

Android Training Modules

February 1, 2011



Hands-On Training Modules

These modules are designed to run 75 minutes with hands-on exercises or 40 minutes lecture-only.

Title	Topics	Exercise
Android: Guts & GUIs	<ul style="list-style-type: none">● What makes up an Android application● Using XML for UI layout● Basic widgets● Introductions to LinearLayout, RelativeLayout, and TableLayout (note: takes two module slots, 150 minutes total)	LunchList: Create a simple form
Development Tools	<ul style="list-style-type: none">● Hierarchy Viewer● DDMS and log results● DDMS and simulating calls● DDMS and file upload/download● DDMS and screenshots● Making and using SD card images	LunchList: Create a fancier form
Selection Widgets	<ul style="list-style-type: none">● The role and use of adapters● Lists● Spinners● Grids● Auto-completion fields	LunchList: Add a selection list to the form
Fancy Lists	<ul style="list-style-type: none">● Using custom layouts in list entries● Populating list entries● Recycling views● Using the holder pattern	LunchList: Add icons to the selection list
Even More Widgets	<ul style="list-style-type: none">● Date and time pickers● Tabbed dialogs● ViewFlipper	LunchList: Make the form span multiple tabs: list on one, details on another
Menus and Messages	<ul style="list-style-type: none">● Adding option menus● Adding popup menus● Showing dialogs● Raising toasts	LunchList: Add a menu to the app that raises a toast
Threads	<ul style="list-style-type: none">● The “one thread to rule them all” rule● Handlers● Alternatives to Handler● Progress bars	LunchList: Add a menu choice that does fake background processing and updates a progress bar
The Activity Lifecycle	<ul style="list-style-type: none">● How activities are supposed to behave● onCreate() and onDestroy()● onPause() and onResume()● onSaveInstanceState()	LunchList: Stop the fake background processing when the activity is paused, pick up again when resumed

Resources	<ul style="list-style-type: none"> ● General theory of resources ● Drawables ● Values ● XML ● Offering multiple resource sets 	LunchList: Support a different layout for landscape instead of portrait views
Databases	<ul style="list-style-type: none"> ● Introducing SQLite ● SQLiteOpenHelper and creating a database ● Opening and closing a database ● Working with cursors ● Inserts, updates, and deletes 	LunchList: List shows database entries; menu choices for add/edit/delete, used in form
Using Multiple Activities	<ul style="list-style-type: none"> ● Having multiple activities in your project ● Starting other activities from your project ● Getting results from activities ● Using URIs to launch system activities 	LunchList: Remove the tabs, put the list on one activity and the detail form on a sub-activity
Preferences and Files	<ul style="list-style-type: none"> ● Using shared preferences ● Preference screens ● Preference XML configuration ● Preference change notification ● Simple reading and writing of files 	LunchList: Allow users to configure default values
Handling Rotation Events	<ul style="list-style-type: none"> ● Theory of rotation in Android ● Using onSaveInstanceState() ● Using onRetainNonConfigurationInstance() ● Using onConfigurationChanged() ● Blocking screen rotation 	LunchList: Allow a partially-completed form to be kept over a rotation
Using HttpClient	<ul style="list-style-type: none"> ● Overview of native networking options ● Simple HTTP GET using HttpClient ● Posting forms ● Cookies ● Authentication ● XML and JSON 	Patchy: update identi.ca status on using REST API
Extending Android with JARs	<ul style="list-style-type: none"> ● Simple mechanics of the libs/ directory ● What will and will not work ● How best to integrate a third-party JAR 	Patchy: experiment with third-party Twitter API wrappers
Creating and Using Services	<ul style="list-style-type: none"> ● Role of services ● Service lifecycle methods ● Sharing state in local services ● Threads and local services ● Starting and stopping services ● Getting updates via shared state ● Getting updates via Intents and BroadcastReceivers <p>(note: takes two module slots, 150 minutes total)</p>	Patchy: poll for friends timeline updates, populate ListView with friends timeline updates
Remote Services	<ul style="list-style-type: none"> ● Role of remote services ● Creating AIDL interfaces ● Exporting and importing AIDL interfaces ● Callbacks from service to client 	Patchy: make the polling service be remote, using AIDL for control and callbacks for async updates

