Mobile Application Development

(Android Overview)

Department of
Computer Science & Engineering

Objectives

- Why Mobile Application Development
- History of Android platform
- Platform Architecture
- Application Building Blocks
- Application Life Cycle
- Development Tools
- Some Views and ViewGroups

Few reasons to become MAD…

- Smart Phones
  - Internet access anywhere
  - Social networking
- Billions of mobile users
- Scope of collecting revenue directly

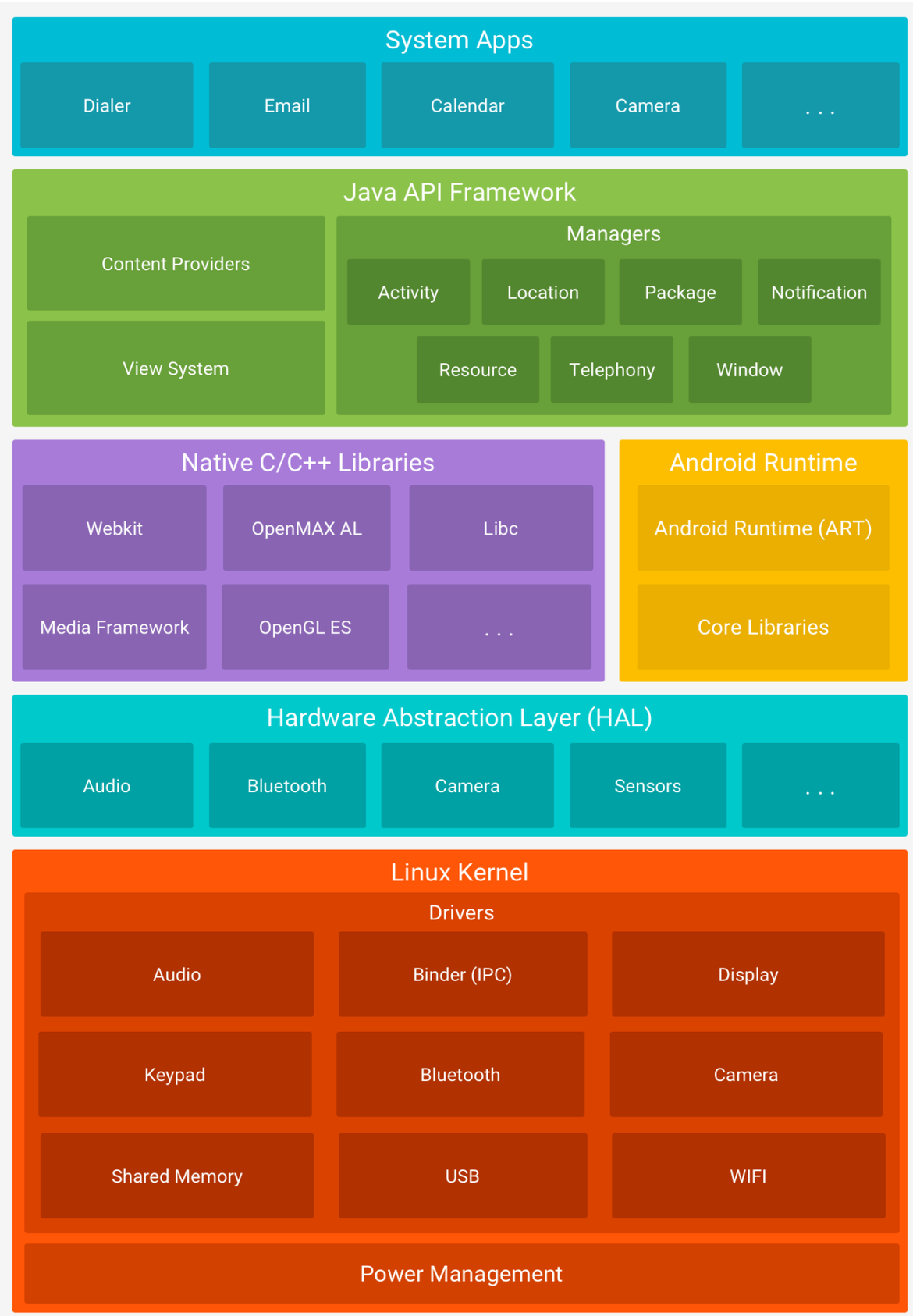
A better prospect for the students who want to built their carrier as a software developer

