



# 1. Android and It's Tools

**4 Marks**

**Prof. Mr. Ghanawajeer D.J.**



# Syllabus

**1.1 Introduction to Android: Open Handset alliance, Android Ecosystem**

**1.2 Need of Android: Features of Android**

**1.3 Tools & Software required for Developing an Android Application.**

**1.4 Android Architecture.**



# Course Outcome

- 1. Interpret features of Android Operating System.**



# Unit Outcomes

**1a.Explain the given basic terms related to Android System**

**1b.Explain with sketches Android Architecture for the given Application.**

**1c.Identify tools and software required for developing the given Android Application with justification.**

**1d.Explain significance of the given component in Android Architecture.**

# 1.1 Introduction

- ❖ Android 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*, led by Google, and other companies.
- ❖ based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets.
- ❖ Android 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.

# History of Android

- ❖ Android Inc. founded in Palo Alto, California, United States in October 2003 by Andy Rubin [co-founder of Danger], Rich Miner [co-founder of Wildfire Communication Inc.], Nick Sears [once VP at T-Mobile], and Chris White [Headed design and interface development at Web TV] to develop “smarter mobile devices that are more aware of its owner's location and preferences”.
- ❖ In July 2005, Google acquired Android Inc. for at least \$50 million.

# History of Android

Founder of Android Inc.



**Andy Rubin**



**Rich Miner**



**Nick Sears**



**Chris White**

# History of Android

- ❖ On November 5, 2007, the Open Handset Alliance, a consortium of technology companies including Google, HTC, Sony and Samsung and chipset makers such as Qualcomm unveiled itself, with a goal to develop open standards for mobile devices.
- ❖ That day, **ANDROID WAS UNVEILED AS ITS FIRST PRODUCT**, a mobile device platform built on the **LINUX KERNEL**.
- ❖ In September 2008 T-Mobile announced the availability of the T-Mobile G1, the first smartphone based on the Android Platform.



# History of Android

- ❖ The first commercially available smartphone running Android was the HTC Dream, released on October 22, 2008.



# History of Android

- ❖ A few days after that, Google announced the availability of Android SDK Release Candidate 1.0.
- ❖ In October 2008, Google made the source code of the Android Platform available under Apache's open source license.

# All Android Versions



# All Android Versions

## ❖ **Android 1.0 (2008) [Android Astro 1.0 ]**

- Beginning with that initial version of the operating system running on the HTC Dream, join us as we take a look at how Android has changed in the past decade.
- Copy and paste feature in the web browser is not present.
- Android 1.0 was obviously far less developed than the operating system we know and love today, but there are a few similarities.



# All Android Versions

- For example, most agreed that Android pretty much nailed how to deal with notifications, and it included the pull-down notification window that blew the notification system in iOS out of the water.



# All Android Versions

## ❖ **Android 1.5 Cupcake (2009)**

- The first major update to Android not only got a new version number, but it was the first to use Google's naming scheme, too.



# All Android Versions

- Cupcake was significant for a number of reasons, but the most important is probably that it was the first version of Android to have an on-screen keyboard — before that, manufacturers had to include physical keyboards on their devices.



# All Android Versions

## ❖ Android 1.6 Donut (2009)

- Android Donut gave users a pretty big update — a much bigger update than the 0.1 version number increase suggests. For example, Donut brought Android to millions of people by adding support for CDMA networks such as Verizon, Sprint, and a number of big networks in Asia.





# All Android Versions

- Donut was really targeted at making Android more user-friendly, but some of the biggest updates were under the hood.
- For example, Donut was the first version of Android to support different screen sizes, meaning that manufacturers could create devices with the display sizes they wanted and still run Android.



# All Android Versions

## ❖ Android 2.0/2.1 Eclair (2009)

- Eclair was the first device to feature Google Maps navigation, for instance, kicking off what soon became the death of the in-car GPS unit.
- While Maps has changed a lot since then, a few important features showed up in the service that are still present today, such as turn-by-turn navigation and voice guidance.



# All Android Versions

- The internet browser in Android Eclair also got revamped for the new operating system.
- Google added HTML5 support to the browser and the ability to play videos, putting Eclair as par with the ultimate mobile internet machine at the time — the iPhone.



# All Android Versions

## ❖ Android 2.2 Froyo (2010)

- Android Froyo was first released in 2010, and proved why it was an advantage to have a Nexus phone.
- Froyo was aimed more at refining the Android experience, offering users five home screen panels instead of three, and showing off a redesigned Gallery app.



# All Android Versions

- Froyo was the first version of Android to bring mobile hot spot support.
- Users also finally got the PIN lock screen, which was perfect for those who didn't really like the pattern lock screen that was previously offered in Android.



# All Android Versions

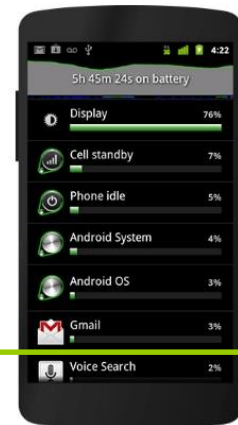
## ❖ **Android 2.3 Gingerbread (2010)**

- Google chose the Samsung-built Nexus S for this one, however, a phone that was derived from Samsung's highly-successful Galaxy S. Gingerbread was another big Android refinement, and it saw a redesign of Android's stock widgets and home screen.



# All Android Versions

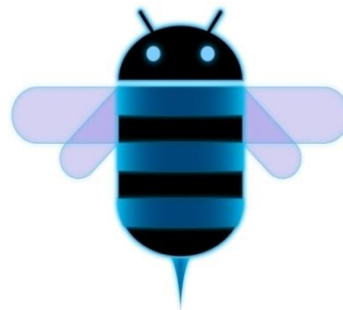
- Gingerbread also came with an improved keyboard, which offered new coloration for the keys, as well as improved multi-touch support, which allowed users to press multiple keys to access a secondary keyboard.
- Last but not least is that Gingerbread added support for the front-facing camera.



# All Android Versions

## ❖ Android 3.0 Honeycomb (2011)

- Google had been making waves in the smartphone industry for a few years now, which made Honeycomb a very interesting release purely because it was targeted at tablets. It was even first showcased on a Motorola device that would eventually become the Xoom.





# All Android Versions

- Honeycomb provided a few design cues as to what would appear in future versions of Android.
- Instead of accenting the operating system with the classic green Android color, for example, Google switched to blue accents.



# All Android Versions

- Honeycomb was the fact that it removed the need for the physical button. Instead, the home, back, and menu buttons were all included in the software as virtual buttons, meaning they could be hidden or shown based on the application.

# All Android Versions

## ❖ Android 4.0 Ice Cream Sandwich (2011)

- The Nexus S was a great phone, but it wasn't the be all and end all of Google's partnership with Samsung. The two paired up once again for the release of the Galaxy Nexus, which showcased Ice Cream Sandwich, an operating system that brought many of Honeycomb's features over to the smartphone.



# All Android Versions

- The operating system brought over the aforementioned virtual buttons, as well as the tweaked and refined interface that made use of the blue highlights.
- Other small features, such as face unlock, data usage analysis, and new apps for mail and calendar, were also included in the update.



# All Android Versions

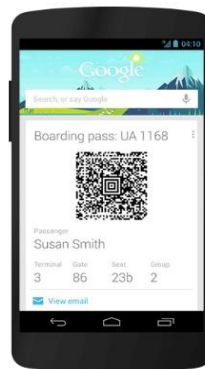
## ❖ Android 4.1 Jelly Bean (2012)

- Android Jelly Bean signaled a new era for the operating system, even if the OS seemed more or less the same as its predecessor.
- The most important of which was Google Now, which could be accessed with a quick swipe from the home screen and brought information — i.e. calendar events, emails, weather reports — all to a single screen.



# All Android Versions

- Apart from Google Now, a number of other important additions were implemented in Jelly Bean, such as Project Butter, which was aimed at drastically improving Android's touch performance by tripling buffering graphics.
- This eliminated a lot of the stutter in Android and made it a much smoother experience overall.



