Instructions: for Android Practical file

* Experiment number and name must be specified for each experiment, with Font Size 14. Must be bold and underlined.
* Heading must be bold and underlined. Sub-heading Font Size 12.
* Regular Font Size : 12
* Font : Times New Roman
* Figure no. must be specified for each screenshot.
* Print will be taken in one-side format.
* Attach experiments in Stick file, NO sessional file.

**Lakshmi Narain College of Technology Bhopal (M.P.)**

**Department of Information Technology**

**Practical File**

IT-606 ANDROID PROGRAMMING

**Name: Manas Khare**

**Semester: VI**

**Section: A**

**Roll No: 55**

**Enroll No: 0103IT181055**

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| EXPERIMENT  NO. | EXPERIMENT TITLE |  | SIGNATURE |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Experiment – 1**

**Introduction of Android**

**Introduction**

Android is a Linux based operating system. It is designed primarily for touchscreen mobile devices such as smartphones and tablet computers. The operating system has developed a lot in the last 15 years starting from black and white phones to recent smartphones or mini computers. One of the most widely used mobile OS these days is android. The android is software that was founded in Palo Alto of California in 2003.

It is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance (OHA), led by Google, and other companies.

It offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.

The history and versions of android are interesting to know. The code names of android ranges from A to J currently, such as Aestro, Blender, Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwich, Jelly Bean, KitKat and Lollipop. Let's understand the android history in a sequence:

* Initially, Andy Rubin founded Android Incorporation in Palo Alto, California, United States in October, 2003.
* In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.
* The key employees of Android Incorporation are Andy Rubin, Rich Miner, Chris White and Nick Sears.
* Originally intended for camera but shifted to smart phones later because of low market for camera only.
* Android is the nick name of Andy Rubin given by co-workers because of his love to robots.
* In 2007, Google announces the development of android OS.
* In 2008, HTC launched the first android mobile.

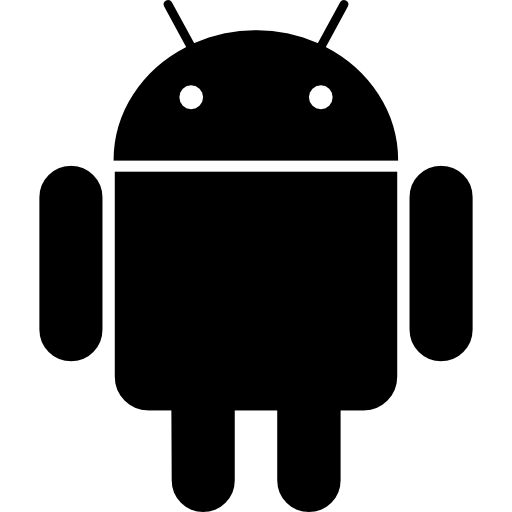


Fig. 1: Logos of Android.

**Features**

Android is a powerful operating system and supports great features like-

* Open Platform

Android provides the open platform facility. Android application can be developed using any kind of operating system.

* Compatibility

Android provides cross-platform means android application can run on any type screen, size and resolutions including mobile phones, tablets etc. Google provides the facility to run the application only on the compatible devices such as mobile phones or tablets.

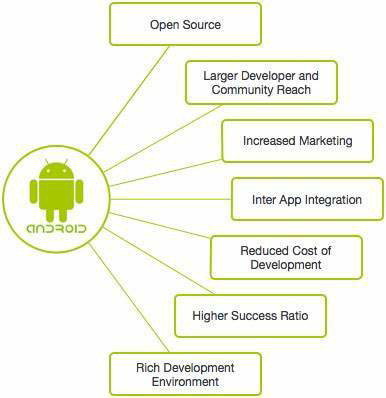


Fig.2 : Some features of Android.

* Beautiful UI

Android OS basic screen provides a beautiful and intuitive user interface.

* Connectivity

GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.

* Storage

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

* Media support

H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP are used.

* Messaging

SMS and MMS

* Web browser

Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.

* Multi-touch

Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.

