PPT ON
MOBILE APPLICATIONS AND SERVICES
M.TECH
III SEM (IARE-R18)

UNIT-I

Introduction to mobile computing

Mobile Computing is a technology that allows transmission of data, voice and video via a computer or any other wireless enabled device without having to be connected to a fixed physical link.

The main concept involves –

- Mobile communication
- Mobile hardware
- Mobile software

Mobile Communication

- The mobile communication in this case, refers to the infrastructure put in place to ensure that seamless and reliable communication goes on.
- These would include devices such as protocols, services, bandwidth, and portals necessary to facilitate and support the stated services.
- The data format is also defined at this stage. This ensures that there is no collision with other existing systems which offer the same service.
- Since the media is unguided/unbounded, the overlaying infrastructure is basically radio wave-oriented.
- That is, the signals are carried over the air to intended devices that are capable of receiving and sending similar kinds of signals

Cont..

Fig: Mobile Communication

Fig: Mobile Hardware

Mobile Hardware

- Mobile hardware includes mobile devices or device components that receive or access the service of mobility. They would range from portable laptops, smart phones, tablet Pc's, Personal Digital Assistants
- These devices will have a receptor medium that is capable of sensing and receiving signals.
- These devices are configured to operate in full- duplex, whereby they are capable of sending and receiving signals at the same time.
- They don't have to wait until one device has finished communicating for the other device to initiate communications.
- Above mentioned devices use an existing and established network to operate on. In most cases, it would be a wireless network

Mobile software

- Mobile software is the actual program that runs on the mobile hardware. It deals with the characteristics and requirements of mobile applications.
- This is the engine of the mobile device. In other terms, it is the operating system of the appliance.
- It's the essential component that operates the mobile device.
- Since portability is the main factor, this type of computing ensures that users are not tied or pinned to a single physical location, but are able to operate from anywhere.
- It incorporates all aspects of wireless communications.

Introduction to Android Development Environment

- Android is a mobile operating system that is based on a modified version of Linux. It was originally developed by a startup of the same name, Android, Inc.
- In 2005, as part of its strategy to enter the mobile space, Google purchased Android and took over its development work (as well as its development team).
- Google wanted Android to be open and free; hence, most of the Android code was released under the open-source Apache License, which means that anyone who wants to use Android can do so by downloading the full Android source code.
- Moreover, vendors (typically hardware manufacturers) can add their own proprietary extensions to Android and customize Android to differentiate their products from others.

Cont..

- Such companies include Motorola and Sony Ericsson, which for many years have been developing their own mobile operating systems. When the iPhone was launched, many of these manufacturers had to scramble to find new ways of revitalizing their products.
- These manufacturers see Android as a solution — they will continue to design their own hardware and use Android as the operating system that powers it. The main advantage of adopting Android is that it offers a unified approach to application development.
- Developers need only develop for Android, and their applications should be able to run on numerous different devices, as long as the devices are powered using Android.
- In the world of smartphones, applications are the most important part of the success chain. Device manufacturers therefore see Android as their best hope to challenge the onslaught of the iPhone, which already commands a large base of applications. Android Versions Android has gone through quite a number of updates since its first release

Cont..

Table 1-1 shows the various versions of Android and their codenames.

Android Version	Release Date	Codename
1.1	9 February 2009	
1.5	30 April 2009	Cupcake
1.6	15 September 2009	Donut
2.0/2.1	26 October 2009	Éclair
2.2	20 May 2010	Froyo
2.3	6 December 2010	Gingerbread
3.0	Unconfirmed at the time of writing	Honeycomb

Cont..

Features of Android As Android is open source and freely available to manufacturers for customization, there are no fixed hardware and software configurations. However, Android itself supports the following features:

- Storage — Uses SQLite, a lightweight relational database, for data storage. Chapter 6 discusses data storage in more detail.
- Connectivity — Supports GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth (includes A2DP and AVRCP), WiFi, LTE, and WiMAX. Chapter 8 discusses networking in more detail.
- Messaging — Supports both SMS and MMS. Chapter 8 discusses messaging in more detail.
- Web browser — Based on the open-source WebKit, together with Chrome's V8 JavaScript engine.
- Media support — Includes support for the following media: 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, WAV, JPEG, PNG, GIF, and BMP

Cont..

- Hardware support — Accelerometer Sensor, Camera, Digital Compass, Proximity Sensor, and GPS Multi-touch — Supports multi-touch screens Multi-tasking — Supports multi-tasking applications
- Flash support — Android 2.3 supports Flash 10.1.
- Tethering — Supports sharing of Internet connections as a wired/wireless hotspot Architecture of Android In order to understand how Android works, take a look at Figure 1-1, which shows the various layers that make up the Android operating system (OS).

Architecture of Android

In order to understand how Android works, take a look at Figure 1-1, which shows the various layers that make up the Android operating system (OS).

The Android OS is roughly divided into five sections in four main layers:

- **Linux kernel** — This is the kernel on which Android is based. This layer contains all the low- level device drivers for the various hardware components of an Android device.
- **Libraries** — These contain all the code that provides the main features of an Android OS. For example, the SQLite library provides database support so that an application can use it for data storage. The WebKit library provides functionalities for web browsing.
- **Android runtime** — At the same layer as the libraries, the Android runtime provides a set of core libraries that enable developers to write Android apps using the Java programming language.

Cont..

Dalvik is a specialized virtual machine designed specifically for Android and optimized for battery-powered mobile devices with limited memory and CPU.

- **Application framework** — Exposes the various capabilities of the Android OS to application developers so that they can make use of them in their applications.
- **Applications** — At this top layer, you will find applications that ship with the Android device (such as Phone, Contacts, Browser, etc.), as well as applications that you download and install from the Android Market. Any applications that you write are located at this layer.

Android devices in the market

Android devices come in all shapes and sizes. As of late November 2010, the Android OS can be seen powering the following types of devices:

- Smart phones
- Tablets
- E-reader devices
- Net books
- MP4 player

Internet TVs Chances are good that you own at least one of the preceding devices. Figure 1-2 shows (clockwise) the Samsung Galaxy S, the HTC Desire HD, and the LG Optimus One smart phones.

Cont..

- Besides smart phones and tablets, Android is also beginning to appear in dedicated devices, such as e-book readers. Figure 1-4 shows the Barnes and Noble’s NOOKcolor, which is a color e-Book reader running the Android OS.
- In addition to these popular mobile devices, Android is also slowly finding its way into your living room. People of Lava, a Swedish company, has developed an Android-based TV, call the Scandinavia Android TV (see Figure 1-5).
- Google has also ventured into a proprietary smart TV platform based on Android and co-developed with companies such as Intel, Sony, and Logitech. Figure 1-6 shows Sony’s Google TV.

Obtaining the required tools

- Now that you know what Android is and its feature set, you are probably anxious to get your hands dirty and start writing some applications! Before you write your first app, however, you need to download the required tools and SDKs.
- For Android development, you can use a Mac, a Windows PC, or a Linux machine. All the tools needed are free and can be downloaded from the Web.
- Most of the examples provided in this book should work fine with the Android emulator, with the exception of a few examples that require access to the hard- ware.

Android development tools (Adt)

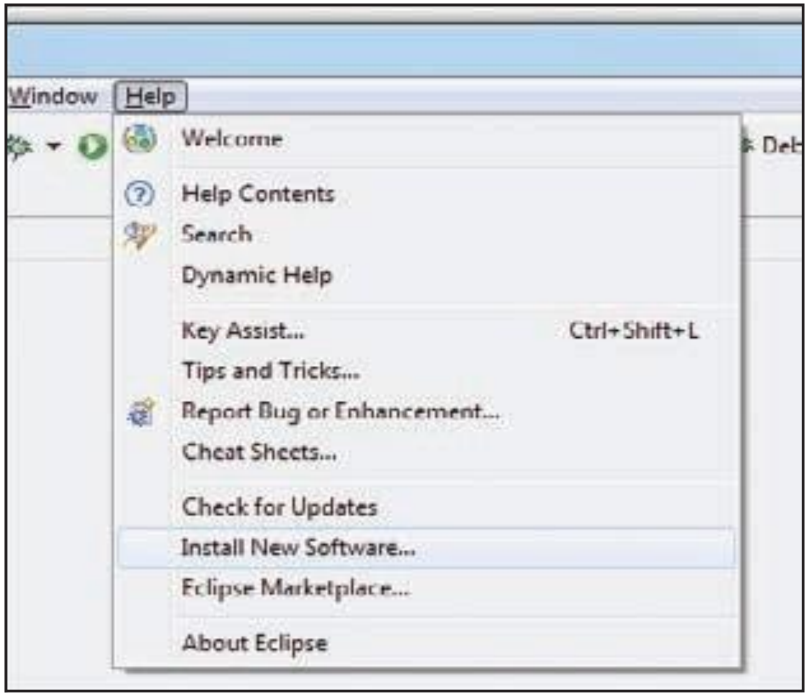
- The Android Development Tools (ADT) plug-in for Eclipse is an extension to the Eclipse IDE that supports the creation and debugging of Android applications. Using the ADT, you will be able to do the following in Eclipse:
- Create new Android application projects.
- Access the tools for accessing your Android emulators and devices.
- Compile and debug Android applications.
- Export Android applications into Android Packages (APK).
- Create digital certificates for code-signing your APK.