History of Android Platform

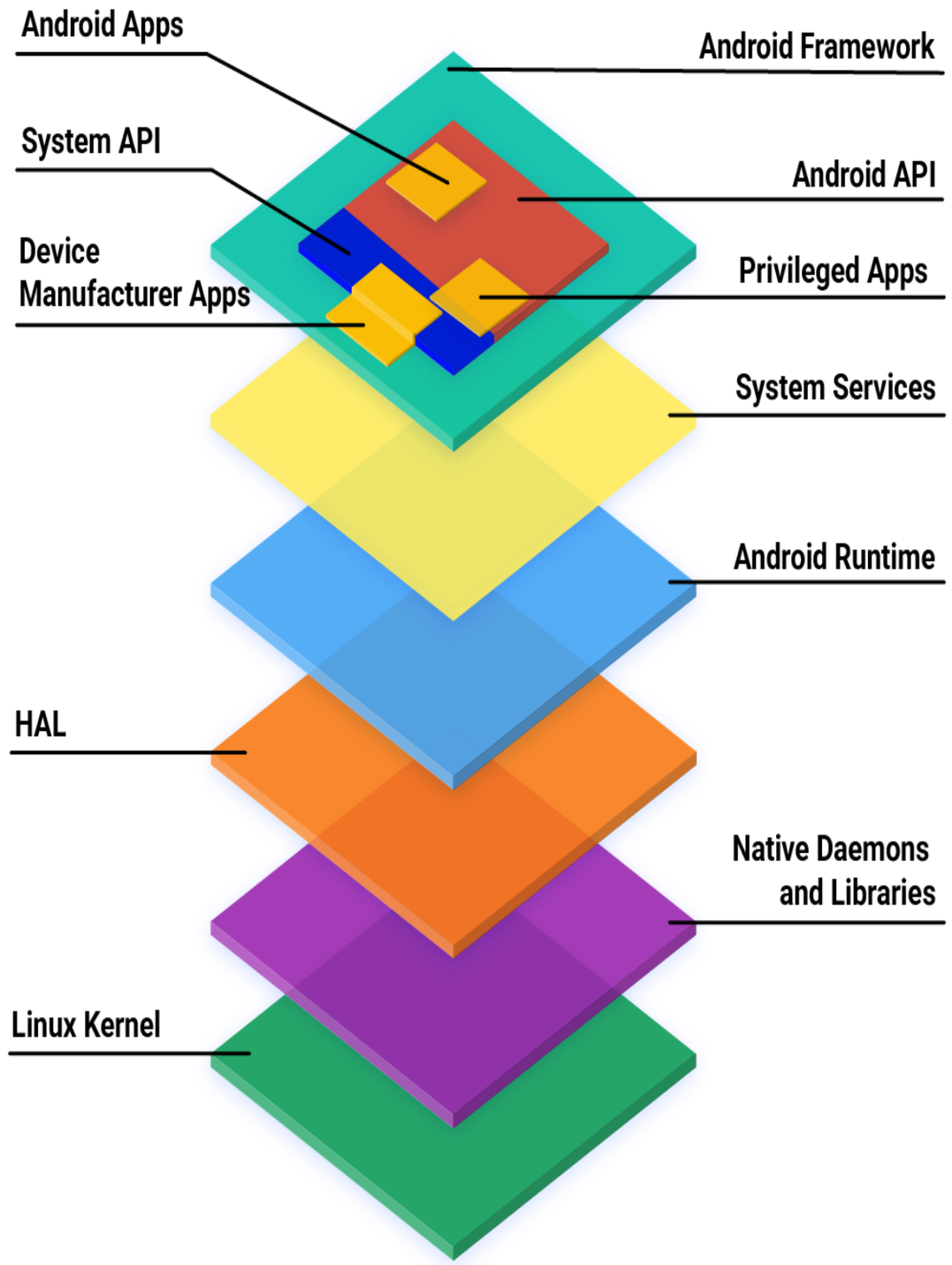
- Began in 2003 as a project of the American technology company Android Inc.
  - to develop an operating system **for digital cameras**
  - In 2004 the project changed to become an operating system **for smartphones**
- Was bought by Google Inc., in 2005 for only $50 million
  - At Google, the Android team decided to base their project on Linux (an open-source operating system for personal computers)

- Fast application development in Java
  - Kotlin – introduced in recent year
- Open source under the Apache 2 license

