LAB REPORT

CPET565-499 Lab1(12th September,2012)

By Muhammad S Mansur

Table of Contents

OBJECTIVE

This lab is designed to provide a step by step procedure to start off your own android app development on a Windows operating system. An application that displays "Hello World" is created with results being displayed on an Android Emulator using both Eclipse and the command line tools.

COMPUTER EQUIPMENT & SOFTWARE USED

The following equipment and software's will be required to conduct the test;

- Dell Inspiron (N5110) Laptop Intel Core i5-2450M Processor, 6GB DDR3, 500GB HDD, 64 bit
- Windows 7 Home Premium Edition
- Java JDK version 1.6,
- Android SDK
- Eclipse Classic 3.7.2 Indigo
- Eclipse Plug-In (ADT Plugin)
- Android 4.0.3 (API 15)

PROCEDURE

Part1: Download Java SE Development Kit

- ➢ Turn on your laptop/Desktop which has Windows 7 operating system installed on it
- ➢ Download a copy of Java SE Development Kit (JDK) version 1.6 from the below link

http://www.oracle.com/technetwork/java/javase/downloads/index.html

Part2: Download Android SDK

- ➢ Download a copy of the Android SDK

http://developer.android.com/sdk/index.html

- ➢ Download SDK for Windows
- ➢ Download "installer_r20.0.3-windows.exe" (67.2 MB from dl.google.com)
- ➢ View downloaded file, and chose Run "installer_r20.0.3-windows.exe"
- ➢ Click Next> appear on the Android SDK Tools Setup window
- ➢ Welcome to the Android SDK Tools Setup Wizard (click Next)
- ➢ Detect Java SE Development Kit (Found JDK version 1.7). This is automatically detected
- ➢ Click Next
- ➢ Choose Users (click the radio button: Install just for me), then click Next

- ➢ Choose Install Location (choose default: C:\Users\YourName\AppData\Local\Android\android-sdk), then click Next
- ➢ Choose Star Menu Folder – Android SDK Tools, then click Install
- ➢ Installation Complete – click Next
- ➢ Completing the Android SDK Tools Setup Wizard
- ➢ Click check box: Start SDK Manager (to download system images, etc)
- ➢ Click Finish
- ➢ Install all software packages and tools
- ➢ It will show Android SDK Manager window – which shows all packages available for download
- ➢ Adding Platforms and Packages. It includes a node for Android SDK Tools, Android SDK Platform-tools, each of the platform releases, and a collection of extras.
- ➢ Uncheck Android 4.1 (API 16)
- ➢ Click Check box: Android 4.0.3 (API 15), to install all tools, click on Install 11 packages.
- ➢ Click INSTALL to install the recommended packages (other updates)
- ➢ Review Android SDK Manager Log, click Close when you are OK with the installation

Part3: Download Eclipse

- ➢ Choose: Eclipse Classic 3.7.2 Indigo
  from http://www.eclipse.org/downloads/packages/release/indigo/sr2
- ➢ Download and extract (unzip) Eclipse Classic 3.7.2 Indigo, bring Eclipse to the desktop
- ➢ Run Eclipse Classic 3.7.2 from the extracted folder eclipse-SDK-3.7.2-win32-x86_64 => eclipse folder, then bring the eclipse.exe to the Desktop
- ➢ Start Eclipse from its installed folder
- ➢ Run => downloads\eclipse-SDK-3.7.2-win32-x86_64eclipse.exe

Part4: Install the Eclipse Plugin

- ➢ Installing the Eclipse Plugin
- ➢ Go to the link, http://developer.android.com/sdk/installing/installing-adt.html
- ➢ Eclipse 3.7.2 (Eclipse Classic download) select Help > Install New Software
- ➢ Click Add
- ➢ Place ADT Plugin into the Name box
- ➢ Copy from https://dl-ssl.google.com/android/eclipse/ into the URL
- ➢ And Install and Restart Eclipse