# All Android Versions

## ❖ Android 4.4 KitKat (2013)

- The launch of Android 4.4 KitKat coincided with the launch of the Nexus 5, and it came with a number of great features.
- For example, KitKat represented one of the biggest aesthetic changes to the operating system to date, modernizing the look of Android.



# All Android Versions

- The blue accents found in Ice Cream Sandwich and Jellybean were replaced with a more refined white accent, and a number of the stock apps that came with Android were redesigned to show lighter color schemes.
- It also brought a new phone dialer, full-screen apps, and a new Hangouts app, which offered SMS support along with support for the Hangouts messaging platform.



# All Android Versions



# All Android Versions

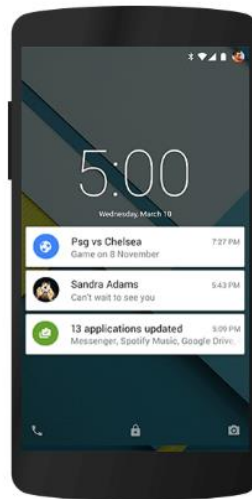
## ❖ Android 5.0 Lollipop (2014)

- Android Lollipop, which debuted alongside the Nexus 6, was the first to feature Google's "Material Design" philosophy.
- Google replaced the aging Dalvik VM with Android Runtime, for example, which boasted ahead-of-time compilation.



# All Android Versions

- Android 5.0 also saw the addition of another version of Android, dubbed Android TV, which brought Android to the big screen and is still in use on plenty of TVs today.



# All Android Versions

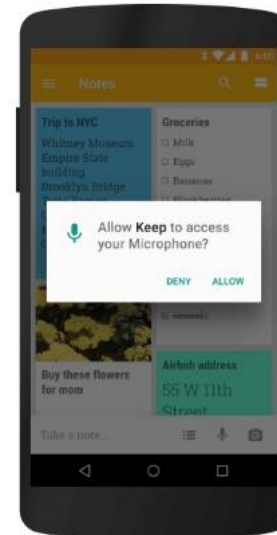
## ❖ **Android 6.0 Marshmallow (2015)**

- Android Marshmallow brought about both design changes and changes under the hood.
- Most notably, the app menu almost completely changed. Google used a white background instead of black, for instance, and added a search bar to help users quickly find the app they needed.



# All Android Versions

- In Marshmallow, you got access to a more comprehensive set of volume controls, allowing you to change the volume for the device, media, and alarms.
- Security also got a pretty big boost within the operating system.



# All Android Versions

## ❖ Android 7.0 Nougat (2016)

- Android 7.0 Nougat arguably marked one of the biggest upgrades to Android in its 10 years largely because of how smart the operating system got.
- In fact, perhaps the biggest change to Android in Nougat is that Google Now was replaced with the now much-respected Google Assistant.



# All Android Versions

- Notifications were presented from screen to screen, and unlike previous iterations of Android, they could be grouped together for easy management.
- Multitasking also got a boost with Nougat. Whether you're using a phone or a tablet, you'll be able to use split-screen mode, allowing you to use two apps at once without having to exit out of each app every few minutes..

# All Android Versions





# All Android Versions

## ❖ Android 8.0 Oreo (2017)

- Android Oreo took the Android platform to version 8.0, and in particular brought a ton of multitasking features.
- Picture-in-picture and native split-screen both made their debuts in Android Oreo, meaning you could continue watching your favorite show on Netflix while browsing the web.



# All Android Versions

- Android Oreo also gave us a whole lot more control over notifications.
- With Oreo, users were given the ability to turn notification channels on or off, meaning you could get super granular with which notifications show up and what happens when they appear.
- A few other smaller features showed up in Oreo, too. For example, Google did away with the blob style for emojis, replacing them with emojis that were a little more in line with other platforms.

# All Android Versions

- Oreo also gave us auto-enable Wi-Fi, a smart text selector, and so on.



# All Android Versions

## ❖ Android 9.0 Pie (2018)

- Android Pie brought with it a number of visual changes that made it the most significant update to Android in a few years.
- Android 9.0 Pie also brought some changes to notifications, including extra control over the types of notifications that show up and where they show up, as well as Google's "Digital Wellbeing," a feature that essentially tells you how often you use your phone, the apps that you use the most.



# All Android Versions

- Other features include adaptive battery, which limits how much battery background apps can use, as well as “App Actions,” which are deep-links to certain app features that show up straight from the app drawer.



# All Android Versions

## ❖ Android 10 (2019)

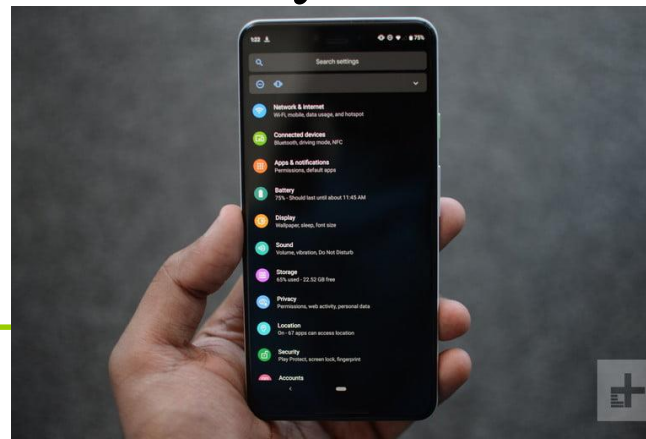
You might notice that there's no sugary name for this version of Android. That's no mistake — along with the rollout of the latest version of Android, Google also announced a rebranding of the operating system, doing away with the naming scheme and instead sticking with version numbers only.



android

# All Android Versions

- Android 10 marked the end of the Android navigation buttons. While Android 9 kept the back button, Android 10 now uses gestures instead.
- Android 10 also brought a Systemwide dark mode, which makes things a little easier on the eyes and could save battery too.



# All Android Versions

## ❖ Android 11 (2020)

Android 11 operating system is the eleventh big release of Android. It is the 18th version of Android mobile OS, which was released on 8 September 2020.

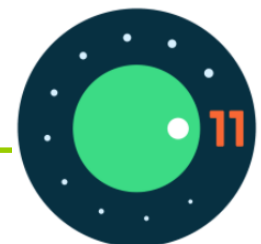
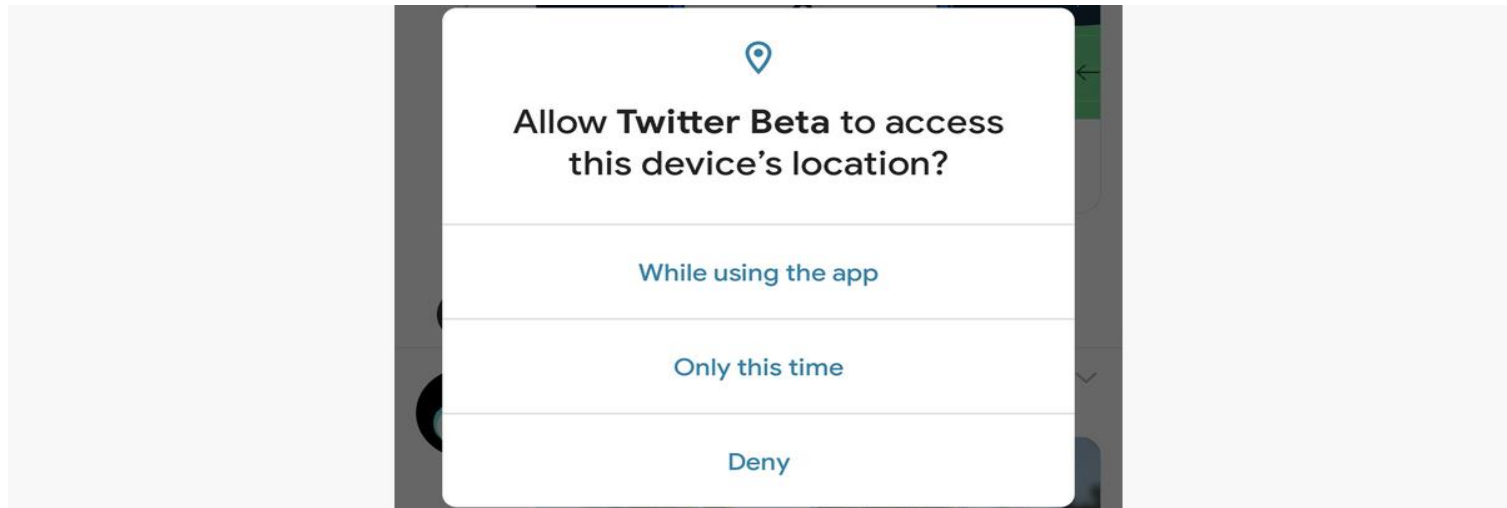
The update builds upon the expanded permissions system introduced in Android 10 and adds in the ability for users to grant apps certain permissions those related to location access, camera access, and microphone access only.





