

**UNIT I – INTRODUCTION TO MOBILE APPLICATION**

**Native App Development**

Native apps are built for specific platforms and are written in the languages the platform accepts (for example, Swift and Objective-C are common languages for iOS apps and Java or Kotlin are common for Android apps). Native apps are fast and responsive, distributed in app stores and don't require an internet connection. Overall, native apps offer a better user experience but are more expensive to develop than other options. Native apps are also built using the specific Integrated Development Environment (IDE) for the given operating systems. Both Apple and Google provide app developers with their own development tools, interface elements, and SDK.

**Advantages of Native Apps**

- Native apps are very fast and responsive because they are built for a specific platform
- They have the best performance
- They are distributed in app stores
- They are more interactive, intuitive and run much smoother in terms of user input and output
- Native allows developers to access the full feature set of the given platform
- Internet connection is not required, although it depends on the functionality
- Overall better user experience.

**Disadvantages of Native Apps**

- Difficult programming languages to learn which means you need experienced developers
- More expensive
- Not the best option for very simple apps

**Web App Development**

Web apps are hosted on web browsers, and are essentially websites that look like native apps. However, instead of installing the application to the device's home screen, like a native app, users interact with the app through a web view. This type of app is easy to build, easy to maintain and an inexpensive option; however, they require a web browser, are much slower than native apps, and can't leverage device utilities. Web apps are also much less interactive and intuitive than native apps.

Web apps load in browsers like Chrome, Safari, or Firefox, and don't need to be downloaded from app stores like mobile apps. Web apps also don't take up storage on the user's device.

Typically, web apps are built in JavaScript, CSS, and HTML5 and run inside a browser (Safari, Chrome, etc.). There is no software development kit for developers to use; however, there are templates that developers can work with.

### **Advantages of Web Apps**

- Easy to build
- Easy to maintain
- Inexpensive
- Build one app for all platforms – iOS, Android, etc. as long as it can run in a browser

### **Disadvantages of Web Apps**

- Needs a browser to run. Users have to take another step to type in the URL, which contributes to a complicated user experience
- Much slower than native apps
- Less interactive and intuitive than native apps
- No icon on the mobile desktop as you would if it were downloaded from app stores

### **Hybrid App Development**

Hybrid apps are essentially a combination of native and web apps. Hybrid apps are less expensive than native apps, don't require a browser. However, they're slower than native apps and are not customizable to individual platforms like native app. Hybrid apps work across multiple platforms and behave like native apps. These types of apps are built with HTML, CSS, or JavaScript and run in a web view.

A hybrid app consists of two parts: first, the backend code which is built using languages such as HTML, CSS, or JavaScript, and second, a native shell that is downloadable and loads the code using a web view.

### **Native Apps vs. Mobile Apps**

A **Native App** is an app developed essentially for one particular mobile device and is installed directly onto the device itself. Users of native apps usually download them via app stores online or the app marketplace, such as the Apple App Store, the [Google Play store](#) and so on. An example of a native app is the Camera+ app for [Apple's iOS](#) devices.

A **Web App**, on the other hand, are basically Internet-enabled apps that are accessible via the mobile device's web browser. They need not be downloaded onto the user's [mobile device](#) in order to be accessed. The Safari browser is a good example of a mobile Web app.

## **MOBILE OPERATING SYSTEM AND APPLICATION**

Much like the Linux or Windows operating system, a mobile operating system is the software platform on top of which other programs can run on mobile devices.

Mobile operating systems are those that are designed specifically to power smartphones, tablets, and wearables, the mobile devices we take with us wherever we go. The top popular mobile operating systems are Android and iOS, but others include BlackBerry OS, webOS, and watchOS.

### **What Does a Mobile Operating System Do**

When you first start up a mobile device, you typically see a screen of icons or tiles. They are placed there by the operating system. Without an OS, the device wouldn't even start. The mobile operating system is a set of data and programs that runs on a mobile device. It manages the hardware and makes it possible for smartphones, tablets, and wearables to run apps. A mobile OS also manages mobile multimedia functions, mobile and internet connectivity, the touch screen, Bluetooth connectivity, GPS navigation, cameras, speech recognition, and more in a mobile device. Most operating systems are not interchangeable among devices. If you have an Apple iOS phone, you can't load the Android OS on it and vice versa.

