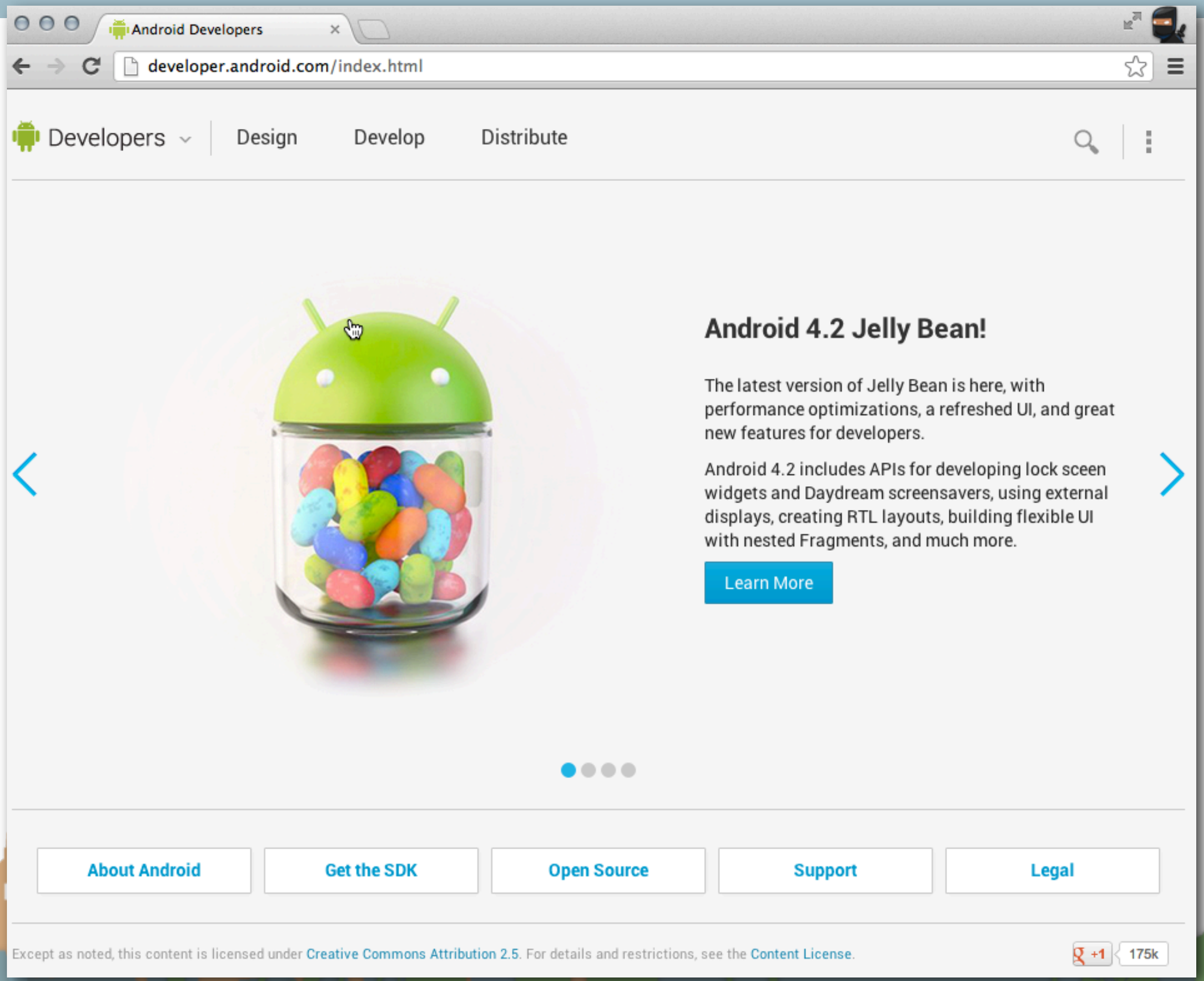
User Interaction: Intro to Android

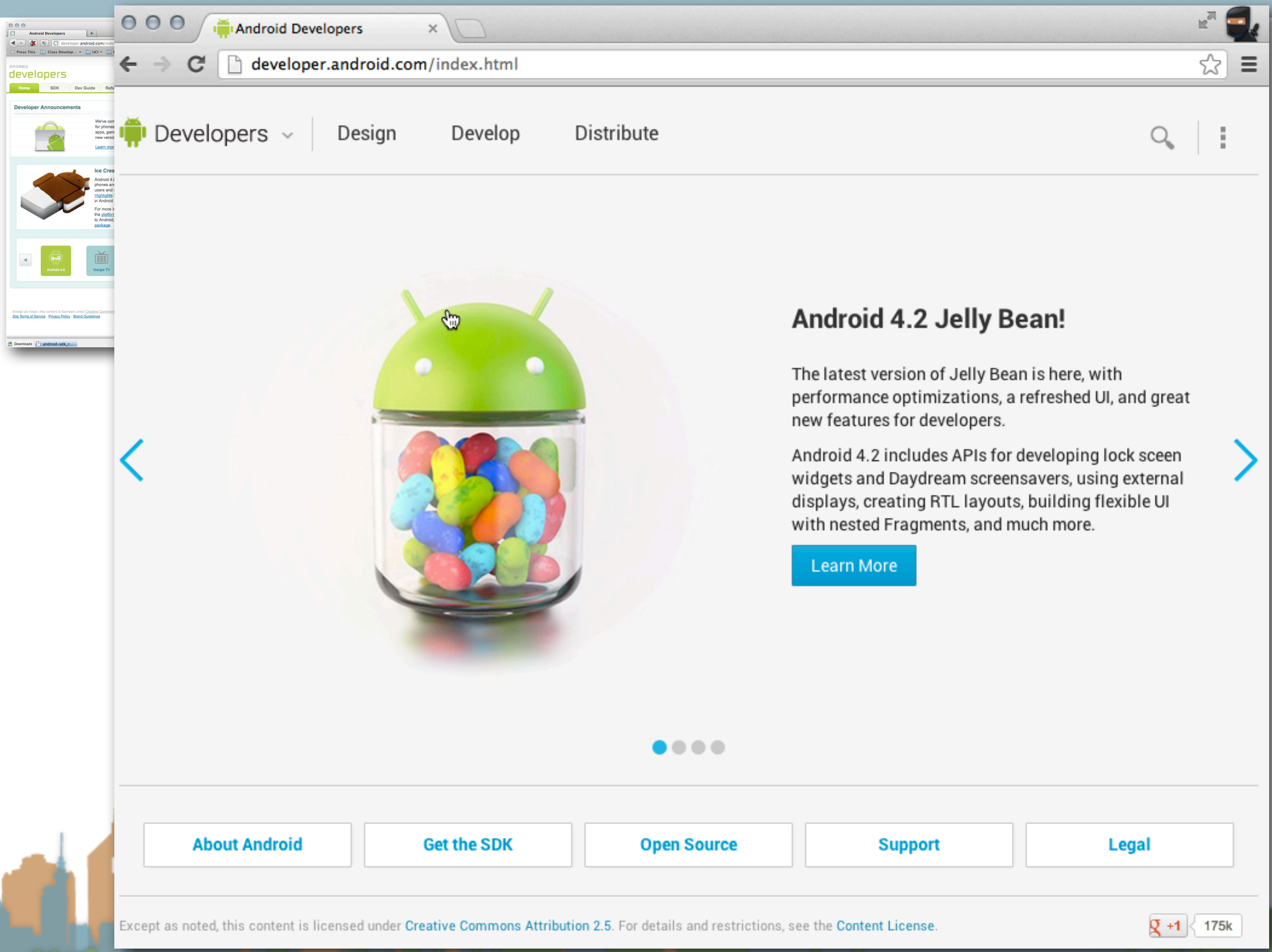
Assoc. Professor Donald J. Patterson
INF 134 Fall 2012

