

Android Development Overview



Topics

- Android and iOS Contrasted
- Android Tools
- Android Studio
- Android Studio Demo

Android vs iOS: Similarities

- Wide adoption / large developer communities
- Controlled by single companies
 - Android by Google
 - iOS by Apple
- Evolving rapidly
- Vibrant ecosystems

Android vs iOS: Differences

iOS

- Apple controls both software and hardware
- iOS is proprietary
- Xcode requires Mac
- Apple app approval takes significantly longer
- Language: Swift [some Objective-C]

Android

- Google control software stack only
- Android is open sources
- Android studio is cross platform
- Shorter app approval turnaround time
- Language: Kotlin [some Java]

Android @ Google IO 2024

- Gemini integration
- Jetpack Glance (custom widgets?)
- New UI design guidelines
- Camera-Viewfinder-Compose
- Jetpack Media3
- Vulkan + Angle + OpenGL ES
- Digital Credential Manager
- Wear OS 5
- Android Health/Health Connect
- Stylus
- Jetpack Compose Compiler with Kotlin 2.x
- Kotlin Multiplatform

New in Android Studio (Google IO 2024)

- Gemini in Android Studio (Studio Bot)
- Preview (Android Studio Koala)
- More Firebase integration
 - App Deployment Dashboard
 - Crash Analytics
 - Android Device Streaming

Tooling

- Android

- Android Studio (based on IntelliJ latest update is usually lagging behind IntelliJ update)
- Latest version: Koala (Koala Feature Drop)
- Google's IDE for developing apps on Android devices
- Successor to early Eclipse / ADT Plugin
- Cross platform: Windows, Linux, OS X



- iOS

- Xcode 15.x
- Apple's IDE for apps running on iOS, iPadOS, watchOS, macOS, and tvOS
- Only on OSX



Android Studio Tooling



Version	Release
1.x	2014 - 2015
2.x	2016 - 2017
3.x	2018 - 2020
4.x	2020 - 2021

Version Name	Release
Arctic Fox (based on IntelliJ 2020.3.1)	July 2021
Bumblebee (2021.1.1)	Jan 2022
Chipmunk (2021.2.1)	May 2022
Dolphin (2021.3.1)	Sep 2022
Eclectic Eel (2022.1.1)	Jan 2023
Flamingo (2022.2.1)	April 2023
Giraffe (2022.3.1)	July 2023
Hedgehog (2023.1.1)	Nov 2023

Version Name	Release
Iguana (2023.2.1)	Feb 2024
JellyFish (2023.3.1)	Apr 2024
Koala (2024.1.1)	Jun 2024
Koala "Feature Drop" (2024.1.2)	> Jun 2024

Android Studio Features

- Integration with git
- Autocompletion / syntax highlighting / code folding
- Integrated debugger
- Integrated profiler
- Layout Editor to ease UI implementation
- Collapsible panes



Gradle Build System

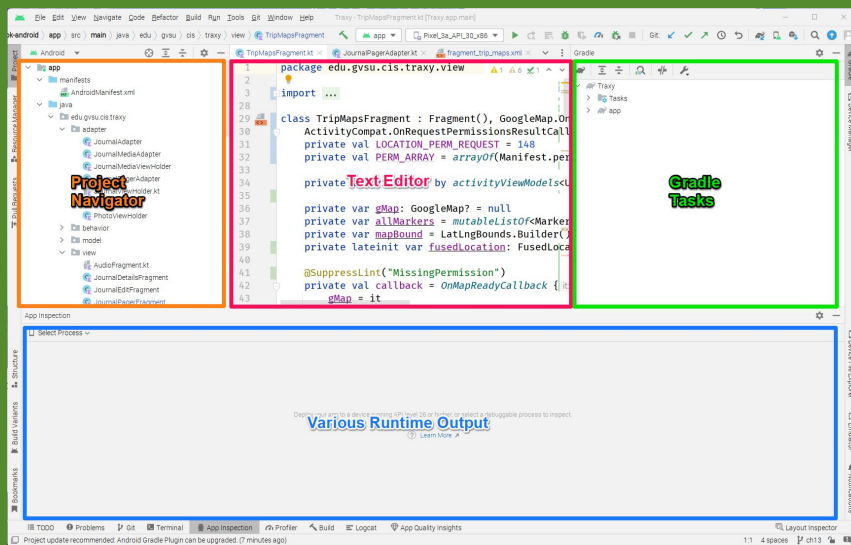
- Build Automation Tool
 - Like Makefile, CMake
- Android-specific functionality provided by plugins (“browser extensions”)
- Manage library dependencies



