

Inhaltsverzeichnis

Vorwort	13
1 Von der Idee zur App Store-App	19
1.1 Hard- und Software, die man braucht	19
1.2 Der kostenlose Entwickler-Account	20
1.3 Das iOS SDK laden und installieren	21
1.4 Bestandteile des iOS SDK	23
1.4.1 Xcode	24
1.4.2 Interface Builder	25
1.4.3 iPhone und iPad Simulator	26
1.4.4 Instruments	27
1.4.5 Icon Composer	28
1.4.6 Pixie	30
1.4.7 Dashcode	31
1.4.8 Core Image Fun House	32
1.5 Registrieren und Einreichen der App	33
2 Die Grundlagen von Objective-C	35
2.1 Cocoa und Cocoa Touch	35
2.2 Objective-C	36
2.3 Erste Schritte in Xcode	37
2.4 Hallo, Debugger	40
2.5 Variablen in Objective-C	46
2.5.1 Variablen benennen	46
2.5.2 Datentypen	47
2.6 Variablen deklarieren	48
2.7 Debugger-Ausgabe von Variableninhalten	50
2.8 Berechnungen und Datentypen	54
2.9 Verzweigungen und Bedingungen	56
2.10 Schleifen	58
2.11 Prozeduren und Funktionen	60
2.12 Werteparameter und Referenzparameter	61
2.13 Funktionsdeklaration und Headerdateien	63
2.14 Konstanten und Präprozessor-Ersetzungen	65
2.15 Kommentare	66

3	Objective-C – objektorientiert	67
3.1	Erste Schritte im Interface Builder	67
3.2	Steuerelemente platzieren und ausrichten	73
3.3	Steuerelemente definieren	76
3.4	Auf Events reagieren	77
3.5	IBAction und IBOutlet à la Xcode 4	81
3.6	Eigene Methoden definieren	84
3.7	Methoden-Recherche in der Hilfe	86
3.8	Die Label-Farbe ändern	89
3.9	Eine Methode mit zwei Parametern	89
3.10	Rückblick und Ausblick	90
4	Schritt für Schritt zur App	91
4.1	Xcode 3: Eine App in sechs Schritten	91
4.1.1	Gestalten der Formulare im Interface Builder	91
4.1.2	Erzeugen der Objektvariablen in der Headerdatei	92
4.1.3	Verknüpfung der Objektvariablen mit den Steuerelementen	92
4.1.4	Methoden zur Behandlung der Ereignisse deklarieren	93
4.1.5	Verknüpfung der IBAction-Methoden mit den Steuerelementen	94
4.1.6	Ereignisbehandlungsroutinen codieren	94
4.2	Xcode 4: Eine App in drei Schritten	95
4.2.1	Gestaltung des Formulars	95
4.2.2	Objektvariablen erstellen	96
4.2.3	Methoden zur Behandlung der Ereignisse	97
5	Mit dem Benutzer kommunizieren	99
5.1	Vorbereitung im Interface Builder	100
5.2	Instanzvariablen anlegen	100
5.3	Instanzvariablen und Steuerelemente verknüpfen	101
5.4	Ereignisbehandlungsroutinen deklarieren	102
5.5	Exkurs: Eventhandler-Varianten	103
5.6	Ereignisroutinen verknüpfen	104
5.7	Texteingaben übertragen	107
5.8	Activity Indicator per Schaltfläche steuern	107
5.9	Den Schalter (Switch) nutzen	109
5.10	Segmentwechsel bearbeiten	110
5.11	Schieberegler (Slider) auswerten	111
5.12	Das Keyboard stört	114
6	Ereignisse untersuchen und debuggen	117
6.1	Ein Code für alle Ereignisse	117
6.2	Test per NSLog	121
6.3	Debuggen	122
6.3.1	Übersichtlichkeit ist Trumpf	122
6.3.2	Wann passiert eigentlich was?	123
6.3.3	Tiefe Einblicke im Debugger-Fenster	126