Notifications	<ul style="list-style-type: none"> ● Role of notifications ● Raising a notification ● Augmenting notifications via hardware ● Reacting to selected notifications ● Handling multiple notification-worthy events 	Patchy: raise notifications when posts hit the timeline containing a keyword
Location Tracking	<ul style="list-style-type: none"> ● Android's location tracking model ● Getting updates as you move ● Getting the latest fix ● Proximity alerts ● Choosing providers via criteria ● Testing via DDMS 	Patchy: add “L:30.010,-90.007”-style markup on user request to tweets, using current location
Maps	<ul style="list-style-type: none"> ● Rules and API keys ● Adding a map, with center and zoom levels ● Overlays and MyLocationOverlay ● Custom overlays with push-pins ● Zoom controls 	Patchy: for updates with “L:30.010,-90.007”-style markup in timeline, show location
Media	<ul style="list-style-type: none"> ● Supported audio formats ● Simple media playback ● Streaming audio ● Supported video formats ● Simple video playback ● Streaming video 	Patchy: add option to view “helpcast” on how to use the application, playing back a video
Integrating WebKit	<ul style="list-style-type: none"> ● Launching a standalone Browser activity ● Adding a WebKit to your application ● Loading local content ● Controlling what happens on link clicks 	Patchy: launch browser on in-post links; show “online help” from a local file
Advanced WebKit	<ul style="list-style-type: none"> ● Exposing Java code to Javascript ● Calling back into Javascript from Java ● Richer local content stores 	Patchy: online help is customized with user info via injected Java
Animation	<ul style="list-style-type: none"> ● Theory of animation, XML, and Java ● Translation animations ● Alpha animations ● Completion listeners ● Acceleration interpolaters ● Animation sets 	Patchy: make the status update widgets show and hide based on a menu choice
Intent Filters	<ul style="list-style-type: none"> ● Role of filters ● Intent-matching rules ● Filters in your manifest ● Filters in dynamic BroadcastReceivers 	Patchy: publish status updates via broadcast Intents
Using Content Providers	<ul style="list-style-type: none"> ● Content provider “MIME” types ● Searching for content ● Adding, changing, and removing content ● Working with content files 	Patchy: highlight tweets from people in your contacts
System Events	<ul style="list-style-type: none"> ● Starting your service on system boot ● Connectivity status ● Battery status 	Patchy: poll less frequently when the battery is low
Alarm Manager	<ul style="list-style-type: none"> ● AlarmManager for periodic actions ● WakeLocks for work done while phone is asleep 	Patchy: poll via a scheduled alarm

Searches	<ul style="list-style-type: none"> ● Integrate global (Web) search into your application ● Enable your application for local search ● Tie into the Quick Search Box 	LunchList: search restaurants by parts of their name
Introspection	<ul style="list-style-type: none"> ● Picking a piece of content ● Choosing an application for an Intent ● Finding actions available for a piece of content 	Contacter: display list of available actions to perform on a selected contact
Basic App Widgets	<ul style="list-style-type: none"> ● App widgets and Android security ● Creating an AppWidgetProvider ● Defining the metadata ● Updating the app widget 	LunchList: display a random restaurant name in an app widget
Advanced App Widgets	<ul style="list-style-type: none"> ● Using an IntentService ● Responding to widget clicks ● Other update options: varying schedule, on click, on other event ● UI framework integration (HTC Sense, MOTOBLUR) 	LunchList: button click selects another random restaurant
Testing	<ul style="list-style-type: none"> ● Instrumentation ● Scripting your testing 	LunchList: use Test Monkey to “bash test” the app

Lecture-Only Training Modules

These modules are designed to run 40 minutes lecture-only, except as noted.

Title	Topics
Camera	<ul style="list-style-type: none">● Working with previews● Taking pictures
Handling Multiple Devices and API Versions	<ul style="list-style-type: none">● Filtering, to prevent applications from being installed where they will not work● Writing code that can dynamically react to available hardware capabilities● Writing code that can adapt to multiple API versions
Handling Screen Sizes	<ul style="list-style-type: none">● Writing UIs that are independent of screen size and density● Writing UIs that are aware of, and can react to, screen size and density● Taking advantage of larger screens
Going to Production	<ul style="list-style-type: none">● Code signing● Distributing through the Android Market● Distributing through alternative markets● Distributing through your own Web site● Manual distribution● Handling updates <p>(note: runs about one hour)</p>
Android Security Overview	<ul style="list-style-type: none">● Kernel security and its impact on Android applications● Requesting permissions for use by your application● Requiring permissions for other applications to integrate with yours
Barcodes with ZXing	<ul style="list-style-type: none">● Role of QR codes● Using QR codes with Google Googles and Barcode Scanner● Integrating your application with Barcode Scanner
Android Beyond Java	<ul style="list-style-type: none">● HTML5, Flash, and other Web-based application frameworks● Other JVM languages (Scala, Clojure)● Introduction to the Native Development Kit● Role of scripting languages and the Android Scripting Environment
SMS	<ul style="list-style-type: none">● Sending an SMS via the user's choice of messaging applications● Sending an SMS directly via the SmsManager● Limitations on working with SMS in other ways in Android
Android Internals	<ul style="list-style-type: none">● Android architecture high-level overview● Role of Dalvik vis-a-vis Java development● The Android open source project <p>(note: runs about one hour)</p>
Limits in Mobile Development	<ul style="list-style-type: none">● Enumerating the constraints encountered in mobile development● Hinting at how Android helps you deal with these constraints <p>(note: designed to be delivered early in training event)</p>