

Extracted from:

Developing Android on Android

Automate Your Device with Scripts and Tasks

This PDF file contains pages extracted from *Developing Android on Android*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

Copyright © 2013 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina

Developing Android on Android

Automate Your Device
with Scripts and Tasks



Mike Riley

edited by Jacquelyn Carter

Developing Android on Android

Automate Your Device with Scripts and Tasks

Mike Riley

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic courses, workshops, and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at <http://pragprog.com>.

The Android robot is reproduced or modified from work created and shared by Google and used according to terms described in the Creative Commons 3.0 Attribution License.

The team that produced this book includes:

Jacquelyn Carter (editor)
Potomac Indexing, LLC (indexer)
Molly McBeath (copyeditor)
David J Kelly (typesetter)
Janet Furlow (producer)
Juliet Benda (rights)
Ellie Callahan (support)

Copyright © 2013 The Pragmatic Programmers, LLC.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.
ISBN-13: 978-1-937785-54-3
Encoded using the finest acid-free high-entropy binary digits.
Book version: P1.0—November 2013

*This book is dedicated to my three favorite
M's: Marinette, Marielle, and Mitchell.*

Introduction

In this book, we're going on a journey of discovery. We're going to discover how amazing the Android OS is and how it is transforming the way people communicate. We're also going to learn how to leave our legacy desktop PCs behind, even for native Android application development needs.

The idea for this book was the result of a conversation I had with Pragmatic Bookshelf publisher Dave Thomas. He had just acquired a Galaxy S3 Android phone and wanted to know what kind of cool things he could do with it. Since I have been an Android user since the release of the first commercial Android phone, the G1, I had a few suggestions on where to start. As he became more enthusiastic about the broad possibilities of customization and personalization that the Android platform has to offer, a new book on the subject started to crystallize.

The objectives of this book are simple. You will learn about how to apply and codify your mobile automation needs in an Android program. Using both scripting and native application development approaches, we will build several programs that not only teach you how to quickly automate your mobile lifestyle but also give you the skills to extend these programs beyond their tutorial roots.

Why Android? Why Now?

The Android OS is several years old, and its design principles (a modern, true multitasking mobile OS with built-in memory, permissions management, and so on) have been the same since its inception. So why is this book relevant now compared to five years ago, when Android was first introduced?

Obviously, the platform has matured considerably in that time. It has also greatly benefited from its open source approach by fostering significantly faster innovation compared to closed, proprietary operating systems. Take a look at a first-generation iPhone compared to the iPhone 5. While the hardware has vastly improved, the primary user interface is nearly identical. Consider

the differences between Android 1.0 running on a G1 phone and Android 4.2 running on a Nexus 4. The differences are striking. The user interface, hardware support, design aesthetics, and everything but the original design principles have rapidly evolved for the better. One benefit from this co-evolution of hardware and software is that you can do things on a modern Android device that was the stuff of fiction five years ago. To think that on your Android device you can now do computing on a scale that was the exclusive domain of desktop PCs for the last thirty years is awe-inspiring.

This evolutionary path is also manifested in Android application development tools. Once clunky and incomplete, the Android SDK and Eclipse plug-ins are finally capable of stable, team-based, test-driven development. While the user interface construction toolkit could still use more polish, every other aspect of the typical Android development and emulation on a desktop PC is polished and professional.

One of the most exciting aspects of Android programming, testing, and deployment is that its application development life cycle can now be done directly on the Android device. This is a big deal. When compared to other mobile operating systems that require an expensive PC outfitted with a decent processor and plenty of RAM to run the target emulator, the projects discussed in this book require only your Android device. When you code and run applications on the same device, it greatly accelerates the development process, just as it did during the desktop PC era.

Let's also not forget that, like a desktop computer, Android's home screen can be highly customized and extended via custom wallpapers, animations, icons, folder actions, transition animations, and much more. This degree of personalization allows you to make your Android device fit your aesthetic values, daily workflow, and communication and notification preferences, not the other way around. Third-party extensions and widgets also help push the envelope of what is possible, further contributing to Android's success and dominant market position.

Who This Book Is For

This book is for anyone who is interested in doing much more with an Android device than downloading and using apps from the Google Play store. If you love your Android phone or tablet and you love to tinker with technology, this is the perfect book for you. And while prior programming experience is not required, it will be helpful to understand some of the scripts that we will create in the chapters ahead.

Requirements

This is a book about Android, so it should come as no surprise that a must-have requirement is an Android phone or tablet running Android OS 4.2 (known by its friendly code name Jelly Bean) or newer. The screenshots used throughout this book were taken on a Galaxy Nexus phone and Nexus 7 tablet.

In addition to the Android phone or tablet, you should have an active account on the Google Play store, since a good portion of the software used in this book is exclusively distributed via the Google Play service.

Lastly, while it's not required, I strongly recommend you obtain a quality Bluetooth keyboard known to be compatible with the Android OS. I have yet to use a Bluetooth keyboard that could not be paired with Android, but keyboards designed specifically with Android in mind are optimal since they often have special keys associated with functions such as play/pause music, volume control, toggle between applications, lock the screen, and so on. My current favorite mobile Bluetooth keyboard is the Logitech Tablet Keyboard for Win8/RT and Android, shown in the following figure.¹ It is a full-size keyboard and thus larger than other mobile Bluetooth keyboards that have a smaller footprint or fold in half for greater portability. Plus, Logitech's full-size keyboard combined with the protective cover doubles as a phone or tablet stand.