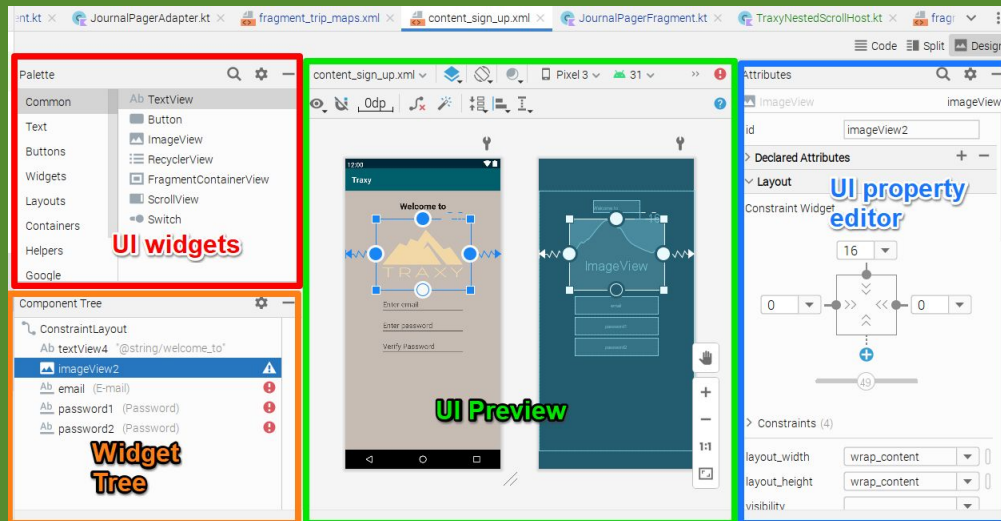
Android vs. iOS

	iOS	Android
Programming Model	Model-View-Controller (MVC)	Model-View-ViewModel (MVVM)
App Unique Identifier	Bundler ID	Package Name
Language	.swift	Kotlin (.kt)
UI design	NIB, XIB	Layout (.xml), Jetpack Compose (.kt)
App Navigation	Storyboard	Navigation Graph
Virtual Environment	Simulator	Emulator + Android Virtual Device