# All Android Versions

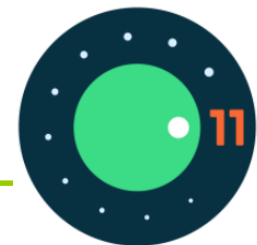
## ❖ Android 11 (2020)



# All Android Versions

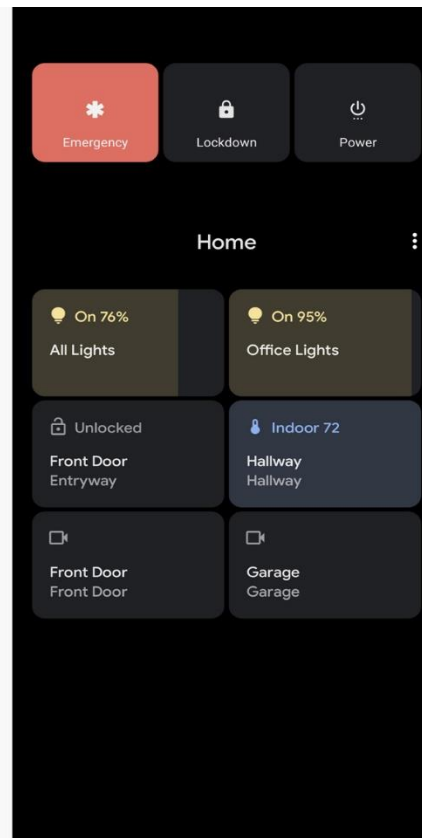
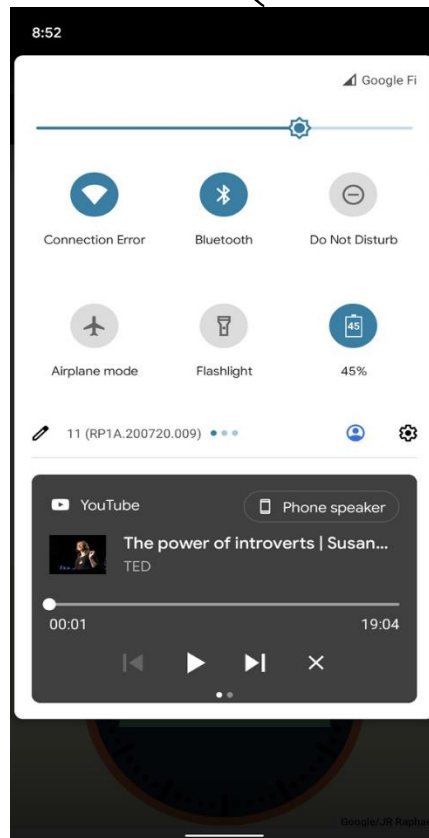
## ❖ Android 11 (2020)

Android 11 refines the system notification area to emphasize and simplify conversation-related alerts; it introduces a new streamlined media player that contains controls for all audio- and video-playing apps in a single space; and it adds in a new contextual menu of connected-device controls for any smart products associated with your account.



# All Android Versions

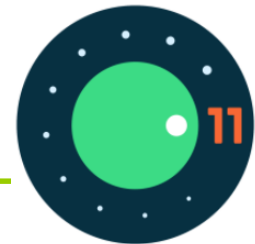
## ❖ Android 11 (2020)



# All Android Versions

## ❖ Android 11 (2020)

Android 11 has plenty of other small but significant improvements —including a new Notification History section, a native screen recording function, and an automated scheduling system for the system-wide Dark Theme.



# Open Handset Alliance

- ❖ The Open Handset Alliance (OHA) is a consortium whose goal is to develop open standards for mobile devices, promote innovation in mobile phones and provide a better experience for consumers at a lower cost.
- ❖ The Open Handset Alliance is made up of group of 84 technology and mobile companies including telecom-related entities, wireless carriers, semiconductor companies, handset manufacturers and software companies.

# Open Handset Alliance

- ❖ However, several major wireless companies and manufacturers are absent from the coalition, including Nokia, Symbian, Apple, RIM, Microsoft, Verizon and Cingular.
- ❖ The OHA introduced a mobile device operating system called AndroidSymbian operating system.

# Open Handset Alliance

## Mobile Operators



open handset alliance

## Handset Manufacturers



## Semiconductor Companies



## Software Companies



## Commercialization Companies



# Open Handset Alliance

Operators	Software Co.	Commercializat.	Semiconductor	Handset Manf
China Mobile	Ascender Corp.	Aplix	Audience	ACER
China Unicom	eBay	Noser Engineering	Broadcom Corp.	ASUS
KDDI Corp.	Esmertec	Astonishing Tribe	Intel Corp.	HTC
NTT DoCoMo	Google	Wind River Systems	Marvell Tech.	LG
Sprint Nextel	LivingImage	Omron Software	Group	Motorola
T-Mobile	NMS Comm.	...	Nvidia Corp.	Samsung
Telecom Italia	Nuance Comm.	Teleca	Qualcomm	ASUSTek
Telefónica	PacketVideo		SiRF Tech. Holdings	Garmin
Vodafone	SkyPop		Synaptics	Huawei Tech
Softbank	SONIVOX		Texas Instr.	LG
...	...		AKM Semicond.	Samsung
Ericsson	Borqs		ARM	...
			Atheros Comm	Sony Ericsson
			...	Toshiba
			EMP	





HTC G1,  
Droid,  
Tattoo



Motorola Droid (X)



Suno S880



Samsung Galaxy



Sony Ericsson

# Tablets



Velocity Micro Cruz



Gome FlyTouch



Acer beTouch



Dawa D7



Toshiba Android  
SmartBook



Cisco Android Tablet



# Android Eco-System

- ❖ An open source platform for mobile, embedded and wearable devices.
- ❖ Google is the principle maintainer.
- ❖ Other companies contribute to the system.
- ❖ Each device manufacturer can customize Android to suite their needs.

# Android Eco-System

- ❖ **Android Ecosystem** is nothing but the relationship between Users, Developers/ Programmers, and Hardware equipment makers, the Android ecosystem is nothing but the mutual dependence between Users, Developers, and equipment makers. They are dependent of each other so one cannot exist without the other.



# Android Eco-System

❖ The main block of the android ecosystem is:

- **Android User**
- **Developer**
- **Equipment Maker**

# Android Eco-System

## ❑ **Android User (Users buy handsets and software Application):**

- Android users have more space for customizability for their android devices.
- Android users are smarter than other users and they are perceived to have greater levels of support.

# Android Eco-System

## ❑ **Android User (Users buy handsets and software Application):**

- Android users are also more likely to prefer saving their cost and love the openness of the platform also they like to customize their device.
- Android users are fancier to prefer saving money and also android user like customizing their android handset/device.

# Android Eco-System

## ❑ Developers (sell Application):

- Android Developers are the professional software developer in designing applications as well as developing applications for Android.
- Some of the following tasks where an android developer can play his role in the development of android apps:
  - Design and build advanced applications for the android platform.





# Android Eco-System

## ❑ Developers (sell Application):

- Collaborate and define with development teams for design and deliver new cool features.
- Troubleshoot and fix bugs in new and existing applications for Users.
- Evaluate and implement new development tools to work with outside data sources and APIs.

# Android Eco-System

## ❑ Equipment Maker:

➤ Android equipment's are available in the market in a huge amount.

