Revised: 7/23/2019

Android Virtual Devices

Michael C. Hackett

Computer Science Department

Community College of Philadelphia

What is an Android Virtual Device?

- An Android Virtual Device is an emulated cell phone, tablet, or other device capable of running a version of Android OS.
  - Emulator (like AVDs): Uses software implementation of the hardware contained in a physical device like a cell phone or tablet.
    - In an AVD, **all** hardware for the device is emulated.
    - Similar to a Virtual Machine.
  - Simulator (like the iPhone simulator in Xcode): Only simulates an iPhone or iPad.

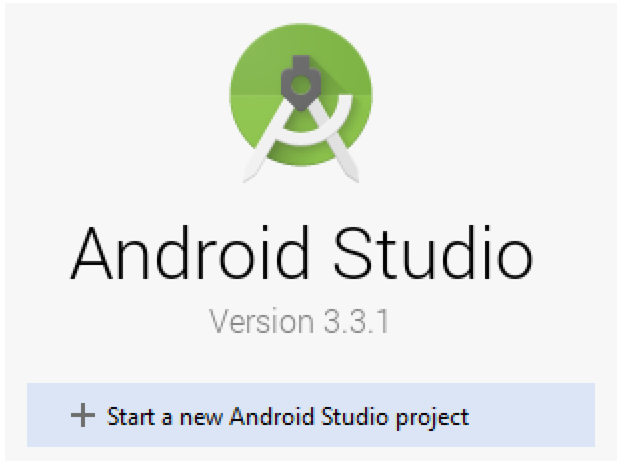
  - Emulators are usually slower than simulators.

Why use an Android Virtual Device?

- AVDs can be created for a wide range of devices.
  - This allows for development and testing of apps for many different devices, without needing (potentially hundreds of) physical phones or tablets.
- It’s a bit easier developing for iPhones and iPads.
  - iOS only runs on a limited number of Apple devices, all generally using the same hardware.
- Android is everywhere.
  - Different hardware, screen sizes, physical buttons, etc.

Setting up an Android Virtual Device

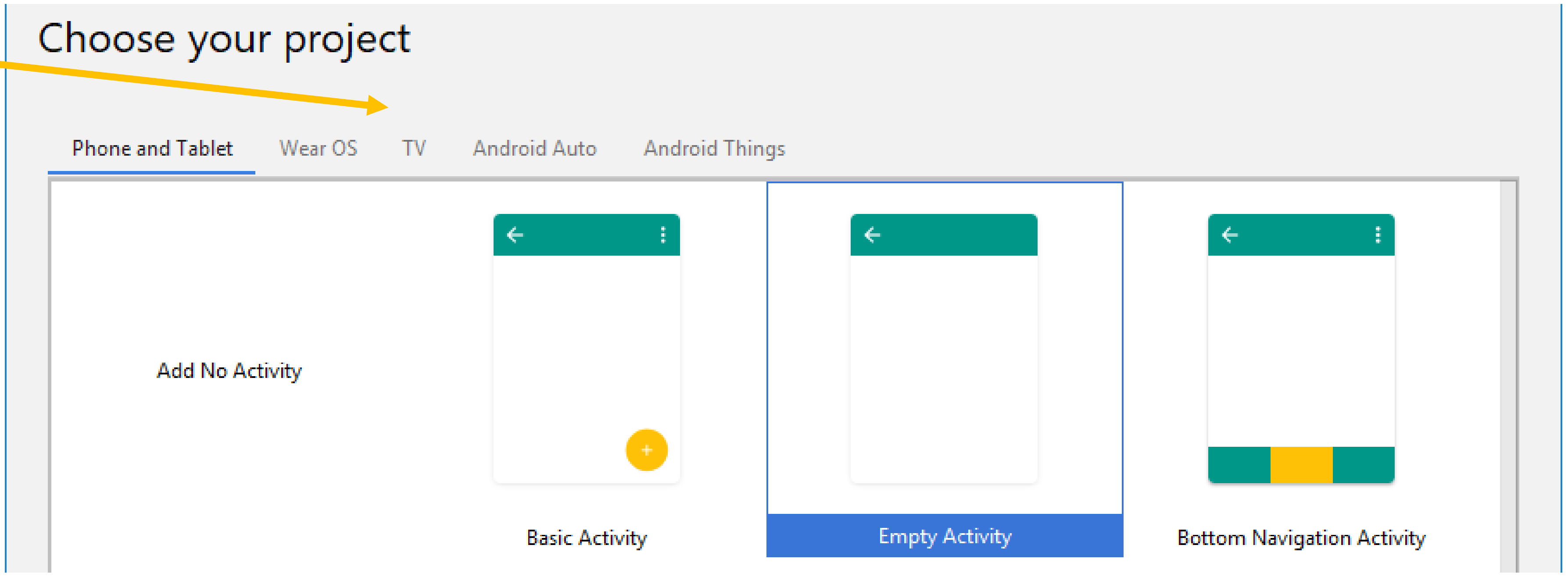
- This exercise goes through the process to create and start an AVD.
- Open Android Studio and select **Start a new Android Studio project**.
  - Be sure to first complete the Android Studio Setup instructions in the Track A section of Module 5.