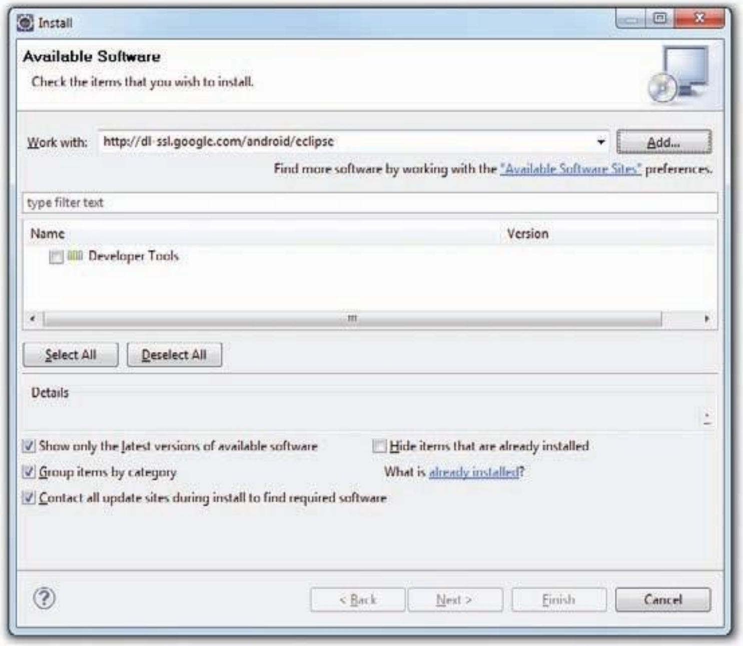
  To install the ADT, first launch Eclipse by double-clicking on the eclipse.exe file located in the eclipse folder

Cont..

- When Eclipse is first started, you will be prompted for a folder to use as your workspace. In Eclipse, a workspace is a folder where you store all your projects. Take the default suggested and click OK.
- Once Eclipse is up and running, select the Help ➪ Install New Software-menu item
- In the Install window that appears, type http://dl-ssl.google.com/android/eclipse in the text box and click Add.
- After a while, you will see the Developer Tools item appear in the middle of the window. Expand it, and it will reveal its content: Android DDMS, Android Development Tools, and Android Hierarchy Viewer. Check all of them and click Next

Cont..





Check the items that you wish to install.

Work with: http://dl-ssl.google.com/android/eclipse

Find more software by working with the "Available Software Sites" preferences.

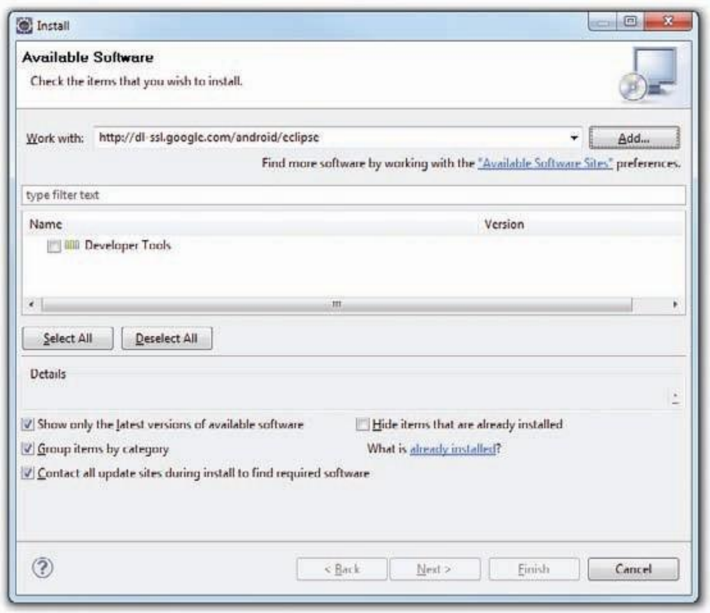
type filter text

Name	Version
Developer Tools	

[✓] Show only the latest versions of available software
[ ] Hide items that are already installed
[✓] Group items by category
What is already installed?
[✓] Contact all update sites during install to find required software

Cont..

- When you see the installation details, as shown in Figure 1-11, click Next.



Install

Available Software

Check the items that you wish to install.

Work with: http://dl-ssl.google.com/android/eclipse

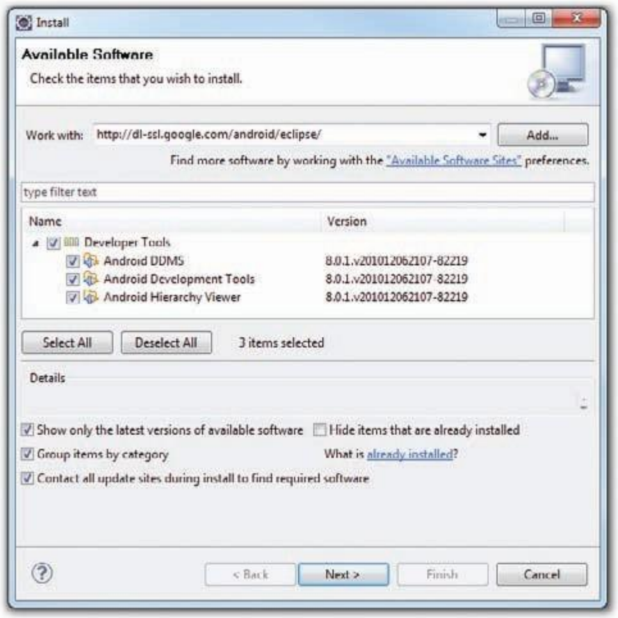
Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
☐ Developer Tools	

Details

☑ Show only the latest versions of available software
☐ Hide items that are already installed
☑ Group items by category
What is already installed?
☑ Contact all update sites during install to find required software



Install

Available Software

Check the items that you wish to install.

Work with: http://dl-ssl.google.com/android/eclipse/

Find more software by working with the "Available Software Sites" preferences.

type filter text

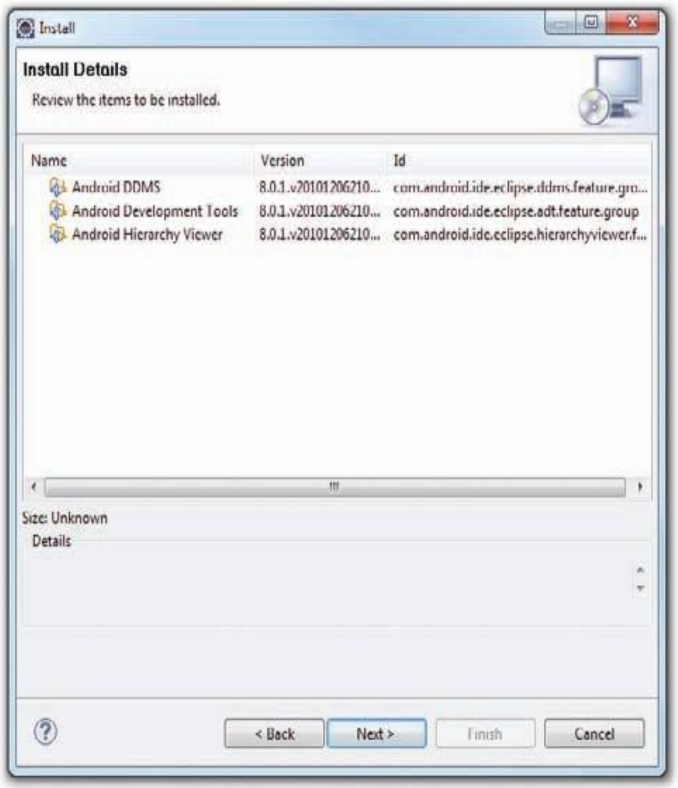
Name	Version
☑ Developer Tools	
☑ Android DDMS	8.0.1.v201012062107-82219
☑ Android Development Tools	8.0.1.v201012062107-82219
☑ Android Hierarchy Viewer	8.0.1.v201012062107-82219

3 items selected

Details

☑ Show only the latest versions of available software
☐ Hide items that are already installed
☑ Group items by category
What is already installed?
☑ Contact all update sites during install to find required software

Cont..


Install

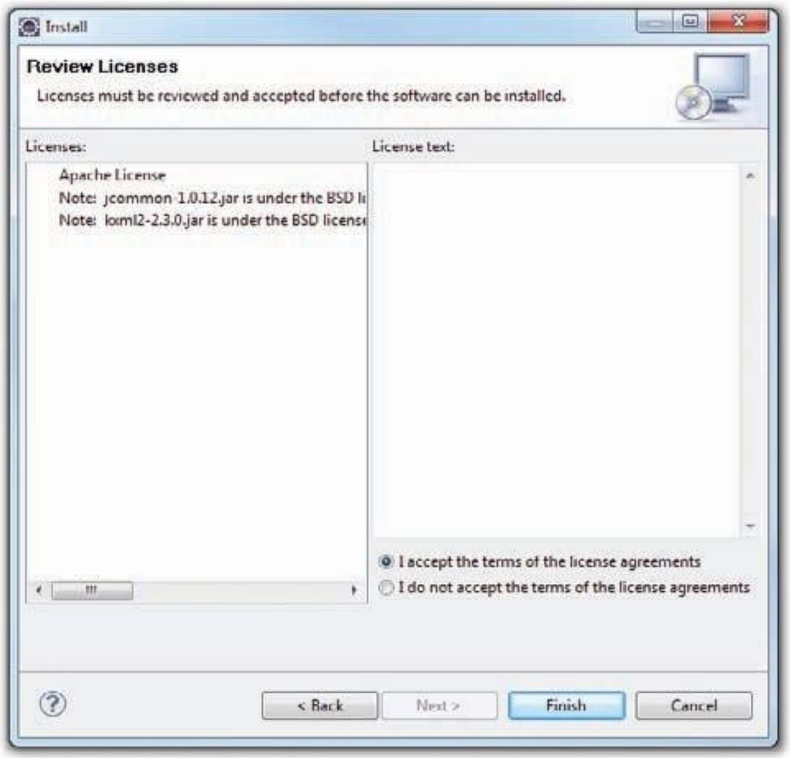
Install Details

Review the items to be installed.

