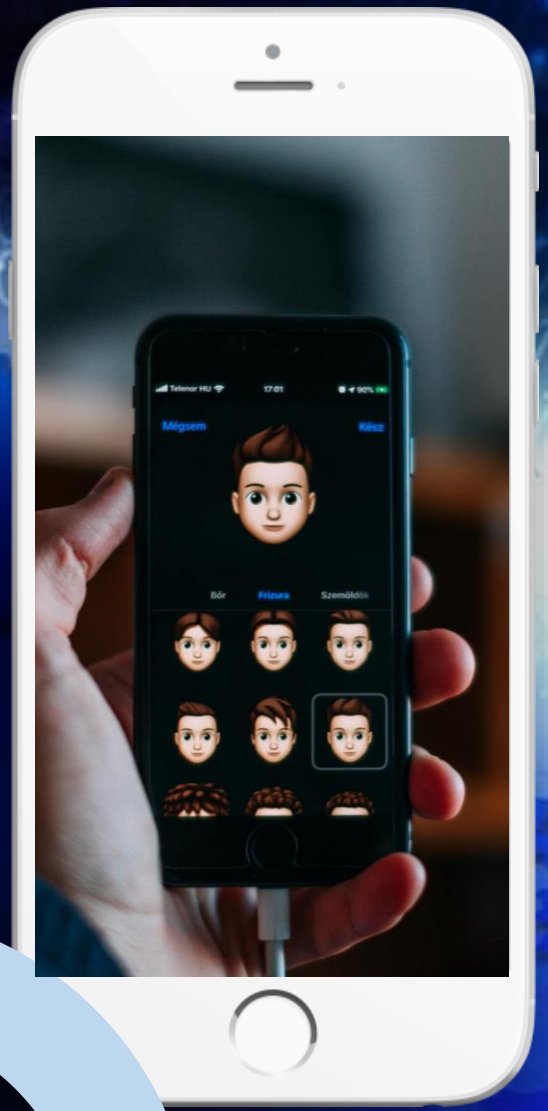


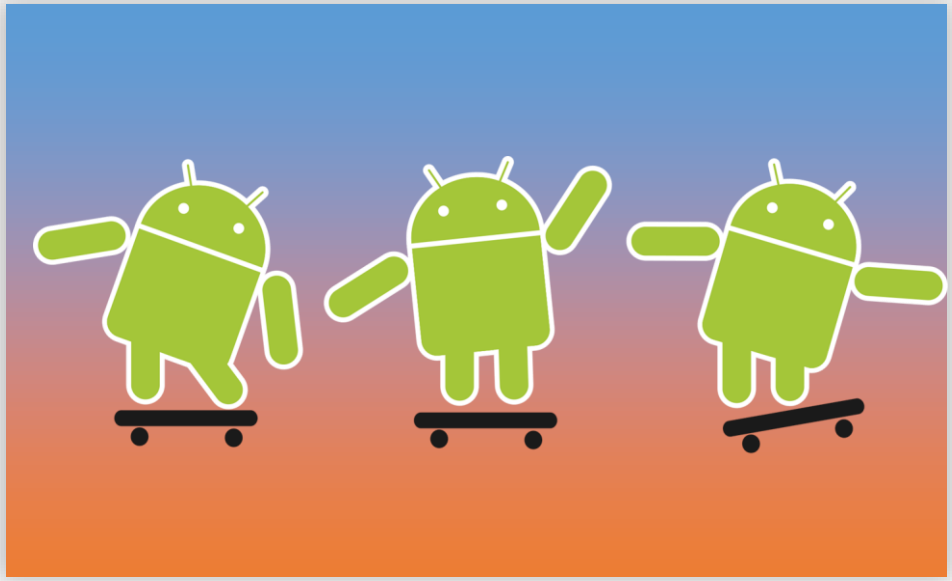
Introduction to Android – Mobile Apps

Presented By Putri Hayati, S.ST, M.Kom

Read More

UNIVERSITAS BUDI LUHUR

Contents



Android Ecosystem

What is Android?

Mobile operating system based on Linux kernel

Powers devices such as watches, TVs, and cars

Over 2 Million Android apps in Google Play store

User Interface for touch screens

Used on over 80% of all smartphones

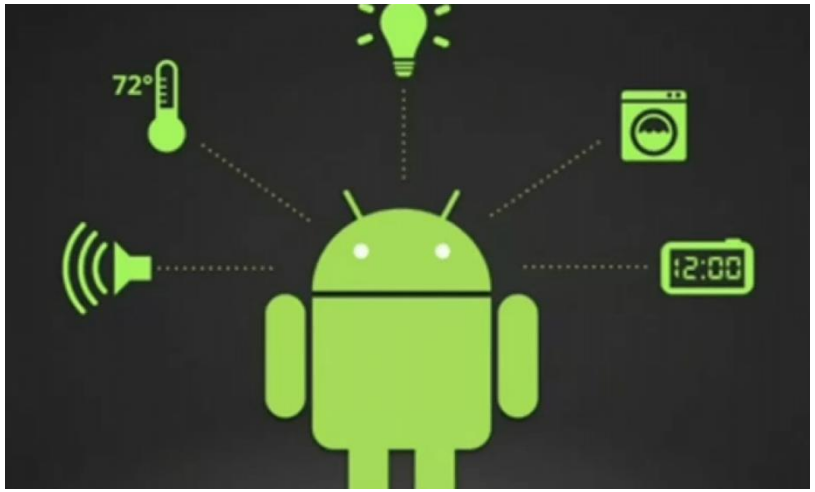
Highly customizable for devices

Open source

Android and sensors

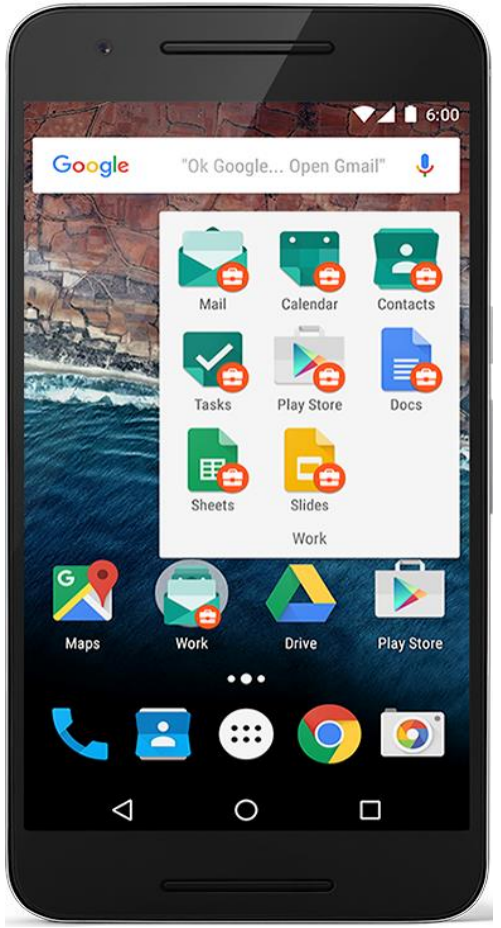
Sensors can discover user action and respond :

- Device contents rotate as needed
- Walking adjusts position on map
- Tilting steers a virtual car or controls a physical toy
- Moving too fast disables game interactions

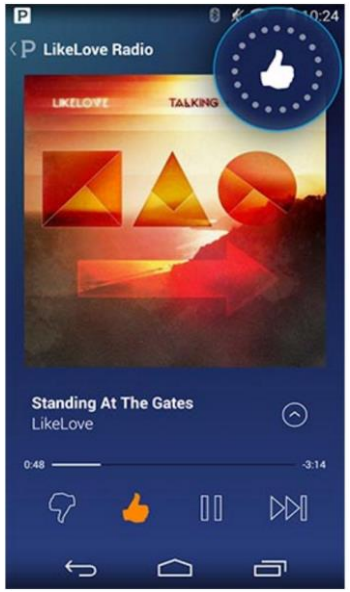


Android home screen

- Launcher icons for apps
- Self-updating widgets for live content
- Can be multiple pages
- Folders to organize apps
- "OK Google"