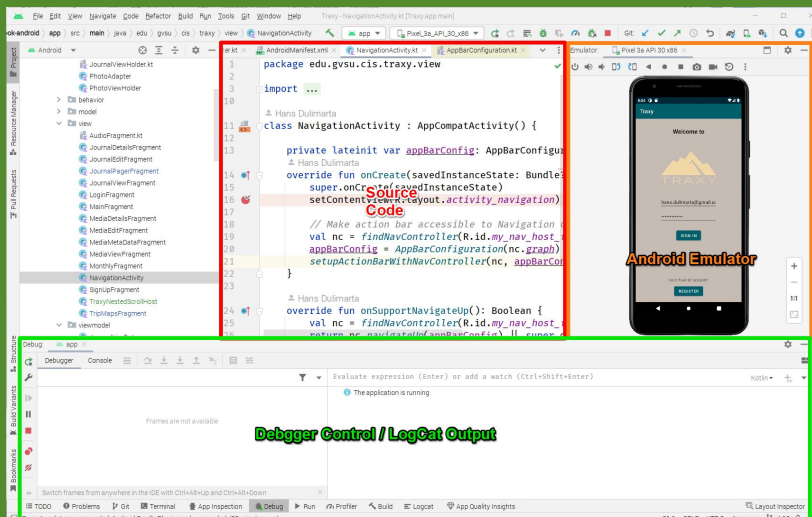
Android Studio Main Screen



Android Studio Layout Editor



Android Studio Debugger



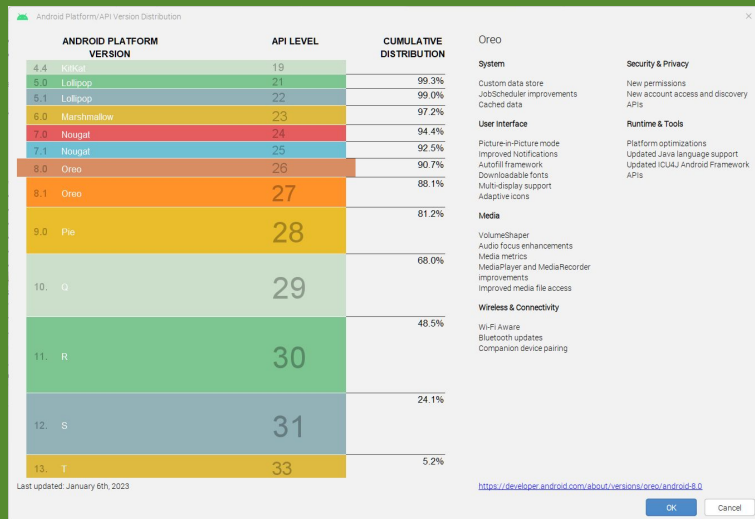
Android SDK Versions / API Levels

Version	Codename	API Level	Release Date	Key Features
1.5	Cupcake	3	Apr 2009	Virtual Kbd, Clipboard
1.6	Donut	4	Sep 2009	Multiple screen densities
2.0 - 2.1	Eclair	5 - 7	Oct 2009	Speech Recognition
2.2	Froyo	8	May 2010	Voice Actions
2.3	Gingerbread	9-10	Dec 2010	
3.0 - 3.2	Honeycomb	11 - 13	Feb 2011	Tablet support
4.0	Ice Cream Sandwich	14 - 15	Oct 2011	Gestures
4.1 - 4.3	Jelly Bean	16 - 18	Jul 2012	Project Butter: smoother animation
4.4	Kit Kat	19 - 20	Oc 2013	Project Svelte: memory optimization
5.0 -5.1	Lollipop	21 - 22	Nov 2014	Material Design

Android SDK Versions / API Levels

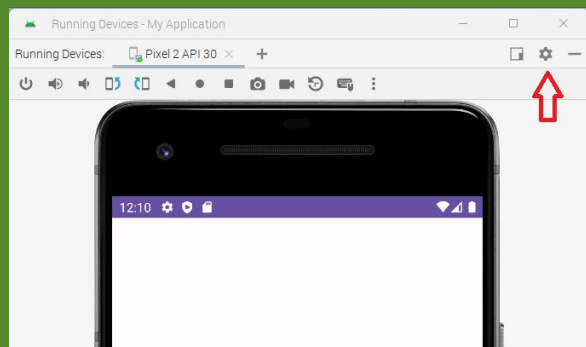
Version	Codename	API Level	Release Date	Key Features
6.0	Marshmallow	23	Oct 2015	Runtime Permission
7.x	Nougat	24 - 25	Aug 2016	Virtual Reality
8.0	Oreo	26 - 27	Aug 2017	Project Treble
9.0	Pie	28	Aug 2018	AI
10	Android 10	29	Sep 2019	Dark Mode, Improved Security
11	Android 11	30	Sep 2020	Conversations and Predictive tools
12	Android 12	31 - 32	Oct 2021	Material You
13	Android 13	33	Aug 2022	Enhanced privacy and Bluetooth updates
14	Android 14	34	Oct 2023	Health Connect (such as Fitbit, Google Fit)
15	Android 15	35	Jun 2024	Better Camera Control, Better Media, MIDI, PDF Improvement, Font, I18N, Vulkan

New Project: Help Me Choose



Android Emulators

- Run a full Android System Stack
 - Linux OS
- Create Android Virtual Devices (AVDs) to emulate various Android devices
- May not emulate some hardware: NFC, Bluetooth
- Docked/Detached View
 - **Emulator Setting Icon** ⇒ View Mode



Enable Developer Option on Android Devices

1. Go to Settings ⇒ About Phone
2. [Go to “About Phone” or “About Software”]
3. Scroll Down until you see “Build number”
4. Tap “Build number” several times