### ➤ Smartwatches:

- A smartwatch is a handheld, wearable device that closely relates a wristwatch or other time device.
- In addition to telling time, many smartwatches are wireless connectivity oriented such as Bluetooth capable.

# Android Eco-System

## ❑ Equipment Maker:

### ➤ Smartwatches:

- The traditional watch becomes, in effect, a wireless Bluetooth technology extending the capabilities of the wearer's smartphone to the watch.

### ➤ Smart TV:

- An Android TV box is a small computer that plugs into any TV and gives the user the ability to stream content, locally and online.

# Android Eco-System

## ❑ Equipment Maker:

### ➤ Smart TV:

- Apps can be downloaded from the Google Play Store, installed, and do most anything a standard computer can do from streaming videos to writing an email.

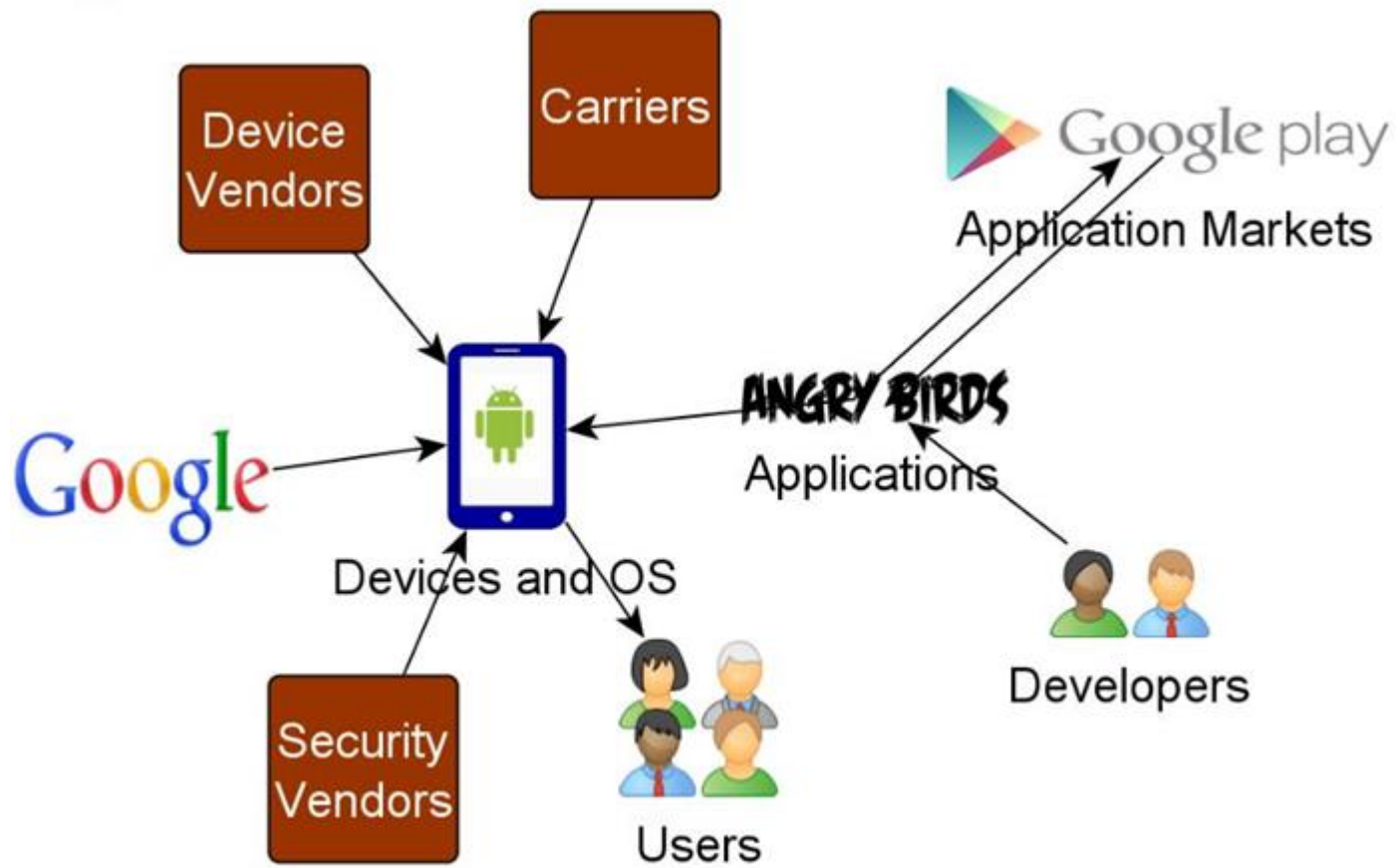
### ➤ Smart Speakers: Smart speakers are booming in the market now, Smart speakers like Google Home, Alexa, We can control our android device via voice using these smart speakers.

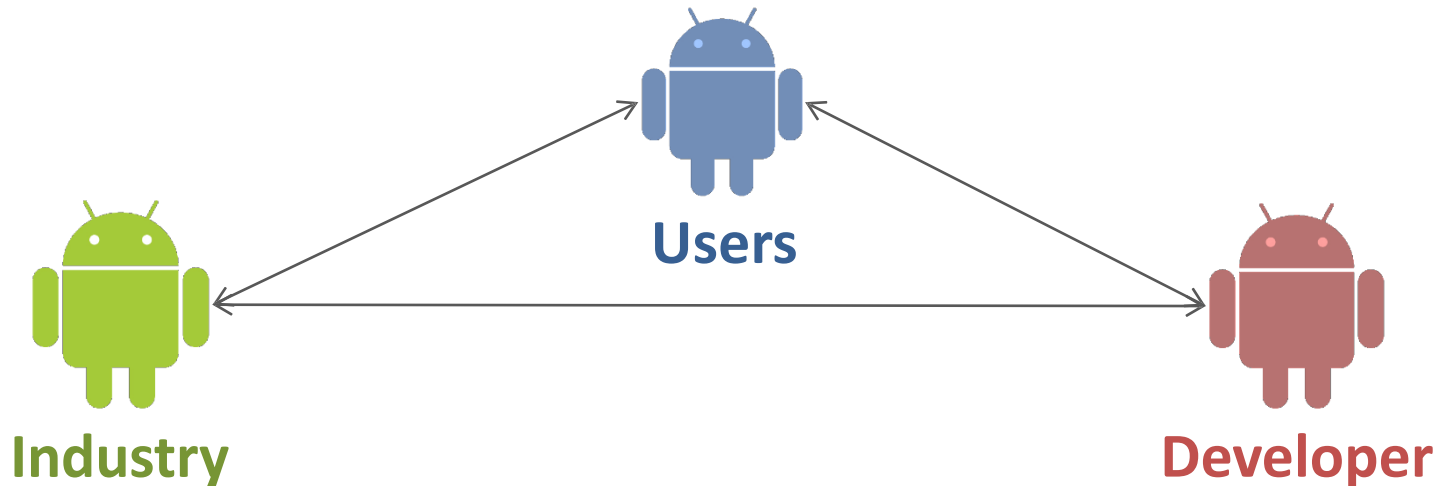
# Android Eco-System

## ❑ Equipment Maker:

- **E-Reader:** E-Reader is a device used for reading e-books, digital newspapers, other reading stuff.

# Android Eco-System





## Industry

- Software stack open-sourced under Apache 2.0 license
- Source available after first handsets ship
- Anyone will be able to build a system image

## Users

- Users have control of their experience
- They control what gets installed
- They choose the defaults

## Developer

- Don not need permission to ship an application
- No hidden or privileged framework APIs
- Can **integrate**, **extend** and **replace** existing components

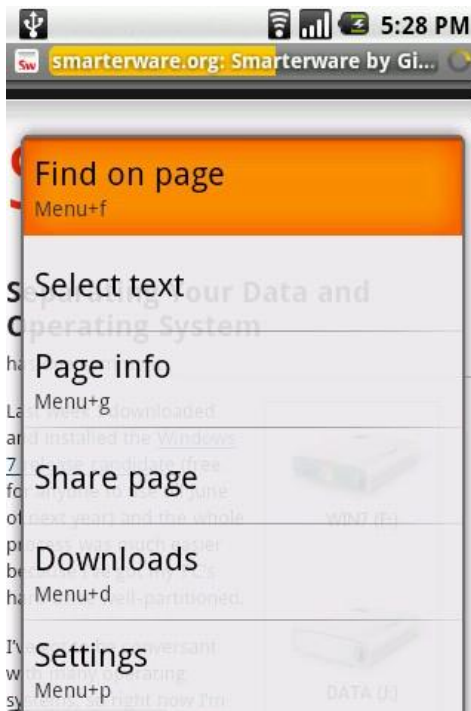


Developer Can:

Integrate

Extend

Replace







## 1.2 Need of Android

There are so many reasons you should choose Android platform for mobile application development.

