

Beginning Android Games

Third Edition



Mario Zechner
J. F. DiMarzio
Robert Green

Apress®

Beginning Android Games

Mario Zechner
Graz, Steiermark, Austria

J. F. DiMarzio
Celebration, Florida, USA

Robert Green
Portland, Oregon, USA

ISBN-13 (pbk): 978-1-4842-0473-3
DOI 10.1007/978-1-4842-0472-6

ISBN-13 (electronic): 978-1-4842-0472-6

Library of Congress Control Number: 2016961227

Copyright © 2016 by Mario Zechner, J. F. DiMarzio, and Robert Green

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director: Welmoed Spahr

Lead Editor: Steve Anglin

Technical Reviewer: Paul Trebilcox-Ruiz

Editorial Board: Steve Anglin, Pramila Balan, Laura Berendson, Aaron Black, Louise Corrigan,

Jonathan Gennick, Robert Hutchinson, Celestin Suresh John, Nikhil Karkal, James Markham,

Susan McDermott, Matthew Moodie, Natalie Pao, Gwenan Spearing

Coordinating Editor: Mark Powers

Copy Editor: April Rondean

Compositor: SPi Global

Indexer: SPi Global

Artist: SPi Global

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc. (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary materials referenced by the author in this text are available to readers at www.apress.com. For detailed information about how to locate your book's source code, go to www.apress.com/source-code/. Readers can also access source code at SpringerLink in the Supplementary Material section for each chapter.

Printed on acid-free paper

Dedicated to my idols, Mom and Dad, and my love, Stefanie.

—Mario Zechner

*Dedicated to my children and everyone involved in helping make
this book happen. Go Sox.*

—J.F. DiMarzio

*Dedicated to my family and everyone who has helped us
along the way.*

—Robert Green

Contents at a Glance

About the Authors	xix
About the Technical Reviewer	xxi
Acknowledgments	xxiii
Introduction	xxv
■ Chapter 1: An Android in Every Home	1
■ Chapter 2: First Steps with Android Studio	15
■ Chapter 3: Game Development 101	33
■ Chapter 4: Android for Game Developers	81
■ Chapter 5: An Android Game Development Framework	159
■ Chapter 6: Mr. Nom Invades Android	199
■ Chapter 7: OpenGL ES: A Gentle Introduction	237
■ Chapter 8: 2D Game Programming Tricks	309
■ Chapter 9: Super Jumper: A 2D OpenGL ES Game	379
■ Chapter 10: OpenGL ES: Going 3D	435
■ Chapter 11: 3D Programming Tricks	471
■ Chapter 12: Android Invaders: The Grand Finale	521
■ Chapter 13: Going Native with the NDK	567
■ Chapter 14: Marketing and Monetizing	579
■ Chapter 15: Publishing Your Game	587
Index	595

Contents

About the Authors	xix
About the Technical Reviewer	xxi
Acknowledgments	xxiii
Introduction	xxv
■ Chapter 1: An Android in Every Home	1
A Brief History of Android	1
Fragmentation	3
The Role of Google	3
The Android Open Source Project.....	3
Google Play.....	4
Google I/O	4
Android's Features and Architecture	4
The Kernel	5
The Runtime and Dalvik.....	6
System Libraries.....	6
The Application Framework.....	7
The Software Development Kit.....	7
The Developer Community	8
Devices, Devices, Devices!.....	9
Hardware	9
The Range of Devices.....	10

Compatibility Across All Devices	11
Mobile Gaming Is Different.....	12
A Gaming Machine in Every Pocket.....	12
Always Connected	13
Casual and Hardcore	13
Big Market, Small Developers	14
Summary.....	14
■ Chapter 2: First Steps with Android Studio	15
Setting Up the Development Environment	15
Setting Up the JDK.....	15
Installing Android Studio.....	16
Setting Up the Android SDK.....	16
A Quick Tour of Android Studio	18
Creating a New Project in Android Studio and Writing Your Code	20
Creating the Project.....	20
Exploring the Project	21
Writing the Application Code	23
Running the Application on a Device or Emulator	25
Connecting a Device.....	25
Creating an Android Virtual Device	25
Installing Advanced Emulator Features	27
Running an Application.....	28
Debugging and Profiling an Application	30
LogCat.....	31
Summary.....	32
■ Chapter 3: Game Development 101	33
Genres: To Each One’s Taste.....	33
Casual Games.....	34
Puzzle Games	35