Platform Architecture



Android Open System Platform (AOSP)



Application Building Blocks

- Activity
- Intent
- IntentReceiver
- Service
- ContentProvider

- **Mandatory Vs. Optional**
  - Consider:
    - Traffic Navigation App
    - Alarm App
    - Notepad App

Activities

- Typically correspond to one UI screen
- But, they can:
  - Be faceless
  - Be in a floating window
  - Return a value

Intents

- Think of Intents as a verb and object; a description of what you want done
  - E.g. VIEW, CALL, PLAY etc..
- System matches Intent with Activity that can best provide the service
- Activities and IntentReceivers describe what Intents they can service

Intents: Example

System picks best component for that action

Client component makes a request for a specific action

New components can use existing functionality

IntentReceivers

- Components that respond to broadcast ‘Intents’
- Way to respond to external notification or alarms
- Apps can invent and broadcast their own Intent

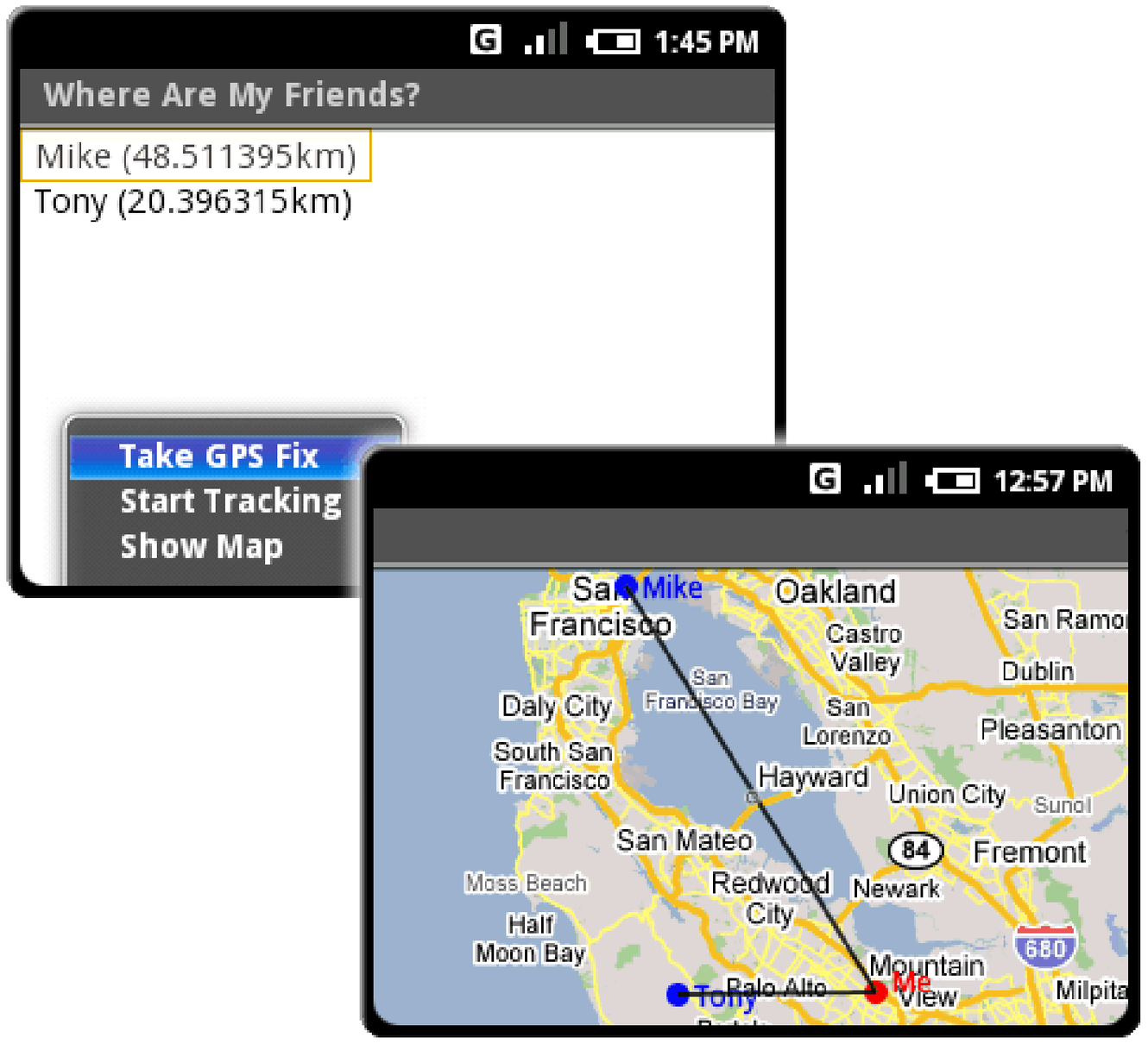
Services

- Faceless components that run in the background
  - E.g. music player, network download etc…