Part5a: Building the App through Eclipse

- ➢ Building the First App, (Creative Commons Attribution 2.5)
- ➢ In Eclipse, click the wizard for New Android Project in the toolbar
- ➢ Now fill in the following info;
  - o Application Name: "Mansur First App."
  - o Project Name: MansurFirstApp
  - o Package Name: "com.example.MansurFirstApp "

o   Build SDK: Choose Android 4.0.3 (API 15)
o   Minimum Required SDK: Use the default Android 2.2
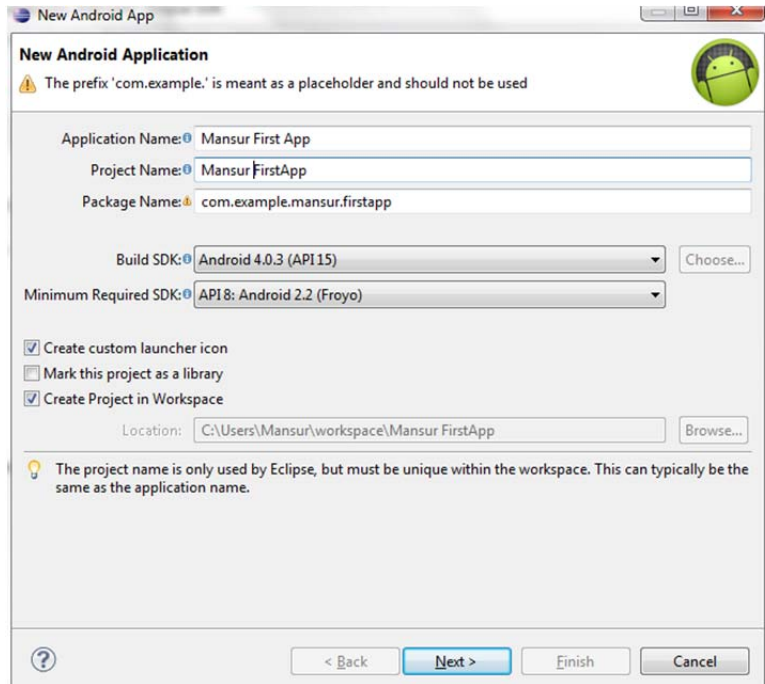
➢ Click Next.
➢ Now configure the launcher icon. Give background color to your icon as you wish;
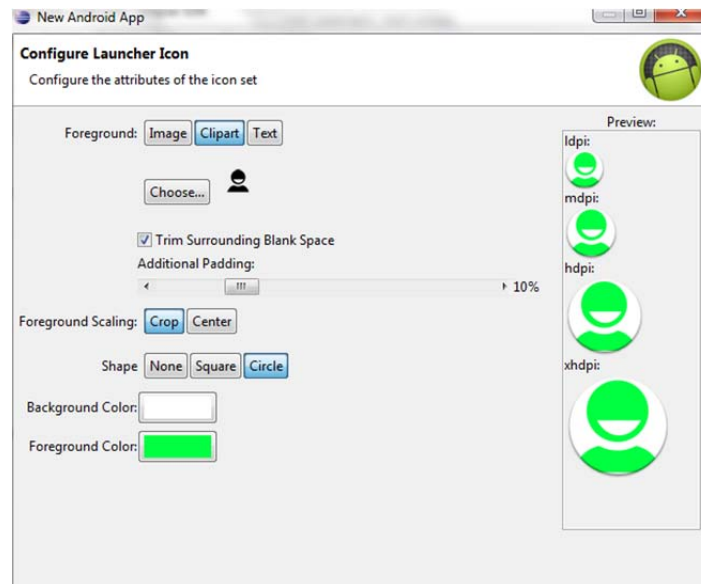
➢ Then choose "blank activity"
➢ Leave all default for the next window and click finish

Part5b: Building the App through Command Line Tools

- Go to the command line, type "cmd" on the windows run program
- Now enter the folder tools by typing the following path;

    cd \Users\Mansur\AppData\Local\Android\android-sdk\tools

- Now type *android list targets* on the cmd
- This prints a list of the available Android platforms that you've downloaded for your SDK

```
C:\Users\Mansur\AppData\Local\Android\android-sdk\tools>android list targets
Available Android targets:
----------
id: 1 or "android-15"
     Name: Android 4.0.3
     Type: Platform
     API level: 15
     Revision: 3
     Skins: HVGA, QVGA, WQVGA400, WQVGA432, WSVGA, WVGA800 (default), WVGA854, W
XGA720, WXGA800
     ABIs : armeabi-v7a

C:\Users\Mansur\AppData\Local\Android\android-sdk\tools>
```

