

Inhalt

Vorwort	XXV
Teil I: Swift	1
1 Die Programmiersprache Swift	3
1.1 Die Geschichte von Swift	3
1.2 Die Bedeutung von Swift im Apple-Kosmos	5
1.3 Das UI-Framework: SwiftUI	5
1.4 Was Sie als App-Entwickler brauchen	6
1.5 Programmieren für Beginner (und darüber hinaus): Playgrounds	8
1.6 Weitere wichtige Ressourcen	10
1.6.1 Apple-Developer-App	10
1.6.2 Apples Developer-Website	11
1.6.3 Swift.org	12
1.6.4 In eigener Sache	13
2 Grundlagen der Programmierung	14
2.1 Grundlegendes	14
2.1.1 Swift Standard Library	14
2.1.2 print	16
2.1.3 Befehle und Semikolons	17
2.1.4 Operatoren	18
2.2 Variablen und Konstanten	19
2.2.1 Erstellen von Variablen und Konstanten	20

2.2.2	Variablen und Konstanten in der Konsole ausgeben	21
2.2.3	Type Annotation und Type Inference.....	22
2.2.4	Gleichzeitiges Erstellen und Deklarieren mehrerer Variablen und Konstanten	24
2.2.5	Namensrichtlinien.....	24
2.3	Kommentare.....	25
3	Schleifen und Abfragen.....	27
3.1	Schleifen	27
3.1.1	For-In	27
3.1.2	While.....	29
3.1.3	Repeat-While.....	30
3.2	Abfragen	31
3.2.1	If.....	32
3.2.2	Switch.....	37
3.2.3	Guard	40
3.3	Control Transfer Statements	42
3.3.1	Anstoßen eines neuen Schleifendurchlaufs mit continue.....	43
3.3.2	Verlassen der kompletten Schleife mit break.....	43
3.3.3	Labeled Statements.....	44
4	Typen in Swift	47
4.1	Integer	49
4.2	Fließkommazahlen.....	50
4.3	Bool.....	51
4.4	String	51
4.4.1	Erstellen eines Strings	52
4.4.2	Zusammenfügen von Strings	52
4.4.3	Character auslesen	54
4.4.4	Character mittels Index auslesen.....	54
4.4.5	Character entfernen und hinzufügen.....	56
4.4.6	Anzahl der Character zählen	58
4.4.7	Präfix und Suffix prüfen.....	58
4.4.8	String Interpolation.....	58
4.5	Array	59

4.5.1	Erstellen eines Arrays.....	60
4.5.2	Zusammenfügen von Arrays.....	61
4.5.3	Inhalte eines Arrays leeren.....	62
4.5.4	Prüfen, ob ein Array leer ist	62
4.5.5	Anzahl der Elemente eines Arrays zählen.....	63
4.5.6	Zugriff auf die Elemente eines Arrays	63
4.5.7	Neue Elemente zu einem Array hinzufügen.....	64
4.5.8	Bestehende Elemente aus einem Array entfernen	65
4.5.9	Bestehende Elemente eines Arrays ersetzen.....	66
4.5.10	Alle Elemente eines Arrays auslesen und durchlaufen	67
4.6	Set	68
4.6.1	Erstellen eines Sets	68
4.6.2	Inhalte eines bestehenden Sets leeren	69
4.6.3	Prüfen, ob ein Set leer ist.....	70
4.6.4	Anzahl der Elemente eines Sets zählen	70
4.6.5	Element zu einem Set hinzufügen.....	70
4.6.6	Element aus einem Set entfernen	71
4.6.7	Prüfen, ob ein bestimmtes Element in einem Set vorhanden ist ..	71
4.6.8	Alle Elemente eines Sets auslesen und durchlaufen.....	72
4.6.9	Sets miteinander vergleichen.....	72
4.6.10	Neue Sets aus bestehenden Sets erstellen	75
4.7	Dictionary	77
4.7.1	Erstellen eines Dictionaries.....	77
4.7.2	Prüfen, ob ein Dictionary leer ist.....	79
4.7.3	Anzahl der Schlüssel-Wert-Paare eines Dictionaries zählen.....	79
4.7.4	Wert zu einem Schlüssel eines Dictionaries auslesen.....	79
4.7.5	Neues Schlüssel-Wert-Paar zu Dictionary hinzufügen.....	80
4.7.6	Bestehendes Schlüssel-Wert-Paar aus Dictionary entfernen	81
4.7.7	Bestehendes Schlüssel-Wert-Paar aus Dictionary verändern.....	82
4.7.8	Alle Schlüssel-Wert-Paare eines Dictionaries auslesen und durchlaufen	82
4.8	Tuple	83
4.8.1	Zugriff auf die einzelnen Elemente eines Tuples.....	85
4.8.2	Tuple und switch	86