Intro to Android



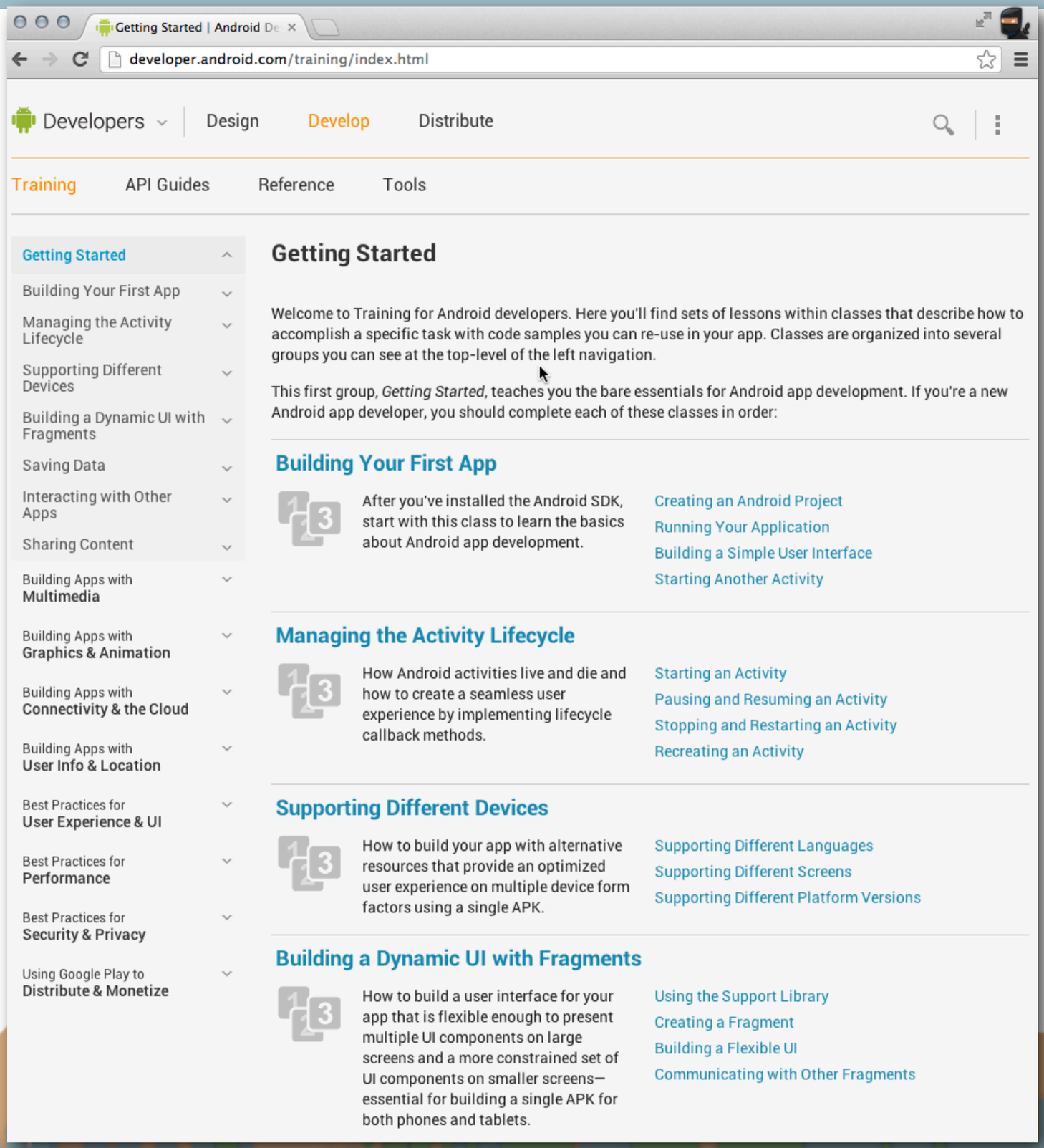
http://developer.android.com/guide/index.html

Intro to Android

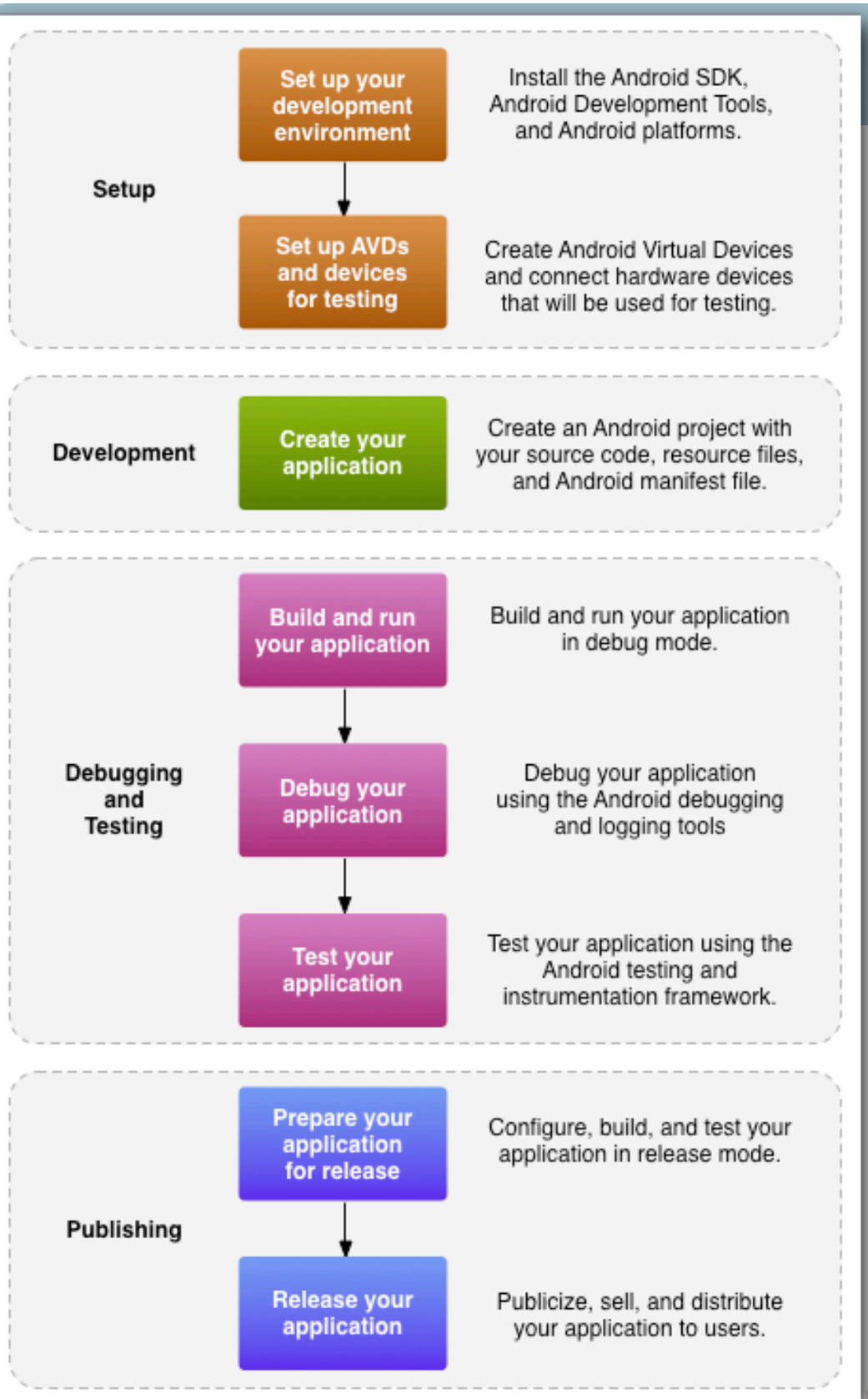


http://developer.android.com/guide/index.html

Intro to Android



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



- SDK (Software development kit)
- AVD (Android Virtual Device)
- .apk (Android application package file)
- “platform”: Gingerbread v2.3.4 is our target (Release 10)
- ADT (Android Development Tools) Eclipse plug-ins plus command line tools

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

Actually Developing for Android 2.3.6

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

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

Actually Developing for Android 2.3.6

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

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

Actually Developing for Android 2.3.6

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

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

Intro to Android: Requirements

- OS
  - Windows XP (32), Vista (32/64)or Windows 7(32/64)
  - Mac OS 10.5.8 or later (Leopard, x86 only)
  - Linux (e.g., Ubuntu Lucid Lynx (32))
- IDE
  - Eclipse 3.5 or greater
    - “Eclipse IDE for Java Developers” (for example)
- Java
  - JDK 5 or 6 (1.5 or 1.6 (not just JRE!))