Name	Version	Id
Android DDMS	8.0.1.v20101206210...	com.android.ide.eclipse.ddms.feature.gro...
Android Development Tools	8.0.1.v20101206210...	com.android.ide.eclipse.adt.feature.group
Android Hierarchy Viewer	8.0.1.v20101206210...	com.android.ide.eclipse.hierarchyviewer.f...

Size: Unknown

Details


Install

Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

Licenses:

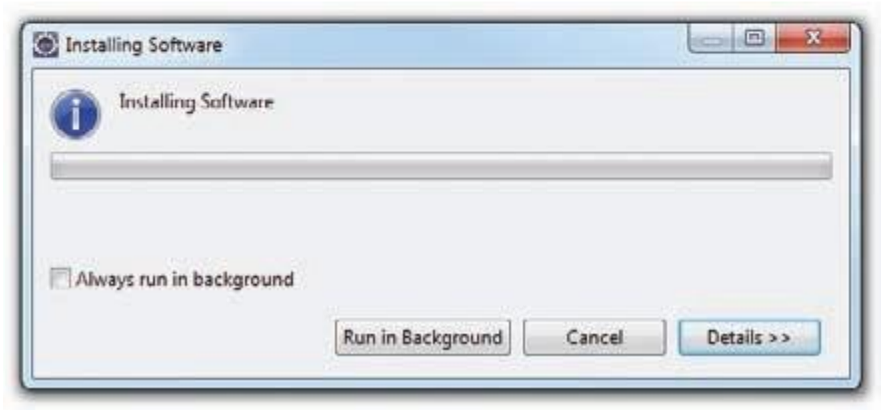
Apache License
Note: jcommon-1.0.12.jar is under the BSD li
Note: ixml2-2.3.0.jar is under the BSD license

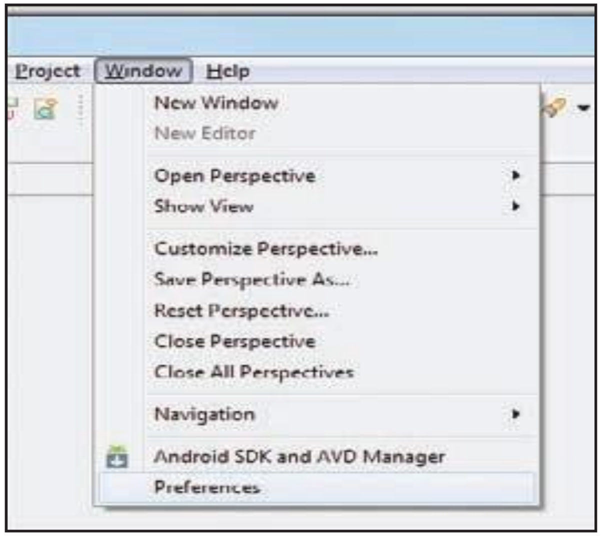
License text:

[✓ I accept the terms of the license agreements]
I do not accept the terms of the license agreements

Eclipse will now proceed to download the tools from the Internet and install them (see Figure 1-13). This will take some time, so be patient

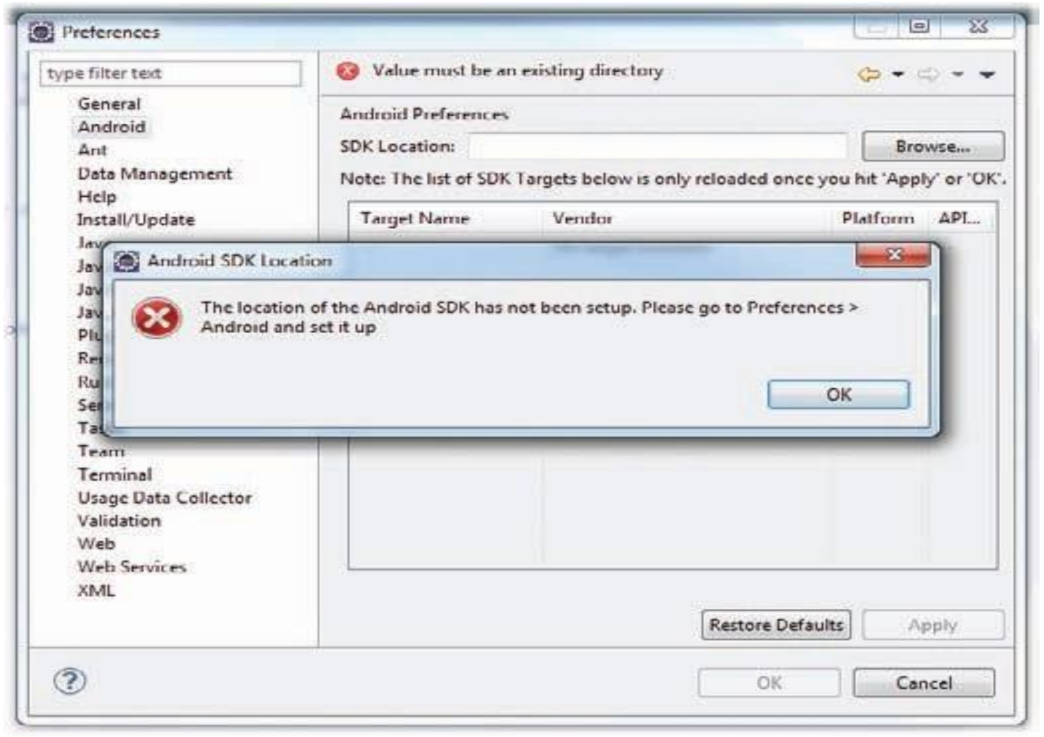
Cont..





Once the ADT is installed, you will be prompted to restart Eclipse. After doing so, go to Window ➪Preferences (see Figure 1-14).

Cont..


Preferences

Value must be an existing directory

Android Preferences

SDK Location:

Note: The list of SDK Targets below is only reloaded once you hit 'Apply' or 'OK'.

Target Name	Vendor	Platform	API...

Android SDK Location

The location of the Android SDK has not been setup. Please go to Preferences > Android and set it up

In the Preferences window that appears, select Android. You will see an error message saying that the SDK has not been set up (see Figure 1-15). Click OK to dismiss it

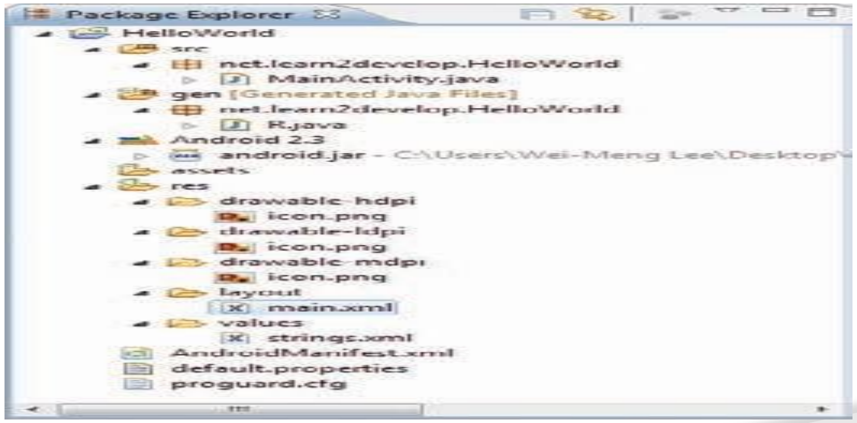
Enter the location of the Android SDK folder. In this example, it would be C:\Android\ android-sdk-windows. Click OK

Anatomy of an Android Application

- Now that you have created your first Hello World Android application, it is time to dissect the innards of the Android project and examine all the parts that make everything work.
- First, note the various files that make up an Android project in the Package Explorer in Eclipse (see Figure 1-30).
- The various folders and their files are as follows:
- src — Contains the .java source files for your project. In this example, there is one file, MainActivity.java. The MainActivity.java file is the source file for your activity. You will write the code for your application in this file.
- Android 2.3 library — This item contains one file, android.jar, which contains all the class libraries needed for an Android application.
- gen — Contains the R.java file, a compiler-generated file that references all the resources found in your project. You should not modify this file

Cont..