### **Types of Mobile Operating Systems**

#### **1. Android OS (Google Inc.)**

The Android mobile operating system is Google's open and free software stack that includes an operating system, middleware and also key applications for use on mobile devices, including smartphones. Updates for the open source Android mobile operating system have been developed under "dessert-inspired" version names (Cupcake, Donut, Eclair, Gingerbread, Honeycomb, Ice Cream Sandwich) with each new version arriving in alphabetical order with new enhancements and improvements.

#### **2. Bada (Samsung Electronics)**

Bada is a proprietary Samsung mobile OS that was first launched in 2010. The Samsung Wave was the first smartphone to use this mobile OS. Bada provides mobile features such as multipoint-touch, 3D graphics and of course, application downloads and installation.

#### **3. BlackBerry OS (Research In Motion)**

The BlackBerry OS is a proprietary mobile operating system developed by Research In Motion for use on the company's popular BlackBerry handheld devices. The BlackBerry platform is popular with corporate users as it offers synchronization with Microsoft

Exchange, Lotus Domino, Novell GroupWise email and other business software, when used with the BlackBerry Enterprise Server.

#### **4. iPhone OS / iOS (Apple)**

Apple's iPhone OS was originally developed for use on its iPhone devices. Now, the mobile operating system is referred to as iOS and is supported on a number of Apple devices including the iPhone, iPad, iPad 2 and iPod Touch. The iOS mobile operating system is available only on Apple's own manufactured devices as the company does not license the OS for third-party hardware. Apple iOS is derived from Apple's Mac OS X operating system.

#### **5. MeeGo OS (Nokia and Intel)**

A joint open source mobile operating system which is the result of merging two products based on open source technologies: Maemo (Nokia) and Moblin (Intel). MeeGo is a mobile OS designed to work on a number of devices including smartphones, netbooks, tablets, in-vehicle information systems and various devices using Intel Atom and ARMv7 architectures.

#### **6. Palm OS (Garnet OS)**

The Palm OS is a proprietary mobile operating system (PDA operating system) that was originally released in 1996 on the Pilot 1000 handheld. Newer versions of the Palm OS have added support for expansion ports, new processors, external memory cards, improved security and support for ARM processors and smartphones. Palm OS 5 was extended to provide support for a broad range of screen resolutions, wireless connections and enhanced multimedia capabilities and is called Garnet OS.

#### **7. Symbian OS (Nokia)**

Symbian is a mobile operating system (OS) targeted at mobile phones that offers a high-level of integration with communication and personal information management (PIM) functionality. Symbian OS combines middleware with wireless communications through an integrated mailbox and the integration of Java and PIM functionality (agenda and contacts). Nokia does not maintain Symbian as an open source development project.

#### **8. webOS (Palm/HP)**

WebOS is a mobile operating system that runs on the Linux kernel. WebOS was initially developed by Palm as the successor to its Palm OS mobile operating system. It is a proprietary Mobile OS which was eventually acquired by HP and now referred to as webOS (lower-case w) in HP literature. HP uses webOS in a number of devices including several smartphones and HP TouchPads.

## 9. Windows Mobile (Windows Phone)

Windows Mobile is Microsoft's mobile operating system used in smartphones and mobile devices – with or without touchscreens. The Mobile OS is based on the Windows CE 5.2 kernel. In 2010 Microsoft announced a new smartphone platform called Windows Phone 7.

### Mobile Databases

Mobile databases are separate from the main database and can easily be transported to various places. Even though they are not connected to the main database, they can still communicate with the database to share and exchange data.

The mobile database includes the following components:

1. The main system database that stores all the data and is linked to the mobile database.
2. The mobile database that allows users to view information even while on the move. It shares information with the main database.
3. The device that uses the mobile database to access data. This device can be a mobile phone, laptop etc.
4. A communication link that allows the transfer of data between the mobile database and the main database.

