模式。
- 搜尋需求的位置(含正式地址或經緯度)。
- 路徑規劃, 如 Fig. 9 所示。
- 顯示目前位置並發送訊息(含簡訊及電子信箱), 如 Fig. 10 所示。



Fig. 9 路徑規劃與行車估計



Fig. 10 實際電子信箱收信

6. 結論及未來工作

本研究實作之系統《The Medical Facilities for Google Maps and Specific Information on Android》，都已具備本研究的初始目標，使用者可透過手機來取得基本的 Google map 功能，隨時提供顯示所在周邊的各大綜合醫院及衛生署認可藥局的資料並連結其網址，使用者可直接進入網站查詢醫生的排班表與預約掛號，也能隨時隨地的知道現在位置，不怕在就醫的過程中迷路延誤時間，使用者也可使用規劃路徑來找出最短的路程以減少就醫的時間。在旅途中臨時須要藥品也可以利用地圖來顯示藥局並可以知道藥局的基本資訊來做選擇，或是發生意外時，可利用簡訊與 e-mail 從聯絡簿中選取聯絡人發出目前的位置求救。在操作介面上，配合實機使用觸控式螢幕並選用較大、清楚的圖案，讓較年長的使用者可以更容易了解每個選項的功能不須要在鍵盤上找尋，直接點選圖案即可。

研究起初，從對於 Android 平台和撰寫 java 程式完全陌生，直到建構出 Google map 的功能並加入一些我們的想法，Android 提供特有的 java 函式庫，讓撰寫的人可以發揮創意，也讓我們能方便的使用其功能。

然而，我們認為本研究仍然有許多的地方

可以加以精進。目前只能顯示台中市的醫院與藥局，如果要適用於全台，必須建立資料庫儲存醫院和藥局的詳細資料，提供使用者可以隨時查找，並且可以自行增加藥局或醫院，或是設為書籤以方便搜尋。使用者部分，可以加入定時回傳並紀錄現在位置的功能，老年人或是患者可在無人陪伴的狀況下也能獨自外出，即使失聯的狀態下也能靠回傳的位置找到使用者，在操作方面除了利用觸碰式的螢幕，是否可結合語音辨識系統，讓雙手不便的人也能利用聲音來找尋需要的資料，系統提供更好的便利性和更創新的應用，讓使用者能在使用、應用上更方便且快速的找尋到目標，這些都是本系統未來可再研究努力的方向。

7. 參考資料

- [1] Kevin Patrick MD,MS William G. Griswold PhD, Fred Raab and Stephen S. Intille PhD, "Health and the Mobile Phone." *American Journal of Preventive Medicine, Volume 35, Issue 2, August 2008, Pages 177-181.*
- [2] Wen-Yuan Jen, "Mobile healthcare services in school-based health center." *International Journal of Medical Informatics, 78 (2009) 425-434.*
- [3] Renee P. McLeod DNSc, APRN, BC, CPNP and Mary Z. Mays PhD, "Back to the Future: Personal Digital Assistants in Nursing Education" *Nursing Clinics of North America, Volume 43, Issue 4, December 2008, Pages 583-592.*
- [4] James E. Katz, Ronald E. Rice, "Public views of mobile medical devices and services: A US national survey of consumer sentiments towards RFID healthcare technology" *International Journal of Medical Informatics, Volume 78, Issue 2, February 2009, Pages 104-114.*
- [5] Android Developers, <<http://developer.android.com/intl/zh-CN/index.html>>
- [6] Developer Resources for Java Technology, <<http://java.sun.com/>>
- [7] Eclipse.org home, <<http://www.eclipse.org/>>
- [8] Android Maps API - Google Code, <<http://code.google.com/intl/zh-TW/android/maps-api-signup.html>>
- [9] Google Android Emulator Skins, <<http://www.android.encke.net/>>

- [10] Open Handset Alliance,
<<http://www.openhandsetalliance.com/index.html>>
- [11] Android Market,
<<http://www.android.com/market/>>
- [12] HTC Magic,
<<http://www.htc.com/www/product/magic/overview.html>>
- [13] MyGeoPosition.com,
<<http://www.mygeoposition.com/>>
- [14] 行政院衛生署藥物資訊網,
<<http://drug.doh.gov.tw/drugcompany.php>>
- [15] Gasolin, “Google ! Android 手機應用程式設計入門”, 文魁, 2009.
- [16] 楊文誌, “Google Android 程式設計與應用”, 旗標, 2009.
- [17] Gasolin, “Google ! Android 手機應用程式設計入門”, 第二版, 文魁, 2009.
- [18] 佘志龍, 陳昱勛, 鄭名傑, 陳小鳳, 郭秩均, “Google Android SDK 開發範例大全”, 悅知文化, 2009.
- [19] DiveIntoAndroid – androidbmi,
<<http://code.google.com/p/androidbmi/wiki/DiveIntoAndroid>>
- [20] Android 中文資源站,
<<http://android.cool3c.com/>>
- [21] 瘋人院院長的筆記本,
<<http://wiki.cheyingwu.tw/Android>>
- [22] ysl 的程式天堂,
<<http://ysl-paradise.blogspot.com/>>