

6. Using and Configuring the Android Studio AVD Emulator

The Android Virtual Device (AVD) emulator environment bundled with Android Studio 1.x was an uncharacteristically weak point in an otherwise reputable application development environment. Regarded by many developers as slow, inflexible and unreliable, the emulator was long overdue for an overhaul. Fortunately, Android Studio 2 introduced an enhanced emulator environment providing significant improvements in terms of configuration flexibility and overall performance and further enhancements have been made for Android Studio 3.

Before the next chapter explores testing on physical Android devices, this chapter will take some time to provide an overview of the Android Studio AVD emulator and highlight many of the configuration features that are available to customize the environment.

6.1 The Emulator Environment

When launched, the emulator displays an initial splash screen during the loading process. Once loaded, the main emulator window appears containing a representation of the chosen device type (in the case of [Figure 6-1](#) this is a Nexus 5X device):



Figure 6-1

Positioned along the right-hand edge of the window is the toolbar providing quick access to the emulator controls and configuration options.

6.2 The Emulator Toolbar Options

The emulator toolbar ([Figure 6-2](#)) provides access to a range of options relating to the appearance and behavior of the emulator environment.

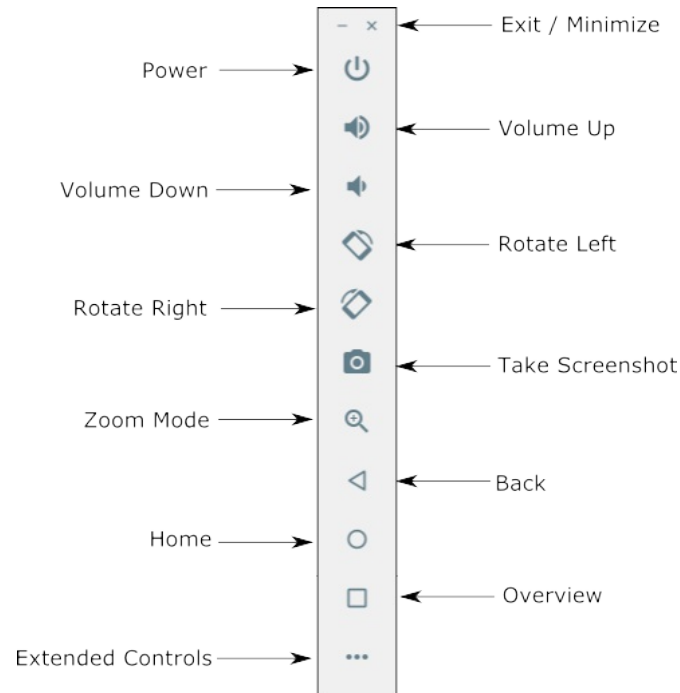


Figure 6-2

Each button in the toolbar has associated with it a keyboard accelerator which can be identified either by hovering the mouse pointer over the button and waiting for the tooltip to appear, or via the help option of the extended controls panel.

Though many of the options contained within the toolbar are self-explanatory, each option will be covered for the sake of completeness:

- **Exit / Minimize** – The uppermost ‘x’ button in the toolbar exits the emulator session when selected while the ‘-’ option minimizes the entire window.
- **Power** – The Power button simulates the hardware power button on a physical Android device. Clicking and releasing this button will lock the device and turn off the screen. Clicking and holding this button will initiate the device “Power off” request sequence.
- **Volume Up / Down** – Two buttons that control the audio volume of playback within the simulator environment.

- **Rotate Left/Right** – Rotates the emulated device between portrait and landscape orientations.
- **Screenshot** – Takes a screenshot of the content currently displayed on the device screen. The captured image is stored at the location specified in the Settings screen of the extended controls panel as outlined later in this chapter.
- **Zoom Mode** – This button toggles in and out of zoom mode, details of which will be covered later in this chapter.
- **Back** – Simulates selection of the standard Android “Back” button. As with the Home and Overview buttons outlined below, the same results can be achieved by selecting the actual buttons on the emulator screen.
- **Home** – Simulates selection of the standard Android “Home” button.
- **Overview** – Simulates selection of the standard Android “Overview” button which displays the currently running apps on the device.
- **Extended Controls** – Displays the extended controls panel, allowing for the configuration of options such as simulated location and telephony activity, battery strength, cellular network type and fingerprint identification.

6.3 Working in Zoom Mode

The zoom button located in the emulator toolbar switches in and out of zoom mode. When zoom mode is active the toolbar button is depressed and the mouse pointer appears as a magnifying glass when hovering over the device screen. Clicking the left mouse button will cause the display to zoom in relative to the selected point on the screen, with repeated clicking increasing the zoom level. Conversely, clicking the right mouse button decreases the zoom level. Toggling the zoom button off reverts the display to the default size.

Clicking and dragging while in zoom mode will define a rectangular area into which the view will zoom when the mouse button is released.