Action and Arcade Games.....	37
Tower-Defense Games.....	40
Social Games.....	41
Beyond the Genres	43
Game Design: The Pen Is Mightier Than the Code.....	44
Core Game Mechanics.....	44
A Story and an Art Style.....	47
Screens and Transitions	48
Code: The Nitty-Gritty Details	54
Application and Window Management	54
Input	55
File I/O	58
Audio.....	59
Graphics.....	62
The Game Framework.....	73
The Game and Screen Interfaces	75
A Simple Example	77
Frame Rate–Independent Movement.....	79
Summary.....	80
■ Chapter 4: Android for Game Developers	81
Defining an Android Application: The Manifest File	81
The <manifest> Element.....	82
The <application> Element.....	83
The <activity> Element	84
The <uses-permission> Element.....	85
The <uses-feature> Element	87
Game Project Setup in Six Easy Steps.....	88
Google Play Filters.....	89
Defining the Icon of Your Game	90

Android API Basics	91
Creating a Test Project.....	91
The Activity Life Cycle.....	95
Input Device Handling.....	102
File Handling.....	118
Audio Programming.....	127
Playing Sound Effects.....	127
Streaming Music	130
Basic Graphics Programming	133
Best Practices	156
Summary.....	157
■ Chapter 5: An Android Game Development Framework	159
Plan of Attack	159
The AndroidFileIO Class.....	159
AndroidAudio, AndroidSound, and AndroidMusic: Crash, Bang, Boom!	161
AndroidInput and AccelerometerHandler	166
AccelerometerHandler: Which Side Is Up?	166
CompassHandler	167
The Pool Class: Because Reuse Is Good for You!.....	169
KeyboardHandler: Up, Up, Down, Down, Left, Right	171
Touch Handlers.....	174
AndroidInput: The Great Coordinator	181
AndroidGraphics and AndroidPixmap: Double Rainbow	182
Handling Different Screen Sizes and Resolutions	183
AndroidPixmap: Pixels for the People.....	186
AndroidGraphics: Serving Our Drawing Needs	187
AndroidFastRenderView: Loop, Stretch, Loop, Stretch.....	191
AndroidGame: Tying Everything Together	193
Summary.....	197

■ Chapter 6: Mr. Nom Invades Android	199
Creating the Assets	199
Setting Up the Project	201
MrNomGame: The Main Activity.....	202
Assets: A Convenient Asset Store	202
Settings: Keeping Track of User Choices and High Scores	203
LoadingScreen: Fetching the Assets from Disk	205
The Main Menu Screen	207
The HelpScreen Classes.....	210
The High-Scores Screen.....	212
Rendering Numbers: An Excursion	213
Implementing the Screen	214
Abstracting the World of Mr. Nom: Model, View, Controller	217
The Stain Class.....	219
The Snake and SnakePart Classes	219
The World Class	223
The GameScreen Class	227
Summary.....	235
■ Chapter 7: OpenGL ES: A Gentle Introduction	237
What Is OpenGL ES and Why Should I Care?	237
The Programming Model: An Analogy.....	238
Projections.....	240
Normalized Device Space and the Viewport.....	242
Matrices.....	242
The Rendering Pipeline.....	243
Before We Begin	244
GLSurfaceView: Making Things Easy Since 2008.....	245
GLGame: Implementing the Game Interface.....	247

Look Mom, I Got a Red Triangle!	254
Defining the Viewport	254
Defining the Projection Matrix	255
Specifying Triangles.....	258
Putting It Together	261
Specifying Per-Vertex Color	265
Texture Mapping: Wallpapering Made Easy	268
Texture Coordinates.....	268
Uploading Bitmaps	270
Texture Filtering.....	271
Disposing of Textures	272
A Helpful Snippet.....	273
Enabling Texturing	273
Putting It Together	273
A Texture Class	276
Indexed Vertices: Because Re-use Is Good for You	278
Putting It Together	279
A Vertices Class	281
Alpha Blending: I Can See Through You	284
More Primitives: Points, Lines, Strips, and Fans	288
2D Transformations: Fun with the Model-View Matrix	289
World and Model Space.....	289
Matrices Again.....	290
An Initial Example Using Translation.....	291
More Transformations.....	296
Optimizing for Performance	301
Measuring Frame Rate	302
What's Making My OpenGL ES Rendering So Slow?	303
Removing Unnecessary State Changes	304