Setting up an Android Virtual Device

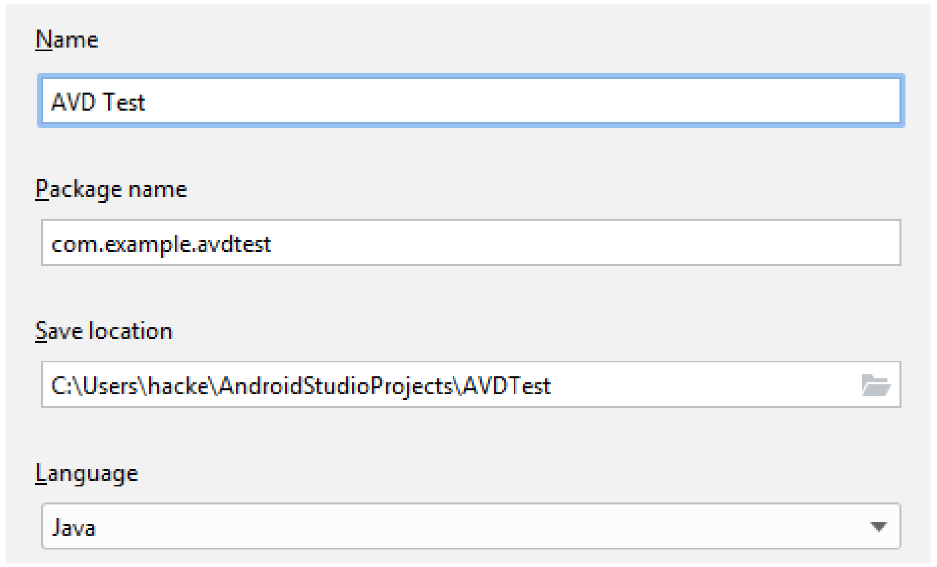
- Be sure **Empty Activity** is selected and click **Next**.

Other types of
Android development



Setting up an Android Virtual Device

- Name the project **AVD Test**



Name

AVD Test

Package name

com.example.avdtest

Save location

C:\Users\hacke\AndroidStudioProjects\AVDTest

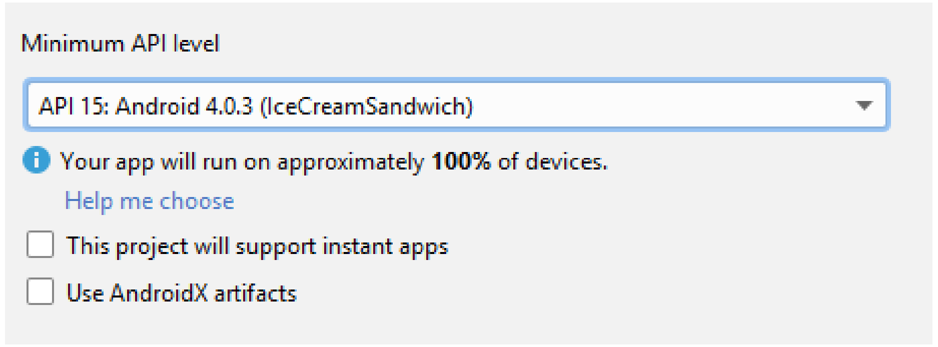
Language

Java

Setting up an Android Virtual Device

- The bottom section of this page lets you set the minimum SDK (Android version) that this app will support.
  - Leave it set to the default choice.

- Click **Finish**



Minimum API level

API 15: Android 4.0.3 (IceCreamSandwich)