### **1. Zero/negligible Development Cost :**

The development tools like Android SDK, JDK, and Eclipse IDE etc. are free to download for the android mobile application development. Also Google charge a small fee \$25, to distribute your mobile app on the Android Market.



# 1.2 Need of Android

## 2. Open Source:

- The Android OS is an open-source platform based on the Linux kernel and multiple open-source libraries.
- In this way developers are free to contribute or extend the platform as necessary for building mobile apps which run on Android devices.



## 1.2 Need of Android

### 3. Multi-Platform Support:

- In market, there are a wide range of hardware devices powered by the Android OS, including many different phones and tablet. Even development of android mobile apps can occur on Windows, Mac OS or Linux.



# 1.2 Need of Android

## 4. Multi-Carrier Support:

- World wide a large number of telecom carriers like Airtel, Vodafone, Idea Cellular, AT&T Mobility, BSNL etc. are supporting Android powered phones.

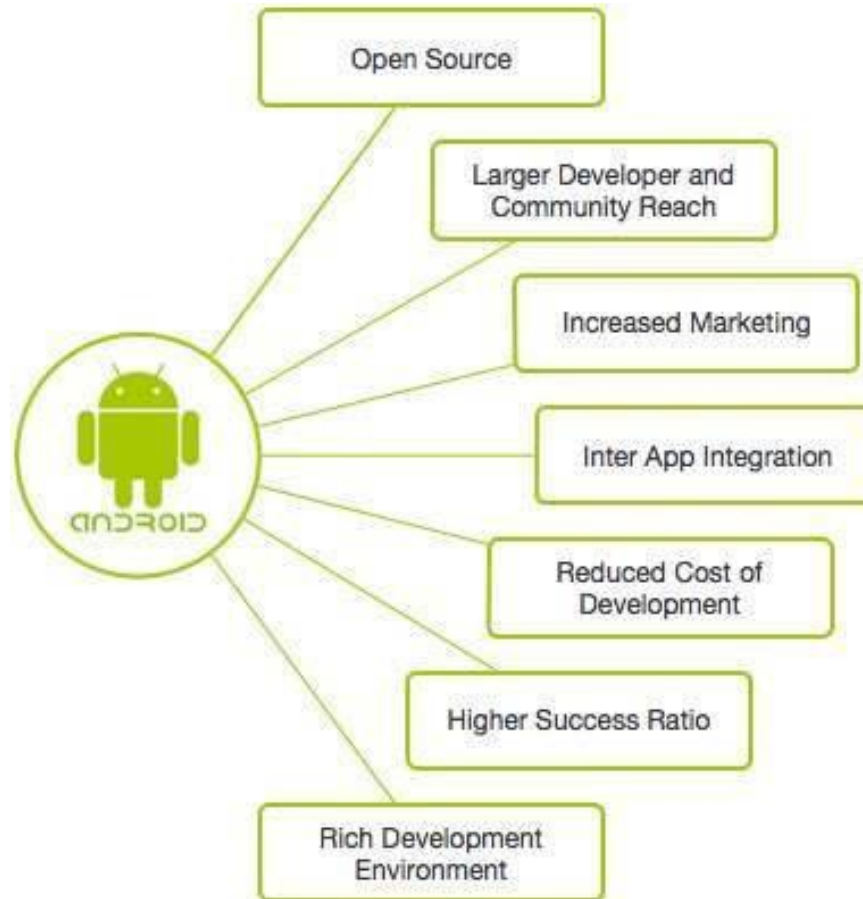


# 1.2 Need of Android

## 5. Open Distribution Model:

- Android Market place (Google Play store) has very few restrictions on the content or functionality of an android app.
- So the developer can distribute theirs app through Google Play store and as well other distribution channels like Amazon's app store.

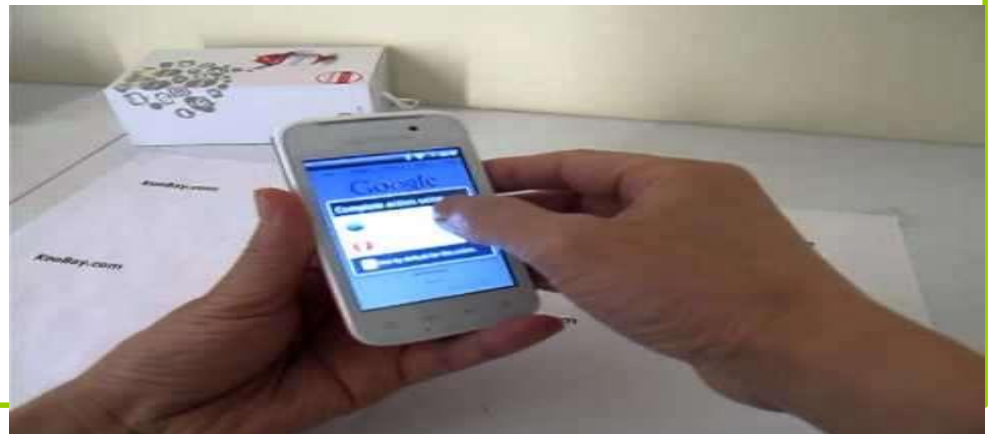
# Features of Android



# Features of Android

## ❑ Interface:

Android's default user interface is mainly based on direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, along with a virtual keyboard.



# Features of Android

## ❑ Applications :

- Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language.
- Java may be combined with C/C++, [91] together with a choice of non-default runtimes that allow better C++ support.



# Features of Android

- The Go programming language is also supported, although with a limited set of application programming interfaces (API).
- In May 2017, Google announced support for Android app development in the Kotlin programming language.



# Features of Android

## ❑ 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

# Features of Android

## ❑ Connectivity:

- Android supports connectivity technologies including GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi (no connections through Proxy server and no Ad hoc wireless network) LTE, NFC and WiMAX.

## ❑ Storage:

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

# Features of Android

## ❑ Messaging:

- SMS and MMS are available forms of messaging, including threaded text messaging and now Android Cloud to Device Messaging Framework is also a part of the Android Messaging service.



# Features of Android

## ❑ Media Support:

- Android supports the following audio/video/still media formats:
  - H.264 (in 3GP or MP4 container)
  - MPEG-4 SP
  - AMR
  - AAC
  - HE-AAC (in MP4 or 3GP container)



- MP3
- MIDI
- WAV
- JPEG
- PNG
- GIF
- BMP

# Features of Android

## ❑ Multi-Touch:

- Android has support for multi-touch which was originally 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.

# Features of Android

## ❑ Multi-Touch:



# Features of Android

- Application framework enabling reuse and replacement of components
- Dalvik virtual machine optimized for mobile devices (register based)
- Integrated browser based on the open source WebKit engine
- Optimized graphics powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES specification (hardware acceleration optional)



# Features of Android

- Media support for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- GSM Telephony (hardware dependent)
- Bluetooth, EDGE, 3G, WiFi, NFC (hardware dependent)
- Camera, GPS, compass, accelerometer, gyroscope (hardware dependent)
- Rich development environment including a device emulator, tools for debugging, memory and performance profiling, and a plugin for the Eclipse IDE

# Features of Android

- JPEG – Joint Photographic Experts Group
- GIF – Graphics Interchange Format,
- PNG – Portable Network Graphics
- BMP – Windows Bitmap
- TIFF – Tag Image File Format
- WAV – Waveform PCM Audio
- WMA – Windows Media Audio
- BWF – Broadcast Wave Format
- DAT – Digital Audio Tape
- AMR-Adaptive Multi-Rate

# Features of Android

- MPEG-Moving Picture Experts Group
- AAC-Advanced Audio Coding
- AVI-Audio Video Interleave
- MP3-MPEG Audio Layer III
- MP4-MPEG-4 AVC (Advanced Video Coding)
- MKV-Matroska Video
- DVI-Digital Video Interactive

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Java JDK:**

- The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Java JDK:**

- It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Java JDK: JVM**

- JVM (Java Virtual Machine) is a specification that provides runtime environment in which java bytecode can be executed.
- JVMs are available for many hardware and software platforms. JVM is a part of Java Run Environment (JRE).

