

From Zero to Market

JAXMUG August 2012





About Me

About Me



- Michael Gidron
- michael.gidron@gmail.com
- Software Engineer
- Windows Mobile, Palm and Android
- Android Apps for the last 2 years

Android is Free and Open

No agreements required

Open Source

Google "Android Open Source Project"

Lots of resources

Documentation is getting better



Android Versions

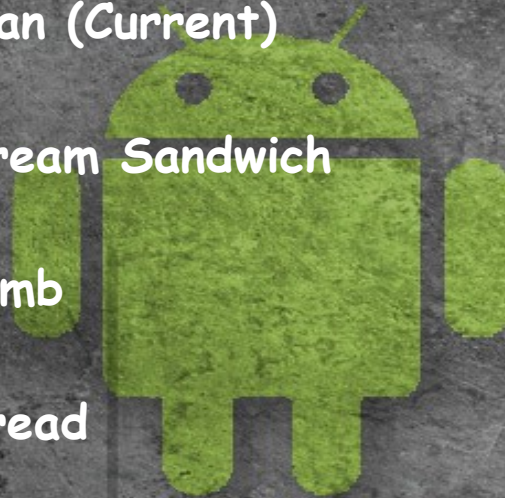
- 4.1 Jelly Bean (Current)

4.0.x Ice Cream Sandwich

- 3.x Honeycomb

- 2.3 Gingerbread

- 2.2 Froyo



Current Market Distribution

Version	Codename	API Level	Distribution
1.5	Cupcake	3	0.2%
1.6	Donut	4	0.5%
2.1	Eclair	7	4.2%
2.2	Froyo	8	15.5%
2.3 - 2.3.2	Gingerbread	9	0.3%
2.3.3 - 2.3.7		10	60.3%
3.1	Honeycomb	12	0.5%
3.2		13	1.8%
4.0 - 4.0.2	Ice Cream Sandwich	14	0.1%
4.0.3 - 4.0.4		15	15.8%
4.1	Jelly Bean	16	0.8%

What you need to get started

Microsoft Windows, Mac OS, and Linux (Ubuntu, Red Hat, ect.)

Eclipse 3.6 or higher

Android SDK

Android ADT Plugin



Publishing your apps

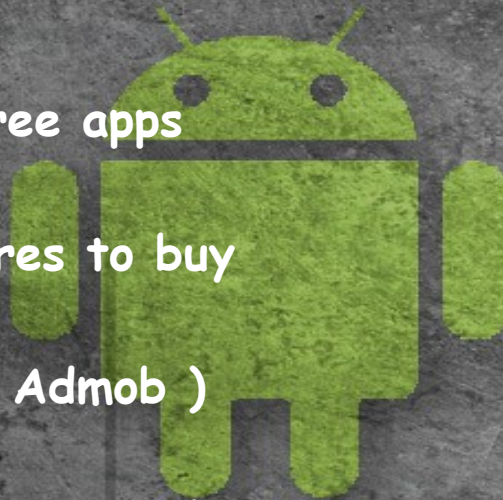
Everyone love free apps

Free with features to buy

Free with Ads (Admob)

Sell for a one time purchase

Sell Subscriptions



Where to publish your apps



One time \$25 fee to **GOOGLE**

One time \$99 fee to **AMAZON**

What is Android ?



Android Application

Application Framework

Activity M. Window M. Content P. View System
Package M. Tel. Manager Resource M. Location M. Notification

Android RT

Core Libraries

Libraries

Media SQLite
Surface M. SGL
Freetype Webkit
Open GL/ libc
SSL

Android RT Dalvik VM

Linux Kernel

Supporting Languages

Java

C/C++ NDK

HTML5, CSS, Javascript



Development Environment

The screenshot displays the Eclipse IDE interface for an Android project named "Snake". The central editor shows the following Java code:

```
/* Copyright (C) 2007 The Android Open Source Project */

package com.example.android.snake;

import android.app.Activity;

/**
 * Snake: a simple game that everyone can enjoy.
 *
 * This is an implementation of the classic Game "Snake", in which you control a
 * serpent roaming around the garden looking for apples. Be careful, though,
 * because when you catch one, not only will you become longer, but you'll move
 * faster. Running into yourself or the walls will end the game.
 */
public class Snake extends Activity {

    private SnakeView mSnakeView;

    private static String ICICLE_KEY = "snake-view";

    /**
     * Called when Activity is first created. Turns off the title bar, sets up
     * the content views, and fires up the SnakeView.
     */
}
```