The Concept of Binding Vertices.....	305
In Closing.....	308
Summary.....	308
■ Chapter 8: 2D Game Programming Tricks	309
Before We Begin.....	309
In the Beginning . . . There Was the Vector.....	309
Working with Vectors.....	310
A Little Trigonometry.....	312
Implementing a Vector Class	314
A Simple Usage Example.....	317
A Little Physics in 2D.....	322
Newton and Euler: Best Friends Forever	323
Force and Mass	324
Playing Around, Theoretically	324
Playing Around, Practically	325
Collision Detection and Object Representation in 2D	330
Bounding Shapes.....	330
Constructing Bounding Shapes	331
Game Object Attributes.....	333
Broad-Phase and Narrow-Phase Collision Detection	334
An Elaborate Example.....	341
A Camera in 2D.....	353
The Camera2D Class	356
An Example.....	358
Texture Atlas: Because Sharing Is Caring.....	359
An Example.....	361
Texture Regions, Sprites, and Batches: Hiding OpenGL ES	364
The TextureRegion Class.....	365
The SpriteBatcher Class	366

Sprite Animation.....	372
The Animation Class	372
An Example.....	374
Summary.....	377
■ Chapter 9: Super Jumper: A 2D OpenGL ES Game	379
Core Game Mechanics	379
Developing a Backstory and Choosing an Art Style.....	380
Defining Screens and Transitions	380
Defining the Game World.....	382
Creating the Assets	384
The UI Elements.....	384
Handling Text with Bitmap Fonts	385
The Game Elements.....	387
Texture Atlas to the Rescue	389
Music and Sound	390
Implementing Super Jumper.....	392
The Assets Class.....	392
The Settings Class	395
The Main Activity	396
The Font Class	397
The GLScreen Class.....	399
The Main Menu Screen.....	399
The Help Screens.....	402
The High-Scores Screen.....	405
The Simulation Classes	407
The Game Screen	422
The WorldRenderer Class	429
Summary.....	433

■ Chapter 10: OpenGL ES: Going 3D	435
Before We Begin.....	435
Vertices in 3D	436
Vertices3: Storing 3D Positions.....	436
An Example.....	438
Perspective Projection: The Closer, the Bigger	442
Z-buffer: Bringing Order to Chaos	444
Fixing the Previous Example.....	445
Blending: There's Nothing Behind You	447
Z-buffer Precision and Z-fighting	449
Defining 3D Meshes	450
A Cube: Hello World in 3D	451
An Example.....	454
Matrices and Transformations, Again	456
The Matrix Stack.....	457
Hierarchical Systems with the Matrix Stack.....	459
A Simple Camera System	465
Summary.....	469
■ Chapter 11: 3D Programming Tricks	471
Before We Begin.....	471
Vectors in 3D	471
Lighting in OpenGL ES.....	476
How Lighting Works.....	476
Light Sources.....	477
Materials.....	478
How OpenGL ES Calculates Lighting: Vertex Normals	479
In Practice.....	480
Some Notes on Lighting in OpenGL ES	493

Mipmapping	493
Simple Cameras	498
The First-Person, or Euler, Camera	498
An Euler Camera Example	501
A Look-At Camera.....	507
Loading Models	509
The Wavefront OBJ Format.....	509
Implementing an OBJ Loader	510
Using the OBJ Loader	515
Some Notes on Loading Models	515
A Little Physics in 3D.....	515
Collision Detection and Object Representation in 3D	516
Bounding Shapes in 3D	516
Bounding Sphere Overlap Testing.....	517
GameObject3D and DynamicGameObject3D	519
Summary.....	520
■ Chapter 12: Android Invaders: The Grand Finale.....	521
Core Game Mechanics	521
Developing a Backstory and Choosing an Art Style.....	523
Defining Screens and Transitions	524
Defining the Game World.....	525
Creating the Assets	526
The UI Assets	526
The Game Assets	527
Sound and Music	529
Plan of Attack	529
The Assets Class	530
The Settings Class.....	533
The Main Activity	534
The Main Menu Screen	535

The Settings Screen	537
The Simulation Classes	540
The Shield Block Class	541
The Shot Class.....	541
The Ship Class	542
The Invader Class	543
The World Class	546
The GameScreen Class	552
The WorldRender Class	559
Optimizations	564
Summary.....	565
■ Chapter 13: Going Native with the NDK	567
What Is the Android NDK?	567
The Java Native Interface.....	568
Setting Up the NDK.....	569
Setting Up an NDK Android Project	570
Creating Java Native Methods.....	570
Creating the C/C++ Header and Implementation.....	571
Building the Shared Library.....	573
Putting It All Together	575
Summary.....	577
■ Chapter 14: Marketing and Monetizing	579
Monetizing Your Game.....	579
Advertising	580
Ad Providers	580
Ad Aggregation Services	580
Game Advertising Tips	580
In-App Products.....	581
Virtual Currency.....	581