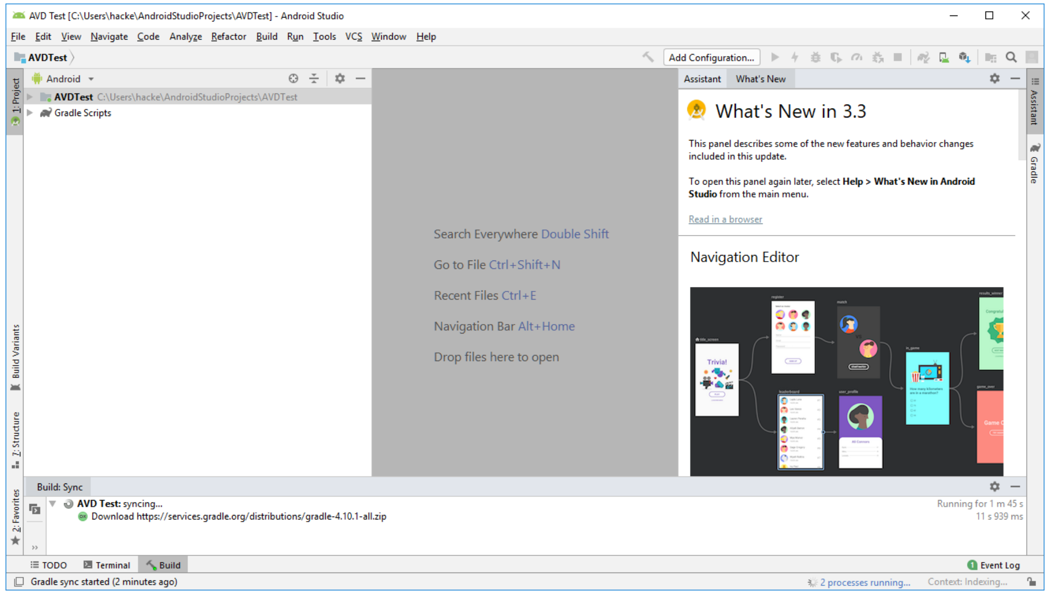
Your app will run on approximately **100%** of devices.

Help me choose

- [ ] This project will support instant apps
- [ ] Use AndroidX artifacts

Android Studio

- Android Studio's interface should look familiar.
  - It is based on IntelliJ
- If you know IntelliJ, you know Android Studio.
  - Android development is primarily done using Java.
    - If you can program in Java, you can program Android apps.



What's New in 3.3

This panel describes some of the new features and behavior changes included in this update.

To open this panel again later, select **Help > What's New in Android Studio** from the main menu.

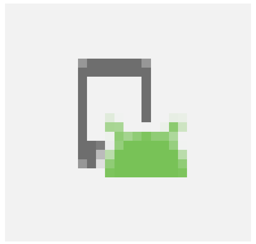
Read in a browser

Navigation Editor

```
AVD Test: syncing...
    Download https://services.gradle.org/distributions/gradle-4.10.1-all.zip
                                                        Running for 1 m 45 s
                                                                11 s 939 ms
Gradle sync started (2 minutes ago)          2 processes running...   Context: Indexing...
```

Setting up an Android Virtual Device

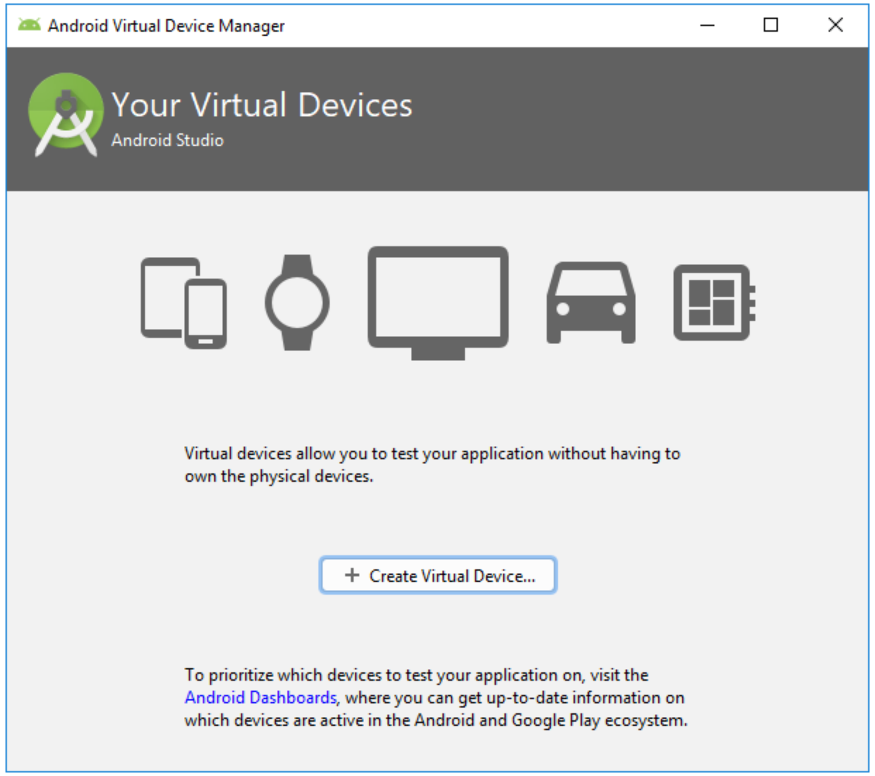
- In the top right corner of Android Studio, you'll see the icon for the AVD Manager.



- Click this icon to open the AVD Manager.