4.9	Optional.....	89
4.9.1	Deklaration eines Optionals	89
4.9.2	Zugriff auf den Wert eines Optionals	90
4.9.3	Optional Binding	92
4.9.4	Implicitly Unwrapped Optional	94
4.9.5	Optional Chaining	96
4.9.6	Optional Chaining über mehrere Eigenschaften und Funktionen	100
4.10	Any und AnyObject.....	105
4.11	Type Alias	105
4.12	Value Type versus Reference Type	106
4.12.1	Reference Types auf Gleichheit prüfen	108
5	Funktionen	110
5.1	Funktionen mit Parametern.....	111
5.1.1	Argument Labels und Parameter Names.....	113
5.1.2	Default Value für Parameter	115
5.1.3	Variadic Parameter	116
5.1.4	In-Out-Parameter.....	117
5.2	Funktionen mit Rückgabewert	119
5.3	Function Types.....	121
5.3.1	Funktionen als Variablen und Konstanten	122
5.4	Verschachtelte Funktionen.....	123
5.5	Closures	124
5.5.1	Closures als Parameter von Funktionen	125
5.5.1.1	Implicit Return	128
5.5.1.2	Shorthand Argument Names.....	128
5.5.2	Trailing Closures	129
5.5.3	Autoclosures	130
6	Enumerations, Structures und Classes	132
6.1	Enumerations.....	132
6.1.1	Enumerations und switch	135
6.1.2	Associated Values.....	136
6.1.3	Raw Values	138

6.2	Structures	141
6.2.1	Erstellen von Structures und Instanzen.....	141
6.2.2	Eigenschaften und Funktionen	142
6.2.2.1	Memberwise Initializer	146
6.3	Classes	148
6.3.1	Erstellen von Klassen und Instanzen	149
6.3.2	Eigenschaften und Funktionen	149
6.4	Enumeration vs. Structure vs. Class	151
6.4.1	Gemeinsamkeiten und Unterschiede	152
6.4.2	Wann nimmt man was?	153
6.4.2.1	Enumeration	153
6.4.2.2	Structure	153
6.4.2.3	Class.....	154
6.5	self.....	154
7	Eigenschaften und Funktionen von Typen	157
7.1	Properties	157
7.1.1	Stored Property	158
7.1.2	Lazy Stored Property	161
7.1.2.1	Einsatzzweck von Lazy Stored Properties.....	164
7.1.3	Computed Property.....	164
7.1.4	Read-Only Computed Property.....	167
7.1.5	Property Observer.....	169
7.1.6	Property Wrapper	173
7.1.6.1	Standardwerte festlegen	175
7.1.6.2	Weitere Parameter ergänzen	176
7.1.6.3	Projected Value.....	178
7.1.7	Type Property.....	184
7.2	Globale und lokale Variablen.....	186
7.3	Methoden	189
7.3.1	Instance Methods.....	190
7.3.1.1	mutating.....	191
7.3.2	Type Methods	192
7.4	Subscripts	193

8	Initialisierung	199
8.1	Aufgabe der Initialisierung	200
8.2	Erstellen eigener Initializer	201
8.3	Initializer Delegation	207
8.3.1	Initializer Delegation bei Value Types	207
8.3.2	Initializer Delegation bei Reference Types	208
8.4	Failable Initializer	211
8.5	Required Initializer	214
8.6	Deinitialisierung	215
9	Vererbung	217
9.1	Überschreiben von Eigenschaften und Funktionen einer Klasse	220
9.2	Überschreiben von Eigenschaften und Funktionen einer Klasse verhindern	223
9.3	Zugriff auf die Superklasse	224
9.4	Initialisierung und Vererbung	225
9.4.1	Zwei-Phasen-Initialisierung	226
9.4.2	Überschreiben von Initializern	233
9.4.3	Vererbung von Initializern	235
9.4.4	Required Initializer	236
10	Speicherverwaltung mit ARC	237
10.1	Strong Reference Cycles	241
10.1.1	Weak References	243
10.1.2	Unowned References	247
10.1.3	Weak Reference vs. Unowned Reference	249
11	Weiterführende Sprachmerkmale von Swift	251
11.1	Nested Types	251
11.2	Extensions	253
11.2.1	Computed Properties	254
11.2.2	Methoden	254
11.2.3	Initializer	255
11.2.4	Subscripts	257
11.2.5	Nested Types	258
11.3	Protokolle	259