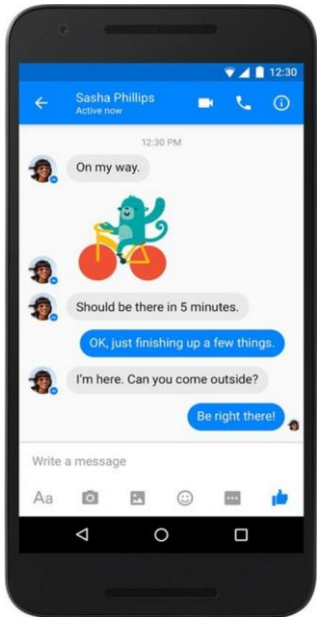


Android app examples



Pandora

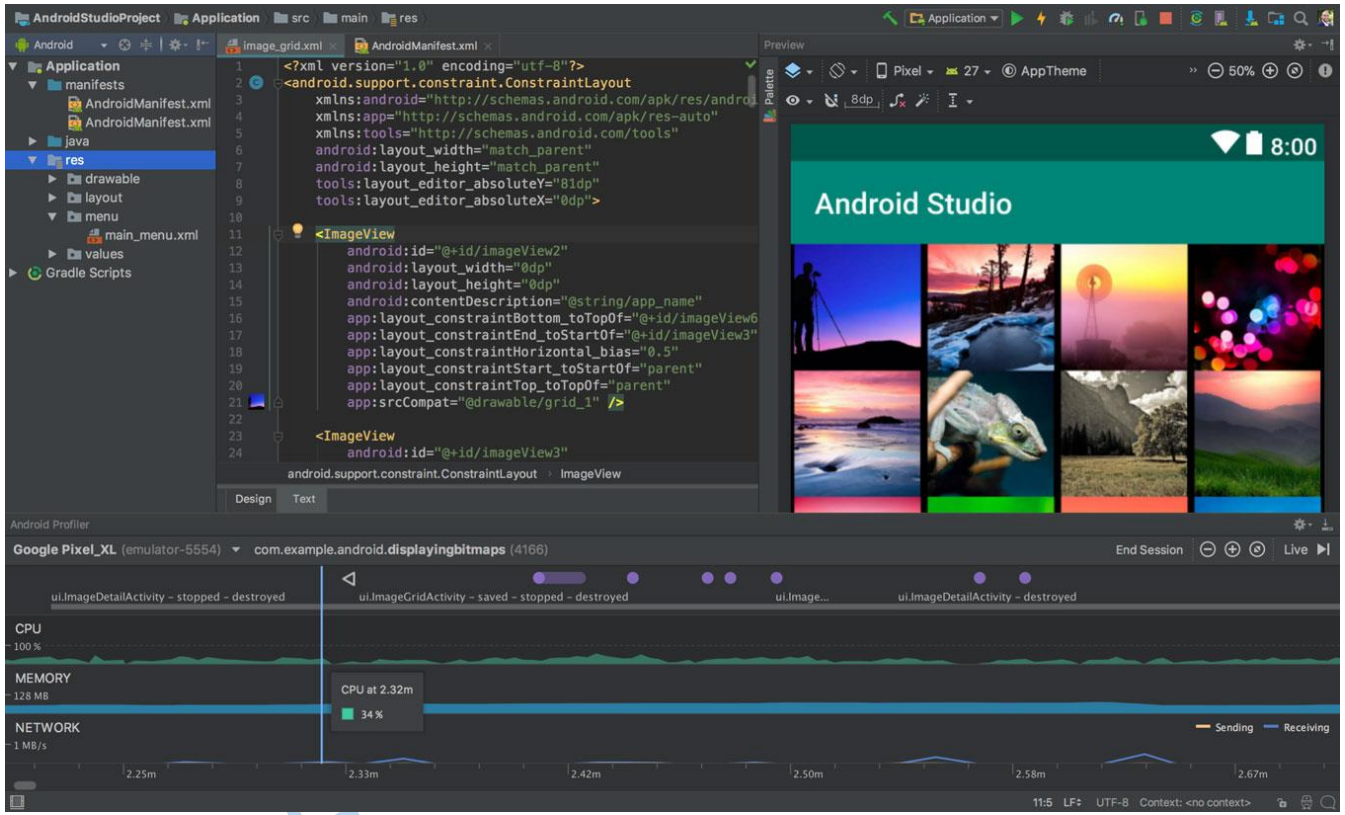
Pokemon GO



Facebook Messenger

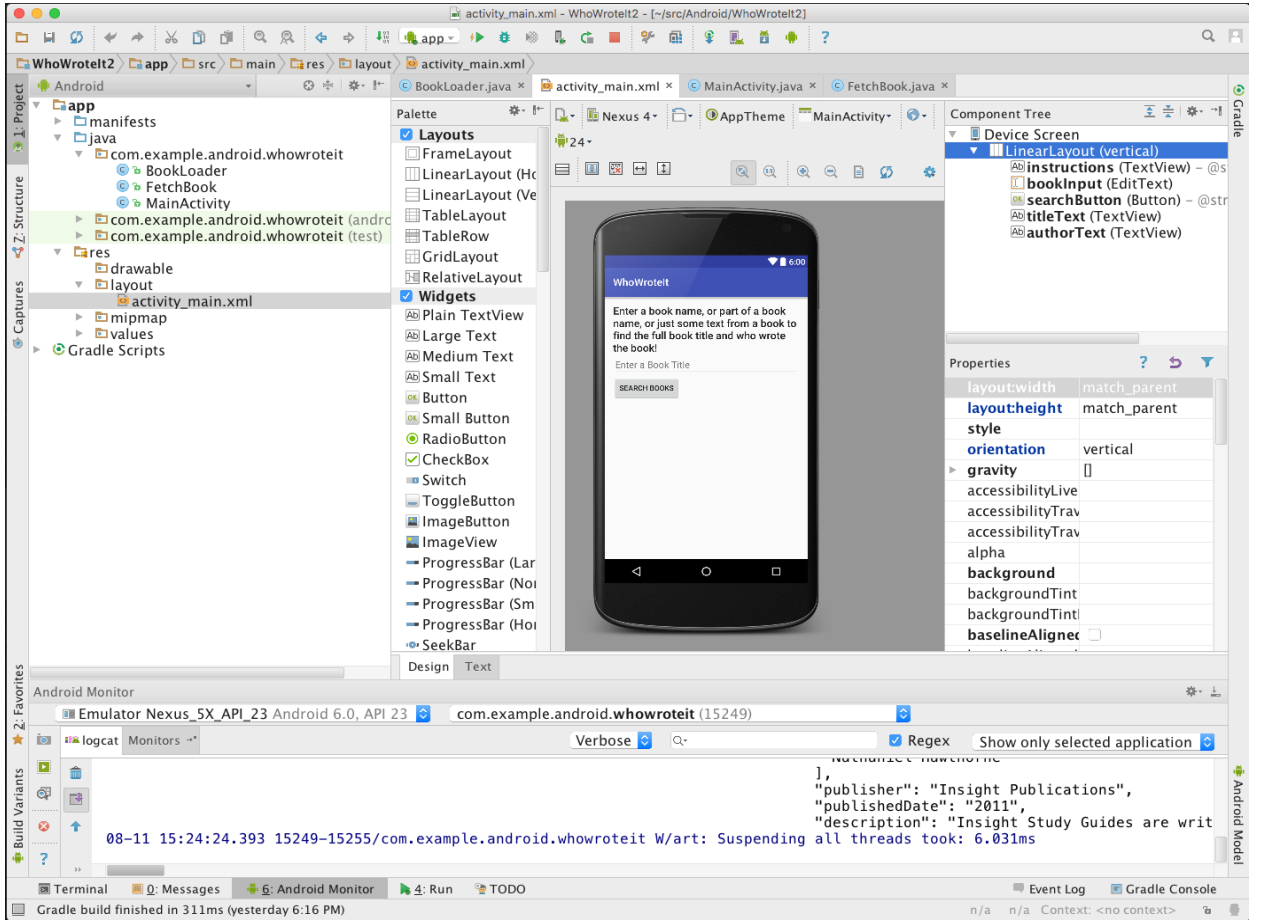
Android Software Developer Kit (SDK)

- Development tools (debugger, monitors, editors)
- Libraries (maps, wearables)
- Virtual devices (emulators)
- Documentation (developers.android.com)
- Sample code



Android Studio

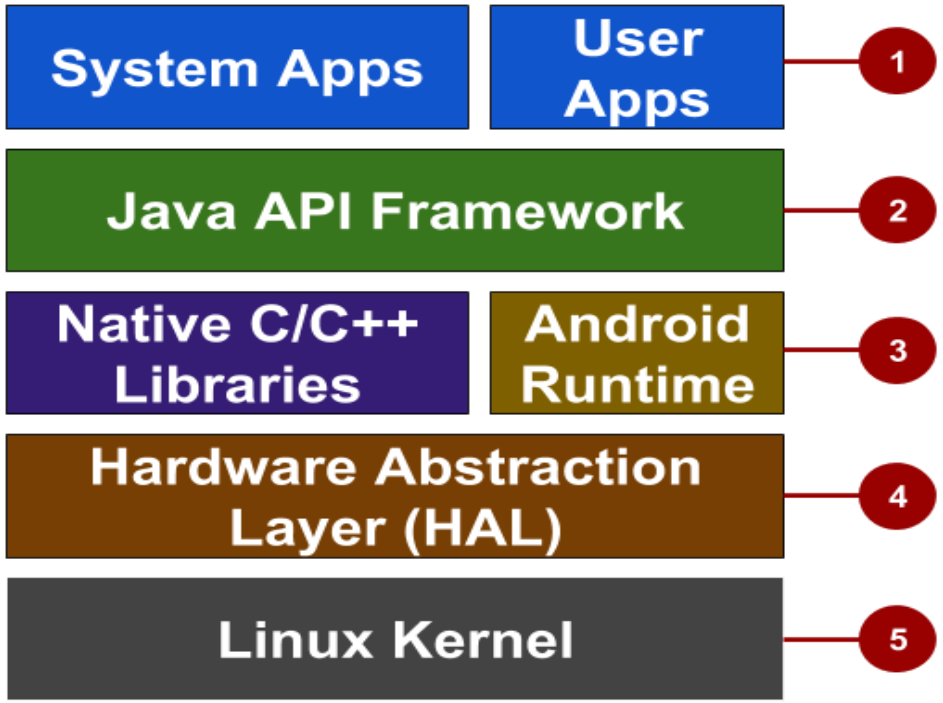
- Official Android IDE
- Develop, run, debug, test, and package apps
- Monitors and performance tools
- Virtual devices
- Project views
- Visual layout editor