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

Actually Developing for Android

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

Actually Developing for Android

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

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

Intro to Android: SDK

- Option 1: Download and unpack the appropriate "Android SDK /ADT bundle"

ADT Bundle

Platform	Package	Size	MD5 Checksum
Windows 32-bit	adt-bundle-windows-x86.zip	417851015 bytes	42d9a6c15113d405a97eed05e6d42e2b
Windows 64-bit	adt-bundle-windows-x86_64.zip	417851515 bytes	73bdd1168fce0e36a27255a4335c865d
Mac OS X 64-bit	adt-bundle-mac-x86_64.zip	382957959 bytes	a320f8bbaee8572a36e68c434564bdd0
Linux 32-bit	adt-bundle-linux-x86.zip	411065882 bytes	39687b06fedfea7487ff0824a4d32ee8
Linux 64-bit	adt-bundle-linux-x86_64.zip	411217430 bytes	b0590fe9c1533da9b20ea65525b77677

- This is the SDK and a version of eclipse all set up and ready to go

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

Intro to Android: SDK

- Option 2: Download and unpack the appropriate "Android SDK"

SDK Tools Only

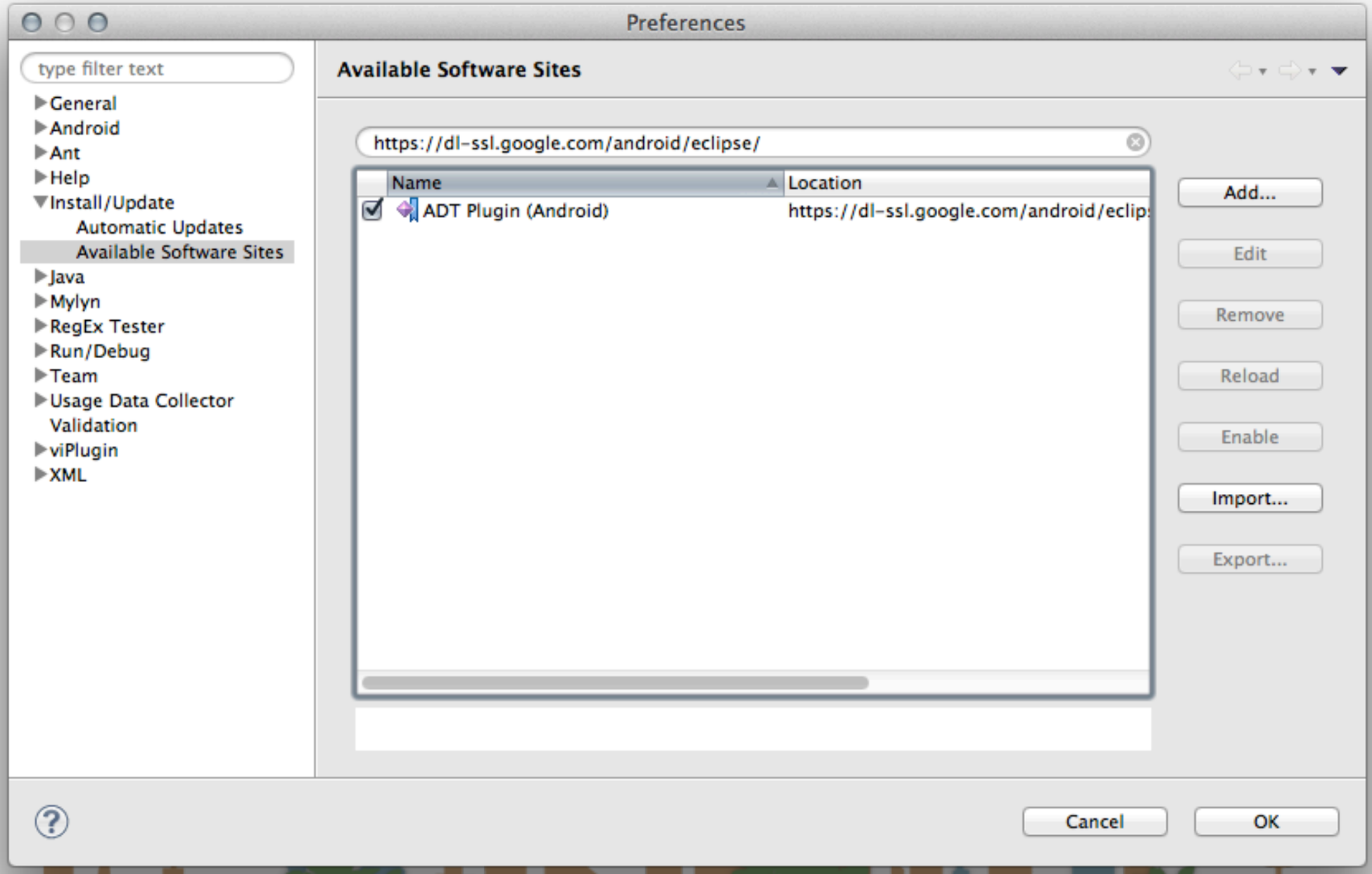
Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r21-windows.zip	99093893 bytes	7311452823470365f7975a545f8a2be4
	installer_r21-windows.exe (Recommended)	77523031 bytes	29ca8cb8f0bc8db627fa2adc2139a3cc
Mac OS X	android-sdk_r21-macosx.zip	65792626 bytes	67e46adca90dd18d7291443f6c15d6af
Linux	android-sdk_r21-linux.tgz	91378351 bytes	7f8d73b629f808cdcfc9f9900bbd7580

- This is just the SDK if you want to add Android to an existing Eclipse IDE installation

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

Intro to Android: SDK

- Option 2: Install the Eclipse Android ADT tools



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

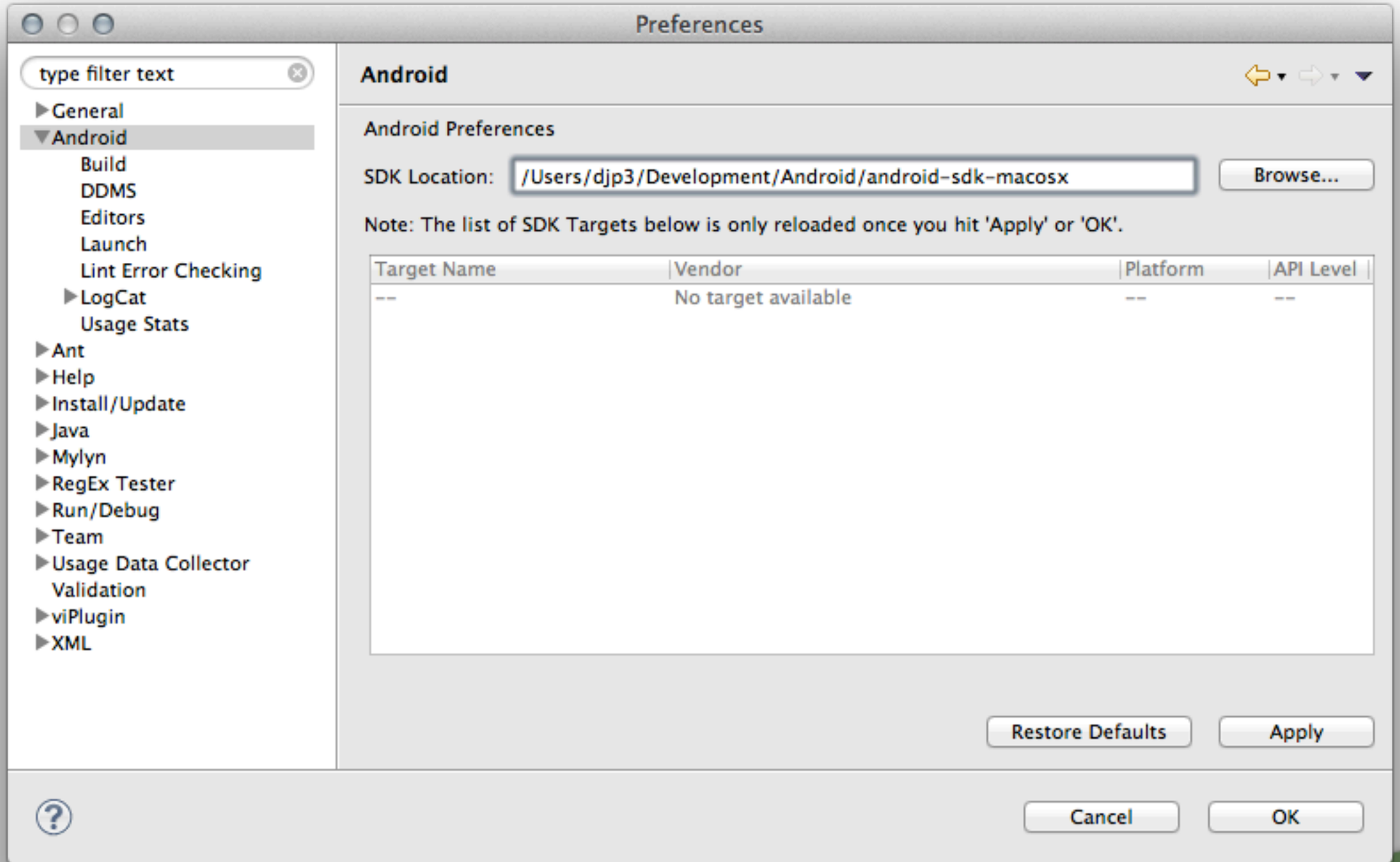
Intro to Android: Eclipse Plug-in

- The Eclipse Plug-in is called the "ADT"
  - Android Development Tools
  - Support for menu options in Eclipse which support
    - Automatically building Android projects
    - User-Interface building for Android
    - Debugging support for Android
    - Packaging files for the Android Market (.apk files)
- The ADT is installed from within Eclipse

