



# Mobile Application Development I

**Royal University of Phnom Penh**

**Faculty of Engineering**

**Lecturer: Mr. Sao Kimsong**



# About Lecturer

**Name:** Sao Kimsong (Mr.)

**Experiences:** 10 years in Full Stack Software Development.

**Education:** Master in IT Engineering, RUPP (2019)

**Contact:** 017591779 | Telegram: <https://t.me/kimsong>

**Email:** [sao.Kimsong@rupp.edu.kh](mailto:sao.Kimsong@rupp.edu.kh)

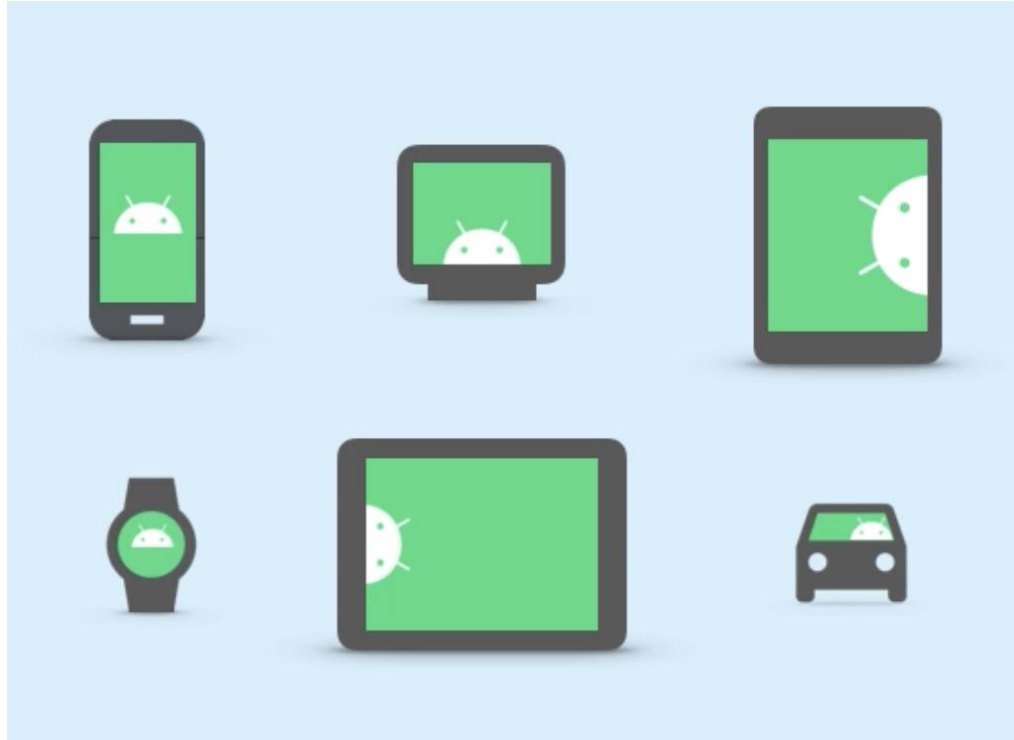
**Current Position:** Senior Manager, Application & Integrations at Amret MFI

# Introduction to Android

# What Is It?

- Open-source mobile platform
- 14 major platform releases so far
- 2.5 billion monthly active Android devices
- 2+ billion monthly active Google Play users
- The world's most popular mobile OS
- Powered by Google

# Available across different form factors



# Versions

Name	Version	API	Release Date
Alpha	1.0	1	2008
Beta	1.1	2	2009
Cupcake	1.5	3	2009
Donut	1.6	4	2009
Eclair	2.0 - 2.1	5 - 6	2009
Froyo	2.2 - 2.2.3	8	2010
Gingerbread	2.3 – 2.3.7	9 - 10	2010

# Versions

Name	Version	API	Release Date
Honeycomb	3.0 – 3.2.6	11 - 13	2011
Ice Cream Sandwich	4.0 – 4.0.4	14 - 15	2011
Jelly Bean	4.1 – 4.3.1	16 - 18	2012
KitKat	4.4 – 4.4.4	19	2013
Lollipop	5.0 – 5.x	21 - 22	2014
Marshmallow	6.0 – 6.x	23	2015
Nougat	7.0 – 7.x	24 - 25	2016

# Versions

Name	Version	API	Release Date
Oreo	8.0 – 8.x	26 - 27	2017
Pie	9.0 – 9.x	28	2018
Quince Tart	10.0 – 10.x	29	2019
Red Velvet Cake	11.0 – 11.x	30	2020
Snow Cone	12.0 – 12.x	31-32	2021
Tiramisu	13.0 – 13.x	33	2022
<b>UpsideDownCake</b>	<b>14.0</b>	<b>34</b>	<b>2023</b>



