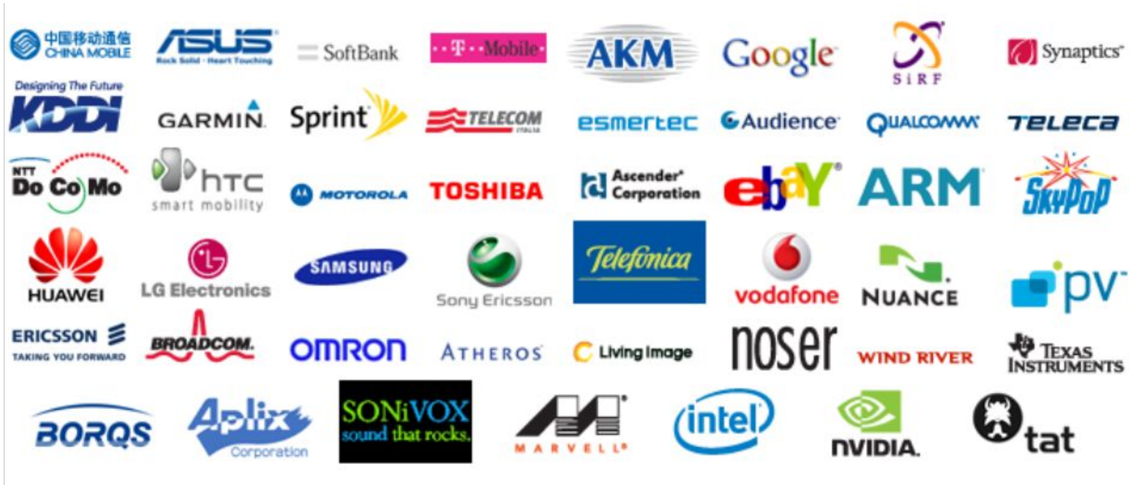
TESTING ANDROID

TUTORIAL FOR BLACK BOX NEWCOMERS

FOR WHAT ANDROID WAS MADE FOR AND BY WHOM

Android is an operating system developed by the Open Handset Alliance led by Google

The Open Handset Alliance, a consortium of several companies which include



The goal of the Open Handset Alliance is to develop open standards for mobile devices.

The Android operating system is used on smartphones, netbooks, tablet computers, Google TV, and other devices.

