Setting Up Your Android Development Environment

For Windows XP v1.3

By GoNorthWest

5 February 2012

Setting up the Android development environment can be a bit…well…challenging if you don't have all the pieces in place before you start. The aim of this guide is to visually walk you through the process of downloading what you need, installing it, and making the correct configurations. Once you do that, you're all set to actually work with the source code you download from Buzztouch! Hopefully these instructions will make things just a bit easier for you!

Let's get started!

1. **Download required software**. In order to make this all work, there are several pieces of software you have to download and install:

   a. **Eclipse**: I chose to download Eclipse Classic 3.7.1, which was the most recent version available at the time of this writing. I downloaded the 32-bit version for Windows XP, but there is also a 64-bit version if you are running 64-bit Windows. Here are the links I used to get this package.

   http://www.eclipse.org/downloads/

   http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops/R-3.7.1-201109091335/eclipse-SDK-3.7.1-win32.zip

   http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops/R-3.7.1-201109091335/eclipse-SDK-3.7.1-win32-x86_64.zip

   b. Android Development SDK: This is the software development kit (SDK), much like the iOS SDK for Macs, that must be used within Eclipse for development. Here are the links I used for this portion.

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

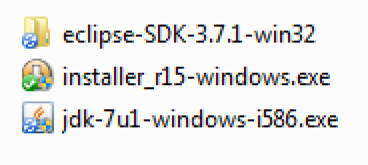
   http://dl.google.com/android/installer_r15-windows.exe

   http://developer.android.com/sdk/eclipse-adt.html#installing

   c. Java JDK: You have to use the Java Developers Kit here…not just the Java Runtime Engine (JRE), which most of you already have installed. This is a very crucial distinction…without it, things will not work.

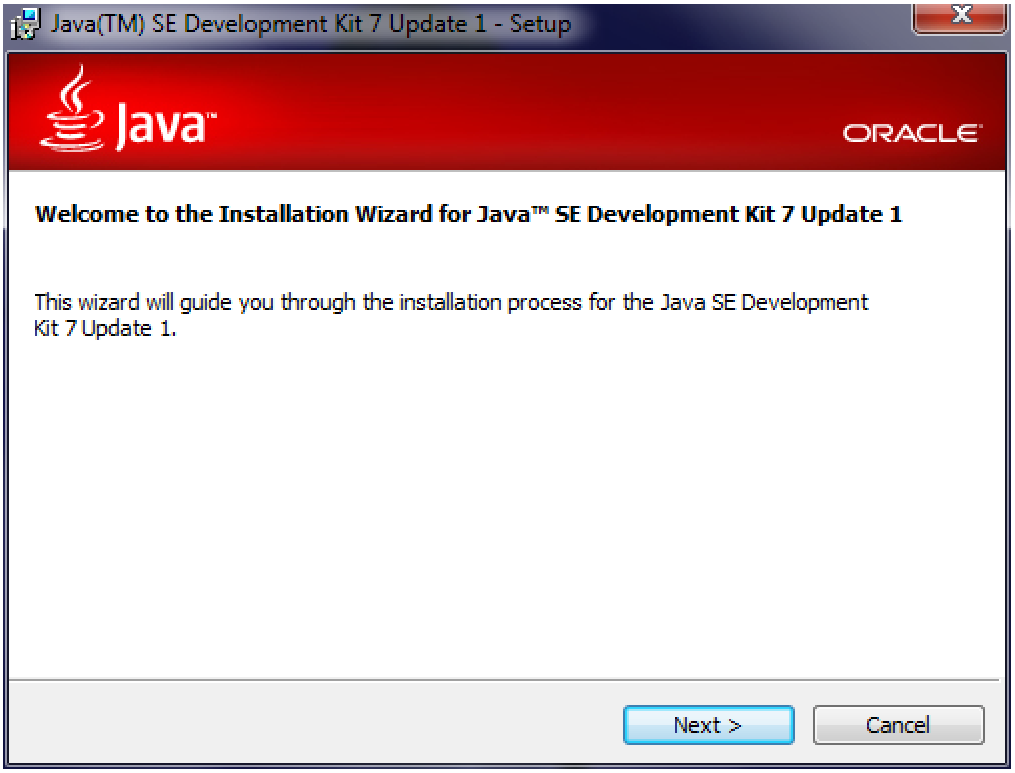
   http://www.oracle.com/technetwork/java/javase/downloads/index.html http://www.oracle.com/technetwork/java/javase/downloads/jdk-7u1-download-513651.html

d. Once you get all the software downloaded, you should have a file listing that looks something like this :



eclipse-SDK-3.7.1-win32
installer_r15-windows.exe
jdk-7u1-windows-i586.exe

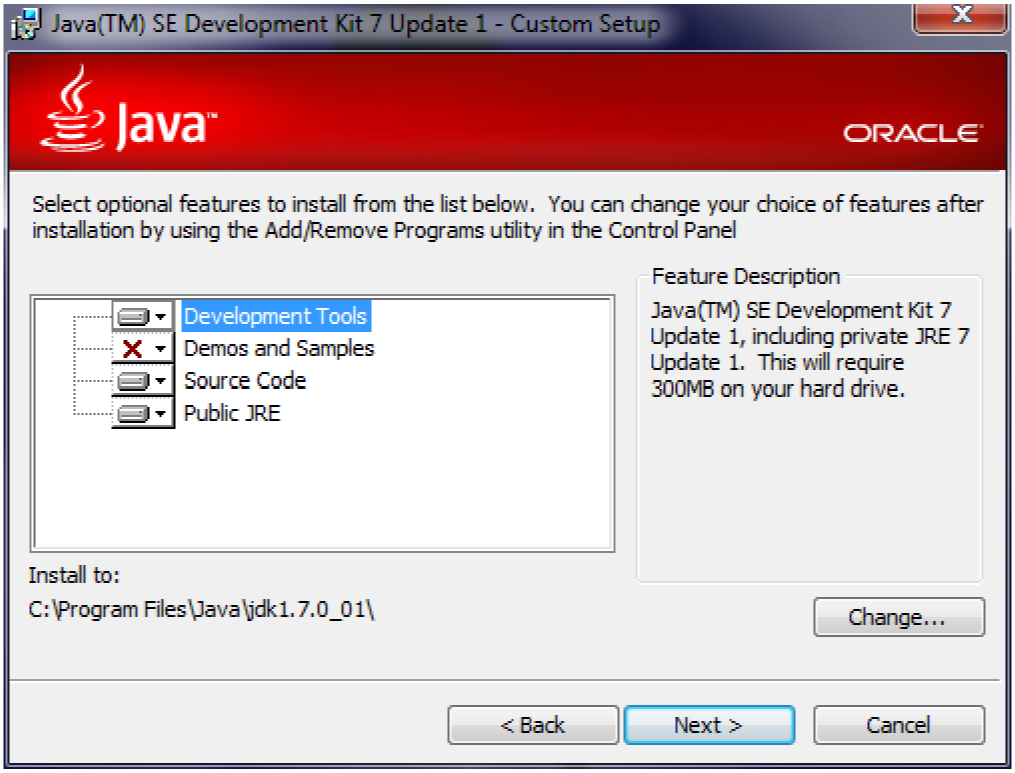
2. **Install the Java JDK (jdk-7u1-windows-i586.exe).** Double-click on the JDK install package you downloaded from above, and follow the prompts. This MUST be installed before you can install the Android SDK…without it, SDK installation will fail. Your install screens should look something like the following:



Java(TM) SE Development Kit 7 Update 1 - Setup

Welcome to the Installation Wizard for Java™ SE Development Kit 7 Update 1

This wizard will guide you through the installation process for the Java SE Development Kit 7 Update 1.



Java(TM) SE Development Kit 7 Update 1 - Custom Setup

Select optional features to install from the list below. You can change your choice of features after installation by using the Add/Remove Programs utility in the Control Panel

- Development Tools
- ☒ Demos and Samples
- Source Code
- Public JRE

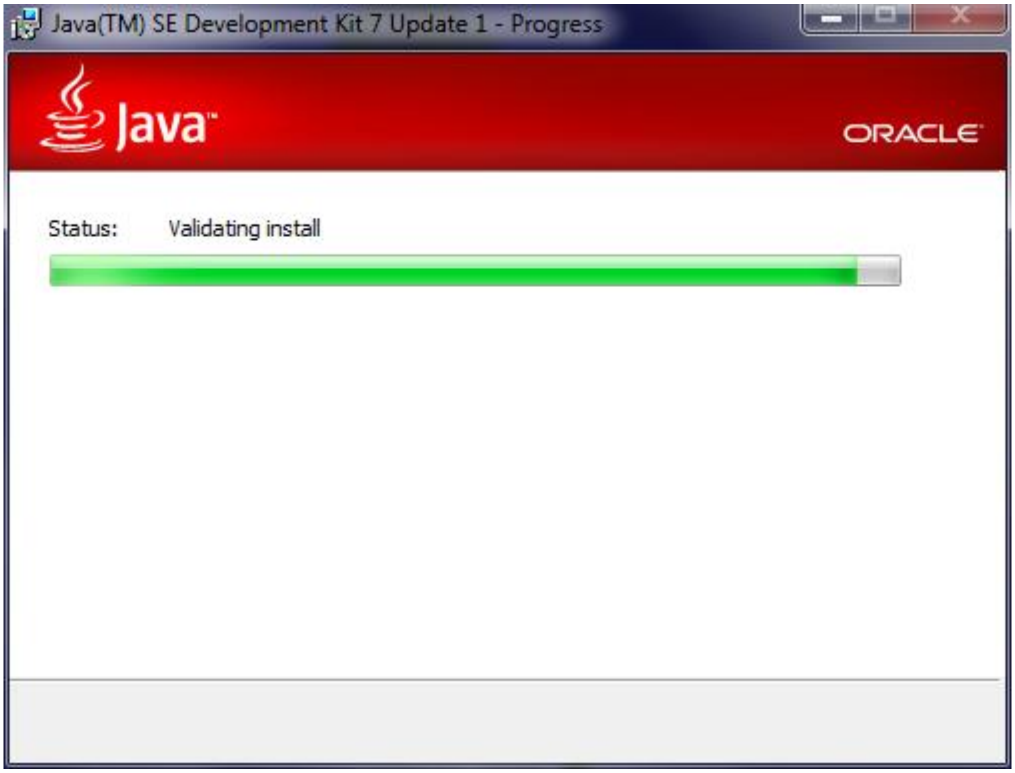
Feature Description

Java(TM) SE Development Kit 7 Update 1, including private JRE 7 Update 1. This will require 300MB on your hard drive.

Install to:

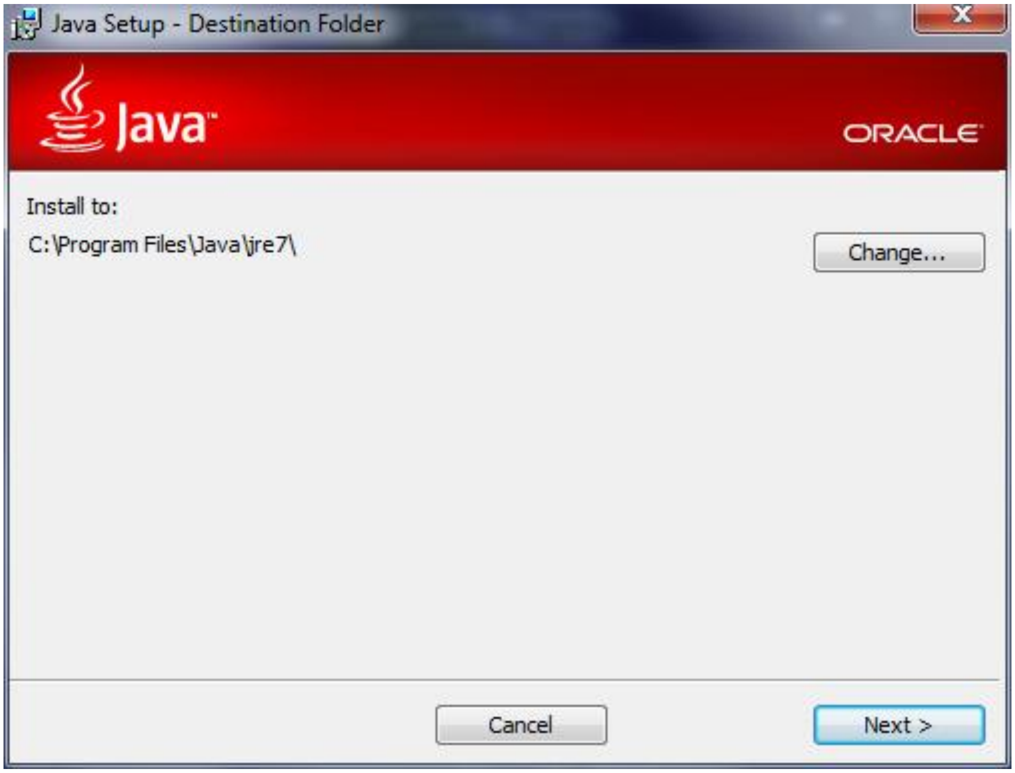
C:\Program Files\Java\jdk1.7.0_01\

Use the screen above to decide which components you want and don’t want. When in doubt, just go with the defaults…it’ll be what you need for your environment.