Setting up an Android Virtual Device

- Click **Create Virtual Device**

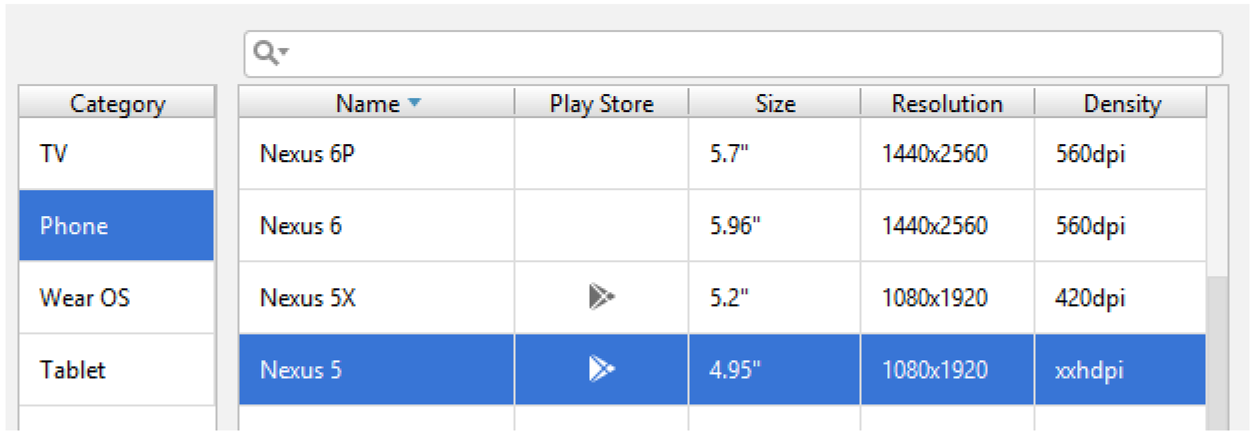


Virtual devices allow you to test your application without having to own the physical devices.

To prioritize which devices to test your application on, visit the Android Dashboards, where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.

Setting up an Android Virtual Device

- A list of known (Google) devices will be shown.
  - Profiles for other devices can be imported.
- Select **Nexus 5** from the list of choices.
- Click **Next**

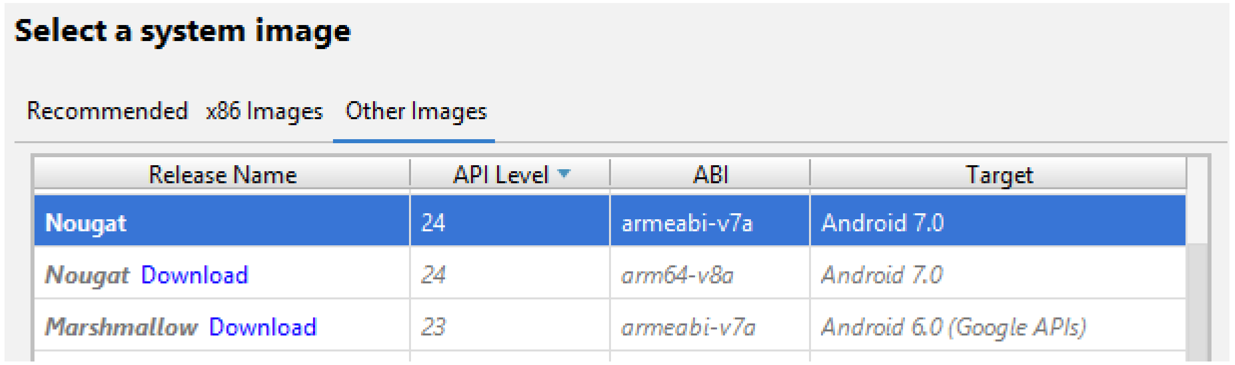


Category
TV
Phone
Wear OS
Tablet

Name	Play Store	Size	Resolution	Density
Nexus 6P		5.7"	1440x2560	560dpi
Nexus 6		5.96"	1440x2560	560dpi
Nexus 5X	▶	5.2"	1080x1920	420dpi
Nexus 5	▶	4.95"	1080x1920	xxhdpi

Setting up an Android Virtual Device

- Click the **Other Images** tab.
- Click the **Download** link next to **Nougat** (Android 7.0/API Level 24)
  - ABI: armeabi-v7a
  - The Download link is missing below because I already have it installed.



Select a system image

Recommended x86 Images Other Images

Release Name	API Level	ABI	Target
Nougat	24	armeabi-v7a	Android 7.0
Nougat Download	24	arm64-v8a	Android 7.0
Marshmallow Download	23	armeabi-v7a	Android 6.0 (Google APIs)

- On the License Agreement window that opens, select **Accept** and then click **Next**
  - Wait for the download to finish.
  - When completed, click the **Finish** button.

