UNIT 4 ANDROID DEVELOPMENT ENVIRONMENT

* 1. Introduction
  2. Objectives
  3. [Reasons for Android Development](#_bookmark0)
  4. [Android Development Platforms, Features and Tools](#_bookmark1)
     1. Check Your Progress
  5. [Configuring Android Development Environment](#_bookmark2)
     1. [Video - V5: Setting Up Android Development Environment](#_bookmark3)
     2. Check Your Progress
     3. [Video-V6: Install Android for Windows 10](#_bookmark4)
  6. Summary
  7. Further readings

# INTRODUCTION

In this unit you will be able to get familiar with available Android development platforms. You need to watch the video and install the required tools to start developing your first Android application.

# OBJECTIVES

Upon completion of this unit you should be able to:

* + - explain development platforms and distinguish each against their features and capabilities.
    - describe the background, and platform versions, system features, Android tools for the development environment.

**Outcomes**

* configure the Android development environment in a computer.



**Terminology**

**command line:** commands entered as inputs without IDE

**SDK:** Software Development Kit

**JDK:** Java Development Kit

**Android Studio:** Official Android platform to develop Android apps

# REASONS FOR ANDROID DEVELOPMENT

Today, mobile telephones have fundamentally changed the way of people interact. It is evident that mobile applications will be the future of handheld devices, Television and Automobile. Moreover, developers have started embedding Android in home appliances and other devices.

Android Development Environment

There are many reasons for the popularity of Android apps, such as:

* + - Google provides one window solution, as Play Store, to upload and download the application either free or with minimal charges. For uploading and distributing the app, developers have no need of any approval of someone.
    - Developer is the owner of his / her app and has the total control on product. However, Google has rights to unpublish any Android application in play store, if it is not complying with Google’s licenses. For instance, if application contains malicious code or violate license, Google has right to unpublish the application.
    - Android has open source operating system, open source software development kit (SDK) and good documentation.
    - Android applications are not limited to mobile devices (Phones & Tabs), but can be run on TVs, wearable devices, vehicles and even refrigerators.
    - (Source: First Thrust Towards Android, Android Programming, Course Material for Open Distance Learning, Commonwealth of Learning 2016)

# ANDROID DEVELOPMENT PLATFORMS, FEATURES AND TOOLS

In unit 2, you have learned Android architecture and major components of the Android platform. Let’s look at the Android platform and the features they are providing.

Android Studio is the official IDE for Android development, and with a single download it includes everything you need to begin developing Android apps as you can see below

* + - IntelliJ IDE + Android Studio plugin
    - Android SDK Tools
    - Android Platform-tools
    - A version of the Android platform
    - Android Emulator with an Android system image including Google Play Services

Android Studio provides tools for building apps on every type of Android device. Code editing, debugging, performance tooling, a flexible build system, and an instant build or deploy system are included in Android studio. Let's see what are the systems requirements to install Android studio in different operating systems.

###### System Requirements

System requirements for Windows, Mac OS and Linux are given below. Windows - Microsoft® Windows® 7/8/10 (32- or 64-bit)

* + - 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
    - 2 GB of available disk space minimum,
    - 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
* 1280 x 800 minimum screen resolution
* For accelerated emulator: 64-bit operating system and Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality

Mac - Mac® OS X® 10.10 (Yosemite) or higher, up to 10.12 (macOS Sierra)

Features of Android

* 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
* 2 GB of available disk space minimum,
* 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
* 1280 x 800 minimum screen resolution Linux - GNOME or KDE desktop
* Tested on Ubuntu® 12.04, Precise Pangolin (64-bit distribution capable of running 32-bit applications)
* 64-bit distribution capable of running 32-bit applications
* GNU C Library (glibc) 2.19 or later
* 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
* 2 GB of available disk space minimum,
* 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
* 1280 x 800 minimum screen resolution

For accelerated emulator: Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality, or AMD processor with support for AMD Virtualization™ (AMD-V™).

