The No Nonsense Android Guide

by Edward Pie

About This Book

It's not my fault that I wanted to and have written this book, The No Nonsense Android Guide".

It isn't uncommon to find millions of books out there that make it hard to speed up the learning process by bothering you with unnecessary technicalities and plenty talks. I've had my fair share of this nonsense where you urgently need some bit of knowledge to get some work done just to sit and read long chains of nonsense before finding one thing that make sense. After finding this one line that makes sense, you now have to figure out how to apply it to the problem at hand.
As developers we are faced with two major tasks; working late into the nights to beat deadlines and also having to learn constantly to keep up with fast growing technologies. You don't learn for a day, and your skillset becomes rusty.

If you liked this book because of its title then I guess you are a brother, a brother who has no time to read nonsense.

About The Author

Edward Pie is a self-motivated developer with experience across various field and technologies. He is an expert Java developer with many works to his credit. He is a naturally gifted teacher and an occasional speaker. And O, he has an entrepreneurial spirit. He likes to build mobile apps for Android, iOS and our old and dying buddy, J2ME. He hates only a thing, NONSENSE!
Email : hackstockpie@gmail.com
Mobile : +233200662782

With Thanks To

A special heart-felt thanks to all the authors who worked had to make my learning curve hell for me and ignited in me a no-nonsense spirit to write this book. To Mr. & Mrs. Pie, I express my sincere gratitude for all the ordeal they've been through because of my up bring and education.

A million thanks to Mr. Ben Nortey of the Metro Institute Of Technology for given me the opportunity to teach Android Application Development in over 7 universities in Ghana. I thank anyone who contributed in one way or the other to my making.

Finally, thanks to my ever-faithful girlfriend, Cyberpunk. You've always been an awesome MacBook Pro.

Introduction

Android is an operating system based on the Linux Kernel, and designed primarily for touchscreen mobile devices such as smartphones and tablet computers.

Initially developed by Android Inc., which Google backed financially and later bought in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance – a consortium of hardware, software and telecommunication companies devoted to enhancing open standards for mobile devices.

The user interface of Android is based on direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching and reverse pinching to manipulate on-screen objects. Internal hardware such as accelerometers, gyroscopes and proximity sensors are used by some applications to respond to additional user actions, for example adjusting the screen from portrait to landscape depending on how the device is oriented. Android allows users to customize their home screens with shortcuts to applications and widgets, which allow users to display live content, such as emails and weather information, directly on the home screen. Applications can further send notifications to the user to inform them of relevant information, such as new emails and text messages.

Google under the Apache License releases android's source code; this permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Most Android devices ship with a combination of open source and proprietary software. As of July 2013, Android has the largest number of applications ("apps"), available for download in Google Play store which has had over 1 million apps published, and over 50 billion downloads.[16] A developer survey conducted in April–May 2013 found that Android is the most used platform among developers: it is used by 71% of the mobile developers population.

... I hope you got this book to learn Android Apps Development, not history

We hate nonsense right? Let's get dirty

Getting Started

To start developing Android applications you need to have installed the Java Development Kit (JDK), Android Development Tools (ADT) and optionally an IDE such as Eclipse, Netbeans or IntelliJ IDEA. Installing an IDE is optional because you can develop Android apps with just the ADT, a build tool such as Apache Ant or Maven and just any basic text editor such as Sublime Text or Notepad++.

Though it's optional to develop Android applications using an IDE, it is highly beneficial to use a good IDE. An IDE, an abbreviation for Integrated Development Environment, is a suite of tools, which work collectively to make a programmers life easier. IDEs help with syntax highlighting, code completion, debuggers, code snippet generators and other features which will otherwise be difficult to have when you choose to use regular text editors and a build tool.
Because we hate to read crap and waste time as well, I suggest you use an IDE. In fact I'll be using an IDE for all demonstrations in this book.

Assumptions

Having picked up this book to learn Android applications development, I assume you already know how to program in not just any programming language but Java. Unlike other authors, I wont lie to you by telling you it's easy to just jump into the world of Android applications development simply because you know how to program in C, C++, C# or any other Java-like programming language.
Java is the numero uno programming language used for the development of Android applications. To this effect, I'll suggest that you take a crush course in Java if you don't know how to use the language already. You can find millions of Java programming tutorials on the Internet but I suggest you check out tutorialspoint.com for awesome crush course in Java.

Having made the above assumption, I wont show you how to install the Java Development Kit on your computer. Being a Java developer already I assume you know how to setup your development environments. After all, no one writes Java programs without the JDK unless you are a certified voodoo programmer.