### *Advantages of Mobile Databases*

1. The data in a database can be accessed from anywhere using a mobile database. It provides wireless database access.
2. The database systems are synchronized using mobile databases and multiple users can access the data with seamless delivery process.
3. Mobile databases require very little support and maintenance.
4. The mobile database can be synchronized with multiple devices such as mobiles, computer devices, laptops etc.

### *Disadvantages of Mobile Databases*

1. The mobile data is less secure than data that is stored in a conventional stationary database. This presents a security hazard.
2. The mobile unit that houses a mobile database may frequently lose power because of limited battery. This should not lead to loss of data in database.

## **Android**

**Android** is a software package and linux based operating system for mobile devices such as tablet computers and smartphones. It is developed by Google and later the OHA (Open Handset Alliance). Java language is mainly used to write the android code even though other languages can be used. The goal of android project is to create a successful real-world product that improves the mobile experience for end users. There are many code names of android such as Lollipop, Kitkat, Jelly Bean, Ice cream Sandwich, Froyo, Ecliar, Donut etc which is covered in next page.

### **What is Open Handset Alliance (OHA)**

It's a consortium of 84 companies such as google, samsung, AKM, synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc. It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Plateform.

### **Features of Android**

After learning what is android, let's see the features of android. The important features of android are given below:

- 1) It is open-source.
- 2) Anyone can customize the Android Platform.
- 3) There are a lot of mobile applications that can be chosen by the consumer.
- 4) It provides many interesting features like weather details, opening screen, live RSS (Really Simple Syndication) feeds etc.

It provides support for messaging services(SMS and MMS), web browser, storage (SQLite), connectivity (GSM, CDMA, Blue Tooth, Wi-Fi etc.), media, handset layout etc.

<b>Sr.No.</b>	<b>Feature &amp; Description</b>
1	<b>Beautiful UI</b> Android OS basic screen provides a beautiful and intuitive user interface.

2	<b>Connectivity</b> GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.
3	<b>Storage</b> SQLite, a lightweight relational database, is used for data storage purposes.
4	<b>Media support</b> 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.
5	<b>Messaging - SMS and MMS</b>
6	<b>Web browser</b> Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.
7	<b>Multi-tasking</b> User can jump from one task to another and same time various application can run simultaneously.
8	<b>Resizable widgets</b> Widgets are resizable, so users can expand them to show more content or shrink them to save space.
9	<b>Multi-Language - Supports single direction and bi-directional text.</b>
10	<b>Wi-Fi Direct</b> A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.

### Categories of Android applications

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

- Entertainment
- Tools
- Communication
- Productivity
- Personalization
- Music and Audio
- Social
- Media and Video
- Travel and Local etc.

### **History of Android**

The history and versions of android are interesting to know. The code names of android ranges from AtoJ 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.

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

### **Android Versions, Codename and API**

Let's see the android versions, codenames and API Level provided by Google.



Version	Code name	API Level
1.5	Cupcake	3
1.6	Donut	4
2.1	Eclair	7
2.2	Froyo	8
2.3	Gingerbread	9 and 10
3.1 and 3.3	Honeycomb	12 and 13
4.0	Ice Cream Sandwich	15
4.1, 4.2 and 4.3	Jelly Bean	16, 17 and 18
4.4	KitKat	19
5.0	Lollipop	21
6.0	Marshmallow	23
7.0	Nougat	24-25
8.0	Oreo	26-27

