

中華大學資訊工程學系

100學年度專題製作期末報告

好友萬萬歲

指導老師：

俞征武 教授

組員：

B09729002 林芸如

B09729013 孫梓鈞

B09729004 李冠達

B09629030 胡濠倉

專題編號: PRJ2011-CSIE-10006

中華民國 101 年 1 月 17 日

目錄

第一章、簡介-----	4
第二章、研究背景及目的-----	5
2.1、研究背景-----	5
2.1.1、Android手機-----	6
圖2.1、Android LOGO-----	6
圖2.2、Android 模擬器-----	7
2.1.2、Location Based Services定位技術-----	8
圖2.3:衛星定位-----	8
圖2.4:GPS衛星分布圖-----	9
2.2、研究目的-----	10
第三章、研究方法及步驟-----	11
3.1、系統架構-----	11
圖3.1、系統架構-----	12
3.2、存取資料架構圖-----	13
圖3.2、存取架構-----	13
3.3、資料庫規劃與架構-----	14
圖3.3、ER-Mode-----	14
第四章、遭遇問題與解決方法-----	16

4.1、資料庫建置與連線-----	16
4.2、Android Google Map-----	16
第五章、研究成果-----	21
圖5.1、系統進入畫面-----	21
圖5.2、註冊介面-----	22
圖5.3、主選單-----	23
圖5.4.1、權限系統-----	23
圖5.4.2、單一權限開放-----	24
圖5.5、好友查詢介面-----	24
圖5.6、告知附近好友存在-----	25
圖5.7、訊息介面-----	26
圖5.8、顯示好友所在-----	27
圖5.9、新增好友-----	28
圖5.10、我的位置-----	28
第六章、未來研究方向-----	29
附錄、程式碼-----	30
1. 資料庫部分-----	30
2. Android-----	33
參考文獻-----	36

第一章、簡介

台灣的手機擁有率是世界第一，平均 2270 萬人中將近 1870 萬擁有手機，而且門號還超越台灣的總人口，由此可見手機在台灣占有多少重要性。智慧型手機的崛起，也造成了手機業界的革命，現在手機不僅僅能夠通話就好，還必須帶給人們相當的便利性。而近年來最具震撼的發表其中之一，那莫過於是 Google 所推出的 Android 了，多樣化的功能，以及前衛的外觀，吸引了不少消費者購買，並且越來越多的使用者使用，軟體數量也迅速增加，未來潛力無窮！在 Google 的 Android 的 Market 裡，擁有相當多的軟體可供使用者付費下載或免費下載使用，也造就了 Android 不可動搖的地位，其豐富的多樣性以及便利性也讓 Android 擁有相當多的使用者，可見 Android 有相當大的實力以及發展空間。

但目前的手機除了通話方便之外，在其他方面，其實也不輸市面上有著一定處理核心等級的電腦，不但能提供多方通話，還能上網、聽歌、照相...等等，幾乎說是應有盡有。

那如果手機可以來查詢我的好友在我的附近?讓我若要找他或是給她驚喜，可以藉由手機來得知對方的位子。所以如何以現代智慧型手機的便利性與多功能性達到這樣的效果是我們這次的研究目標。

第二章、研究背景及目的

2.1 研究背景

手機現在是大家的必需品，而現在的人只會拿來玩遊戲，接電話，上臉書，股票..等!但有想過手機也可以幫我們做事情嗎?而手機的功能也越來越多樣化，以及便利性也越來越強大了，而尤其在近期大家都會使用 GOOGLE MAP 來搜尋住址或是當作導航，而手機不僅有這些，除了 GOOGLEMAP 地圖外，另外還有陀螺儀，加速度偵測，可以使手機上的軟體有著更加多樣化的功能。而且兩方都有提供開發平台供使用者撰寫程式，這代表著你在他們網站上所下載使用的軟體，不單單是由兩方的工程師所撰寫，而是有更多人來開發軟體使得我們對於軟體的選擇可以更加的多樣化。

2.1.1 Android 手機

Android 為一個行動裝置系統，最初由Google 開發，最後由公司集團OpenHandset Alliance(包含Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel and T-Mobile 等)共同開發，使用Linux kernel 2.6 版。有Google Map 地圖、2D 與3D 的圖形處理功能(採用Opengl ES1)、資料庫(SQLite)、網路瀏覽器、簡訊(SMS、MMS)、相機、多媒體播放(包括看電視)、及GTalk(類式及時通)等功能。



圖 2.1:Android Logo



圖 2.2: Android 模擬器

2.1.2 Location Based Services 定位技術

Location Based Service 定位的技術通常使用高精確性的全球衛星定位系統(GPS)做定位，而GPS 接收不到的地方(如室內)可採用 Cell-ID 與LAC 作定位。

◆ GPS 定位技術：

全球衛星定位系統(GPS)是結合衛星和無線技術，能提供使用者精確的位置。太空中有24 顆衛星(如圖四)，地面上有一個主控站、三個資料注入站和五個監測站做為接收器，至少3 顆衛星，就可確認該地區的經緯度及海拔高度等資訊。越多衛星做偵測，接收到的位置越精準，偵測方式如圖三，以P1、P2、P3三顆衛星偵測範圍的焦點為目前所在的位置。