Setting up an Android Virtual Device

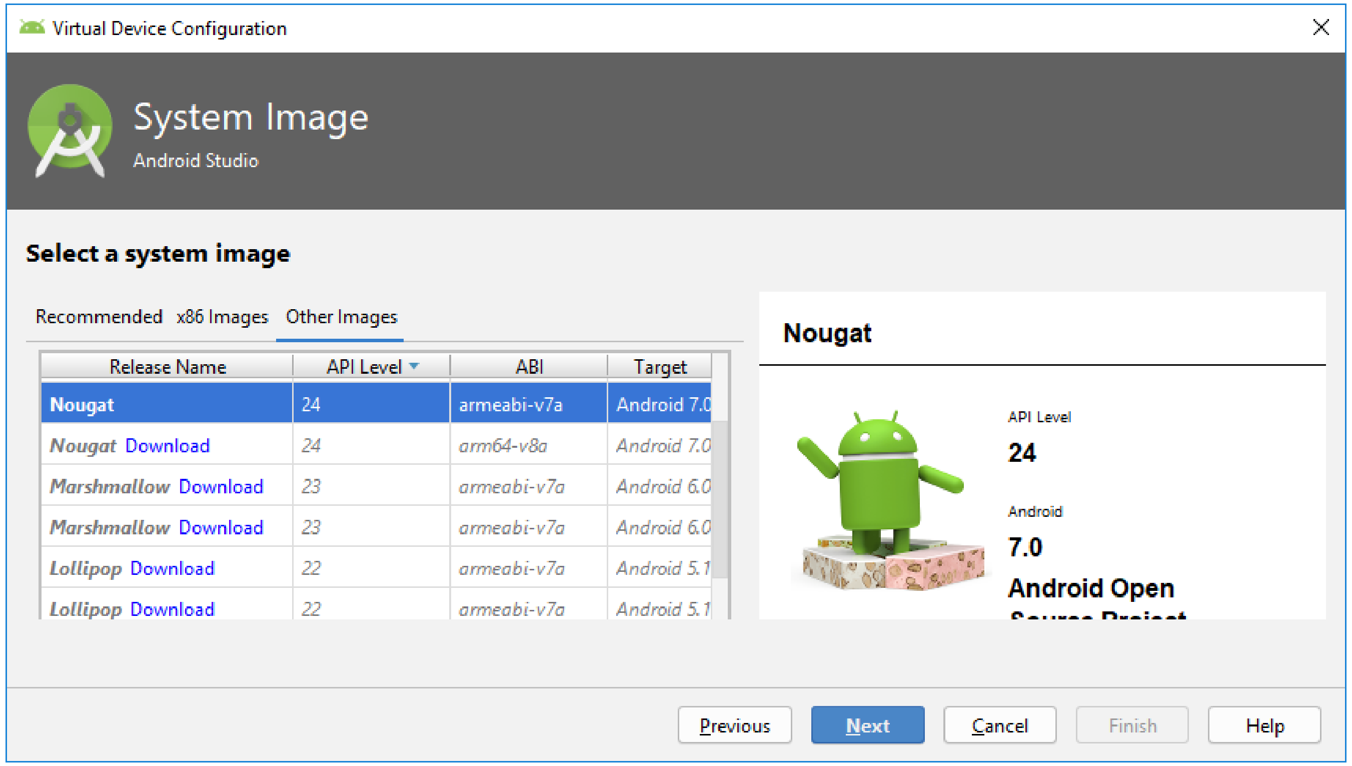
- The ARM architecture images are slower than the x86 images.

- X86 images often have trouble running on AMD processors (like mine) and can run into virtualization issues.
  - It’s easier to use something that I know will work for everyone.

- ARM images are not yet available for Android O or P.
  - For this course, it doesn’t really matter that we are using an older version/older phone.

Setting up an Android Virtual Device

- Click the **Next** button



Virtual Device Configuration

System Image
Android Studio

Select a system image

Recommended x86 Images Other Images

Release Name	API Level	ABI	Target
Nougat	24	armeabi-v7a	Android 7.0
Nougat Download	*24*	*arm64-v8a*	*Android 7.0*
Marshmallow Download	*23*	*armeabi-v7a*	*Android 6.0*
Marshmallow Download	*23*	*armeabi-v7a*	*Android 6.0*
Lollipop Download	*22*	*armeabi-v7a*	*Android 5.1*
Lollipop Download	*22*	*armeabi-v7a*	*Android 5.1*