While in zoom mode the visible area of the screen may be panned using the horizontal and vertical scrollbars located within the emulator window.

6.4 Resizing the Emulator Window

The size of the emulator window (and the corresponding representation of the device) can be changed at any time by clicking and dragging on any of the corners or sides of the window.

6.5 Extended Control Options

The extended controls toolbar button displays the panel illustrated in [Figure 6-3](#). By default, the location settings will be displayed. Selecting a different category from the left-hand panel will display the corresponding group of controls:

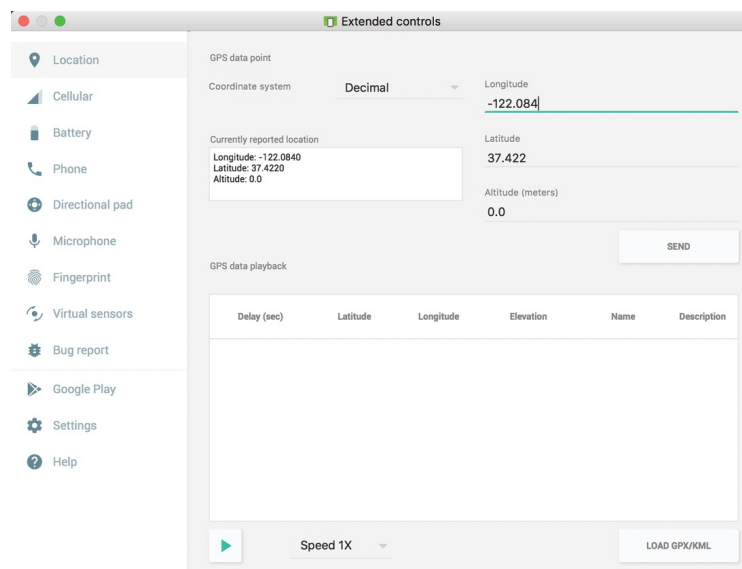


Figure 6-3

6.5.1 Location

The location controls allow simulated location information to be sent to the emulator in the form of decimal or sexagesimal coordinates. Location information can take the form of a single location, or a sequence of points representing movement of the device, the latter being provided via a file in either GPS Exchange (GPX) or Keyhole Markup Language (KML) format.

A single location is transmitted to the emulator when the *Send* button is clicked. The transmission of GPS data points begins once the “play” button located beneath the data table is selected. The speed at which the GPS data points are fed to the emulator can be controlled using the speed menu adjacent to the play button.

6.5.2 Cellular

The type of cellular connection being simulated can be changed within the

cellular settings screen. Options are available to simulate different network types (CSM, EDGE, HSDPA etc) in addition to a range of voice and data scenarios such as roaming and denied access.

6.5.3 Battery

A variety of battery state and charging conditions can be simulated on this panel of the extended controls screen, including battery charge level, battery health and whether the AC charger is currently connected.

6.5.4 Phone

The phone extended controls provide two very simple but useful simulations within the emulator. The first option allows for the simulation of an incoming call from a designated phone number. This can be of particular use when testing the way in which an app handles high level interrupts of this nature.

The second option allows the receipt of text messages to be simulated within the emulator session. As in the real world, these messages appear within the Message app and trigger the standard notifications within the emulator.

6.5.5 Directional Pad

A directional pad (D-Pad) is an additional set of controls either built into an Android device or connected externally (such as a game controller) that provides directional controls (left, right, up, down). The directional pad settings allow D-Pad interaction to be simulated within the emulator.

6.5.6 Microphone

The microphone settings allow the microphone to be enabled and virtual headset and microphone connections to be simulated. A button is also provided to launch the Voice Assistant on the emulator.

6.5.7 Fingerprint

Many Android devices are now supplied with built-in fingerprint detection hardware. The AVD emulator makes it possible to test fingerprint authentication without the need to test apps on a physical device containing a fingerprint sensor. Details on how to configure fingerprint testing within the emulator will be covered in detail later in this chapter.

6.5.8 Virtual Sensors

The virtual sensors option allows the accelerometer and magnetometer to be simulated to emulate the effects of the physical motion of a device such as

rotation, movement and tilting through yaw, pitch and roll settings.

6.5.9 Settings

The settings panel provides a small group of configuration options. Use this panel to choose a darker theme for the toolbar and extended controls panel, specify a file system location into which screenshots are to be saved, configure OpenGL support levels, and to configure the emulator window to appear on top of other windows on the desktop.

6.5.10 Help

The Help screen contains three sub-panels containing a list of keyboard shortcuts, links to access the emulator online documentation, file bugs and send feedback, and emulator version information.

6.6 Drag and Drop Support

An Android application is packaged into an APK file when it is built. When Android Studio built and ran the AndroidSample app created earlier in this book, for example, the application was compiled and packaged into an APK file. That APK file was then transferred to the emulator and launched.