The Package Explorer on the left shows the project structure:

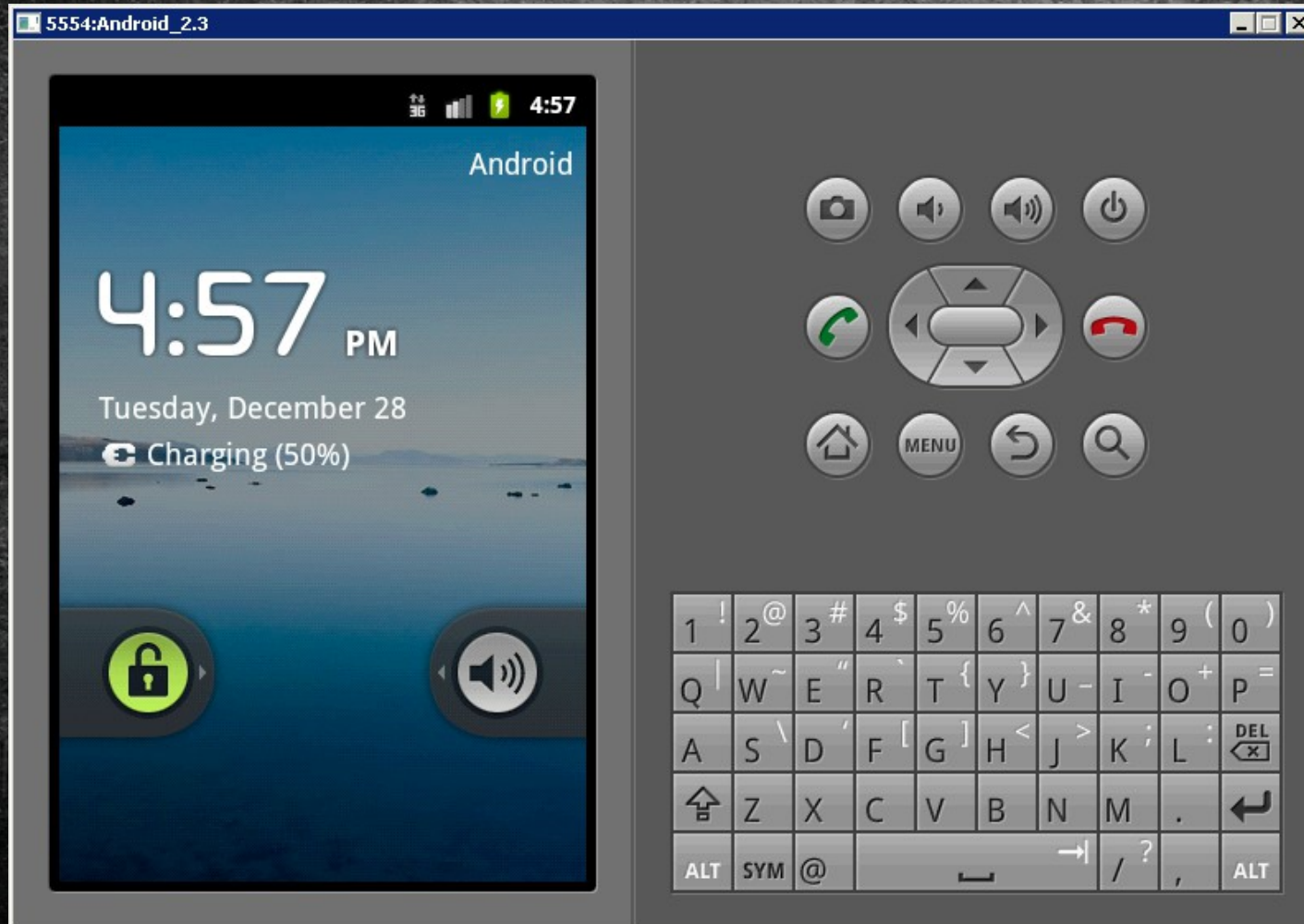
- Snake
 - Android 2.0
 - src
 - com.example.android.snake
 - Snake.java
 - SnakeView.java
 - TileView.java
 - gen [Generated Java Files]
 - assets
 - res
 - tests
 - AndroidManifest.xml
 - default.properties

