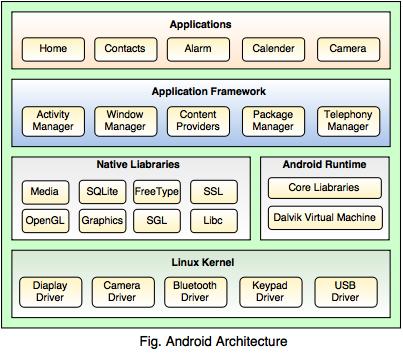
**Android Architecture is categorized into five parts as below:**

The Android software stack consists of a Linux kernel and a collection of C/C++ libraries. They are useful for application framework that provides services and management of the applications and run time.

1. Linux Kernel  
2. Native Libraries  
3. Android Runtime  
4. Application Framework  
5. Applications  
  
Each part of the stack and the elements within each layer are integrated and provide optimal application development and execution environment for mobile devices.  
  
**The following diagram shows the architecture of Android,**  
  
  
  
**1. Linux Kernel**

* Linux is the heart of Android architecture.
* It provides a level of abstraction between the hardware devices and the upper layers of the Android software stack.
* The Android operating system is based on the Linux kernel.
* The Linux kernel is responsible for various device drivers such as Camera driver, Display driver, Bluetooth driver, Keypad driver, Memory management, Process management, Power management, etc.

**2. Native Libraries**

* The native libraries such as Media, WebKit, SQLite, OpenGL, FreeType, C Runtime library (libc) etc. are situated on the top of a Linux kernel.
* Media library is responsible for playing and recording audio and video formats, FreeType is for font support, WebKit is for browser support, SQLite is for database, SSL is for Internet security etc.

**3. Android Runtime**

* Android Runtime is the third section of the architecture and situated on the second layer from the bottom.
* Android Runtime includes core libraries and Dalvik Virtual Machine (DVM) which is responsible to run android application.
* Dalvik Virtual Machine (**DVM**) is like Java Virtual Machine (**JVM**) in Java, but DVM is optimized for mobile Devices.
* DVM makes use of the Linux core features like memory management and multi-threading, which are essential in the Java language.
* DVM provides fast performance and consumes less memory.

**4. Application Framework**

* Application framework is situated on the top of the Native libraries and Android runtime.
* Android framework provides a lot of classes and interfaces for Android application development and higher level services to the applications in the form of Java classes.
* It includes Android API's such as Activity manager, Window manager, Content Provider, Telephony Manager, etc.
* Activity manger is responsible for controlling all the aspects of the application lifecycle and activity stack, Content provider is responsible for allowing the applications to publish and share the data with the other applications, View system is responsible for creating application user interfaces, etc.

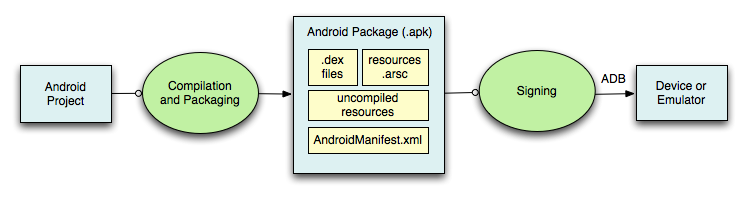
**5. Applications**

* Applications are situated on the top of the Application framework.
* The applications such as Home, Contact, Alarm, Calender, Camera, Browsers, etc. use the Android framework which uses Android runtime and libraries. Android runtime and Native libraries use Linux kernel.
* The user can write his/her application to be installed on this layer only.

**Every Android project contains several folders, like:**

|  |  |
| --- | --- |
| **Folder Name** | **Description** |
| src | The 'src' stands for **Source Code.** It contains the Java Source files. |
| gen | The 'gen' stands for **Generated Java Library.** This library is for Android internal use only. |
| Android 2.2 | The Android Framework Library is stored here. |
| assets | It is used to store raw asset files. |
| libs | It contains private libraries. |
| res | The 'res' stands for **Resource file.** It can store resource files such as pictures, XML files, etc. It contains some additional folders such as Drawable, Layout and Values.  **there are three types of drawable folders,** 1. drawable-mdpi 2. drawable-hdpi 3. drawable-ldpi  The above drawable folders are required in order to adapt to different screen resolutions. |
| AndroidManifest.xml | This file indicates the Android definition file. This file contains the information about the Android application such as minimum Android version, permission to access Android device capabilities such as Internet access permission, phone permission etc. |
| MainLayout.xml | This file describes the layout of the page. So all the components such as textboxes, labels, radio buttons, etc. are displaying on the application screen. |
| Activity class | The application occupies the entire device screen which needs at least one class inherits from the Activity class. OnCreate() method initiates the application and loads the layout page. |

**Building An Application**

[](http://blog.appliedinformaticsinc.com/wp-content/uploads/2015/11/ZnJvbT1jc2RuJnVybD1jbWJ3NUNabGxtWnB4R2N0bDJjdFFHYnBWbll2TVhabkZXYnA5U2J2Tm1Ma2wyYnlSbWJoNWljbEIzYnNWbWRsUjJMdm9EYzBSSGE.png)

This flow diagram shows the process that depicts our application.

**Android Project**: This contains our source code as well as non source code that make up the application.

**Compilation And Packaging:**Compiles our source code and prepares the resources.

**Apk:**Output of compilation and packaging process is Apk (Android package). An APK file contains all of the codes of program (such as .dex files), resources, assets, certificates, and manifest file.

**Signing**: APK is digitally signed to identify developer.

**Device or Emulator:**Finally your application is installed in your device or emulator.