(Source: https://source.android.com/source/requirements.html, CC:BY: 2.5)

**Command Line Tools**

The Android SDK tools available from the SDK Manager provide additional command-line tools to help you during your Android development. The tools are classified into two groups: SDK tools and platform tools. SDK tools are platform independent and are required no matter which Android platform you are developing on. Platform tools are customized to support the features of the latest Android platform.

##### Additional Command Line Tools

The Android SDK tools available from the SDK Manager provide additional command-line tools to help you during your Android development. The tools are classified into two groups: SDK tools and platform tools. SDK tools are platform independent and are required no matter which Android platform you are developing on. Platform tools are customized to support the features of the latest Android platform.

Android Development Environment

##### SDK Tools

The SDK tools are installed with the SDK starter package and are periodically updated. The SDK tools are required if you are developing Android applications. The most important SDK tools include the Android SDK Manager (Android sdk), the AVD Manager (Android AVD) the emulator (emulator), and the Dalvik Debug Monitor Server (DDMS). A short summary of some frequently-used SDK tools is provided below.

**Virtual Device Tools**

###### Android Virtual Device Manager

The AVD Manager provides a graphical user interface in which you can create and manage Android Virtual Devices (AVDs) that run in the Android Emulator.

###### Android Emulator (emulator)

A QEMU(short for quick emulator) based device emulation tool that you can use to debug and test your applications in an actual Android run-time environment.

###### mksdcard

Helps you create a disk image that you can use with the emulator, to simulate the presence of an external storage card (such as an SD card).

### Development Tools

Hierarchy Viewer (hierarchyviewer) - Provides a visual representation of the layout's View hierarchy with performance information for each node in the layout, and a magnified view of the display to closely examine the pixels in your layout.

### SDK Manager

SDK Manager lets you manage SDK packages, such as installed platforms and system images.

sqlite3 - Lets you access the SQLite data files created and used by Android applications.

### Debugging Tools

The debugging tools are further explained in the later units of this material.

##### Android Monitor

Android Monitor is integrated into Android Studio and provides logcat, memory, CPU, GPU, and network monitors for app debugging and analysis.

##### adb

Android Debug Bridge (adb) is a versatile command line tool that lets you communicate with an emulator instance or connected Android-powered device. It also provides access to the device shell.

##### Dalvik Debug Monitor Server (DDMS)

DDMS lets you debug Android apps.

##### Device Monitor

Android Device Monitor is a stand-alone tool that provides a graphical user interface for several Android application debugging and analysis tools.

##### Systrace

This tool lets you analyze the execution of your application in the context of system processes, to help diagnose display and performance issues.

##### traceview

Provides a graphical viewer for execution logs saved by your application.

##### Tracer for OpenGL ES

Allows you to capture OpenGL ES([the standard for Embedded Accelerated 3D](https://www.khronos.org/opengles/) [Graphics](https://www.khronos.org/opengles/)) commands and frame-by-frame images to help you understand how your app is executing graphics commands.

### Build Tools

##### apksigner

Signs APKs and checks whether APK signatures will be verified successfully on all platform versions that a given APK supports.

###### JOBB

Allows you to build encrypted and unencrypted APK expansion files in Opaque Binary Blob (OBB) format.

###### ProGuard

Shrinks, optimizes, and obfuscates your code by removing unused code and renaming classes, fields, and methods with semantically obscure names.

###### zipalign

Optimizes APK files by ensuring that all uncompressed data starts with a particular alignment relative to the start of the file.

### Image Tools

###### Draw 9-patch

Allows you to easily create a [NinePatch](https://developer.android.com/reference/android/graphics/NinePatch.html) (class permits drawing a bitmap in nine or more sections) graphic using a WYSIWYG (What You See Is What You Get) editor. It also previews stretched versions of the image, and highlights the area in which content is allowed.

###### Etc1tool

A command line utility that lets you encode PNG images to the ETC1 compression standard and decode ETC1 compressed images back to PNG.

###### Platform Tools

The platform tools are typically updated every time you install a new SDK platform. Each update of the platform tools is backward compatible with older platforms.