Android Platform Architecture

Android stack

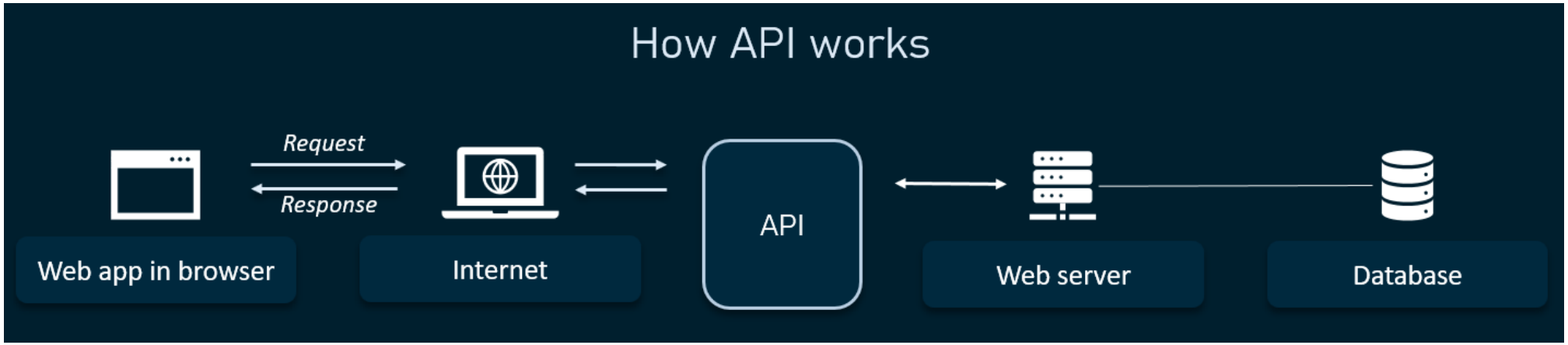
1. System and user apps
2. Android OS API in Java framework
3. Expose native APIs; run apps
4. Expose device hardware capabilities
5. Linux Kernel



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 versions

Cupcake
1.5

Donut
1.6

Eclair
2.0/2.1

Froyo
2.2

Gingerbread
2.3

Honeycomb
3.0/3.1

Ice Cream Sandwich
4.0

Jelly Bean
4.1/4.2/4.3

KitKat
4.4

Lollipop
5.0

Marshmallow
6.0

Nougat
7.0

Oreo
8.0

Pie
9.0

android
10

android

App Development

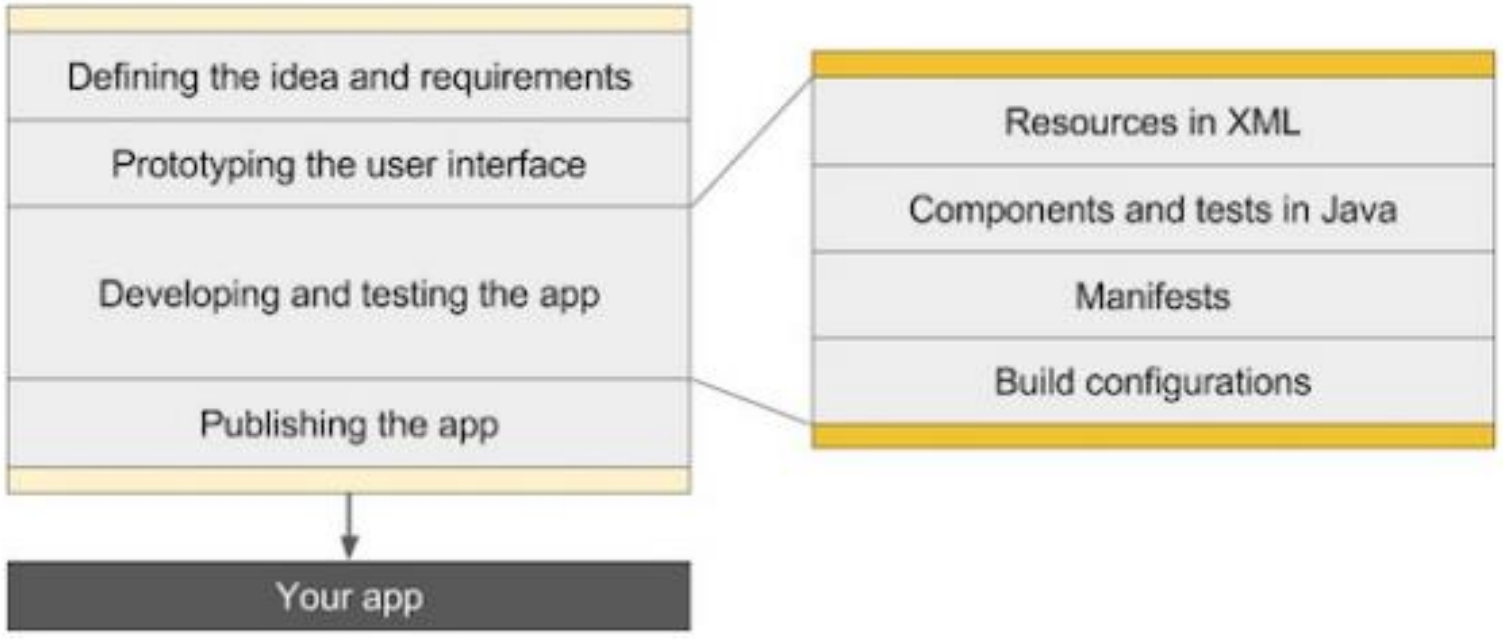
- One or more interactive screens
- Written using Java Programming Language and XML
- Uses the Android Software Development Kit (SDK)
- Uses Android libraries and Android Application Framework
- Executed by Android Runtime Virtual machine (ART)

Create your First App

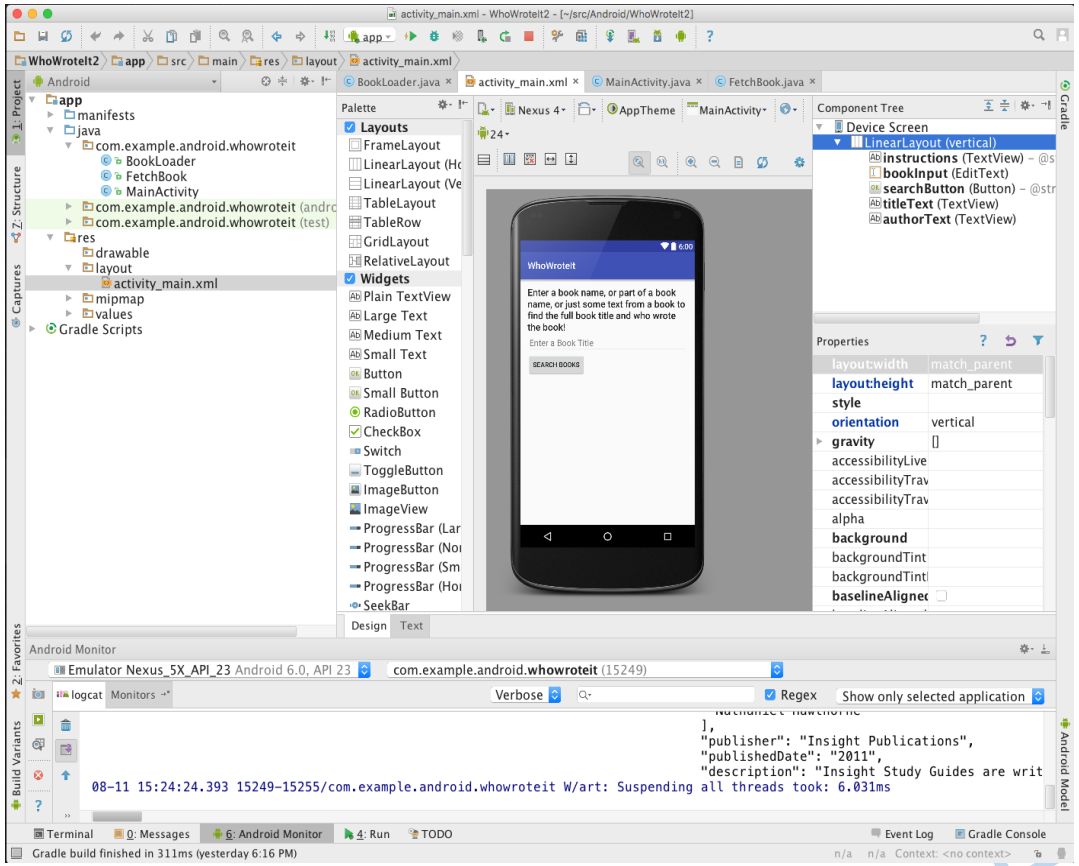
Prerequisites

- Java Programming Language
- Object-oriented programming
- XML - properties / attributes
- Using an IDE for development and debugging