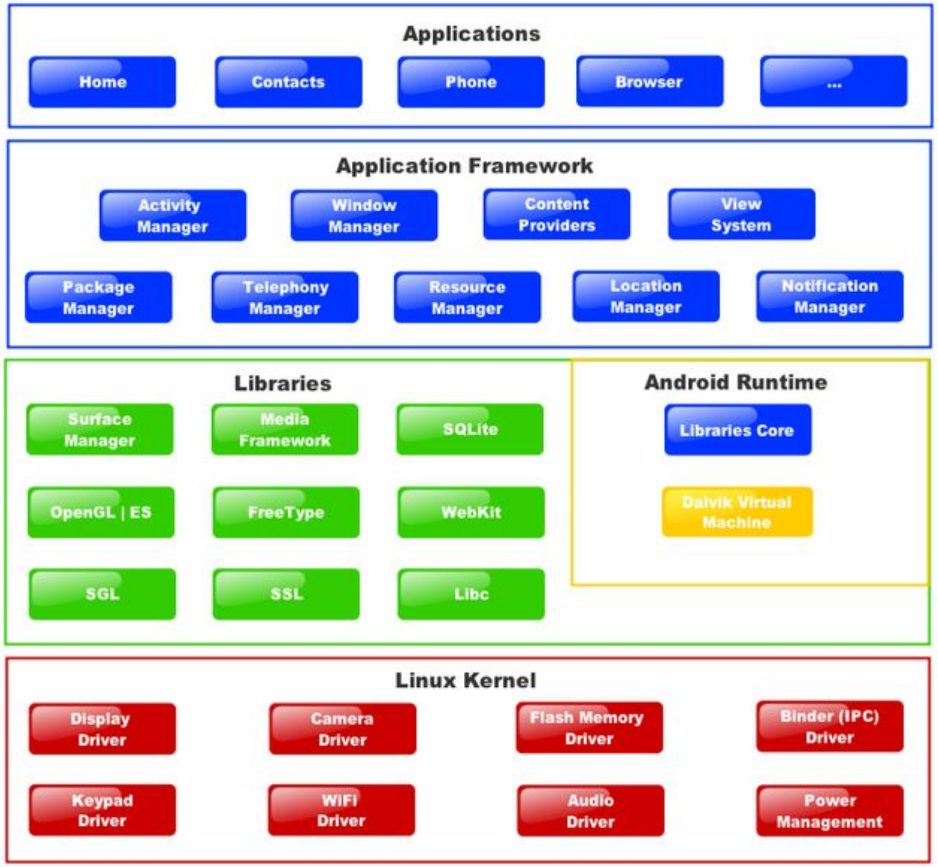
DESIGN

Linux

Android's kernel is derived from the Linux kernel. Google contributed code to the Linux kernel as part of their Android effort, but Android kernel is now a separate version or fork of the Linux kernel.

Android does not have a native X Window System nor does it support the full set of standard GNU libraries, and this makes it difficult to port existing Linux applications or libraries to Android.

ARCHITECTURE



SOFTWARE ARCHITECTURE

System Libraries

- SSL
- SQLite
- WebKit (and LibWebCore for embeddable webpages)
- Audio Manager
- Media Framework
- MediaPlayer

Graphics Libraries

- Scalable Graphics Library (SGL- for 2D)
- OpenGL for Embedded Devices (for 3D)
- FreeType (vector and bitmap fonts)
- Surface Manager
    - Composes 2D and 3D windows, widgets, apps, toolbars and more using Surface Flinger
    - Uses Binder IPC to get buffers from apps to put into frames

Hardware Abstraction Libraries

- GPS, Radio, Camera, Bluetooth
- Hardware drivers must implement in order for applications to use them
- Applications interact with the abstraction libraries, not the driver

ANDROID VS LINUX ARCHITECTURE

- No native window library
- Does not support full set of GNU libraries
- Difficult to port Linux applications to Android
- Most Android applications written in Java
- However, no Java Virtual Machine in the platform
- Java classes compiled into Dalvik virtual machine
- Dalvik - a specialized virtual machine designed specifically for Android

ANDROID SOFTWARE DEVELOPMENT KIT

- The SDK includes a comprehensive set of development tools.
- Includes a debugger, libraries, documentation, sample code
- These tools are accessed through an Eclipse plugin called ADT (Android Development Tools) or from command line
- Developing with Eclipse is preferred (but not required)

FEATURES

Handset layouts

The platform is adaptable to larger, VGA, 2D graphics library, 3D graphics library based on OpenGL ES 2.0 specifications, and traditional smartphone layouts

Storage

SQLite, a lightweight relational database, is used for data storage purposes

Connectivity

Android supports connectivity technologies including GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX

FEATURES

Messaging

SMS and MMS are available forms of messaging, including threaded text messaging and now Android Cloud To Device Messaging Framework(C2DM) is also a part of Android Push Messaging service

Multiple language support

Android supports multiple human languages. The number of languages more than doubled for the platform 2.3 Gingerbread

Web browser

The web browser available in Android is based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine.

FEATURES

Java support

While most Android applications are written in Java, there is no Java Virtual Machine in the platform and Java byte code is not executed. Java classes are compiled into Dalvik executables and run on Dalvik, a specialized virtual machine designed specifically for Android and optimized for battery-powered mobile devices with limited memory and CPU. J2ME support can be provided via third-party applications.

Media support

Android supports the following audio/video/still media formats: WebM, H.263, H.264 (in 3GP or MP4 container), MPEG-4 SP, AMR, AMR-WB (in 3GP container), AAC, HE-AAC (in MP4 or 3GP container), MP3, MIDI, Ogg Vorbis, FLAC, WAV, JPEG, PNG, GIF, BMP.

FEATURES

Streaming media support

RTP/RTSP streaming (3GPP PSS, ISMA), HTML progressive download (HTML5 <video> tag). Adobe Flash Streaming (RTMP) and HTTP Dynamic Streaming are supported by the Flash plugin. Apple HTTP Live Streaming is supported by RealPlayer for Mobile, and by the operating system in Android 3.0 (Honeycomb).

Additional hardware support

Android can use video/still cameras, touchscreens, GPS, accelerometers, gyroscopes, magnetometers, dedicated gaming controls, proximity and pressure sensors, thermometers, accelerated 2D bit blits (with hardware orientation, scaling, pixel format conversion) and accelerated 3D graphics.