Figure 1—The Logitech Tablet Keyboard for Win8/RT and Android

1. <http://www.logitech.com/en-us/tablet-accessories/android/tablet-keyboard-android-win8-rt>

While you are understandably not going to be able to use this keyboard in a cramped moving vehicle such as a bus or train, it works perfectly fine sitting on an airplane fold-down seat tray or desk. And since I'm usually bringing along a backpack during my commutes, the Logitech keyboard adds practically no additional weight or bulk to the bag. Besides, you will find that the keyboard is a sanity saver when editing code or documents on the Android device.

So, that's it—a phone or tablet, an active Google Play account, and maybe an Android-compatible Bluetooth keyboard. For folks like me who have been around since the dawn of the personal computer era, it is simply amazing to think how far we have come in the past forty years and how much further we'll go in the next forty years.

Jailbreaking and Rooting

Unlike traditional desktop operating systems, mobile OSs like Android and iOS are locked down in such a way so that the system-level files cannot be tampered with by ordinary users. This keeps the device more secure by preventing malicious applications from modifying system files without the user's knowledge. Similarly, it prevents the user from altering these files.

Jailbreaking is a term used in the mobile device market to mean a procedure that allows users to bypass the normal operating system-level restrictions, typically to gain root-level access (*rooting* the device). Once root access is obtained, the user or application has full read-write access to all aspects of the operating system files. This allows modification of the device's behavior in ways that weren't originally intended by the OS developer.

While Android OS tinkerers can benefit from rooting a device by understanding the internal workings of the OS better, average users could be putting the contents and operation of their phones' security at risk if they are untrained in the various aspects of mobile OS security best practices. In the early days of Android, when many features were immature or missing, jailbreaking and rooting were more attractive, since doing so provided power users with a degree of customization that matched their needs. These custom modifications could range from modifying system-level virtual private network (VPN) software stacks to changing the look and feel of the home screen.

Android today is a much more mature operating system, so many of these limitations have a viable and more secure alternative. The projects in this book do not require jailbreaking or rooting your device. Unless you are a security researcher or a technology tinkerer who likes to crack things open

to see how they work, there is no overwhelming reason for average users to consider jailbreaking and rooting their Android devices.

What's in This Book

Now that we have packed our bags for the journey, let's look at the road map we'll use to progress along the trail.

In the first part of the book, we will look at the variety of options we have to customize our Android experience. The home screen, lock screen, widgets, backgrounds, icons, and touch behaviors can all be personalized to your liking. Unlike some mobile operating systems that enforce a structured, inflexible design aesthetic, Android offers desktop-like customization in a mobile package. We will dive into specific examples for home screen renovation. We will also take a look at extending our Android experience by calling upon a variety of Android's hardware capabilities, such as using the headphone jack to transform our Android applications into better, more convenient, and information-rich wearable-computing user experiences.

In Part II, we will dip our toe in the automation waters by taking a look at a very powerful application called Tasker. We will use Tasker to automate several personal workflow needs and get introduced to some basic conditional programming and control flow while we're at it. We will also delve further into the programming landscape with the introduction of Scripting Layer for Android (SL4A). SL4A will allow us to write scripts in Python, Ruby, and other popular interpreted languages that will execute on Android and give us access to most of the system-level calls exposed by the Android SDK. We will conclude the Explore section of the book by actually programming Android using the native SDK. But instead of using a personal computer loaded with the Android SDK, emulator, and related development tools, we will write, compile, test, and deploy these native applications entirely on our Android phone or tablet.

In Part III of the book, we will apply what we learned in the first two parts by first creating a custom Android widget entirely on the device—no PC required. The final set of projects in the book will wrap these scripting and automation technologies in friendly user interfaces. These projects will show the versatility and automation opportunities that Android has to offer. The book also includes appendixes that review a variety of programming tools that run on the Android platform, as well as offer additional web resources to further your own project ideas.

By the end of the book, you should be well prepared to continue the journey on your own to create an Android experience that perfectly complements your information-interaction lifestyle.

Online Help

Many websites are devoted to the dissemination of Android news, reviews, hacking, modding, and programming. Check out [Appendix 1, *Android Programming Tools*, on page ?](#), for a list of some of my favorites. It should go without saying that for Android development, Google's <http://developer.android.com> website offers the official word on Android application development. This isn't just a repository of bland technical documentation but a wealth of useful and well-written articles, tutorials, and tech notes from the folks responsible for various portions of the operating system. It's a resource that any serious Android developer should have permanently bookmarked.

There are a number of footnotes in the book featuring web links to more online resources. I also encourage you to post specific questions or comments about the ideas presented in the book at the book's web forum. Should you happen to spot an error, feel free to mention it on the book's website errata page. You're also welcome to contact me directly via my mike@mikeriley.com email address or follow me on Twitter @mriley. I look forward to hearing from you!

With that, we're ready to take a look at all the things we can customize in a nonrooted device running the stock Android 4.2 or newer operating system.

Mike Riley

mike@mikeriley.com

November 2013