

行動版的 Google Map

郭保宏



01

各平台使用比較

行動版 GoogleMap 圖台

分類	圖台套件	繼承類別	說明	圖台金鑰
iOS	MapKit.framework	MKMapViewDelegate		不需要金鑰
Android	maps.jar (選擇各 sdk 版本下, 引用 Google APIs)	v1: MapActivity	2012.12.3 起聲明不支援 Map API v1, 到 2013.3.3 前仍接受申請 Map API v1 key	1. 開發期金鑰 2. 上架金鑰 https://code.google.com/apis/console
		v2: Activity (MapFragment)	需申請 Map API V2 key	1. 開發期金鑰 2. 上架金鑰
		v3: Activity	使用 WebView 來操作 Map 網頁	js 金鑰可有可無
Web 或 Phone Gap	tinyMap	jquery.tinyMap-3.2.0.min.js	tinyMap 的 API	
	googlemap	http://maps.google.com/maps/api/js?sensor=true http://maps.googleapis.com/maps/api/jsv=3&key=API_KEY&sensor=false		

基本上, API 不能共用, 但各自的 API, 有各自好用的地方, 以及需要自行調整的地方

02

IOS

圖台使用 API 分享

iOS 圖台基本使用介紹

圖台初始化

	使用步驟	程式片段
1	引用 mapKit.framework	在 Build Phases 功能設定
2	建立程式檔案 MyMapController.h	<pre>#import <UIKit/UIKit.h> #import <MapKit/MapKit.h> @interface MyMapController : UIViewController <MKMapViewDelegate>{ MKMapView *map; } @end</pre>
3	初始化圖台 MyMapController.m 在 viewDidLoad 裡初始化	<pre>// 建立 MapView map = [[MKMapView alloc] initWithFrame:CGRectMake(0.0f, 40.0f, 320.0f, 400.0f)]; // 顯示使用者目前位置 (藍色圓點) map.showsUserLocation = YES; //MapView 的環境設置 map.mapType = MKMapTypeStandard; map.scrollEnabled = YES; map.zoomEnabled = YES; // 將 MapView 顯示於畫面 [self.view addSubview:map];</pre>

圖台使用者定位開發技巧

	使用步驟	程式片段
1	引用 CoreLocation.framework	在 Build Phases 功能設定
2	繼承 CLLocationManagerDelegate	<pre>#import <CoreLocation/CoreLocation.h> #import <MapKit/MapKit.h> @interface MyMapController : UIViewController <MKMapViewDelegate, CLLocationManagerDelegate >{ } @property CLLocationManager *iCLLocationManager ; @end</pre>
3	因應 iOS8, 取得使用者變動的經緯度座標 或參考網址: http://dev.00don.com/?p=103	<pre>// Create location manager with filters set for battery efficiency. self.iCLLocationManager = [[CLLocationManager alloc] init]; // check if the app can respond to the new selector found in iOS8 if([self.iCLLocationManager respondToSelector:@selector(requestAlwaysAuthorization)]) { [self.iCLLocationManager requestAlwaysAuthorization]; // 永久授權 [self.iCLLocationManager requestWhenInUseAuthorization]; // 使用中授權 } self.iCLLocationManager.distanceFilter = kCLLocationAccuracyHundredMeters; self.iCLLocationManager.desiredAccuracy = kCLLocationAccuracyBest; // Start updating location changes. [self.iCLLocationManager startUpdatingLocation];</pre>
4	因應 iOS8, 專案 Resource 裡, plist 加入 2 個參數設定	NSLocationWhenInUseUsageDescription -> 只有打開 APP 的時候 NSLocationAlwaysUsageDescription -> 任何時間
5	取得新座標的函式, 可取得: coordinate.latitude, coordinate.longitude, horizontalAccuracy, verticalAccuracy, timestamp, Speed, course	<pre>-(void)locationManager:(CLLocationManager *)manager didUpdateLocations:(NSArray *)locations { self.location = locations.lastObject; //self.location.coordinate.latitude]; }</pre>