Development Process

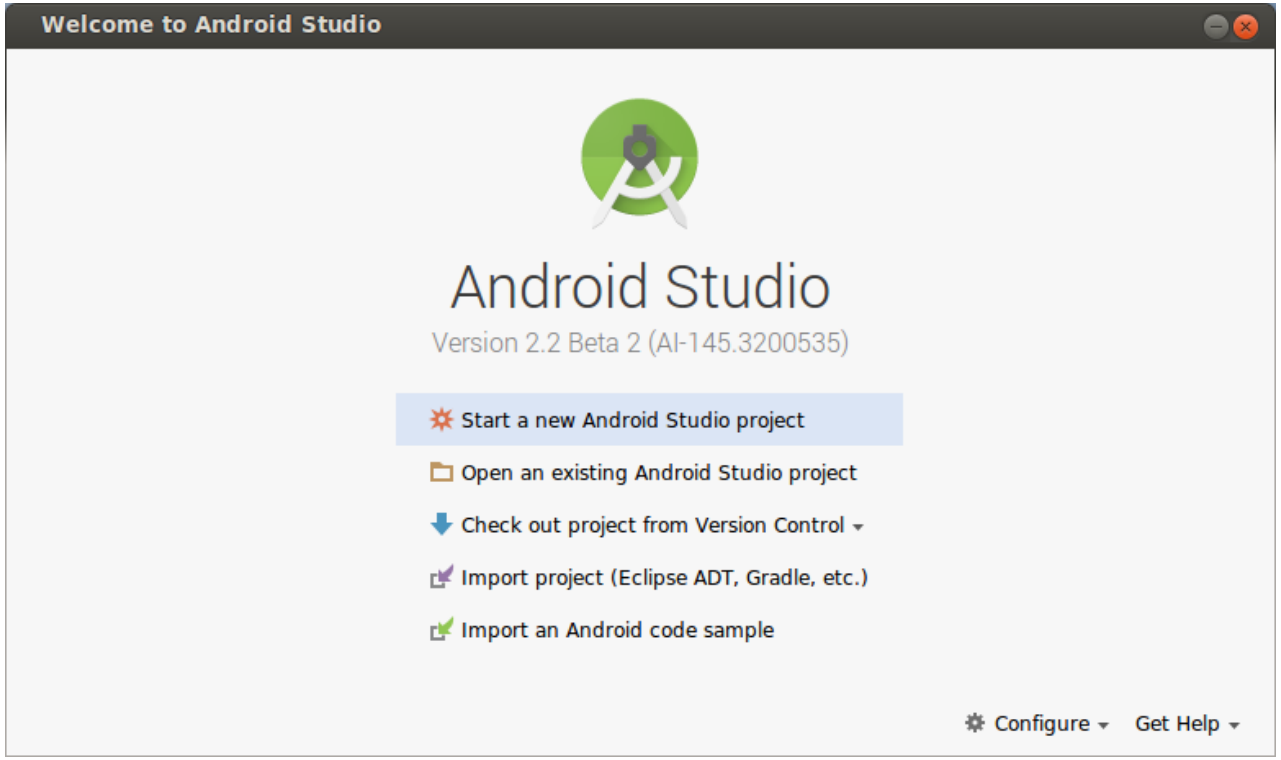


What is Android Studio ?

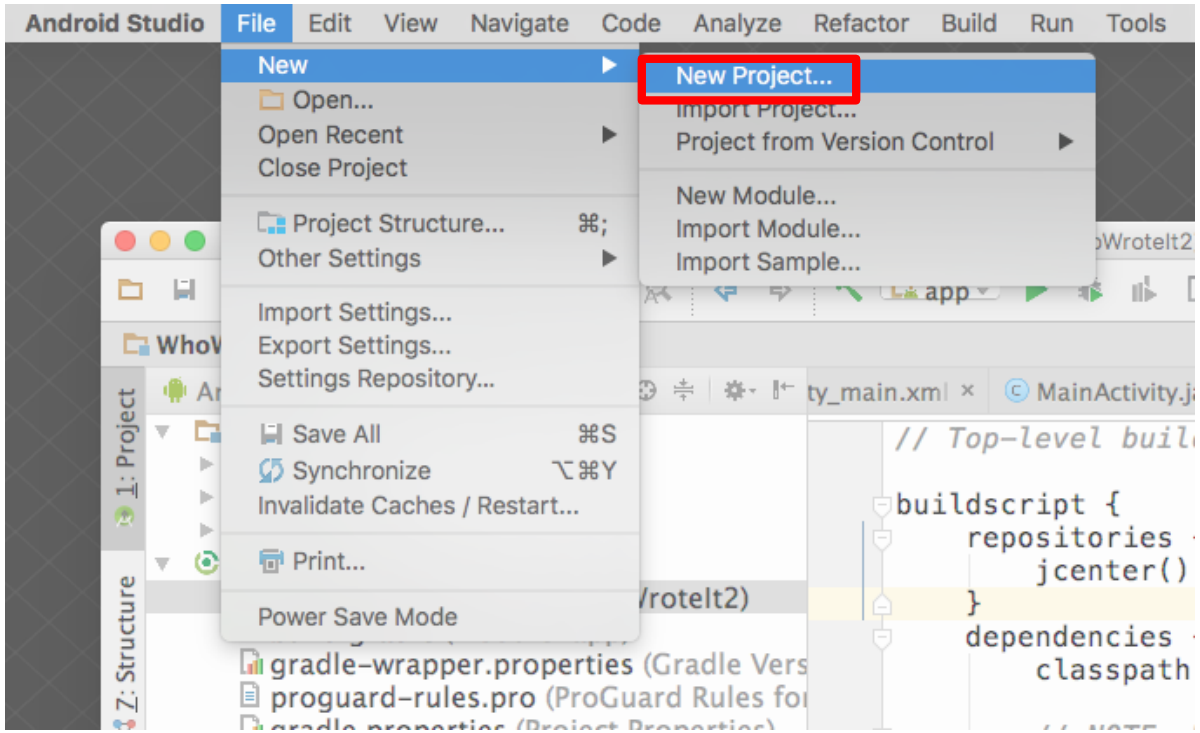


- Android IDE
- Project structure
- Templates
- Layout Editor
- Testing tools
- Gradle-based build
- Log Console
- Debugger
- Monitors
- Emulators