11.3.1	Deklaration von Eigenschaften und Funktionen	260
11.3.1.1	Properties	261
11.3.1.2	Methoden	263
11.3.1.3	Subscripts	265
11.3.1.4	Initializer	267
11.3.2	Der Typ eines Protokolls	271
11.3.3	Protokolle und Extensions	273
11.3.3.1	Bestehenden Typ mittels Extension um Protokoll ergänzen	273
11.3.3.2	Protokoll mittels Extension um neue Funktionen und Standardimplementierung erweitern	275
11.3.4	Vererbung in Protokollen	278
11.3.5	Class-only-Protokolle	279
11.3.6	Optionale Eigenschaften und Funktionen	280
11.3.6.1	Umgang mit Protokoll-Type	282
11.3.7	Protocol Composition	283
11.3.8	Delegation	284
11.3.9	Übersicht diverser vorhandener Protokolle	287
11.3.9.1	Hashable	287
11.3.9.2	Identifiable	289
11.4	Key-Path	289
12	Type Checking und Type Casting	293
12.1	Type Checking mit „is“	296
12.2	Type Casting mit „as“	297
13	Error Handling	300
13.1	Deklaration und Feuern eines Fehlers	301
13.2	Reaktion auf einen Fehler	304
13.2.1	Mögliche Fehler mittels do-catch auswerten	305
13.2.2	Mögliche Fehler in Optionals umwandeln	308
13.2.3	Mögliche Fehler weitergeben	309
13.2.4	Mögliche Fehler ignorieren	311

14	Generics	312
14.1	Generic Functions	313
14.2	Generic Types	316
14.3	Type Constraints	319
14.4	Associated Types	320
15	Nebenläufigkeit	324
15.1	Asynchronen Code schreiben und aufrufen	324
15.2	Mehrere asynchrone Funktionen parallel ausführen	328
15.3	Priorität von Tasks festlegen	329
15.4	Ergebnisse über Task zurückliefern und auswerten	330
15.5	Actors	337
15.6	Main Actor	339
16	Dateien und Interfaces	342
16.1	Modules und Source Files	342
16.2	Access Control	343
16.2.1	Access Level	343
16.2.1.1	Private Access	343
16.2.1.2	File-private Access	344
16.2.1.3	Internal Access	345
16.2.1.4	Public Access	345
16.2.1.5	Open Access	346
16.2.1.6	Zusammenfassung und Übersicht	346
16.2.2	Explizite und implizite Zuweisung eines Access Levels	347
16.2.3	Besonderheiten	349
16.2.3.1	Variablen und Konstanten	349
16.2.3.2	Tuples	349
16.2.3.3	Type Aliase	350
16.2.3.4	Funktionen	350
16.2.3.5	Enumerations	350
16.2.3.6	Properties	351
16.2.3.7	Subscripts	351
16.2.3.8	Getter und Setter	351
16.2.3.9	Initializer	352