Being a hater of nonsense too, I assume you mean business. I assume you are serious about this whole Android applications development thingy and will do anything right, whether easy or difficult for as long as it take to learn, understand and apply whatever you learn in this book. If you lack the passion to persist for

as long as it takes to understand anything this book teaches you, you are suffering from a disease called WANNABESITIS. A wanna-be is one who imitates the behavior, customs, or dress of an admired person or group but never wants to go through what these role models went through to become what they are today.

This is a book for DOERS not WANNABES…

Environment Setup - Theory

As of this writing, there are trillions of IDEs for developing Android Applications. This makes it hard for starters to know exactly which of these is the industry standard. And O, by trillions I mean 5 IDEs namely, Eclipse, IntelliJ IDEA, NetBeans, ADT Bundle and Android Studio. So you see how confusing it can be to get started? If you aren't confused yet then you aren't serious enough to read this book. Thinking 5 different IDEs just for Android development isn't a mess makes you a typical textbook example of a non-serious person. Get confused now!

Eclipse, IntelliJ IDEA, NetBeans, ADT Bundle, Android Studio, *Rick Titball, PantherKing, SackRider, Thrasher, DeadStick,* Blah! Blah! Blah! All this is an attempt to make you confused and if you still aren't confused, STOP READING. You are too serious!

So now, which is which?

NetBeans is a great IDE but it certainly is not good for Android development.

Eclipse is a highly extensible IDE, which can be used to build not just Java or Android apps but any other technology for which an eclipse plugin exists.

ADT Bundle is an intelligent combination of the Eclipse IDE and the Android Development Tools from Google. Did I say "Intelligent combination"? O yeah, you couldn't expect less since it came from Google right?

IntelliJ IDEA is an awesome IDE too. There are free and commercial versions of IntelliJ IDEA. Like Eclipse, it is also highly extensible via plugins. This means you can develop applications using different programming languages and technologies with IntelliJ. It has an Android plugin for building, debugging and deploying to the Play market.

Android Studio is a promise from Google to provide a mind-blowing IDE for Android development. It has received some attention and I admit it will be the best when it's completely done but for now it's quite buggy. You can try it out though because we will all eventually move to Android Studio in the near future when it is completely done. And yeah, Android Studio is based on the community edition of IntelliJ IDEA.

These combinations of technologies has given us the following 2 simultaneous equations to solve:

ADT Bundle = Eclipse + Android Development Tools
Android Studio = IntelliJ IDEA + Android Development Tools

In this book, we will use the **ADT Bundle** but we'll keep our eyes on the progress of Android Studio when it's cooked well enough. Don't worry, the transition wont hurt. It's a promise.

Environment Setup – Practical

Having installed and properly added the Java Development Kit to your path, it's now time to setup the ADT Bundle. As I said previously, the ADT Bundle is a combination of Eclipse and the Android Development Toolkit. It's an attempt to help you get started as soon as possible. Initially, we had to manually download and setup the Eclipse IDE, download the Android SDK and also install the ADT plugin in Eclipse. Not too many people had success with this approach so Google bundled it up for us and perfectly called it ADT Bundle. It comes with the Eclipse IDE, the Android Development/Developer Tools/Toolkit, and the latest Android SDK properly configured in-house by Google all for your coding pleasure. ADT Bundle is a life saver.

A simple Google search on "ADT Bundle" should land you on the download page. There are 32-bit and 64-bit versions so be mindful to download the right setup per your computer's architecture.