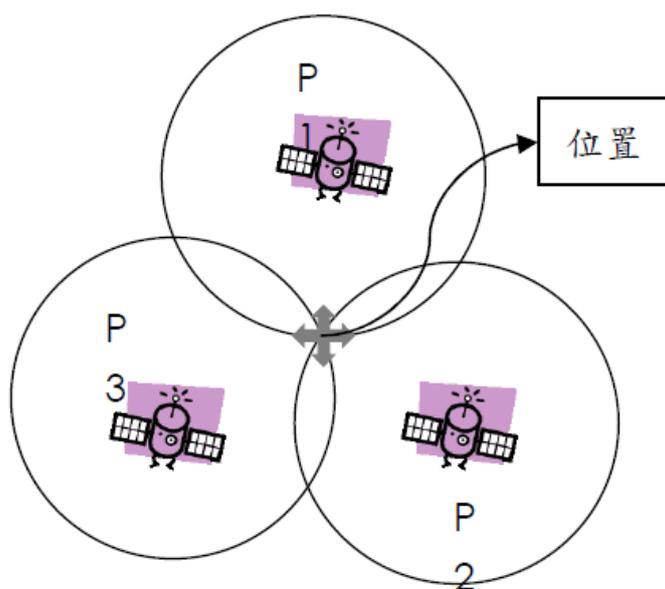


圖 2.3:衛星定位

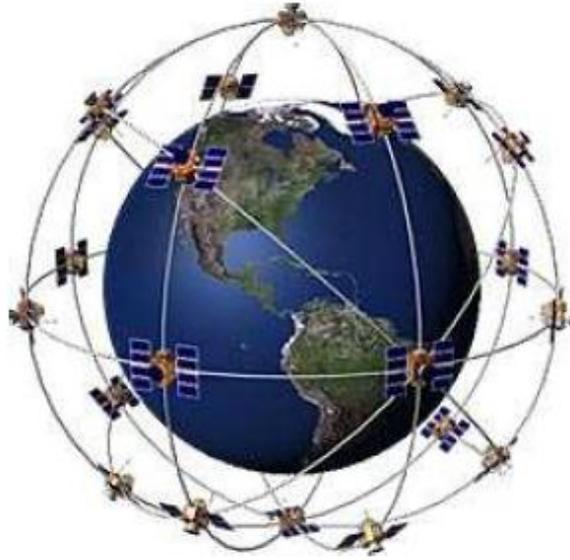


圖 2.4:GPS 衛星分布圖

2.2 研究目的

本專題的主要目標是透過完成一個可以查詢好友在哪的系統來學習。

- (1) 培養開發技術的能力：Android 良好的實驗環境，能夠引起研究者的興趣，進而深入技術層面探討，透過討論和研究，產生靈感和創意。
- (2) 結合簡訊及無線網路等多種技術，使user 和user 之間的溝通更有效率，並可節省使用者成本。
- (3) 結合GPS、MYSQL、JSP及Tomcat技術，查詢好友現在的位址，並且利用GPS顯示出讓對方得知，以方便搜尋及追蹤。

第三章、研究方法及步驟

3.1 系統架構

我們在手機上的系統架構如圖3.1。開始登入我們的系統並確認驗證身分後，即可進入操作畫面，功能將分為：

1. 新增好友：從電話簿中已存在的聯絡人，選擇並加入到本系統的好友資料庫中。
2. GPS定位系統：透過定位系統可以知道目前所在位置，也可選擇輸入地址或關鍵字查詢地點。
3. 開放權限：可以設定好友是否開放權限，讓對方查詢自己位置。
4. 好友查詢：若好友有開放權限，便透過GPS查詢好友的相關位置，當好友出現在附近時，跳出傳送訊息的介面，讓對方得知且一起吃飯或者碰面等活動。