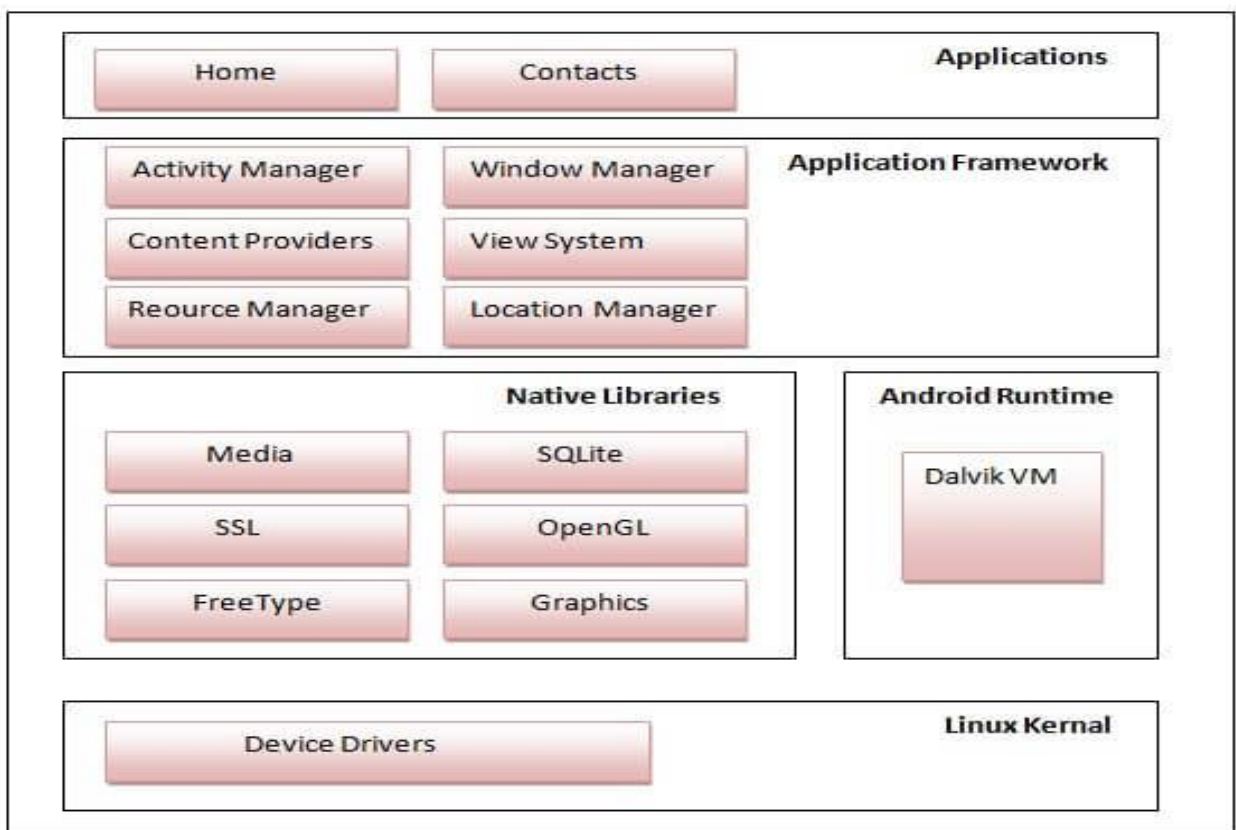
**What is API level?**

API Level is an integer value that uniquely identifies the framework API revision offered by a version of the Android platform.

## Android Architecture

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

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



### 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.

### 2) Native Libraries

On the top of linux kernel, there are **Native libraries** such as WebKit, OpenGL, FreeType, SQLite, Media, C runtime library (libc) etc. The WebKit library is responsible for browser support, SQLite is for database, FreeType for font support, Media for playing and recording audio and video formats.

### 3) 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.

### 4) Android 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.

### 5) 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.

### Android Core Building Blocks



An android **component** is simply a piece of code that has a well defined life cycle e.g. Activity, Receiver, Service etc. The **core building blocks** or **fundamental components** of android are activities, views, intents, services, content providers, fragments and AndroidManifest.xml.

#### *Activity*

An activity is a class that represents a single screen. It is like a Frame in AWT.

#### *View*

A view is the UI element such as button, label, text field etc. Anything that you see is a view.

### ***Intent***

Intent is used to invoke components. It is mainly used to:

- Start the service
- Launch an activity
- Display a web page
- Display a list of contacts
- Broadcast a message
- Dial a phone call etc.

For example, you may write the following code to view the webpage.

1. Intent intent=**new** Intent(Intent.ACTION\_VIEW);
2. intent.setData(Uri.parse("http://www.javatpoint.com"));
3. startActivity(intent);

### ***Service***

Service is a background process that can run for a long time. There are two types of services local and remote. Local service is accessed from within the application whereas remote service is accessed remotely from other applications running on the same device.

### ***Content Provider***

Content Providers are used to share data between the applications.

### ***Fragment***

Fragments are like parts of activity. An activity can display one or more fragments on the screen at the same time.