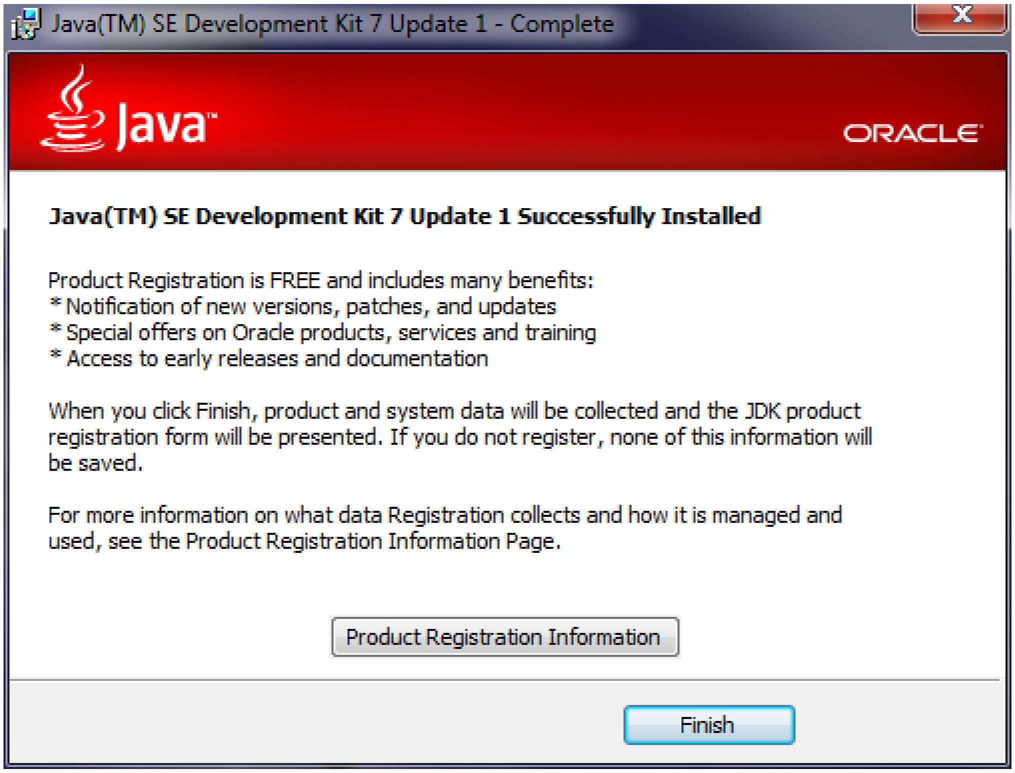
Java(TM) SE Development Kit 7 Update 1 - Progress

Status: Validating install



Install to:

C:\Program Files\Java\jre7\



Java(TM) SE Development Kit 7 Update 1 Successfully Installed

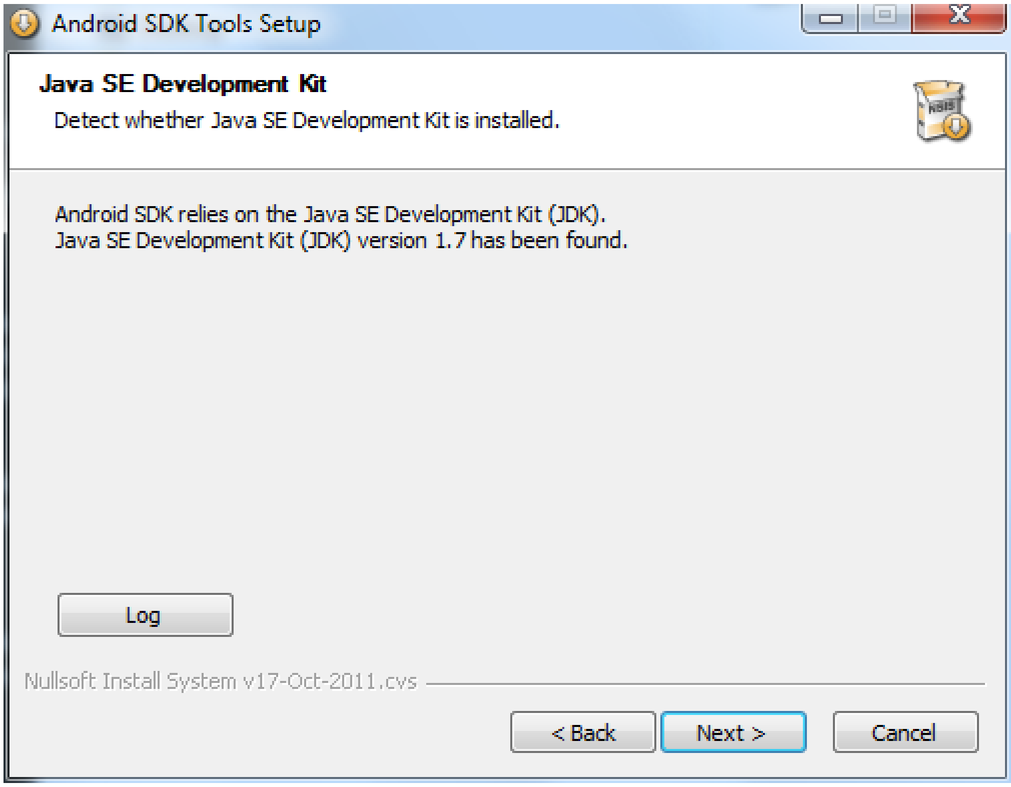
Product Registration is FREE and includes many benefits:
* Notification of new versions, patches, and updates
* Special offers on Oracle products, services and training
* Access to early releases and documentation

When you click Finish, product and system data will be collected and the JDK product registration form will be presented. If you do not register, none of this information will be saved.

For more information on what data Registration collects and how it is managed and used, see the Product Registration Information Page.

Once you have hit the Finish button, you are done with the JDK install, and can move on to the Android SDK Installer.

3. **Install the Android SDK (installer_r15-windows.exe).** Double-click on the Android SDK install package you downloaded from above, and follow the prompts. Your install screens should look something like the following:



Android SDK Tools Setup

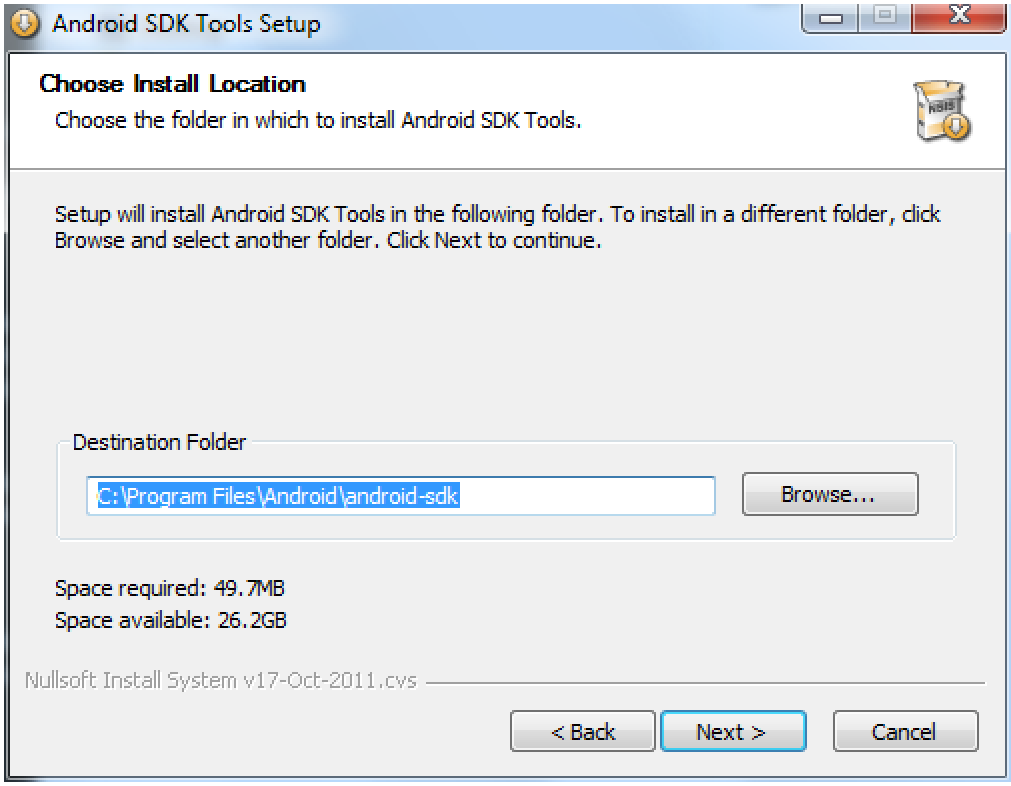
Java SE Development Kit

Detect whether Java SE Development Kit is installed.

Android SDK relies on the Java SE Development Kit (JDK).
Java SE Development Kit (JDK) version 1.7 has been found.

If you don't see the above screen initially, and it complains that it cannot find the JDK, then hit the "Back" button, then "Next" again. This should resolve the problem. If not, check out this article on Stack Overflow :

http://stackoverflow.com/questions/4382178/android-sdk-installation-doesnt-find-jdk



Android SDK Tools Setup

Choose Install Location

Choose the folder in which to install Android SDK Tools.

Setup will install Android SDK Tools in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

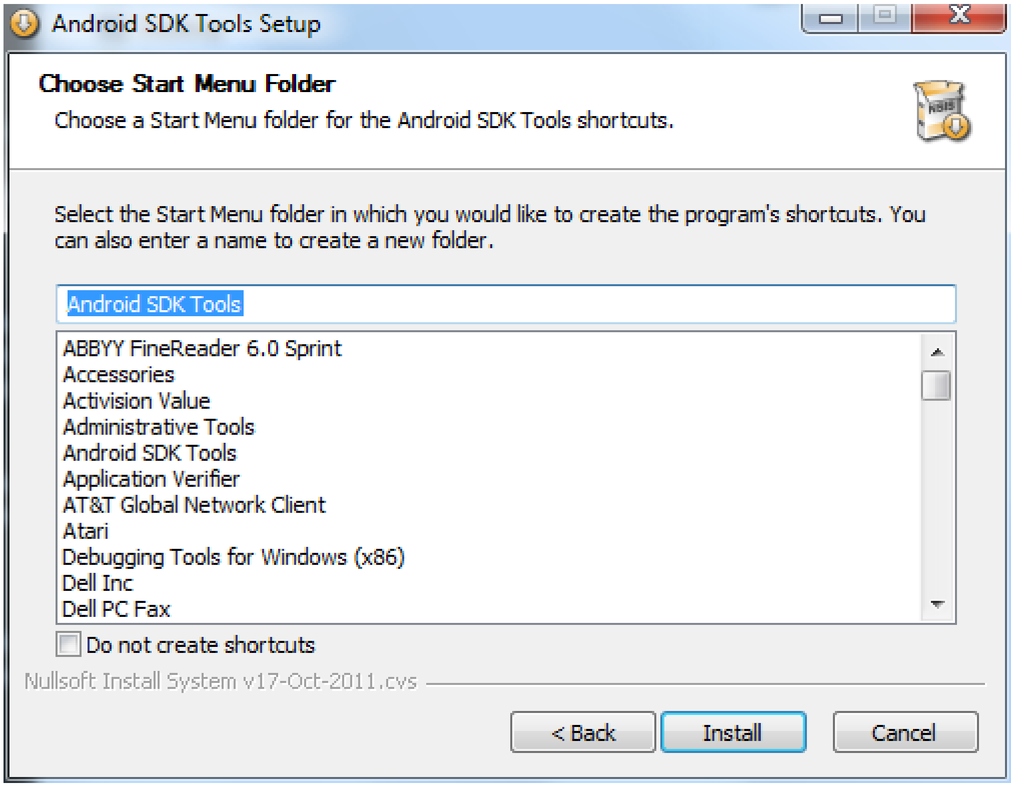
Destination Folder

C:\Program Files\Android\android-sdk

Space required: 49.7MB

Space available: 26.2GB

Nullsoft Install System v17-Oct-2011.cvs



Android SDK Tools Setup

Choose Start Menu Folder

Choose a Start Menu folder for the Android SDK Tools shortcuts.

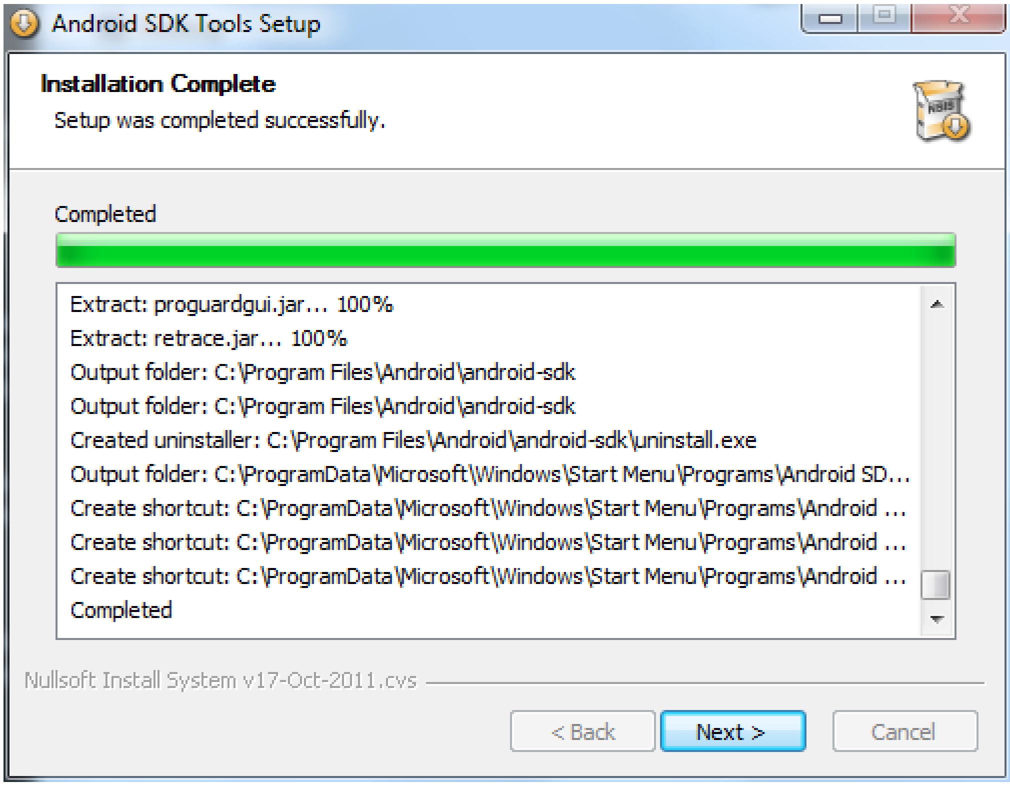
Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

Android SDK Tools

ABBYY FineReader 6.0 Sprint
Accessories
Activision Value
Administrative Tools
Android SDK Tools
Application Verifier
AT&T Global Network Client
Atari
Debugging Tools for Windows (x86)
Dell Inc
Dell PC Fax

□ Do not create shortcuts

Nullsoft Install System v17-Oct-2011.cvs



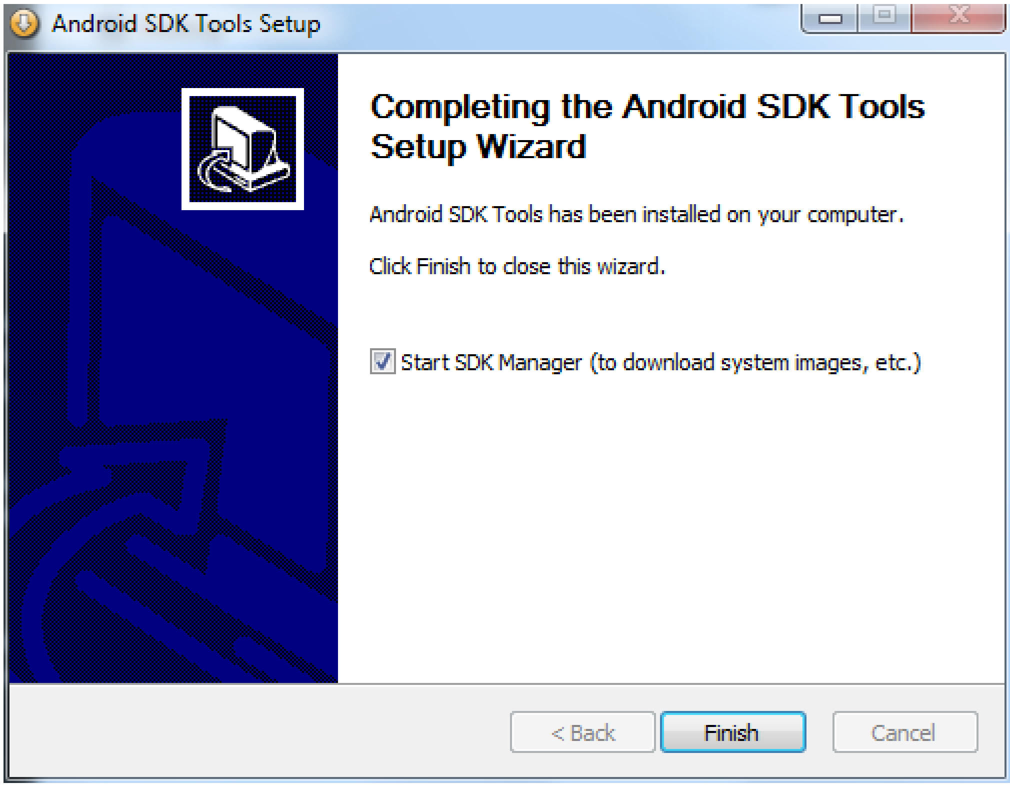
Installation Complete

Setup was completed successfully.

