**Creating an AVD (Android Virtual Device) in Android Studio (AS)**

An Android Virtual Device (AVD) is a configuration that defines the characteristics of an Android phone, tablet, Wear OS, Android TV, or Automotive OS device that you want to simulate in the Android Emulator. The AVD Manager is an interface you can launch from Android Studio that helps you create and manage AVDs.

Configuration

Work the instructions as follows for any OS.

1. Click open AS if it is not opened. If you have any project opened up, close out of   
 that (File > Close Project) to get to your Welcome screen.

If it’s the first time creating a Virtual Device, you will most likely see a More Actions pull down menu (for sure with Windows) as shown below.

A screen shot of a computer

Description automatically generated

Go ahead and click the menu and choose Virtual Device manager and continue with   
 the step 2 on the next page.

[ For future Device creations, once you created a project, at the Welcome screen you can always click on the ellipsis located on the right side of your Welcome screen and choose Virtual Device Manager as shown below].

A screenshot of a computer

Description automatically generated

2. You should now see the Device Manager screen opened. Click on the link at the center of your screen namely, Create virtual device… link. Alternatively, you can click on Create Virtual Device or the + icon as shown below, at your top left side of your Device Manager screen to open some example device selections, especially for any future devices to add on.

A screenshot of a computer

Description automatically generated

-------------------------------------------------------------------------------------------------------

From the Virtual Device Configuration screen, click **Pixel 7 Pro** as a sample selection. Notice the device settings allows for Play Store as an option which is great to allow for connectivity with Google APIs.

A screenshot of a computer

Description automatically generated

Press Next to continue.

Click on the suggested Download link icon  for the *UpSideDownCake* image, API Level 34 to start your download.

A screenshot of a computer

Description automatically generated

If presented with a EULA agreement, agree to it, then click Next to commence the downloading and installation which loads a system image for the chosen device.

Click Finish when complete.

Click Next to Verify the System Image configuration settings. Then click on Finish.

If you like to do a test run of your AVD, at the ‘Android Virtual Devices Manager’ screen, click on the Launch  button at your right-hand side as shown below.

A blue and black rectangle with white text

Description automatically generated

Here a snapshot as shown next, of the resulting emulator for the Pixel 7 Pro

A screenshot of a phone

Description automatically generated

That’s it for now. Exit out of your Device Manager window, close your Emulator device and Welcome to Android Studio window. You can now use your AVD choice when you are creating and running your apps as you will see! If you choose to run/configure a Virtual Device (aka Emulator) once a project is created, open the AVD Manager within AS, by choosing **Tools > Device Manager** from your menu or if you’re at the Welcome screen just click on the vertical ellipsis button to check that option as shown below.

A screenshot of a computer

Description automatically generated

Refs:

<https://developer.android.com/studio/run/managing-avds#avdproperties>

<https://developer.android.com/about/dashboards/index.html>

[Android 14  |  Android Developers](https://developer.android.com/about/versions/14) (latest stable release)

[Android 15  |  Android Developers](https://developer.android.com/about/versions/15) (beta)

Android Platform / API Version Distribution listing up to API LEVEL 33 follows…

A screenshot of a computer

Description automatically generated

Note – the API level of the target device is important, because your app won't be able to run on a system image with an API level that's less than that required by your app, as specified in the**minSdkVersion** attribute of the app manifest file. For more information about the relationship between system API level and minSdkVersion, see [Versioning Your Apps](https://developer.android.com/studio/publish/versioning).

If your app declares a [<uses-library>](https://developer.android.com/guide/topics/manifest/uses-library-element) element in the manifest file, the app requires a system image in which that external library is present. If you want to run your app on an emulator, create an AVD that includes the required library. To do so, you might need to use an add-on component for the AVD platform; for example, the Google APIs add-on contains the Google Maps library.