The Outline view on the right shows the class structure:

- com.example.android.sr
 - import declarations
 - Snake
 - mSnakeView : Snak
 - ICICLE_KEY : String
 - onCreate(Bundle) :
 - onPause() : void
 - onSaveInstanceStat

The Problems view at the bottom shows 0 items. The status bar at the bottom left displays the message: "Project 'Snake' is missing required source folder: 'gen'". The bottom right corner shows "Android SDK Content Loader".

Android Emulator



Eclipse with ADT Plugin

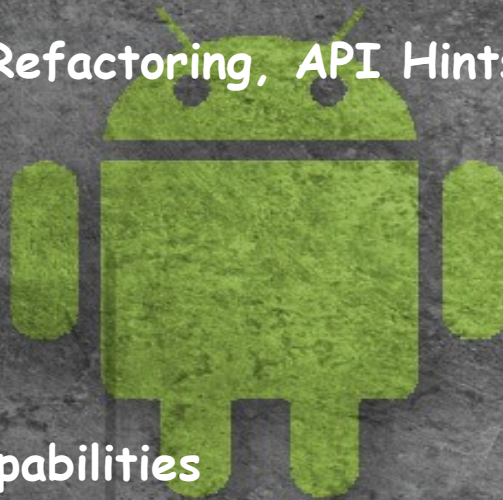
Intellisense, Code Refactoring, API Hints