Completed

```
Extract: proguardgui.jar... 100%
Extract: retrace.jar... 100%
Output folder: C:\Program Files\Android\android-sdk
Output folder: C:\Program Files\Android\android-sdk
Created uninstaller: C:\Program Files\Android\android-sdk\uninstall.exe
Output folder: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Android SD...
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Android ...
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Android ...
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Android ...
Completed
```

Nullsoft Install System v17-Oct-2011.cvs



Completing the Android SDK Tools Setup Wizard

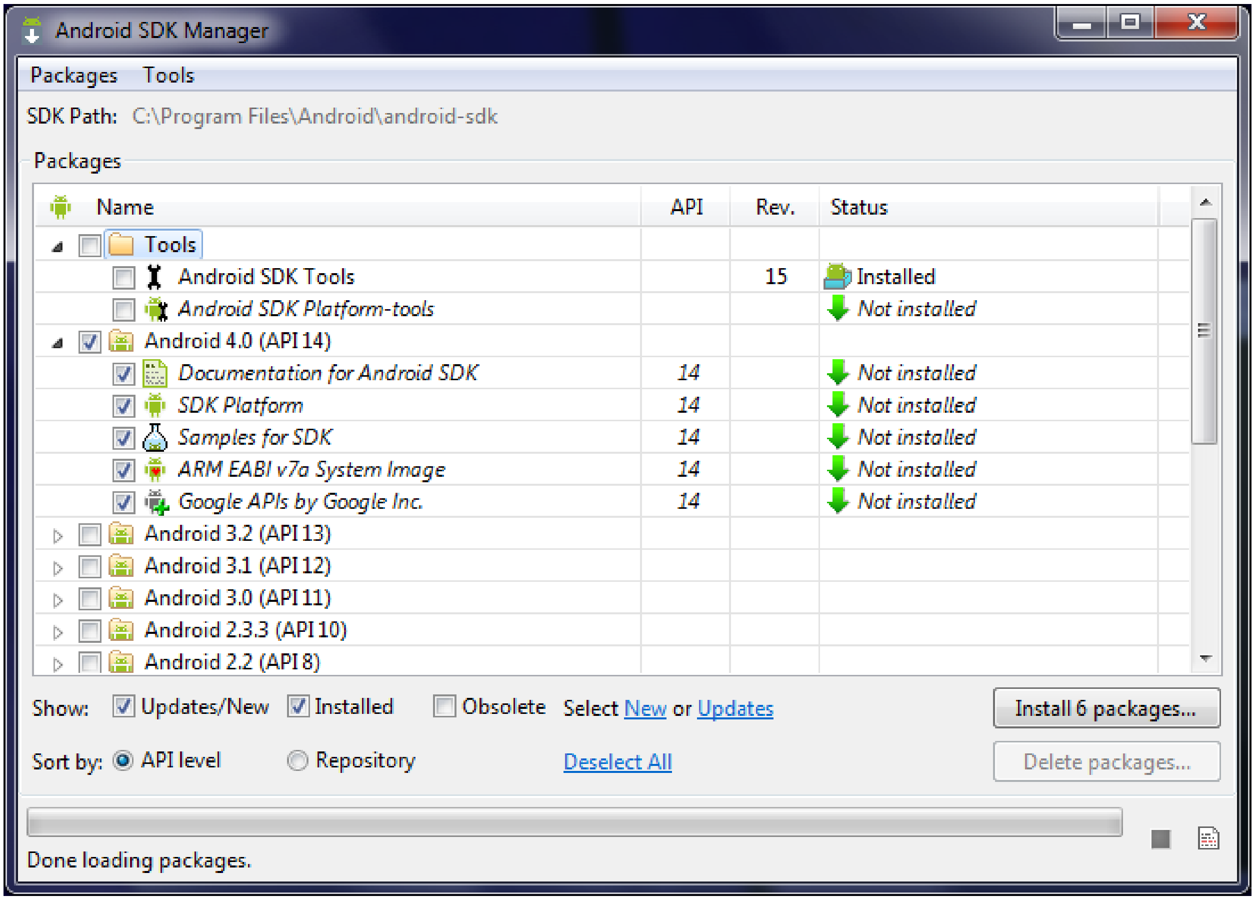
Android SDK Tools has been installed on your computer.

Click Finish to close this wizard.

☑ Start SDK Manager (to download system images, etc.)

You can choose to start the SDK Manager at this point, or move on to Eclipse installation. For the sake of this tutorial, keep "Start SDK Manager" selected, and hit Finish.

4. **Configure the Android SDK for Development Tools and Google API.** This is where you configure the Android SDK Manager to download the tools you will need to create Buzztouch apps. This is essentially downloading the Google API's Platform 2.2 and API level 8. Keep in mind a common mistake, as pointed out in the Readme file provided by Buzztouch with your Android source code : "A common mistake is to assume Android 2.2 API's and Google 2.2 API's are the same, they are not." This is important to keep in mind…if you grab the wrong ones, you won't be able to get your program to work.



SDK Path: C:\Program Files\Android\android-sdk

Packages

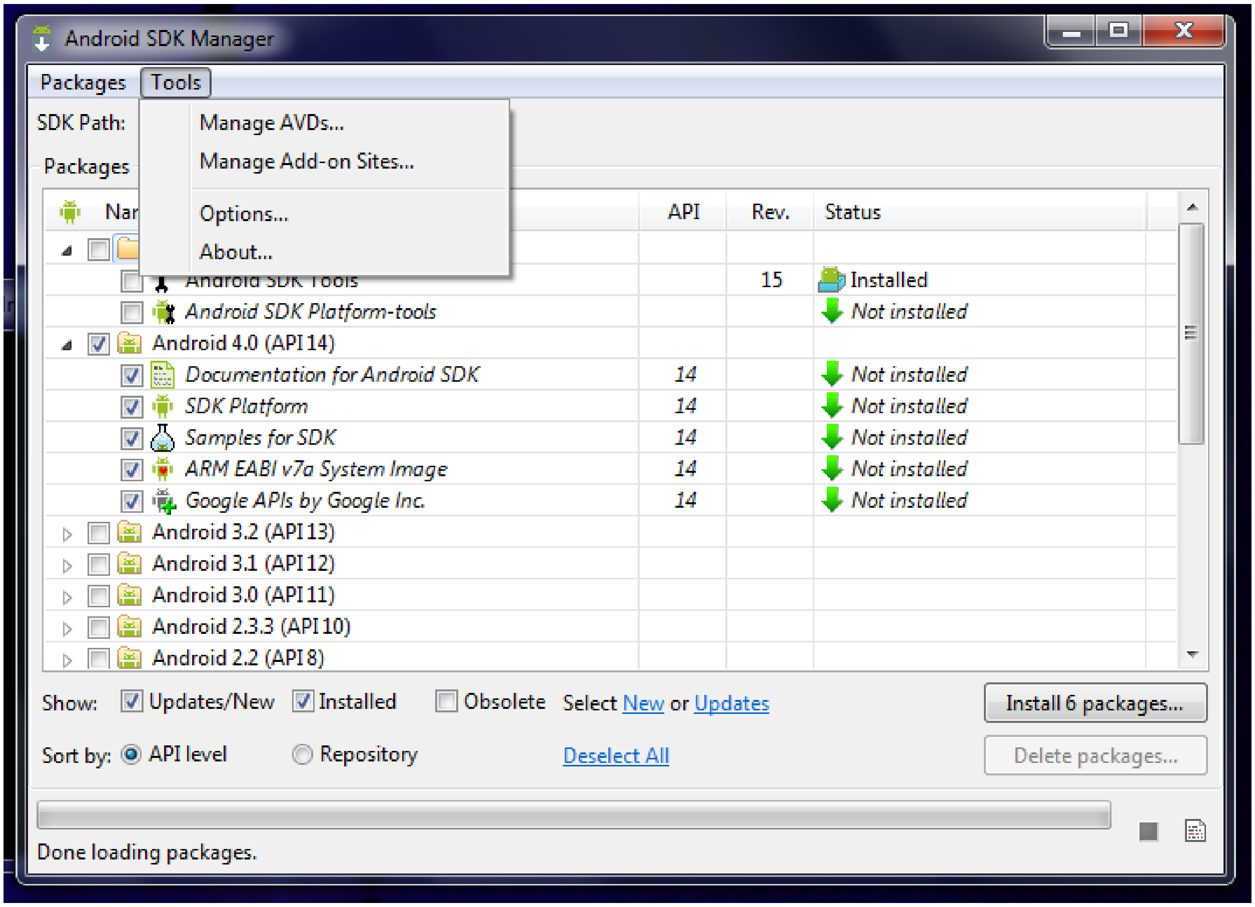
Name	API	Rev.	Status
□ Tools			
□ Android SDK Tools		15	Installed
□ *Android SDK Platform-tools*			*Not installed*
☑ Android 4.0 (API 14)			
☑ *Documentation for Android SDK*	*14*		*Not installed*
☑ *SDK Platform*	*14*		*Not installed*
☑ *Samples for SDK*	*14*		*Not installed*
☑ *ARM EABI v7a System Image*	*14*		*Not installed*
☑ *Google APIs by Google Inc.*	*14*		*Not installed*
□ Android 3.2 (API 13)			
□ Android 3.1 (API 12)			
□ Android 3.0 (API 11)			
□ Android 2.3.3 (API 10)			
□ Android 2.2 (API 8)			

Show: ☑ Updates/New ☑ Installed □ Obsolete Select New or Updates

Sort by: ● API level ○ Repository Deselect All

Done loading packages.

This should be the first screen you see. If you select what I selected above, you'll get what you need for developing with Buzztouch. In fact, you'll get more than what you need, but sometimes it's difficult to narrow down exactly where all the necessary packages are, so just grab everything!



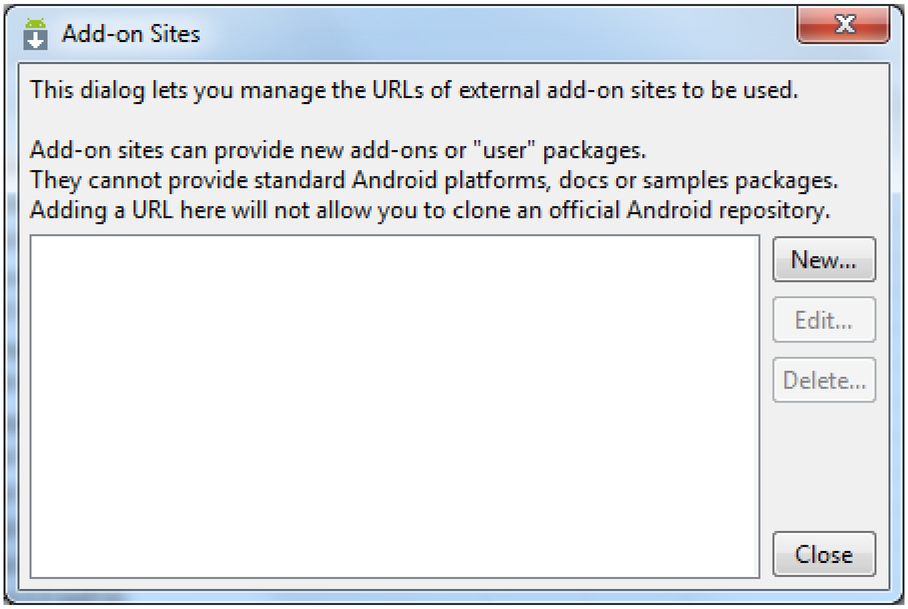
Nar	API	Rev.	Status
☐			
☐ Android SDK Tools		15	Installed
☐ Android SDK Platform-tools			Not installed
☑ Android 4.0 (API 14)			
☑ Documentation for Android SDK	14		Not installed
☑ SDK Platform	14		Not installed
☑ Samples for SDK	14		Not installed
☑ ARM EABI v7a System Image	14		Not installed
☑ Google APIs by Google Inc.	14		Not installed
☐ Android 3.2 (API 13)			
☐ Android 3.1 (API 12)			
☐ Android 3.0 (API 11)			
☐ Android 2.3.3 (API 10)			
☐ Android 2.2 (API 8)			

Show: ☑ Updates/New ☑ Installed ☐ Obsolete

Sort by: ● API level ○ Repository

Done loading packages.

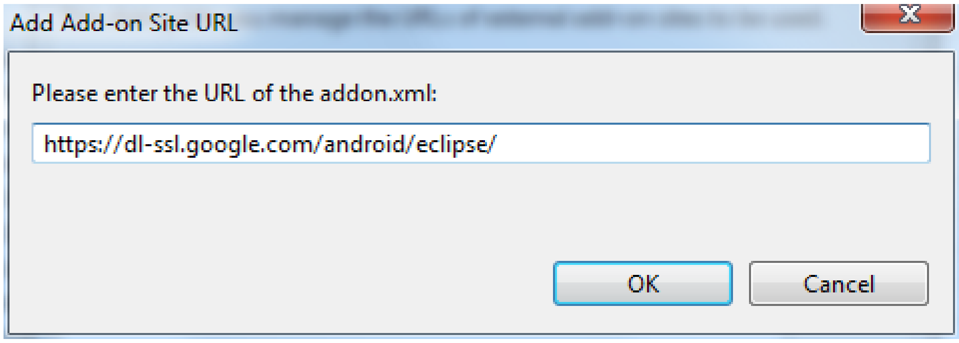
In order to download the packages, you need to define Add-on Sites. Go to Tools…Manage Add-on Sites, and you'll get the following:



This dialog lets you manage the URLs of external add-on sites to be used.

Add-on sites can provide new add-ons or "user" packages.
They cannot provide standard Android platforms, docs or samples packages.
Adding a URL here will not allow you to clone an official Android repository.

