



Vorwort

Jan Tittel, Jochen Baumann

Apps für iOS entwickeln

Am Beispiel einer realen App

ISBN (Buch): 978-3-446-43192-8

ISBN (E-Book): 978-3-446-43314-4

Weitere Informationen oder Bestellungen unter

<http://www.hanser-fachbuch.de/978-3-446-43192-8>

sowie im Buchhandel.

1

Einführung

Der Buchmarkt hält bereits einiges an Literatur zur App-Entwicklung für Smartphones bereit. Angefangen vom einfachen, dünnen Buch für Einsteiger bis hin zum dicken Wälzer und Nachschlagewerk für Profis. Warum also ein weiteres Buch?

Wir sind selbst seit Jahren als Entwickler tätig und haben mit dem Aufkommen des Booms selbstverständlich auch mit der App-Entwicklung für Smartphones begonnen. Dabei haben wir insbesondere zwei Schwachstellen ausgemacht.

Zum einen holen die meisten Bücher oder Tutorien sehr weit aus und beginnen mit den einfachsten Dingen. Ein Großteil der Entwickler startet mit einer App allerdings nicht sein erstes Projekt, sondern verfügt bereits über umfangreiche Kenntnisse in anderen Programmiersprachen. Für viele Entwickler ist daher ein Schnelleinstieg vollkommen ausreichend.

Zum anderen ist es für eine erfolgreiche App unerlässlich, diese für mehrere Plattformen, zumindest für die beiden derzeit erfolgreichsten (iOS und Android), zu entwickeln. Zusätzlich vermissen wir oftmals die Umsetzung von realen Beispielen, da es dabei dann doch einige Aspekte zu berücksichtigen gilt, die in einer rein technisch funktionalen Beschreibung der App-Entwicklung untergehen.

So wurde die Idee geboren, eine tatsächlich existierende Beispiel-App für mehrere Plattformen zu entwickeln und daraus gleichzeitig ein Buch – einen Schnelleinstieg für IT-Profis – zu machen. Durch dieses Konzept ist es für Sie als Leser möglich, sich an ein und demselben Beispiel einen Überblick über die verschiedenen Plattformen zu verschaffen und gleichzeitig mit möglichst wenig Aufwand die Unterschiede bei der Entwicklung herauszuarbeiten. Das vorliegende Buch konzentriert sich daher neben den notwendigen Grundlagen auf die Umsetzung eines umfangreicheren Beispiels.

Parallel erscheint dieses Buch auch für Android (*Apps für Android entwickeln*, Carl Hanser Verlag 2013, ISBN 978-3-446-43191-1). Weitere Plattformen sind bereits in Planung, deren konkrete Umsetzung hängt aber auch von der Entwicklung des Marktes ab, sodass wir an dieser Stelle keine falschen Versprechungen machen möchten.

■ 1.1 Das Smartphone

Smartphones gibt es schon eine ganze Weile. Berücksichtigt man den technischen Fortschritt, könnte man auch sagen, seit einer halben Ewigkeit. Die ersten Geräte waren allerdings eher unter den Namen *Pocket PC* oder *PDA* geläufig. Wenngleich es von Windows Mobile 2003 auch eine Smartphone-Edition gab, verfügten diese Geräte im Gegensatz zu den Pocket PCs über keinen Touchscreen. Das Smartphone, wie wir es heute kennen, ist technisch gesehen keine neue Erfindung von Apple. Mit dem ersten iPhone 2007 lieferte Apple allerdings als erster Hersteller ein bedienerfreundliches Gesamtpaket. Während Geräte vor dem iPhone hauptsächlich von Geschäftsleuten und Technik-Freaks verwendet wurden, entwickelte sich mit dem iPhone langsam der Massenmarkt.

Der *App Store* folgte erst ein Jahr später, 2008, und löste unter Entwicklern stark gemischte Gefühle aus. So ließen sich Apps nicht frei vertreiben wie für Windows Mobile, sondern nur über den App Store, wobei Apple gleichzeitig eine Provision für den Umsatz von kostenpflichtigen Apps einbehält.

Letztendlich war es aber auch das stetig wachsende Angebot an Apps und somit immer mehr Möglichkeiten für den Anwender, welches für den starken Auftrieb des iPhones gesorgt hat. Gleichzeitig wuchsen damit wiederum die Möglichkeiten für Entwickler, sodass das Konzept von Apple für beide Seiten aufgegangen ist.

Google zog mit *Android* und dem heute *Play Store* genannten Markt zunächst langsam voran, aber dann vor allem dank deutlicher preisgünstigerer Geräte von unterschiedlichen Herstellern rasant an Apples iPhone vorbei. Nach den aktuellen Statistiken werden mittlerweile grob drei bis vier Mal so viele Geräte mit Android wie das iPhone verkauft. Als Entwickler sollten Sie Android daher, allein schon aufgrund der Masse, ebenso genügend Aufmerksamkeit widmen.