7	Arbeiten mit Views	131
7.1	Infodialog anzeigen	132
7.2	Arbeiten und navigieren mit mehreren Views	145
7.2.1	Steuerelemente im Interface Builder vorbereiten	146
7.2.2	Instanzvariablen und Eventroutinen deklarieren	148
7.2.3	Code beim Wechsel der Schaltfläche	150
7.2.4	Ein wenig Animation muss sein	152
8	Listen und Tabellen	155
8.1	Die Tabelle und ihre Datenquelle	156
8.2	Ein UITableView-Praxistest	157
8.3	Parameter nutzen	160
8.4	Formatierte Zellen	161
8.5	Symbolbilder anzeigen	162
8.6	Mehrere Abschnitte anzeigen	165
8.7	Speicher-Hygiene	168
8.8	Touch und Action!	169
9	Listenauswahl mit dem Picker	171
9.1	Die Standardschritte zum ersten Pickerfeld	172
9.2	Die Zahl der Wahl	176
9.3	Angepasste Spaltenbreite	179
9.4	NSArray – der Begleiter von UIPickerView und UITableView	180
9.5	NSMutableArray – das flexible NSArray	183
9.6	Datumsauswahl	184
9.7	Kalender und Zeiten	188
9.7.1	NSDate – Aufbewahrungsort für Datumsangaben	189
9.7.2	NSCalendar – globale Zeitdarstellung	190
9.7.3	NSDateComponents – Datum und Zeit auf dem Seziertisch	190
10	Testen auf dem iPhone	195
10.1	Das Prozedere im Überblick	195
10.2	Geräte für die Entwicklung freigeben	196
10.3	Zertifikate erzeugen, anfordern und installieren	196
10.4	Entwicklungsgeräte hinzufügen	202
10.5	Eindeutige APP-ID	203
10.6	Das Provisioning Profile	206
10.7	Der erste Start auf dem iPhone, iPad oder iPod touch	208
10.8	Der erste Start	210
11	Eine Applikation für den App Store	211
11.1	Das Prinzip iHo.me	211
11.2	iHo.me und seine Programmierer-Funktionen	212
11.2.1	Ist die Kurz-URL frei?	212
11.2.2	Der nächste Vorschlag	213
11.2.3	URL reservieren	213
11.2.4	Echtwort-Vorschläge	214

Inhaltsverzeichnis

11.3	Das Grundgerüst der Applikation	215
11.3.1	Ein auf Tabs basiertes Applikationsgerüst	215
11.3.2	Ein wenig optisches Makeup	221
11.4	Die Infoseite	223
11.5	Das Tiny URL-Interface	225
11.5.1	Deklarieren der Instanzvariablen	226
11.5.2	Antwortmethoden deklarieren	227
11.5.3	Verbindungen im Interface Builder	227
11.5.4	Erste Code-Arbeiten	228
11.5.5	Lesen aus einer Web-Ressource	230
11.5.6	Eine gute Abkürzung	232
11.5.7	Startarbeiten	246
11.6	Feintuning für den App Store	246
12	App Store- und Ad Hoc-Vertrieb	249
12.1	Zertifikat anfordern	250
12.2	Zertifikatsanforderung zur Prüfung hochladen	252
12.3	Noch leichter: Der Assistent	253
12.4	iOS Distribution Provisioning Profile	255
12.4.1	Ad Hoc-Profile erzeugen und herunterladen	256
12.4.2	App Store-Profile erzeugen und downloaden	258
12.5	Anpassungen in der Xcode-Applikation	258
12.5.1	Zutaten komplett?	259
12.5.2	Code Signing-Identity auswählen	261
12.5.3	Der Build-Prozess	262
12.6	Einreichen im App Store	264
12.7	Warten auf Apple	269
12.8	Verträge und Bankinformationen	271
13	iPad ahoi	273
13.1	iPad vs. iPhone/iPod touch	273
13.1.1	Der Size Inspector für die richtige Proportion	274
13.2	Das Split View-Prinzip des iPad	277
13.2.1	Die Master-Detail-Schablone	277
A	Code-Rezepte, Tipps und Tricks	279
A.1	Xcode, Simulator, Interface Builder und Tools	279
A.1.1	Wechsel zwischen Header- und Moduldatei	279
A.1.2	Code-Vervollständigung	279
A.1.3	Projektname umbenennen	280
A.1.4	Ein neues Framework in Xcode 4 hinzufügen	280
A.1.5	Nach dem Approval-Prozess	281
A.1.6	Die richtige Methode finden	281
A.1.7	XCode-Shortcuts für Typinator	283
A.1.8	Visuelle Gestaltung ohne Interface Builder	284
A.1.9	Eigene Code-Vervollständigung mit Completion Dictionary	284