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Java JDK: JVM**

- The JVM performs following operation:
  - Loads code
  - Verifies code
  - Executes code
  - Provides runtime environment

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Java JDK: JRE**

- The Java Runtime Environment (JRE) is a set of software tools for development of Java applications.
- It combines the Java Virtual Machine (JVM), platform core classes and supporting libraries.



## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **Android SDK:**

- The Android SDK (software development kit) is a set of development tools used to develop applications for Android platform.

## 1.3 Tools & Software Required

- ❖ **Software Development IDE and Tools:**
  - **Android SDK:**
    - The Android SDK includes the following:
      - Required libraries
      - Debugger
      - An Emulator

# 1.3 Tools & Software Required

## ❖ **Software Development IDE and Tools:**

### ■ **Android SDK:**

- The Android SDK includes the following:
  - Relevant documentation for the Android application program interfaces (APIs)
  - Sample source code
  - Tutorials for the Android OS

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **IDE: Android Studio**

- Developed by Google, Android Studio is an all-rounder integrated development environment.
- Android has Gradle-base support that has features like visual layout editor, intelligent code editor, real-time profilers and APK analyzer.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **IDE: Visual Studio – Xamarin**

- Xamarin was launched in 2011 which is the best free IDE for delivering an enterprise-quality, cross platform approach.
- Xamarin supplies add-ins to Microsoft Visual Studio that allows developers to build Android, iOS, and Windows apps within the IDE.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **IDE: IntelliJ IDEA**

- Using this IDE, you can do in-depth coding, quick navigation, and error analysis. It supports mobile app development with the help of Java, Scala, Kotlin, Groovy.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **IDE: Eclipse IDE**

- It is one of the most popular IDEs of Android apps.
- The open-source software is free to use.

Released under the Eclipse Public License, it holds a large community having plenty of plugins and configurations.

## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **IDE: Eclipse IDE**

- Highly customizable offers full support for Java programming language and XML.



# 1.3 Tools & Software Required

## ❖ **Software Development IDE and Tools:**

### ■ **App Development Tools : ADB (Android Debug Bridge)**

- ADB is a command-line communication bridge between the android devices and computer devices on which you are doing the coding. ADB allows the installation and debugging of the android apps on the devices.

# 1.3 Tools & Software Required

## ❖ **Software Development IDE and Tools:**

- **App Development Tools : AVD Manager (Android Virtual Device)**
  - AVD is an emulator, or you can say the virtual device that allows you to test and debug the android apps. AVD has a range of characteristics, including android phones, tablets, wearOS, androidTV, etc.

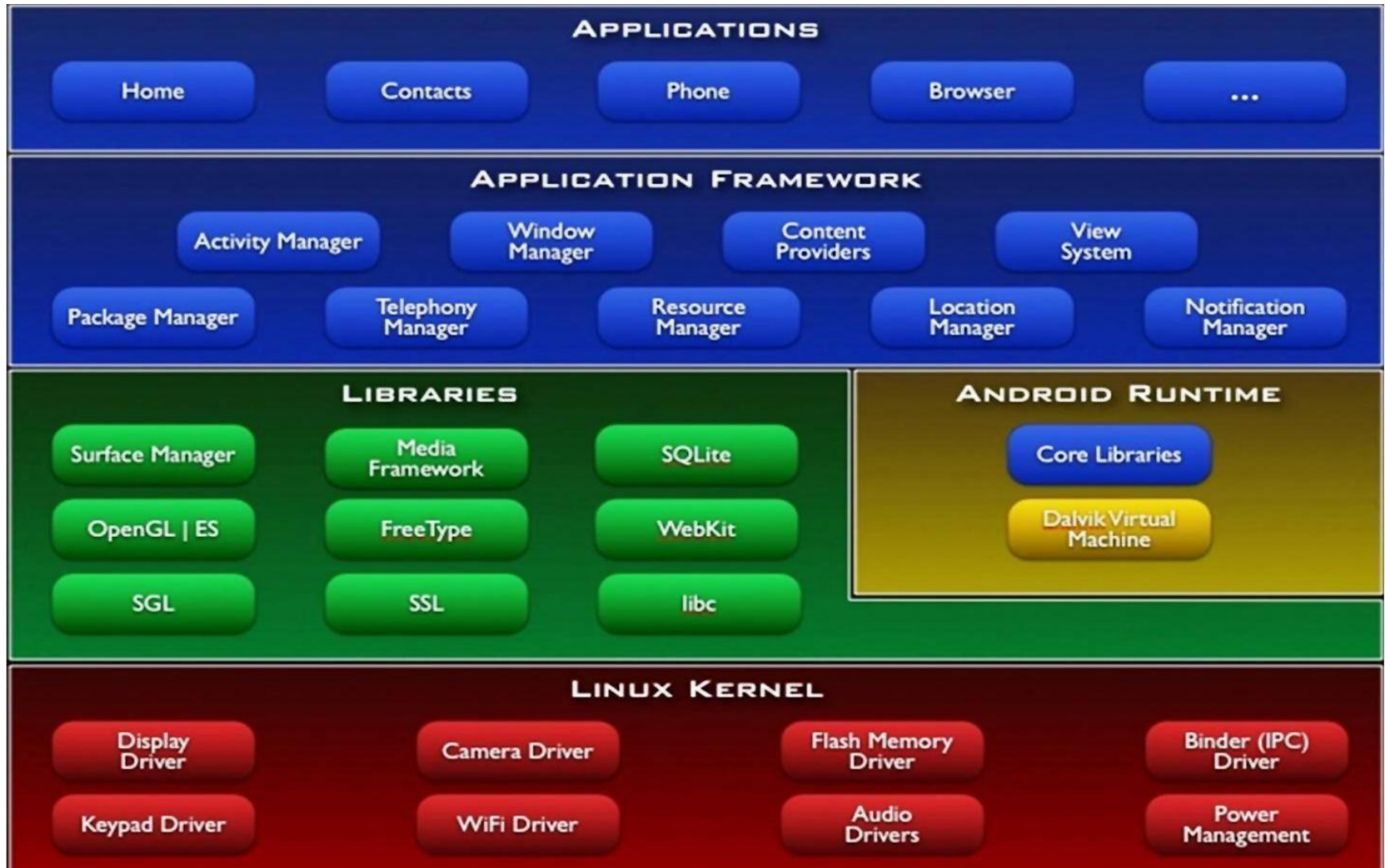
## 1.3 Tools & Software Required

### ❖ **Software Development IDE and Tools:**

#### ■ **App Development Tools : Gradle Build Tool**

- Gradle Build is also a powerful and most widely used tool for android app development. It is primarily used to speed up the android app development process by the developers. The numerous API and plugins available in the Gradle build tool automate the whole procedure and made the deployment faster.

# 1.4 Android Architecture



## 1.4 Android Architecture

**Android architecture** or **Android software stack** is categorized into five parts:

- Linux kernel
- Native Libraries (middleware)
- Android Runtime
- Application Framework
- Applications

# Linux Kernel

- Android OS is built on top of the Linux 2.6 Kernel
  - Linux Core functionality
    - Memory management
    - Process management
    - Networking
    - Security settings
  - Hardware drivers

## LINUX KERNEL

Display  
Driver

Camera Driver

Flash Memory  
Driver

Binder (IPC)  
Driver

Keypad Driver

WiFi Driver

Audio  
Drivers

Power  
Management

# Linux Kernel

- The basic Layer is the Linux Kernel. The entire Android OS is built on the top the Linus 2.6 Kernel with the architectural changes made by Google.
- It is highly trusted for years together for giving better security and networking etc.
- This layer has 10 major components which are nothing but drivers.
- Android run time gives .dex file which converted from the .class file. .dex file goes to the Linux Kernel and call the suitable drivers so that it can communicate with the corresponding drivers.

