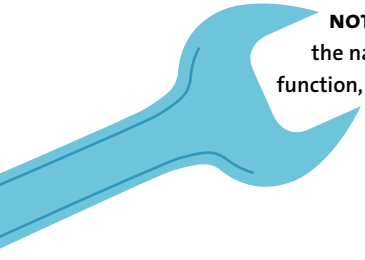
BEFORE WE BEGIN

When it comes to iOS apps, it doesn't matter if you're a new developer whose crowning achievement is a simple coin-flip app, or an experienced developer who's creating the next Flipboard. If you develop apps for iOS you've used frameworks, whether you were aware of it or not.

At a high level, frameworks provide access to low level services through system APIs. These services can range from the creation and management of simple runtime objects like arrays, strings, buttons, and text fields to lower hardware access of cameras, motion accelerometers, and GPS.

Frameworks are a defining characteristic that make a computer program an app for iOS. At the end of the day, all iOS apps are based on and executed in an Objective-C runtime environment. Code in this environment can be written with a mixture of C, C++, and Objective-C, but to execute a binary in iOS and run an app on the iPhone or iPad, that app must ultimately interact with iOS frameworks.



NOTE: Because OS was built on the foundation of Mac OS X, many of the native frameworks carry over with very little loss in performance or function, giving you the power of a desktop platform on a mobile device.

Before we begin, you should know my assumptions and expectations about your background in iOS development. The last thing I want is for you to get half way through this book and realize it's not what you were looking for, or even worse, to reach the end and wish there were more. So let's take a step back and cover some prerequisites, followed by a look at my goals for this book.

PREREQUISITES

In iOS there are two frameworks that are absolutely essential, *Foundation* and *UIKit*. This book covers Apple developed frameworks throughout the iOS architecture including frameworks in the Cocoa Touch layer, Media layer, and Core Services layer. Because Foundation and UIKit are so essential to even the simplest iOS apps, I'm assuming a basic understanding of how these frameworks operate. This enables us to spend more time on the frameworks that will give your app an edge—taking advantage of the power of iOS to make your app unique.

Because Foundation and UIKit define the base classes for all objects in iOS, it's impossible to develop an app that executes in the iOS runtime without them. For this reason, when a new iOS project is created in Xcode, these frameworks are automatically included by default.