Usually, you directly use only one of the platform tools—the Android Debug Bridge (adb). Android Debug Bridge is a versatile tool that lets you manage the state of an

Features of Android

Android Development Environment

emulator instance or Android-powered device. You can also use it to install an Android application (APK) file on a device.

The other platform tools, such as aidl, aapt, dexdump, and dx, are typically called by the Android build tools, so you rarely need to invoke these tools directly. As a general rule, you should rely on the build tools to call them as needed.

Note: The Android SDK provides additional shell tools that can be accessed through adb, such as bmgr and logcat.

###### a ) bmgr

A shell tool you can use to interact with the Backup Manager on Android devices supporting API Level 8 or greater.

###### b) logcat

Provides a mechanism for collecting and viewing system debug output.

(Source: <https://developer.android.com/studio/command-line/index.html>CC:BY: 2.5)

Now, you have and learnt the systems requirements (hardware/software features)to set up the Android development platform and the Android command line tools. It is vital to determine the specific features of each Android version and how it has been developed to performing better. In unit 1 we learnt the history of Android and how each version of Android evolved. Next, we will summarize the Android platform versions with their unique features.

### Android platform versions and specific features

There are rapid developments and new versions for the Android and Table 4.1 summarize the specific features of different Android platform versions up to now.

###### Table 4.1 summary the specific features of different Android platform versions

|  |  |
| --- | --- |
| **Android Platform version** | **Specific Features** |
| Android 1.6 - Donut | Quick search box Screen size diversity  Android market |
| Android 2.1 - Eclair | Google maps navigation Home screen customization  Speech-to- Text |
| Android 2.2- Froyo | Voice actions Portable Hotspot  Performance |
| Android 2.3- Gingerbread | Gaming APIs |

Features of Android

|  |  |
| --- | --- |
|  | Near Field Communication (NFC)  Battery Management |
| Android 3.0- Honeycomb | Tablet-friendly design System bar  Quick settings |
| Android 4.0- Ice cream Sandwich | Custom home screen Data usage control  Android Beam |
| Android 4.1- Jelly Bean | Google now  Actionable Notifications Account switching |
| Android 4.4- Kitkat | Voice: OK Google Immersive Design  Smart Dialer |
| Android 5.0- Lollipop | Material Design Multiscreen  Notifications |
| Android 6.0- Marshmallow | Now on tap Permissions  Battery works smarter |
| Android 7.0 - Nougat | Multi Locale language settings Multi-window view  Quick switch between apps Data Saver  Notification Controls Display Size |

You can read more information online about the relative numbers of devices that are running different versions in the following link.

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

Android Development Environment

Thus, creating apps in Android for various mobile devices are increasing day by day. Since developing native apps is expensive, the demand for cross platform app development tools is also increasing. It is essential to know cross platforms and tools for mobile application development to develop apps for enhancing the market capacity.

You can watch the online presentation given in the link below to get familiar with the cross platforms for mobile application development.

[bit.ly/XPlatformMobileDev](http://bit.ly/XPlatformMobileDev)

### 4.3.1 Check Your Progress

###### Explore most popular cross platforms and name three major cross platforms.

**Briefly describe one of the major cross platform with the features and uses of them.**

Hint: Check your answers with Answer guide at the end of this unit.

Now you are aware of the native and cross platform for mobile application development. We will now explore how the development environment is set up and configured.

# CONFIGURING ANDROID DEVELOPMENT ENVIRONMENT

To develop an Android application, first you have to setup the Android development environment.

First download and install the Android Studio, Java and then download and install every individual tool (like Java, SDK Manger, DDMS tool, AVD Manager, etc.)

After installing Android tooling, it is possible to integrate development with various IDEs like, IntelliJ IDEA, Eclipse variants etc. Otherwise you can use any other java IDE or editor tool like notepad to compose the code.

#### Video -V5: Setting Up Android Development Environment

URL: [**https://tinyurl.com/y9kcehov**](https://tinyurl.com/y9kcehov)

Let us now watch this video of setting up Android development environment set-by-step approach. This will help you at set up your Android development environment before you start programming with Android.