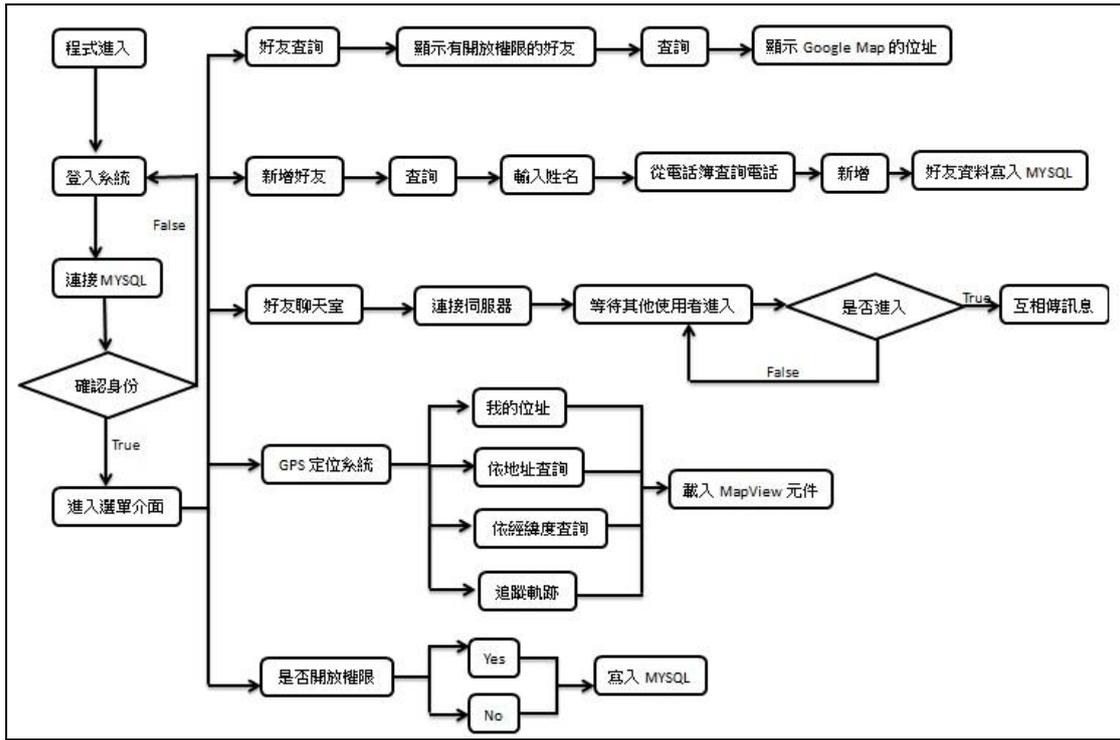


圖3.1 系統架構

3.2 存取資料架構圖

我們利用網頁伺服器來達到存取的效果，見圖3.2。一開始我們認為能直接跳過伺服器作存取的動作，雖然可行，但會發生安全性上的問題，所以改由伺服器當成中間人。

首先由手機上登入於網頁伺服器，再配合各式需求選擇功能後，將會傳送網頁上已寫好的各式SQL語法給資料庫伺服器，而當資料庫執行完成後網頁上會顯示出結果，並回傳給手機使用。

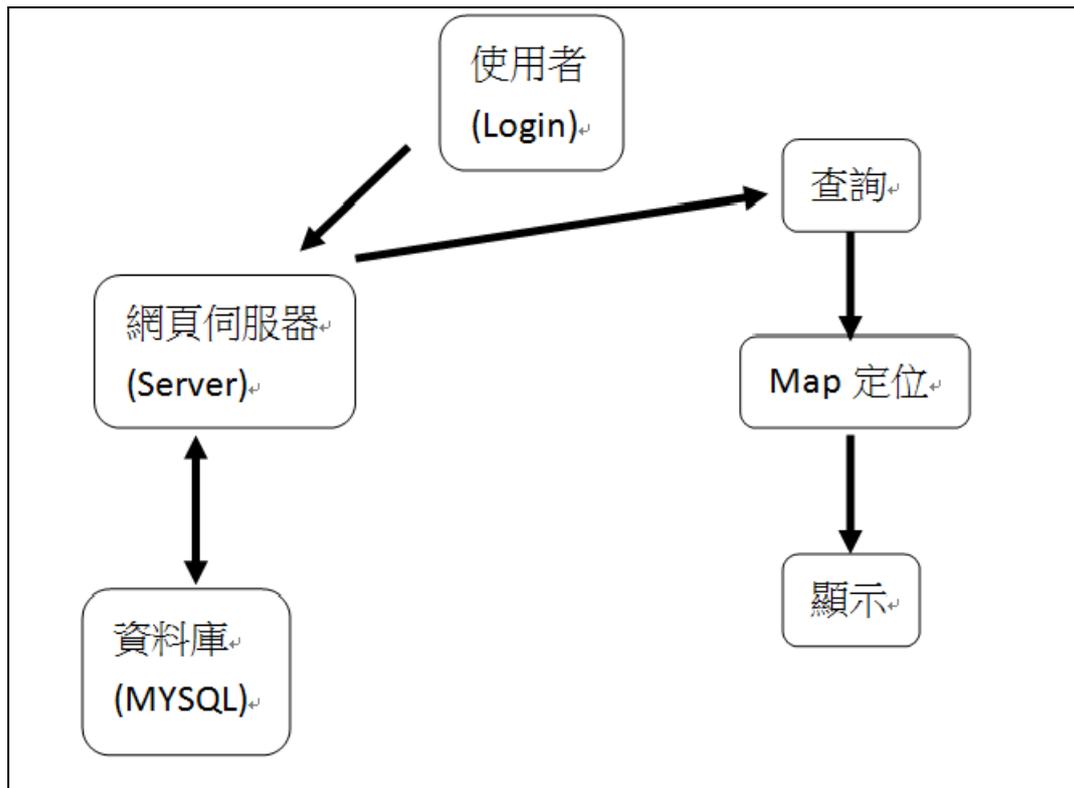


圖3.2、存取架構

3.3 資料庫規劃及架構

資料庫的設計概念是以“使用者”為主體，接著聯想到”使用者”身邊，基本上我們只設計兩張資料表:帳號及使用者檔案表。主要資料都是放在檔案表裡面，有帳號、朋友姓名、經緯度、朋友手機。利用帳號作為標記、好讓使用者能快速找到自己帳號的資料，且也不會看到他人好友資料或被看到自己好友資料。根據以上的說明，以下是我們專題的ER-Model。