- assets — This folder contains all the assets used by your application, such as HTML, text files, databases, etc.
- res — This folder contains all the resources used in your application. It also contains a other subfolders: drawable-<*resolution*>, layout, and values. Chapter 3 talks more about how you can support devices with different screen resolutions and densities.
- AndroidManifest.xml — This is the manifest file for your Android application. Here you spec- ify the permissions needed by your application, as well as other features (such as intent-filters, receivers, etc.).



```
Package Explorer
HelloWorld
  src
    net.learn2develop.HelloWorld
      MainActivity.java
  gen [Generated Java Files]
    net.learn2develop.HelloWorld
      R.java
  Android 2.3
    android.jar - C:\Users\Wei-Meng Lee\Desktop\
  assets
  res
    drawable-hdpi
      icon.png
    drawable-ldpi
      icon.png
    drawable-mdpi
      icon.png
    layout
      main.xml
    values
      strings.xml
  AndroidManifest.xml
  default.properties
  proguard.cfg
```

UNIT-2

VUI and Mobile Development

- Voice user interaction represents a unique challenge for mobile developers. As human beings, we are social, vocal creatures, which means that using our voice to communicate should be natural. However, voice interaction is still a significant departure from the keyboards and touchscreens that you might be accustomed to using. All the traditional guidelines for designing for iOS or Android, or graphical user interfaces in general, have very little relevance to VUIs.
- One of the biggest challenges of VUI design is understanding your users' expectations. Human beings have certain innate assumptions about how communication takes place between two speakers, which can often lead to ambiguity as the computer struggles to compensate for this missing context. For example, a query of "One lump or two?" would be perfectly understandable to a person asking for tea, but could easily trip up a virtual assistant.

Cont..

- Companies and developers, especially those with an app already in place, need to think hard about the ways that they can use VUI to truly enhance the user experience.
- When it comes to VUI, it's all too easy to get it wrong by trying to hop on the bandwagon of the newest, hottest technology.
- For example, a company might decide to overhaul its telephone support line by installing a VUI just because it's "easier" for callers, but fail to consider how callers will interact differently with the new VUI than they would with the old push-button keypad interface.
- Instead, consider how you can use VUI to supplement your existing brand and make it more approachable and helpful.

Cont..

Of course, good VUI design is much harder than it looks. Some of the practices to avoid while building your VUI include:

- Not asking the user a question when the app expects a response.
- Not being clear about the user’s options. For example, “Which sport do you want scores for, football or basketball?” is a better question than “Do you want scores for football or basketball?”, since the user might simply respond “Yes.”
- Giving the user too many choices or being too verbose (for example, “Say ‘football’ for football. Say ‘basketball’ for basketball…”).
- Confirming the user’s query too often. Confirmations should be reserved for important actions such as sending a message or making a purchase.

Text-to-Speech Techniques

- Text-to-speech (TTS) synthesis ultimate goal is to create natural sounding speech from arbitrary text. Moreover, the current trend in TTS research calls for systems that enable production of speech in different speaking styles with different speaker characteristics and even emotions.
- Speech synthesis generally refers to the artificial generation of human voice – either in the form of speech or in other forms such as a song. The computer system used for speech synthesis is known as a speech synthesizer.
- There are several types of speech synthesizers (both hardware based and software based) with different underlying technologies.

Cont..

- The goal of a text-tospeech system is to automatically produce speech output from new, arbitrary sentences. The text-to-speech synthesis procedure consists of two main phrases.
- The first is text analysis, in which the input text is transcribed into a phonetic or some other appropriate representation, and the second is the actual generation of speech waveforms, in which the acoustic output is produced from the information obtained from the first phase.
- Phases of Text -to -speech system The quality of a speech synthesizer is measured based on two primary factors – its similarity to normal human speech (naturalness) and its intelligibility (ease of understanding by the listener).
- Ideally, a speech synthesizer should be both natural and intelligible, and speech synthesis systems always attempt to maximize both characteristics.

UI Design Principles

- When used together, design principles make the UI designer’s job much easier. They remove a lot of the guesswork and make interfaces more predictable and, therefore, easier to use.
- Chris Mears from The UX Review gave us this piece of advice on designing for mobile:
- “One of the main use cases for mobile is killing time. But that doesn’t mean you should waste that of your users. Make sure you understand the main tasks they want to accomplish on your app through research and make those the focus of the interface.”
- Before we go any further, let’s define six of the most common user interface design principles; the structure principle, the simplicity principle, the visibility principle, the feedback principle, the tolerance principle, and finally, the reuse principle.

Cont..

The Structure Principle

- Design should organize the user interface purposefully, in meaningful and useful ways based on clear, consistent models that are apparent and recognizable to users, putting related things together and separating unrelated things, differentiating dissimilar things and making similar things resemble one another. The structure principle is concerned with overall user interface architecture.

The Simplicity Principle

- The design should make simple, common tasks easy, communicating clearly and simply in the user’s own language, and providing good shortcuts that are meaningfully related to longer procedures

Cont..

The Visibility Principle

- The design should make all needed options and materials for a given task visible without distracting the user with extraneous or redundant information. Good designs don’t overwhelm users with alternatives or confuse them with unnecessary information.

The Feedback Principle

- The design should keep users informed of actions or interpretations, changes of state or condition, and errors or exceptions that are relevant and of interest to the user through clear, concise, and unambiguous language familiar to users.

Cont..

The Tolerance Principle

- The design should be flexible and tolerant, reducing the cost of mistakes and misuse by allowing undoing and redoing, while also preventing errors wherever possible by tolerating varied inputs and sequences and by interpreting all reasonable actions.

The Reuse Principle

- The design should reuse internal and external components and behaviors, maintaining consistency with purpose rather than merely arbitrary consistency, thus reducing the need for users to rethink and remember.

Multichannel and Multimodal UIs

- Multimodality in mobile computing has become a very active field of research in the past few years. Soon, mobile devices will allow smooth and smart interaction with everyday life's objects, thanks to natural and multimodal interactions.
- The multi-modal, multi-channel and multi device notions are presented and are referenced by the name and partial acroynm "multi-DMC". A multi-DMC referential is explained, in order to understand what kind of notions have to be sustained in such systems.
- Next we have three case studies that illustrate the issues faced when proposing systems able to support at the same time different modalities including voice or gesture, different devices, like PC or smart phone and different channels such as web or telephone

Cont..

- Developed at Xerox PARC in 1973 and popularized by the Macintosh, this type of graphical user interfaces is still largely used today, on most computers. However, in recent years, numerous scientific researches focus on post-WIMP interfaces.
- It is no longer limited to a single way of interacting with a computer system, but considering the different solutions to offer user interfaces as natural as possible.
- With the introduction of many types of mobile devices, such as cellphones, Personal Digital Assistant (PDA), pocket PC, and the rise of their capabilities (Wifi, GPS, RFID, NFC...) designing and deploying mobile interactive software that optimize the human-computer interaction has become a fundamental challenge.
- Modern terminals are natively equipped with many input and output resources needed for multimodal interactions, suchas camera, vibration, accelerometer, stylus, etc

Cont..

Even if there is a risk of overlapping categories, design issues are often classified into management, technical and social issues.

- Management: mainly deals with registration and later identification of users and devices as they enter and leave the workspace environment.
- Technical: issue occurs with the control of specific device features and also the technical management of services offering the possibility to introduce (discover) or remove specific components from an interaction. The design problems is then related to fusion (i.e. combining multiple input types) and fission (i.e. combining multiple output types) mechanisms, synchronization and rules management between heterogeneous devices.
- Social issues are more related to social rules and privacy matters. As we know, some devices are inherently unsuitable for supporting privacy, such as microphones, speakers and public displays.

Storing and Retrieving Data

- Android provides several options for you to save your app data. The solution you choose depends on your specific needs, such as how much space your data requires, what kind of data you need to store, and whether the data should be private to your app or accessible to other apps and the user.
- Introduces the different data storage options available on Android:

Internal file storage: Store app-private files on the device file system.

External file storage: Store files on the shared external file system.

This is usually for shared user files, such as photos.

Shared preferences: Store private primitive data in key-value pairs.

Databases: Store structured data in a private database

Internal storage

- By default, files saved to the internal storage are private to your app, and other apps cannot access them (nor can the user, unless they have root access). This makes internal storage a good place for internal app data that the user doesn't need to directly access.
- The system provides a private directory on the file system for each app where you can organize any files your app needs.
- When the user uninstalls your app, the files saved on the internal storage are removed. Because of this behavior, you should not use internal storage to save anything the user expects to persist independently of your app.
- So you should instead use the MediaStore API to save those types of files to the appropriate media collection

Internal cache files

- To keep some data temporarily, rather than store it persistently, you should use the special cache directory to save the data. Each app has a private cache directory specifically for these kinds of files.
- When the device is low on internal storage space, Android may delete these cache files to recover space. However, you should not rely on the system to clean up these files for you.
- You should always maintain the cache files yourself and stay within a reasonable limit of space consumed, such as 1MB. When the user uninstalls your app, these files are removed

External storage

- Every Android device supports a shared "external storage" space that you can use to save files. This space is called external because it's not guaranteed to be accessible—it is a storage space that users can mount to a computer as an external storage device, and it might even be physically removable (such as an SD card).
- Files saved to the external storage are world-readable and can be modified by the user when they enable USB mass storage to transfer files on a computer.
- So before you attempt to access a file in external storage in your app, you should check for the availability of the external storage directories as well as the files you are trying to access.

