**What is Android?**

ANDROID is a **open-source**  Software platform and Operating System based on Linux Kernel with java programming interface for developing the mobile devices, Tablets, Wears, TV’s and Auto’s.

The **Android SDK** provides all necessary tools and APIs to develop the applications. This includes a system, debugger and device emulator as well as its own virtual machine to run android programs.

**Origin of Android:**

Android was founded in Palo Alto, California in October 2003 by Andy Rubin, Rich Miner, Nick Sears and Chris White who work at “***Google***” to develop, in Rubin’s words “..Smarter mobile devices those are more aware of its owner’s location and preferences.”

******

Android was bought by “***Google****”* in 2005

On the 5th of November 2007 the Open Handset Alliance (**OHA**), a consortium of several companies was unveiled with the goal to develop open standards for mobile devices.



**Versions, Version Names and API Levels of Android:**

**Android 1.0**, the first commercial version of the software, was released on September 23, 2008.

HTC dream first commercially sold out mobile which runs on android on OCT 22, 2008.



**Android 1.1** on February 9, 2009 update was released, initially for the HTC Dream only. Android 1.1 was known as “Petit Four” internally, though this name was not used officially. The update resolved bugs, changed the API and added a number of features.

**Android 1.5** on April 30, 2009, the Android 1.5 update was released, based on Linux kernel 2.6.27. This was the first release to officially use a name based on dessert (“CupCake”).

**More Versions, Version Names and API levels**

****

**Android 3.0** Introduced Tabs.

**Android 4.0** Introduced Tabs and Devices.

**Android 4.2.2** Introduced Android Wears.

**Android 5.0** has done Major changes in User Interface and Runtime Environment.

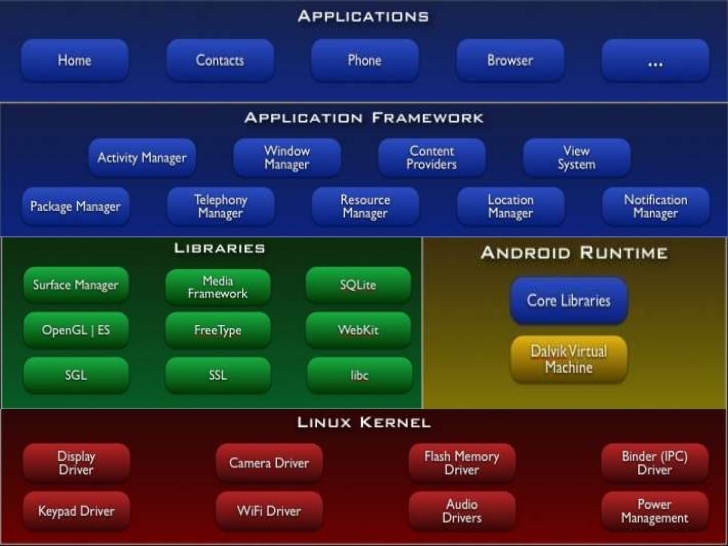
**Android Architecture**

The Android software stack is composed of the elements a Linux Kernel and collection of c/c++ libraries are exposed through an application framework that provides services for and management of, runtime and applications.

It is layered Architecture facilities rapid development of application.

**The Software Stack is split into Four Layers:**

* Linux Kernel.
* Libraries and Android Runtime.
* Application Framework.
* Application Layer.

****

**Android Architecture**

**Linux kernel:**

Core services (including H/W Drivers, Process and Memory management, security, networking and Power management) are handled by a Linux kernel. The kernel is also provides abstraction Layer between hardware and the remainder stack.

**Libraries:**

Running on top of the kernel, Android includes various c/c++ core libraries. These libraries cannot be accessed directly. With the help of application framework, we can access these libraries. These are many libraries such as libc and ssl, as well as: -

1. A media library for play back of audio and video media.
2. A surface manager to provide display management.
3. Graphics libraries that includes SGL and OpenGL for 2D and 3D graphics.
4. SQLite for database support.
5. SSL and webkit for integrated web browser and Internet Security.

**Android Runtime:**

* The Android Runtime was designed specifically for Android to meet the needs of running in an embedded environment where you have limited battery, limited memory, limited CPU.
* **Core Libraries** contains all of the **Java libraries** (Collection classes, utilities, IO, all the utilities and tools) + **Android Specific Libraries** that you’ve come to expected to use.
* **Dalvik Virtual Machine** is a register – based virtual machine, as opposed to java virtual machine, which are stack based accordingly, it uses a completely different byte code than java, the android SDK includes DX tool to translate java byte code to Dalvik byte code that’s been optimized to ensure that a device an run multiple instances efficiently. It relies on the Linux kernel for threading and low-level memory management.

**Application Framework:**

The application framework is the tool kit that all applications can use.

It provides the classes used to create android Applications. It also provides a generic abstraction for hardware access and manages the user interface and application resources.

These are as follows:

**Activity Manager** It manages the lifecycle of applications. It enables proper management for the activities. All the activities are controlled by activity manager.

* **Resource Manager** it provides access to non-code resources such as graphics etc.
* **Notification manager** it enables all applications to display custom alerts in status bar.
* **Location** manager it fires alerts when user enters or leaves a specified geographical location.
* **Package manager** it is use to retrieve the data about installed packages on device.
* **Window Manager** it is use to create views and layouts.
* **Telephony manager** it is use to handle setting of network connection and all information about services on device.

**Application Layer:**

The final layer on top is Applications

All Applications, both Native and Third – party are built-in application layer by means of the same API libraries such as Home, contacts and browser app’s.

Application Layer runs within the android runtime, using **the classes and services made available from the application framework**.

**Security**

Android is a multi-process system, in whicheach 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.

**Android Features**

Android itself supports the following features:

* Storage
* Application Framework enabling reuse and replacement of components
* DVM
* Integrated Browser based on the open - source webkit engine
* Optimized graphics
* Media support
* GSM telephony
* Bluetooth, EDGE, EG, Wifi (H/W dependent).
* Camera, GPS, Compass and Accelerometer
* Rich Development Environment.
* It is an open-source Free.
* Android Has Powerful APIS.
* Excellent Documentation.
* A thriving Developer Community.
* An Individual person can develop and publish
* Required Developer license only to publish not to develop.