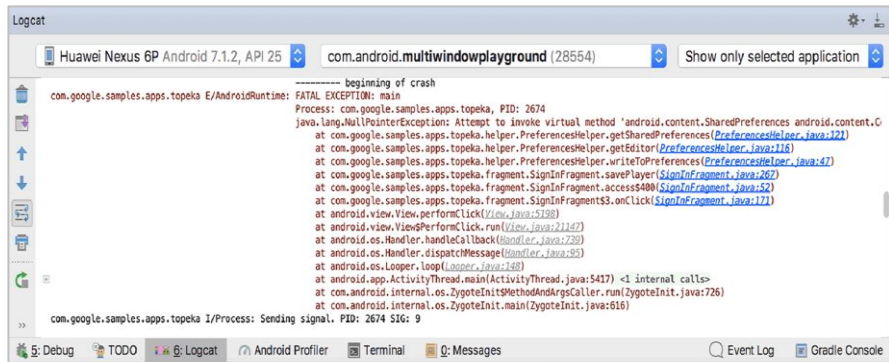
# Android Studio

- The official IDE for developing Android app
- Replaced Eclipse
- 1st stable build released in 2014
- Free
- Android Studio Hedgehog 2023.1.1 (Latest Version)



# Android Logcat

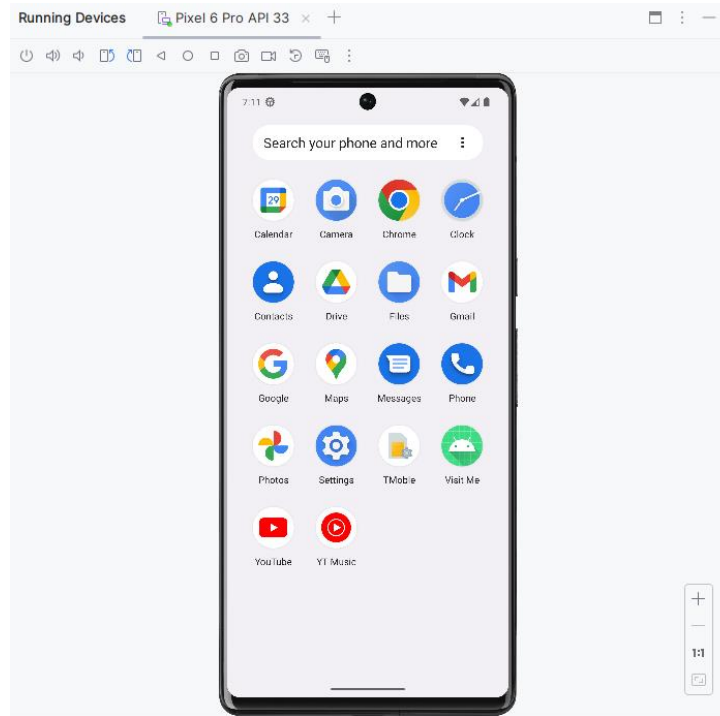
- A tool which dumps a log of system messages, including stack traces when the device throws an error and messages that you have written from your app with the Log class.



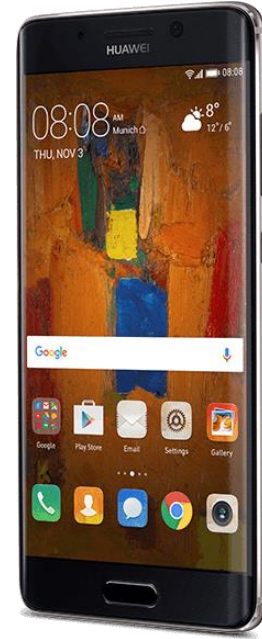
# Testing Environment

- Emulator
- Real device
- Remote device (cloud device).

# Android Emulator



# Android Real Devices



# Remote devices

- Samsung remote test lab
- Firebase test lab.



# Android Application Packages



**.apk**

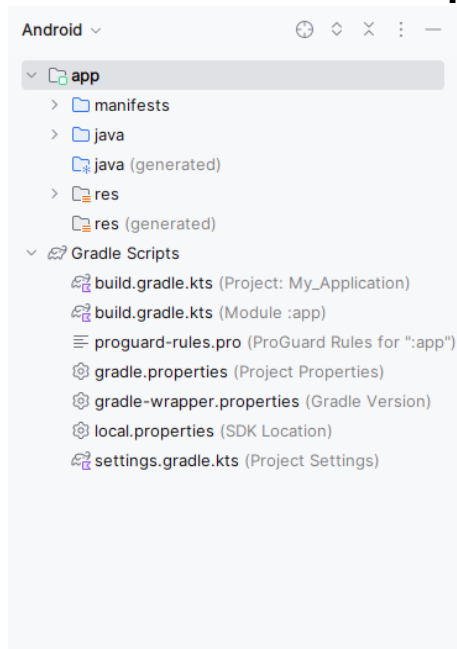
Android Package



Android App Bundle

# Project Structure

- An Android project is divided into two main parts:
  - Module
  - Gradle





# Project Structure

- The modules can be app module or library module
- Gradle are configuration files used to build the modules.
- Each module has its own Gradle script.
- The project itself has also its own Gradle scripts.

