

## 采用 google 地图的 Android 系统导航应用设计

### 一、引言

在当今社会，手机已经成为人们日常生活中不可或缺的工具，以用户体验为核心诉求的智能手机，使手机由单一的通信终端发展成为互联网终端。越来越多的互联网应用被移植到智能手机中来，导航软件在智能手机中的应用成为了研究热点之一。在目前大部分智能手机都提供 GPS 定位服务以及 3G 无线网络越来越普及的双重驱动下，在 Google 地图中实现导航服务，是具有可行性的。

本文提出并实现了一种基于 Google Map Api 的 Android 导航应用，能够给用户 提供人性化和智能化的地图导航服务。

### 二、系统体系结构

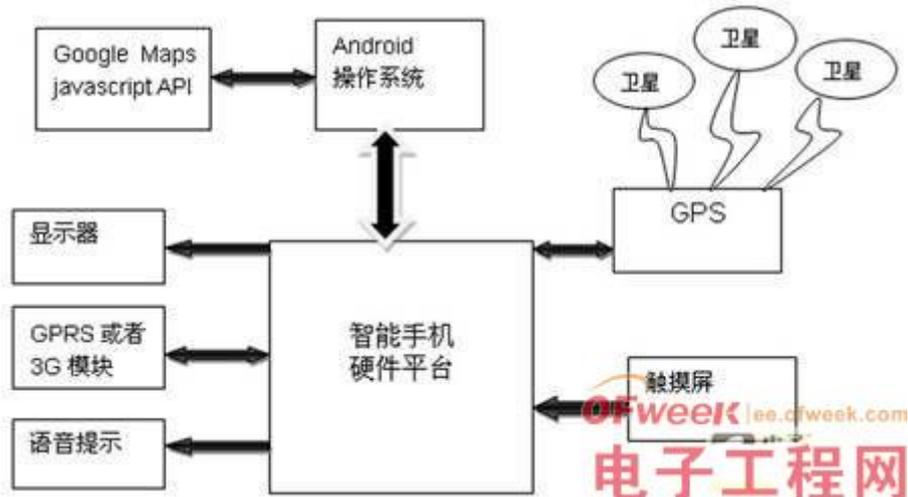


图 1 系统体系结构

### 三、UI 界面设计

Android 应用程序的基本功能单元就是 Activity 类中的一个对象，Activity 主要功能为界面显示和事件处理。在 Activity 中使用 View 和 ViewGroup 控件配 XML 样式来进行界面设计，使用事件处理进行人机交互，主要的事件处理为按钮事件、触屏事件以及一些高级控件的事件监听。通过设置控件与事件处理就形成了 UI 界面。

#### 1、GPS 定位信息的显示

通过 textView 控件显示定位信息。在程序打开时就显示目前用户的具体位置和邮编，门牌号等相关信息，若 GPS 定位不成功或者出错，则显示“无法获取目前地址”。

## 2、用户输入导航起始地点

通过设置 EditView 控件,用户能在改控件中输入自己需要导航的起始地点,通过设置名为“导航”的 Button 控件对输入信息进行提交,当监听到 Button 按钮 OnClick 时则分析起始地址,若起始地址都能被解析,则载入导航地图,若又一个地址解析失败,则通过对话框控件 Dialog 提示用户“输入地址错误,请重新输入”。

## 3、导航界面设计

在导航界面中设置 WebView 控件,WebView 控件通过与 javascript 互相调用,把起始地址等信息传递给服务器中的 HTML 导航网页。网页获得地址等信息后进行导航并通过 WebView 控件显示在应用程序导航界面中。

## 4、语音功能界面设计

在导航界面中设置名为语音导航的 Button 控件,当用户按住该按钮时就为用户提供语音导航服务。

下图为定位与输入地址信息 UI 界面:



图 2 定位与输入地址信息 UI 界面

## 四、GPS 定位功能实现

全球卫星定位系统 (Global Positioning System, GPS) 是一个中距离圆型轨道卫星导航系统, 它可以为地球表面超 98% 的地区提供准确的定位、测速和高精度的时间标准服务, 在各个领域都有广泛的应用。

Android 支持 GPS 服务的 API. 该服务可以用来获取当前设备的地理位置, 应用程序通过该 API 获得地理位置, 并可以定时请求更新当前设备的地理定位信息。该系统将用到 GPS API 中包几个重要的功能类:

LocATionManager, 用于管理 Android 用户定位服务; LocationProvider, 提供多种定位方式供开发者选择; Criteria, 是应用通过 LocationProvider 中设置的属性来选择合适的定位提供者。

获取用户当前位置的步骤如下:

1、在 AndroidManifest.xml 中声明相应的权限:

```
< android:name=" "android.permission.ACCESS_COARSE_LOCATION" "
permission>
```

2、获取 LocationManager 对象:

```
LocationManager locationManager =(LocationManager) getSystemService
(Context.LOCATION_SERVICE) ;
```

3、设置 Criteria 对象:

```
Criteria criteria = newCriteria () ;
criteria.setAccuracy (Criteria.ACCURACY_FINE) ;
criteria.setAltitudeRequired (false) ;
criteria.setBearingRequired (false) ;
criteria.setCostAllowed (false) ;
criteria.setPowerRequirement (Criteria.POWER_LOW) ;
```

4、设置 Provider:

```
String provider = locationManager.getBestProvider(criteria, true);
Location location= locationManager.getLastKnownLocation(provider);
```

五、导航功能实现

在正式版本的 Android SDK 中，移除了原有 M5 版本里面的 DrivingDirection package，所以无法透过程序来规划导航线路。因此导航功能的实现我们是通过载入 web 导航网页的方式实现的。Android 提供了一个名为 WebView 的控件来专门浏览网页。