Select the "New" button and you'll be asked for a URL. Use the following:

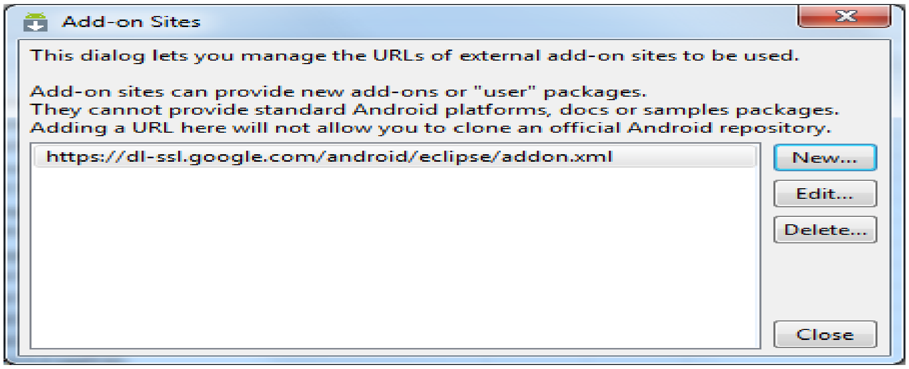


Add Add-on Site URL

Please enter the URL of the addon.xml:

https://dl-ssl.google.com/android/eclipse/

Hit OK, and you’l l see the following screen:



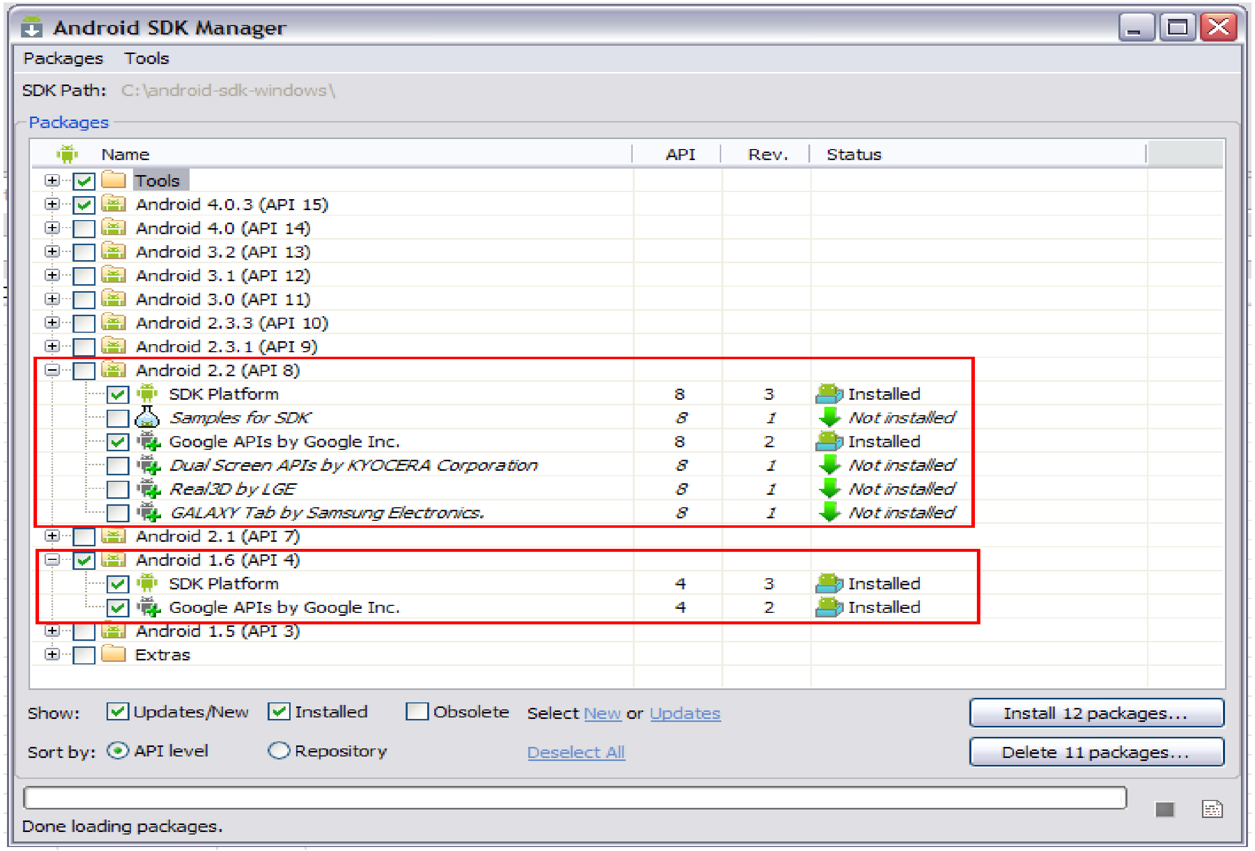
Add-on Sites

This dialog lets you manage the URLs of external add-on sites to be used.

Add-on sites can provide new add-ons or "user" packages.
They cannot provide standard Android platforms, docs or samples packages.
Adding a URL here will not allow you to clone an official Android repository.

https://dl-ssl.google.com/android/eclipse/addon.xml

Now you can hit Close, and get this screen:



Android SDK Manager

SDK Path: C:\android-sdk-windows\

Packages

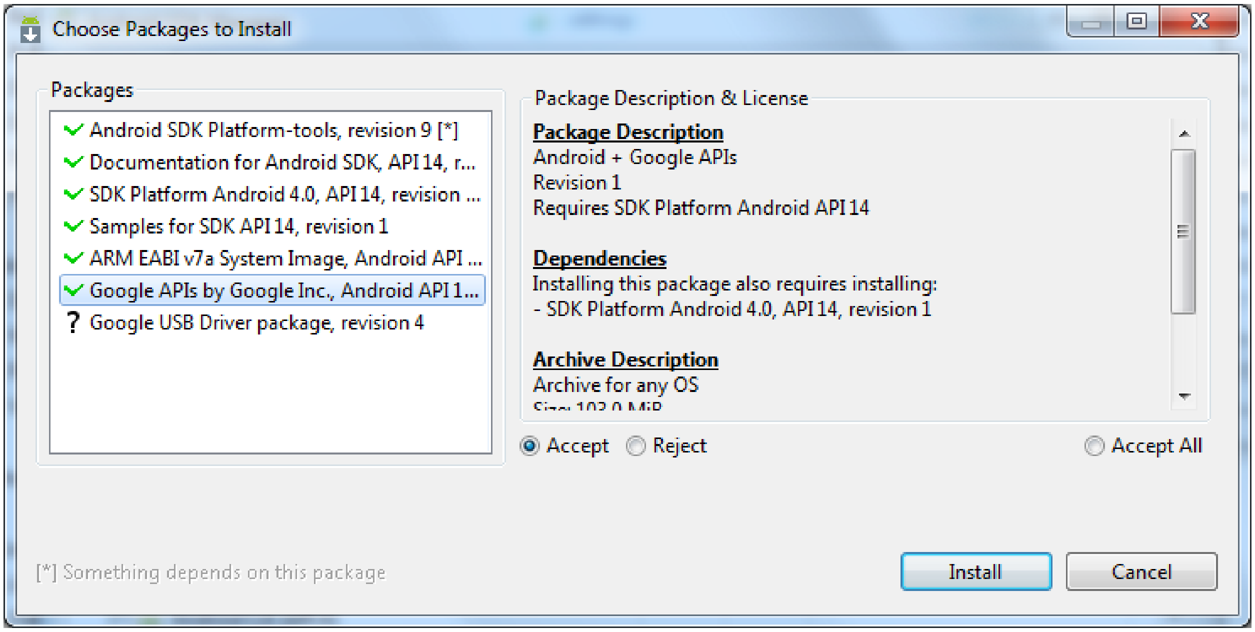
Name	API	Rev.	Status
☑ Tools			
☑ Android 4.0.3 (API 15)			
☐ Android 4.0 (API 14)			
☐ Android 3.2 (API 13)			
☐ Android 3.1 (API 12)			
☐ Android 3.0 (API 11)			
☐ Android 2.3.3 (API 10)			
☐ Android 2.3.1 (API 9)			
☐ Android 2.2 (API 8)			
☑ SDK Platform	8	3	Installed
☐ *Samples for SDK*	*8*	*1*	*Not installed*
☑ Google APIs by Google Inc.	8	2	Installed
☐ *Dual Screen APIs by KYOCERA Corporation*	*8*	*1*	*Not installed*
☐ *Real3D by LGE*	*8*	*1*	*Not installed*
☐ *GALAXY Tab by Samsung Electronics.*	*8*	*1*	*Not installed*
☐ Android 2.1 (API 7)			
☑ Android 1.6 (API 4)			
☑ SDK Platform	4	3	Installed
☑ Google APIs by Google Inc.	4	2	Installed
☐ Android 1.5 (API 3)			
☐ Extras			

Show: ☑ Updates/New ☑ Installed ☐ Obsolete Select New or Updates

Sort by: ● API level ○ Repository Deselect All

Done loading packages.

Now you can hit the **Install X Packages** button, and the following screen should appear:



Choose Packages to Install

Packages

- ✓ Android SDK Platform-tools, revision 9 [*]
- ✓ Documentation for Android SDK, API 14, r...
- ✓ SDK Platform Android 4.0, API 14, revision ...
- ✓ Samples for SDK API 14, revision 1
- ✓ ARM EABI v7a System Image, Android API ...
- ✓ Google APIs by Google Inc., Android API 1...
- ? Google USB Driver package, revision 4

Package Description & License

Package Description
Android + Google APIs
Revision 1
Requires SDK Platform Android API 14

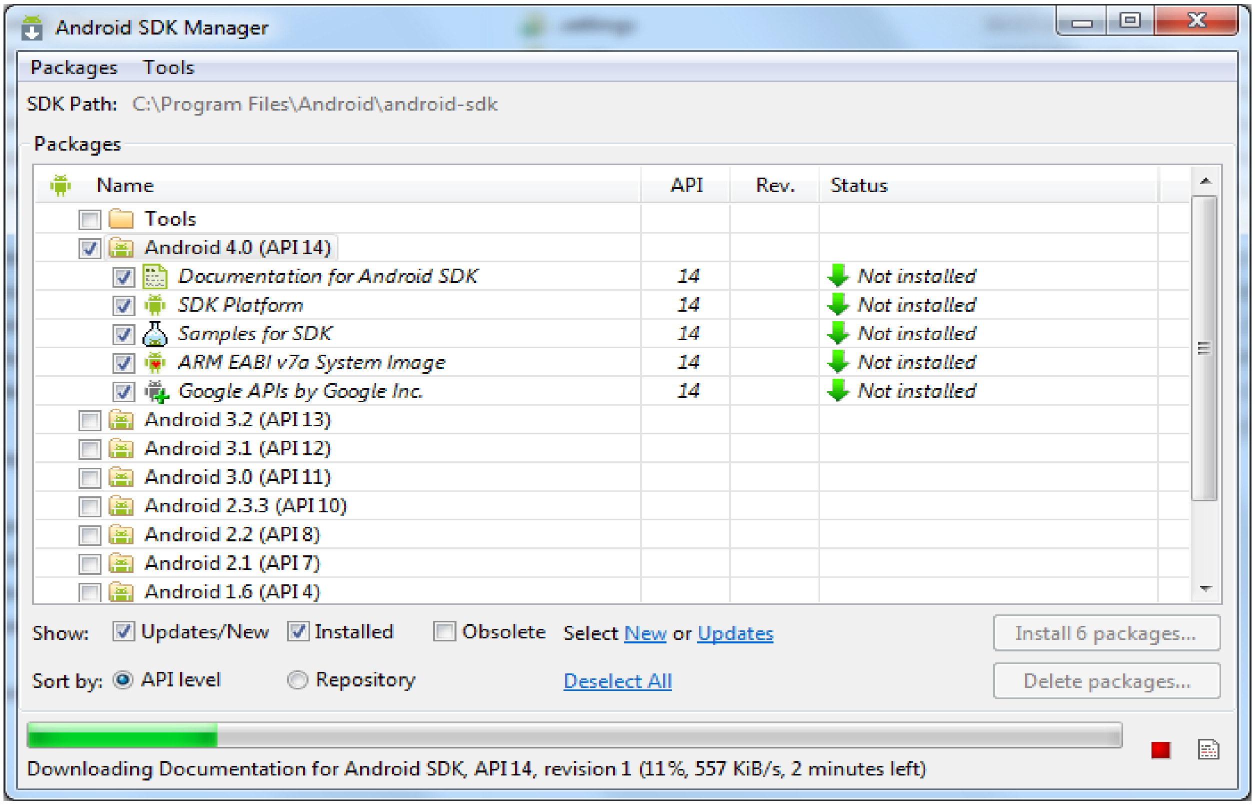
Dependencies
Installing this package also requires installing:
- SDK Platform Android 4.0, API 14, revision 1

Archive Description
Archive for any OS
Size: 103.0 MiB

● Accept ○ Reject ○ Accept All

[*] Something depends on this package

This is one of those better to grab everything moments. Make sure there is a green checkmark next to everything you see above, and then hit Install. Now grab a cub of coffee or SoBe…this is gonna take a bit.