* Multi-tasking

User can jump from one task to another and same time various application can run simultaneously.

* Resizable widgets

Widgets are resizable, so users can expand them to show more content or shrink them to save space.

* Multi-Language

Supports single direction and bi-directional text.

* GCM

Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.

* Wi-Fi Direct

A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.

* Android Beam

A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together.

**Applications**

* Foreground applications

Games are the best example. In this category applications are useful only when they are in the foreground. They are alive when they are visible, otherwise they are suspended.

* Background applications

These applications don’t have user interaction or they have limited interactivity. Best example is the alarm clock on your device. These are very uncommon apps.

* Intermediate applications

These applications are biased towards one of the above categories of apps.

There are few apps which have limited interactivity and do most of the work in background by remaining hidden.The example is an mp3 music player. Another category will involve lots of user interactions and does some part of computation in the background. Best example is a news app.

* Widgets and Live Wallpaper:

Some applications are designed only for the home screen.

There are many android applications in the market. The top categories are –

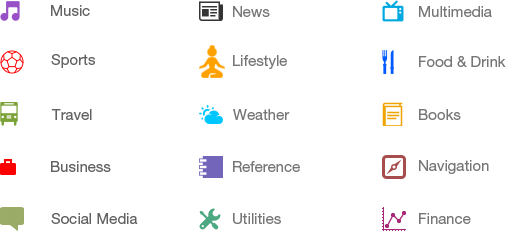


Fig.3: Categories of Android applications.

**Architecture**

Android architecture is categorized into five parts:

1. Linux kernel
2. Native libraries (middleware),
3. Android Runtime
4. Application Framework
5. Applications.

Android operating system is a stack of software components which is roughly divided into five sections and four main layers.

1. Linux Kernel

It is the heart of android architecture that exists at the root of android architecture. Linux kernel is responsible for device drivers, power management, memory management, device management and resource access.

1. Native Libraries

On the top of Linux kernel, there are Native libraries such as Web Kit, OpenGL, Free Type, SQLite, Media, C runtime library (libc) etc.

The Web Kit library is responsible for browser support, SQLite is for database, Free Type for font support, Media for playing and recording audio and video formats.

1. Android Runtime

In android runtime, there are core libraries and DVM (Dalvik Virtual Machine) which is responsible to run android application. DVM is like JVM but it is optimized for mobile devices. It consumes less memory and provides fast performance.

1. Application Framework

On the top of Native libraries and android runtime, there is android framework. Android framework includes Android API's such as UI (User Interface), telephony, resources, locations, Content Providers (data) and package managers. It provides a lot of classes and interfaces for android application development.

1. Applications

On the top of android framework, there are applications. All applications such as home, contact, settings, games, browsers are using android framework that uses android runtime and libraries. Android runtime and native libraries are using Linux kernel.

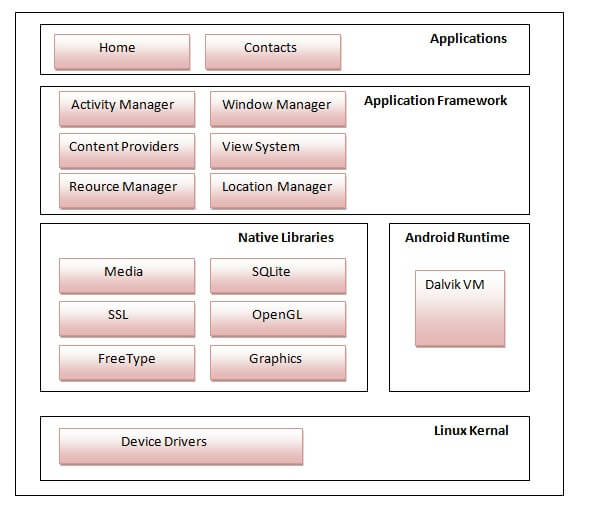


Fig.4: Android Architecture.