ContentProviders

- Enables sharing of data across applications
  - E.g. address book, photo gallery, GPS
- Provides uniform APIs for:
  - querying
  - delete, update and insert
  - Content is represented by URI and MIME type

Location Manager

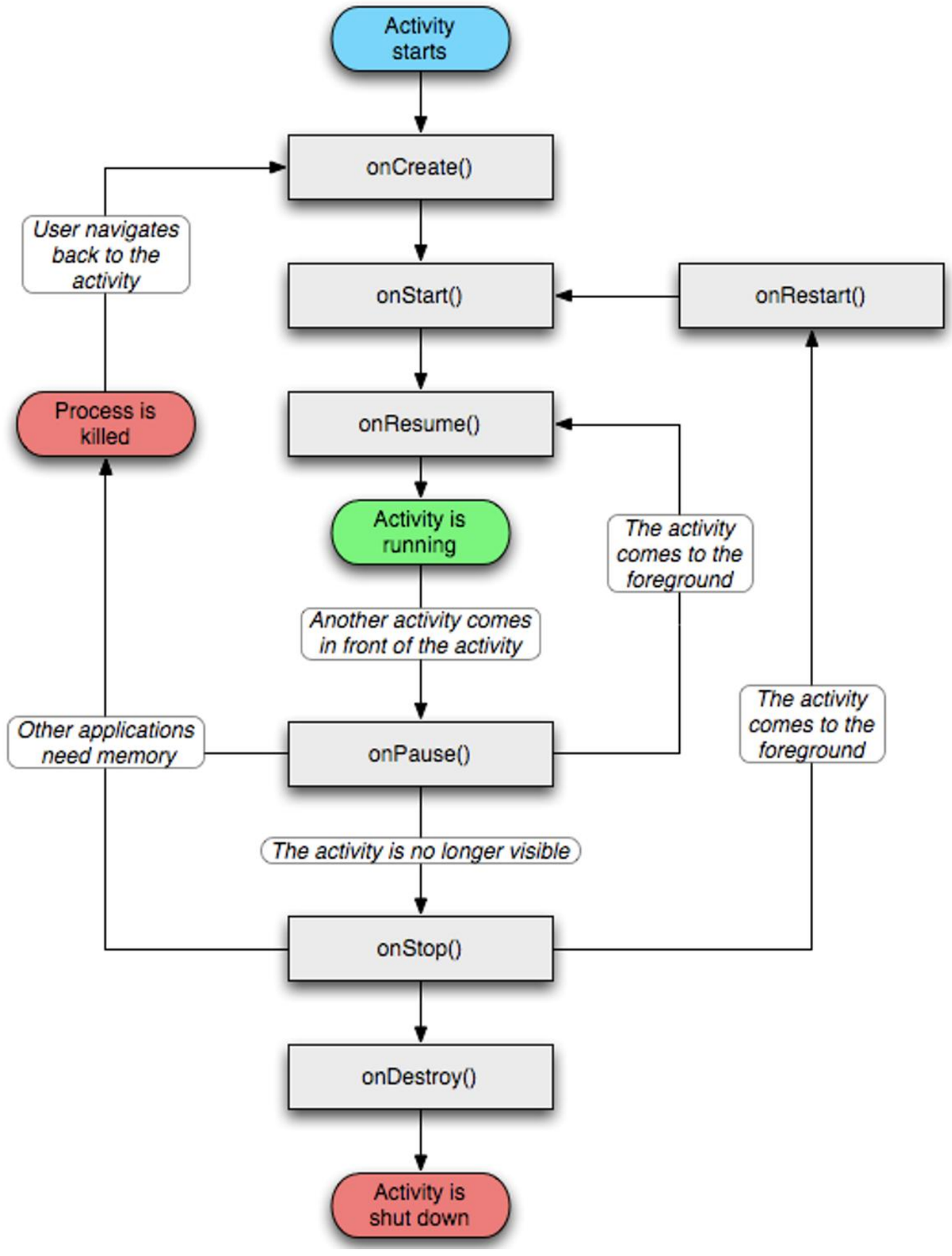


Which Application Building Blocks are mandatory?