Android SDK Manager

SDK Path: C:\Program Files\Android\android-sdk

Packages

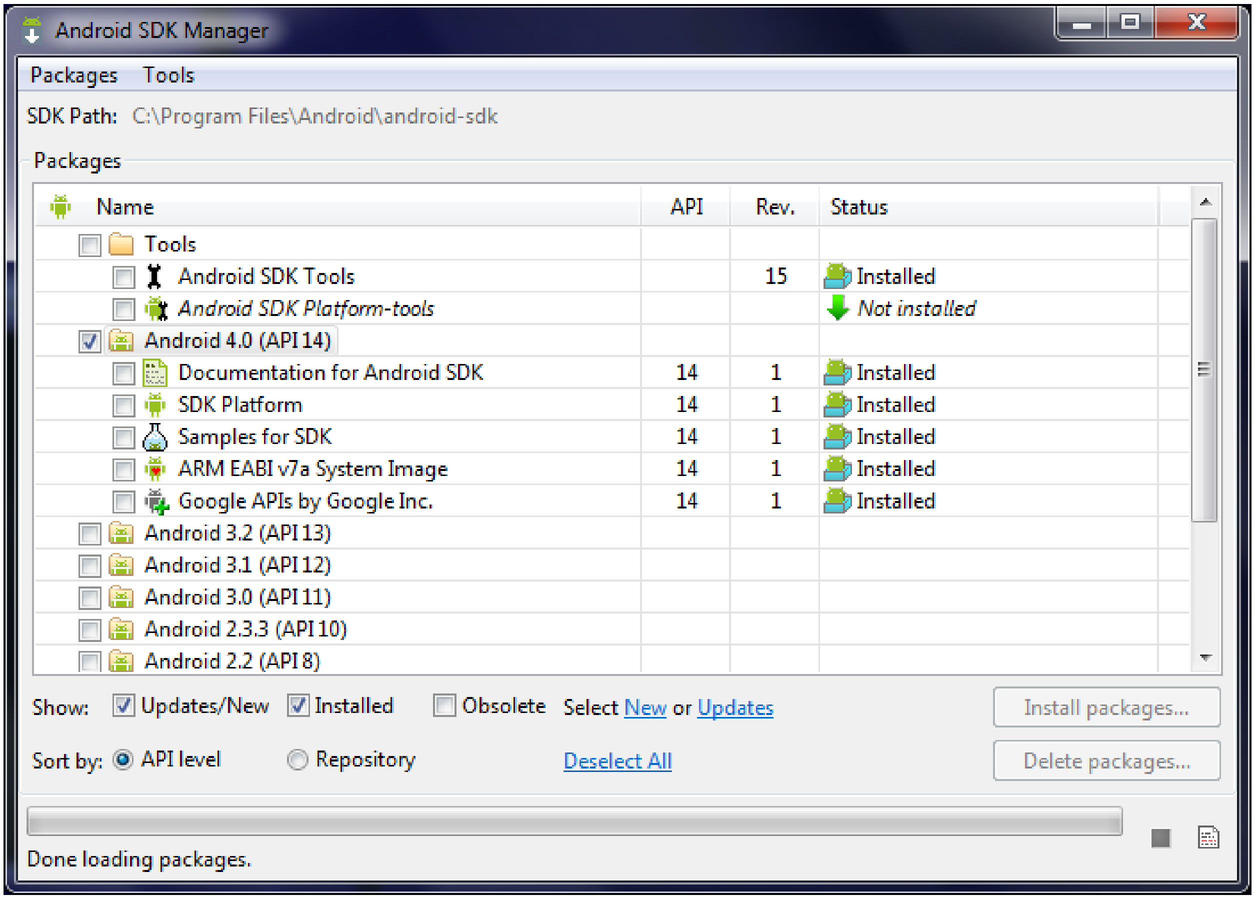
Name	API	Rev.	Status
□ Tools			
☑ Android 4.0 (API 14)			
☑ *Documentation for Android SDK*	*14*		*Not installed*
☑ *SDK Platform*	*14*		*Not installed*
☑ *Samples for SDK*	*14*		*Not installed*
☑ *ARM EABI v7a System Image*	*14*		*Not installed*
☑ *Google APIs by Google Inc.*	*14*		*Not installed*
□ Android 3.2 (API 13)			
□ Android 3.1 (API 12)			
□ Android 3.0 (API 11)			
□ Android 2.3.3 (API 10)			
□ Android 2.2 (API 8)			
□ Android 2.1 (API 7)			
□ Android 1.6 (API 4)			

Show: ☑ Updates/New ☑ Installed □ Obsolete Select New or Updates

Sort by: ● API level ○ Repository Deselect All

Downloading Documentation for Android SDK, API 14, revision 1 (11%, 557 KiB/s, 2 minutes left)

It’s now downloading and installing the stuff you need. Depending on your connection speed and system speed, this could literally take 30 minutes or more. I left mine and came back later. When it’s done, you’ll get this screen:



SDK Path: C:\Program Files\Android\android-sdk

Packages

Name	API	Rev.	Status
□ Tools			
□ Android SDK Tools		15	Installed
□ *Android SDK Platform-tools*			*Not installed*
☑ Android 4.0 (API 14)			
□ Documentation for Android SDK	14	1	Installed
□ SDK Platform	14	1	Installed
□ Samples for SDK	14	1	Installed
□ ARM EABI v7a System Image	14	1	Installed
□ Google APIs by Google Inc.	14	1	Installed
□ Android 3.2 (API 13)			
□ Android 3.1 (API 12)			
□ Android 3.0 (API 11)			
□ Android 2.3.3 (API 10)			
□ Android 2.2 (API 8)			

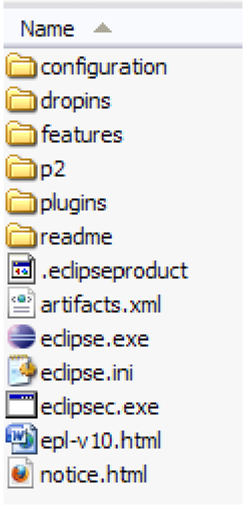
Show: ☑ Updates/New ☑ Installed □ Obsolete

Sort by: ● API level ○ Repository

Done loading packages.

Success! Now time to install Eclipse!

5. **Install Eclipse.** This is the easiest thing you’ll do today. Simply unzip the package you downloaded into the folder of your choice. I recommend **c:\eclipse.** Your folder structure should now basically look like what you see below. The **eclipse.exe** file is what you launch to start the program. Feel free to create a shortcut on your desktop, or pin it to your Start Menu (which I did). You’ll want to run this program as Administrator.



Name

configuration
dropins
features
p2
plugins
readme
.eclipseproduct
artifacts.xml
eclipse.exe
eclipse.ini
eclipsec.exe
epl-v10.html
notice.html

6. **Set up your Android keystores (Debug and Release).** In order to get your application installed on a device, it must be digitally signed with a certificate who's private key is held with the developer. If you are testing your app on a device, you can use a Debug Key. If you want to sell your app, or put it on a marketplace somewhere, you'll need to use a Release Key.

   a. Debug Key : When you install the Android SDK, a default debug keystore/key is created with predetermined names/passwords:

      i. Keystore name : "debug.keystore"

      ii. Keystore password : "android"

      iii. Key alias : "androiddebugkey"

      iv. Key password : "android"

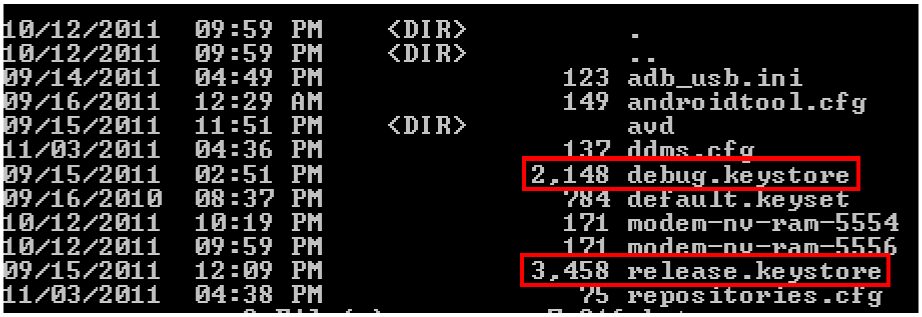
      v. CN : "CN=Android Debug, O=Android, C=US"

   On my Windows system, the keystore is located at C:\Documents and Settings\mf\.android. You'll need this path to configure Eclipse when you're ready to create an APK package of your app.

   To set up a Release Key, it's best to follow the instructions that can be found here :

   http://developer.android.com/guide/publishing/app-signing.html#setup

   Once you have done this, you should see two keystores in your folder :

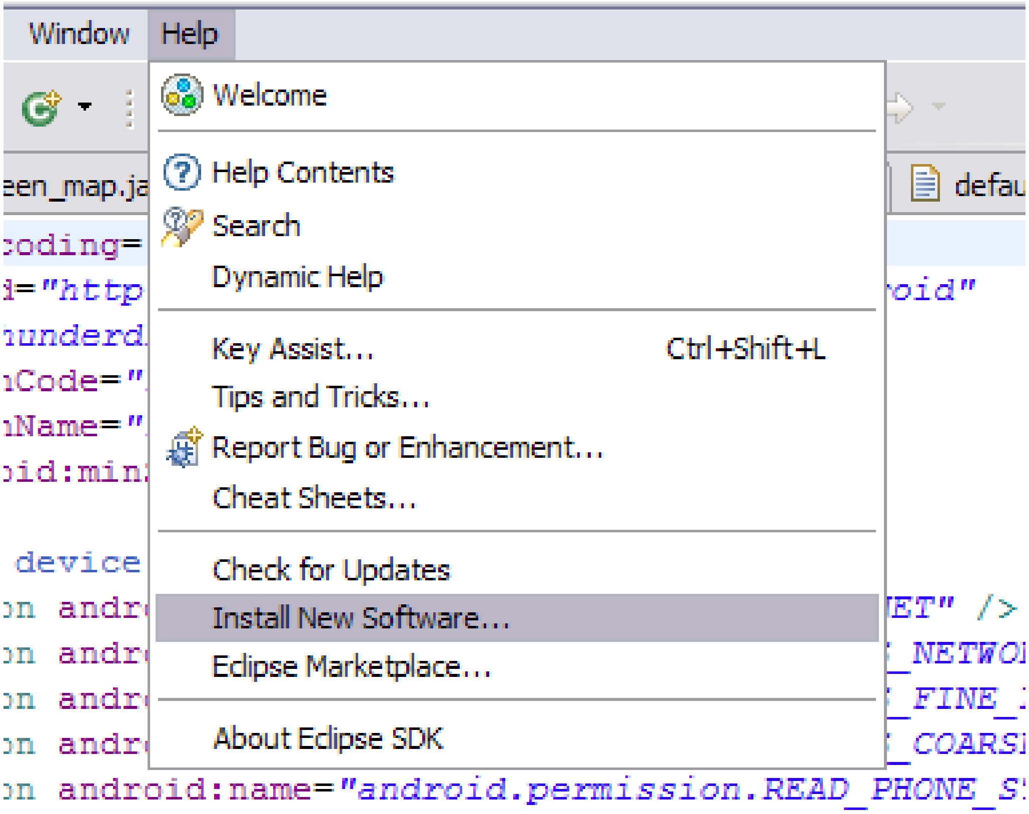


```
10/12/2011  09:59 PM    <DIR>          .
10/12/2011  09:59 PM    <DIR>          ..
09/14/2011  04:49 PM               123 adb_usb.ini
09/16/2011  12:29 AM               149 androidtool.cfg
09/15/2011  11:51 PM    <DIR>          avd
11/03/2011  04:36 PM               137 ddms.cfg
09/15/2011  02:51 PM             2,148 debug.keystore
09/16/2010  08:37 PM               784 default.keyset
10/12/2011  10:19 PM               171 modem-nv-ram-5554
10/12/2011  09:59 PM               171 modem-nv-ram-5556
09/15/2011  12:09 PM             3,458 release.keystore
11/03/2011  04:38 PM                75 repositories.cfg
```

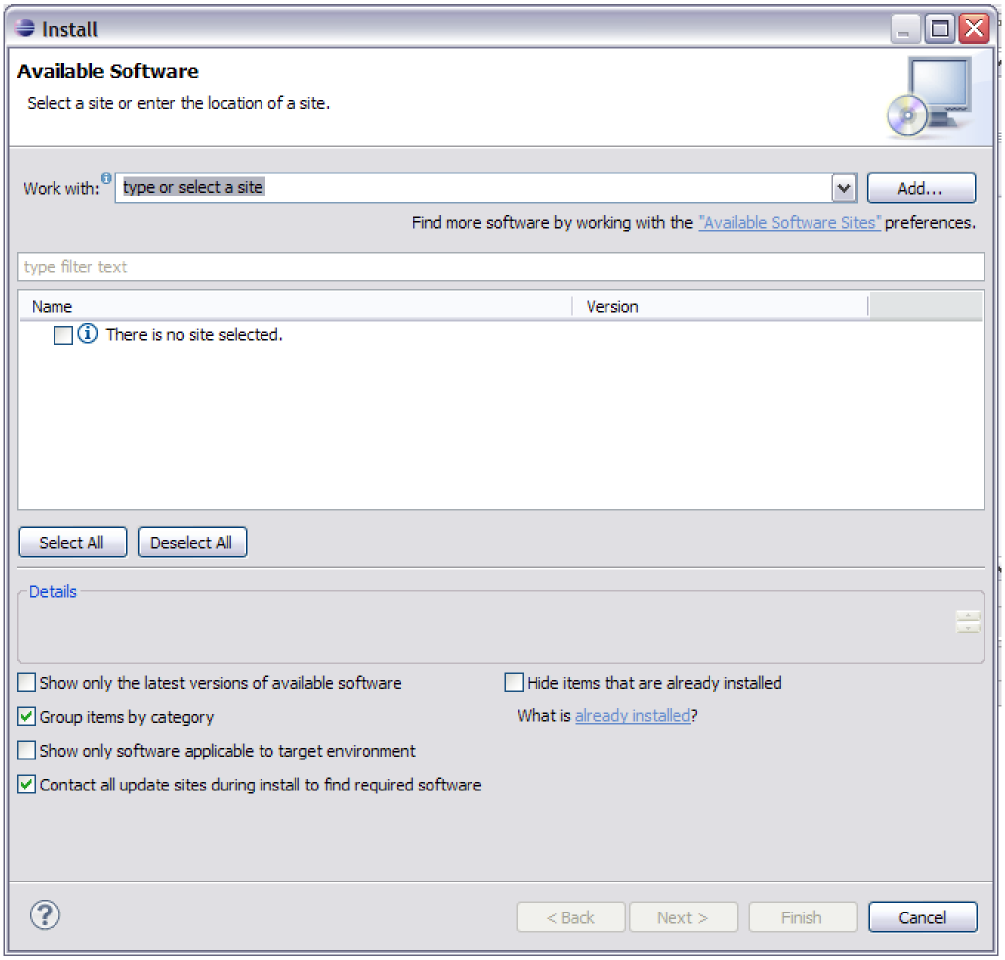
You'll reference these later when you package up your application. But it's best to get it taken care of now, so you don't get surprised later!

7. **Install Eclipse Android Development Tools (ADT).** We now need to install the Android Development Tools for Eclipse. This is a pretty simple process, and doesn't take too long. It'll give you the extra few things you need for developing Android applications (remember that Eclipse is an IDE capable of developing many, many things).

   Start Eclipse by double-clicking on the Eclipse icon you created earlier. After Eclipse has launched go to **Help...Install New Software:**



This will bring up the following install dialogue box. You'll see that there is no site selected, and thus no option on which software to install.



Available Software

Select a site or enter the location of a site.

Work with: type or select a site

Find more software by working with the "Available Software Sites" preferences.

type filter text

□ There is no site selected.

□ Show only the latest versions of available software

☑ Group items by category

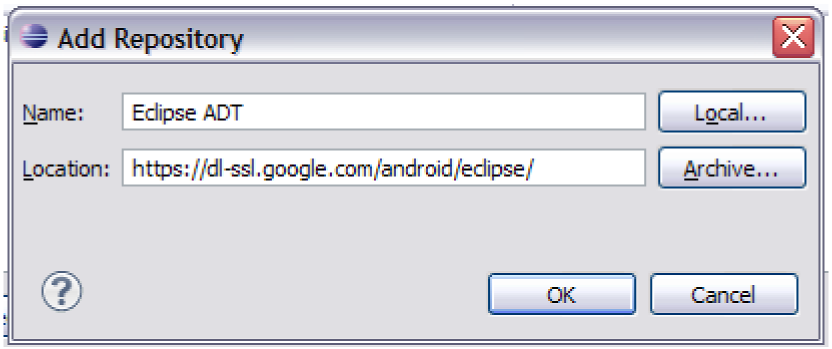
□ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

□ Hide items that are already installed

What is already installed?

