

Instructions:

1. Create new project
2. Add the following permissions to your Android Manifest

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

3. Go to:
https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID&reusekey=true and follow steps to get a Google API Key. Be sure to enable Google Maps Android API
4. Take the API Key you created in the previous step and add it to the application object in your AndroidManifest

```
<meta-data android:name="com.google.android.maps.v2.API_KEY"
android:value="YOUR_API_KEY"/>
```

5. Obtain an OpenWeatherMap API Key at
https://home.openweathermap.org/users/sign_up
6. File -> New -> New Module
7. Select Import .JAR/.AAR Package
8. Select the .aar file located in Final_SDK
9. If you want to use the map solution provided in the sdk then go to your main activity and extend WeatherMapActivity.
10. In the onCreate method call super, call initializeMapFragmentWithID passing the id of the view you want the map to appear in, call initializeWeatherManagerWithAppId passing the OpenWeatherMap API Key. Calling detectLocation then will request permission to use gps and if granted will get current weather for your nearest city. Tapping on the map will get weather info for where you tapped. Tapping on a weather marker will display more details for that location.

If you follow these steps you should be good to go!

If you want to make changed to sdk functionality

1. Open the project in WeatherSDK folder
2. Make changes
3. Build -> Build APK
4. Navigate to WeatherSDK/jdweather/build/outputs/aar
5. This is the new .aar file that you will include into your project following the instructions above