FEATURES

Multi-touch

Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero. The feature was originally disabled at the kernel level (possibly to avoid infringing Apple's patents on touch-screen technology at the time). Google has since released an update for the Nexus One and the Motorola Droid which enables multi-touch natively.

Bluetooth

Supports A2DP, AVRCP, sending files (OPP), accessing the phone book (PBAP), voice dialing and sending contacts between phones. Keyboard, mouse and joystick (HID) support is available in Android 3.1+, and in earlier versions through manufacturer customizations and third-party applications.

FEATURES

Video calling

Android does not support native video calling, but some handsets have a custom version of the operating system that supports it, either via the UMTS network (like the Samsung Galaxy S) or over IP. Video calling through Google Talk is available in Android 2.3.4 and later. Gingerbread allows Nexus S to place Internet calls with a SIP account. This allows for enhanced VoIP dialing to other SIP accounts and even phone numbers. Skype 2.1 offers video calling in Android 2.3, including front camera support.

Multitasking

Multitasking of applications is available.

FEATURES

Voice based features

Google search through voice has been available since initial release. Voice action calling, texting, navigation, etc. are supported on Android 2.2 onwards.

Tethering

Android supports tethering, which allows a phone to be used as a wireless/wired Wi-Fi hotspot. Before Android 2.2 this was supported by third-party applications or manufacturer customizations.

Screen capture

Android does not support screenshot capture as of 2011. This is supported by manufacturer and third-party customizations. Screen Capture is available through a PC connection using the DDMS developer's tool.

APPLICATIONS

Applications are usually developed in the Java language using the Android Software Development Kit, but other development tools are available, including a Native Development Kit for applications or extensions in C or C++, and Google App Inventor, a visual environment for novice programmers.

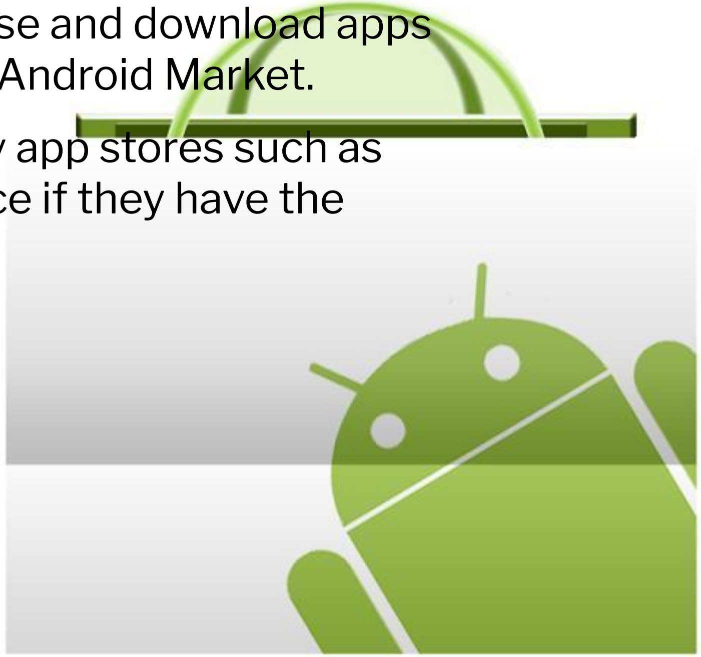
Android application package file (APK) is the file format used to distribute and install application software and middleware onto the Google Android operating system. To make an APK file, a program for Android is first compiled, and then all of its parts are packaged into one file. This holds all of that program's code such as (.dex files), resources, assets, certificates, and manifest file. As is the case with many file formats, APK files can have any name needed, but must end with the four character, three letter extension, .apk.

APK files are a variant of the JAR file format, with a .apk file extension.

ANDROID MARKET

Android Market is the online software store developed by Google for Android devices. An application program ("app") called "Market" is preinstalled on most Android devices and allows users to browse and download apps published by third-party developers, hosted on Android Market.

Alternatively, users can install apps from third party app stores such as the Amazon Appstore, or directly onto the device if they have the application's APK file.



SECURITY

Android is a multi-process system, in which each application (and parts of the system) runs in its own process. Most security between applications and the system is enforced at the process level through standard Linux facilities, such as user and group IDs that are assigned to applications.

Additional finer-grained security features are provided through a "permission" mechanism that enforces restrictions on the specific operations that a particular process can perform, and per-URI permissions for granting ad-hoc access to specific pieces of data.