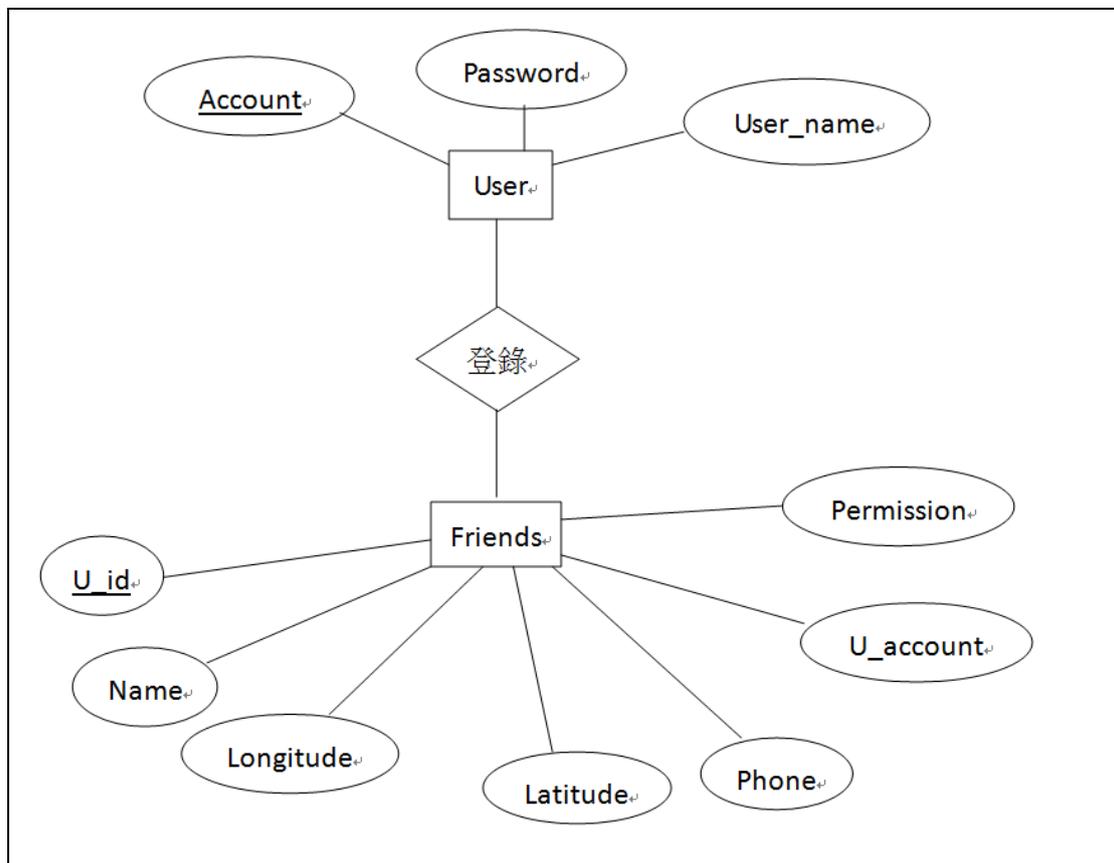


圖3.3、 ER-Model

透過資料的關聯性完成資料的存取，以下將介紹資料庫主要儲存的內容及資料存取的流程。

(1) User table

儲存每位使用者的帳號，包含Account、Password、User_name。

表一、User table

Account	Password	User_name
0958263XXX	198702XX	Bass
0987761XXX	198912XX	Olddad
0958787XXX	198710XX	Eric

(2) Friends table

此資料表主要儲存是使用者好朋友的資料，包含U_id、Name、Longitude、Latitude、Phone、U_account、Permission。

表二、Friends table

U_id	Name	Longitude	Latitude	Phone	U_account	Permission
1	olddad	121.517081	25.047924	0987761XXX	0958263XXX	1
2	bass	120.952838	24.759974	0958263XXX	0987761XXX	0
3	eric	120.964035	24.812736	0958787XXX	0987761XXX	1
4	bass	120.952838	24.759974	0958263XXX	0958787XXX	1

第四章、遭遇問題與解決方法

因為Android是屬於新的東西，所以剛開始需要花很多時間去學習新的語法，新的架構以及新的環境。而我們所設計的系統又需要用到資料庫、網頁伺服器，所以我們也在這部分上遇到許多困難。

(1) 資料庫部分

首先是網頁的架設，在一開始我們並不懂如何利用網頁去連結資料庫，然而在查詢怎麼實作時，又發現了多種網頁作法，PHP、ASP、JSP等等，最終則是選擇了利用JSP進行實作。