Shared preferences

- If you don't need to store a lot of data and it doesn't require structure, you should use SharedPreferences. The SharedPreferences APIs allow you to read and write persistent key-value pairs of primitive data types: booleans, floats, ints, longs, and strings.
- The key-value pairs are written to XML files that persist across user sessions, even if your app is killed. You can manually specify a name for the file or use per-activity files to save your data.
- The API name "shared preferences" is a bit misleading because the API is not strictly for saving "user preferences," such as what ringtone a user has chosen. You can use SharedPreferences to save any kind of simple data, such as the user's high score. However, if you *do* want to save user preferences for your app, then you should read how to create a settings UI, which uses the AndroidX Preference Library to build a settings screen and automatically persist the user's settings.

Databases

- Android provides full support for SQLite databases. Any database you create is accessible only by your app. However, instead of using SQLite APIs directly, we recommend that you create and interact with your databases with the Room persistence library.
- The Room library provides an object-mapping abstraction layer that allows fluent database access while harnessing the full power of SQLite.
- Although you can still save data directly with SQLite, the SQLite APIs are fairly low-level and require a great deal of time and effort to use.
- There is no compile-time verification of raw SQL queries

Synchronization and Replication of Mobile Data

- Synchronization and Replication The following scenario will show, how mobile computing, synchronization and replication are linked to each other. Imagine a sales person, who is travelling from customer to customer and is collecting orders from them.
- The sales person is collecting all data on a laptop where he is also able to access data that indicates how long an order will take to be delivered and provides the possibility to calculate customer and order specific conditions.
- When the sales person is at a customer site, most of the time communication between the laptop and the central sales person’s corporate database, where all required data should be entered or accessed, is not possible.

Cont..

- This is where data replication comes in. As already stated, the needed data has to be copied onto the sales person’s device in order to provide the required functionality.
- On a daily basis the sales person needs to send the new orders to the central system by any means of communication link or medium.
- Mobile databases appear and disappear over time in a network of connected and stationary databases as well as other mobile databases.
- If more than one process at a time tries to modify a data item, it is necessary to ensure that no two processes access the data item at the same time. Accesses to the data item are put into a sequential order, which is called process synchronization.[

Working with a Content Provider

Content provider basics

- A content provider manages access to a central repository of data. A provider is part of an Android application, which often provides its own UI for working with the data.
- However, content providers are primarily intended to be used by other applications, which access the provider using a provider client object.
- Together, providers and provider clients offer a consistent, standard interface to data that also handles inter-process communication and secure data access.
- Typically you work with content providers in one of two scenarios; you may want to implement code to access an existing content provider in another application, or you may want to create a new content provider in your application to share data with other applications.

Cont..

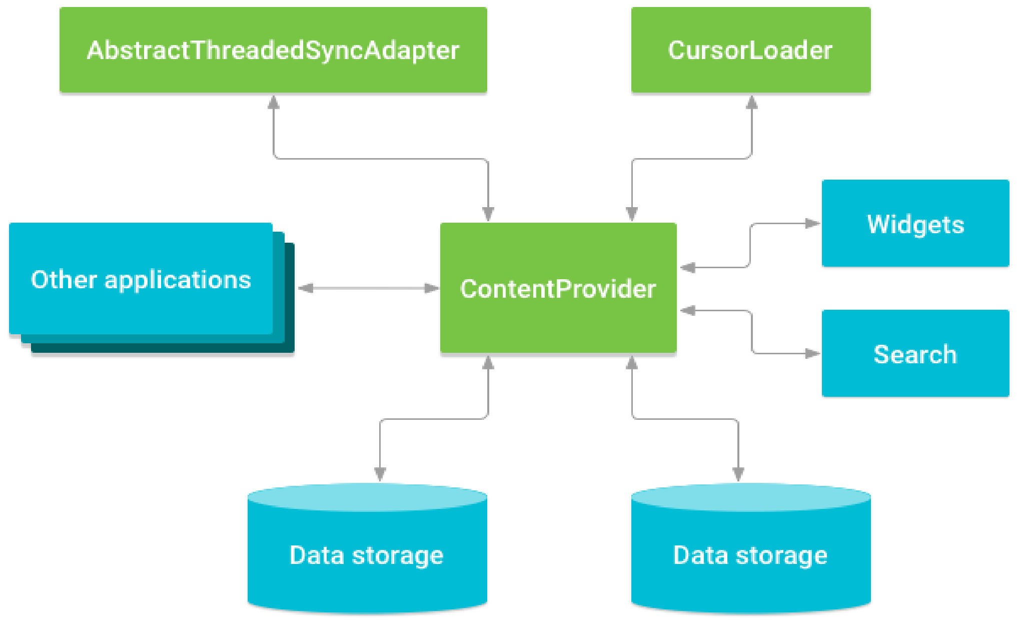


Figure 1. Relationship between content provider and other components.

Accessing a provider

- When you want to access data in a content provider, you use the ContentResolver object in your application'sContext to communicate with the provider as a client. The ContentResolver object communicates with the provider object, an instance of a class that implements ContentProvider. The provider object receives data requests from clients, performs the requested action, and returns the results. This object has methods that call identically-named methods in the provider object, an instance of one of the concrete subclasses of ContentProvider. The ContentResolver methods provide the basic "CRUD" (create, retrieve, update, and delete) functions of persistent storage.
- A common pattern for accessing a ContentProvider from your UI uses a CursorLoader to run an asynchronous query in the background. The Activity or Fragment in your UI call a CursorLoader to the query, which in turn gets the ContentProvider using the ContentResolver

Cont..

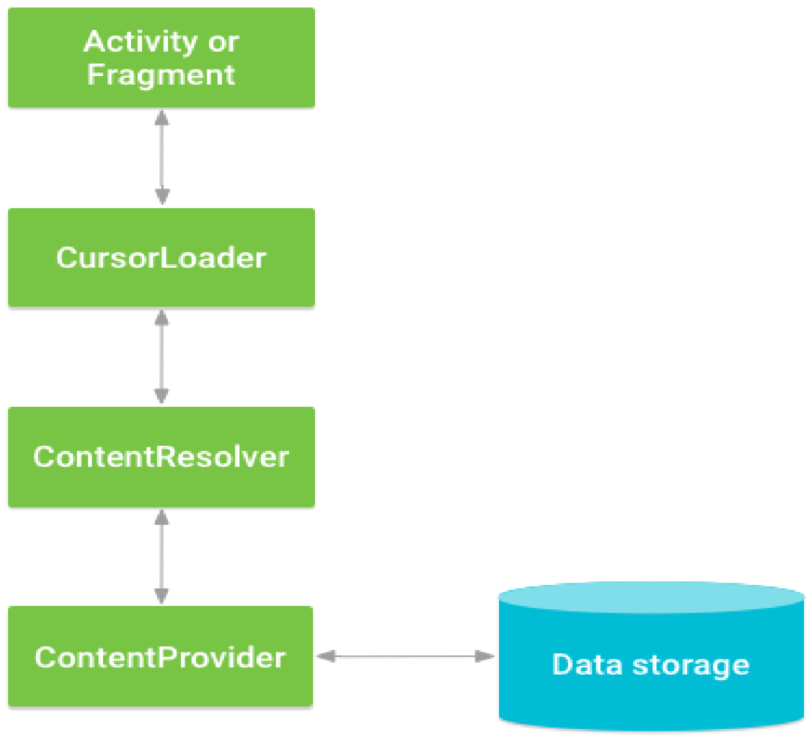


Figure 2. Interaction between ContentProvider, other classes, and storage

UNIT-3

Network And The Web : state Machine

- State machine models can be learned from a set of existing traces (passive learning) or a set of traces that are generated while learning (active learning). The traces are generated by interacting with an application and observing input and output combinations.
- The drawback of passive learning is that the model is incomplete in the variety of existing traces, i.e., when the set of traces does not describe particular application behavior, the inferred model does not specify this behavior either. Active learning overcomes the shortcoming by querying for the information it needs to know.
- Software systems, and therefore also Android applications, can function as a data source to generate the required traces because the applications can respond interactively..

Cont..

- The drawback of applying active learning to software systems is that interacting with an application consumes time, as each input needs to be simulated and each output is required to be observed.
- To improve the learning process, different active learning algorithms have been developed, such as the L* and TTT algorithm.
- Active state machine learning is often implemented according to the Minimally Adequate Teacher (MAT) framework as first proposed by Angluin, which is composed of a learner and a teacher The goal of the learner is to infer the state machine model of a system under test (SUT), by posing membership queries and equivalence queries to the teacher

Correct Communications Model:

- The term "communication" has been derived from the Latin "communis," that means "common"1 . Thus "to communicate" means "to make common" or "to make known", "to share" and includes verbal, non-verbal and electronic means of human interaction.
- Scholars who study communication analyze the development of communication skills in humans and theorize about how communication can be made more effective.
- It is the meaningful exchange of information between two or a group of people. Communicative competence designates the capability to install intersubjective interactions, which means that communication is an inherent social interaction

Cont..