- Activity
- Intent
- IntentReceiver
- Service
- ContentProvider

- **Mandatory Vs. Optional**
  - Consider:
    - Traffic Navigation App
    - Alarm App
    - Notepad App

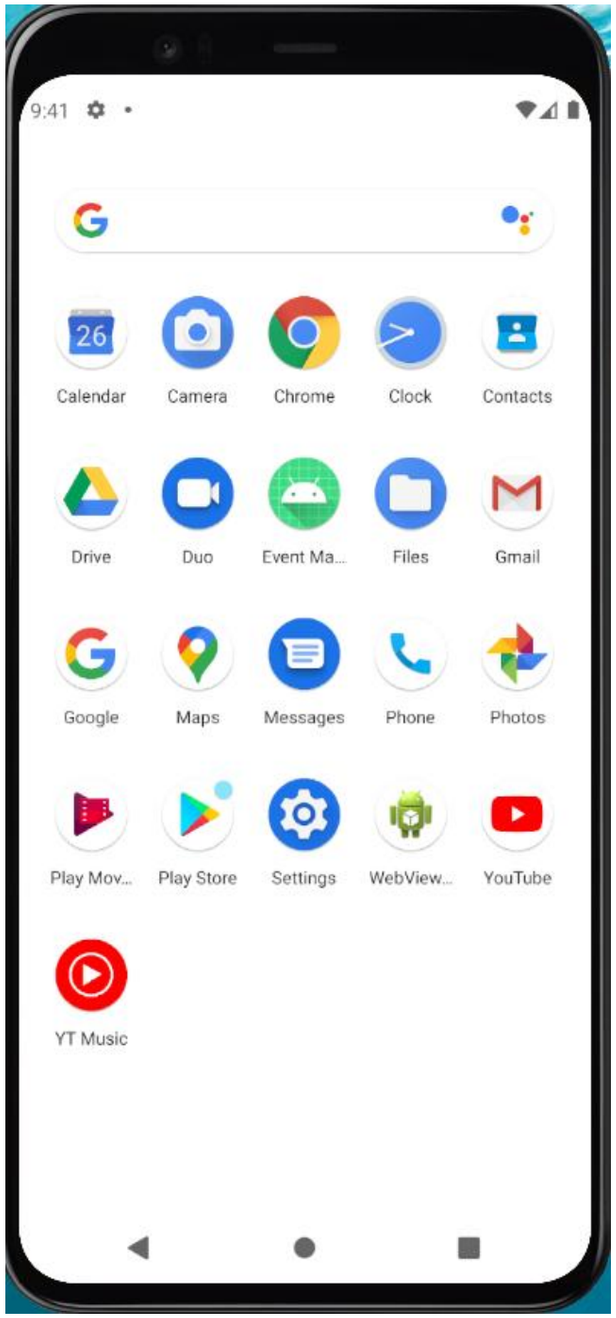
Activity Lifecycle



Development Tools

- Android SDK 2.0 or higher
  developer.android.com
  - Java language
  - Kotlin language
    - Recent version of Android SDK supports conversion of java code to kotlin
- Android Studio