16.2.3.10 Vererbung	353
16.2.3.11 Extensions	354
16.2.3.12 Protokolle	354
Teil II: Xcode	355
17 Grundlagen, Aufbau und Einstellungen von Xcode	357
17.1 Über Xcode	358
17.2 Arbeiten mit Xcode	359
17.2.1 Dateien und Formate eines Xcode-Projekts	359
17.2.1.1 Projekte	360
17.2.1.2 Targets	364
17.2.1.3 Schemes	365
17.2.2 Umgang mit Dateien und Ordnern	365
17.2.2.1 Dateien in Xcode hinzufügen	365
17.2.2.2 Ordner in Xcode hinzufügen	367
17.2.2.3 Bereits vorhandene Dateien einem Xcode-Projekt hinzufügen	367
17.2.2.4 Dateien und Ordner löschen	368
17.3 Der Aufbau von Xcode	369
17.3.1 Toolbar	369
17.3.2 Navigator	372
17.3.3 Editor	376
17.3.4 Inspectors	382
17.3.5 Debug Area	385
17.4 Einstellungen	386
17.4.1 General	386
17.4.2 Accounts	387
17.4.3 Behaviors	389
17.4.4 Navigation	390
17.4.5 Themes	390
17.4.6 Text Editing	391
17.4.7 Key Bindings	392
17.4.8 Source Control	393
17.4.9 Components	394
17.4.10 Locations	395

17.5	Projekteinstellungen	396
17.5.1	Einstellungen am Projekt.....	397
17.5.2	Einstellungen am Target	400
17.5.2.1	General.....	401
17.5.2.2	Signing & Capabilities.....	401
17.5.2.3	Resource Tags	403
17.5.2.4	Info	405
17.5.2.5	Build Settings	405
17.5.2.6	Build Phases	406
17.5.2.7	Build Rules.....	407
17.5.3	Einstellungen am Scheme	408
17.5.3.1	Neues Scheme erstellen	410
17.5.3.2	Schemes verwalten	411
17.5.3.3	Ausführungsmöglichkeiten eines Schemes.....	412
18	Dokumentation, Devices und Organizer.....	413
18.1	Dokumentation	413
18.1.1	Aufbau und Funktionsweise.....	414
18.1.2	Direktzugriff im Editor.....	418
18.2	Devices und Simulatoren	419
18.2.1	Simulatoren.....	422
18.2.2	Devices	423
18.3	Organizer	425
19	Debugging und Refactoring.....	427
19.1	Debugging.....	427
19.1.1	Einfache Konsolenausgaben.....	429
19.1.2	Logging	430
19.1.3	Arbeiten mit Breakpoints	434
19.1.3.1	Breakpoints setzen und aktivieren	434
19.1.3.2	Variables View einsetzen	435
19.1.3.3	Ausführung der App fortsetzen	436
19.1.3.4	Breakpoints konfigurieren.....	437
19.1.3.5	Breakpoints vollständig deaktivieren.....	438
19.1.3.6	Breakpoint Navigator	439
19.1.4	Debug Navigator	440

19.2	Refactoring	443
19.3	Instruments	445
20	Tipps und Tricks für das effiziente Arbeiten mit Xcode	449
20.1	Code Snippets	449
20.2	Open Quickly	452
20.3	Related Items	452
20.4	Navigation über die Jump Bar	454
20.5	MARK, TODO und FIXME	455
20.6	Shortcuts für den Navigator	456
20.7	Clean Build	456
20.8	Playgrounds	456
20.8.1	Was sind Playgrounds?	457
20.8.2	Code schreiben und testen	459
20.8.3	Dateien hinzufügen	464
20.8.4	Kommentare und Dokumentation	466
20.8.5	Swift Playgrounds-App	470
	Teil III: App-Entwicklung	473
21	Grundlagen der App-Entwicklung	475
21.1	Die Basis: SwiftUI	475
21.2	Bestandteile einer App	477
21.2.1	Umsetzung der Daten	478
21.2.2	Umsetzung der Ansichten	478
21.2.3	Weitere Frameworks	478
21.3	Die Syntax von SwiftUI	479
21.4	Aufbau einer App	480
21.5	Das View-Protokoll	481
21.6	Aktualisierung von Views mittels Status	482
21.7	Grundlagen des Status	484
21.8	Anpassung von Views mittels Modifier	487
21.9	Gruppierung von Views mittels Containern	490
21.10	Praxis: Unsere erste App	492
21.10.1	Bestandteile des neuen Projekts	497