在此部分發現到如果要讓JSP網頁對資料庫連線必須要下一款名為JDBC的驅動程式。

接著開始架設網頁(選用Tomcat)，也遇到了無法外部連結問題，原來只要更改防火牆即可。

(2) Android Google Map

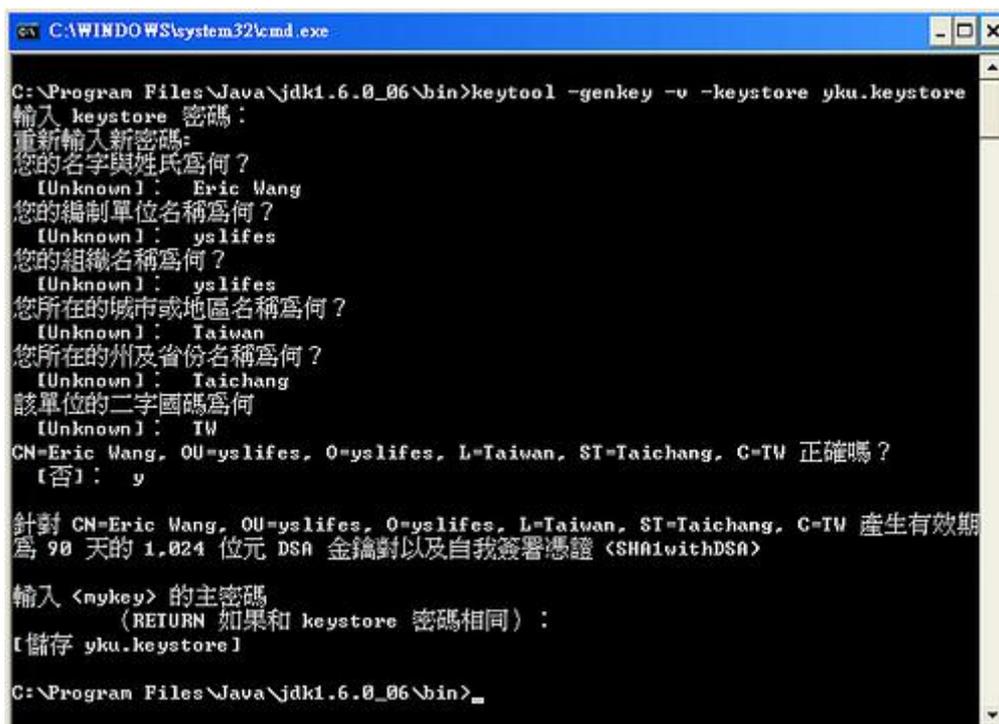
一開始想使用Google map在模擬器的時候，卻只有顯示出網格的部分，而無法出現一般地圖，之後查了書籍後得知解決方法，須申請API KEY才能使用。

而解決方法如下步驟:

- Step 1: 要向Google申請服務的指紋碼，需要先在本機端產生MD5的指紋證書，再利用這個證書產生指文碼。**如何產生指紋證書**呢？只需要在jdk的安裝路徑bin目錄下，輸入

```
keytool -genkey -v -keystore 名稱.keystore
```

其中”名稱.keystore”請自行設定喜好的名稱，再來畫面會要求你輸入一些基本資訊，大概內容如下圖。



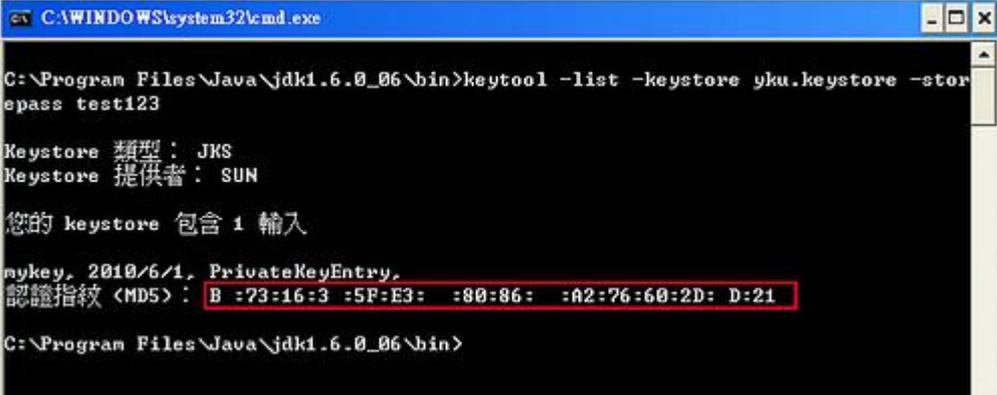
```
C:\WINDOWS\system32\cmd.exe
C:\Program Files\Java\jdk1.6.0_06\bin>keytool -genkey -v -keystore yku.keystore
輸入 keystore 密碼:
重新輸入新密碼:
您的名字與姓氏為何?
  [Unknown]: Eric Wang
您的編制單位名稱為何?
  [Unknown]: yslives
您的組織名稱為何?
  [Unknown]: yslives
您所在的城市或地區名稱為何?
  [Unknown]: Taiwan
您所在的州及省份名稱為何?
  [Unknown]: Taichang
該單位的二字國碼為何?
  [Unknown]: TW
CN=Eric Wang, OU=yslives, O=yslives, L=Taiwan, ST=Taichang, C=TW 正確嗎?
  [否]: y
針對 CN=Eric Wang, OU=yslives, O=yslives, L=Taiwan, ST=Taichang, C=TW 產生有效期
為 90 天的 1,024 位元 DSA 金鑰對以及自我簽署憑證 (SHA1withDSA)
輸入 <mykey> 的主密碼
  (RETURN 如果和 keystore 密碼相同):
[儲存 yku.keystore]
C:\Program Files\Java\jdk1.6.0_06\bin>
```

圖4.1、輸入keystore基本資訊

- Step2: 再來是**利用指紋證書來取得指紋碼**，只需要輸入：

```
keytool -list -keystore [路徑/檔案名稱] -storepass 密碼
```

其中” [路徑/檔案名稱]” 可以是上一步產生的 keystore 或是直接取用 android sdk 產生的，密碼部份此範例使用 test123，畫面如下，會產生一組認證指紋碼，把此指紋碼 copy 下來。



```
C:\WINDOWS\system32\cmd.exe
C:\Program Files\Java\jdk1.6.0_06\bin>keytool -list -keystore yku.keystore -storepass test123
Keystore 類型: JKS
Keystore 提供者: SUN
您的 keystore 包含 1 輸入
mykey, 2010/6/1, PrivateKeyEntry,
認證指紋 (MD5): B :73:16:3 :5F:E3: :80:86: :A2:76:60:2D: D:21
C:\Program Files\Java\jdk1.6.0_06\bin>
```