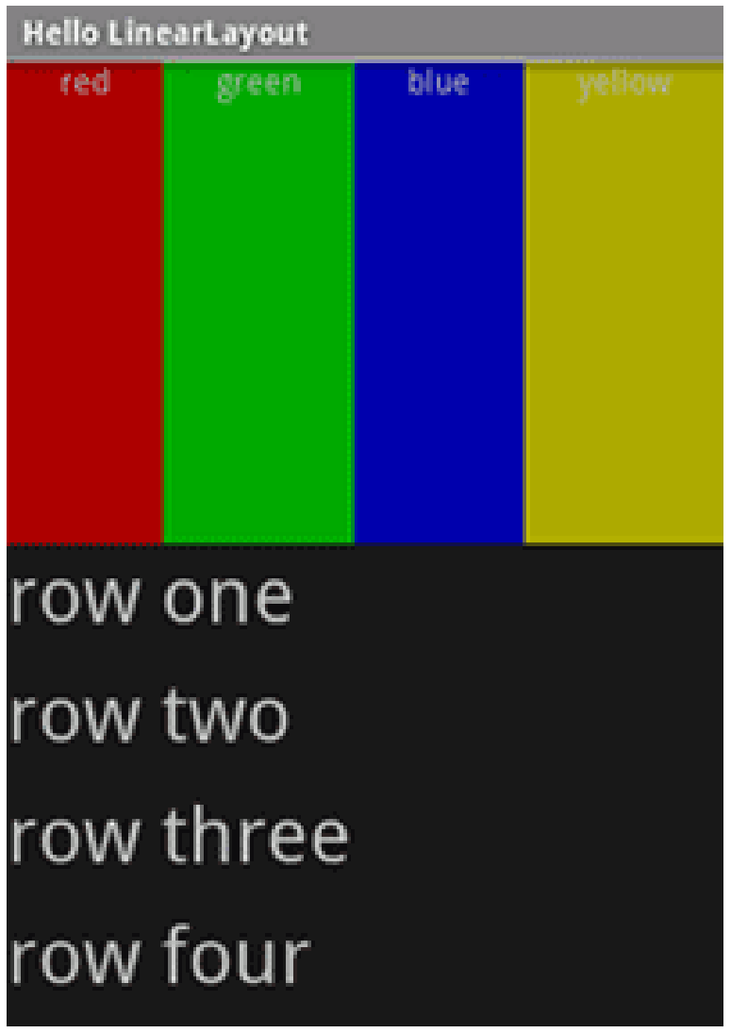
The Emulator

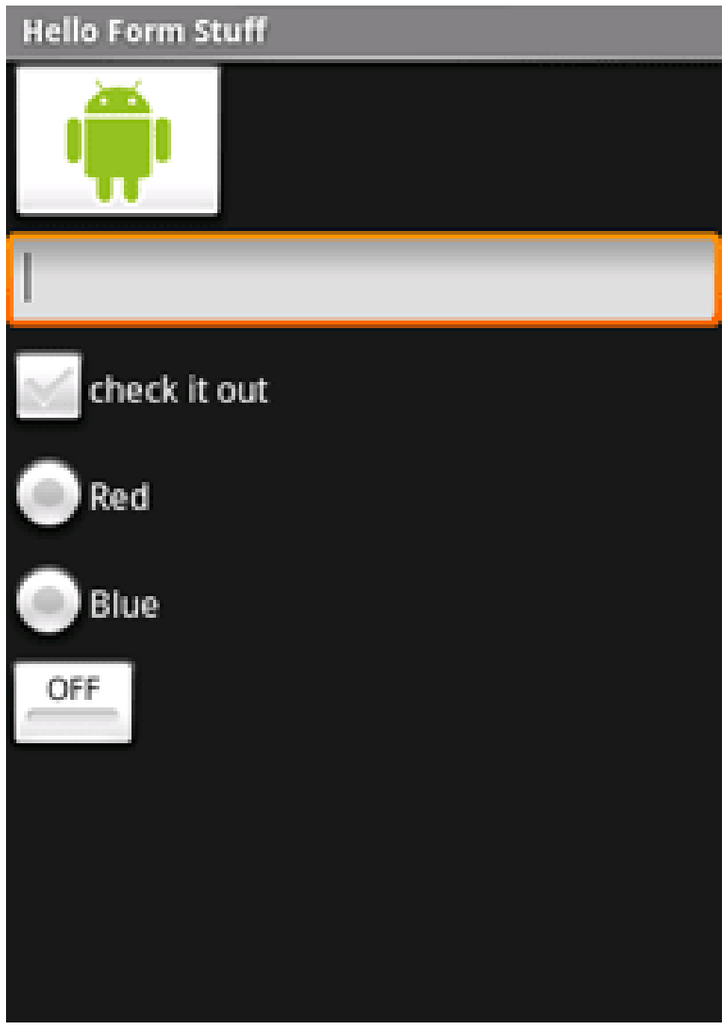


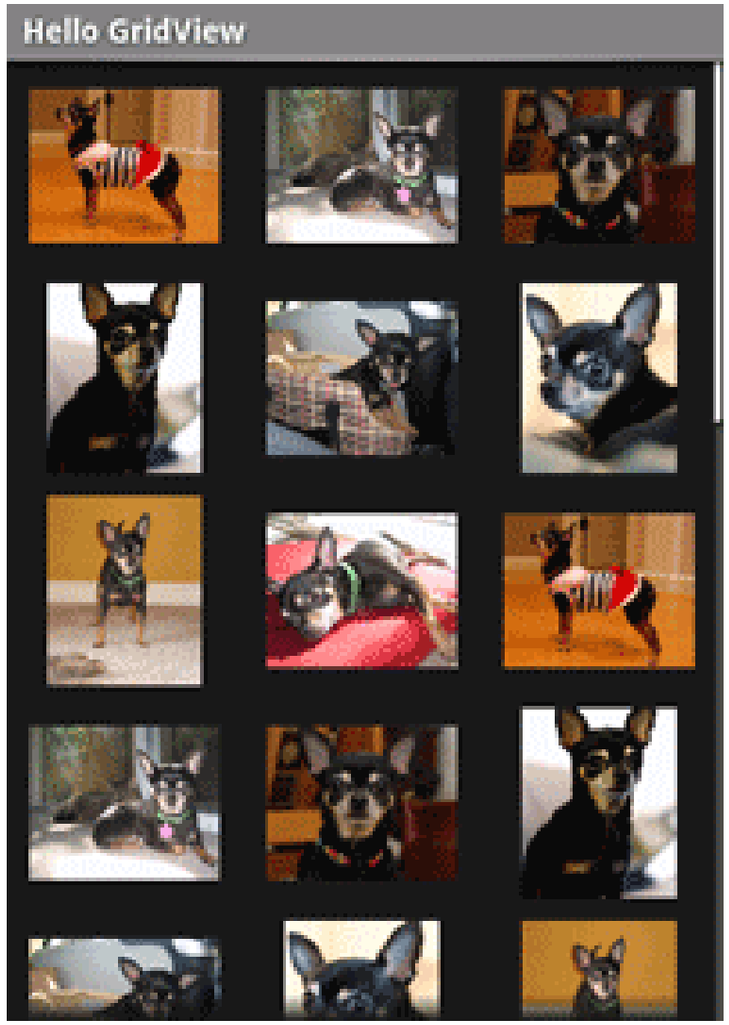
- A virtual mobile device
- Emulates/simulates runs the same image as it is a real device

- Limitations:
  - No Camera support
  - No GPS
  - Etc.