圖台自製縮放 index 函式

	說明	程式片段
1	自製在圖台移動的函式： 傳入座標及縮放 index(3~21) (由於 iOS 圖台平移位置，只能傳縮放比例，在用習慣要 zoom 到第幾層，index(3~21) 比比例實用許多)	<pre>- (void) moveMapCenterTo : (CLLocationCoordinate2D) aCoordinate withZoomLevel : (int) aLevel { float regionDelta = 0; if(aLevel == 0){ regionDelta = [self getRegionDelta : [self getZoomLevel : self.mapView]]; } else regionDelta = [self getRegionDelta : aLevel]; MKCoordinateSpan span = MKCoordinateSpanMake(regionDelta,regionDelta); MKCoordinateRegion userPos = MKCoordinateRegionMake(aCoordinate, span); [map setRegion : userPos animated:YES]; }</pre>
2	轉換要 zoom 的 index 數	<pre>- (int) getZoomLevel : (MKMapView *) aMapView { return 21 - round(log2(aMapView.region.span.longitudeDelta * MERCATOR_RADIUS * M_PI / (180.0 * aMapView.bounds.size.width))); }</pre>
3	將 index 轉換為縮放比例	<pre>- (float) getRegionDelta : (int) aLevel { float regionDelta = 0; switch (aLevel) { case 0:regionDelta = 0;break; case 1:regionDelta = 225.0;break; case 2:regionDelta = 112.5;break; case 3:regionDelta = 56.25;break; case 4:regionDelta = 28.125;break; case 5:regionDelta = 14.0625;break; case 6:regionDelta = 7.031250;break; case 7:regionDelta = 3.515625;break; case 8:regionDelta = 1.757812;break; case 9:regionDelta = 0.878906;break; case 10:regionDelta = 0.439453;break; case 11:regionDelta = 0.219727;break; case 12:regionDelta = 0.109863;break; case 13:regionDelta = 0.054932;break; case 14:regionDelta = 0.027466;break; case 15:regionDelta = 0.013733;break; case 16:regionDelta = 0.006866;break; case 17:regionDelta = 0.003433;break;</pre>

圖台套疊功能

	函式	使用說明
1	addAnnotations	圖台加入點位陣列，產生多點位在圖台上
2	addOverlays	圖台加入點位陣列，套疊在線或面在圖台上
3	viewforAnnotation	圖台顯示各種類別下的點位 marker 技巧：利用自製多組 MKAnnotation 子類別來區分多種類別，不同類別中，可自訂自己的圖示。
4	viewForOverlay	圖台顯示各種類別下的線或面的圖層 技巧：利用自製多組 MKPolyline 子類別， MKPolygon 子類別來區分多種類別，不同類別中，可自訂自己的線或面的顏色，線的粗細。
5	didSelectAnnotationView	點擊點位 marker 時的觸發事件
6	regionDidChangeAnimated	圖台四周範圍移動時，觸發的事件
7	didAddAnnotationViews	圖台套疊多個點位下，可以讓特定種類點位顯示在最上面
8	清除特定類別下的點位	<pre>for (id<MKAnnotation> annotation in self.mapView.annotations) { if([annotation isKindOfClass : [WeatherMarker class]]){ [self.mapView removeAnnotation : annotation]; } }</pre>
9	清除特定類別下的圖層	<pre>for (id<MKOverlay> overlay in self.mapView.overlays) { if(![overlay isKindOfClass:[MKUserLocation class]]){ [self.mapView removeOverlay : overlay]; } }</pre>

	調校項目	使用說明
1	一次套疊太多點位在圖台上，如果只套疊目前手機畫面 4 角範圍下的點位或圖層，可透由取得四邊的最小最大經度、最小最大緯度，來套疊點位	<pre>MKMapRect rect = map.visibleMapRect; // ne: 北東, se: 南東, nw: 北西, sw: 南西 MKMapPoint nePoint = MKMapPointMake(rect.origin.x + rect.size.width, rect.origin.y); MKMapPoint swPoint = MKMapPointMake(rect.origin.x, rect.origin.y + rect.size.height); CLLocationCoordinate2D neCoord = MKCoordinateForMapPoint(nePoint); CLLocationCoordinate2D swCoord = MKCoordinateForMapPoint(swPoint); // 利用上下左右 4 角範圍，只套疊範圍下的點位或圖層 TopX : neCoord.latitude TopY : neCoord.longitude BottomX : swCoord.latitude BottomY : swCoord.longitude</pre>