圖4.2、產生MD5認證碼

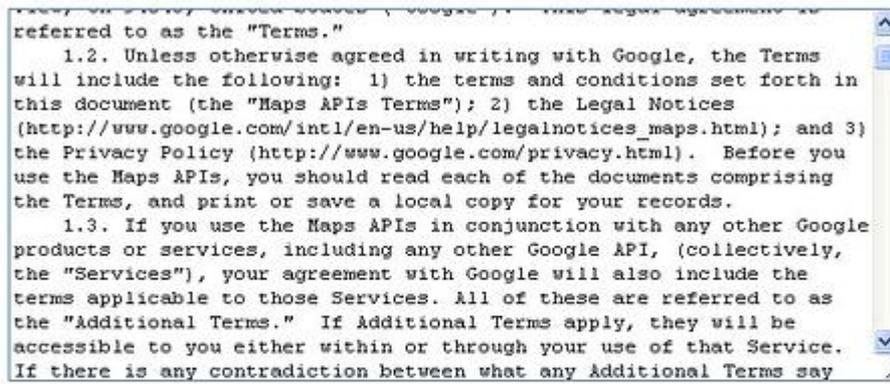
- **Step3: 向 Google 申請 Maps API key**，網址：

<http://code.google.com/android/maps-api-signup.html>

需要同意條款及給於自行產生的指紋碼，把上一步所得結果貼到 input 裡，按 Generate API key。

If you use different keys for signing development builds and release builds, you will need to obtain a separate Ma signed by the corresponding certificate.

You also need a [Google Account](#) to get a Maps API key, and your API key will be connected to your Google Ac



referred to as the "Terms."
1.2. Unless otherwise agreed in writing with Google, the Terms will include the following: 1) the terms and conditions set forth in this document (the "Maps APIs Terms"); 2) the Legal Notices (http://www.google.com/intl/en-us/help/legalnotices_maps.html); and 3) the Privacy Policy (<http://www.google.com/privacy.html>). Before you use the Maps APIs, you should read each of the documents comprising the Terms, and print or save a local copy for your records.
1.3. If you use the Maps APIs in conjunction with any other Google products or services, including any other Google API, (collectively, the "Services"), your agreement with Google will also include the terms applicable to those Services. All of these are referred to as the "Additional Terms." If Additional Terms apply, they will be accessible to you either within or through your use of that Service. If there is any contradiction between what any Additional Terms say



I have read and agree with the terms and conditions ([printable version](#))

My certificate's MD5 fingerprint:

圖4.3、申請API KEY

然後會 response 一個金鑰及 xml 配置範例，直接 copy 範例裡的內容使用就可以了。

Google 地圖 API

[Google 程式碼首頁](#) > [Google 地圖 API](#) > Google 地圖 API 申請

感謝您申請 Android Maps API 金鑰！

您的金鑰為：

OXBuz_iCXxd6EQiHEBdjdeH [REDACTED]

此金鑰適合所有使用以下指紋憑證所簽署的應用程式：

B6:73: [REDACTED]

此處提供您 xml 配置的範例，您可依照自己的方法使用地圖功能：

```
<com.google.android.maps.MapView
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:apiKey="OXBuz_iCXxd6EQiHEBdjdeH [REDACTED]"
/>
```

請參閱 [API 文件](#) 以取得詳細資訊。

圖4.4、產生API KEY和XML範例

以上申請流程就已經完成了。然後再把提供的 XML 放置
Layout 裡，就可顯示出 Google Map。

第五章、研究成果

以下則是我們成果展現: 圖5.1為我們系統進入時的畫面，一開始定位自己的位址並且顯示在GOOLEMAP上，”記住我”方塊則是讓使用者在每次都登入時不必重新再輸入一次帳號密碼。而第一次使用的使用者可按註冊鈕，就進入選單。