Resource Editor

Device Debugging

Device exploring capabilities

UI Builder with device preview



What you need to learn

Visit android.com

Beginners - Learn Java Language

Experiences Coders - Follow a few tutorials

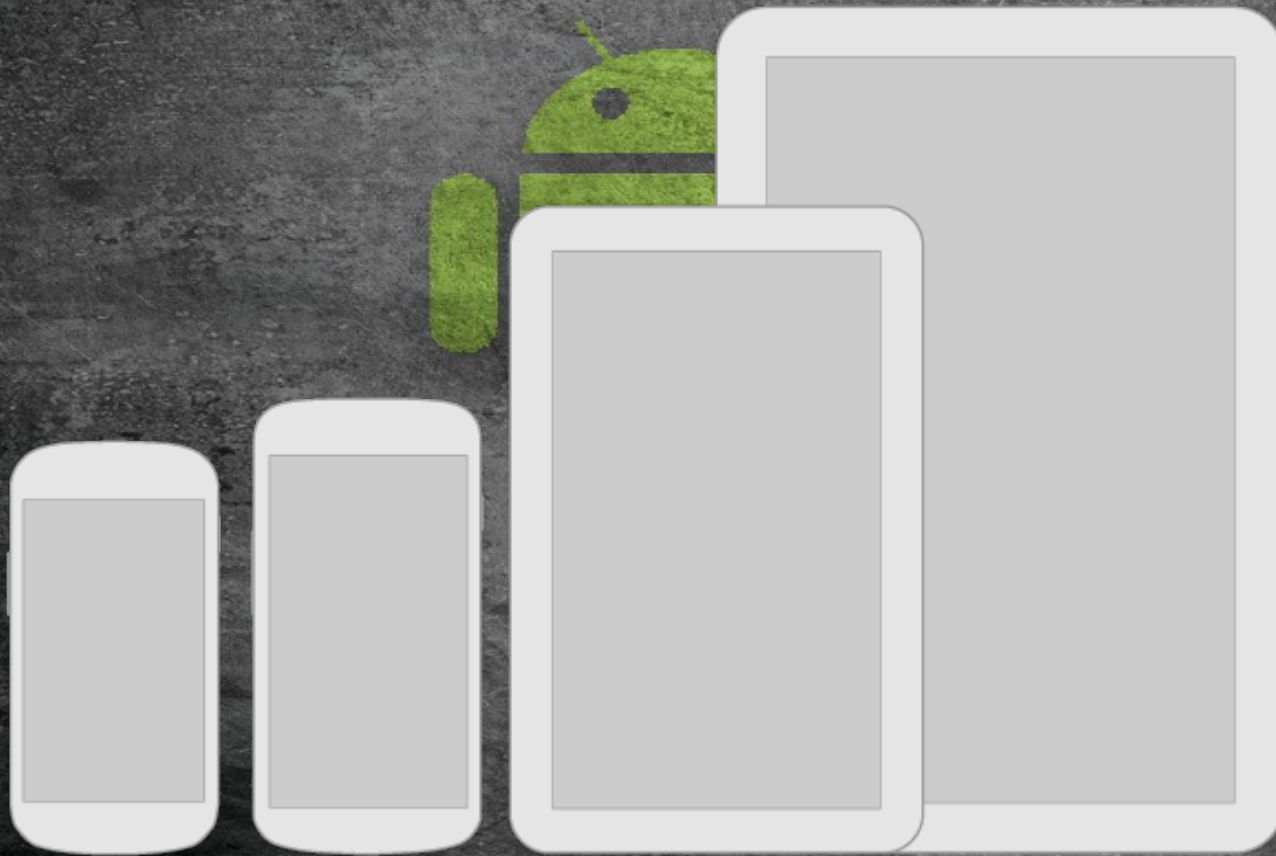
Browse the documentation

Buy a book

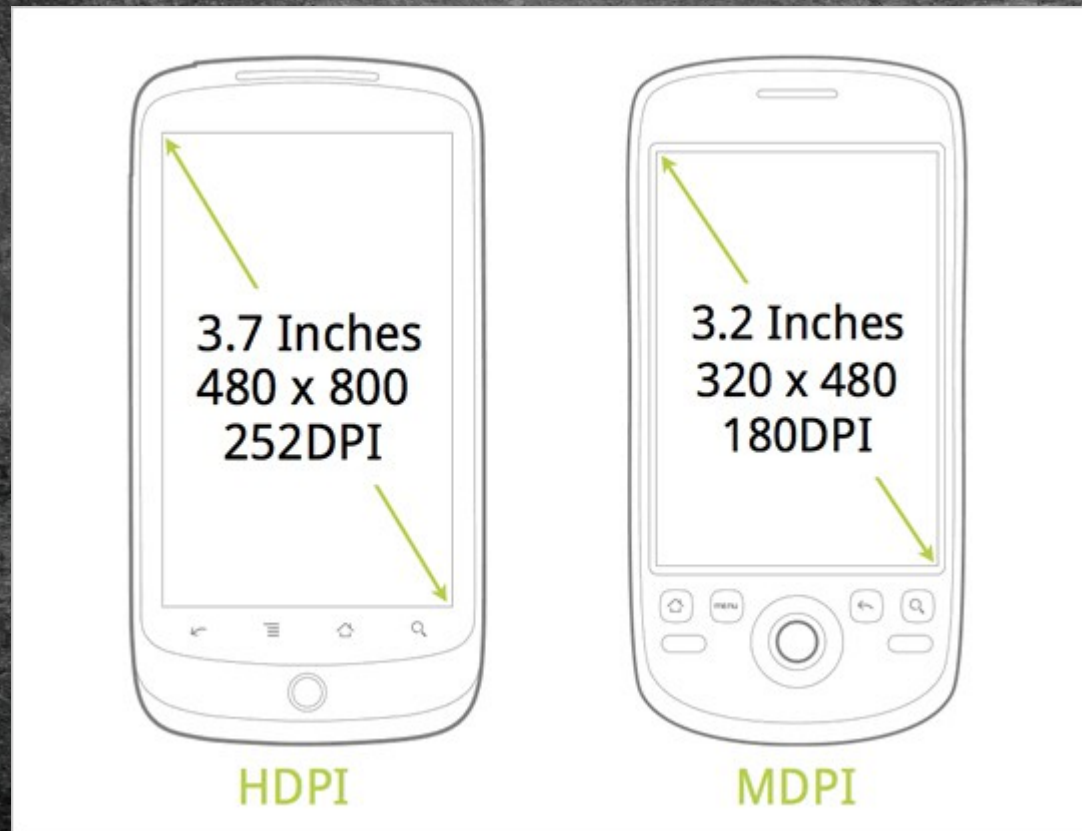
Excellent free video course on marakana.com