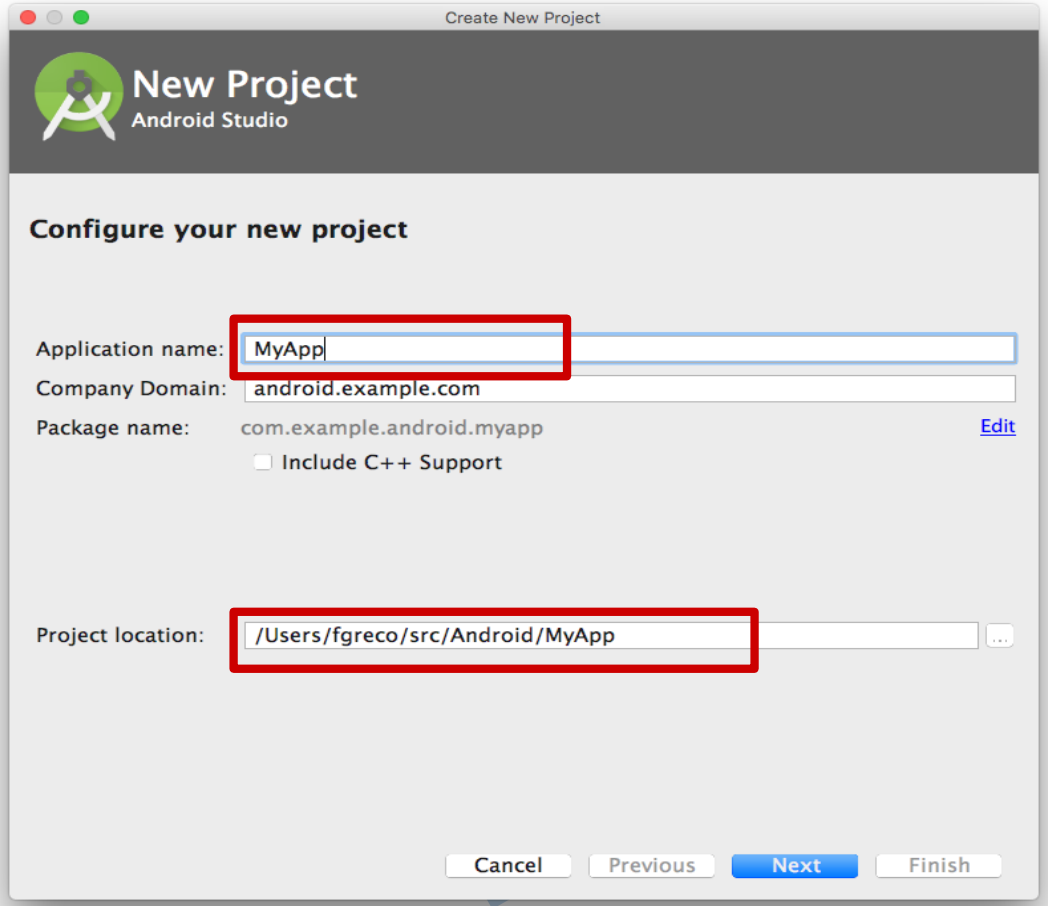
Start Android Studio



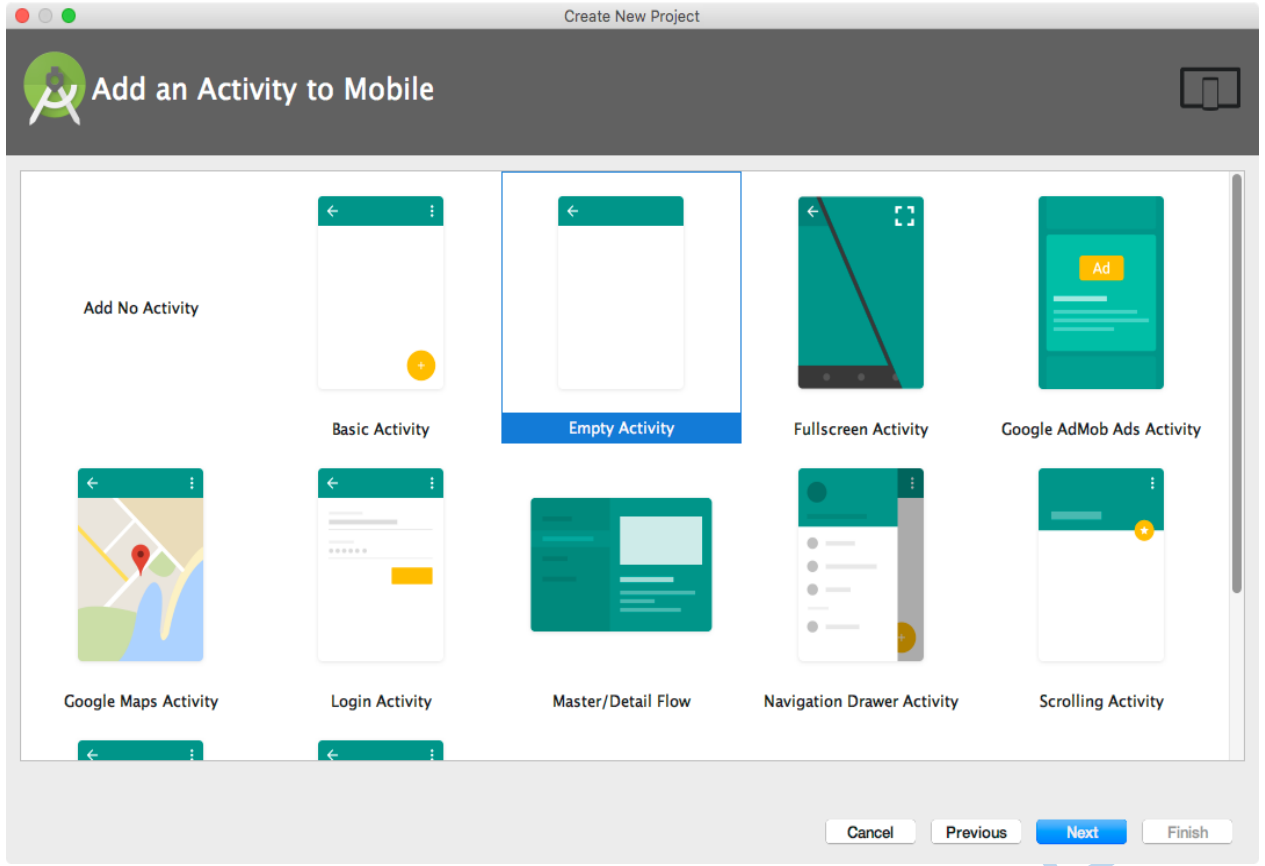
Create a project inside Android Studio



Name your app



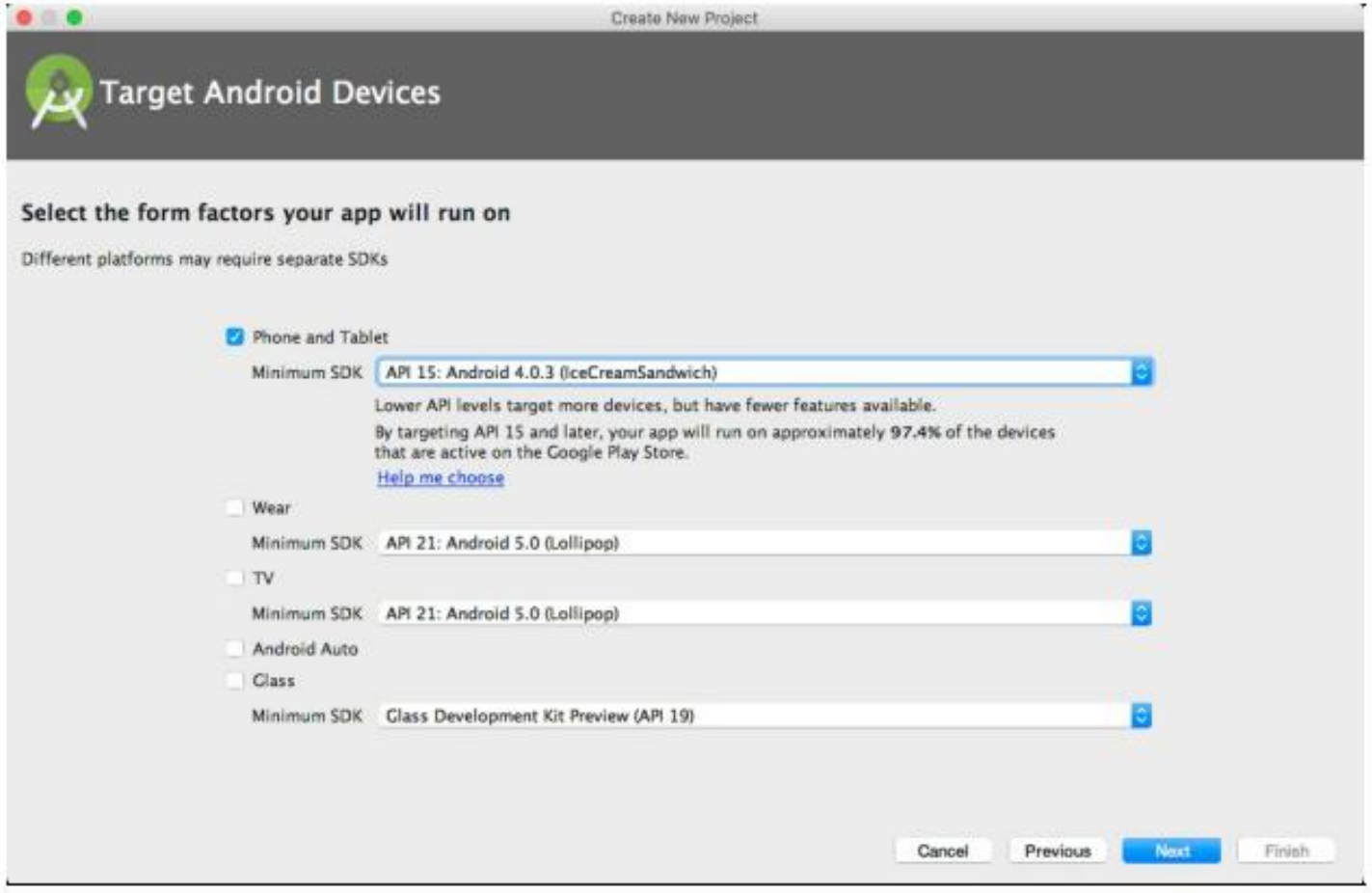