To Sell or Not to Sell	582
Licensing	582
Getting Your Game Discovered	583
Social Network Integration	583
Discovery Services	583
Blogs and Conventional Web Media	584
Monetizable by Design	584
Discoverable by Design	585
Summary.....	585
■ Chapter 15: Publishing Your Game	587
A Word on Testing.....	587
Becoming a Registered Developer	588
Signing Your Game’s APK	589
Putting Your Game on Google Play	592
Publish!	594
More on the Developer Console	594
Summary.....	594
Index.....	595

About the Authors



Mario Zechner has been programming games since age 12. Over the years, he has created many prototypes and finished games on various platforms. He's the creator of libgdx, an Android game development framework used by many top-grossing Android games. Before he joined his current employer, Mario was the tech-lead at a mobile gaming startup in San Francisco. Mario's current day job involves working in the fields of information extraction and retrieval, visualization, and machine learning. You can meet Mario at conferences, where he gives talks and workshops on game programming, read his blog at www.badlogicgames.com, or follow him on Twitter @badlogicgames.

J. F. DiMarzio is a seasoned Android developer and author. He began developing games in Basic on the TRS-80 Color Computer II in 1984. Since then, he has worked in the technology departments of companies such as the U.S. Department of Defense and the Walt Disney Company. He has been developing on the Android platform since the beta release of version .03, and he has published two professional applications and one game on the Android Marketplace. DiMarzio is also an accomplished author. Over the last 15 years, he has released 12 books, including *Android: A Programmer's Guide*. His books have been translated into four languages and published worldwide. DiMarzio's writing style is very easy to read and understand, which makes the information in the topics that he presents more retainable.

■ ABOUT THE AUTHORS



Robert Green is the founder of the game studio Battery Powered Games in Portland, Oregon. He has developed over a dozen Android games, including *Deadly Chambers*, *Antigen*, *Wixel*, *Light Racer*, and *Light Racer 3D*. Before diving full time into mobile video game development and publishing, Robert worked for software companies in Minneapolis and Chicago, including IBM Interactive. Robert's current focus is cross-platform game development and high-performance mobile gaming with the BatteryTech SDK. Robert often updates his personal blog with game programming tidbits at www.rbgrn.net. You can check out his professional game development software at www.batterypoweredgames.com.

About the Technical Reviewer



Paul Trebilcox-Ruiz is a software engineer at Sphero (they make awesome robots that you can control from your phone) in Boulder, Colorado, and a computer science graduate from California State University, Fresno. His main interests lie in the Android platform, Android TV, and wearable computing. He also actively participates in hackathons in the Denver-Boulder area, writes an Android technical blog, and presents to Meetups in the area.

Acknowledgments

I'd like to thank the excellent team at Apress, who made this book possible. Specifically, I'd like to thank Candace English, Adam Heath, Matthew Moodie, Damon Larson, James Compton, Christine Ricketts, Tom Welsh, Bill McManus, and many more good souls who have worked on this book.

Thanks to Robert Green for being a great co-author and for living through the hell of early Android versions with me.

Another big thanks to my splendid team of contributors, who carried on the libgdx effort while I was busy writing this book. I'd especially like to thank my friend Nathan Sweet—the months we spent together in San Francisco and Graz were some of the most memorable times in my life; Ruben Garat and Ariel Coppes from Gemserk, who make sure we stay on target with libgdx; Justin Shapcott, who keeps cleaning up my mess in the libgdx repository; Søren Nielsen, one of the most knowledgeable artists I've ever had the pleasure to work with, who's a helping hand in the libgdx IRC channel; and, finally, all the folks on the libgdx IRC channel and the forums, whom I consider to be extended family.

Last but not least, thanks to Mom and Dad for being folks to look up to, and to my love, Stefanie, who put up with all the long nights alone in bed, as well as my grumpiness. Lyrt!

—Mario Zechner

I'd like to thank Mark, Steve, Paul, and the team at Apress for making this book happen. I'd also like to thank Stacey, Katrina, and the agents and team at Studio B for handling the hard part of the process for over 15 years. Finally, I'd like to thank Christian, Sophia, Giovanni, Jennifer, Katie, Sarah, Joseph, and the rest of my family for being patient and understanding throughout.