Click the **Add…** button and the you'll get this:

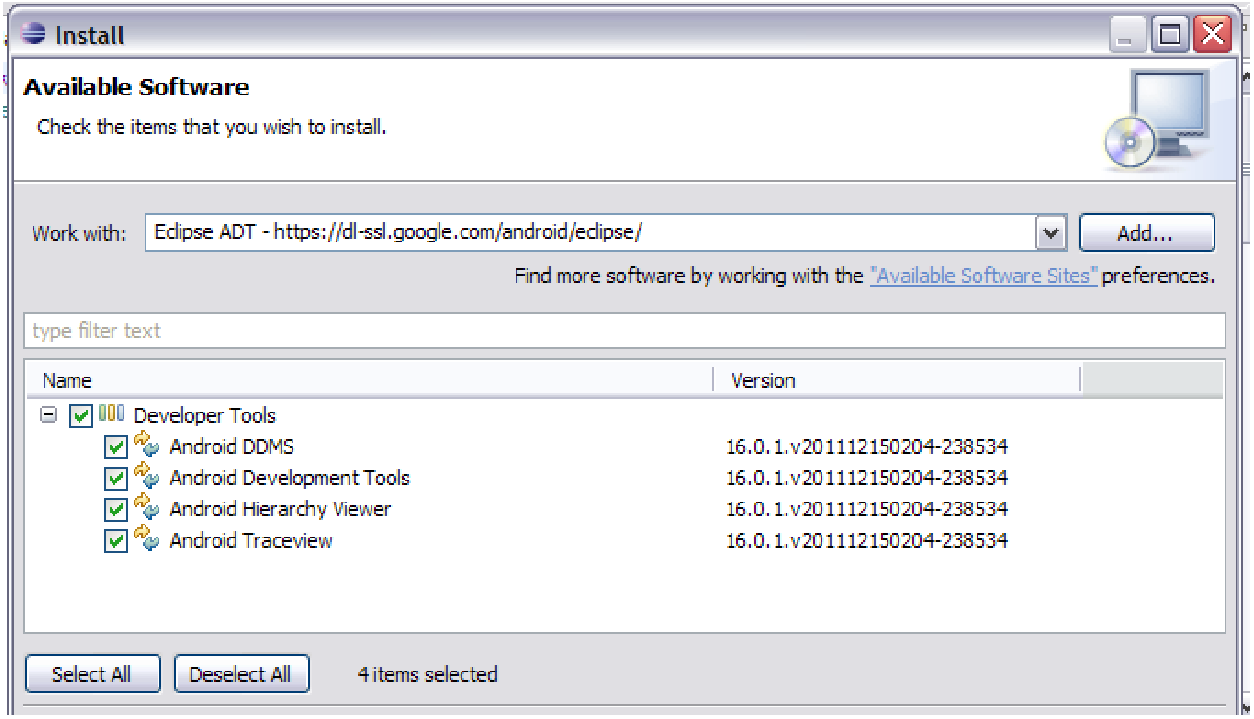


Name: Eclipse ADT

Location: https://dl-ssl.google.com/android/eclipse/

Specify the same information you see above. For clarity, the link you want is:

https://dl-ssl.google.com/android/eclipse. You can give it whatever name you want. Hit **OK**. Once you've done that, you'll get the following screen:



Available Software

Check the items that you wish to install.

Work with: Eclipse ADT - https://dl-ssl.google.com/android/eclipse/

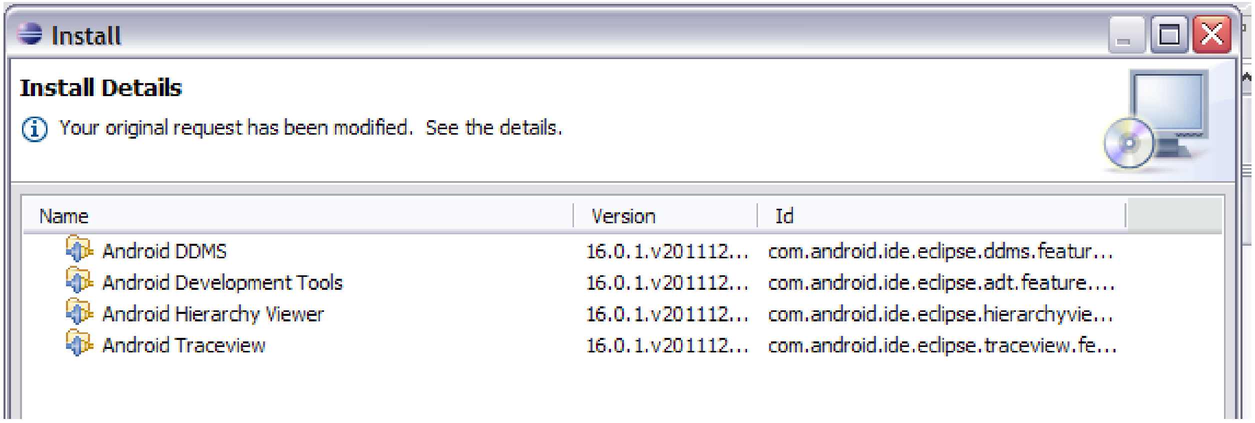
Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
⊟ ☑ Developer Tools	
☑ Android DDMS	16.0.1.v201112150204-238534
☑ Android Development Tools	16.0.1.v201112150204-238534
☑ Android Hierarchy Viewer	16.0.1.v201112150204-238534
☑ Android Traceview	16.0.1.v201112150204-238534

4 items selected

Well, that's half the screen, but you get the picture! These are your options for install, and it includes the Android Development Tools (ADT). I went ahead and selected everything figuring I may need some of the other stuff in the future. Hit **Next**.

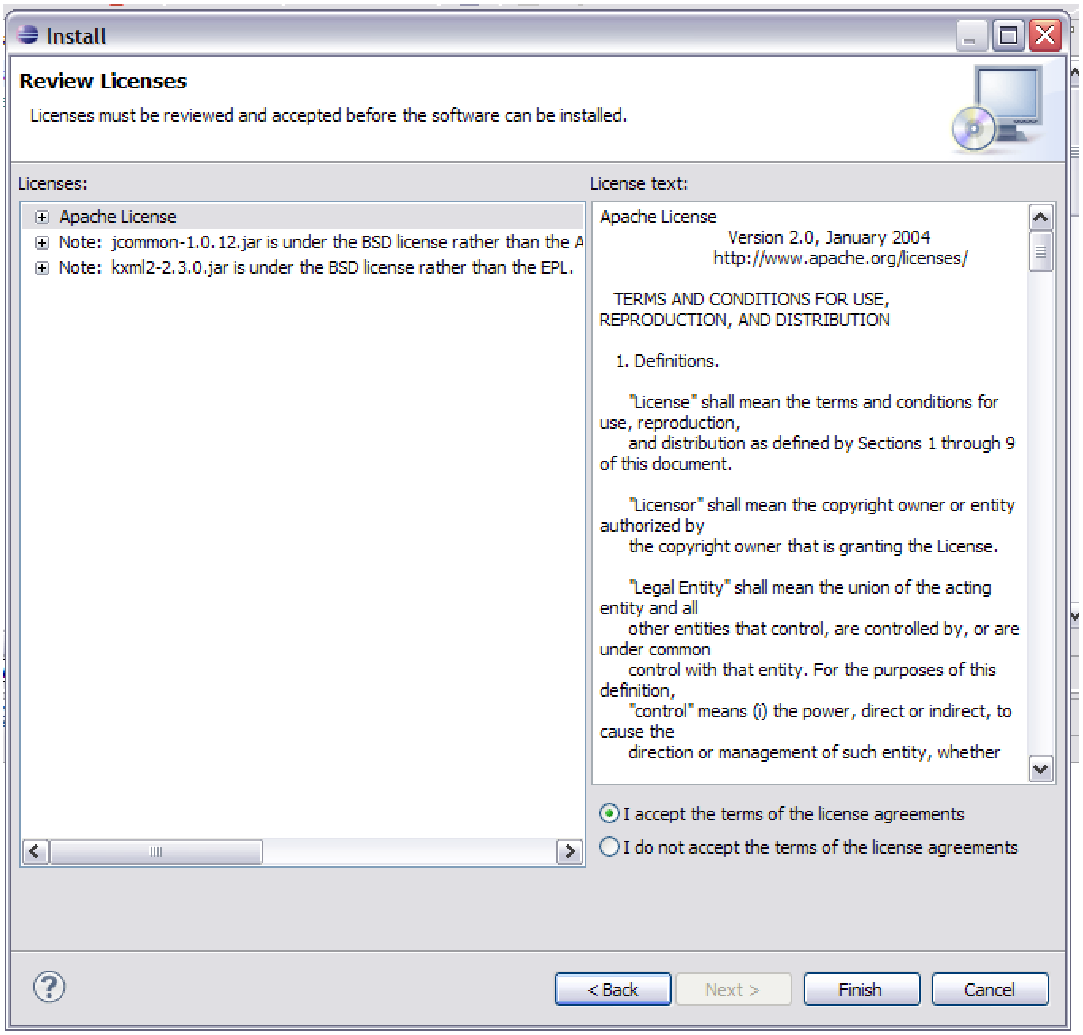


Install Details

Your original request has been modified. See the details.

Name	Version	Id
Android DDMS	16.0.1.v201112...	com.android.ide.eclipse.ddms.featur...
Android Development Tools	16.0.1.v201112...	com.android.ide.eclipse.adt.feature....
Android Hierarchy Viewer	16.0.1.v201112...	com.android.ide.eclipse.hierarchyvie...
Android Traceview	16.0.1.v201112...	com.android.ide.eclipse.traceview.fe...

This screen verifies the details of what you are going to install. Hit **Next**.



Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

Licenses:

- Apache License
- Note: jcommon-1.0.12.jar is under the BSD license rather than the A
- Note: kxml2-2.3.0.jar is under the BSD license rather than the EPL.

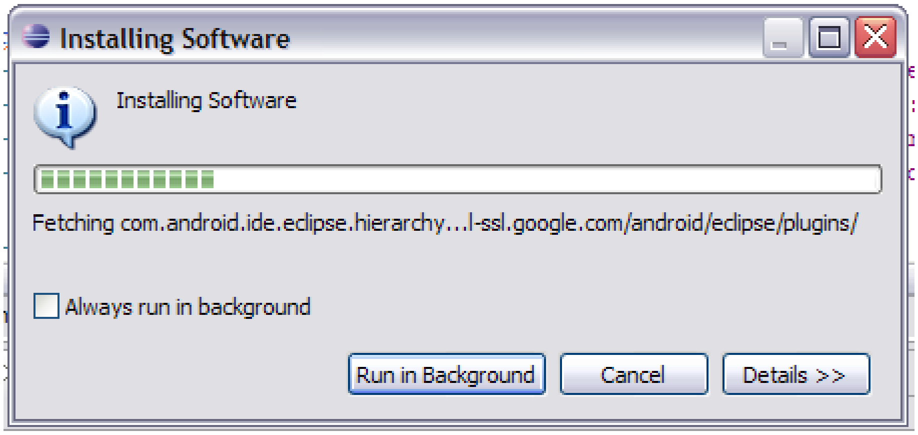
License text:

```
Apache License
                    Version 2.0, January 2004
                 http://www.apache.org/licenses/

  TERMS AND CONDITIONS FOR USE,
REPRODUCTION, AND DISTRIBUTION

  1. Definitions.

      "License" shall mean the terms and conditions for
use, reproduction,
      and distribution as defined by Sections 1 through 9
of this document.

      "Licensor" shall mean the copyright owner or entity
authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting
entity and all
      other entities that control, are controlled by, or are
under common
      control with that entity. For the purposes of this
definition,
      "control" means (i) the power, direct or indirect, to
cause the
      direction or management of such entity, whether
```

● I accept the terms of the license agreements

○ I do not accept the terms of the license agreements

If you feel the need, review the licenses for the stuff you are going to install here. Most important point on this screen is to select **I accept the terms of the license agreements**. If you don't, you won't be able to install. Hit **Next**.

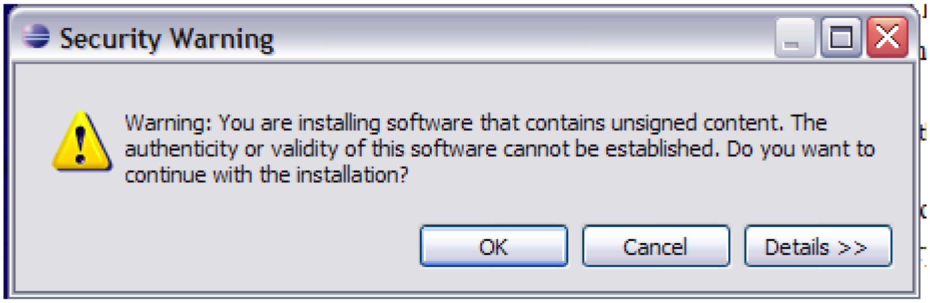


Installing Software

Fetching com.android.ide.eclipse.hierarchy...l-ssl.google.com/android/eclipse/plugins/

□ Always run in background

Congratulations! ADT is now installing! Sit back and relax…at least until you see this screen:



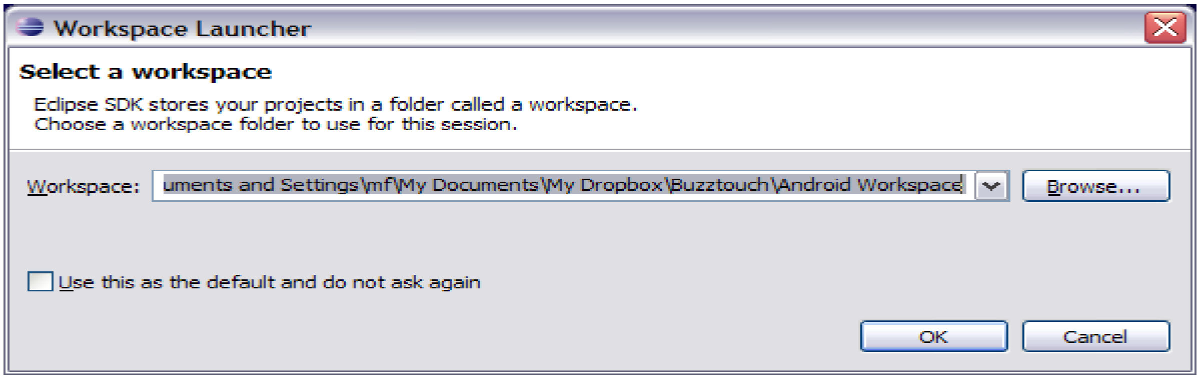
Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?