# Application Components

- App components are the essential building blocks of an Android app
- Each component is an entry point through which the system or a user can enter your app
- Some components depend on others.

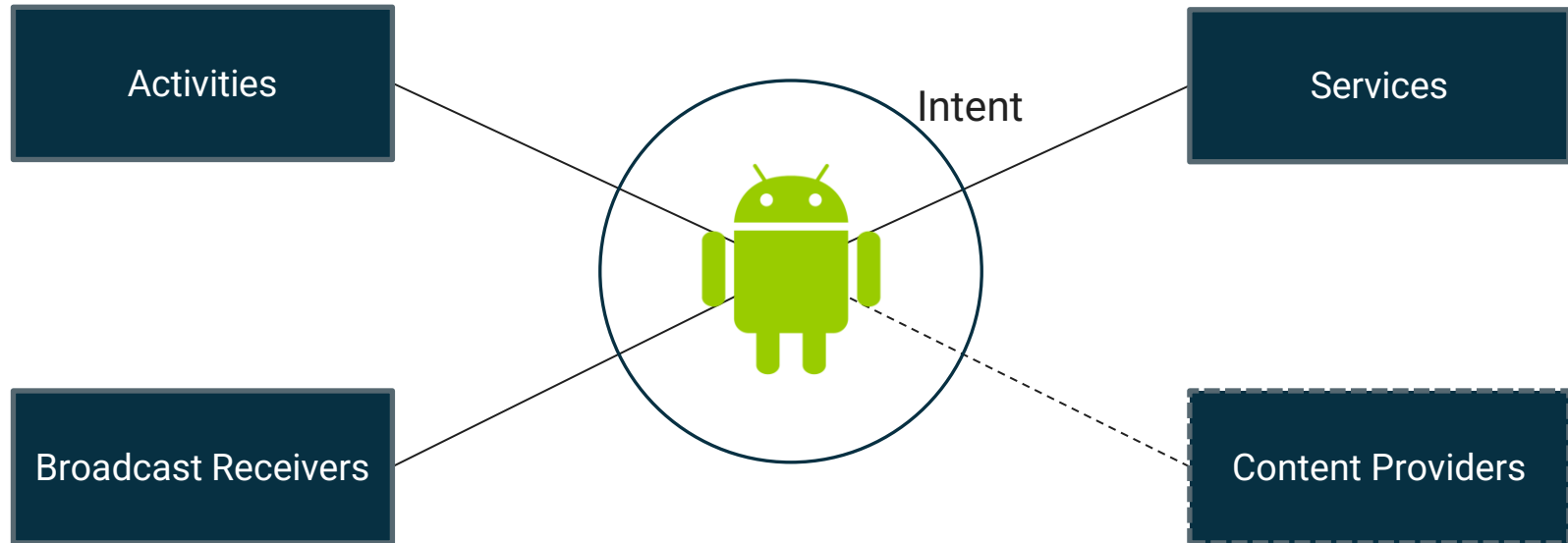
# Application Components

- All components must be registered in the Manifest file
- There are four different types of app components:
  - Activities
  - Services
  - Broadcast receivers
  - Content providers.

# Application Components

Component	Description	Example
Activities	The entry point for interacting with the user.	Login Activity
Services	Long running operation in the background without providing user interface.	Download Service
Broadcast Receivers	Special system/app event listeners.	SMS Receiver
Content Providers	Standard interface that connects data in one process with code running in another process.	Contacts Provider

# Application Components



# What Is Intent?

- Are a message-passing mechanism
  - Within your application
  - Between applications (Camera, Gallery, ...)
  - Between the **system** and your **application** (Sound, dial action, ..)

# Intent

- The primary pieces of information in an intent are **action** and **data**. And some more as secondary
  - For example, your app wants to tell the system to dial a number of 017 591 779. So your app needs to pass an intent to the system in the following information:
  - Action: **ACTION\_DIAL**
  - Data: **tel:017591779**

# Intent Types

- Two types of intent:
  - **Explicit intent:** component specified. Usually, it's used within your app.
  - **Implicit intent:** component unspecified. Usually, it's used between system and applications.



# Context

- This is an abstract class whose implementation is provided by the Android system.
- It allows access to application-specific resources and classes, as well as up-calls for application-level operations such as launching activities, broadcasting and receiving intents, etc.

# Context – Cont.

- There are mainly two types of Context that are available in Android.
  - Application Context
  - Activity Context

# Layout

- An XML resource which defines the visual structure for UI.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fitsSystemWindows="true">

        <com.google.android.material.appbar.MaterialToolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize" />

    </com.google.android.material.appbar.AppBarLayout>

    <include layout="@layout/content_main" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```