Pick activity template



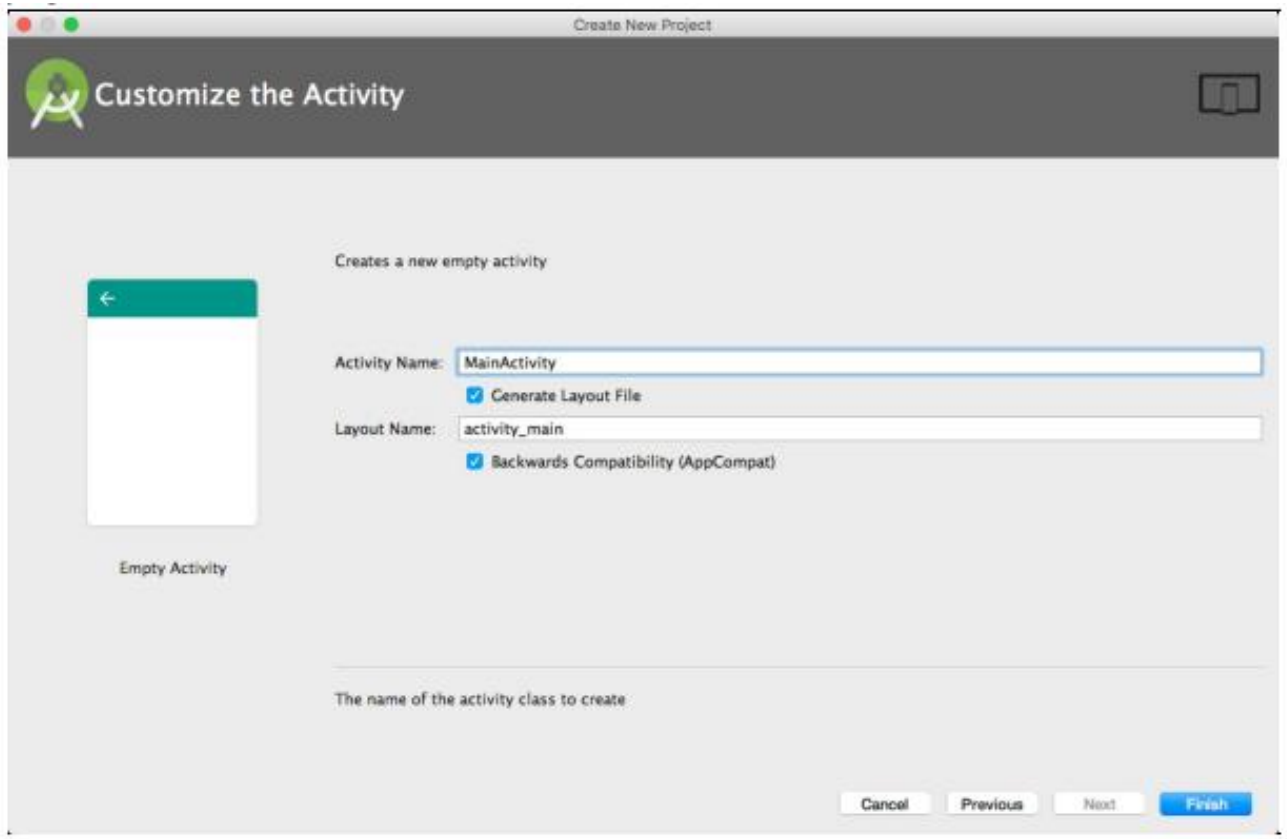
Choose templates for common activities, such as maps or navigation drawers.

Pick Empty Activity or Basic Activity for simple and custom activities.