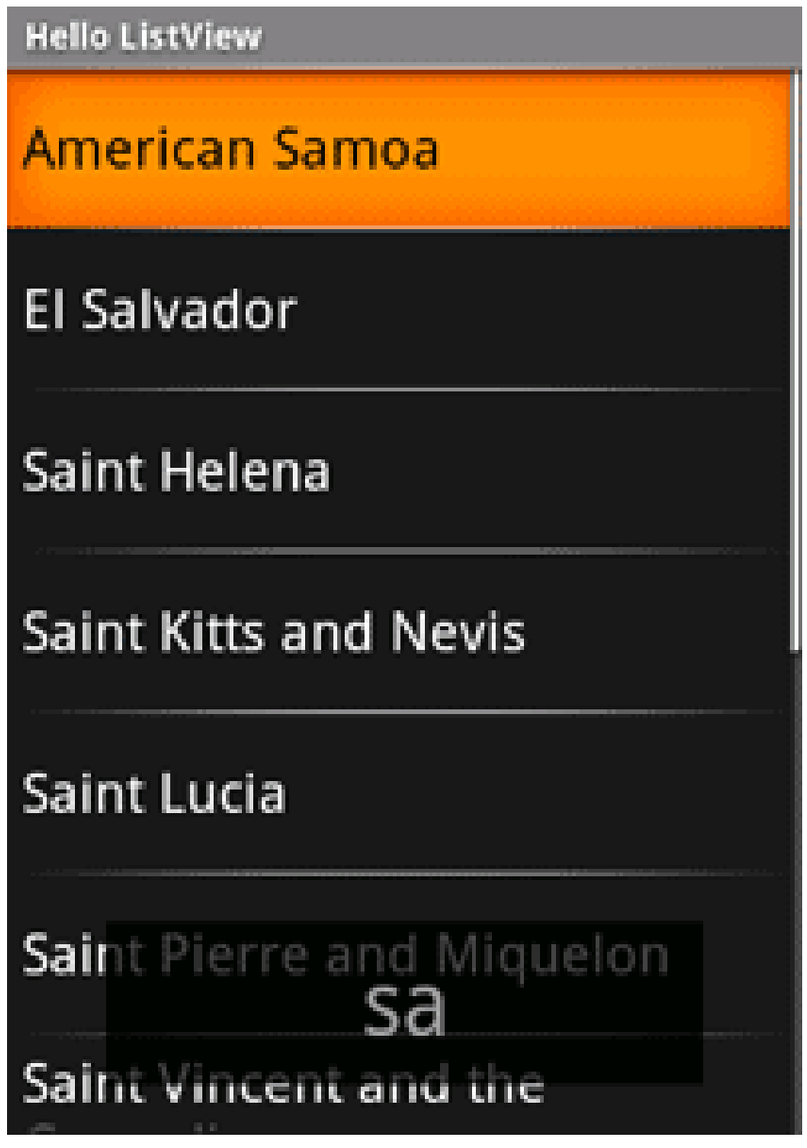
Types of View Groups and Views

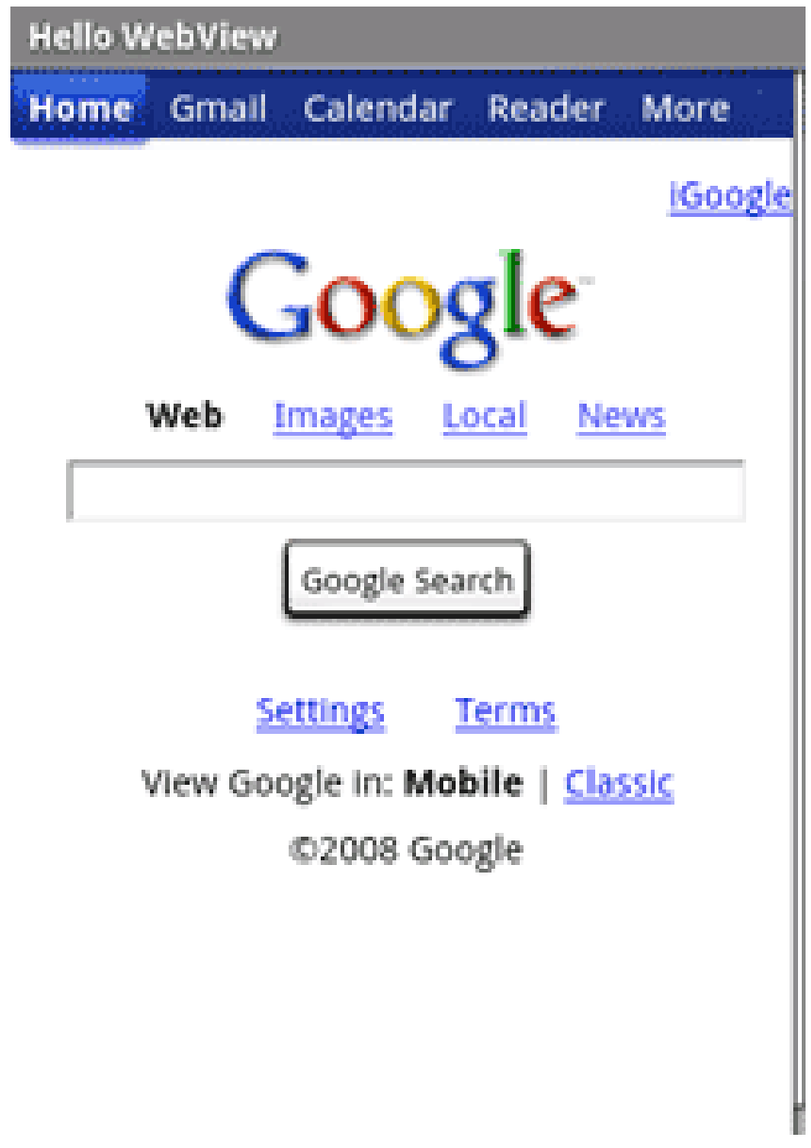






Types of View Groups and Views





Hello World

- A great starting point: http://developer.android.com/guide/tutorials/hello-world.html
- Generating UIs
  - Views
    - E.g. TextView, EditText, Button, ListView, ImageView
  - ViewGroups/Layouts: views are placed here
    - E.g. LinearLayout, RelativeLayout, GridLayout

Thanks!