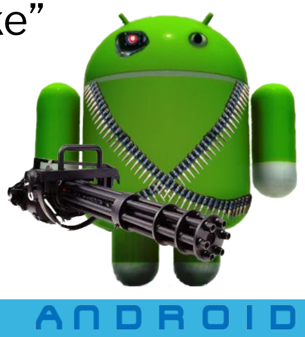
PRACTICE

PREPARING COMPUTER

1. Download the Android SDK from http://developer.android.com/
2. Install it and upgrade using SDK Manager

PREPARING DEVICE

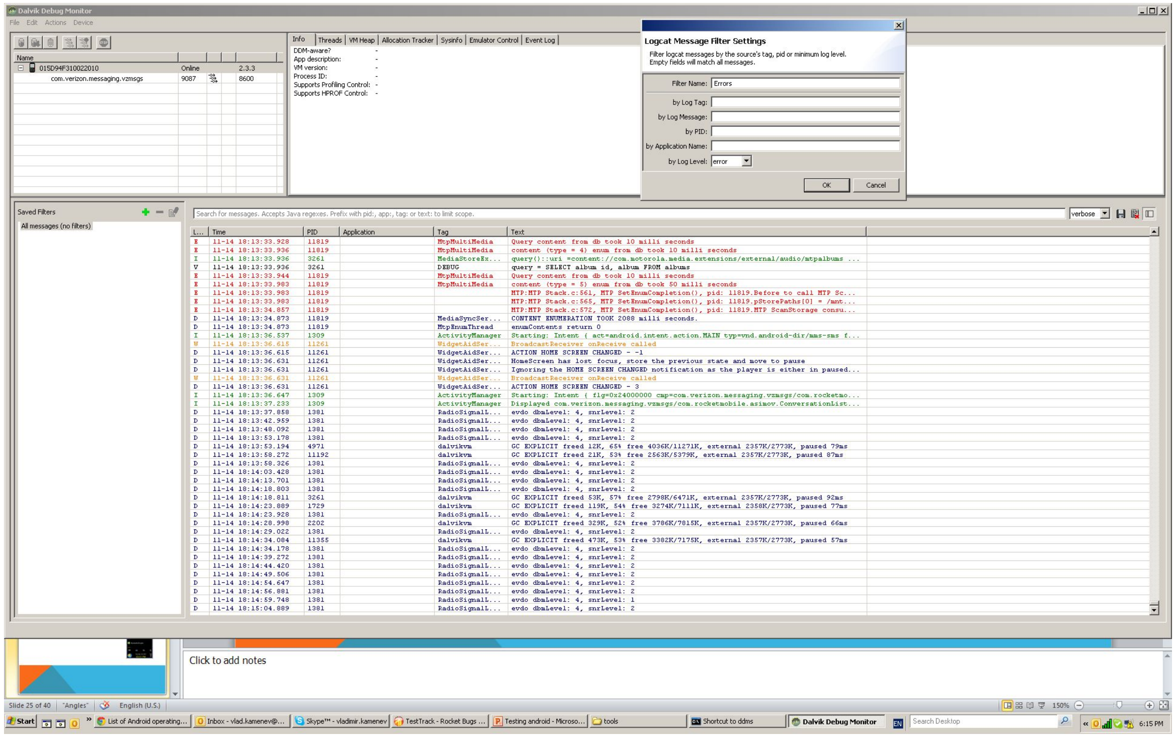
1. On computer side: Install USB driver to computer
2. On computer side: Use ddms tool (sdk_folder\Tools) to check connection
3. On device side: Settings ->Applications ->"Unknown sources" should be turned on
4. On device side: Settings ->Applications ->Development->"USB debugging" should be turned on
5. On device side: Settings ->Applications ->Development->"Allow mock locations" should be turned on
6. On device side: Settings ->Applications ->Development->"Stay awake" should be turned on



SCREEN CAPTURE

1. On computer side: launch ddms (SDK_Folder\Tools)
2. On device side: Connect device to computer using USB cable
3. On computer side: In Dalvik Debug Monitor, make sure that device was recognized and available
4. On computer side: In Dalvik Debug Monitor -> Press Ctrl+S to Screen Capture
5. On computer side: In “Device Screen Capture” dialog you could save screen to file in *.png format.