### ***AndroidManifest.xml***

It contains informations about activities, content providers, permissions etc. It is like the web.xml file in Java EE.

### ***Android Virtual Device (AVD)***

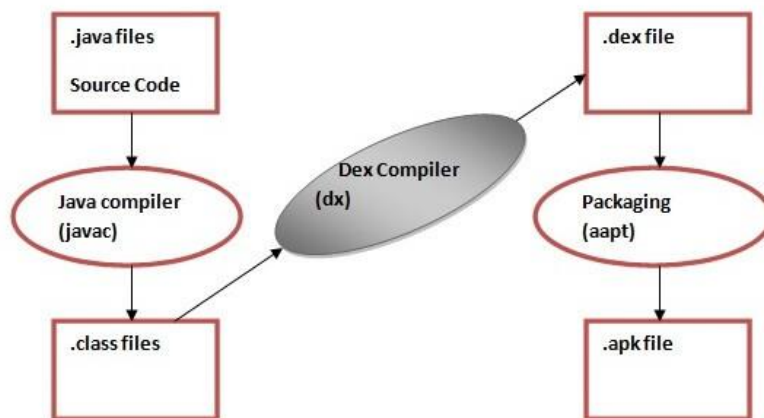
It is used to test the android application without the need for mobile or tablet etc. It can be created in different configurations to emulate different types of real devices.

## Dalvik Virtual Machine | DVM

As we know the modern JVM is high performance and provides excellent memory management. But it needs to be optimized for low-powered handheld devices as well. 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. The Dex compiler converts the class files into the .dex file that run on the Dalvik VM. Multiple class files are converted into one dex file.

Let's see the compiling and packaging process from the source file:



- The **javac tool** compiles the java source file into the class file.
- The **dx tool** takes all the class files of your application and generates a single .dex file. It is a platform-specific tool.
- The **Android Assets Packaging Tool (aapt)** handles the packaging process.

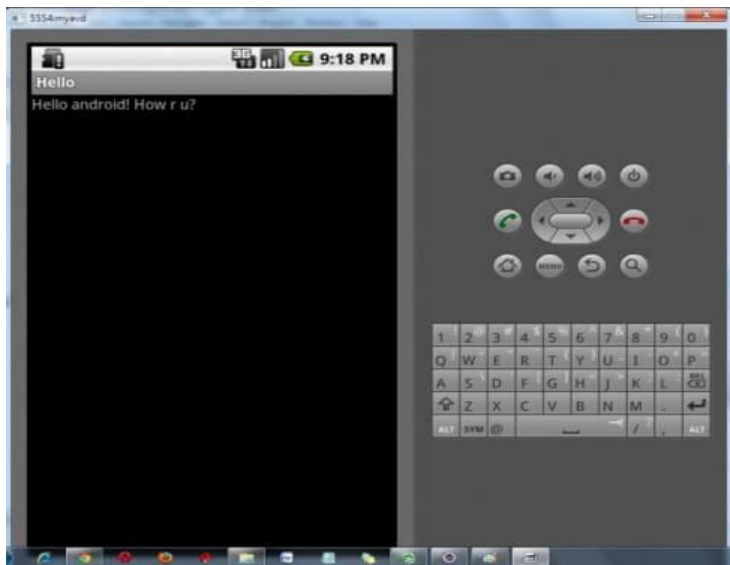
## Android Emulator

**Android Emulator** is used to run, debug and test the android application. If you don't have the real device, it can be the best way to run, debug and test the application. It uses an open source processor emulator technology called **QEMU**.

The emulator tool enables you to start the emulator from the command line. You need to write:

```
emulator -avd <AVD NAME>
```

In case of Eclipse IDE, you can create AVD by **Window menu > AVD Manager > New**. In the given image, you can see the android emulator, it displays the output of the hello android example.



## References

1. Barry Burd, “Android Application development – All in one for Dummies”, 2<sup>nd</sup> Edition.
  2. <https://developer.android.com/>
  3. <https://abhiandroid.com/>
  4. <https://www.javatpoint.com/>
  5. <https://www.tutorialspoint.com/android/index.htm>
-