Platforms & Device Fragmentation



Screen Sizes



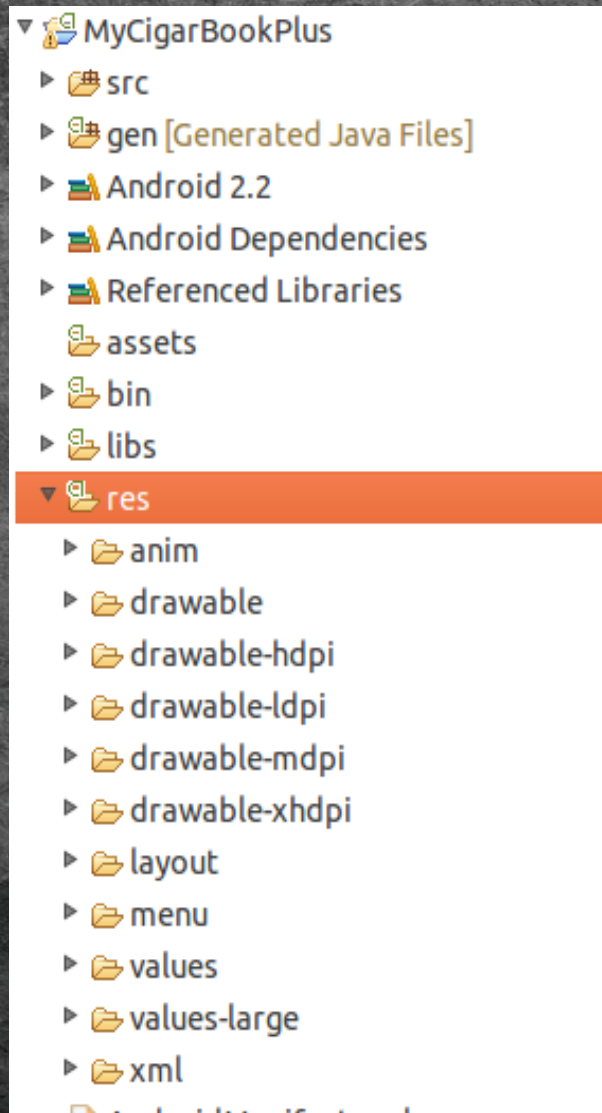
Screen Sizes

	Low density (120), <i>ldpi</i>	Medium density (160), <i>mdpi</i>	High density (240), <i>hdpi</i>	Extra high density (320), <i>xhdpi</i>
Small screen	QVGA (240x320)		480x640	
Normal screen	WQVGA400 (240x 400) WQVGA (240x432)	HVGA (320x480)	WVGA (480x800) WVGA854 (480x854) 600x1024	640x960
Large screen	WVGA800 (480x800) WVGA854 (400x854)	WVGA800 (480x800) WVGA854 (480x854) 600x1024		
Extra large screen	1024x600	WXGA (1280x800) 1024x768 1280x768	1536x1152 1920x1152 1920x1200	2048x1536 2560x1536 2560x1600

Icon Sizes
















How do you manage all of this?