“DEVICE SCREEN CAPTURE” DIALOG


Dalvik Debug Monitor
File Edit Actions Device
Info | Threads | VM Heap | Allocation Tracker | Sysinfo | Emulator Control | Event Log
Saved Filters
All messages (no filters)
Click to add notes

Name			
015D94F310022010	Online		2.3.3
com.verizon.messaging.vzmsgs	9087		8600

DDM-aware? -
App description: -
VM version: -
Process ID: -
Supports Profiling Control: -
Supports HPROF Control: -

Logcat Message Filter Settings

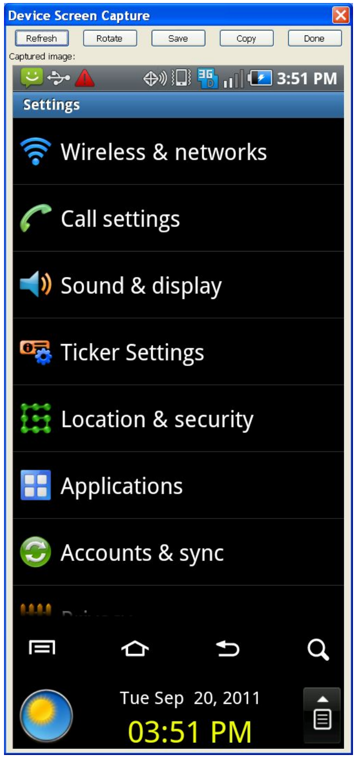
Filter logcat messages by the source's tag, pid or minimum log level.
Empty fields will match all messages.

Filter Name: Errors
by Log Tag:
by Log Message:
by PID:
by Application Name:
by Log Level: error

OK Cancel

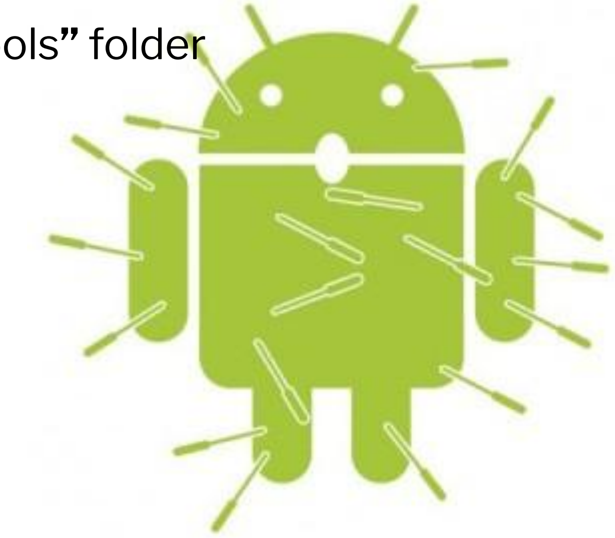
Search for messages. Accepts Java regexes. Prefix with pid:, app:, tag: or text: to limit scope.

```
L... Time                PID    Application  Tag                Text
E    11-14 18:13:33.928  11819               MtpMultiMedia      Query content from db took 10 milli seconds
E    11-14 18:13:33.936  11819               MtpMultiMedia      content (type = 4) enum from db took 10 milli seconds
I    11-14 18:13:33.936  3261                MediaStoreEx...    query()::uri =content://com.motorola.media.extensions/external/audio/mtpalbums ...
V    11-14 18:13:33.936  3261                DEBUG              query = SELECT album id, album FROM albums
E    11-14 18:13:33.944  11819               MtpMultiMedia      Query content from db took 10 milli seconds
E    11-14 18:13:33.983  11819               MtpMultiMedia      content (type = 5) enum from db took 50 milli seconds
E    11-14 18:13:33.983  11819                                  MTP:MTP Stack.c:561, MTP SetEnumCompletion(), pid: 11819.Before to call MTP Sc...
E    11-14 18:13:33.983  11819                                  MTP:MTP Stack.c:565, MTP SetEnumCompletion(), pid: 11819.pStorePaths[0] = /mnt...
E    11-14 18:13:34.857  11819                                  MTP:MTP Stack.c:572, MTP SetEnumCompletion(), pid: 11819.MTP ScanStorage consu...
D    11-14 18:13:34.873  11819               MediaSyncSer...    CONTENT ENUMERATION TOOK 2088 milli seconds.
D    11-14 18:13:34.873  11819               MtpEnumThread      enumContents return 0
I    11-14 18:13:36.537  1309                ActivityManager    Starting: Intent { act=android.intent.action.MAIN typ=vnd.android-dir/mms-sms f...
W    11-14 18:13:36.615  11261               WidgetAidSer...    BroadcastReceiver onReceive called
D    11-14 18:13:36.615  11261               WidgetAidSer...    ACTION HOME SCREEN CHANGED - -1
D    11-14 18:13:36.631  11261               WidgetAidSer...    HomeScreen has lost focus, store the previous state and move to pause
D    11-14 18:13:36.631  11261               WidgetAidSer...    Ignoring the HOME SCREEN CHANGED notification as the player is either in paused...
W    11-14 18:13:36.631  11261               WidgetAidSer...    BroadcastReceiver onReceive called
D    11-14 18:13:36.631  11261               WidgetAidSer...    ACTION HOME SCREEN CHANGED - 3
I    11-14 18:13:36.647  1309                ActivityManager    Starting: Intent { flg=0x24000000 cmp=com.verizon.messaging.vzmsgs/com.rocketmo...
I    11-14 18:13:37.233  1309                ActivityManager    Displayed com.verizon.messaging.vzmsgs/com.rocketmobile.asimov.ConversationList...
D    11-14 18:13:37.858  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:13:42.959  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:13:48.092  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:13:53.178  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:13:53.194  4971                dalvikvm           GC EXPLICIT freed 12K, 65% free 4036K/11271K, external 2357K/2773K, paused 79ms
D    11-14 18:13:58.272  11192               dalvikvm           GC EXPLICIT freed 21K, 53% free 2563K/5379K, external 2357K/2773K, paused 87ms
D    11-14 18:13:58.326  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:03.428  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:13.701  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:18.803  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:18.811  3261                dalvikvm           GC EXPLICIT freed 53K, 57% free 2798K/6471K, external 2357K/2773K, paused 92ms
D    11-14 18:14:23.889  1729                dalvikvm           GC EXPLICIT freed 119K, 54% free 3274K/7111K, external 2358K/2773K, paused 77ms
D    11-14 18:14:23.928  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:28.998  2202                dalvikvm           GC EXPLICIT freed 329K, 52% free 3786K/7815K, external 2357K/2773K, paused 66ms
D    11-14 18:14:29.022  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:34.084  11355               dalvikvm           GC EXPLICIT freed 473K, 53% free 3382K/7175K, external 2357K/2773K, paused 57ms
D    11-14 18:14:34.178  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:39.272  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:44.420  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:49.506  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:54.647  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:56.881  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
D    11-14 18:14:59.748  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 1
D    11-14 18:15:04.889  1381                RadioSignalL...    evdo dbmLevel: 4, snrLevel: 2
```



