

BVC ANDROID BASICS CAMP,2017

**By T.T.V.Vamsi Kirshna, N.Naveen and team
Headed by Mrs.Padma Priya mam**



GOOD MORNING

WHO ARE WE?

WHY THIS COURSE?

WHAT ARE YOU GOING TO
LEARN?



Andy Rubin

California, 2003

HISTORY OF ANDROID

Google 2005

2008

API History

<i>Codename</i>	<i>Version</i>	<i>Released</i>	<i>API Level</i>
<i>Honeycomb</i>	3.0 - 3.2.6	Feb 2011	11 - 13
<i>Ice Cream Sandwich</i>	4.0 - 4.0.4	Oct 2011	14 - 15
<i>Jelly Bean</i>	4.1 - 4.3.1	July 2012	16 - 18
<i>KitKat</i>	4.4 - 4.4.4	Oct 2013	19 - 20
<i>Lollipop</i>	5.0 - 5.1.1	Nov 2014	21 - 22
<i>Marshmallow</i>	6.0 - 6.0.1	Oct 2015	23
<i>Nougat</i>	7.0	Sept 2016	24

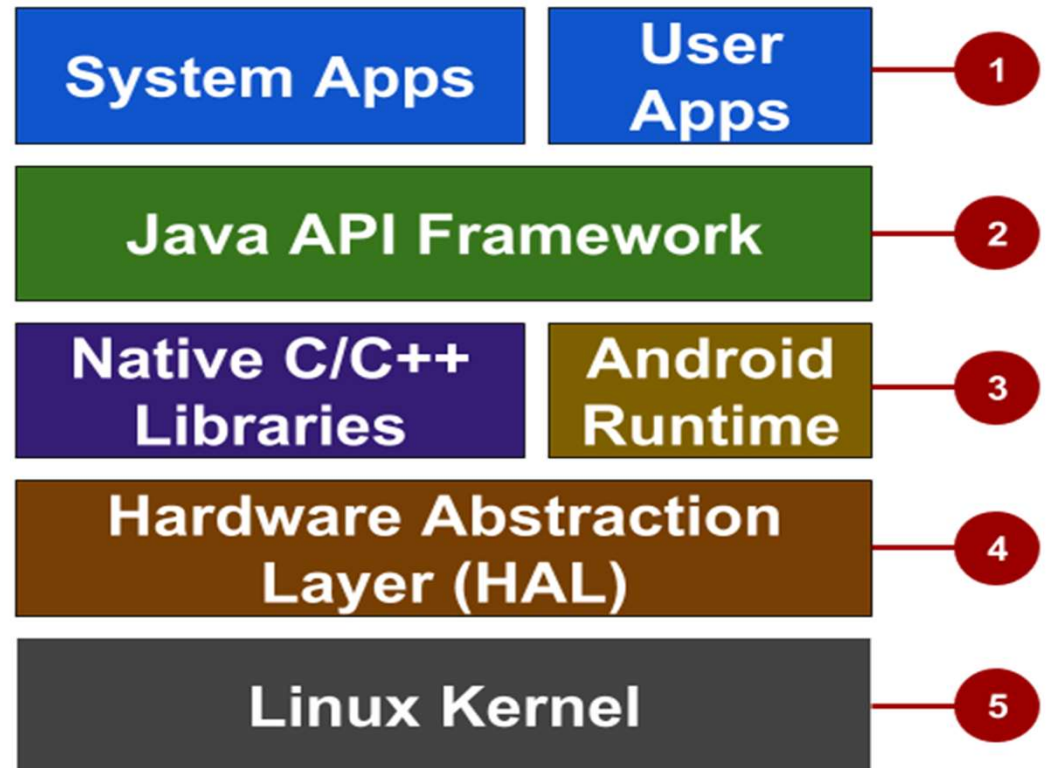
WHY ANDROID?

**WHAT DO YOU NEED TO KNOW TO
DEVELOP ANDROID?**

**WHAT DO YOU NEED TO HAVE TO
DEVELOP ANDROID?**

Android stack

1. System and user apps
2. Android OS API in Java framework
3. Expose native APIs; run apps
4. Expose device hardware capabilities
5. Linux Kernel



System and user apps

- System apps have no special status
- System apps provide key capabilities to app developer



Example:

Your app can use a system app to deliver a SMS message.

Java API Framework

The entire feature-set of the Android OS is available to you through APIs written in the Java language.

- View class hierarchy to create UI screens
- Notification manager
- Activity manager for life cycles and navigation
- Content providers to access data from other apps

Android runtime

Each app runs in its own process with its own instance of the Android Runtime.

C/C++ libraries

- Core C/C++ Libraries give access to core native Android system components and services.

Hardware Abstraction Layer (HAL)

- Standard interfaces that expose device hardware capabilities as libraries

Examples: Camera, bluetooth module

Linux Kernel

- Threading and low-level memory management
- Security features
- Drivers

APP DEVELOPMENT

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#)

What is an Android app?

- One or more interactive screens
- Written using [Java Programming Language](#) and [XML](#)
- Uses the Android Software Development Kit (SDK)
- Uses Android libraries and Android Application Framework
- Executed by Android Runtime Virtual machine (ART)

Challenges of Android development

- Multiple screen sizes and resolutions
- Performance: make your apps responsive and smooth
- Security: keep source code and user data safe
- Compatibility: run well on older platform versions
- Marketing: understand the market and your users
(Hint: It doesn't have to be expensive, but it can be.)

App building blocks

- Resources: layouts, images, strings, colors as XML and media files
- Components: activities, services, ..., and helper classes as Java code
- Manifest: information about app for the runtime
- Build configuration: APK versions in Gradle config files

Component types

- **Activity** is a single screen with a user interface
- **Service** performs long-running tasks in background
- **Content provider** manages shared set of data
- **Broadcast receiver** responds to system-wide announcements

Think of Android as a hotel

- Your app is the guest
- The Android System is the hotel manager
- Services are available when you request them (intents)
 - In the foreground (activities) such as registration
 - In the background (services) such as laundry
- Calls you when a package has arrived (broadcast receiver)
- Access the city's tour companies (content provider)

67 Human Universals

age grading, athletic sports, bodily adornment, [calendar](#), [cleanliness training](#), community organisation, [cooking](#), cooperative labour, cosmology (study of the universe), courtship, dancing, decorative art, divination (predicting the future), [division of labour](#), dream interpretation, [education](#), eschatology (what happens at the end of the world), ethics, ethno-botany (the relationship between humans and plants), etiquette, faith healing, family feasting, fire making, folklore, food taboos, funeral rites, [games](#), gestures, [gift giving](#), government, [greetings](#), [hailing taxis](#),* hairstyles, [hospitality](#), [housing](#), hygiene, incest taboos, inheritance rules, joking, [kin groups](#), kinship nomenclature (the system of categorising relatives), [language](#), [law](#), luck superstitions, magic, marriage, [mealtimes](#), [medicine](#), obstetrics, pregnancy usages (childbirth rituals), penal sanctions (punishment of crimes), personal names, population policy, postnatal care, [property rights](#), propitiation of supernatural beings, puberty customs, religious ritual, residence rules, sexual restrictions, soul concepts, [status differentiation](#), surgery, tool making, [trade](#), [visiting](#), [weather control](#), weaving.

WHAT'S UP FOR TOMORROW?

THE END?

NOPE, THIS IS JUST A BEGINNING



THANK YOU
HAVE A NICE DAY