Nothing to be worried about here. Just hit **OK** and move on. It's all good. After install is done, the program will ask if you would like to reset Eclipse so the changes can take effect. Say **Reset Eclipse**, wait for it to restart, and you can move on to the next step

8. **Configure Eclipse.** We're just about done setting things up! Crazy how long this takes! Start by clicking on the Eclipse icon you created earlier, or by double-clicking the eclipse.exe file in your c:\eclipse folder. You should see a splash screen similar to the following :

(c) Copyright Eclipse contributors and others, 2000, 2011. All rights reserved. Eclipse is a trademark of the Eclipse Foundation,Inc. Oracle and Java are registered trademarks of Oracle and/or itsaffiliates.Other names may be trademarks of their respective owners.

This means the program is loading, but it does so very slowly. Must be related to all that Java! In a few minutes, you'll see the following dialogue:



Select a workspace

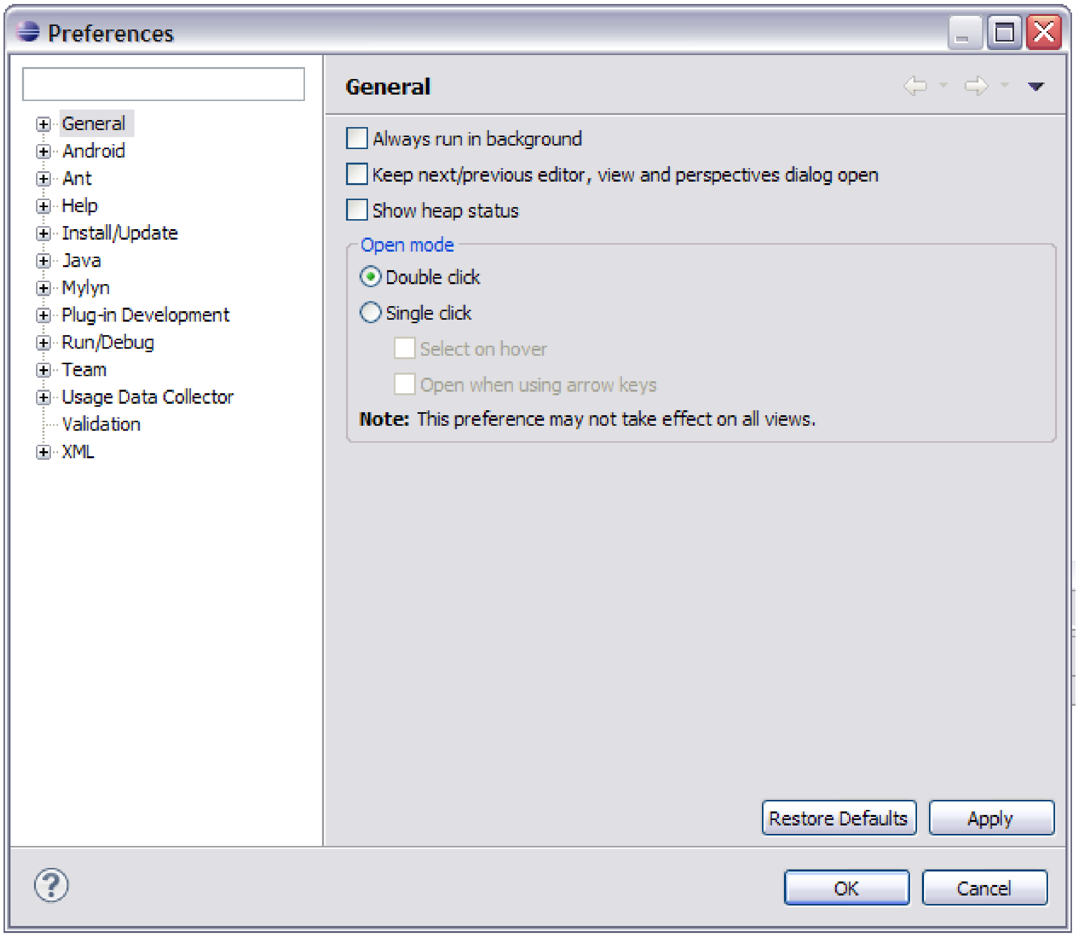
Eclipse SDK stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: uments and Settings\mf\My Documents\My Dropbox\Buzztouch\Android Workspace

☐ Use this as the default and do not ask again

You can choose whatever you want here. I use DropBox for my Workspace, so that way everything is synced between computers. I'm not entirely sure how to explain the Workspace, but you need it, and it can be anywhere.

Once the workspace has been configured, the main program will come up. You now want to go to **Windows > Preferences**, and the following screen will display:



Preferences

General

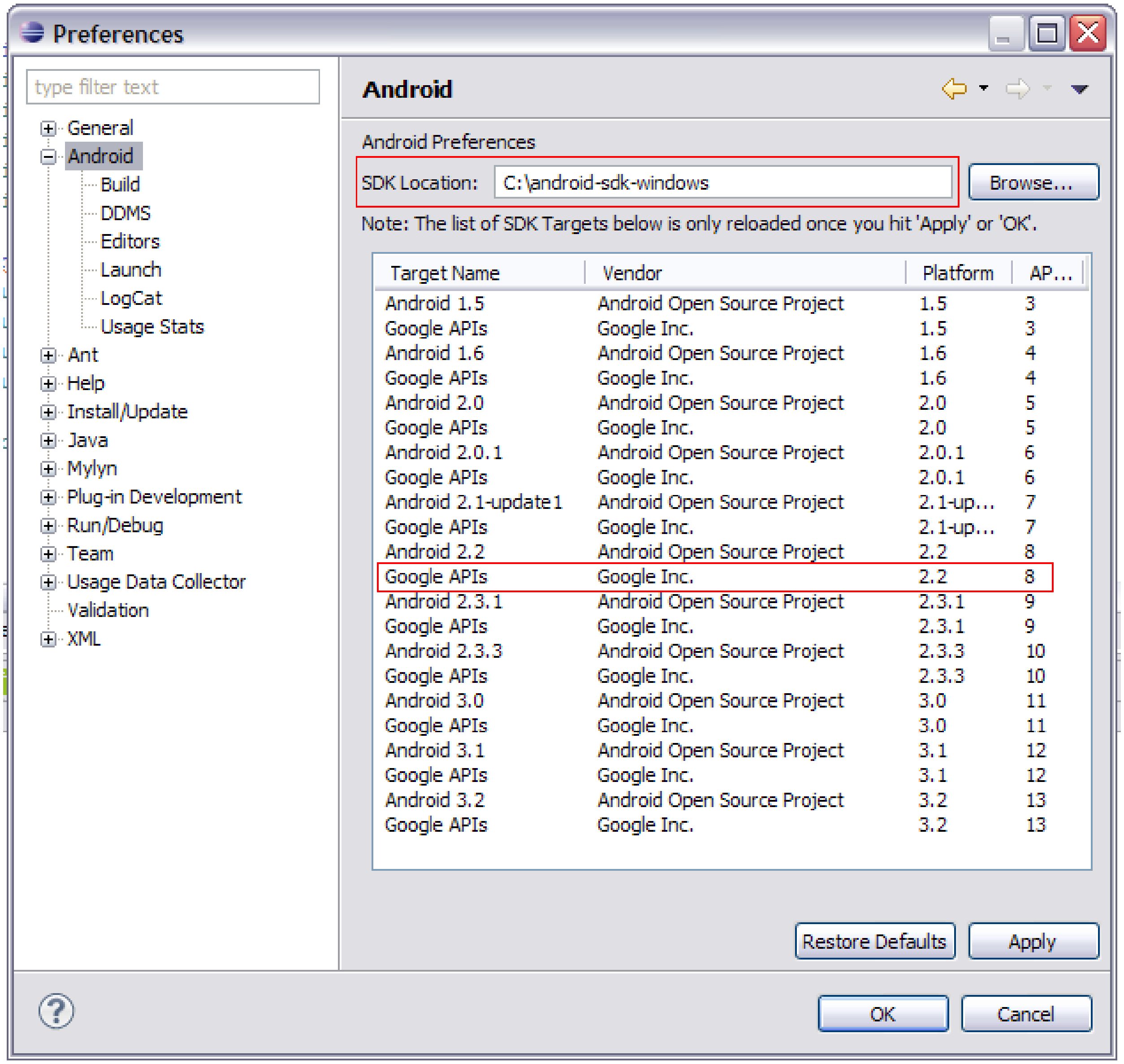
□ Always run in background
□ Keep next/previous editor, view and perspectives dialog open
□ Show heap status

Open mode

● Double click
○ Single click
□ Select on hover
□ Open when using arrow keys

Note: This preference may not take effect on all views.

Now, expand the Android section, and you should see the following screen :



Android Preferences

SDK Location: C:\android-sdk-windows

Note: The list of SDK Targets below is only reloaded once you hit 'Apply' or 'OK'.

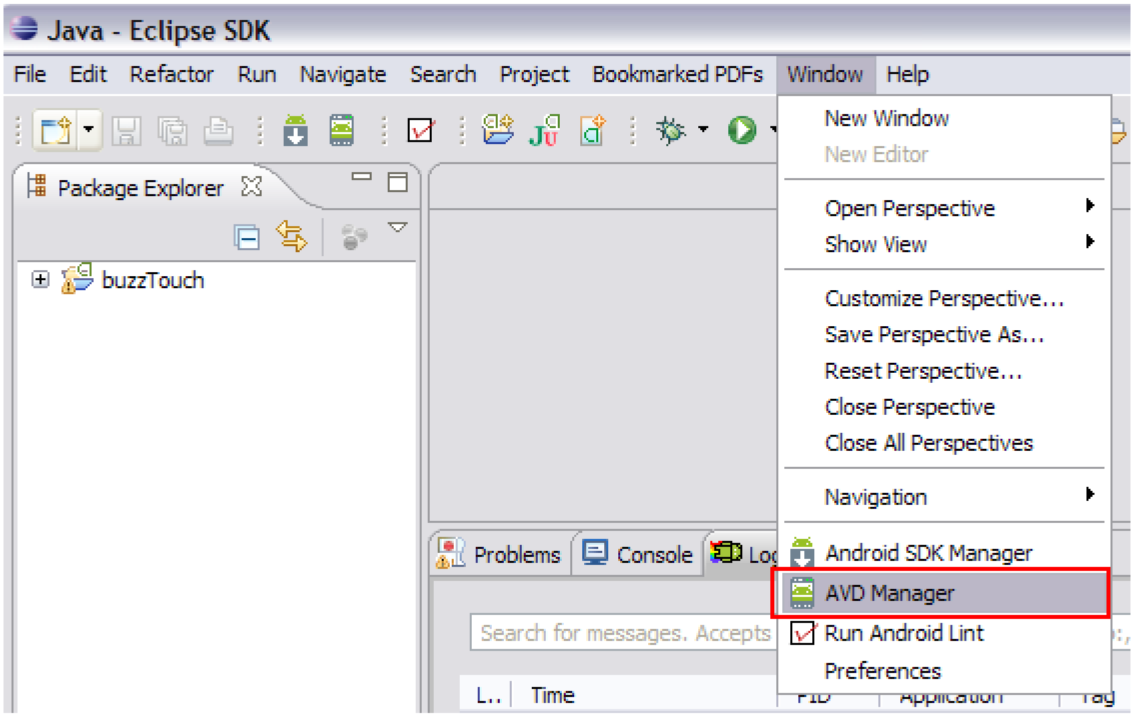
Target Name	Vendor	Platform	AP...
Android 1.5	Android Open Source Project	1.5	3
Google APIs	Google Inc.	1.5	3
Android 1.6	Android Open Source Project	1.6	4
Google APIs	Google Inc.	1.6	4
Android 2.0	Android Open Source Project	2.0	5
Google APIs	Google Inc.	2.0	5
Android 2.0.1	Android Open Source Project	2.0.1	6
Google APIs	Google Inc.	2.0.1	6
Android 2.1-update1	Android Open Source Project	2.1-up...	7
Google APIs	Google Inc.	2.1-up...	7
Android 2.2	Android Open Source Project	2.2	8
Google APIs	Google Inc.	2.2	8
Android 2.3.1	Android Open Source Project	2.3.1	9
Google APIs	Google Inc.	2.3.1	9
Android 2.3.3	Android Open Source Project	2.3.3	10
Google APIs	Google Inc.	2.3.3	10
Android 3.0	Android Open Source Project	3.0	11
Google APIs	Google Inc.	3.0	11
Android 3.1	Android Open Source Project	3.1	12
Google APIs	Google Inc.	3.1	12
Android 3.2	Android Open Source Project	3.2	13
Google APIs	Google Inc.	3.2	13

There are two things of importance here. First, you need to **specify your SDK location**, which is indicated in the top red box. Browse to the root folder for your SDK installation, and then hit OK to select. Then hit Apply, and it will populate all the SDKs you have installed. This brings us to the second important item, which is making sure you have the **Google APIs 2.2/8** installed, as shown above in the second red box. if you don't, you need to install them, as your BT v1.5 app won't work without them!

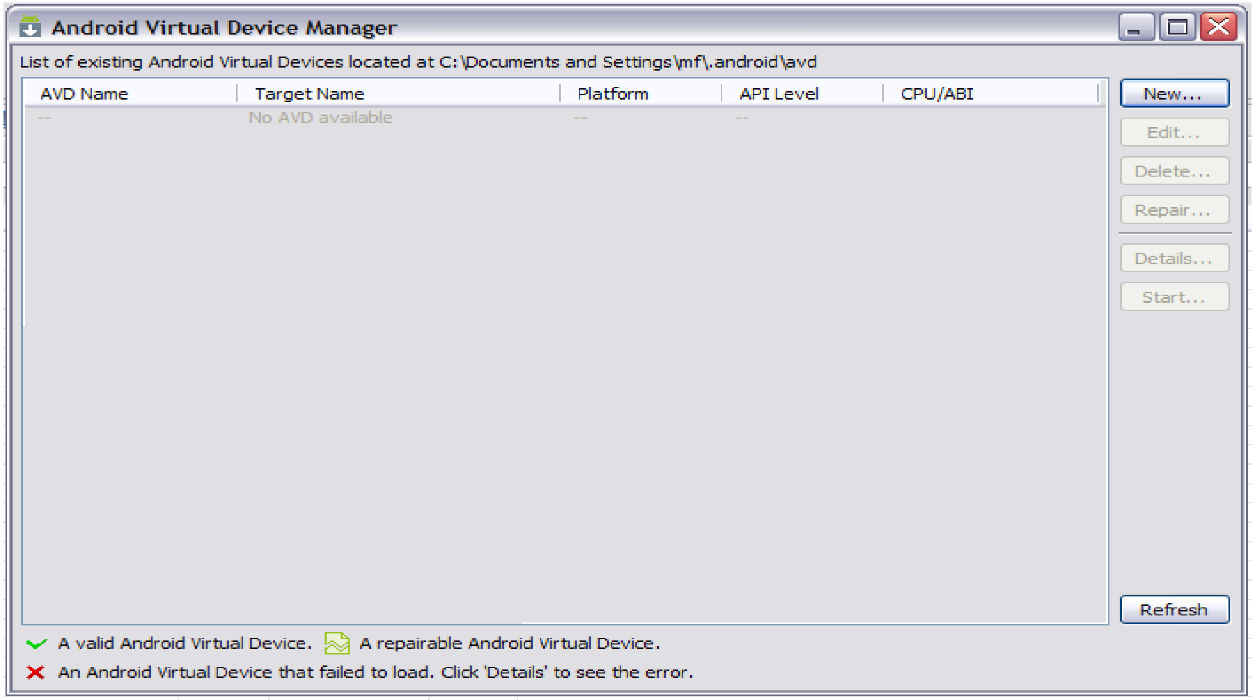
If everything looks good here, then hit OK to close out of this dialogue, and you'll be back to the main interface for Eclipse! Now it's time to move on to setting up your Android Virtual Device (AVD), or emulator, which is how you will test your applications in Eclipse before packaging them up for test on a real device, or distribution in one of the various Android markets.