- Now run the following command to create your project

    *android create project --target 1 --name MansurFirstApp1 --path
    C:\Users\Mansur\workspace/MansurFirstApp1 --activity MainActivity --package
    com.example.Mansurfirstapp1*

- Go to *C:\Users\Mansur\workspace\MansurFirstApp1*
- And run "*ant debug*"
- Now run the following command in screenshot to install the project

```
     4 DIR(S)   173,117,212,672 bytes free
C:\Users\Mansur\workspace\MansurFirstApp1\bin>adb install MansurFirstApp1-debug.
apk
45 KB/s (14265 bytes in 0.307s)
     pkg: /data/local/tmp/MansurFirstApp1-debug.apk
Success
```

Part6a: Open the Android Virtual Manager & Device through Eclipse

- In Eclipse, click Android Virtual Device Manager 📱from the toolbar.
- In the Android Virtual Device Manager panel, click New
- Give the following details as per snapshot below;
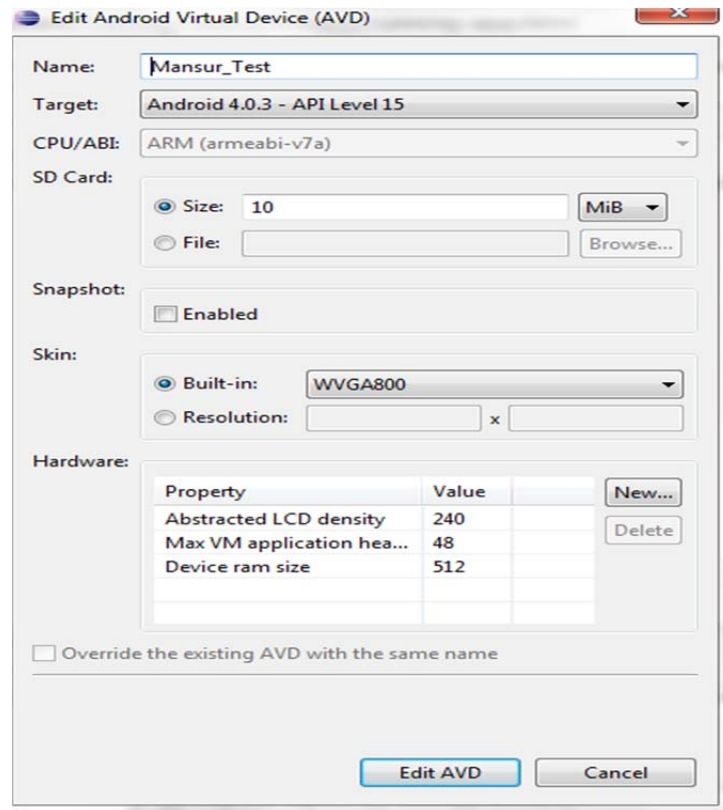
- ➢ Click Create AVD.
- ➢ Select the new AVD from the Android Virtual Device Manager and click Start
- ➢ After the emulator boots up, close all other windows to free the Eclipse window
- ➢ To run the app from Eclipse, open one of your project's files and click Run ▶ from the toolbar. Eclipse installs the app on your AVD and starts it.
- ➢ Emulator shows the window of a standard Android phone
- ➢ Locate MansurFirstActivity and open it.
- ➢ You have successfully run your application on an emulator

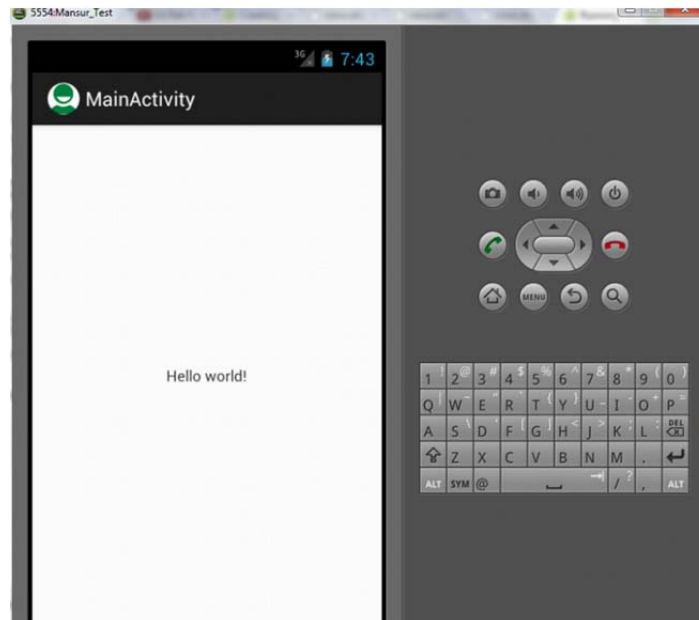**Part6b: Open the Android Virtual Manager & Device through CLI**