- Humans convey information through a variety of methods: speaking, telephones, email, blogs, TV, art, hand gestures, facial expressions, body language and even social contexts5 . Communication can occur instantaneously in closed, intimate settings or over great periods of time in large public forums, like the Internet.
- However, all forms of communication require the same basic elements: a speaker or sender of information, a message, and an audience or recipient. The sender and recipient must also share a common language or means of understanding each other for communication to be successful.
- As such, a study of communication often examines the development and structure of language, including the mathematical languages used in computer programming. The act of communicating draws on several interpersonal and intrapersonal skills

Cont..

Android Networking and Web

- Android lets your application connect to the internet or any other local network and allows you to perform network operations.
- A device can have various types of network connections. This chapter focuses on using either a Wi-Fi or a mobile network connection.

Checking Network Connection

- Before you perform any network operations, you must first check that are you connected to that network or internet e.t.c. For this android provides **ConnectivityManager** class. You need to instantiate an object of this class by calling **getSystemService()** method this connected states, there are other states a network can achieve

Cont..

They are listed below :

Sr.No	State
1	Connecting
2	Disconnected
3	Disconnecting
4	Suspended
5	Unknown

Cont..

Performing Network Operations

- After checking that you are connected to the internet, you can perform any network operation. Here we are fetching the html of a website from a url.
- Androidprovides **HttpURLConnection** and **URL** class to handle these operations. You need to instantiate an object of URL class by providing the link of website
- this connect method, there are other methods available in HttpURLConnection class

Cont..

They are listed below –

Sr.No	Method & description
1	**disconnect()** This method releases this connection so that its resources may be either reused or closed
2	**getRequestMethod()** This method returns the request method which will be used to make the request to the remote HTTP server
3	**getResponseCode()** This method returns response code returned by the remote HTTP server
4	**setRequestMethod(String method)** This method Sets the request command which will be sent to the remote HTTP server
5	**usingProxy()** This method returns whether this connection uses a proxy server or not

Cont..

To experiment with this example, you need to run this on an actual device on which wifi internet is connected .

Steps	Description
1	You will use Android studio IDE to create an Android application under a package com.tutorialspoint.myapplication.
2	Modify src/MainActivity.java file to add Activity code.
3	Modify layout XML file res/layout/activity_main.xml add any GUI component if required.
4	Modify AndroidManifest.xml to add necessary permissions.
5	Run the application and choose a running android device and install the application on it and verify the results.

Cont..

Bluetooth

- Bluetooth was conceived as a protocol for exchanging data over short distances. It uses the same frequency range as 2.4 GHz Wi-Fi but with 79 channels (for versions below4.0), and 40 channels for 4.0 and above.
- However, unlike most Wi-Fi implementations, it uses a technology called frequency-hopping spread spectrum which regularly changes the channel in order to avoid congestion caused by competing networks.
- Data throughput rates are also a lot higher, ranging from 1 Mbit/s for Bluetooth 1.2 to 24 Mbit/s for 4.0 and above. All iOS and most Android devices support Bluetooth. Some applications of standard Bluetooth include wireless headsets and data tethering

Cont..

Wi-Fi and Cellular Technology

- Most people have used, and are familiar, with these technologies. Cellular networks use radio signals served by fixed-location transceivers (cell base towers) to transmit voice and data and connect mobile devices to the public Internet at speeds that range from the slow 2G to lightning-fast LTE.
- A cellular connection is, generally speaking, more expensive from a cost per Mb perspective, but is more readily available when you're on the move. All Android and iOS phones and select models of iPad and Android tablets support some form of cellular network connectivity.
- Wi-Fi is defined as a wireless local area network connection,which may, or may not, connect to resources like the Internet. These networks usually offer faster transmission than cellular connections and are much less expensive per Mb, or even Free. Virtually all iOS and Android devices support Wi-Fi.

Cont..

Measure UI performance

- In order to improve performance you first need the ability to measure the performance of your system, and then diagnose and identify problems that may arrive from various parts of your pipeline.
- dumpsys is an Android tool that runs on the device and dumps interesting information about the status of system services. Passing the gfx info command to dumpsys provides an output in logcat with performance information relating to frames of animation that are occurring during the recording phase.
- > adb shell dumpsys gfx info <PACKAGE_NAME>This command can produce multiple different variants of frame timing data

Cont..

Aggregate frame stats

- With Android 6.0 (API level 23) the command prints out aggregated analysis of frame data to logcat, collected across the entire lifetime of the process. For example:
- These high level statistics convey at a high level the rendering performance of the app, as well as its stability across many frames.

Precise frame timing info

- With Android 6.0 comes a new command for gfxinfo, and that’s frame stats which provides extremely detailed frame timing information from recent frames, so that you can track down and debug problems more accurately

Cont..

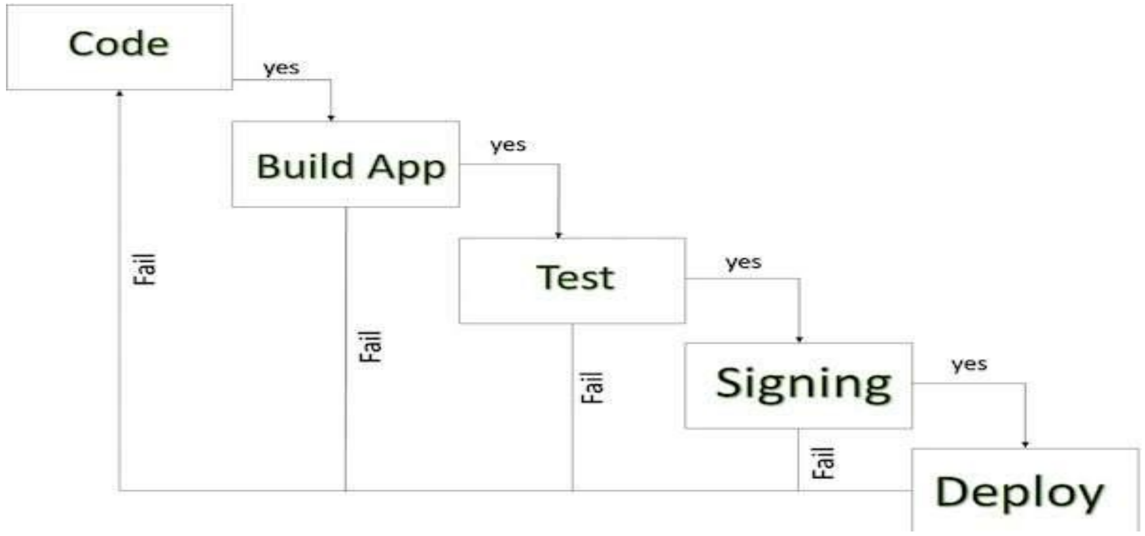
Framestats data format

- Since the block of data is output in CSV format, it's very straightforward to paste it to your spreadsheet tool of choice, or collect and parse with a script. The following table explains the format of the output data columns. All timestamps are in nanoseconds.

UNIT - 4

Putting It All Together And Multimedia

- Android application publishing is a process that makes your Android applications available to users. Infect, publishing is the last phase of the Android application development process.



Android development life cycle

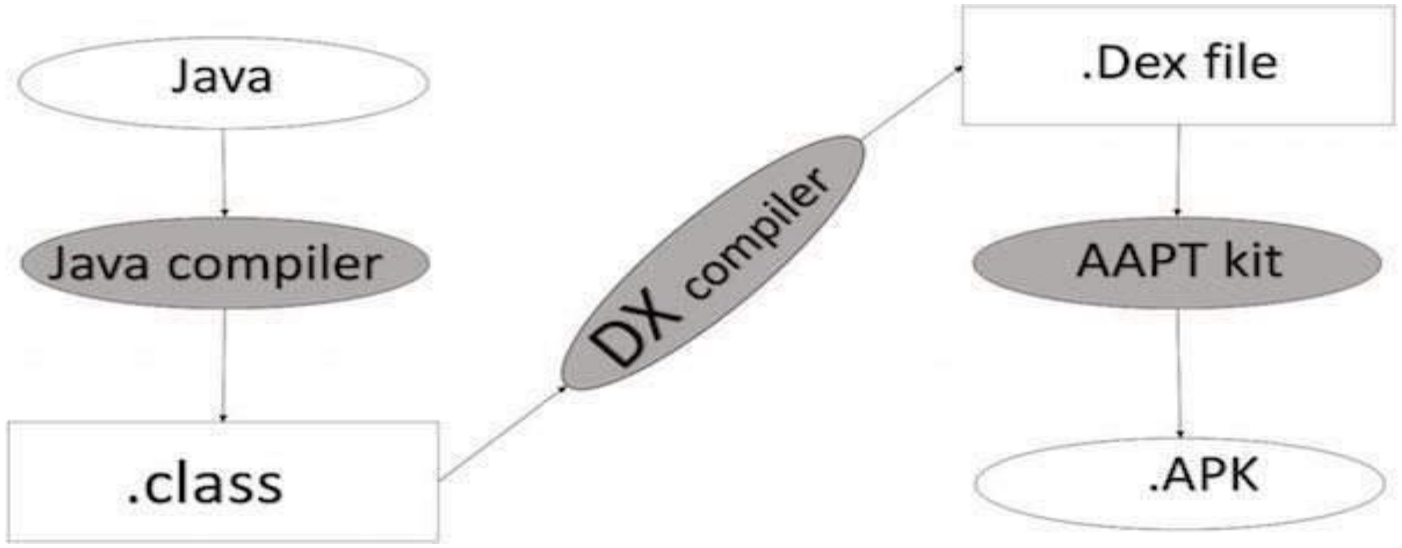
Cont..