Nougat

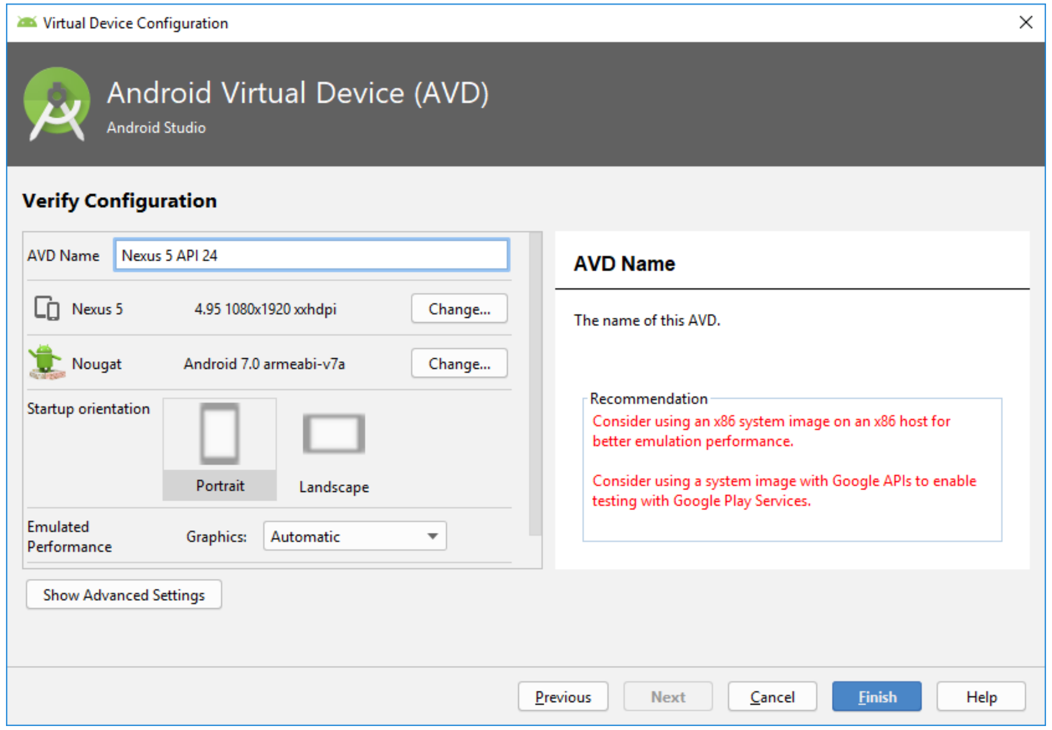
API Level
24

Android
7.0

Android Open

Setting up an Android Virtual Device

- Keep the default settings.
- Click **Finish**.



Virtual Device Configuration

Android Virtual Device (AVD)
Android Studio

Verify Configuration

AVD Name: Nexus 5 API 24

Nexus 5 — 4.95 1080x1920 xxhdpi — Change...

Nougat — Android 7.0 armeabi-v7a — Change...

Startup orientation: Portrait / Landscape

Emulated Performance — Graphics: Automatic

Show Advanced Settings

AVD Name

The name of this AVD.

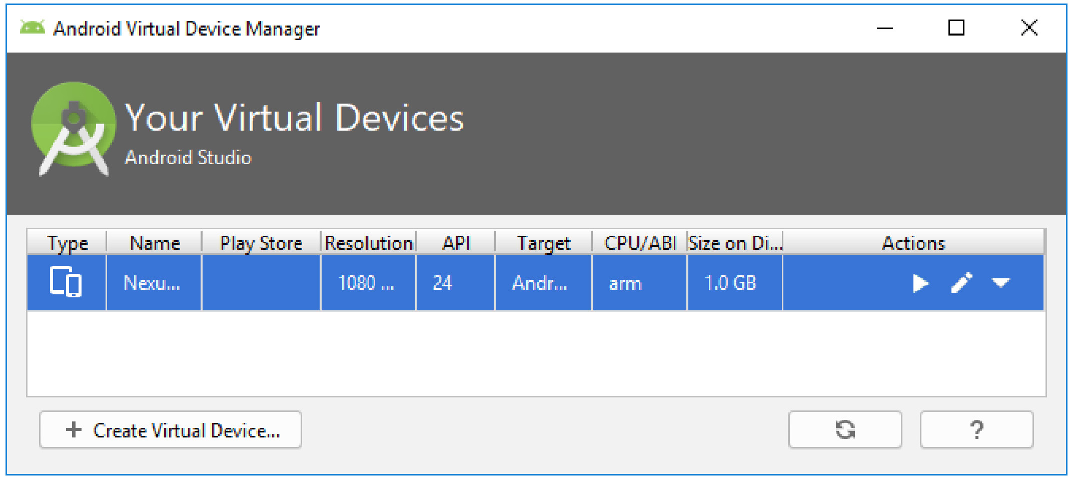
Recommendation

Consider using an x86 system image on an x86 host for better emulation performance.

Consider using a system image with Google APIs to enable testing with Google Play Services.

Setting up an Android Virtual Device

- With an AVD now setup, Android Studio will generate a bunch of files for the project.
- Click the AVD Manager icon again.
- You'll see the AVD we just created listed.