TAKING LOG (FOR CRASHES)

1. On computer side: Start ->Run ->”cmd” (run console)
2. In console type: “cd C:\SDK_Folder\platform-tools”
3. Type: adb logcat -d > log.txt
4. The log file would be in “C:\SDK_Folder\platform-tools” folder



TAKING TRACE LOG (FOR ANR)

1. In console type: “cd C:\SDK_Folder\platform-tools”
2. Type: adb pull /data/anr/traces.txt c:\temp\traces.txt
3. The log file would be in “C:\temp\” folder

INSTALLING APPLICATIONS

1. Place file, that should be installed to Device to “C:\SDK_Folder\platform-tools” folder
2. Start ->Run ->”cmd” (run console)
3. In console type: “cd C:\SDK_Folder\platform-tools”
4. Type: adb install application_name.apk

UNINSTALLING APPLICATIONS

1. On device side: Settings -> Applications -> Manage Applications -> “Select application to uninstall” -> Press “Uninstall” button

TESTING TIPS FOR NEWCOMERS

Layout:

Rotate device – Portrait and landscape mode – it is two different UI. Keep looking on application while rotating. For devices that support multiple orientations, Android detects a change in orientation when the user turns the device so that the display is "landscape" (long edge is horizontal) instead of "portrait" (long edge is vertical). When Android detects a change in orientation, its default behavior is to destroy and then re-start the foreground Activity. You should consider testing the following: Is the screen re-drawn correctly? Any custom UI code you have should handle changes in the orientation. Does the application maintain its state? The Activity should not lose anything that the user has already entered into the UI. The application should not "forget" its place in the current transaction.