Step	Activity
1	**Regression Testing** Before you publish your application, you need to make sure that its meeting the basic quality expectations for all Android apps, on all of the devices that you are targeting. So perform all the required testing on different devices including phone and tablets.
2	**Application Rating** When you will publish your application at Google Play, you will have to specify a content rating for your app, which informs Google Play users of its maturity level. Currently available ratings are (a) Everyone (b) Low maturity (c) Medium maturity (d) High maturity.
3	**Targeted Regions** Google Play lets you control what countries and territories where your application will be sold. Accordingly you must take care of setting up time zone, localization or any other specific requirement as per the targeted region.
4	**Application Size** Currently, the maximum size for an APK published on Google Play is 50 MB. If your app exceeds that size, or if you want to offer a secondary download, you can use APK Expansion Files, which Google Play will host for free on its server infrastructure and automatically handle the download to devices.

Cont..

5	**SDK and Screen Compatibility It is important to make sure that your app is designed to run properly on the Android platform versions and device screen sizes that you want to target.**
6	**Application Pricing** Deciding whether you app will be free or paid is important because, on Google Play, free app's must remain free. If you want to sell your application then you will have to specify its price in different currencies.
7	**Promotional Content** It is a good marketing practice to supply a variety of high-quality graphic assets to showcase your app or brand. After you publish, these appear on your product details page, in store listings and search results, and elsewhere.
8	**Build and Upload release-ready APK** The release-ready APK is what you you will upload to the Developer Console and distribute to users. You can check complete detail on how to create a release-ready version of your app: Preparing for Release.
9	**Finalize Application Detail** Google Play gives you a variety of ways to promote your app and engage with users on your product details page, from colourful graphics, screen shots, and videos to localized descriptions, release details, and links to your other apps. So you can decorate your application page and provide as much as clear crisp detail you can provide.

Export Android Application Process



APK development process

Cont..

Before exporting the apps, you must some of tools

- **Dx tools**(Dalvik executable tools): It going to convert **.class file** to **.dex file**. it has useful for memory optimization and reduce the boot-up speed time
- **AAPT**(Android assistance packaging tool):it has useful to convert **.Dex file** to**.Apk**
- **APK**(Android packaging kit): The final stage of deployment process is called as .apk.
- You will need to export your application as an APK (Android Package) file before you upload it Google Play marketplace.

Cont..

Peer-to-peer networks Peer-to-peer networks are one of the trends in the field of internetworking. In p2p-networks, the hosts, or peers, act as client and server. lists some characteristics for p2p-system. These are

- Decentralization
- Scalability
- Anonymity
- Self-Organization
- Cost of Ownership
- Ad-Hoc connectivity
- Performance
- Security
- Transparency and Usability
- Fault Resilience

Interoperability Probably the most significant of these are Decentralization, Scalability and Ad-Hoc connectivity.

Mobile agents

- Mobile agents are defined as programs that can migrate from host to host. The program is able to save its state, "jump" to another host, and continue the processing in the new location. Mobile agents are also capable of multiplying or cloning themselves.

  Some applications for mobile agents:

  - Parallel computing
  - Data collection
  - E-commerce

Android Multimedia

- Android provides many ways to control playback of audio/video files and streams. One of this way is through a class called **Media Player**.
- Android is providing Media Player class to access built-in media player services like playing audio, video e.t.c. In order to use Media Player, we have to call a static Method **create()** of this class. This method returns an instance of Media Player class.

```
MediaPlayer.start();
mediaPlayer.pause();
```

- On call to **start()** method, the music will start playing from the beginning. If this method is called again after the **pause()** method, the music would start playing from where it is left and not from the beginning.

Cont..

Sr.No	Method & description
1	**isPlaying()** This method just returns true/false indicating the song is playing or not
2	**seekTo(position)** This method takes an integer, and move song to that particular position millisecond
3	**getCurrentPosition()** This method returns the current position of song in milliseconds
4	**getDuration()** This method returns the total time duration of song in milliseconds
5	**reset()** This method resets the media player
6	**release()** This method releases any resource attached with MediaPlayer object

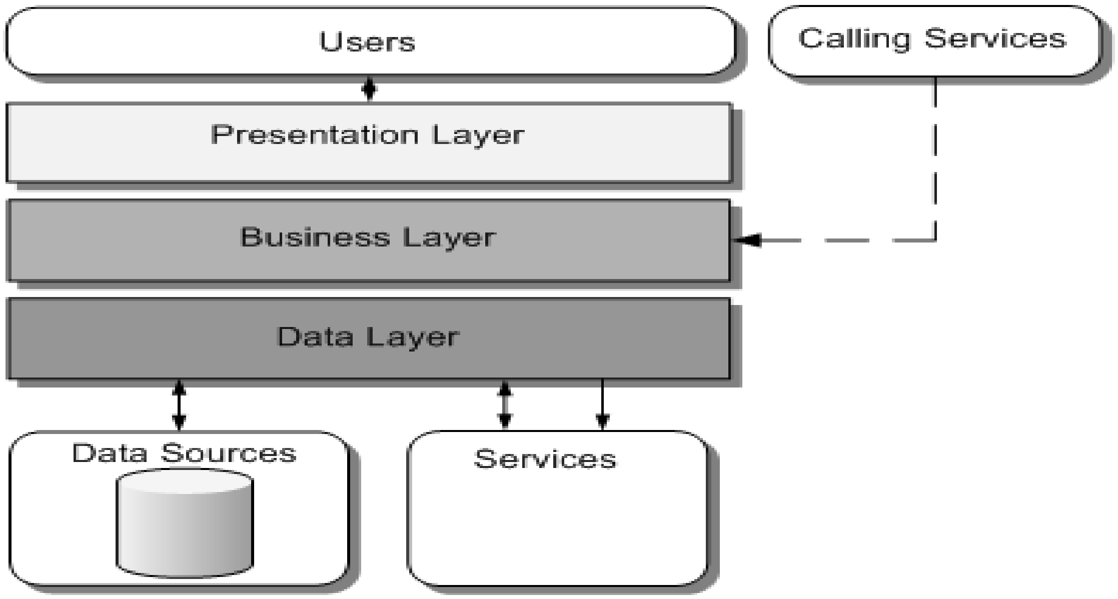
Cont..

7	**setVolume(float leftVolume, float rightVolume)** **This method sets the up down volume for this player**
8	**setDataSource(FileDescriptor fd)** This method sets the data source of audio/video file
9	**selectTrack(int index)** This method takes an integer, and select the track from the list on that particular index
10	**getTrackInfo()** This method returns an array of track information

These methods are listed above −

Platforms And Additional Issues, Security And Hacking

- Application architecture is a set of technologies and models for the development of fully-structured mobile programs based on industry and vendor-specific standards. As you develop the architecture of your app, you also consider programs that work on wireless devices such as smart phones and tablets.



Cont..

Mobile app architecture design usually consists of multiple layers, including:

- **Presentation Layer** - contains UI components as well as the components processing them.
- **Business Layer** - composed of workflows, business entities and components.
- **Data layer** - comprises data utilities, data access components and service agents.

Things to consider before attempting mobile app architecture development

As building a better application architecture is crucial to the success of your project, there are several things to keep in mind before you start designing your app architecture:

Cont..

- There are different types of smart phones and it is important to evaluate the device type and its characteristics before choosing a specific app architecture. You should keep in mind the following device features:
- Screen resolution
- Screen size
- CPU Features
- Storage Space
- Memory
- Availability of the development framework

Android Hacking Applications

- There are several popular applications that are used by developers to hack Android devices to make them faster, increase battery life, and customize screensavers, ringtones, alerts, and more.

  The list of hacks available to make improvements to an Android is large and growing every day. Tweaks or hacks can be either surface ones or the deep-system kind, depending on what the hack can do.

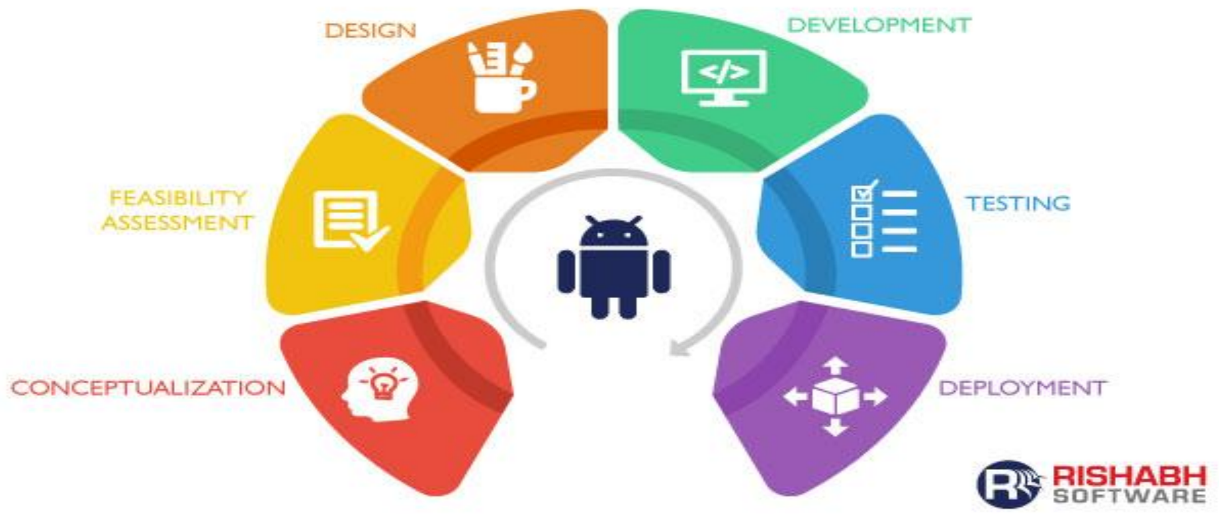
  Popular surface tweaks or hacks are:
- Tusker - for location based automation
- Ability to install custom keyboards like Swype and Swift Key
- Deep system tweaks include downloading new kernels and radios to increase speed and battery life

UNIT-V

Platforms And Additional Issues, Security And Hacking

- Android is one of the most popular operating systems for mobile devices. It is the operating system of choice for companies across developing markets.
- Android app development process is a sum-mean of critical phases. It comprises designing, creation, development, and post-development.
- To develop a successful Android app a robust and fail-proof mobile app development process is much required. With this article, we aim to share some of those proven strategies.
- It helps create opportunities for Android developers to create unique and powerful apps.