Android Virtual Device Manager

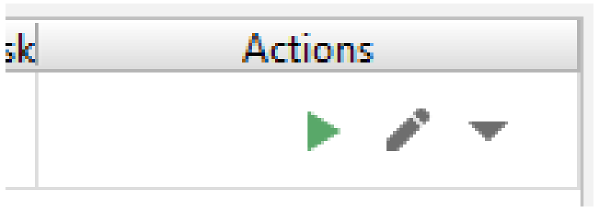
Your Virtual Devices

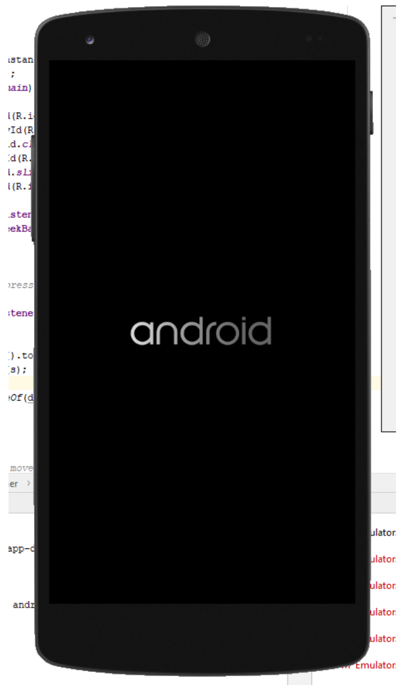
Android Studio

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Di...	Actions
	Nexu...		1080 ...	24	Andr...	arm	1.0 GB	

Setting up an Android Virtual Device

- To start the AVD, click the Play icon shown in the Actions column.

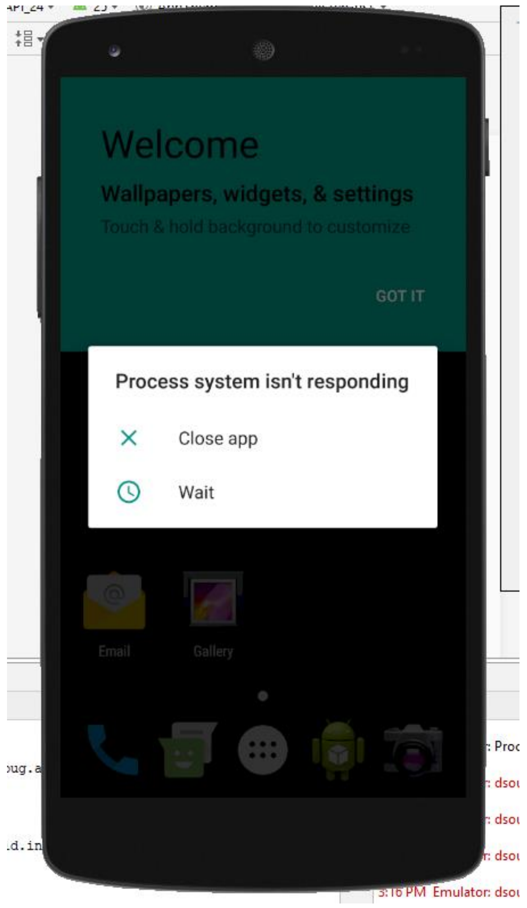




- Wait for the AVD to load.
  - This may take a while….

Setting up an Android Virtual Device

- It’s common to see this **Process system isn’t responding** message after the AVD has started.
  - It’ll be slow when it first starts.
    - It might take a few more moments for it to become responsive.
- Click **Close app**
- Click **GOT IT** to dismiss the Welcome message.


Welcome
Wallpapers, widgets, & settings
Touch & hold background to customize
GOT IT
Email
Gallery

