Extracted from:

Build Location-Based Projects for iOS

GPS, Sensors, and Maps

The Pragmatic Bookshelf

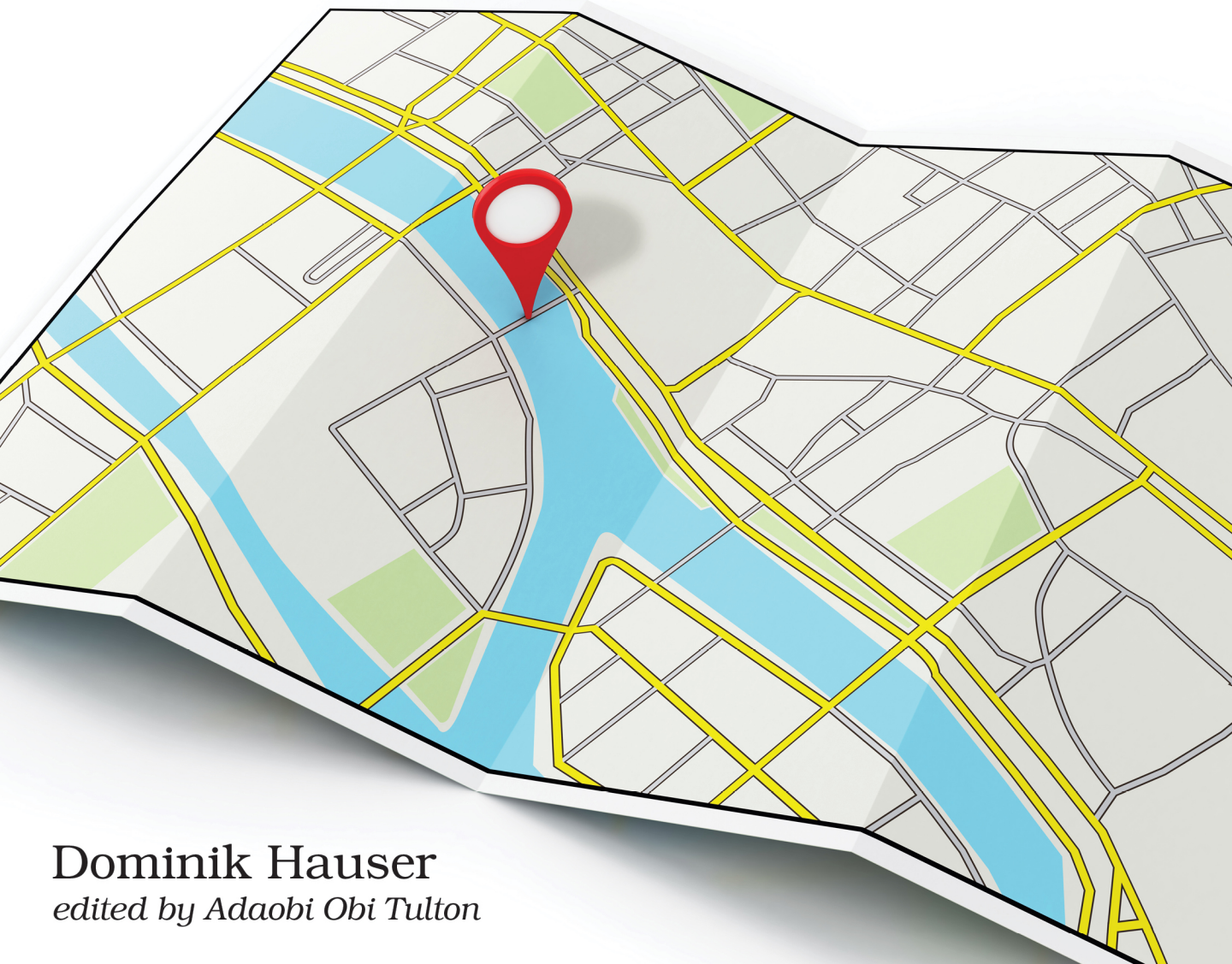
Raleigh, North Carolina

The Pragmatic Programmers

Pragmatic exPress

Build Location-Based Projects for iOS

GPS, Sensors, and Maps



Dominik Hauser

edited by Adaobi Obi Tulton

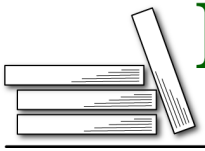
Build Location-Based Projects for iOS

GPS, Sensors, and Maps

Dominik Hauser

The Pragmatic Bookshelf

Raleigh, North Carolina



Our Pragmatic books, screencasts, and audio books can help you and your team create better software and have more fun. Visit us at *https://pragprog.com*.

The team that produced this book includes:

Publisher: Andy Hunt
VP of Operations: Janet Furlow
Executive Editor: Dave Rankin
Copy Editor: Rachel Monaghan
Layout: Gilson Graphics

For sales, volume licensing, and support, please contact *support@pragprog.com*.

For international rights, please contact *rights@pragprog.com*.

ISBN-13: 978-1-68050-781-2
Encoded using the finest acid-free high-entropy binary digits.
Book version: P1.0—August 2020

Introduction

Learning new skills works best when it's fun. You dive deeper and deeper. Hours pass and you can't stop reading and experimenting.