- ✦ Launch the Android Virtual device
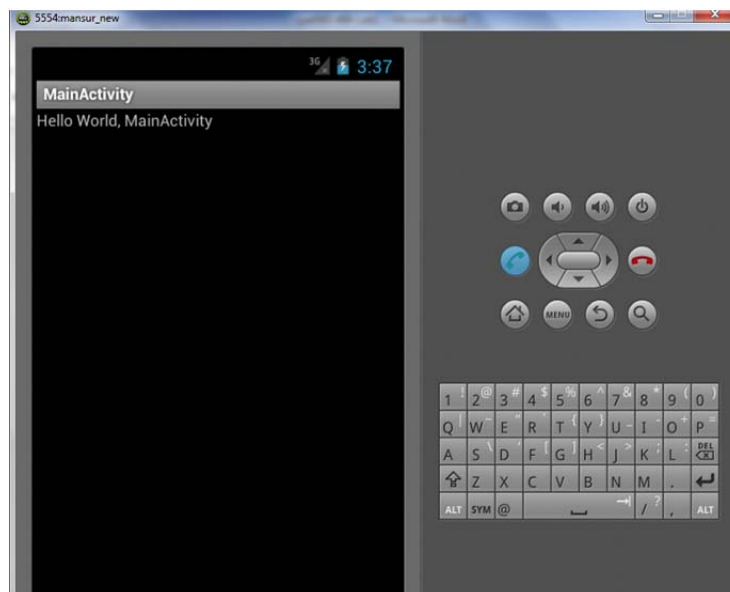

- ✦ Run app on the Emulator

DATA

The application built was displayed as per below incase when done through Eclipse;

Results are successful with "Hello world" displayed on the middle of the screen.

While using Command line the following display was shown on the emulator

This was different then incase of using eclipse and some configuration files were searched and skimmed through.

The Main code for the application can be seen in the file AndroidManifest.xml as shown below

```
C:\Users\Mansur\workspace\MansurFirstApp1\bin>more AndroidManifest.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
        package="com.example.Mansurfirstapp1"
        android:versionCode="1"
        android:versionName="1.0">
    <application android:label="@string/app_name" android:icon="@drawable/ic_lau
ncher">
        <activity android:name="MainActivity"
                    android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

The difference was found in the Layout file which clearly showed why the outputs were different as compared below;

```
C:\Users\Mansur\workspace\MansurFirstApp1\res\layout>more main.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Hello World, MainActivity"
    />
</LinearLayout>
```

```
C:\Users\Mansur\workspace\MansurFirstApp\res\layout>more activity_main.xml
<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:text="@string/hello_world"
        tools:context=".MainActivity" />

</RelativeLayout>
```

CONCLUSION:

The previous lab helped through step by step installation of all the required software tools and plugin's required for Android App development. Additionally, a small application that displayed "hello world" was also successfully built through Eclipse and command line and then displayed on the Android emulator. The following difficulties were faced and should be avoided in future (their solutions are also mentioned below);

- After the first Eclipse installation the software was unable to build a new project. *The Software was uninstalled and then reinstalled again.*

- During the CLI application building issues were faced in debugging and installing the Project created. JRE was installed again, "ant" path was added in the environment variable, tool.jar file was added in the following path C:\Program Files\Java\jre7\lib

REFERENCES:

- Creative Commons Attribution 2.5. (n.d.). *Building your first app*. Retrieved from http://developer.android.com/training/basics/firstapp/index.html

- Lin, D. P. (n.d.). *Lab Activity 1: Tools for Android-Based Applications*. Retrieved 2012, from http://www.etcs.ipfw.edu/~lin/CPET565/2012F/Labs/Lab1-HelloAndroid-2012-9-5.html