Cont..



Here are some of the app development process guidelines for developing Android applications.

Android Application Development Process Steps:

Conceptualization – Refining the app idea into a solid base of the application is the first and most significant stage in the development process of Android application.

- The initial analysis of the app must include the demographics, behavior patterns, and goals of the buyer persona as all the other stages of app creation will depend on the said traits of users.
- During this stage, all the necessary groundwork for the following process is laid down. It is beneficial to do substantial research and brainstorming before jumping to the next step.
- Another pivotal part of this stage is the competitor analysis to figure out what features can make the app stand out in the market

Cont..

Feasibility Assessment – Enterprises can gain a clear understanding of the app visuals through wireframes, detailed sketches of the conceptualized product to refine their ideas and arrange design components in a precise manner.

To assess whether the concept of the application is technically feasible or not, the app developers need access to public data through public APIs sourcing.

By the end of feasibility testing, the team may have a completely different app idea if their original functionality is not feasible.

Cont..

- **Design** – In this phase of the Android development, the UX (user experience) designer architects the design elements' interaction, while the UI (user interface) designer builds the app's persona by keeping in mind the modern user's preference.

  Application designing is a multi-step process for drawing clear visual directions and offering an abstract of the final product.

- **Development** – During this stage, a working prototype is developed to validate the functionality, assumptions, and understanding of the project scope. The app goes through a broad set of steps as the development progresses from core functionality development to light testing and then releasing the app for an external group of users for further field testing the concept.

  If an application has a broader scope than the usual, the creation process gets divided into smaller modules through agile methodology, and the entire mobile app development process is applicable for each of these small parts.

Cont..

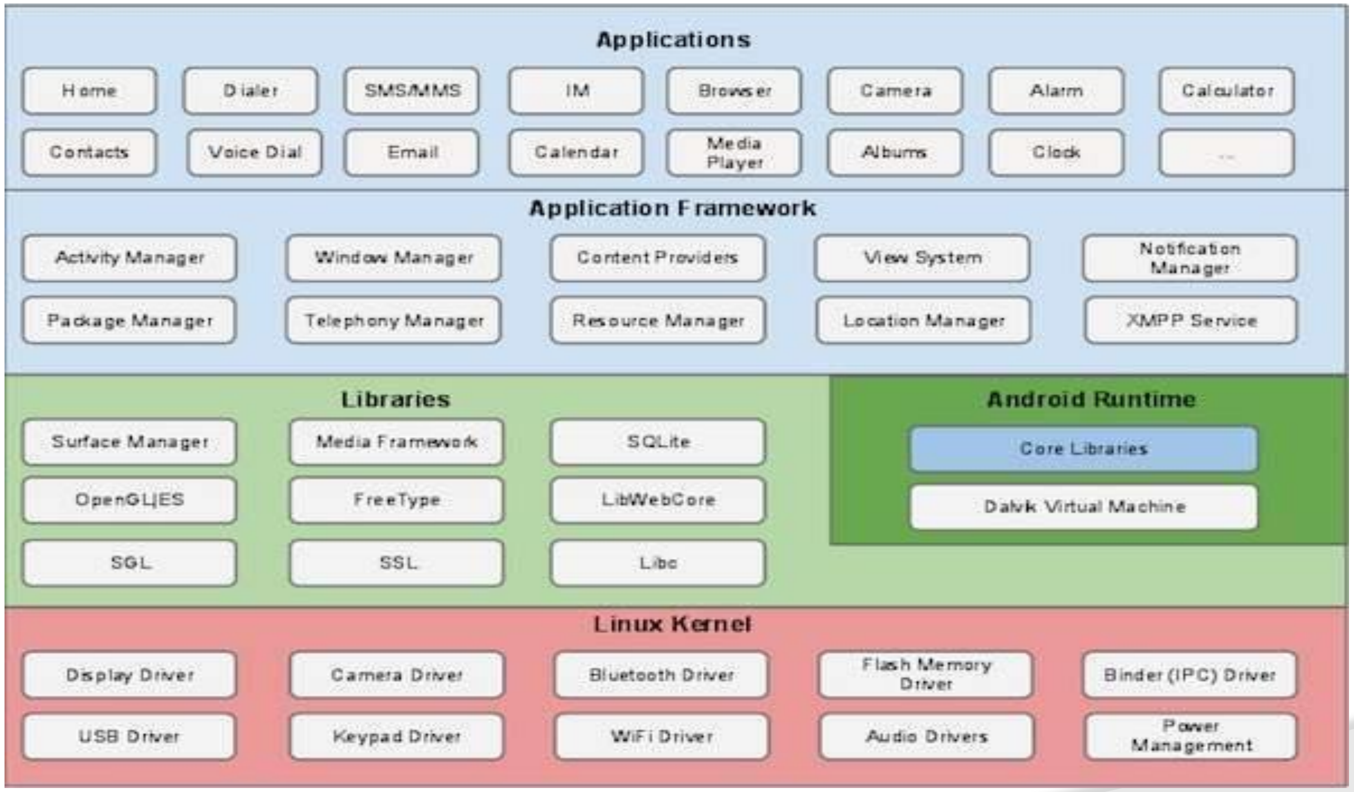
Testing & Deployment – One of the critical components of the app development process, it is a good idea to test at early stages, often for usability, interface & security checks, stress, compatibility, and performance.

After fixing the bugs, the app moves to the deployment phase and is ready for release via a formal launch.

Different application stores have different policies of application launch, and therefore the deployment phase plan is aligned according to the app store. However, keep in mind that this is not the end.

Architecture:

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram



Cont..

Linux kernel

- At the bottom of the layers is Linux - Linux 3.6 with approximately 115 patches. This provides a level of abstraction between the device hardware and it contains all the essential hardware drivers like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.

Libraries

- On top of Linux kernel there is a set of libraries including open-source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.

Cont..

Android Libraries

- This category encompasses those Java-based libraries that are specific to Android development. Examples of libraries in this category include the application framework libraries in addition to those that facilitate user interface building, graphics drawing and database access. A summary of some key core Android libraries available to the Android developer is as follows −
- **android.app** − Provides access to the application model and is the cornerstone of all Android applications.
- **android.content** − Facilitates content access, publishing and messaging between applications and application components.
- **android.database** − Used to access data published by content providers and includes SQLite database management classes.

Cont..

- **android.opengl** − A Java interface to the OpenGL ES 3D graphics rendering API.
- **android.os** − Provides applications with access to standard operating system services including messages, system services and inter-process communication.
- **android.text** – Used to render and manipulate text on a device display.
- **android.view** − The fundamental building blocks of application user interfaces.
- **android.widget** − A rich collection of pre-built user interface components such as buttons, labels, list views, layout managers, radio buttons etc.
- **android.webkit** − A set of classes intended to allow web-browsing capabilities to be built into applications.

Android Runtime

- This is the third section of the architecture and available on the second layer from the bottom. This section provides a key component called **Dalvik Virtual Machine** which is a kind of Java Virtual Machine specially designed and optimized for Android.
- The Dalvik VM makes use of Linux core features like memory management and multi-threading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine.
- The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language

Cont..

Application Framework

The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.

The Android framework includes the following key services −

- **Activity Manager** − Controls all aspects of the application lifecycle and activity stack.
- **Content Providers** − Allows applications to publish and share data with other applications.
- **Resource Manager** − Provides access to non-code embedded resources such as strings, color settings and user interface layouts.
- **Notifications Manager** − Allows applications to display alerts and notifications to the user.
- **View System** − An extensible set of views used to create application user interfaces.

Android Hacking Applications

There are several popular applications that are used by developers to hack Android devices to make them faster, increase battery life, and customize screensavers, ringtones, alerts, and more.

The list of hacks available to make improvements to an Android is large and growing every day. Tweaks or hacks can be either surface ones or the deep-system kind, depending on what the hack can do. Popular surface tweaks or hacks are:

- Tusker - for location based automation
- Ability to install custom keyboards like Swype and SwiftKey
- Deep system tweaks include downloading new kernels and radios to increase speed and battery life