03

ANDROID
圖台使用分享

Android 圖台基本使用介紹

開發者 apis console: <https://code.google.com/apis/console>

版本	版本差異說明	應用專案
v1	<ol style="list-style-type: none">1. 開發者 console 已不能申請 Map API v1 key, 原來的專案還是可以使用2. v1 和 v2 的 API 都是 java 原生, 但完全不同的 API	大甲媽祖遶境進香
v2	<ol style="list-style-type: none">1. 開發期, 需導入 Google Play Services2. User 下載時, 手機內必需 SDK2.3 以上才有 Google Play Services 服務3. 地圖是封裝在 MapFragment 類別中4. 使用 Vector tiles 技術使地圖顯示得更快, 頻寬使用的更少。5. Caching 比較少, 所以看到的地圖空白區域會更少。	<ol style="list-style-type: none">1. 天眼衛星車隊管理2. 富民運輸3. 司機派遣
v3	<ol style="list-style-type: none">1. 使用 WebView 來操作 Map 網頁, 而且是把網頁放在手機上, 而不是去瀏覽 Server 上的網頁。	

03₁

ANDROID
V1 圖台使用分享

v1 圖台初始化

	使用步驟	程式片段
1	Google APIs Console 網頁開啓 v1 服務，及申請你的開發時期和上架時期的金鑰	利用你開發電腦的 (SHA1 碼；專案 package)，在 google apis console 申請一組 Android key，這一組就是你的開發時期的金鑰。而正式發佈的金鑰，是你用 eclipse 在 Export Signed Application Package 所產生的 keystore 裡，就有一組正式憑證的 SHA1 碼，就可以申請正式的金鑰，只有在上架時，才換這組金鑰，從 Google Play 下載下來時的 APP，才看的到圖台
2	定義 maplayout 裡，放圖台的 tag	<pre><com.google.android.maps.MapView android:id="@+id/mapview" android:layout_width="fill_parent" android:layout_height="fill_parent" android:apiKey="06CJMDB32xxxQKZxxx329Hhminxxxli5Rxxxxxx"></pre> (apiKey 要分開發時期和上架的金鑰，放錯會是沒有圖的圖台)
3	定義使用 map 函式在 AndroidManifest.xml	<pre><uses-library android:name="com.google.android.maps" android:required="true"></uses-library></pre>
4	初始化圖台物件	<pre>private MapView map; @Override public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.maplayout); map = (MapView) findViewById(R.id.mapview); map.setClickable(true);}</pre>

v1 圖台使用者定位開發技巧

	使用步驟	程式片段
1	全域變數：使用者位置圖層	<pre>private MyLocationOverlay m_MyLocationOverlay;</pre>
2	使用者圖層變數初始化	<pre>m_MyLocationOverlay = new MyLocationOverlay(MyMapActivity.this, map); m_MapView.getOverlays().add(m_MyLocationOverlay); 以下 2 行可放在 onResume 去重新啟動 m_MyLocationOverlay.enableMyLocation(); m_MyLocationOverlay.enableCompass(); 以下 2 行可放在 onPause() 去停止運作 m_MyLocationOverlay.disableMyLocation(); m_MyLocationOverlay.disableCompass();</pre>
3	取得新座標的函式	<pre>Public void getUserLoction(){ GeoPoint point = m_MyLocationOverlay .getMyLocation(); }</pre>

v1 圖台基本功能

	函式	使用說明
1	map.getZoomLevel()	取得目前 Zoom 到圖台第幾層
2	map.getController().setZoom(16)	縮放到圖台第幾層
3	map.getController().animateTo(geopoint)	平移到以 geopoint 為中心的圖台中心
4	map.getOverlays().add(MyOverlay)	圖台加入套疊點位物件 (ItemizedOverlay<OverlayItem>) (進階可以客製 1 個物件繼承抽象 ItemizedOverlay<OverlayItem>, 可以搜集同性質 marker, 及客製點擊 marker 彈出視窗) (請注意: 我們需要呼叫 populate 函式, 通知 Map 來更新這個 Overlay。並且, 最好是在呼叫 populate 之前, 呼叫 setLastFocusedIndex(-1), 清掉之前的 Focus Item)
5	map.getOverlays().remove(MyOverlay)	圖台移除套疊物件 (可能是同性質的點位, 線, 或面)