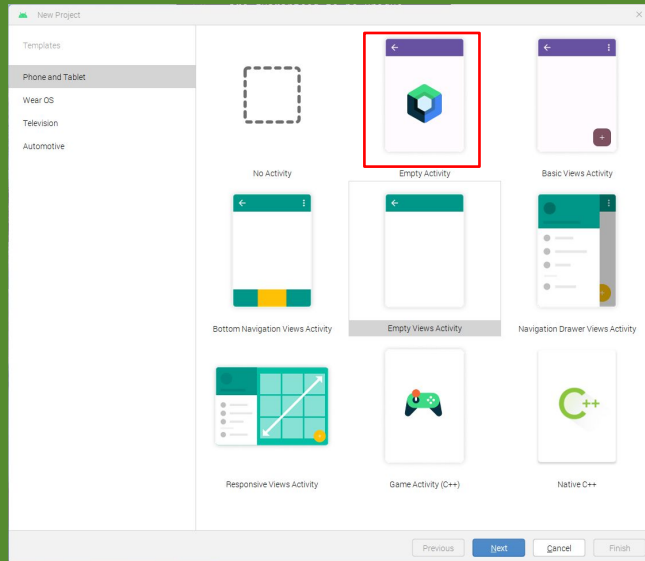
After Developer Option is Enabled

1. Go to Settings ⇒ System ⇒ Developer Options

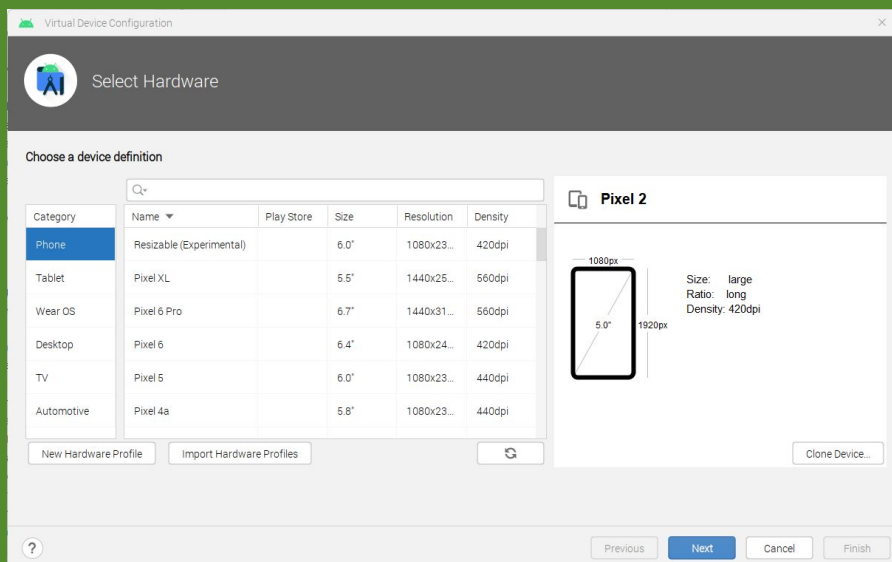
Build/Run First Android App



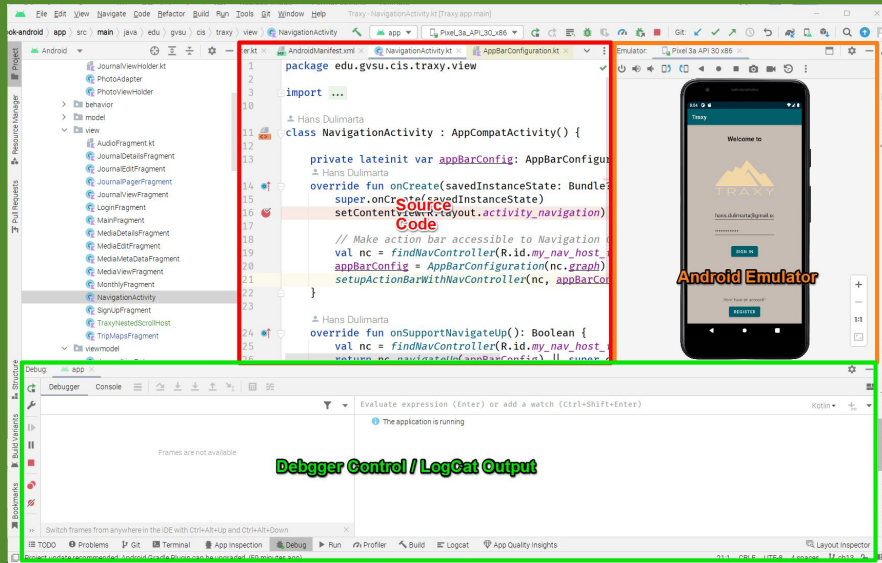
New Project ⇒ Empty Activity



Android Virtual Devices



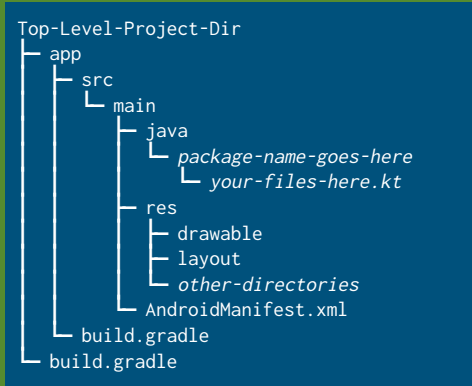
Debugger



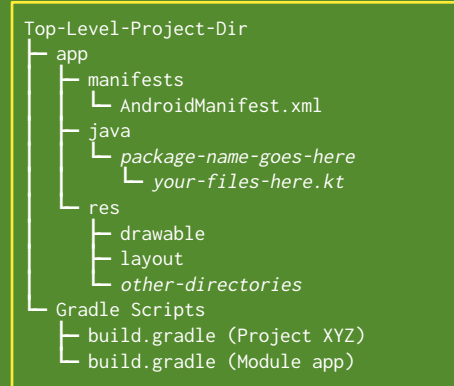
Android Project Directory Structure

- Project Viewer options
 - Project
 - Project Files
 - **Android** (select this one)
- Important Project Files
 - AndroidManifest.xml
 - Kotlin/Java Files
 - Resource Directories
 - Drawable: icons, images, vector graphics
 - Layout: UI design
 - Strings (for language translation)
 - Menu structure

Android Project Directory Structure (Live Demo)



Physical Directory Structure



Logical Android Organization

Gradle Files: Groovy vs. Kotlin Script

```
// Groovy
android {
    namespace "edu.gvsu.cis.deletemelater"
    compileSdk 34

    defaultConfig {
        applicationId "edu.gvsu.cis.deletemelater"
        minSdk 24
        targetSdk 34
        versionCode 1
        versionName "1.0"
    }

    kotlinOptions {
        jvmTarget "1.8"
    }
}
```

build.gradle

```
// Kotlin Script
android {
    namespace = "edu.gvsu.cis.deletemelater"
    compileSdk = 34

    defaultConfig {