# Native Libraries

- **Android's Native Libraries:**

- **Libc:** c standard lib.
- **SSL:** Secure Socket Layer
- **SGL(Scalable Graphics Libraries):** 2D image engine
- **OpenGL|ES:** 3D image engine
- **Media Framework:** media codecs
- **SQLite:** Database engine
- **WebKit:** Kernel of web browser
- **FreeType:** Bitmap and Vector
- **SufaceManager:** Compose window manager with off-screen buffering.





# Native Libraries

- Each Component of the frame like button, textfield, image(2D, 3D) is rendered & interacted by the **Surface Manager**.
- **Open GL ES**: This Component is provides mostly 3D graphics libraries which have to implemented by Java application and it is rendered by the application layer when they want some 3D graphics.
- **SQLite**: We can use SQLite which is an open source database as our applications are light weight.

# Native Libraries

- **Media Framework:** Picture Quality Should be good, Multimedia actions should be good which includes Sounds, so to get those activities or application Android API has to call Media Framework Component to play and view the video-audio.
- **SGL(Scalable Graphics Libraries):** It is responsible for the implementing low level graphics by using JNI.
- **FreeType:** It is responsible for to support the font quality, the image(bitmap images).

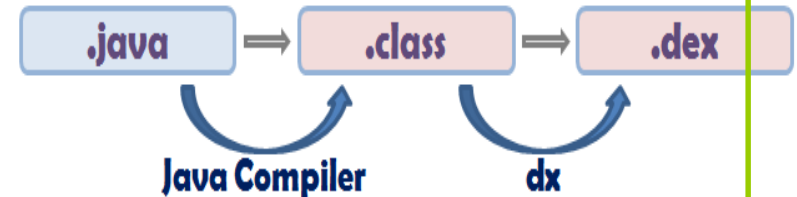
# Native Libraries

- **SSL(Secured Socket Layer):** It is used in internal applications. If we want go through online from mobile then at that time we need some secure socket layer. It is also used in case of Chat & messaging system.
- **WebKit:** It is responsible for giving the browser supports. Eg. Google chrome, Apples Safari.
- **Libc:** When we want to access some code C libraries headers then this component is useful.

# Android Runtime

- **Core Libraries:**

- Provides the functionality of the JAVA Programming Language



- **Dalvik VM:**

- A type of Java Virtual Machine
- Register based (not stack machine like JVM)
- Optimization for low memory requirements, provides Security
- Executes .dex (Dalvik-Executable) files instead of .class
- DX tool converts classes to .dex format



Each Android application:

- runs on its **own Process**
- runs on its **own Instance of Dalvik VM**
- is assigned its **own Linux user ID**

# Android Runtime

- The android runtime provides a key component called Dalvik Virtual Machine which is a kind of java virtual machine. It is specially designed and optimized for android. The Dalvik VM is the process virtual machine in the android operating system. It is software that runs apps on android devices.
- The Dalvik VM makes use of Linux core features like memory management and multithreading which is in java language. The Dalvik VM enables every Android application to run its own process. The Dalvik VM executes the files in the .dex format.

# Application Framework

- The application framework layer provides many higher-level services to applications such as windows manager, view system, package manager, resource manager, etc. The application developers are allowed to make use of these services in their applications.

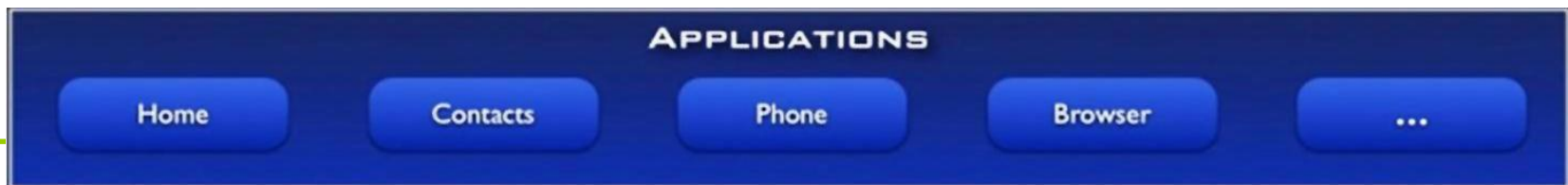
# Application Framework

- The blocks that our applications directly interact with.
- **Important blocks:**
  - **Activity Manager:** Manages the activity life cycle of applications
  - **Content Providers:** Manage the data sharing between applications
  - **Telephony Manager:** Manages all voice calls. We use telephony manager if we want to access voice calls in our application.
  - **Location Manager:** Location management, using GPS or cell tower
  - **Resource Manager:** Manage the various types of resources we use in our Application



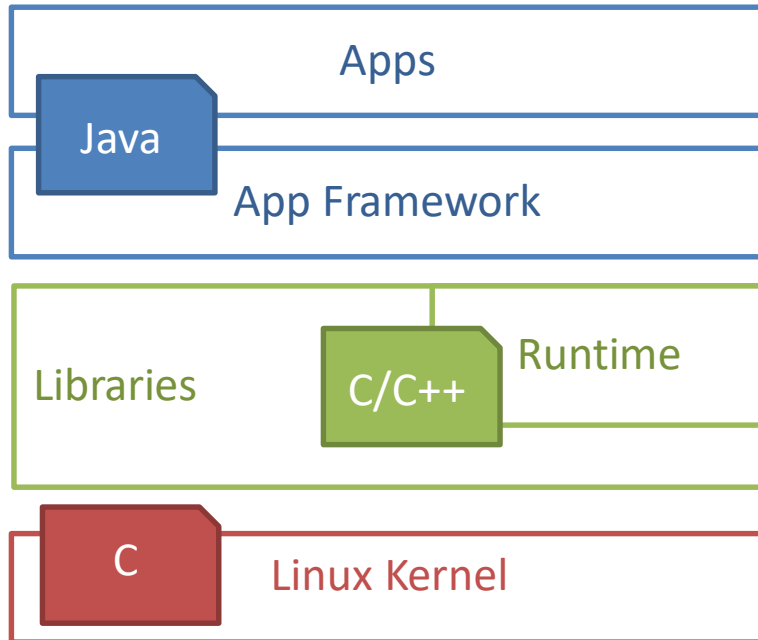
# Applications

- You will find all the android applications at the top layer and you will write your application and install it on this layer. Examples of such applications are contacts, books, browsers, services, etc. Each application performs a different role in the overall applications.
- As developers, we are able to write an app which replaces any existing system app.