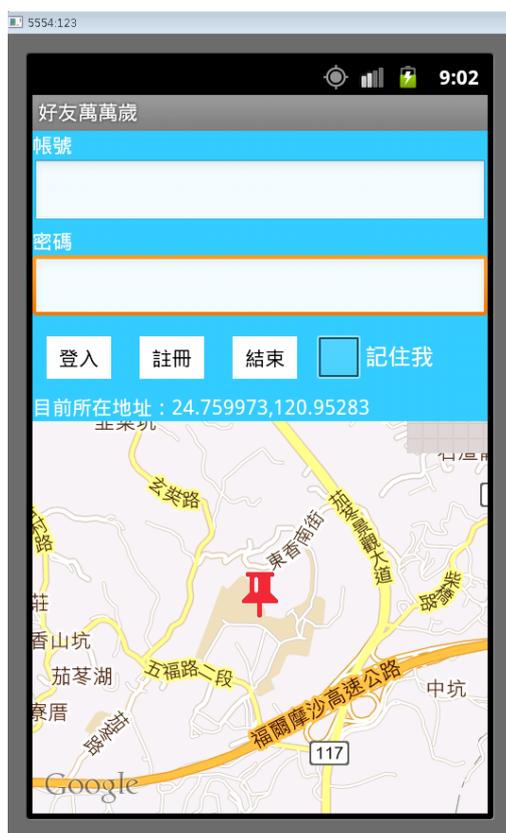


圖5.1、登入畫面

圖5.2是註冊系統，若尚未使用過的使用者須先註冊。

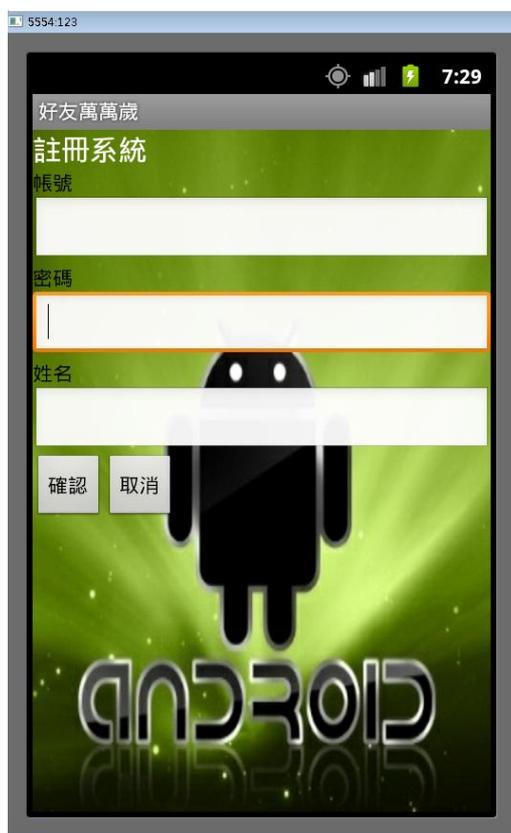


圖5.2 註冊系統

當成功登入後，就會進入我們主選單部分(圖5.3)有好友查詢、新增好友等功能，以及是否開放權限(圖5.4.1)給對方得知自己的地理位置，有單一開放(圖5.4.2)或者全部開放的選擇，保障自己隱私。



圖5.3、選單畫面

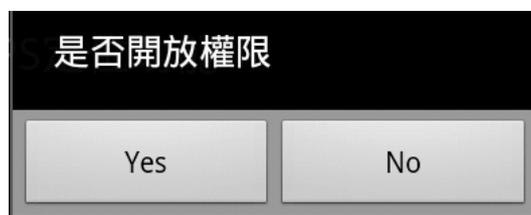


圖5.4.1、權限系統



圖5.6:告知附近好友存在

如果我們查詢的對象距離我們半徑一公里內的話，則會傳送訊息(圖5.7)給對方，而在顯示地圖的畫面也會標示出兩者相距多少公尺(圖5.8)。藍色原點代表自己，圖釘代表對方。



圖5.7、訊息介面



圖5.8、顯示好友所在

功能之一(圖5.9)可從聯絡人裡選擇好友的姓名，可加入到這個應用程式方便使用，如果對方尚未使用這個APP的話則顯示”他尚未使用APP”並且傳送簡訊告知好友是否也想使用這個APP。

功能之二(圖5.10)隨時隨地顯示自己最新的位置，且可隨時更新自己的位置。



圖5.9、新增好友



圖5.10、我的位置

第六章、未來研究方向

因我們以GPS定位系統為出發點，所以我們未來研究的方向可能有下列幾項：

(1) 預防老人、小孩走失

透過本系統的權限開放設定之後，可以偵測尋找家人的相關位置，並預防走失，也可以觀察記錄小孩下課後有沒有直接回家，還是有沒有說謊到不良場所逗留。

(2) 旅行社應用

現今旅遊風靡，旅行社或是參加的遊客將可以透過本系統，去追蹤旅客或是導遊的所在地點，便不會發生迷路、找不到集合地點或是臨時找不到導遊的情形。

(3) 跟車系統

和現在的計程車公司合作，讓搭乘計程車的乘客可以更加安心搭乘，乘客可以追蹤現在計程車的路程，並可得知是否安全到達目的地，對於計程車司機來說，也可以證明自己是否有安全將乘客送達。

(4) 商家合作

透過定位系統，也可以讓商家偵測附近的行人，並且傳遞訊息給予周遭鄰近的人，使之對商家產生感興趣並前往消費。

附錄、程式碼

1. 資料庫部分

1.1 JSP 網路連線

- 宣告部分

```
String code = cin.getEditableText().toString();
//擷取自螢幕輸入的字元↵

String uriAPI = "http://192.168.222.71:8070/my.jsp";
//IP 位址↵

HttpPost httpRequest = new HttpPost(uriAPI);
//使用 POST 請求↵

List <NameValuePair> p = new ArrayList <NameValuePair>();
//輸入資料的所存放的陣列群 設為 P↵

p.add(new BasicNameValuePair("number", code));
//將 CODE 放入為 number 的陣列↵
```

附圖 1.1

- 放在按鈕點擊後

```
httpRequest.setEntity(new URLEncodedFormEntity(p, HTTP.UTF_8));
//送出剛剛所輸入的字元↵

httpResponse = new DefaultHttpClient().execute(httpRequest);
//取的連線↵

if(httpResponse.getStatusLine().getStatusCode()==200)
//連線 200 為成功↵

strResult = EntityUtils.toString(httpResponse.getEntity());
//取回網頁上所查詢後的字元
```

附圖 1.2

1.2 JSP 網頁部分

- 連線變數宣告部分

```
Connection con = null;
Statement stmt = null;
ResultSet rs = null;
String code = request.getParameter("username");
String code2 = request.getParameter("pass");
```