—J.F. DiMarzio

I'd like to thank Mario for answering so many questions and helping me work through a few tricky bugs over the years. Ryan Foss is an artist and programmer I worked with to build several games and is truly an exceptional person. Scott Lembcke of Howling Moon Software helped me through some difficult programming problems. Zach Wendt, organizer of the Minneapolis IGDA, always hung out late after meetings and talked tech with me. Thanks to Dmitri Salcedo, who's been sticking with me through a tough year and helping us to achieve more than we ever have before. Finally, I thank all of the original licensees of BatteryTech SDK, who believed in the product before anyone else. Thank you all for your support and help; I couldn't have made it without you.

—Robert Green

Introduction

Hi there, and welcome to the world of Android game development. You came here to learn about game development on Android, and we hope to be the people who enable you to realize your ideas.

Together we'll cover quite a range of materials and topics: Android basics, audio and graphics programming, a little math and physics, OpenGL ES, an intro to the Android Native Development Kit (NDK), and, finally, publishing, marketing, and making money from your game. Based on all this knowledge, we'll develop three different games, one of which is even 3D.

Game programming can be easy if you know what you're doing. Therefore, we've tried to present the material in a way that not only gives you helpful code snippets to reuse, but also actually shows you the big picture of game development. Understanding the underlying principles is the key to tackling ever more complex game ideas. You'll not only be able to write games similar to the ones developed over the course of this book, but you'll also be equipped with enough knowledge to go to the Web or the bookstore and take on new areas of game development on your own.

Who This Book Is For

This book is aimed first and foremost at complete beginners in game programming. You don't need any prior knowledge on the subject matter; we'll walk you through all of the basics. However, we need to assume a little knowledge on your end about Java. If you feel rusty on the matter, we'd suggest refreshing your memory by reading *Thinking in Java* by Bruce Eckel (Prentice Hall, 2006), an excellent introductory text on the programming language. Other than that, there are no other requirements. No prior exposure to Android or Eclipse is necessary!

This book is also aimed at intermediate-level game programmers who want to get their hands dirty with Android. While some of the material may be old news for you, there are still a lot of tips and hints contained here that should make reading this book worthwhile. Android is a strange beast at times, and this book should be considered your battle guide.

How This Book Is Structured

This book takes an iterative approach in that we'll slowly but surely work our way from the absolute basics to the esoteric heights of hardware-accelerated game programming goodness. Over the course of the book, we'll build up a reusable code base that you can use as the foundation for most types of games.

If you're reading this book purely as a learning exercise, we suggest going through the chapters in sequence, starting from Chapter 1. Each chapter builds off of the previous chapter, which makes for a good learning experience.

If you're reading this book with the intent to publish a new game at the end, we highly recommend you skip to Chapter 14 and learn about designing your game to be marketable and make money, then come back to the beginning and begin development.

Of course, more experienced readers can skip certain sections they feel confident with. Just make sure to read through the code listings of sections you skim over so you will understand how the classes and interfaces are used in subsequent, more advanced sections.

Downloading the Code

This book is fully self-contained; all the code necessary to run the examples and games is included. However, copying the listings from the book to Eclipse is error prone, and games do not consist of code alone, but also have assets that you can't easily copy out of the book. We took great care to ensure that all the listings in this book are error free, but the gremlins are always hard at work.

To make this a smooth ride, you can download the source code and related assets from the Apress website. For more details, see the book's product page at www.apress.com/9781484204733. The download file will offer you the following:

- The complete source code and assets available from the project's Subversion repository. The code is licensed under the Apache License 2.0 and hence is free to use in commercial and noncommercial projects. The assets are licensed under the Creative Commons BY-SA 3.0. You can use and modify them for your commercial projects, but you have to put your assets under the same license!

For each chapter that contains code, there's an equivalent Android Studio project. The projects do not depend on each other, as we'll iteratively improve some of the framework classes over the course of the book. Therefore, each project stands on its own. The code for both Chapters 5 and 6 is contained in the *ch06-mrnom* project.

Contacting the Authors

Should you have any questions or comments—or even spot a mistake you think we should know about—you can contact Mario Zechner by registering an account and posting at <http://badlogicgames.com/forum/viewforum.php?f=21>, or Robert Green by visiting www.rbgrn.net/contact.

We prefer being contacted through the forums. That way other readers benefit as well, as they can look up already answered questions or contribute to the discussion!