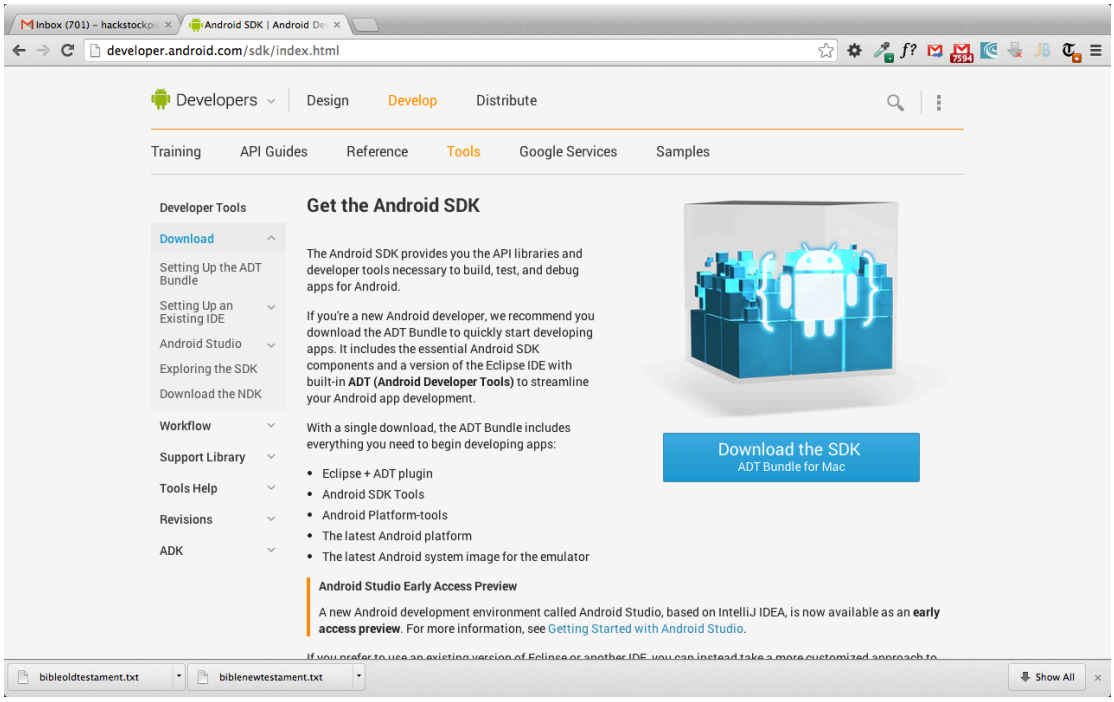

Figure 1.0 : ADT Bundle download page.

The download page will automatically determine your computer's operating system and suggest the right package for you to download.

The package is a zipped file so all you need to do is to unzip it with compression programs such as WinZip, WinRAR or the Unix tar command for black screen gurus.

After unzipping you should have a directory structure similar to the one shown below

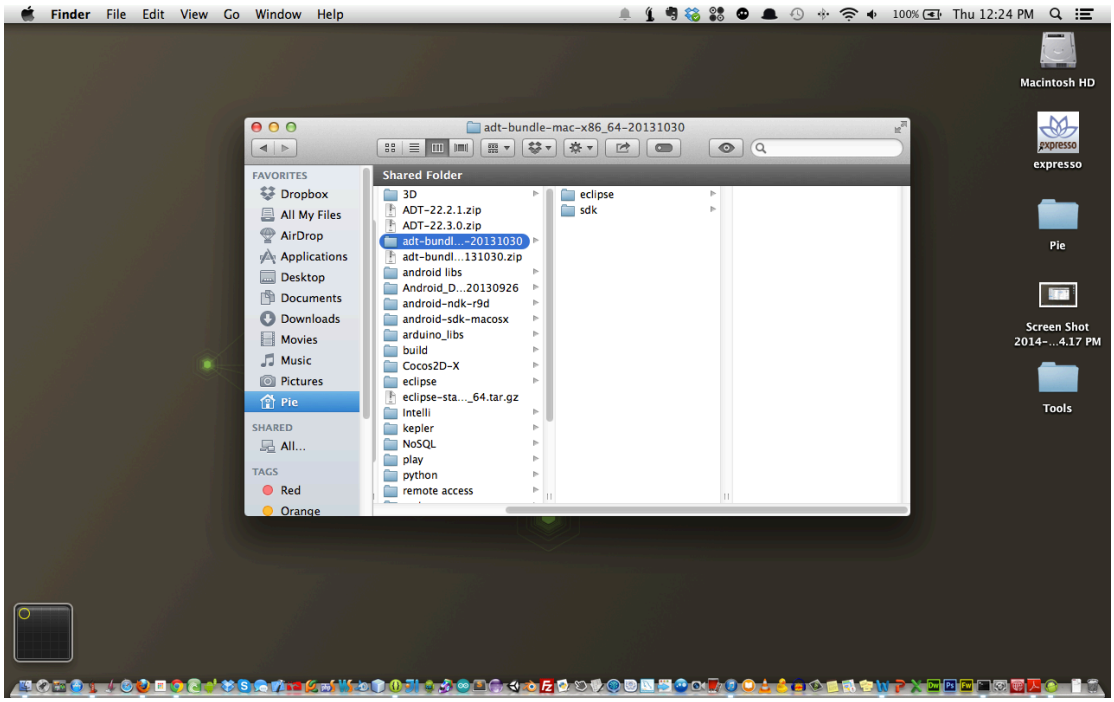

Figure 1.1 : ADT Bundle unzipped