The Android Studio emulator also supports installation of apps by dragging and dropping the corresponding APK file onto the emulator window. To experience this in action, start the emulator, open Settings and select the *Apps & notifications* option followed by the *App Info* option on the subsequent screen. Within the list of installed apps, locate and select the AndroidSample app and, in the app detail screen, uninstall the app from the emulator.

Open the file system navigation tool for your operating system (e.g. Windows Explorer for Windows or Finder for macOS) and navigate to the folder containing the AndroidSample project. Within this folder locate the *app/build/outputs/apk/debug* subfolder. This folder should contain an APK file named *app-debug.apk*. Drag this file and drop it onto the emulator window. The dialog shown in ([Figure 6-4](#)) will subsequently appear as the APK file is installed.

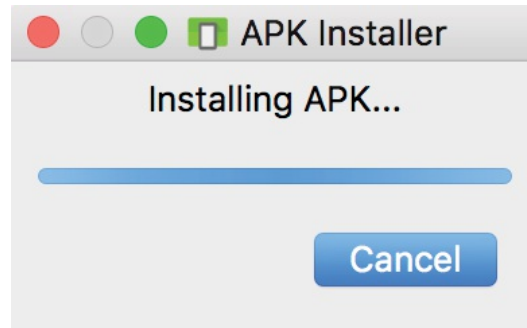


Figure 6-4

Once the APK file installation has completed, locate the app on the device and click on it to launch it.

In addition to APK files, any other type of file such as image, video or data files can be installed onto the emulator using this drag and drop feature. Such files are added to the SD card storage area of the emulator where they may subsequently be accessed from within app code.

6.7 Configuring Fingerprint Emulation

The emulator allows up to 10 simulated fingerprints to be configured and used to test fingerprint authentication within Android apps. To configure simulated fingerprints begin by launching the emulator, opening the Settings app and selecting the *Security & Location* option.

Within the Security settings screen, select the *Use fingerprint* option. On the resulting information screen click on the *Next* button to proceed to the Fingerprint setup screen. Before fingerprint security can be enabled a backup screen unlocking method (such as a PIN number) must be configured. Click on the *Fingerprint + PIN* button and, when prompted, choose not to require the PIN on device startup. Enter and confirm a suitable PIN number and complete the PIN entry process by accepting the default notifications option.

Proceed through the remaining screens until the Settings app requests a fingerprint on the sensor. At this point display the extended controls dialog, select the *Fingerprint* category in the left-hand panel and make sure that *Finger 1* is selected in the main settings panel:

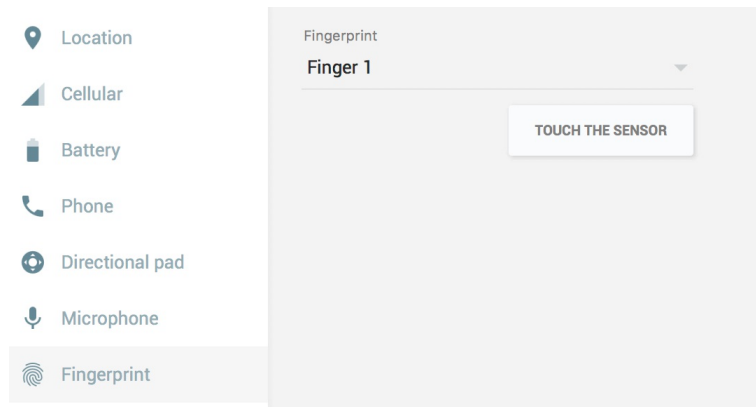


Figure 6-5

Click on the *Touch the Sensor* button to simulate Finger 1 touching the fingerprint sensor. The emulator will report the successful addition of the fingerprint:

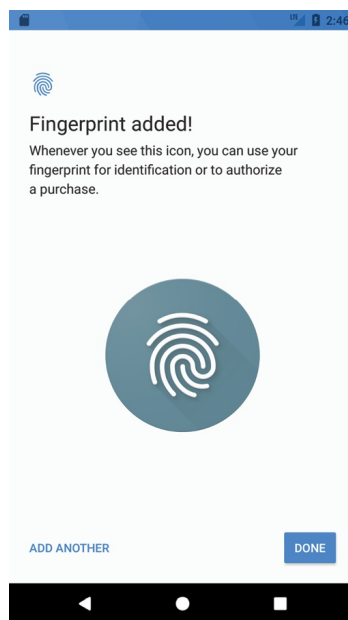


Figure 6-6

To add additional fingerprints click on the *Add Another* button and select another finger from the extended controls panel menu before clicking on the *Touch the Sensor* button once again. The topic of building fingerprint authentication into an Android app is covered in detail in the chapter entitled [“An Android Fingerprint Authentication Tutorial”](#).

6.8 Summary

Android Studio 3 contains a new and improved Android Virtual Device emulator environment designed to make it easier to test applications without

the need to run on a physical Android device. This chapter has provided a brief tour of the emulator and highlighted key features that are available to configure and customize the environment to simulate different testing conditions