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

Intro to Android: SDK

- Option 2: Restart Eclipse and load the SDK location



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

Intro to Android: SDK

- Option 2: Pick which platform you are developing for

Android SDK Manager

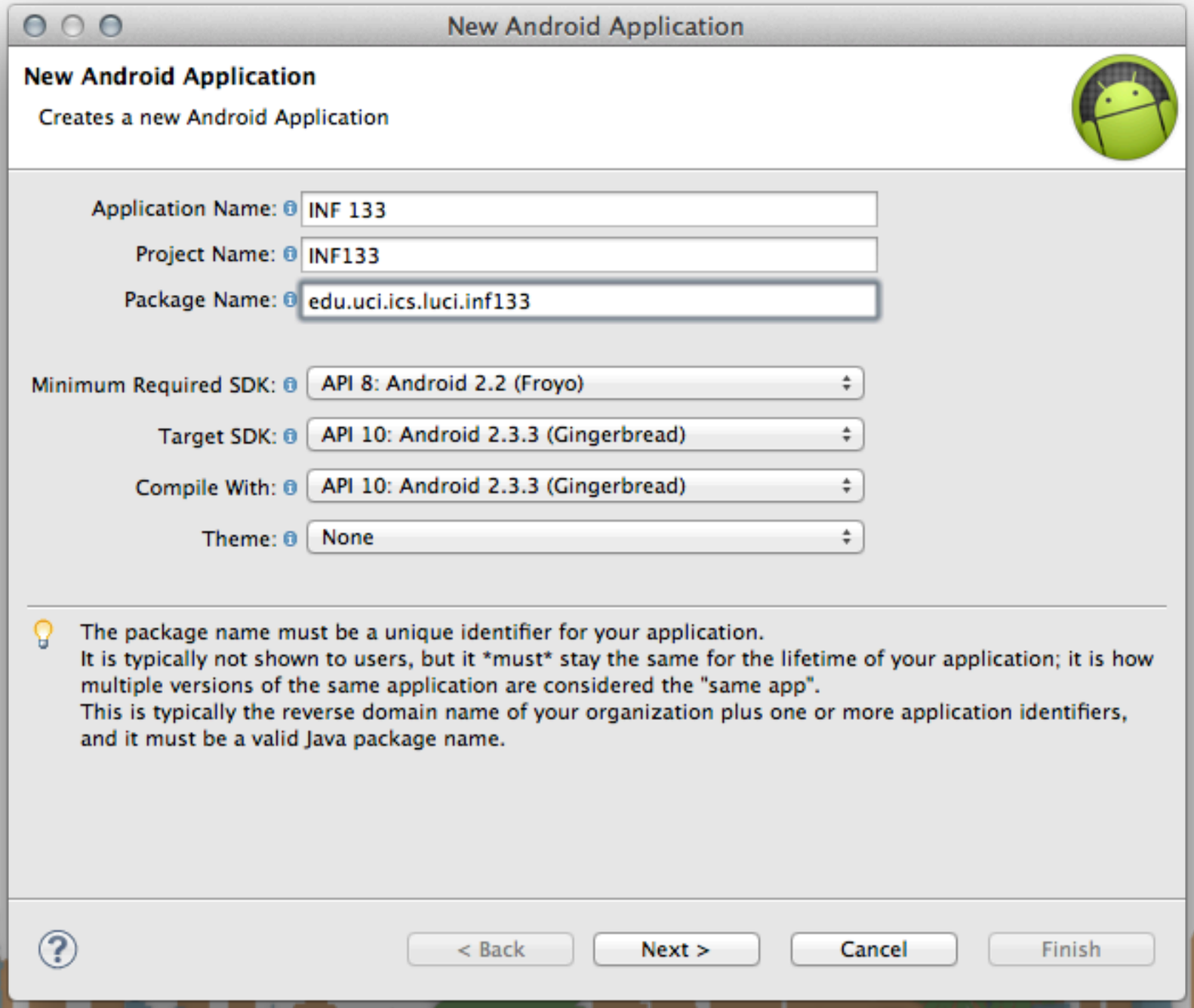
SDK Path: /Users/djp3/Development/Android/android-sdk-macosx

Packages

Name	API	Rev.	Status
Tools			
Android SDK Tools		21	Installed
Android SDK Platform-tools		16	Installed
Android 4.2 (API 17)			
Android 4.1.2 (API 16)			
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)			
SDK Platform	10	2	Installed
Samples for SDK	10	1	Installed
Google APIs	10	2	Installed
Intel Atom x86 System Image	10	1	Installed
Dual Screen APIs	10	1	Installed
Real3D	10	2	Installed
Sony Xperia Extensions EDK 2.0	10	2	Installed
Android 2.2 (API 8)			
Android 2.1 (API 7)			
Android 1.6 (API 4)			
Android 1.5 (API 3)			
Extras			
Android Support Library		11	Installed
Google AdMob Ads SDK		*8*	*Not installed*
Google Analytics SDK		*2*	*Not installed*
Google Cloud Messaging for Android Library		*3*	*Not installed*
Google Play services		*1*	*Not installed*
Google Play APK Expansion Library		*2*	*Not installed*

Intro to Android:

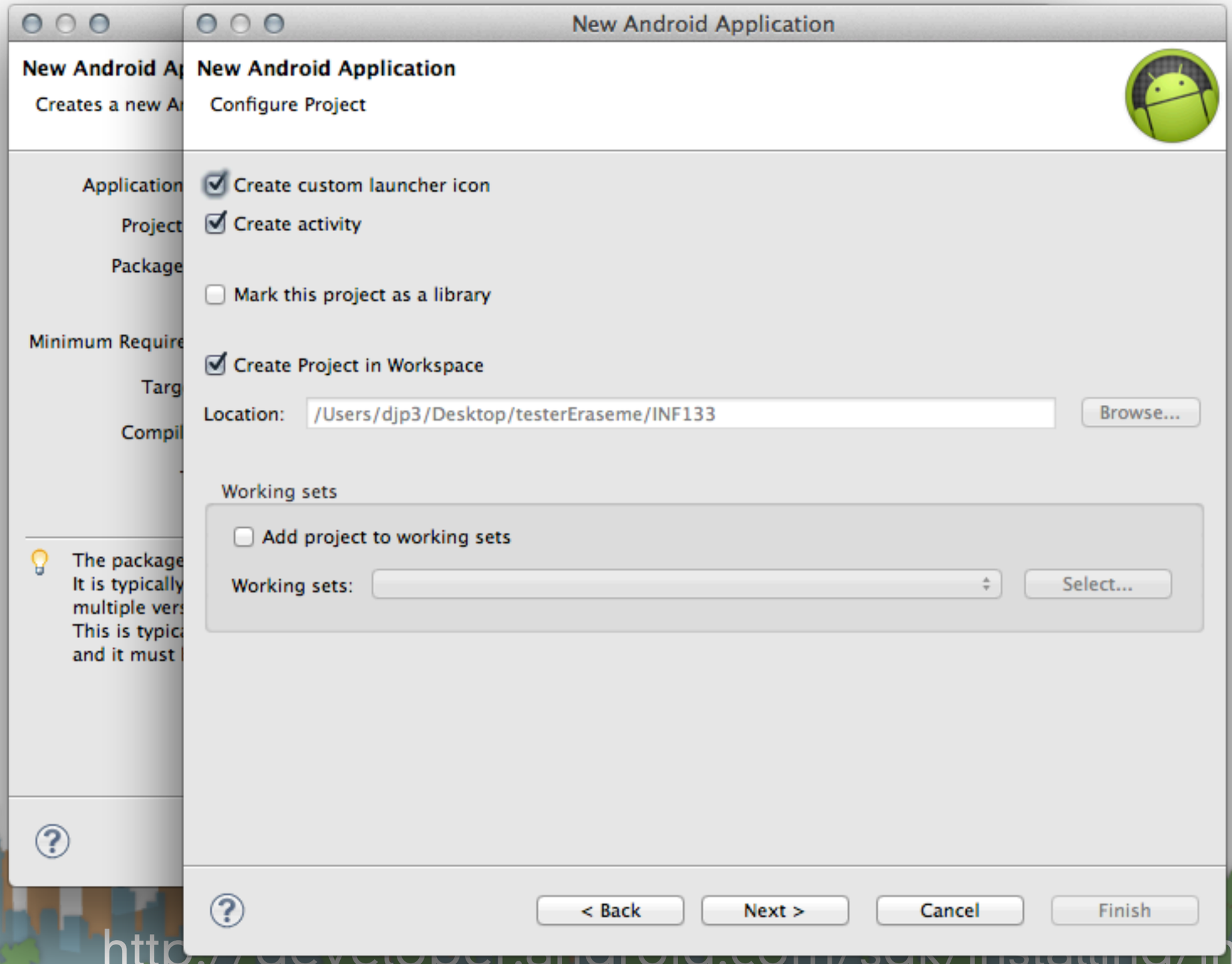
- Set up a project



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

Intro to Android:

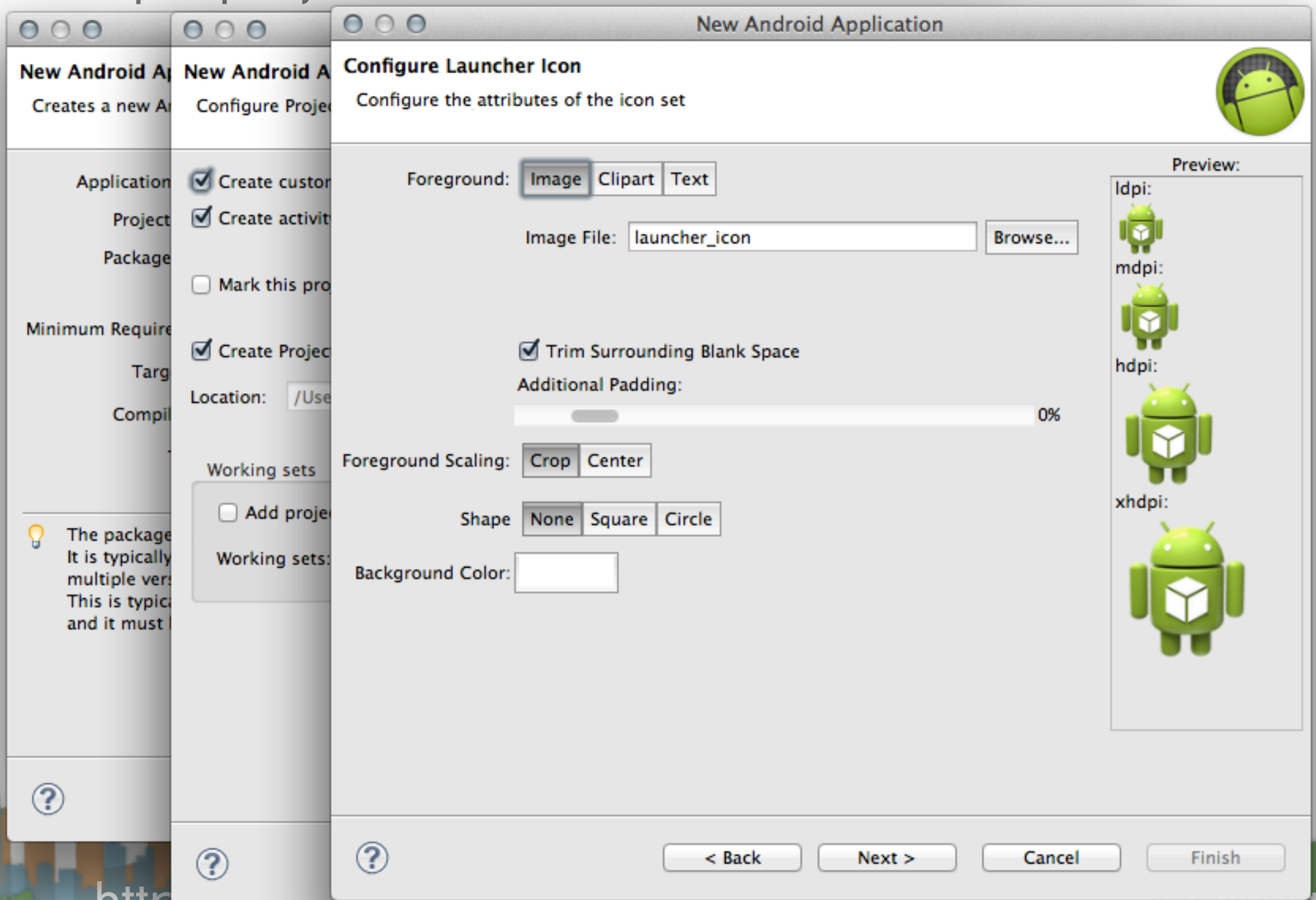
- Set up a project



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

Intro to Android:

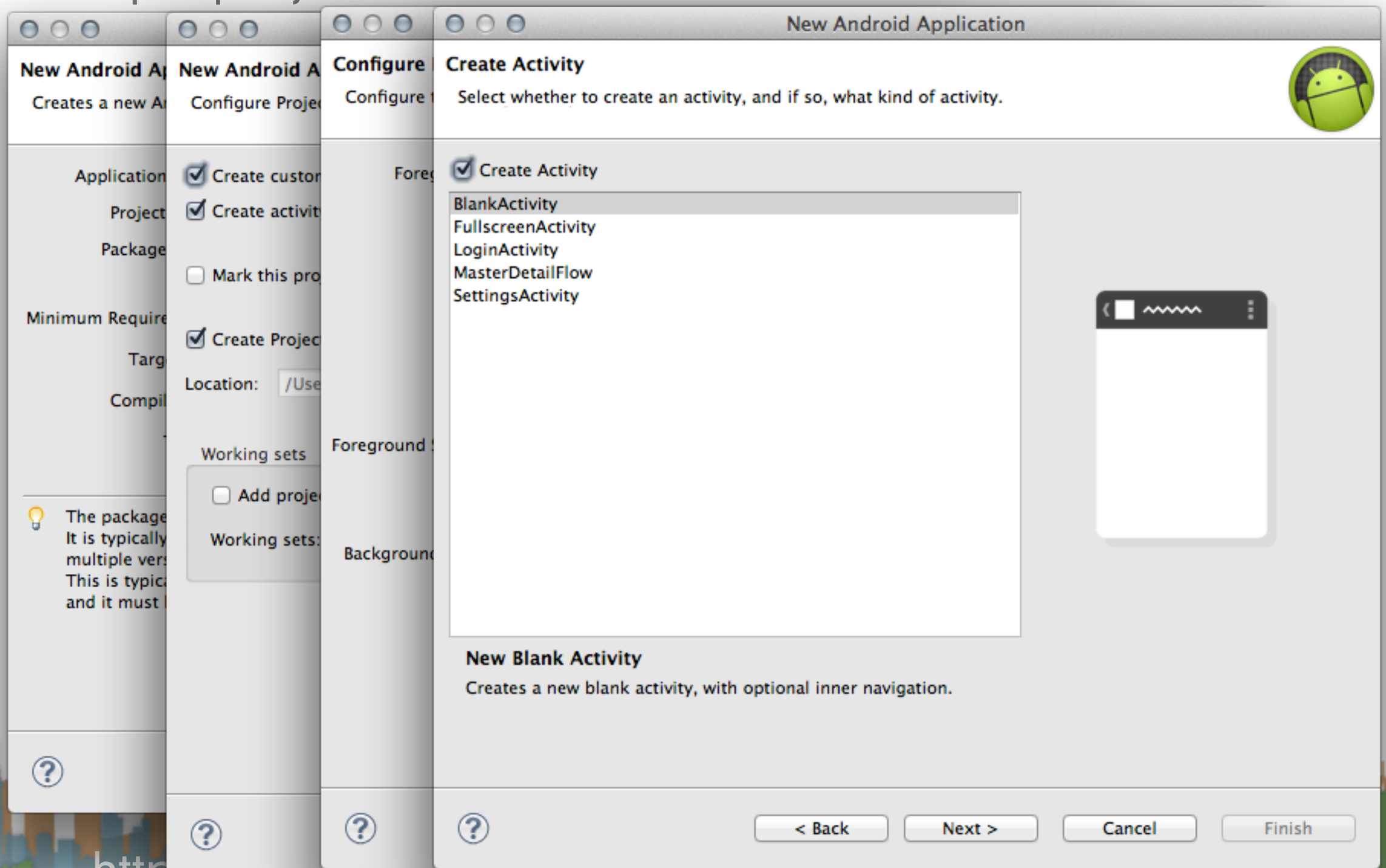
- Set up a project



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

Intro to Android:

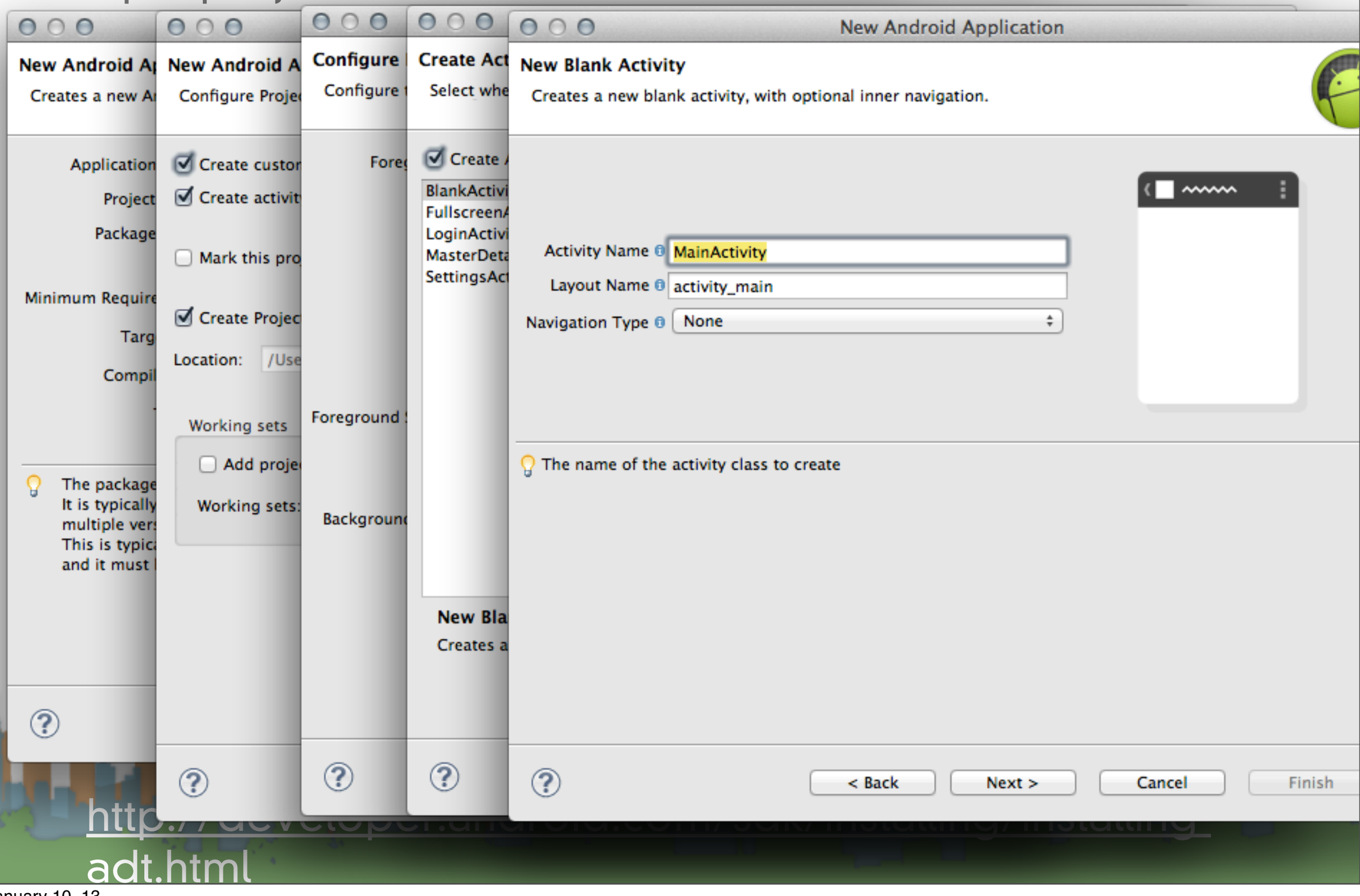
- Set up a project



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

Intro to Android:

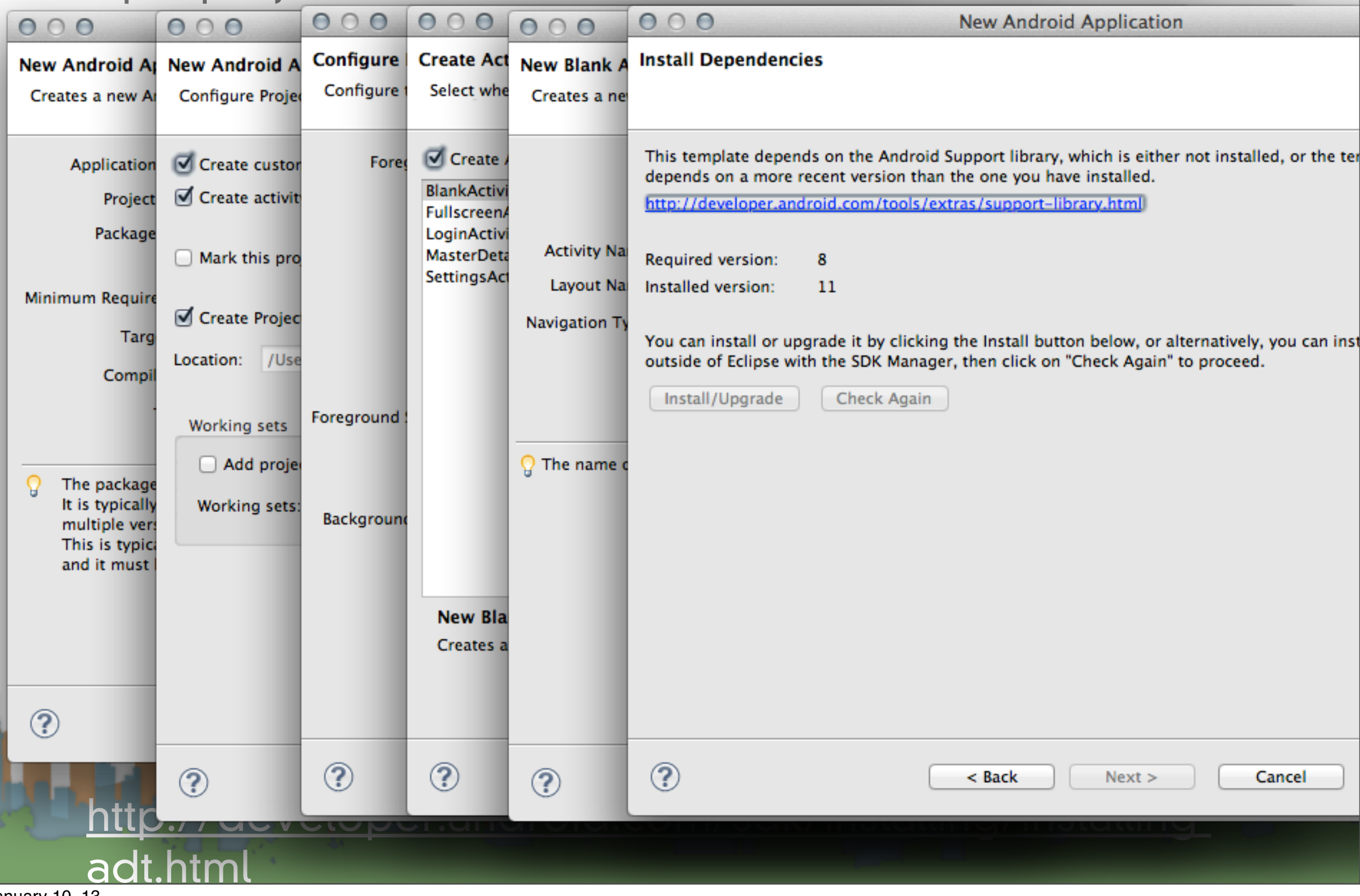
- Set up a project



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

Intro to Android:

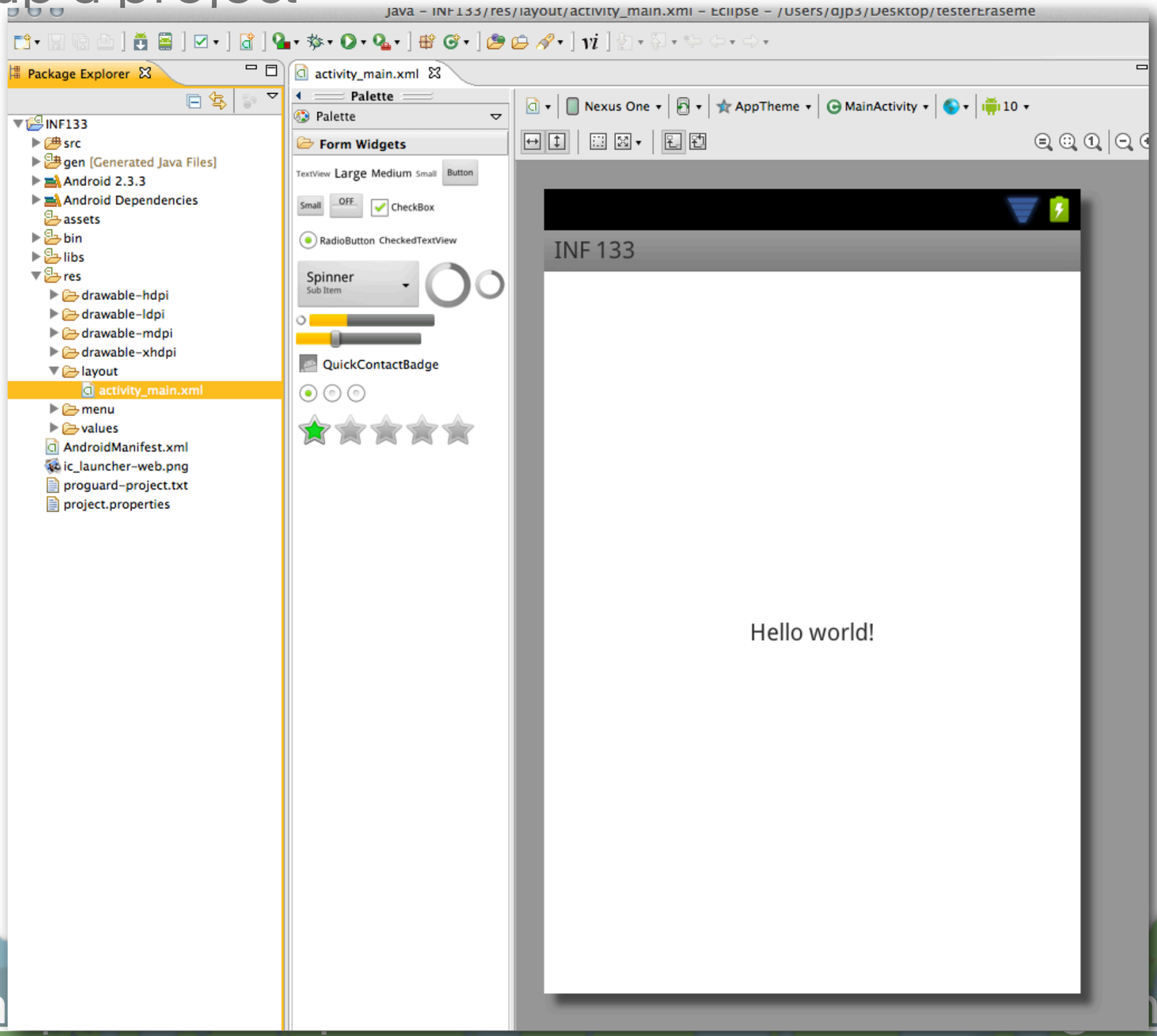
- Set up a project



http://developer.android.com/tools/extras/adt.html

Intro to Android:

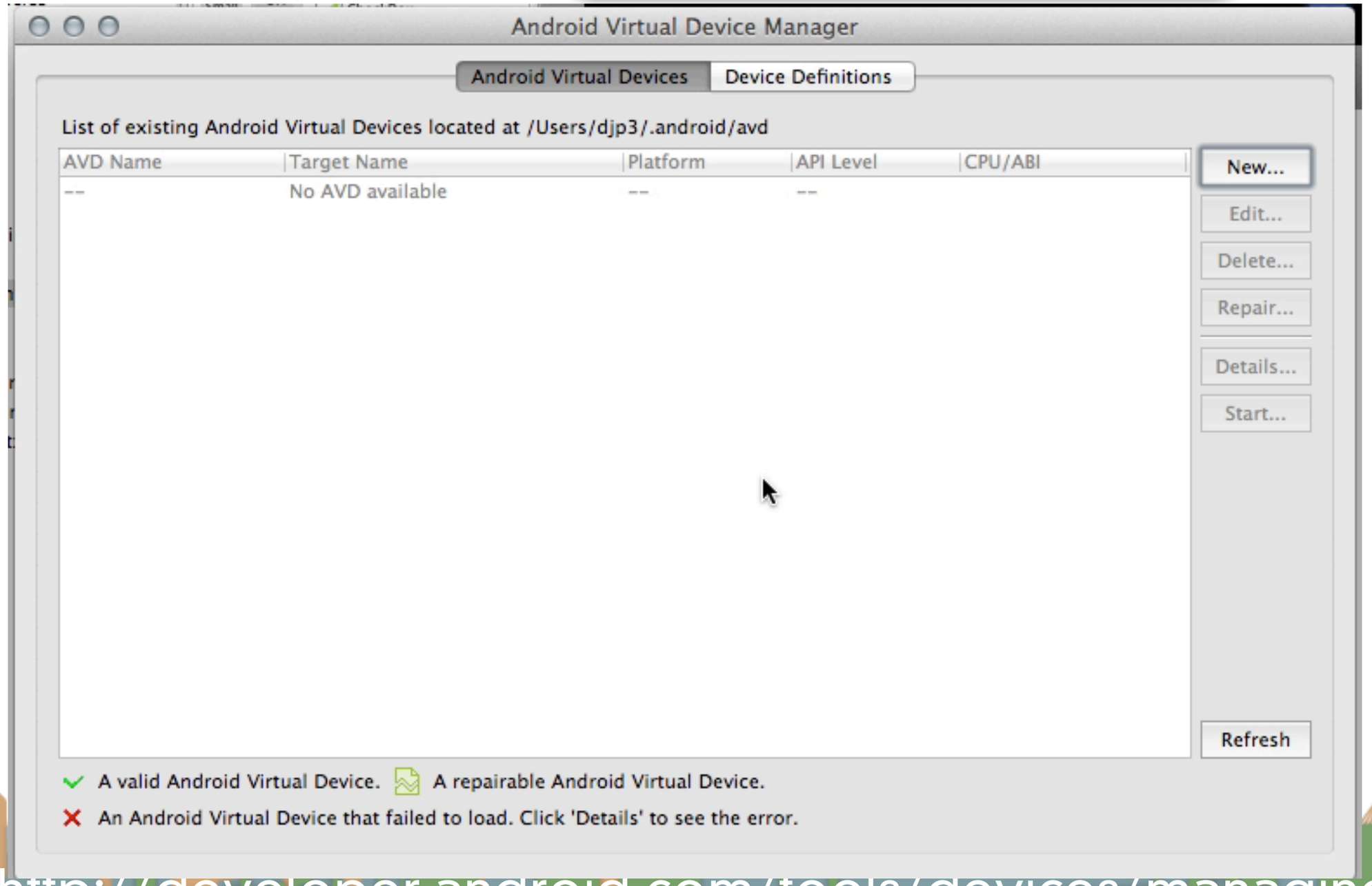
- Set up a project



h[illegible]nstalling-adt.html

Intro to Android: AVD

- Set up a virtual device



http://developer.android.com/tools/devices/managing-avds.html

Actually Developing for Android

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

Actually Developing for Android

- Requirements
- SDK/AVD
- Eclipse Plug-in
- Hello World

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

Making the phone work

- Charge the phone to 100%
  - USB to computer
  - USB to wall plug
- Wipe the phone
  - “menu” -> “settings” -> “privacy” -> “factory data reset”
    - erase the SD card too
- If necessary, go through on-phone tutorial

http://www.google.com/support/android/bin/topic.py?hl=en&topic=28930

Making the phone work

- Go through on-phone tutorial
  - You do not need a SIM card (no phone calls)
  - You should connect to WiFi
  - Do not sync to your Google (skip it)
  - Enable Location reporting
  - Set Date and Time to automatic