Layouts

- ▶ layout
- ▼ layout-land
 - activity_home.xml
- ▼ layout-large-land-v11
 - activity_home.xml
- ▼ layout-large-v11
 - activity_home.xml
 - activity_map.xml
 - activity_now_playing.xml
 - activity_schedule.xml
 - activity_search.xml
 - activity_sessions.xml
 - activity_starred.xml
 - activity_vendors.xml
 - colorstrip.xml
 - fragment_session_detail.xml
 - fragment_tracks_dropdown.xml
 - tab_indicator.xml
- ▼ layout-v11
 - actionbar_indeterminate_progress.xml
 - actionbar.xml
 - tab_indicator.xml

What does a basic project look like?

- ▼  MyProject
 - ▶  src
 - ▶  gen [Generated Java Files]
 - ▶  Android 4.1
 - ▶  Android Dependencies
 - ▶  assets
 - ▶  bin
 - ▶  libs
 - ▶  res
 - ▶  AndroidManifest.xml
 - ▶  ic_launcher-web.png
 - ▶  proguard-project.txt
 - ▶  project.properties

The Manifest File

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myproject"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="15" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/title_activity_main" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```


Layout Resource

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:padding="@dimen/padding_medium"
        android:text="@string/hello_world"
        tools:context=".MainActivity" />

</RelativeLayout>
```



Java Source Code File

```
package com.example.myproject;

import android.os.Bundle;

public class MainActivity extends Activity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.activity_main, menu);
        return true;
    }

}
```

Activities

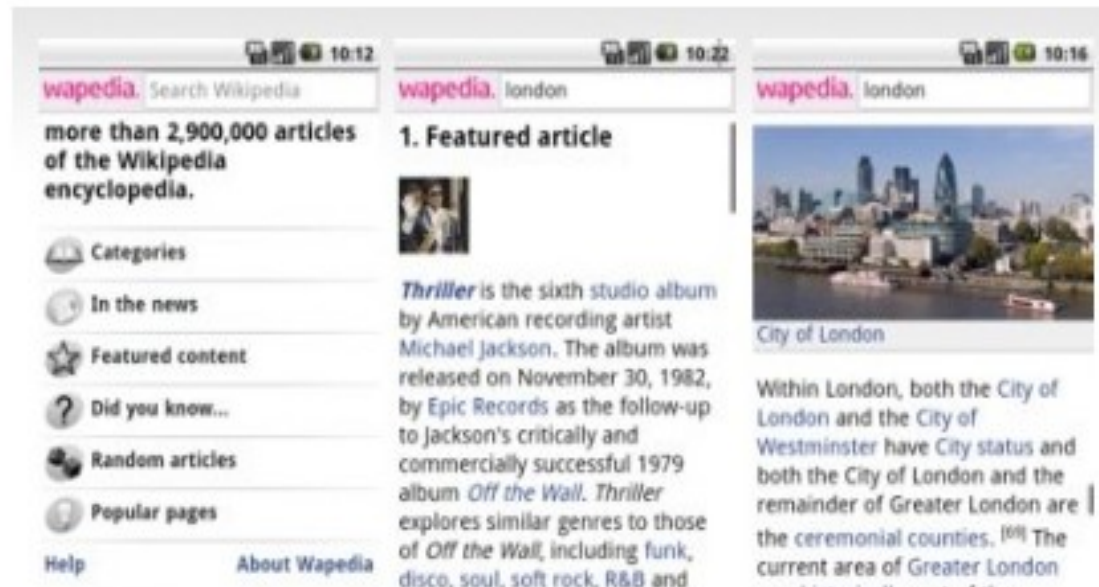
Activity is to an application what a web page is to a website. Sort of.

Android Application

Main Activity

Another Activity

Another Activity



Publishing to Android Market

Sign up for a *GOOGLE* Play membership \$25

Sign your application with a generated hash key
(password protected)

License your application if collecting payments
(*Google License API*)

If your using *Ad Mob*, sign up for *Ad Mob*

In-App Purchase API

Google get 30% of all purchases

INSTANT APP AVAILABILITY !!!