Next, we will be using Android Studio to develop the application. Android Studio is a native Android IDE that is fully dedicated to Android development.ADT Plugin needs to be installed to make it ready for Android development. Eclipse is an open source Java IDE that is compatible for several platforms. In early days Eclipse was the mostly used IDE by developers for Android applications.

**Download and install Java Development Kit (JDK)**

Most of the programming of Android is done using the Java programming language. To download latest Java Development Kit, follow the link given below.

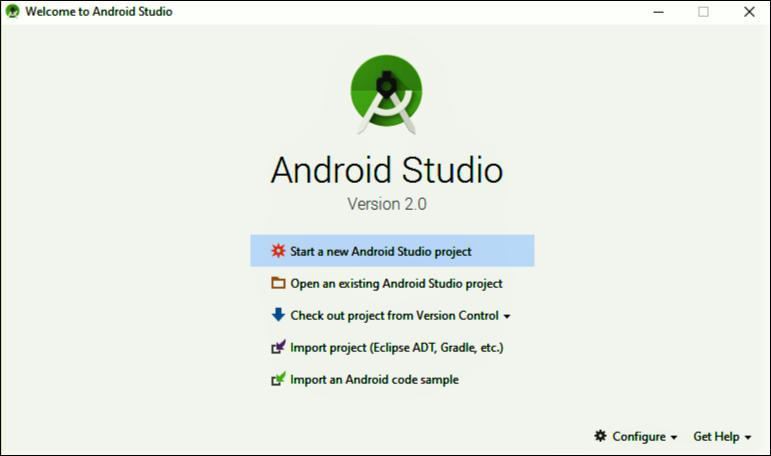
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Features of Android

Now, extract the downloaded zip file and double click on the .exe and follow the instructions.

## Download and install Android studio

You can download the latest Android Studio from the following link: <http://developer.android.com/sdk/index.html>

Download the executable file from the above mentioned link, double click on the file

and follow the installation instructions. To install the latest “Android Studio” you need to install compatible Java SE Development Kit first.

Online reading: You can pursue the following link to grab the installation instructions

for Android Studio.

<http://developer.android.com/sdk/installing/index.html>

After installing and when you will start the Android Studio first time, the very first window will look like Figure 4.1.

**Figure 4.1: First time opening window of Android Studio**

When you click on the first option to start a new Android Studio Project, multiple windows will pop up one after another to setup a new project.

Online reading: Android Studio has numerous features. In this course, you are going to explore and use many features of Android Studio. For the prior reading to understand the Android studio, follow the following web link:

<http://developer.android.com/tools/studio/index.html>

Once you install Android Studio, it is easy to keep the Android Studio IDE and Android SDK tools up to date with automatic updates and the Android SDK Manager.

Following web page provides update instructions of IDE and Android SDK tools.

Android Development Environment

<https://developer.android.com/studio/intro/update.html#channels>

### Check Your Progress

 State the most common tools used in Android application development

#### Video-V6: Install Android for Windows 10

URL[:https://storage.googleapis.com/androiddevelopers/videos/stu](https://storage.googleapis.com/androiddevelopers/videos/studio-install-windows.mp4) [dio-install-windows.mp4](https://storage.googleapis.com/androiddevelopers/videos/studio-install-windows.mp4)

This video will show you how to install and configure the Android development environment using Android Studio in Windows 10.

Furthermore, the following link has videos that shows each step of the recommended setup procedure for Mac and Linux.

<https://developer.android.com/studio/install.html>

###### Lab Exercise:

Download and install JDK and the latest Android Studio version to configure the development environment.

Now, you are ready to go for making an Android App.

# SUMMARY

The first step to start on Android based application development is to set up the development environment. Therefore, it is essential to know how to configure development environment and installing tools for Android application development.

In this unit, you have learned Android development platforms, tools and cross platforms. At the last section of the unit, you learnt about setting up the Android development environment by installing the latest JDK and Android Studio. It is also important to keep the Android studio IDE and Android SDK tools up to date at the development environment.

# FURTHER READINGS

<https://developer.android.com/studio/intro/studio-config?authuser=1>

<https://developer.android.com/studio/install>