Es gibt aber auch noch andere Statistiken. Und die sagen deutlich aus, dass Android-Anwender wesentlich weniger Geld für Apps ausgeben, wohingegen iPhone-Anwender auch in teure Apps investieren. Trotz der deutlich höheren Anzahl an Geräten mit Android, lässt sich zumindest derzeit immer noch mehr Geld mit Apps für iOS verdienen. Das Betriebssystem iOS hieß zu Beginn übrigens noch iPhone OS, wurde dann aber mit der Vorstellung des iPad umbenannt.

Als ernsthafter App-Entwickler bleibt Ihnen derzeit nichts anderes übrig, als den Markt ständig zu beobachten und sich mit allen relevanten Plattformen auseinanderzusetzen.

■ 1.2 An wen richtet sich dieses Buch?

Dieses Buch richtet sich an Entwickler, die bereits grundlegend mit der modernen objektorientierten Programmierung vertraut sind. Vorkenntnisse in Objective-C, der Programmiersprache für Mac OS und iOS, sind nicht notwendig. Hierfür haben wir einen kleinen Crash-Kurs in Objective-C mit in das Buch aufgenommen. Der Schwerpunkt liegt dabei allerdings auf der Syntax und den Grundlagen von Objective-C, die für das weitere Ver-

ständnis hilfreich sind. Es wird nicht auf Fragen für Einsteiger eingegangen, worum es sich beispielsweise bei der objektorientierten Programmierung überhaupt handelt. Dieses allgemeine Wissen setzen wir voraus. Intention dieses Buches ist es, Ihr vorhandenes allgemeines Programmierwissen auf iOS zu übertragen und Ihnen somit einen schnellen und kompakten Einstieg in die App-Entwicklung zu bieten.



HINWEIS: Für absolute Anfänger und Einsteiger in die Anwendungsentwicklung, ohne Vorkenntnisse in irgendeiner objektorientierten Programmiersprache, ist dieses Buch wenig geeignet, da allgemeine Kenntnisse vorausgesetzt werden.

■ 1.3 Buchaufbau und verwendete Technologien

Nach dieser Einführung wird zunächst die Installation und grundlegende Konfiguration der Entwicklungsumgebung Xcode beschrieben. Hierfür wird zwingend ein Mac benötigt, da Xcode nur unter Mac OS läuft. Anschließend zeigen wir Ihnen noch die notwendigen Schritte zur Registrierung für das Entwicklerprogramm von Apple für iOS sowie zur Freischaltung eigener Geräte, damit Sie Ihre Apps nicht nur im mitgelieferten Emulator testen können.

In Kapitel 3 liefern wir Ihnen einen Schnelleinstieg in Xcode und Objective-C. Hierzu stellen wir Ihnen zuerst Schritt für Schritt die Entwicklung einer ersten eigenen App vor und testen diese dann auch im Emulator sowie auf einem eigenen Gerät. Das Kapitel schließt mit einem Crash-Kurs in die Programmiersprache Objective-C ab, in dem wir Ihnen die wichtigsten Grundlagen sowie die Syntax vorstellen.

Wir haben die ersten Kapitel bis zu diesem Punkt bewusst mit zahlreichen Screenshots versehen, damit sich auch die Leser unter Ihnen problemlos mit den Anweisungen zurecht finden, die bislang kaum Erfahrung mit Mac OS haben und sich im Rahmen der App-Entwicklung erstmalig mit der Oberfläche auseinandersetzen.

Die weiteren Kapitel zeigen anhand eines umfangreichen und realen Beispielprojekts die App-Entwicklung für iOS. Das Beispielprojekt umfasst unter anderem die folgenden Technologien und Features:

- Mit dem Storyboard arbeiten
- Benutzeroberflächen erstellen
- Mit Navigationselementen arbeiten
- Daten beziehungsweise Werte zwischen mehreren Ansichten austauschen
- Berücksichtigung unterschiedlicher Bildschirmauflösungen und Geräteausrichtungen mit Autolayout
- Eigene Klassen erstellen

- Auf das Vorhandensein von Hardware-Features prüfen
- Tabellenansichten darstellen und formatieren
- Text erstellen und bearbeiten
- Dateien lesen und schreiben sowie mit dem Dateisystem arbeiten
- E-Mails versenden
- Auf unterschiedliche Gesten reagieren
- Einfache Grafikfunktionen nutzen
- Auf Kamera und Fotos zugreifen
- Tasks im Hintergrund ausführen
- Audio abspielen und aufnehmen
- Auf Änderungen des Audio-Eingangs und -Ausgangs reagieren
- Multimedia-Ereignisse verarbeiten
- Ortungsdienste und Karten verwenden, inklusive Reverse Geocoding
- Timer einsetzen und Code verzögert ausführen
- Benachrichtigungen für App-interne Ereignisse senden und empfangen
- Auf Statusänderungen einer App sowie einer Sicht reagieren
- Datenbanken mit *Core Data* erstellen
- Daten erstellen, abfragen, bearbeiten und löschen
- Daten und Arrays filtern
- Lokale Benachrichtigungen auslösen
- Lokalisierung einer App für mehrere Sprachen

Damit Sie vorab wissen, woran Sie arbeiten und nicht wie in einigen anderen Tutorien ins kalte Wasser geworfen werden und den Sinn und Zweck der beschriebenen Schritte erst zum Schluss verstehen, wird die App im folgenden Abschnitt noch kurz vorgestellt.