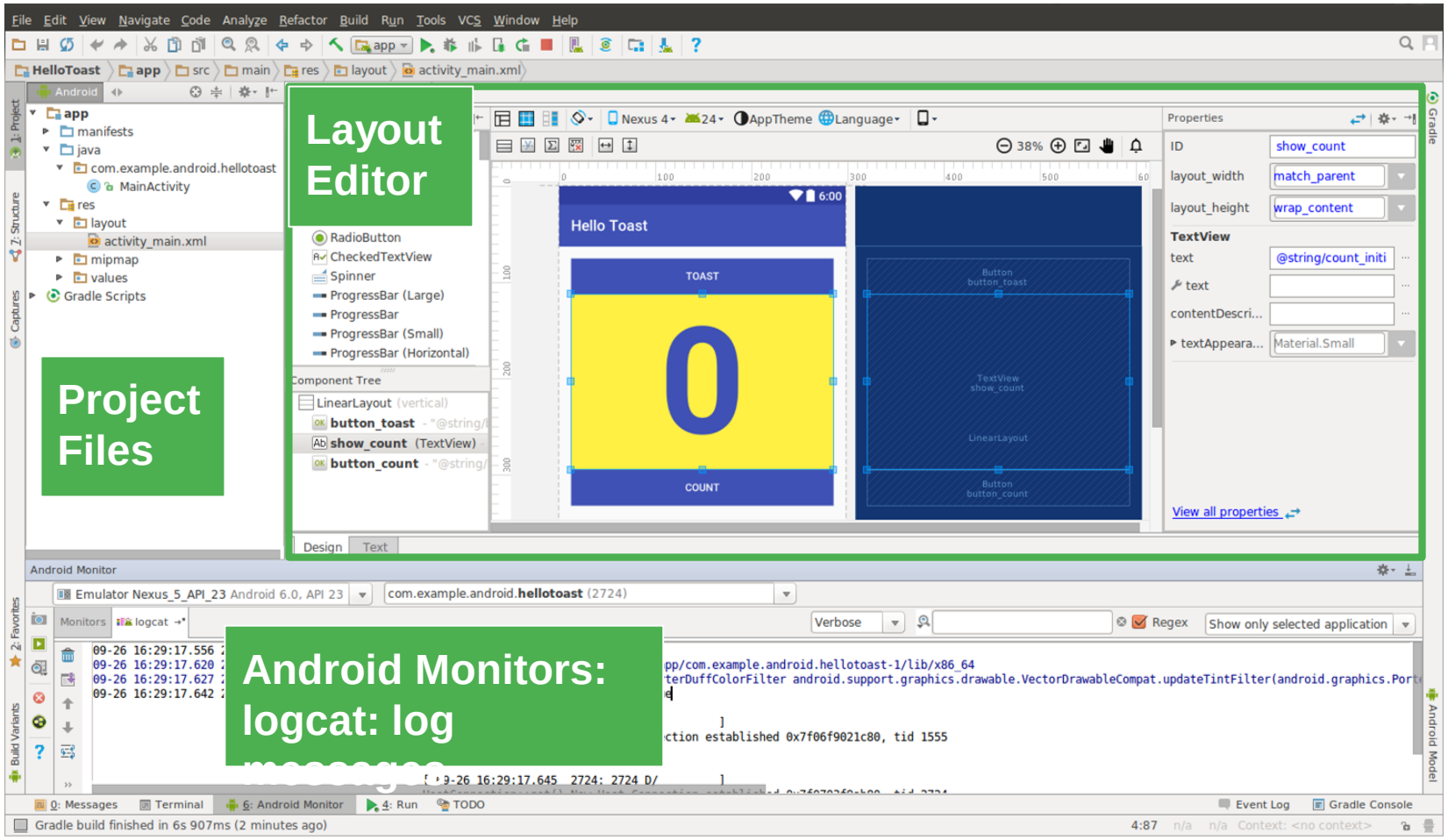
Android Devices



Customize the Activity

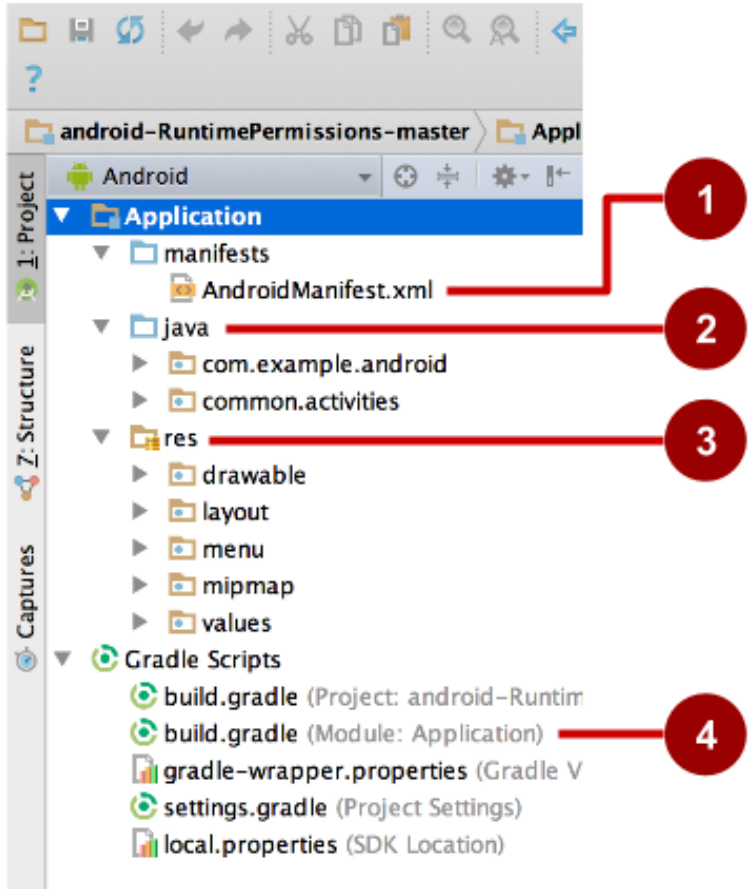


Android Studio Panes



Project folders

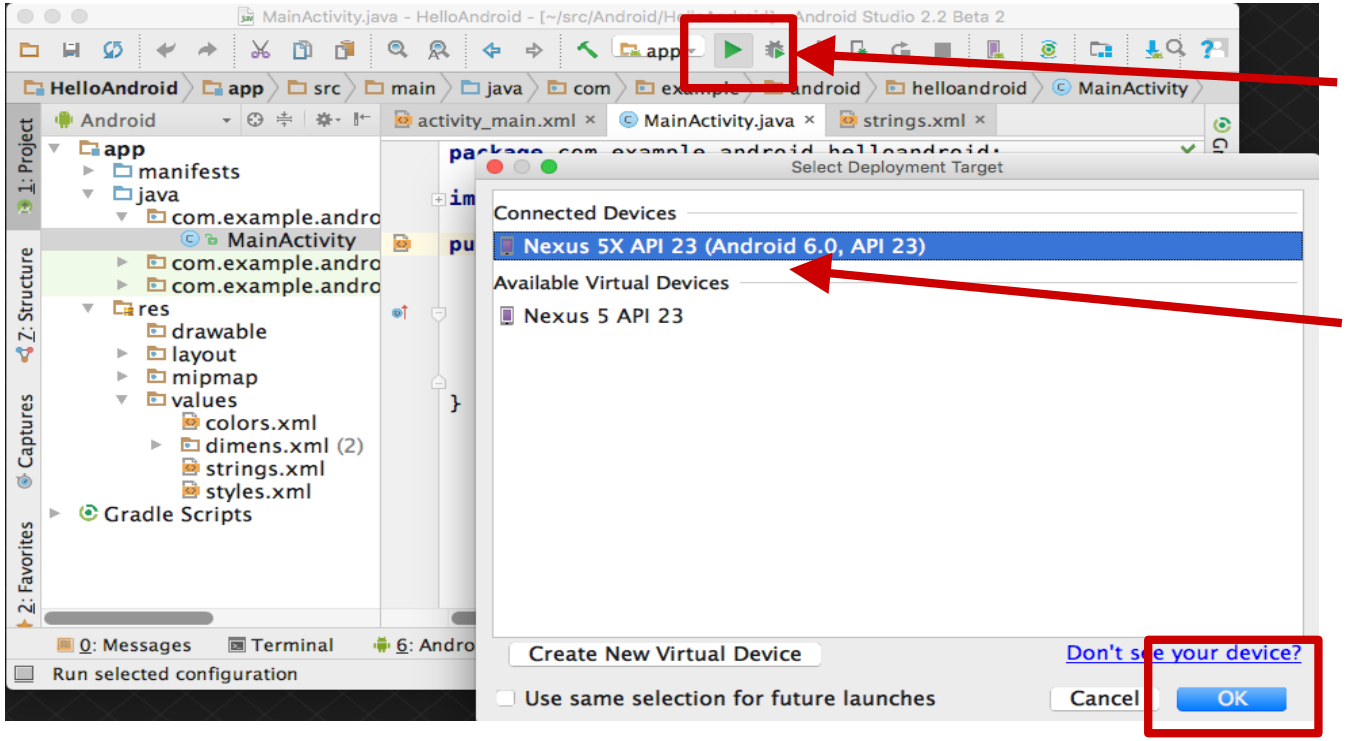
1. manifests—Android Manifest file - description of app read by the Android runtime
2. java—Java source code packages
3. res—Resources (XML) - layout, strings, images, dimensions, colors...
4. build.gradle—Gradle build files



Gradle build system

- Modern build subsystem in Android Studio
- Three build.gradle:
  - project
  - module
  - settings
- Typically not necessary to know low-level Gradle details
- Learn more about gradle at https://gradle.org/

Run your app

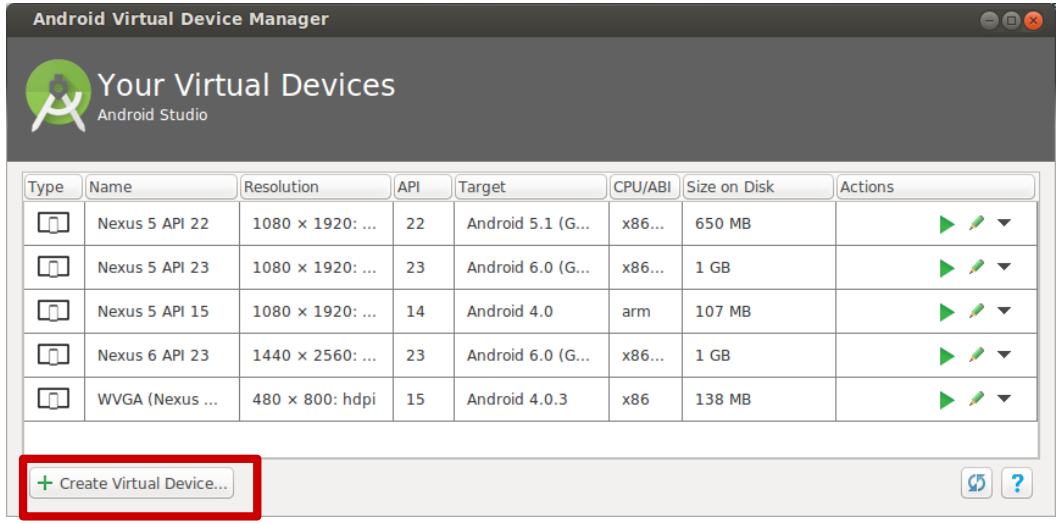


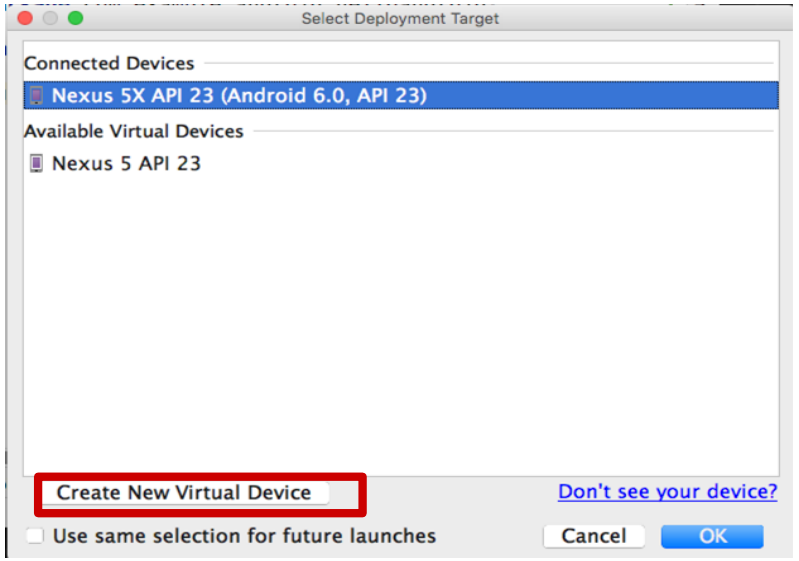
1. Run
2. Select virtual or physical device
3. OK

Create a virtual device

Use emulators to test app on different versions of Android and form factors.

Tools > Android > AVD Manager or:



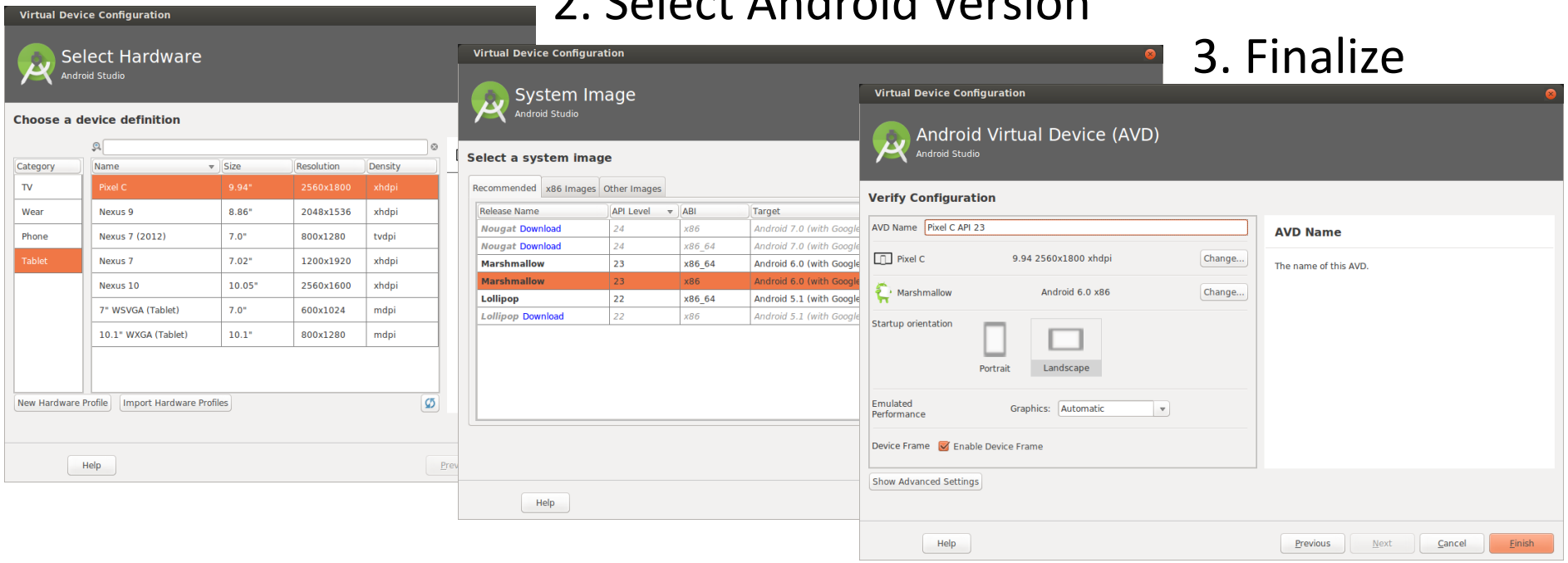


Configure Virtual Device

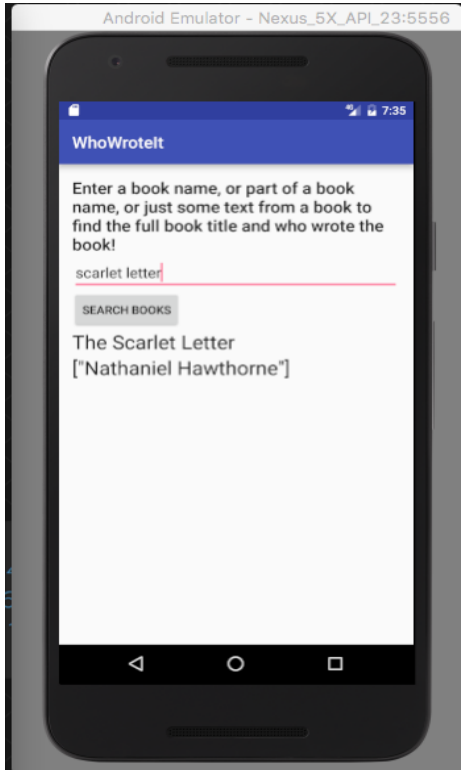
1. Choose hardware

2. Select Android Version

3. Finalize

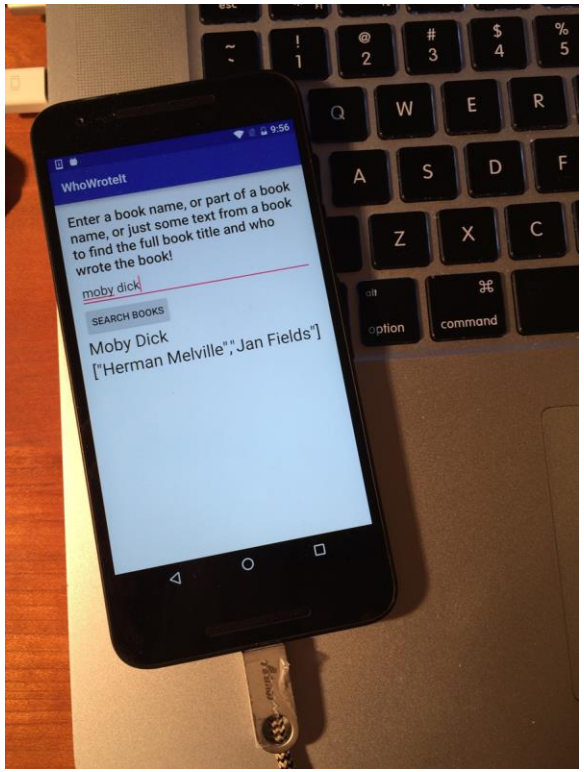


Run on a Virtual Device



Run on a physical device

1. Turn on Developer Options:
   a. **Settings > About phone**
   b. Tap **Build number** seven times
2. Turn on USB Debugging
   a. **Settings > Developer Options > USB Debugging**
3. Connect phone to computer with cable



Windows/Linux additional setup:

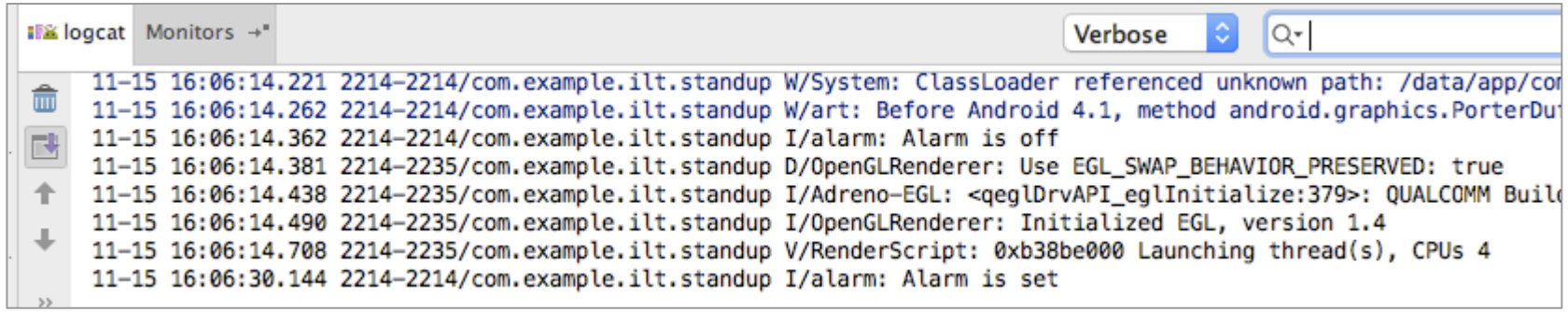
- Using Hardware Devices

Windows drivers:

- OEM USB Drivers

Get feedback as your app runs

- As the app runs, Android Monitor logcat shows information
- You can add logging statements to your app that will show up in logcat.



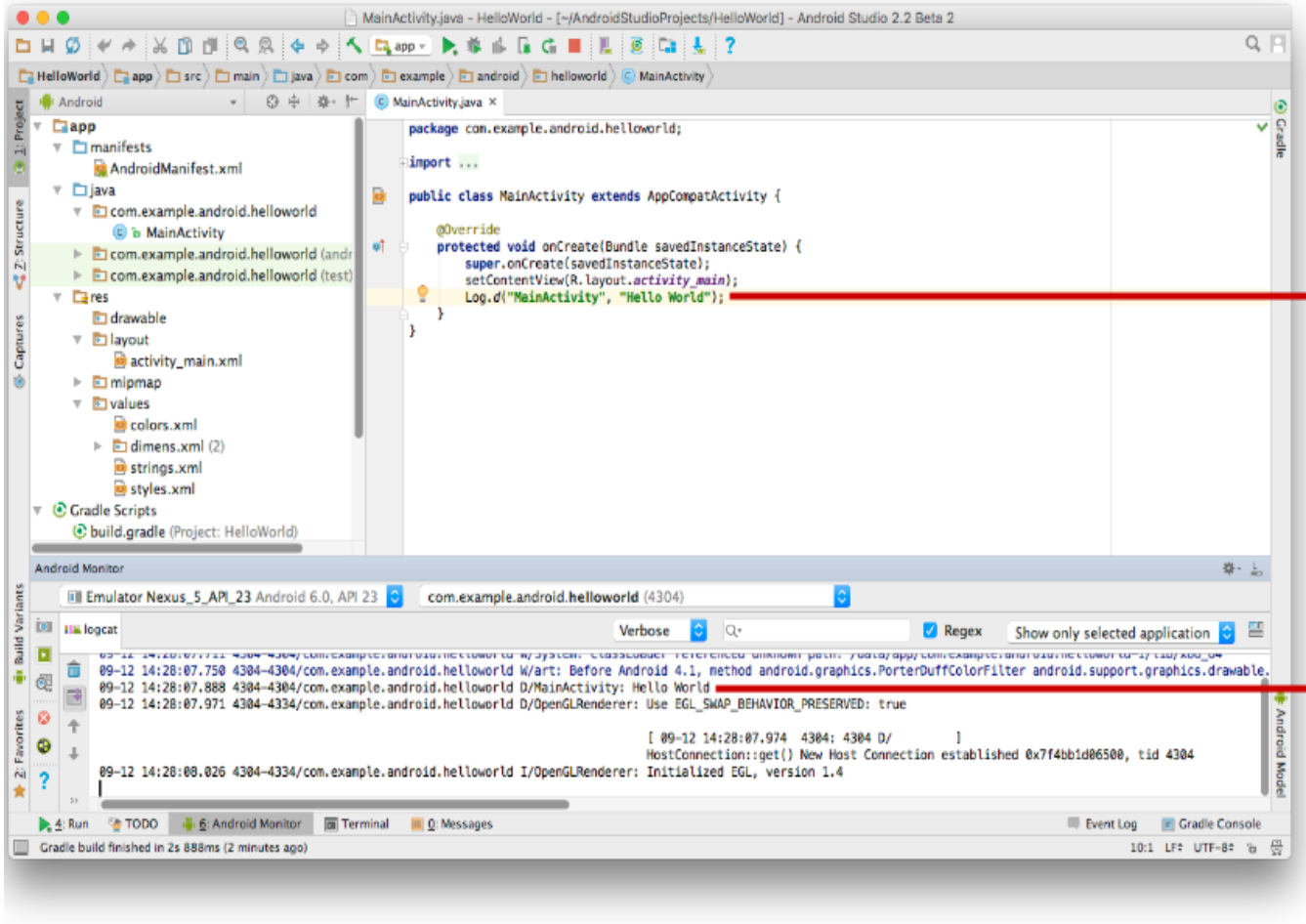
Logging

```
import android.util.Log;

// Use class name as tag
private static final String TAG =
    MainActivity.class.getSimpleName();

// Show message in Android Monitor, logcat pane
// Log.<log-level>(TAG, "Message");
Log.d(TAG, “Creating the URI…”);
```

Android Monitor > logcat pane



1. Log statements in code.
2. logcat pane shows system and logging messages

- Set filters to see what's important to you
- Search using tags

Any Question ?