WebView 控件是基于 WebKit 浏览器网页排版引擎的一个 java 接口，它能够完美支持 html、javascript、css。有时我们可以把整个 UI 甚至数据处理都交给 WebView，配合 PHP 等服务器端程序，这样的 Android 应用开发更为简单 UI 更为丰富。因此本系统的导航功能的主要工作就集中在编写 Google Map Api javascript 版本的导航网页以及 WebView 与 javascript 的数据交互中。

### 1、WebView 的定义及设置：

首先要在 XML 布局文件中定义 WebView 控件；然后在程序中装载这个控件，设置属性；最后通过 loadUrl 方法实现网页的载入。

WebView 常用的类有：WebSettings、WebViewClient、WebChromeClient。

WebSettings 用来设置 WebView 的一些属性状态，如：

```
// 设置可以访问文件
```

```
webSettings.setAllowFileAccess (true) ;
```

```
// 设置支持缩放
```

```
webSettings.setBuiltInZoomControls (true) ;
```

WebViewClient 专门辅助 WebView 处理各种通知、请求等事件类，通过 WebView 的 setWebViewClient 方法来指定一个 WebViewClient 对象，如下：

```
webView.setWebViewClient (newWebViewClient ()  
{  
    public boolean shouldOverrideUrlLoading (WebViewview, String url)  
{  
    view.loadUrl (url) ;  
    return true;  
    }  
})
```

WebChromeClient 类专门用来辅助 WebView 对话框、网站图标、网站 Title、加载进度等。使用方法如下：

```
webView.setWebChromeClient (newWebChromeClient ()  
  
{  
  
@Override  
  
// 该方法为处理 javascript 中的 alert 事件  
  
public boolean onJsAlert (WebView view, String url, Stringmessage,  
JsResult result) {  
  
// TODO Auto-generated method stub  
  
return super.onJsAlert (view, url, message, result) ;  
  
});
```

设置完 WebView 后主要工作就是 Html 网页的编写。

## 2、编写 Html 网页实现线路规划：

线路规划的前提是必须要指定起始地址以及出行方式，本系统用户是通过在 Activity 中输入地址和选择出行方式来向系统提交信息。WebView 必须从应用中获取信息，然后发送给 Html，供 javascript 调用并实现导航。

在 android 应用端，定义一个类保存用户输入的起始地址的经纬度信息：

// 在 javascript 脚本中调用得到 LatlngData 对象

```
public LatlngData getLatlngData ()  
  
{ return latlngData;  
  
}
```

/\* 定义一个 latlngData 类，保存经纬度信息，经纬度由 Geocoder. getFromLocationNAME () 方法按照用户输入地址解析而得到。\*/

```
class LatlngData  
  
{ String startLatlng;
```

```
String endLatlng;

public LatlngData ()

{

this.startLatlng= “” ;

this.endLatlng = “” ;

}

public String getStartLatlng ()

{

return startLatlng;

}

public String getEndLatlng ()

{

return endLatlng;

}

}
```

// 通过 LatlngDataInterface () 方法将数据类绑定给 javascript，这样 javascript 就能够获得了用户的数据了。

```
public class DataJavaScriptInterface{

public LatlngData LatlngDataInterface () {

latlngData = getLatlngData () ;

return latlngData;

}

}
```

在 Html 网页中编写以下代码获得数据，并通过格式转换使数据能够为线路规划所接受，代码如下：

```
var latlngData = window.demo.LatlngDataInterface ();  
  
// 获取起点和终点位置的经纬度信息为 String 格式  
  
startLatlng = latlngData.getStartLatlng ();  
  
endLatlng = latlngData.getEndLatlng ();  
  
// 把 String 格式经纬度转换为 Float 格式  
  
startLat = parseFloat (startLatlng.substring (0, 8) );  
  
startLng = parseFloat (startLatlng.substring (10, 18) );  
  
endLat = parseFloat (endLatlng.substring (0, 8) );  
  
endLng = parseFloat (endLatlng.substring (10, 18) );  
  
通过 DirectionsService 对象计算路线 (使用各种交通方式)。
```

此对象与 Google Maps API Directions Service 进行通信, 该服务将接收路线请求并传回计算结果。可以自行处理这些路线结果, 也可以使用 DirectionsRenderer 对象渲染这些结果:

```
function calcRoute () {  
  
start = beginLatlng;  
  
end = endLatlng;  
  
var request = {  
origin:start,  
destination:end,  
travelMode: google.maps.DirectionsTravelMode.DRIVING  
  
};  
  
directionsService.route (request, function (response, status) {  
  
if (status == google.maps.DirectionsStatus.OK) {  
  
var warnings = document.getElementById ( "warnings_panel" );
```

```
warnings.innerHTML = “” + response.routes [0].warnings + “” ;  
  
directionsDisplay.setDirections (response) ;  
  
showSteps (response) ;  
  
}  
  
});  
  
}
```

通过上述的操作我们基本上实现了线路规划,在此基础上我们能够添加更多的功能,如语音导航、路况提示、周边配套提示等,使用户获得更好的用户体验。最终效果如下图所示:



图 3 最终效果图

## 六、结束语

智能手机导航应用无疑将会有广阔的市场前景,能够为用户提供更多有用信息的导航软件将是未来该应用方向的必然趋势,Google Maps 作为最早的网上电子地图服务,其强大的功能和开放的 API,使得我们能够在其基础上二次开发

---

出很好的应用。具有很大的商用价值。该系统还存在网页载入速度较慢、语音提示功能等问题，还有待进一步研究和开发。

OFweek 电子工程网