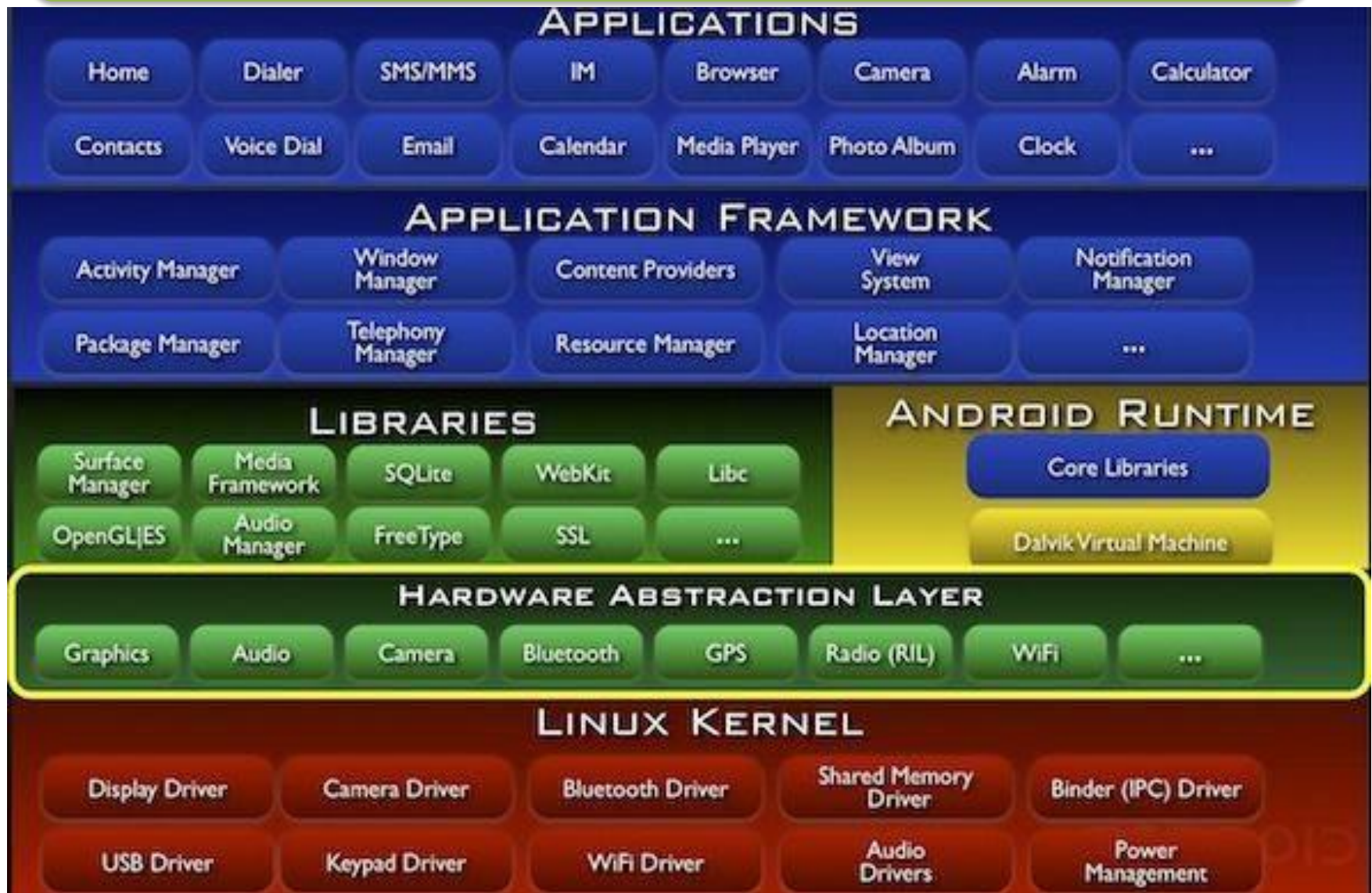


# Android Architecture

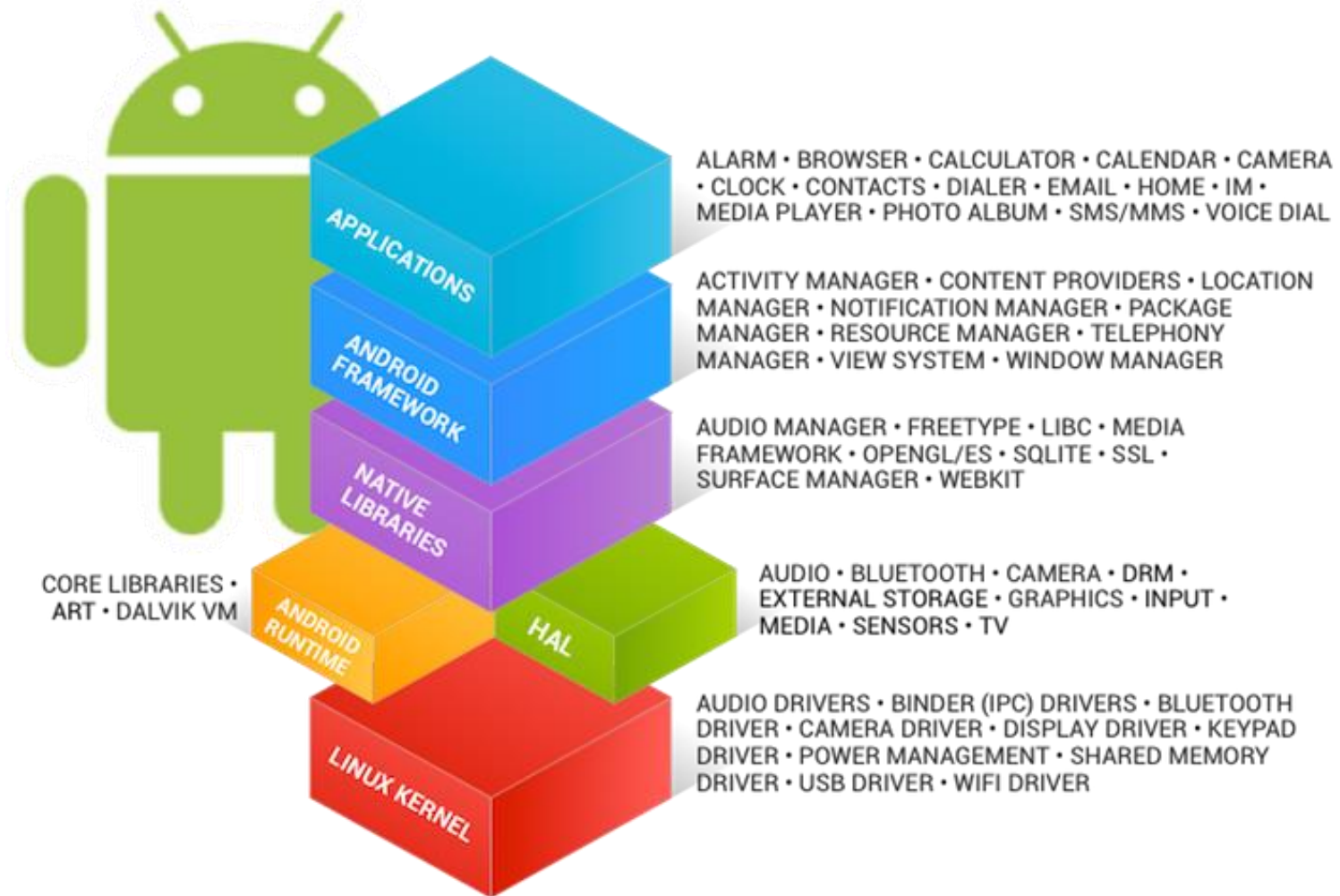


- Design goals
  - Open Source
  - High flexibility
  - High data accessibility
  - Rapid development (XML, Java)
- Used Languages
  - App: Java
  - Framework: Java
  - Libraries: C/C++
  - OS & Drivers: C

# 1.4 Android Architecture



# Hardware Abstraction Layer (HAL)



# Hardware Abstraction Layer (HAL)

- On top of that kernel is the **Hardware Abstraction Layer**: an interface to drivers that can programmatically access hardware elements, such as the camera, disk storage, Wifi antenna, etc.
- These drivers are generally written in C; we won't interact with them directly in this course.

# Hardware Abstraction Layer (HAL)

- The hardware abstraction layer (HAL) defines a standard interface for hardware vendors to implement and allows Android to be agnostic about lower-level driver implementations. The HAL allows you to implement functionality without affecting or modifying the higher level system. HAL implementations are packaged into modules (.so) file and loaded by the Android system at the appropriate time.

# Hardware Abstraction Layer (HAL)

- On top of the HAL is the Runtime and Android Framework, which provides a set of abstraction in the Java language which we all know and love. For this course, Android Development will involve writing Java applications that interact with the Android Framework layer, which handles the task of interacting with the device hardware for us.



# Resources

- Introduction to Android

<https://code.google.com/p/androidgroup/downloads/detail?name=Introduction%20to%20Android.pdf>

- Android Architecture

<http://www.android-app-market.com/android-architecture.html>

- Application Fundamentals

<http://developer.android.com/guide/components/fundamentals.html>