9. **Configure AVD.** Configuring your Android Virtual Device (emulator) is super simple, assuming you have the correct Google API's installed.

To begin, go to **Windows…AVD Manager.**



This will bring up the AVD Manager screen, which, since you currently have no AVDs configured, will look like this:



List of existing Android Virtual Devices located at C:\Documents and Settings\mf\.android\avd

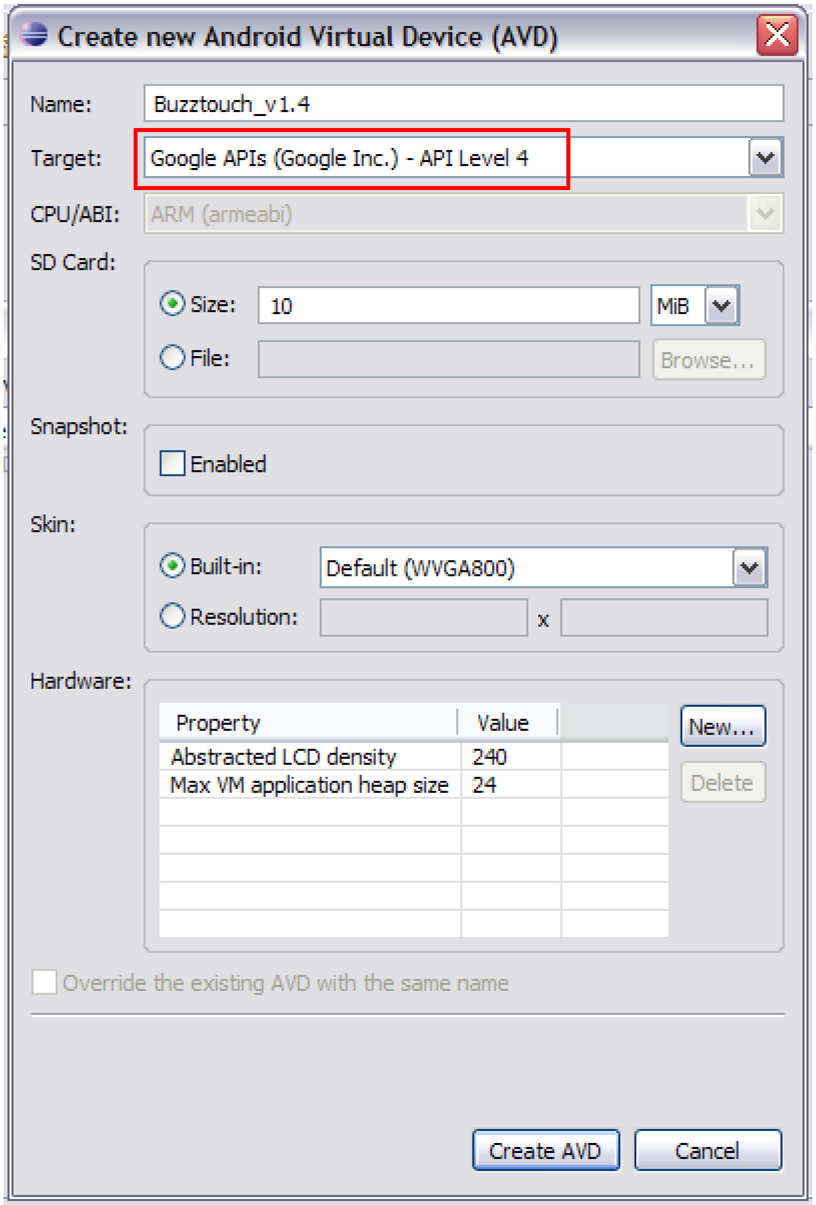
AVD Name	Target Name	Platform	API Level	CPU/ABI
--	No AVD available	--	--	

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

There are two different AVDs that you could create…one for Buzztouch v1.4, and one for Buzztouch v1.5/v2 Since they both use different Google APIs, You'll have to create one for each if you develop for both. If you are only doing one or the other, then you only need to create an AVD for that particular version of Buzztouch.

In order to create an AVD for Buzztouch v1.4, click **New…** and fill in the information like you see below:



Create new Android Virtual Device (AVD)

Name: Buzztouch_v1.4

Target: Google APIs (Google Inc.) - API Level 4

CPU/ABI: ARM (armeabi)

SD Card:
● Size: 10 MiB
○ File:

Snapshot:
□ Enabled

Skin:
● Built-in: Default (WVGA800)
○ Resolution: x

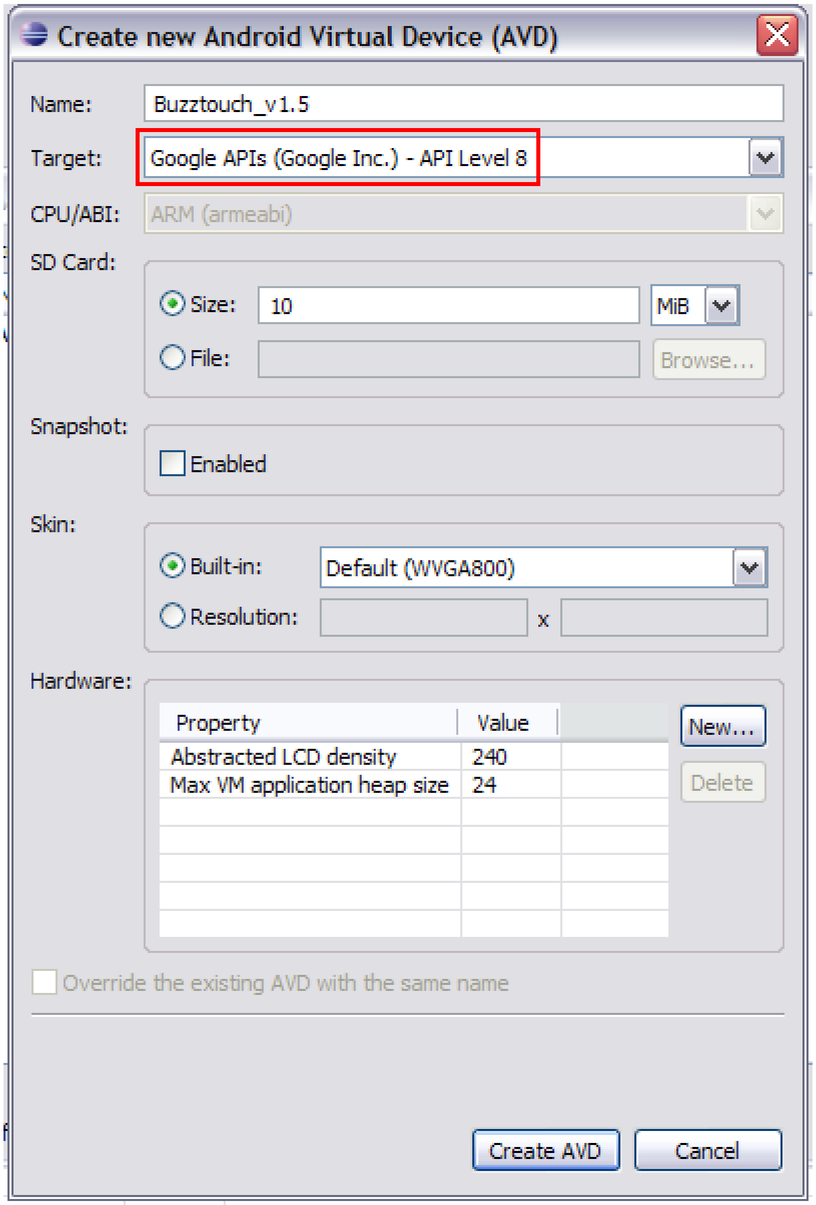
Hardware:

Property	Value
Abstracted LCD density	240
Max VM application heap size	24

□ Override the existing AVD with the same name

You can adjust the size of the SD Card to your liking, but I choose to keep it near the minimum 9MB, simply because I'm not using the SD Card feature, and because it loads faster with a smaller SD Card configured. I also leave the default Built-in skin of WVGA800, but you can change the resolution to a device size of your liking.

In order to create an AVD for Buzztouch v1.5/BT Server 2.0, click **New…** and fill in the information like you see below:



Name: Buzztouch_v1.5

Target: Google APIs (Google Inc.) - API Level 8

CPU/ABI: ARM (armeabi)

SD Card:
● Size: 10 MiB
○ File:

Snapshot:
□ Enabled

Skin:
● Built-in: Default (WVGA800)
○ Resolution: x

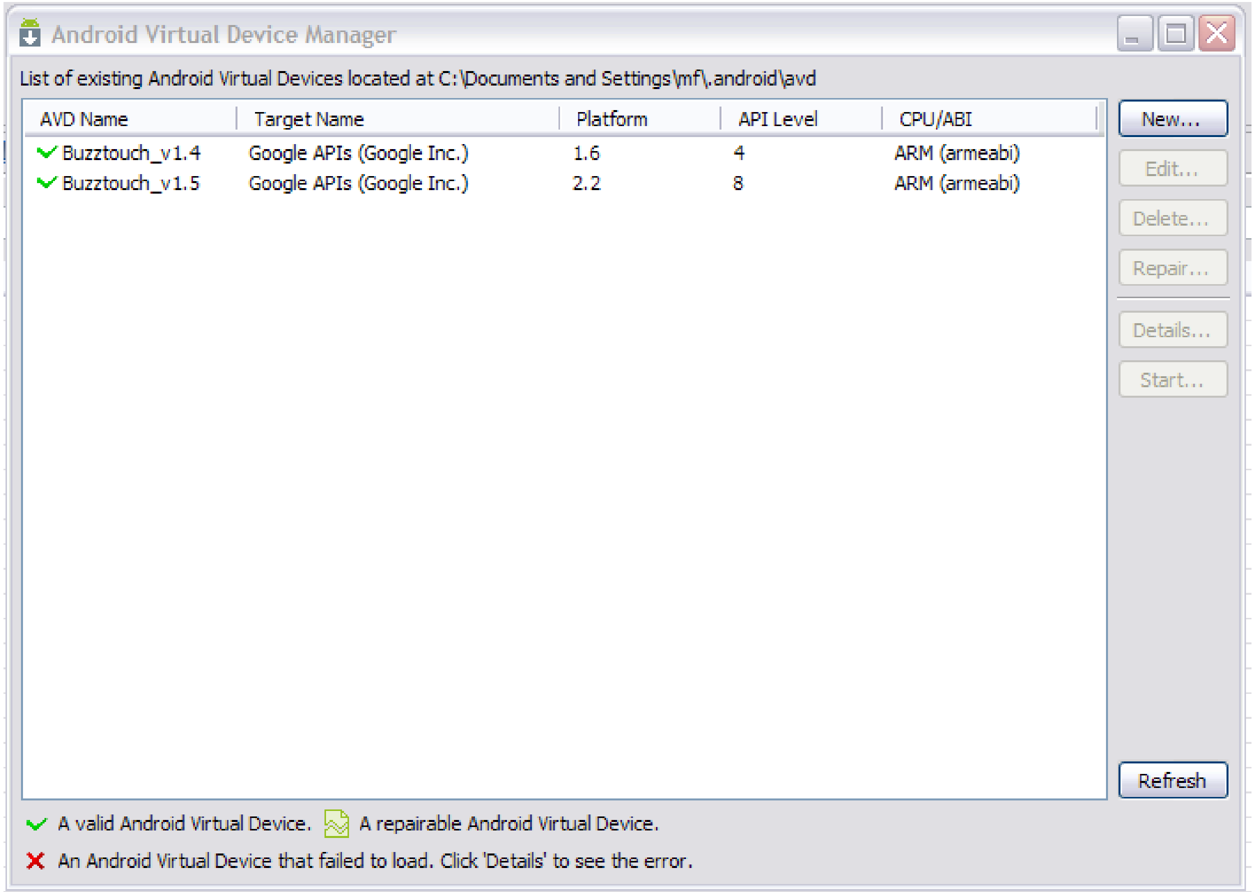
Hardware:

Property	Value
Abstracted LCD density	240
Max VM application heap size	24

□ Override the existing AVD with the same name

You can adjust the size of the SD Card to your liking, but I choose to keep it near the minimum 9MB, simply because I'm not using the SD Card feature, and because it loads faster with a smaller SD Card configured. I also leave the default Built-in skin of WVGA800, but you can change the resolution to a device size of your liking.

Now that you are done creating your virtual devices, your Android Virtual Device manager should look something like what you see below. The important thing is that the correct Google APIs are selected for the version of Buzztouch that you are using, and that there is a green checkmark next to each virtual device, indicating that it is good.



List of existing Android Virtual Devices located at C:\Documents and Settings\mf\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ Buzztouch_v1.4	Google APIs (Google Inc.)	1.6	4	ARM (armeabi)
✓ Buzztouch_v1.5	Google APIs (Google Inc.)	2.2	8	ARM (armeabi)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

At this point, you are done setting up your environment to develop Android applications using the source code you downloaded from Buzztouch! I'd love to be able to tell you that things get super easy from here, but Android development seems to have as many pitfalls as iOS development. So, I've written a few more documents that should help:

http://www.buzztouch.com/resources/Compiling_Your_Android_Application_v1.0.pdf

http://www.buzztouch.com/resources/Adding_Files_To_A_Project_v1.0.pdf

http://www.buzztouch.com/resources/Obtaining_a_Google_Maps_API_Key_v1.0.pdf

http://www.buzztouch.com/resources/Signing_Android_Apps_v1.0.pdf

http://www.buzztouch.com/resources/Problem_Reporting_and_Basic_Debug_v1.0.pdf

Hopefully between these documents, the Buzztouch forums, and Google, you should be able to get all your questions answered! And feel free to contact me directly with any questions you might have!

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

Revision Log			
v1.0	11/26/11	Initial release of document.	GoNorthWest
v1.1	11/28/11	Added Revision Log.	GoNorthWest
v1.2	12/20/11	Added ADT installation instructions.	GoNorthWest
v1.3	02/05/12	Added info about correct Google APIs, Added AVD configuration instructions.	GoNorthWest