

Common Android Errors and Solutions

v1.0

By GoNorthWest

11 April 2012

- **Problem:** AndroidManifest.xml is missing.
 1. The reason for this error is currently unknown. Here are some potential solutions.
 - Turn off everything and restart
 - Verify your installation of Eclipse
 - Verify AndroidManifest.xml actually exists in Package Explorer
 - Right-click on Project, select Android Tools...Build Project Properties
- **Problem:** Project has no default.properties file! Edit the project properties to set one.
 1. There are multiple reasons for this, but it basically refers to a problem with your default.properties file. The easiest way to fix this is to **right-click** on your project in Project Explorer, and pick **Android Tools...Fix Project Properties**.
- **Problem:** "Android requires compiler compliance level 5.0 or 6.0. Found '1.7' instead. Please use Android Tools > Fix Project Properties."
 1. For whatever reason, your project properties are not set correctly.
 - Right-click on your project and select "Properties -> Java Compiler", check "Enable project specific settings" and select 1.5 or 1.6 from "Compiler compliance settings" select box. Try different levels if the first one does not work.
 2. <http://stackoverflow.com/questions/7637144/android-requires-compiler-compliance-level-5-0-or-6-0-found-1-7-instead-plea>
- **Problem:** "Installation error: INSTALL_FAILED_INSUFFICIENT_STORAGE"
 1. It seems like there are a gagillion reasons for this, and no real solutions, but apparently increasing the amount of memory available to your AVD works in many instances.
 - *Solution #1* : Find the .android folder on your system. That's where your AVD configuration files are. In my Windows system, it's located at C:\Documents

and Settings\mf\.android\avd\. Open up the <avdname>.avd file and put this line in it: `disk.dataPartition.size=2048`. What this apparently does it change the size of the partition dedicated to the AVD to 2Gb, which appears to be more than enough for most apps. Since editing this file, I have not received the `INSTALL_FAILED_INSUFFICIENT_STORAGE` message.

- *Solution #2:* Right click the root of your Android Project, go to "Run As" then go to "Run Configurations..." locate the "Android Application" node in the tree at the left, then select your project and go to the "Target" tab on the right side of the window look down for the "Additional Emulator Command Line Options" field (sometimes you'll need to make the window larger) and finally paste "-partition-size 1024" there. Click Apply and then Run to use your emulator.
 - *Solution #3 :* Go to Eclipse's Preferences, then Select "Launch" Add "-partition-size 1024" on the "Default emulator option" field, then click "Apply" and use your emulator as usual.
2. <http://stackoverflow.com/questions/4709137/solution-android-install-failed-insufficient-storage-error>
 3. <http://stackoverflow.com/questions/2239330/how-to-increase-storage-for-android-emulator-install-failed-insufficient-stora>
- **Problem:** Your Android v1.5 or v2.0 app has a map screen in it, and the Google Map tiles are not showing up when the screen is launched. All the pins are there, but no map tiles. Often times this works in the emulator, but then fails to work when you publish the app. Using a combination of experience and some info from @Fred's blog (<http://myskylla.wordpress.com/>), here are the steps to configure maps (not including driving directions) for your Android app.
 1. Obtain a Google Maps API Key. You can use this document to get you through that process:
http://www.buzztouch.com/resources/Obtaining_a_Google_Maps_API_Key_v1.0.pdf
 2. Copy and paste the Release and Debug API Keys into your `\res\values\strings.xml` file. The lines you need are near the top of the file, and look like the following. Once done, save and close the file.
 - `<string name="googleMapsAPIKeyDebug">DebugKey</string>`
 - `<string name="googleMapsAPIKeyRelease">ReleaseKey</string>`
 3. In the `\res\layout\screen_map.xml` file, change the line that reads `android:apiKey="@string/googleMapsAPIKeyDebug"` **TO** `android:apiKey="@string/googleMapsAPIKeyRelease"` (notice all that has changed is the word Debug to Release)

4. In your AndroidManifest.xml file, which is in the root folder of your project, uncomment the lines near the bottom of the file that look like the following. For clarity, check the README file that came with your project.

```
<!-- required libraries for maps -->
```

```
<uses-library android:name="com.google.android.maps"/>
```

5. Save everything, compile your app, sign it using your Release key, and publish!

Comments? Post them in the forum or email me at MrkFleming@gmail.com.

Revision Log			
v1.0	04/11/12	Initial release of document.	GoNorthWest