■ 1.4 Vorstellung des Beispielprojekts

Das Beispielprojekt, welches im Verlauf dieses Buches erstellt wird, ist eine App für die Erstellung von Notizen in Form von Text, Bild oder Audio.



Sie finden die App beziehungsweise den Quellcode nicht nur unter *www.downloads.hanser.de*, sondern auch als reale App im App Store. Geben Sie dafür als Suchbegriff in iTunes *scyte notes* ein. Weitere Informationen zur App erhalten Sie auch auf der dazugehörigen Website *www.scyte.eu*.

Bild 1.1 zeigt die Hauptansicht der App mit der tabellarischen Auflistung aller Text-Notizen.

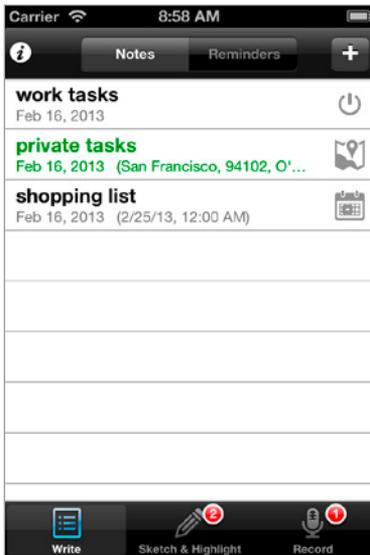


Bild 1.1

Tabellarische Darstellung aller Text-Notizen

Nach der Erstellung einer neuen Notiz oder der Auswahl einer vorhandenen Notiz gelangt der Anwender in die für Texte, Bilder oder Aufnahmen zugehörige Bearbeitungsansicht, wie am Beispiel einer Text-Notiz in Bild 1.2 zu sehen. Die Bearbeitungsmöglichkeiten sind an dieser Stelle bewusst einfach gehalten, da alle Notizen zum einfachen späteren Versenden per E-Mail in gängigen Standard-Dateiformaten, wie *.txt*, *.jpg* und *.wav*, gespeichert werden. Der Anwender ist dadurch nicht an den Server eines Anbieters sowie an ein individuelles Format gebunden. Über iTunes kann der Anwender außerdem jederzeit seine kompletten Notizen auf der lokalen Festplatte speichern.



Bild 1.2

Bearbeitungsansicht für Text-Notizen

Nach dem erstmaligen Speichern einer neuen Notiz gelangt der Anwender automatisch in die Ansicht zum Einrichten von zeit- oder ortbasierten Erinnerungen, wie in Bild 1.3 dargestellt. Eine nachträgliche Änderung ist ebenso jederzeit möglich, indem einfach das entsprechende Icon am rechten Rand des jeweiligen Eintrags in der Tabellenansicht angetippt wird. Pro Notiz kann jeweils nur eine Erinnerung eingerichtet werden. Im Gegensatz zu den Notizen selbst, werden die Erinnerungen nicht in Form von Dateien, sondern in einer Datenbank abgelegt.

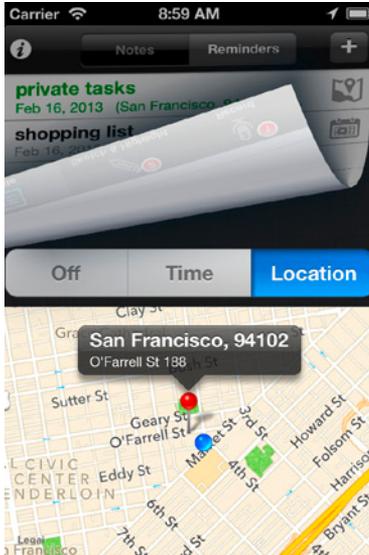


Bild 1.3

Ansicht zum Hinzufügen von zeit- oder ortbasierten Erinnerungen

Neben dem Anfertigen von einfachen Skizzen als Bild-Notiz ist es zusätzlich möglich, mit der Kamera abfotografierten Text mit Markierungen zu versehen. Mit den beiden Stifffarben Rot und Grün wird daher ausschließlich transparent gezeichnet, wie in Bild 1.4 zu sehen.

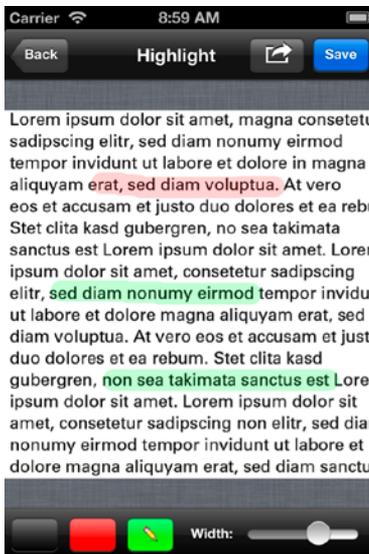


Bild 1.4

Bearbeitungsansicht für Skizzen und Markierungen

