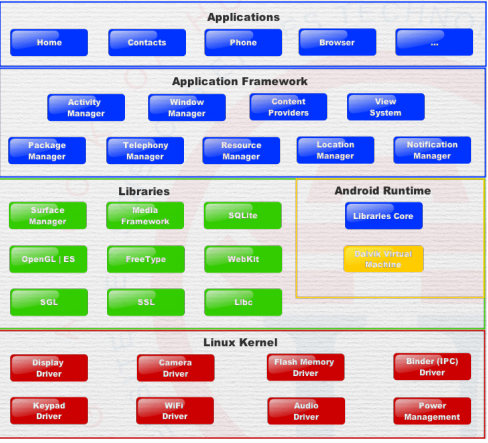
**ANDROID FUNDAMENTAL**

**Chapter Objectives**

Understand content of MVC model and some components od Android

1. **Architecture:**

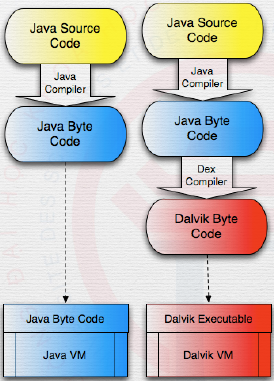


* Application:
  + Written in java
  + Example: Phone, Browser,…
* Application framework:
  + Java, Higher level, User Interface, Location service, Notation,..
  + Window manager, Resource Manager
* Libraries:
  + Mostly in C/C++, Low level, Render text Play media, Local databases
  + OpenCL, SQLite, …
* Linux Kernel:
  + Active development, Secured, Well shaped
  + Display driver, Audio driver

1. **Compilation:**

* Java source code, Java Complier: complier code and run everywhere or on different platform.

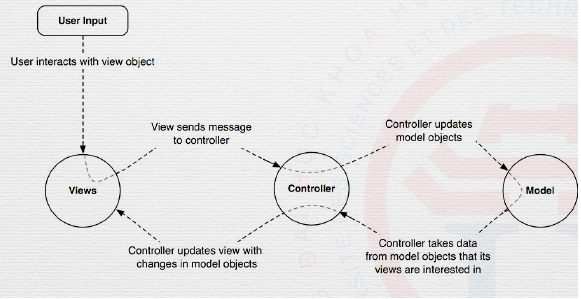
From Source to device:



* Example: Dalvik, ART

1. **Controller:**
2. **Context and Application:**

* MVC Model:



* Context:
  + Central command center
  + Access application-specific data •
    - Settings
    - Private files
    - Resources
    - Assets
  + System services
* Application:
  + A context
  + Can be subclasses:

Example: Global data, early initialization of libraries

* + Android memory management

Example: Garbage collector, Upper limit for each application

* + AndroidManifest.xml

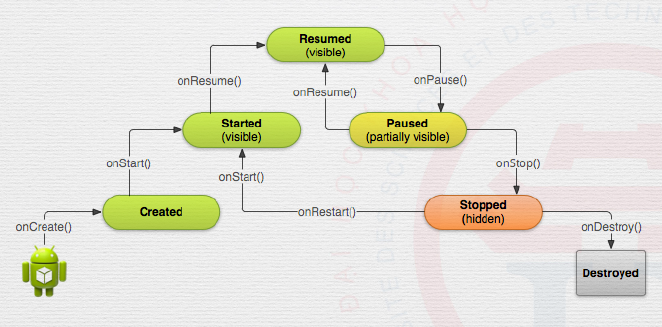
Example: Metadata about the app, target SDK

1. **Activity:**

* A kind of controller - mean in the middle of model and view, update model

to UI

* Activity:
  + fundamental building block
  + Has a unique task or purpose
  + Need at least one per application
  + Handle display of single screen



Activity lifecycle

Activity lifecycle: states different from webpage (all content cleared when closed)

* onCreate() : initialization
* onStart(): visible state
* onPause(): do not have to override (just cases you need)
* onStop()
* onResume(): continue
* Screen orientaion
  + onSaveInstanceState()
  + onDestroy() - will be called if no memory leak
* Create a new activity instance
  + onCreate()
  + onRestoreInstanceState()
* Close current activity: finish()

1. **Fragment:**

* Fragment is used to:
  + Represents a behavior or a portion of user interface
  + Building block of the Fundamental building blocks
  + Supported from Honeycomb [API 11]
  + Optional
* Functional: Adapt UI according to devices - explosion in the variety of devices, screen size, resolution, density, orientation.
* Activity with fragments
  + Simplified
  + Coordinates fragments
  + [optionally, but mostly] uses FragmentManager (or SupportFragmentManager)
  + Put inside a layout XML
  + Dynamically created using codes

1. **View:**

* Basic building blocks of UI - what user interacts with
* Attributes
  + id: findViewById()
  + width, height
  + padding and margin
  + visibility: visible, invisible, non
  + alpha: classic transparent
  + rotation
  + background
  + click
* TextView
  + setText() :can contain one and only one icon
  + drawable, font, gravity, style, align
* ImageView
  + src: setImageResource()
  + scaleType: fitXY, fitStart, fitEnd, centerCrop, centerIn side
  + tint, crop, viewBounds
* View Group
  + Contain children
  + LayoutParams
* Button
  + Push-button
  + onClick()
* EditText
  + TextBoxes: allow to edit a text
  + getText()
  + Selection