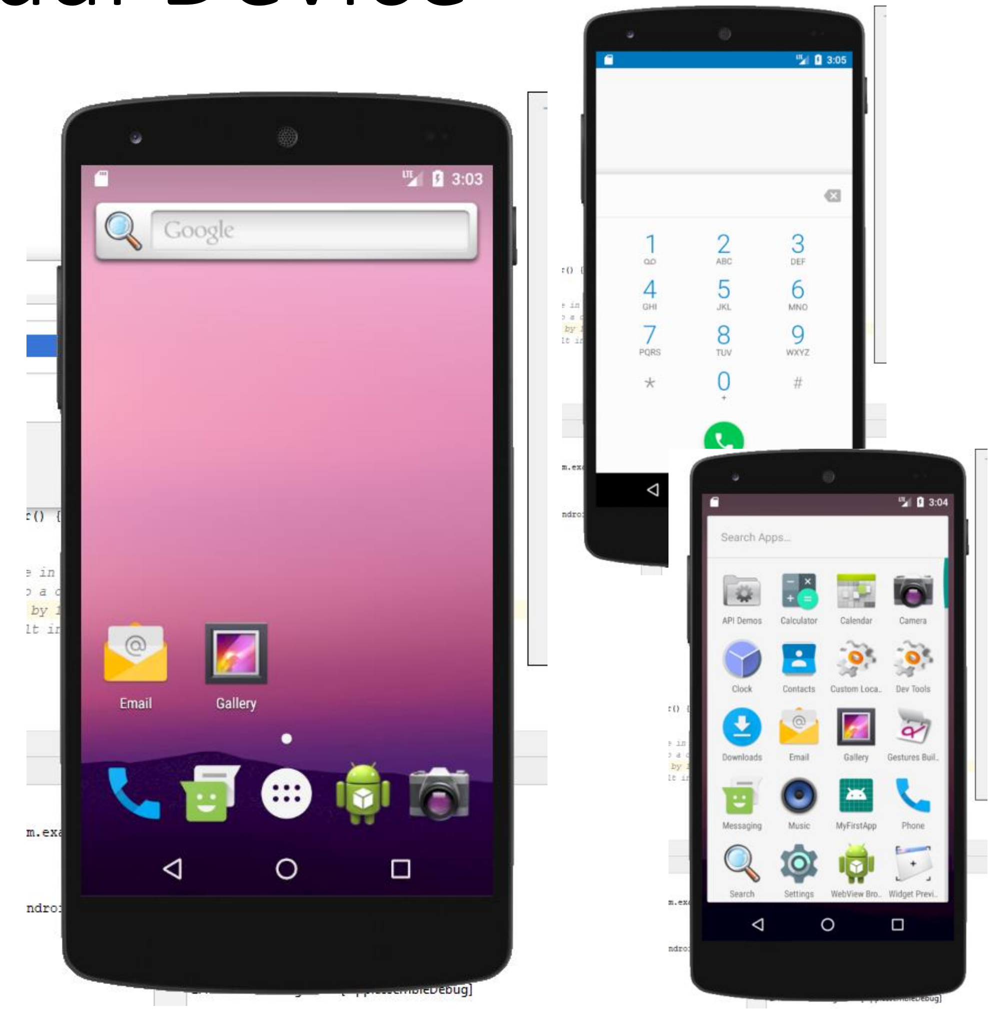
Process system isn't responding

Close app

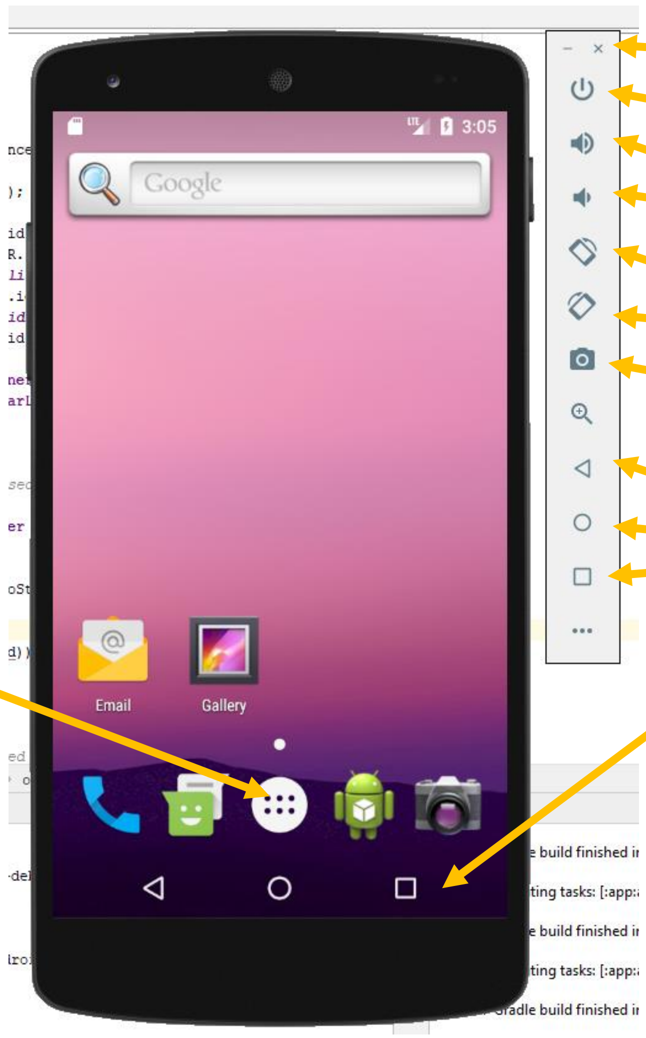
Wait

Setting up an Android Virtual Device

- At this point, the AVD is ready to go!
- Feel free to click/click-and-drag around on the emulated phone.
  - You essentially use the mouse pointer like the user's finger.



AVD Controls



MINIMIZE/CLOSE AVD

Turn off/on screen

Volume

Rotate orientation

Screen Shot

Soft Keys

Launcher