Connectivity:

Check application in fair service conditions, check in airplane mode. Use cookie cans to decrease signal strength or Hoffman box

Messaging:

Try interruption tests

Multiple language support:

Check translation on Portrait and Landscape mode

TESTING TIPS FOR NEWCOMERS

Change in configuration

A situation that is more general than a change in orientation is a change in the device's configuration, such as a change in the availability of a keyboard or a change in system language.

A change in configuration also triggers the default behavior of destroying and then restarting the foreground Activity. Besides testing that the application maintains the UI and its transaction state, you should also test that the application updates itself to respond correctly to the new configuration.

Dependence on external resources

If your application depends on network access, SMS, Bluetooth, or GPS, then you should test what happens when the resource or resources are not available.

For example, if your application uses the network, it can notify the user if access is unavailable, or disable network-related features, or do both. For GPS, it can switch to IP-based location awareness. It can also wait for WiFi access before doing large data transfers, since WiFi transfers maximize battery usage compared to transfers over 3G or EDGE.

HARDWARE DIFFERENCES

Screens : Android devices are categorized into small, normal and large screens and low-, medium-, and high –pixel density.

Keyboards: virtual and physical keyboards

ANDROID 4.0 ICE CREAM SANDWICI

Ice Cream Sandwich was previewed at the May 2011 Google I/O event, and officially launched at the Galaxy Nexus and Ice Cream Sandwich release event on 19 October 2011.The SDK for Android 4.0 was publicly released on 19 October 2011.Gabe Cohen from Google said that ICS was "theoretically compatible" with any Android 2.3.x device in production at that time. The source code for this version is scheduled to be released soon after the Galaxy Nexus ships in November 2011.

FEATURES

Virtual buttons in the UI, in place of capacitive or physical buttons

Separation of widgets in a new tab, listed in a similar list to apps

Easier-to-create folders, with a drag-and-drop style

A customizable launcher

Improved visual voicemail with the ability to speed up or slow down voicemail messages

Pinch-to-zoom functionality in the calendar

Offline search, a two-line preview, and new action bar at the bottom of the Gmail app

Ability to swipe left or right to switch between Gmail conversations

Integrated screenshot capture (accomplished by holding down the Power and Volume-Down buttons)

FEATURES

Improved error correction on the keyboard

Ability to access apps directly from lock screen (similar to the HTC Sense 3.x)

Improved copy and paste functionality

Better voice integration and continuous, real-time speech to text dictation

Face Unlock, a feature that allows users to unlock handsets using facial recognition software

New tabbed web browser, allowing up to 16 tabs

Automatic syncing of browser with users' Chrome bookmarks

Modern Roboto font

Data Usage section in settings that lets users set warnings when they approach a certain usage limit, and disable data when the limit is exceeded

Ability to shut down apps that are using data in the background

Improved camera app with zero shutter lag, time lapse settings, panorama mode, and the ability to zoom while recording

FEATURES

Built-in photo editor

New gallery layout, organized by location and person

Refreshed 'People' app with social network integration, status updates and hi-res images

Android Beam, a NFC feature that lets user exchange web bookmarks, contact info, directions, YouTube, etc.

Hardware acceleration of the UI

Resizeable widgets, already part of 3.1 but new for cellphones

Wi-Fi Direct

1080p video recording for stock android

AUTOMATION USING ROBOTIUM

http://code.google.com/p/robotium/

Robotium is a test framework created to make it easy to write powerful and robust automatic black-box test cases for Android applications. With the support of Robotium, test case developers can write function, system and acceptance test scenarios, spanning multiple Android activities.

Robotium has full support for Activities, Dialogs, Toasts, Menus and Context Menus

The Android testing framework is based on the JUnit 3.

BENEFITS

ANDROID PLATFORM LIMITATIONS

Not able to handle multiple activities

Test execution pretty slow

Test cases are complex to implement

ROBOTIUM BENEFITS

You can develop powerful test cases, with minimal knowledge of the application under test.

The framework handles multiple Android activities automatically.

Minimal time needed to write solid test cases.

Readability of test cases is greatly improved, compared to standard instrumentation tests.

Test cases are more robust due to the run-time binding to GUI components.

Blazing fast test case execution.

Integrates smoothly with Maven or Ant to run tests as part of continuous integration.

LIMITATIONS

- Robotium does not support flash and web based applications.
- Robotium does not work on Canvas.

Any questions?

www.google.com