        applicationId = "edu.gvsu.cis.deletemelater"
        minSdk = 24
        targetSdk = 34
        versionCode = 1
        versionName = "1.0"
    }

    kotlinOptions {
        jvmTarget = "1.8"
    }
}
```

build.gradle.kts

Managing SDKs in Project Configuration

SDK Option	Used At	Description
compileSDKVersion	Compile/Build Time	Determine which features are available at build time
targetSDKVersion	Run-Time	Determine the app behavior at run-time
minSDKVersion	Google Play Store	Determine if app is compatible with user device

General guidelines:

- Keep the minSDKVersion as lowest as possible so more users can install your app
 - However, they may not be able to use **all** the available features
- Keep the compileSDKVersion as **lowest as possible** for the set of feature(s) needed in your app
- Keep the targetSDKVersion as high as possible, but you have to *test and confirm* that your app runs as expected at that API level

Reading Assignment

Engelsma/Dulimarta textbook

- Chapters 1.2, 1.5
- Chapters 2.2, 2.4



Class Exercise

1. Create a new Hello World Project (View-based)
2. Create a new Virtual Device
3. Deploy & Run to Emulator
4. [Deploy & Run to Actual Phone]
5. Enabling Developer Option on Actual Phones
6. Create a new Hello World Project (Jetpack Compose)
7. Deploy & Run to Emulator
8. [Deploy & Run to Actual Phone]