v1 圖台套疊_效能調校

	調校項目	使用說明
1	一次套疊太多點位在圖台上，如果只套疊目前手機畫面4角範圍下的點位或圖層，可透由取得四邊的最小最大經度、最小最大緯度，來套疊點位	<pre>// 提取裝置的寬高 int mWindowHeight = getWindowManager().getDefaultDisplay().getHeight(); int mWindowWidth = getWindowManager().getDefaultDisplay().getWidth(); // 取得圖台的左上、右下兩個點 GeoPoint leftTop = mMViewMain.getProjection().fromPixels(-5, -5); GeoPoint rightBottom = mMViewMain.getProjection().fromPixels(mWindowWidth + 5, mWindowHeight + 5); reloadMapRegion(leftTop.getLongitudeE6(), rightBottom.getLongitudeE6(), leftTop.getLatitudeE6(), rightBottom.getLatitudeE6()); function reloadMapRegion(int aMinX, int aMaxX, int aMinY, int aMaxY) { //(float) (aMinX / 1E6) 最小經度 //(float) (aMaxX / 1E6) 最大經度 //(float) (aMinY / 1E6) 最小緯度 //(float) (aMaxY / 1E6) 最大緯度 // 即可對點位資料作螢幕範疇內，經緯度最小、最大區間的條件過濾 }</pre>

03₋₂

ANDROID
V2 圖台使用分享

v2 圖台初始化

	使用步驟	程式片段
1	Google APIs Console 網頁 開啓 Google Maps Android API v2 服務，及申請你的開發時期和上架時期的金鑰	利用你開發電腦的 (SHA1 碼；專案 package)，在 google apis console 申請一組 Android key，這一組就是你的開發時期的金鑰。而正式發佈的金鑰，是你用 eclipse 在 Export Signed Application Package 所產生的 keystore 裡，就有一組正式憑證的 SHA1 碼，就可以申請正式的金鑰，只有在上架時，才換這組金鑰，從 Google Play 下載下來時的 APP，才看的到圖台
2	導入 google-play-services_lib	<ol style="list-style-type: none">1. 利用 Android SDK Manager 查看你的 Google Play services，是否已安裝2. 導入一個 Google Play services 專案，Eclipse Import 你 sdk 目錄下的 google/google_play_services/libproject/google-play-services_lib 為一個專案3. 你需要 v2 圖台的專案，必須引入 google-play-services_lib 專案為函式庫
3	定義 map 需引用的資源在 AndroidManifest.xml	<pre><uses-feature android:glEsVersion="0x00020000" android:required="true"/> <meta-data android:name="com.google.android.maps.v2.API_KEY" android:value=" 輸入你所申請到的 API Key" /> <!-- 開發時期和上架的金鑰不同 <meta-data android:name="com.google.android.gms.version" android:value="@integer/google_play_services_version" /> <!-- 需導入 google-play-services_lib</pre>
4	定義 maplayout 裡，放圖台的 tag	<pre><fragment android:id="@+id/mapview" android:layout_width="match_parent" android:layout_height="match_parent" class="com.google.android.gms.maps.MapFragment" /></pre>

	使用步驟	程式片段
5	初始化圖台物件	<pre>private GoogleMap map; private UiSettings uiSettings; public class GoogleMapView extends FragmentActivity { if (map == null) { map = ((SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.mapview)).getMap(); if (map != null) setupMap(); } } private void setupMap() { map.setTrafficEnabled(true); map.setMyLocationEnabled(true); uiSettings = map.getUiSettings(); uiSettings.setZoomControlsEnabled(true); uiSettings.setCompassEnabled(true); uiSettings.setMyLocationButtonEnabled(false); uiSettings.setScrollGesturesEnabled(true); uiSettings.setZoomGesturesEnabled(true); uiSettings.setTiltGesturesEnabled(true); uiSettings.setRotateGesturesEnabled(true); }</pre>

	使用步驟	程式片段
1	定位使用者位置物件初始化及取得新座標	<pre>import android.location.LocationListener; import android.location.LocationManager; public class GoogleMapView extends FragmentActivity implements LocationListener { private GoogleMap map; private LocationManager locationManager; @Override protected void onCreate(Bundle savedInstanceState) { LocationManager service = (LocationManager) getSystemService(LOCATION_SERVICE); boolean enabledGPS = service.isProviderEnabled(LocationManager.GPS_PROVIDER); boolean enabledWiFi = service.isProviderEnabled(LocationManager.NETWORK_PROVIDER); if (!enabledGPS) { Toast.makeText(this, "GPS signal not found", Toast.LENGTH_LONG).show(); Intent intent = new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS); startActivity(intent); } locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE); // Define the criteria how to select the location provider -> use Criteria criteria = new Criteria(); provider = locationManager.getBestProvider(criteria, false); Location location = locationManager.getLastKnownLocation(provider); if (location != null) onLocationChanged(location); } @Override protected void onResume() { super.onResume(); locationManager.requestLocationUpdates(provider, 400, 1, this); } @Override protected void onPause() { super.onPause(); locationManager.removeUpdates(this); } @Override public void onLocationChanged(Location location) { double lat = location.getLatitude(); double lng = location.getLongitude(); } }</pre>