A.1.10	Automatische Methodenklammern	285
A.1.11	Groß-/Kleinschreibung im Dateisystem	286
A.1.12	Dateiverzeichnis des iOS Simulators	286
A.1.13	Aus .xib mach .m	286
A.1.14	Info zu Objective-C- und Cocoa-Bezeichnern	286
A.1.15	Schnellsuche nach Dateien und Definitionen	287
A.1.16	Komplette Xcode-Shortcut-Liste	288
A.1.17	Tastenbelegung	289
A.1.18	Liste der Methoden und Funktionen	290
A.1.19	Aufgaben hervorheben	290
A.1.20	Code in logische Abschnitte teilen	291
A.1.21	Zusammenhängende Methoden markieren	291
A.1.22	Variablen global umbenennen	291
A.1.23	Autovervollständigung durchlaufen	292
A.1.24	Automatische Code-Einrückung	292
A.1.25	Codeblock auskommentieren	292
A.1.26	Backslash	292
A.1.27	Nicht zusammenhängende Codeteile markieren	293
A.1.28	Weg mit der „Are you sure to undo“-Abfrage!	293
A.1.29	Neuer Firmenname in eigenen Templates	293
A.1.30	„Base SDK missing“ umgehen	293
A.1.31	Rückwärtskompatibel bleiben	294
A.1.32	Eigene Einstellungen für Xcode	295
A.1.33	Klassenbrowser für die bessere Übersicht	295
A.1.34	Perfekter Browser durch das Klassenchaos	296
A.1.35	Hilfslinien zur Gestaltung	297
A.1.36	Interface Builder-Objektinformationen	298
A.1.37	iPhone/iPad Simulator: Touch-Gesten mit der Maus	298
A.1.38	Schaltflächen-Icons leicht gestalten	298
A.1.39	HTML-Dateien für das Skalieren vorbereiten	299
A.2	Allgemeine Tipps.	299
A.2.1	Ist eine Farbe hell oder dunkel?	299
A.2.2	Ausrichtung des Geräts ermitteln	300
A.2.3	Auf Schüttelgesten reagieren	300
A.2.4	Kalendereintrag hinzufügen	301
A.2.5	Das Blitzlicht des iPhone 4 bzw. 4S nutzen	302
A.2.6	Das iPhone vibrieren lassen	302
A.2.7	Das iPhone wach halten	302
A.2.8	Zeitgeber verwenden	302
A.2.9	Eine Methode zeitversetzt starten	303
A.2.10	Singleton – Aufbewahrung für globale Variablen und Funktionen	303
A.2.11	Die optimale Referenz-App	305
A.2.12	UILabels zum Anklicken	305
A.2.13	Activity-Anzeige	306
A.2.14	Einstellungen speichern und laden	307
A.2.15	iPhone oder Simulator?	307

Inhaltsverzeichnis

	A.2.16	Einstellungen speichern und laden	308
	A.2.17	Einfache Zahlen als Objekt	308
	A.2.18	Zufallsfarbe	309
	A.2.19	TabBar mit kleinerem View	309
A.3		Bugs und Workarounds	310
	A.3.1	Sound auch im Simulator	310
	A.3.2	Verschwundene Interface Builder-Icons	310
	A.3.3	Interface Builder erkennt IBAction und IOutlet nicht mehr	310
	A.3.4	Invalid Entitlements	311
	A.3.5	SigAbt ohne ersichtlichen Grund	312
A.4		Systeminformationen und -funktionen	312
	A.4.1	Den aktuellen Nutzernamen ermitteln	312
	A.4.2	Strukturen debuggen	312
	A.4.3	Aktuelle Lokalisierung ermitteln	312
	A.4.4	iPhone mit JailBreak?	313
	A.4.5	Keyboard ausblenden	313
	A.4.6	Display-Informationen	313
	A.4.7	URL im Safari aufrufen	313
	A.4.8	E-Mail verschicken	314
	A.4.9	Telefonnummer wählen	314
	A.4.10	Google Maps mit Koordinaten aufrufen	315
	A.4.11	Messagebox/Alert anzeigen	315
	A.4.12	Applikationsicon	317
	A.4.13	Splashscreen erzeugen	317
	A.4.14	Splashscreens auf dem iPad	318
	A.4.15	Ein Splashscreen für 1 Sekunde	318
	A.4.16	Statusbar anzeigen und verbergen	318
	A.4.17	Batterieinfos auslesen	318
	A.4.18	Auf Batterieänderungen reagieren	319
	A.4.19	Aktuelle Lage des iPhones/iPads	320
	A.4.20	Annäherungsprüfung	320
	A.4.21	OS-Version ermitteln	321
	A.4.22	UDID auslesen	321
	A.4.23	iTunes-Name des iPhones	322
	A.4.24	iPhone-Autolock vermeiden	322
	A.4.25	Den ersten App-Start ermitteln	322
	A.4.26	Massenweise App-Informationen	323
A.5		Makros für Objective-C	323
	A.5.1	Prüfen, ob eine Datei existiert	323
	A.5.2	Zugriff auf das Haupt-View	323
	A.5.3	#define – das verkannte Genie	323
	A.5.4	Objekt-Inhalte im Debugger anzeigen	324
	A.5.5	#define über mehrere Zeilen	324
	A.5.6	Alertbox mit einfachem Aufruf	324
	A.5.7	Sound abspielen	325
	A.5.8	Bundle- versus Dokumentenverzeichnis	326

