**Glossary**

**.apk file** [1]

Android application package file. Each Android application is compiled and packaged in a single file that includes all of the application's code (.dex files), resources, assets, and manifest file. The application package file can have any name but must use the .apk extension.

**Action** [1]

A description of something that an Intent sender wants done. An action is a string value assigned to an Intent. Action strings can be defined by Android or by a third-party developer. For example, android.intent.action.VIEW for a Web URL, or com.example.rumbler.SHAKE\_PHONE for a custom application to vibrate the phone.

**Activity** [1]

A single screen in an application, with supporting Java code, derived from the [Activity](http://developer.android.com/reference/android/app/Activity.html) class. Most commonly, an activity is visibly represented by a full screen window that can receive and handle UI events and perform complex tasks, because of the Window it uses to render its window. Though an Activity is typically full screen, it can also be floating or transparent.

**Android** [2]

Google's open-source mobile operating system. It's used primarily in smartphones but also can be found on tablets, Mobile Internet Devices (MIDs) or even in kitchen appliances and automobile navigation.

**API** [3]

API, an abbreviation of **a***pplication* **p***rogram* **i***nterface*, is a set of [routines](http://www.webopedia.com/TERM/R/routine.html), [protocols](http://www.webopedia.com/TERM/P/protocol.html), and tools for building [software applications](http://www.webopedia.com/TERM/A/application.html). A good API makes it easier to develop a [program](http://www.webopedia.com/TERM/P/program.html) by providing all the building blocks. A [programmer](http://www.webopedia.com/TERM/P/programmer.html) then puts the blocks together.

**Application** [1]

From a component perspective, an Android application consists of one or more activities, services, listeners, and intent receivers. From a source file perspective, an Android application consists of code, resources, assets, and a single manifest. During compilation, these files are packaged in a single file called an application package file (.apk).

**Bluetooth** [2]

A short-range radio build into smartphones that lets you connect headsets, speakerphones or even computers to your smartphone.

**Broadcast Receiver** [1]

An application class that listens for Intents that are broadcast, rather than being sent to a single target application/activity. The system delivers a broadcast Intent to all interested broadcast receivers, which handle the Intent sequentially.

**Dialog** [1]

A floating window that that acts as a lightweight form. A dialog can have button controls only and is intended to perform a simple action (such as button choice) and perhaps return a value. A dialog is not intended to persist in the history stack, contain complex layout, or perform complex actions. Android provides a default simple dialog for you with optional buttons, though you can define your own dialog layout. The base class for dialogs is [Dialog](http://developer.android.com/reference/android/app/Dialog.html).

**Fragment** [2]

A [Fragment](http://developer.android.com/reference/android/app/Fragment.html) represents a behavior or a portion of user interface in an [Activity](http://developer.android.com/reference/android/app/Activity.html). You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities. You can think of a fragment as a modular section of an activity, which has its own lifecycle, receives its own input events, and which you can add or remove while the activity is running (sort of like a "sub activity" that you can reuse in different activities).

**Google**™[2]

Our benevolent overlord, and owner of Android.

**Intent** [1]

A message object that you can use to launch or communicate with other applications/activities asynchronously. An Intent object is an instance of [Intent](http://developer.android.com/reference/android/content/Intent.html). It includes several criteria fields that you can supply, to determine what application/activity receives the Intent and what the receiver does when handling the Intent. Available criteria include the desired action, a category, a data string, the MIME type of the data, a handling class, and others. An application sends an Intent to the Android system, rather than sending it directly to another application/activity. The application can send the Intent to a single target application or it can send it as a broadcast, which can in turn be handled by multiple applications sequentially. The Android system is responsible for resolving the best-available receiver for each Intent, based on the criteria supplied in the Intent and the Intent Filters defined by other applications. For more information, see [Intents and Intent Filters](http://developer.android.com/guide/components/intents-filters.html).

**Intuitive** [4]

An adjective meaning: perceived by, resulting from, or involving [intuition](http://dictionary.reference.com/browse/intuition).

**Layout Resource** [1]

An XML file that describes the layout of an Activity screen.

**Manifest File** [1]

An XML file that each application must define, to describe the application's package name, version, components (activities, intent filters, services), imported libraries, and describes the various activities, and so on. See [The AndroidManifest.xml File](http://developer.android.com/guide/topics/manifest/manifest-intro.html) for complete information.

**NFC** [2]

Near-field communication. Short-range communication between your phone and something else -- another phone, a cash register, etc. Used by some credit cards as a method of quick payment.

**Open Source** [2]

Software which is liberally licensed to grant the right of users to study, change, and improve it’s design through the availability of its source code.

**SDK** [2]

Stands for Software Development Kit. Generally, a set of tools used to create software for a certain platform following guidelines provided in the kit. For Android, the SDK provides tools to create applications that run on Android devices.

**Service** [1]

An object of class [Service](http://developer.android.com/reference/android/app/Service.html) that runs in the background (without any UI presence) to perform various persistent actions, such as playing music or monitoring network activity.

**View** [1]

An object that draws to a rectangular area on the screen and handles click, keystroke, and other interaction events. A View is a base class for most layout components of an Activity or Dialog screen (text boxes, windows, and so on). It receives calls from its parent object (see viewgroup, below) to draw itself, and informs its parent object about where and how big it would like to be (which may or may not be respected by the parent). For more information, see [View](http://developer.android.com/reference/android/view/View.html).

**Widget** [1]

One of a set of fully implemented View subclasses that render form elements and other UI components, such as a text box or popup menu. Because a widget is fully implemented, it handles measuring and drawing itself and responding to screen events. Widgets are all in the [android.widget](http://developer.android.com/reference/android/widget/package-summary.html) package.

<http://developer.android.com/guide/appendix/glossary.html> [1]

[http://www.androidcentral.com/dictionary [2](http://www.androidcentral.com/dictionary%20%5b2)]

<http://www.webopedia.com/TERM/A/API.html> [3]

<http://dictionary.reference.com/browse/intuitive> [4]