附圖 1.3

其中 code 及 code2 為接收手機上所傳送的字串語法。code 擷取陣列名稱"username"裡面的值，而 code2 則是擷取" pass" 裡面的值，要注意的是，這裡的陣列名稱必需跟手機上的陣列名稱一樣。

- 連線與 sql 語法

```
String url = "jdbc:mysql://140.126.11.15:3306/android";
String user = "*****";
String password = "*****";
String driver = "com.mysql.jdbc.Driver";
Class.forName(driver);
con = DriverManager.getConnection(url, user, password);
stmt = con.createStatement();
String sql = "select * from android.login where account=' " + code + "' and
            pass=' " + code2 + "' ";
rs = stmt.executeQuery(sql);
```

附圖 1.4

第四、五行程式碼宣告 JDBC 驅動程式並使用，第七行則是對該 url 連線並輸入帳號密碼以登入資料庫，第八行創立一個物件好讓第九行的 sql 語法能執行，其中 sql 與法則是看功能需求而改變，在輸

入的變數內，例如 `account`：必須注意到是否有加上兩個單引號，若無，則程式碼將會以數字執行，若有則以字串執行。而在更新語法上，有著跟查詢語法不同的是最後一行：`stmt.executeUpdate`。 `stmt.executeQuery` 需改成 `stmt.executeUpdate` 才能成功執行更新動作。

2. Android 部分

2.1 取得經緯度

```
private GeoPoint getGeoByLocation(Location location){
    GeoPoint gp = null;
    try{
        /* 當Location存在 */
        if (location != null){
            double geoLatitude = location.getLatitude()*1E6;
            double geoLongitude = location.getLongitude()*1E6;
            gp = new GeoPoint((int) geoLatitude, (int) geoLongitude);
        }
    }catch(Exception e){
        e.printStackTrace();
    }
    return gp;
}
```

附圖 1.5

先判斷傳入的 Location 物件是否為 null，若存在讀取 Location 物件資訊，且呼叫 Location 物件的 getLatitude 和 getLongitude 方法取得目前位置的經度和緯度。

2.2 計算距離

```
public double GetDistance(double lat1, double lng1, double lat2, double lng2){  
    double Lat1r = (Math.PI/180)*(lat1);  
    double Lat2r = (Math.PI/180)*(lat2);  
    double Lon1r = (Math.PI/180)*(lng1);  
    double Lon2r = (Math.PI/180)*(lng2);  
    // 地球半徑  
    double R = 6371; //  
    double d = Math.acos(Math.sin(Lat1r)*Math.sin(Lat2r) +  
        Math.cos(Lat1r)*Math.cos(Lat2r)*Math.cos(Lon2r-Lon1r))*R;  
    return d*1000;  
}
```

附圖 1.6

$(\text{Math.PI}/180) * (\text{經度}/\text{緯度})$ 是將角度換算成弧度。而距離是用算圓球體的公式: Great-circle distance 公式，其中 d 為距離， R 是地球半徑，最後算出來的單位是英里，須換成公尺所以乘上 1000。

使用 If 判斷式，如果算出來的距離小於 1 公里的話，就跳到傳送訊息的介面，否則顯示當下的畫面。

2.3 傳送訊息

```
SmsManager smsManager = SmsManager.getDefault();
try{
    //建立自訂Action常數的Intent(給PendingIntent參數之用)
    Intent itSend = new Intent(SMS_SEND_ACTION);
    Intent itDeliver = new Intent(SMS_DELIVERED_ACTION);

    //sentIntent參數為傳送後接受的廣播訊息PendingIntent
    PendingIntent mSendPI = PendingIntent.getBroadcast(getApplicationContext(), 0, itSend, 0);

    // deliveryIntent參數為送達後接受的廣播訊息PendingIntent
    PendingIntent mDeliverPI = PendingIntent.getBroadcast(getApplicationContext(), 0, itDeliver, 0);

    //發送SMS簡訊，注意倒數的兩個PendingIntent參數
    smsManager.sendTextMessage(strDestAddress, null, strMessage, mSendPI, mDeliverPI);
    mTextView01.setText(R.string.str_sms_sending);
}catch(Exception e){
    mTextView01.setText(e.toString());
    e.printStackTrace();
}
}
```

附圖 1.7

`smsManager.sendTextMessage(strDestAddress, null, strMessage, mSendPI, mDeliverPI);` 其中，`strDestAddress`: 對方的手機號碼
`scAddress`: 自己的手機號碼，`strMessage`: 訊息內容，`mSendPI`: 傳送是否成功的回應，`mDeliverPI`: 接收是否成功的回應。

參考文獻

[1]<http://blog.yslives.com/archives/671> ---Google Map 申請、設定及使用

[2]<http://zh.wikipedia.org/wiki/Android> ---ANDROID

[3]<http://zh.wikipedia.org/wiki/GPS> ---全球定位系統

[4]林城，”Google Android 2.X 應用程式開發實戰第二版”，基峯出版社，2011/5

[5]吳亞峰、索依娜，”學會 Android 應用開發的 18 堂關鍵基礎課程”，碩博文化出版社，2011/7

[6]余志龍、陳昱勛、鄭名傑、陳小鳳，”Google Android SDK 開發範例大全 第三版”，悅知文化出版社，2011/4

[7]郭宏志，”深入淺出 Android 專題實作之無線點餐系統、多媒體、遊戲、GPS 與網路應用程式開發”，碩博文化出版社，2011/3