**Dalvik Virtual Machine (DVM)**

The Dalvik Virtual Machine (DVM) is an android virtual machine optimized for mobile devices. It optimizes the virtual machine for memory, battery life and performance. Dalvik is a name of a town in Iceland. The Dalvik VM was written by Dan Bornstein.

Dalvik Virtual Machine is a Register-Based virtual machine. It was designed and written by Dan Bornstein with contributions of other Google engineers as part of the Android mobile phone platform. The Dalvik virtual machine was named after Bornstein after the fishing village “Dalvík” in Eyjafjörður, Iceland, where some of his ancestors used to live.

Working - The Java Compiler(javac) converts the Java Source Code into Java Byte-Code(.class). Then DEX Compiler converts this (.class) file into in Dalvik Byte Code i.e. “.dex” file.

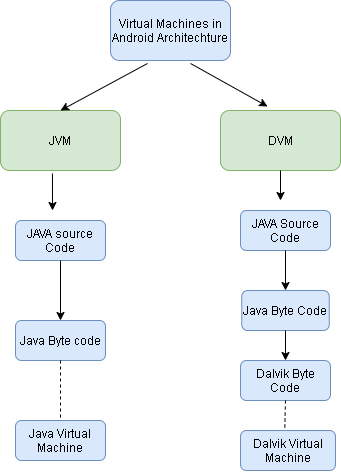


Fig.5: Working of DVM.

Advantages-

* DVM supports the Android operating system only.
* In DVM executable is APK.
* Execution is faster.
* From Android 2.2 SDK Dalvik has its own JIT (Just in Time) compiler.
* DVM has been designed so that a device can run multiple instances of the Virtual Machine effectively.
* Applications are given their own instances.

**Activity Life Cycle**

Android system initiates its program with in an Activity starting with a call on onCreate() callback method.

There is a sequence of callback methods that start up an activity and a sequence of callback methods that tear down an activity as shown in the below Activity life cycle diagram:

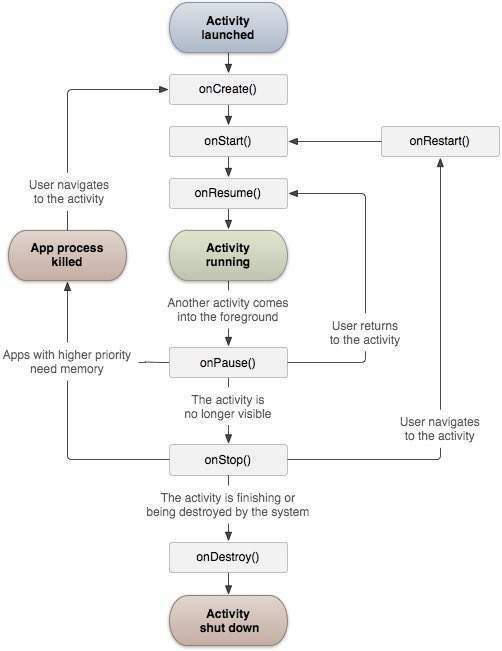


Fig.6: Activity Lifecycle of Android.

|  |  |
| --- | --- |
| S.no. | Description of Activities |
| 1 | onCreate()  This is the first callback and called when the activity is first created. |
|  |  |
| 2 | onStart()  This callback is called when the activity becomes visible to the user. |
| 3 | onResume()  This is called when the user starts interacting with the application. |
| 4 | onPause()  The paused activity does not receive user input and cannot execute any code and called when the current activity is being paused and the previous activity is being resumed. |
| 5 | onStop()  This callback is called when the activity is no longer visible. |
| 6 | onDestroy()  This callback is called before the activity is destroyed by the system. |
| 7 | onRestart()  This callback is called when the activity restarts after stopping it. |

Table.1

**How to convert java files into .apk**

.jar files and .apk files are different. They are not same. Jar file is a component of apk, but its not apk. What you can do is, download android studio (because that is where you’ll create an actual apk file) and import your jar file as a library. Using the methods in the library, you can build an app that uses those methods. That would be an apk file. This apk uses your jar file, and this is how you use your jar file in your android project.