http://www.google.com/support/android/bin/topic.py?hl=en&topic=28930

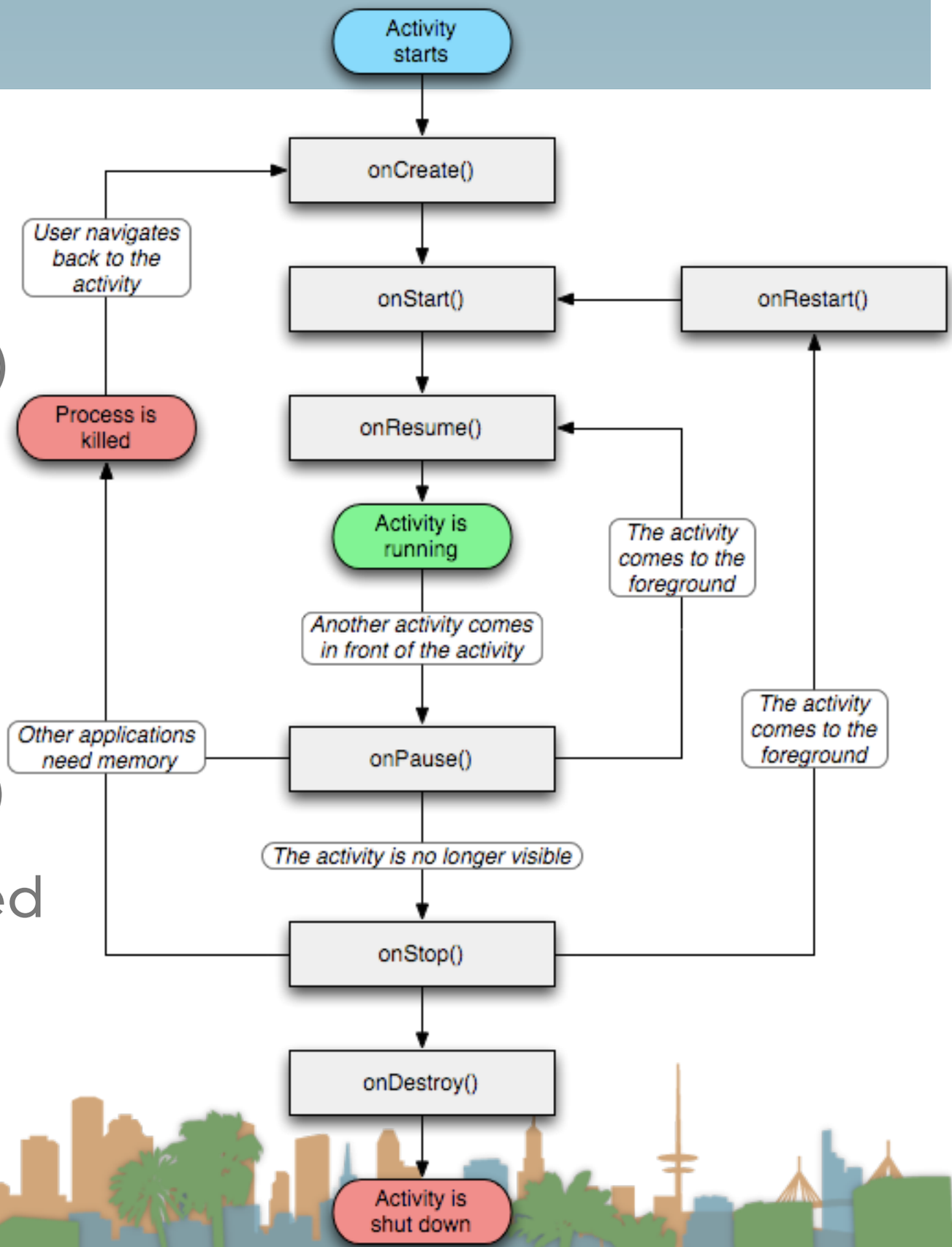
Making the phone work

- Turn on developer mode
  - “home”->”menu”->”settings”->”applications” -> “Development”
    - “USB debugging” on
    - “Stay awake” on
    - “Allow mock locations” on
- Dial *#*#CHECKIN#*#*
  - to update phone software

http://www.google.com/support/android/bin/topic.py?hl=en&topic=28930

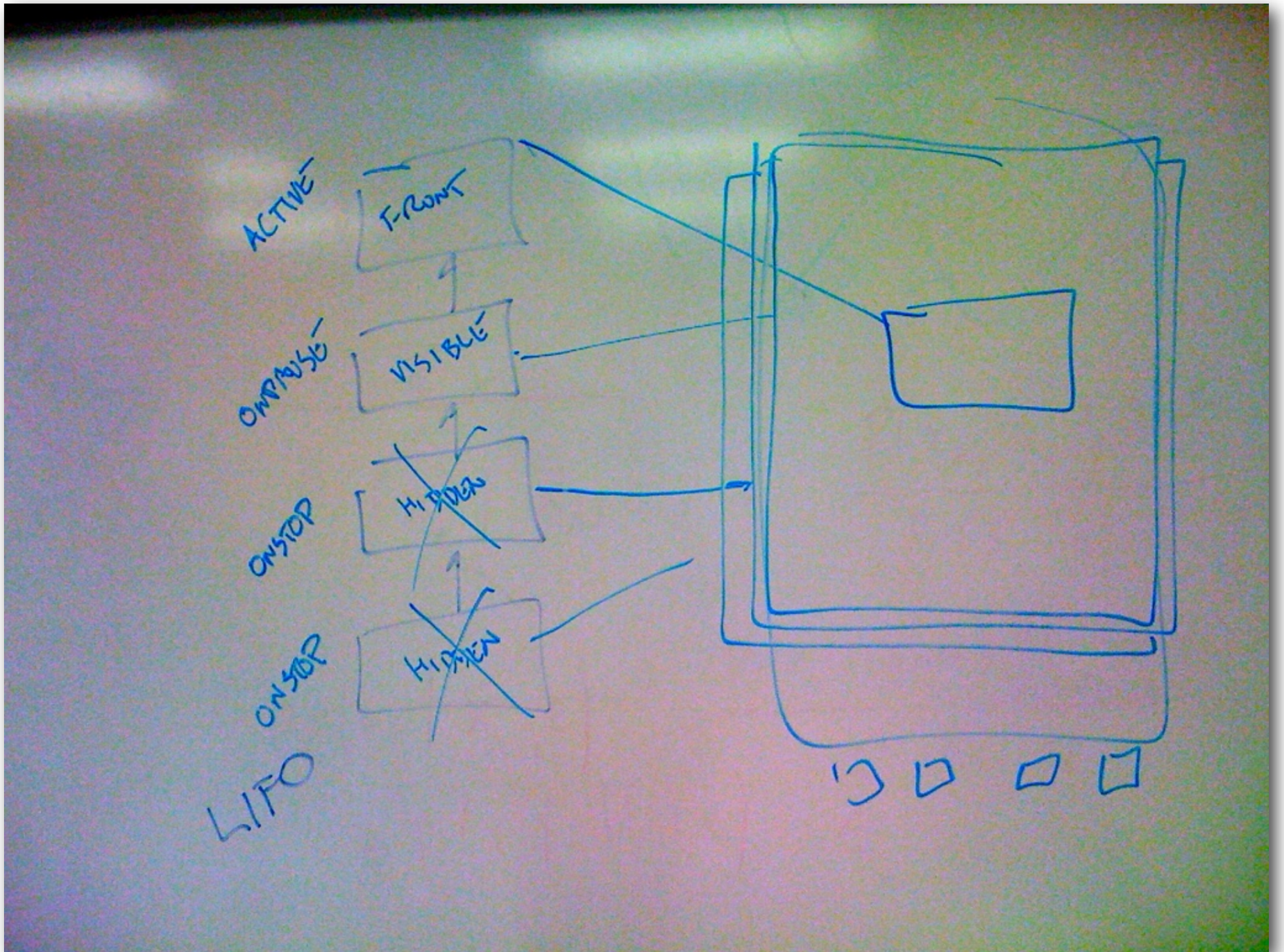
Activity Lifecycle

- Key loops
  - Entire Lifetime
    - onCreate()- onDestroy()
  - Visible Lifetime
    - onStart() - onStop()
  - Foreground Lifetime
    - onResume() - onPause()
- onPause() may be followed by kill



http://developer.android.com/reference/android/app/Activity.html

Activity Stack



Intents

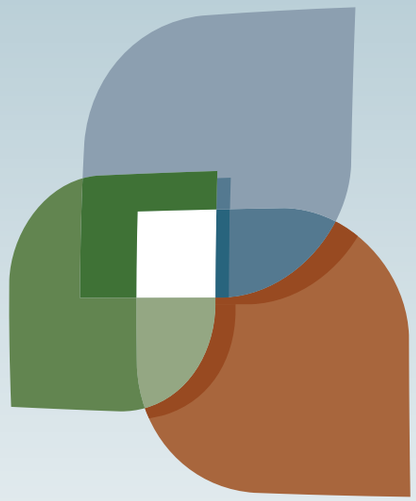
- Activities, Services and Broadcast Receivers are activated through messages, called Intents.
- What do I mean by "messages"?
  - Late-run time binding between components in the same or different applications.
  - An abstract method call
  - Sent between apps on the phone
- An Intent is actually an object holding data about the message

http://developer.android.com/guide/components/intents-filters.html

Intents

- Examples:
  - If you want to send an Intent you might do the following:
    - startActivity(new Intent(Intent.ACTION_DIAL,"tel:5555555555"));
  - If you want to be notified when the user has disconnected the power cord you might create a BroadcastReceiver that registers for the "ACTION_POWER_DISCONNECTED" Intent.
  - (Look at all of the Intents)

http://developer.android.com/reference/android/content/Intent.html



LUCI

Thursday, January 10, 13