A.6	UITableView und UIPickerView	326
A.6.1	UITableView-Zeile anklicken	326
A.6.2	UITableView – individuell	327
A.6.3	UITableView mit festem Hintergrund	329
A.6.4	Zu einem bestimmten Eintrag scrollen	330
A.6.5	Zeile in einer UITableView auswählen	331
A.6.6	Hintergrundfarbe einer Zelle setzen	331
A.6.7	Aktuell ausgewählte Zeile in einem UITableView	331
A.6.8	UIPickerView-Komponenten synchronisieren	331
A.7	UIView und UIViewController	332
A.7.1	UIView mit runden Ecken	332
A.7.2	Rotieren	334
A.7.3	Hintergrundfarbe im View setzen	334
A.7.4	Hintergrundbild im View setzen	334
A.7.5	Einen modalen UIViewController anzeigen	334
A.8	NSString	335
A.8.1	Wörter zählen	335
A.8.2	NSStrings verbinden	335
A.8.3	Numerische Werte in NSString	335
A.8.4	Typumwandlung	336
A.8.5	Führende Nullen	336
A.8.6	Zeichenketten vergleichen	336
A.8.7	Dateiinhalte in NSString einlesen und speichern	337
A.9	UITextField und UILabel	338
A.9.1	Labels vertikal oben ausrichten	338
A.9.2	Höhe eines Labels bestimmen	338
A.9.3	Eingabebeschränkung für Textfelder	338
A.9.4	Textlabel rotieren	339
A.10	NSDate – Rechnen mit Zeit und Datum	340
A.10.1	Sekunden seit Referenzdatum	340
A.10.2	Das Jahr ermitteln	340
A.10.3	Heutiges Datum	340
A.10.4	Datumswerte vergleichen	340
A.10.5	Datumsarithmetik	341
A.10.6	Zeitformatierung	341
A.11	Arbeiten mit XML-Daten	341
A.11.1	Alle XML-Bibliotheken in einer App	343
A.11.2	TBXML in eigene Projekte einbinden	344
A.11.3	Kurse der Europäischen Zentralbank lesen	345
A.12	Dateisystem	347
A.12.1	Der Pfad zum Datenverzeichnis	347
A.12.2	Verzeichnisse und Dateien eines Ordners auflisten	348
A.12.3	Funktionen des Dateisystems	348
A.13	iPad	348
A.13.1	Individuelle Splashscreens	349
A.13.2	iPhone oder iPad?	349

Inhaltsverzeichnis

A.14	Zugriff auf Internet-Ressourcen	349
A.14.1	Einen Netzwerkindikator in der Statusbar anzeigen	349
A.14.2	Inhalt einer Webdatei lesen	350
A.14.3	E-Mail-Adressen validieren	350
A.15	Accelerometer	351
A.15.1	Den Accelerometer aktivieren	352
A.15.2	Auf Accelerometer-Aktionen reagieren	352
A.16	Was bedeutet eigentlich	353
A.16.1	... @interface?	353
A.16.2	... @property?	353
A.16.3	... @implementation?	354
A.16.4	... @synthesize?	355
A.16.5	... @class?	355
A.17	Know-how-Quellen im Internet	355
	Stichwortverzeichnis	357