v2 圖台基本功能

	函式	使用說明
1	<code>map.getCameraPosition().zoom)</code>	取得目前 Zoom 到圖台第幾層
2	<code>map.moveCamera(CameraUpdateFactory.zoomTo(14))</code>	縮放到圖台第幾層
3	<code>map.moveCamera(CameraUpdateFactory.newLatLngZoom(latLng, map.getCameraPosition().zoom))</code>	平移到以 <code>latlng</code> 為中心，圖台層數不變的圖台中心
4	<code>map.addMarker(new MarkerOptions().position(coordinate).title("title").snippet("snippet").icon(BitmapDescriptorFactory.fromResource(R.drawable.ic_launcher))));</code>	圖台加入點位物件
5	<code>map.addCircle(CircleOptions)</code>	圖台加入圓物件
6	<code>map.addPolygon(PolygonOptions)</code>	圖台加入多邊形物件
7	<code>map.addPolyline(PolylineOptions)</code>	圖台加入線物件
8	<code>map.addTileOverlay(TileOverlayOptions)</code>	圖台加入圖層物件
9	<code>map.clear()</code>	清除圖台上所有物件
10	圖台物件 <code>.remove()</code>	Marker 物件、 Polyline 物件、 Polygon 物件從圖台上移除



04

專案經驗分享

大甲媽祖 APP 提供多圖層需求的呈現

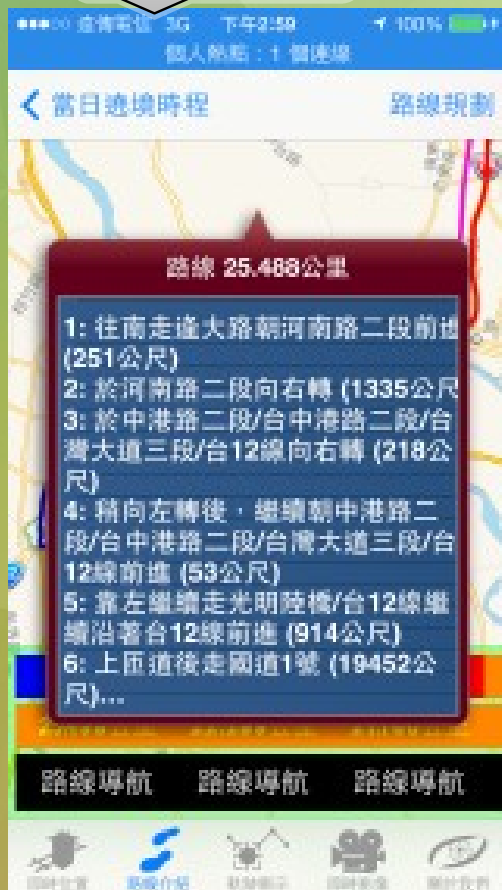


Google 圖台的路線規劃

停轎點路線 規劃



停轎點路線 規劃說明



由於路線規劃是用 WebService 的方式，用的是 Server 端和 Google 要求服務，由於 Google 限制一個網路位置 1 天只能有 2500 次的要求，如果此 APP 服務路線規劃或座標轉住址的需求很大，會很快的喪失 APP 的功能。

原生圖台在 v2 有了很多便利的 API 和效能調整，如果要提供路線規劃或座標轉住址服務可以用 **v3 網頁** 的方式 (以手機為 Client 向 Google 提出要求)，也許就可以滿足每隻手機地址定位的便利功能

■ 值得努力的方向

離線版
GoogleMap

Google 圖台的合用

1. V1 和 V3 合用
2. V2 和 V3 合用

車機訊號

1. Node.js 測試是否可應付上千, 上萬條 client 的接收

感謝大家的聆聽 ^^