But there's also the outside world. And there's the feeling you have after a nice walk in the woods. Wouldn't it be cool if you could combine learning about iOS development with being outside?

This book features four inspiring projects you get to test outside. You'll draw on a map by moving in the real world, measure the length of a swing while swinging, measure how long someone's outside, and share augmented reality views. As we build these projects, we'll also be looking at the three main approaches to constructing user interfaces: storyboards, code, and SwiftUI.

Who This Book Is For

This book is for developers who already have rudimentary experience with iOS development and want to learn about GPS, sensors, and ARKit. And even if you already have experience with location-based tools, the projects in this book might give you new ideas for your own apps.

I assume that you already have Xcode installed and know the basics about its structure. If you have worked through a beginner's book or other beginner resources, you are perfectly prepared to work through the projects in this book. The first three chapters, in particular, cover in detail what you have to do in Xcode to follow along.

Who Am I?

My name is Dominik Hauser. After I finished my PhD in physics, I worked at a university as a tutor and as such was looking for an app about physics and math formulas. It was the beginning of the App Store, and the only apps I could find didn't fit my needs. So I decided to build my own. Two years later I switched careers to become a full-time iOS developer, and I haven't looked back since then.

What I like most about iOS development is when someone finds a cool way to use the available resources to create something nobody has seen before. One of these moments was the invention of the pull-to-refresh control by Loren Brichter in the early days of iOS. The users loved this kind of user interaction, and Apple added it to iOS some years later.

For me iOS still has lots of corners where discoveries can be made. You need to keep your eyes and your mind open, and you need to experiment and explore. Let your mind wander sometimes and try ideas that seem silly.

One day while riding my bike home after work, I had just such an idea: a pull-to-refresh control that's also a break-out game played by scrolling the table view. A few months later I implemented it and posted it to GitHub, a site to share open source projects. I think (and hope) no one ever put that into a real app because it really was silly, but nevertheless, it received more than 2,000 stars.

The projects in this book are meant to be a starting point for your own experiments. I hope while working through the chapters you get ideas for how to create something special and unique from the example apps. Explore, try ideas even if they seem strange or too big, and, most importantly, have fun!

What This Book Covers

To maximize what you learn and to keep the content of the book interesting, we'll use three different approaches for the user interface in this book. In two chapters we build the user interface using a storyboard. We also build user interfaces in code and using SwiftUI. A good iOS developer should be able to choose the best user interface framework for the problem at hand, and learning the three major approaches will make you a better developer.

Chapter 1: Drawing on Maps

In this chapter we build an app that lets us draw on a map by moving around in the real world. We add overlays to maps and fetch the device's location from the GPS sensors. Finally, we share the resulting image on social media.

The user interface in this project is created in code.

Chapter 2: Measuring Length with Gravitation

We use the accelerometer built into the iPhone to measure the period of a swing. We draw the data onto the screen using Core Graphics.

The user interface in this project is created using a storyboard.

Chapter 3: Automating with Geofences

In this chapter we build an app that allows us to set a geofence. Whenever the iPhone enters or exits the geofence, our app stores the time. We use this data to calculate how long the user has been outside of this geofence each day.

The user interface in this project is created using SwiftUI.

Chapter 4: Sharing Augmented Reality

In this chapter we build an app that allows us to draw a virtual text for other app users to find. We use ARKit and SpriteKit to render the text into the augmented reality view.

The user interface in this project is created using a storyboard.

Free Developer Account

You don't need a paid Apple Developer account to work though this book. But Apple restricts the number of apps you can install on your device to three for the free developer account. You can build and run apps on the iOS simulator that comes with Xcode without any restrictions. If you have a free account, I suggest you select the three apps you would like to test on your real iOS device before you start working through each chapter. Keep in mind that the simulator cannot simulate motion events.

In addition, with a free account the apps work only for seven days before you have to deploy them again onto the device using Xcode.

How to Read This Book

The chapters are independent from one another. But if you're new to iOS development in general, you should read the book from cover to cover. The first two chapters explain in detail what to do and why to do it that way, and the last two chapters assume you've read or already understand those explanations.

As I mentioned, throughout the book we use storyboards, code, and SwiftUI to create the user interfaces for the apps. When you have finished the book you can increase your skills and maximize your learning by recreating the projects using a different approach for the user interface. Or you can even build an app that combines two or three of the projects. Invent something new.

Creating with Swift

The code in this book is written in Swift, which is the de facto standard for iOS apps these days. It's a powerful language that's easy for beginners to learn.

I assume that you have a basic knowledge of Swift and that you already have created some simple iOS projects with it. If you need to refresh your Swift knowledge, you can download *The Swift Programming Language* for free from the official Swift website.[1]

If you don't understand some of the code in this book, search *The Swift Programming Language* or go to Stack Overflow[2] to get answers.

Online Resources

This book has a website where you will find links to source code and errata.[3] You are free to use the source code in your own applications as you see fit.

If you purchased the ebook, you can click the gray box above the code extracts to directly download the extract.

Let's have fun!

1. www.apple.com/swift/
2. stackoverflow.com
3. https://pragprog.com/titles/dhios/