21.10.2	Änderung des Textes.....	500
21.10.3	Einsatz der Preview.....	501
22	Views in SwiftUI.....	504
22.1	Textdarstellung und -bearbeitung	504
22.1.1	Text	504
22.1.1.1	Anpassung der Schriftart.....	506
22.1.1.2	Formatierung von Texten.....	512
22.1.1.3	Übersetzung von Texten.....	514
22.1.1.4	Umgang mit Datumsangaben	514
22.1.2	TextField.....	515
22.1.3	SecureField	520
22.1.4	TextEditor	521
22.1.4.1	Formatierung des Textes	522
22.2	Bilder	525
22.2.1	Image-Instanz erstellen	526
22.2.2	Größe einer Image-Instanz ändern.....	529
22.3	Schaltflächen	532
22.3.1	Button.....	532
22.3.2	EditButton	536
22.3.3	PasteButton	538
22.4	Werteauswahl	540
22.4.1	Toggle	540
22.4.2	Slider.....	545
22.4.3	Stepper	551
22.4.4	Picker	557
22.4.5	DatePicker	561
22.4.6	MultiDatePicker	566
22.4.7	ColorPicker	568
22.5	Weitere Views	572
22.5.1	Label.....	572
22.5.2	ProgressView	575
22.5.3	Gauge	578

23	View-Layout	583
23.1	Stacks	583
23.1.1	HStack	583
23.1.2	VStack	588
23.1.3	ZStack	592
23.1.4	Stacks verschachteln	594
23.1.5	Lazy Stacks	595
23.2	Listen	598
23.2.1	List	598
23.2.2	ForEach	622
23.3	Grids	635
23.3.1	Grid	635
23.3.2	LazyHGrid und LazyVGrid	637
23.4	Table	652
23.4.1	Zellen selektieren	654
23.4.2	Sortierung ändern	655
23.5	Container-Views	656
23.5.1	Form	656
23.5.2	Section	658
23.5.3	Group	662
23.5.4	GroupBox	665
23.5.5	ViewThatFits	668
23.6	Weitere Views	670
23.6.1	ScrollView	670
23.6.2	OutlineGroup	676
23.6.3	DisclosureGroup	681
23.6.4	Spacer	686
23.6.5	Divider	690
24	Navigation	692
24.1	NavigationStack und NavigationLink	692
24.1.1	Titel in Navigation-Bar setzen	697
24.1.2	Eigene View zur Darstellung eines NavigationLink nutzen	701
24.1.3	Anzeige einer Ziel-View auf Basis von Daten	703
24.1.4	Programmatische Steuerung des Navigation-Stacks	707

24.2	NavigationSplitView.....	711
24.2.1	Verknüpfung von NavigationSplitView und List	713
24.2.2	Sichtbarkeit der Spalten steuern	715
24.2.3	Breite der Spalten anpassen	717
24.2.4	Verhalten von NavigationSplitView unter den verschiedenen Apple-Plattformen.....	718
24.3	TabView.....	723
24.4	HSplitView und VSplitView	730
24.5	Funktionen zur Präsentation von Views.....	731
24.5.1	Sheet einblenden	731
24.5.2	View über gesamtes Fenster legen	742
24.5.3	Popover einblenden	747
25	Weitere View-Konfigurationen	754
25.1	Toolbar.....	754
25.2	Alerts	764
25.3	Confirmation Dialog.....	767
25.4	Farben	770
25.5	View-Events	771
25.6	Eigene Modifier umsetzen und aufrufen	772
26	Status	776
26.1	Property.....	778
26.2	State	781
26.3	Binding	782
26.4	ObservedObject.....	791
26.4.1	Datenmodell vorbereiten.....	793
26.4.2	Datenmodell in View einbinden.....	794
26.4.3	Auf Änderungen reagieren	797
26.5	StateObject	801
26.6	EnvironmentObject	803
26.7	@Observable-Makro und Bindable.....	809
26.8	Environment	811
26.9	SceneStorage	819
26.10	AppStorage	821
26.11	Source of Truth vs. Derived Value.....	823
26.12	Best Practices	827

27	Datenhaltung	832
27.1	UserDefaults	832
27.1.1	UserDefaults und SwiftUI	834
27.2	SwiftData	835
27.2.1	Grundlegende Funktionsweise von SwiftData	836
27.2.2	Erstellen des eigenen Datenmodells	836
27.2.3	Model-Container erzeugen	838
27.2.4	Elemente im ModelContext verwalten	839
27.2.5	Zusammenspiel mit der Preview	844
27.3	Core Data	846
27.3.1	Grundlegende Funktionsweise von Core Data	846
27.3.2	Grundlegende Elemente beim Einsatz von Core Data	847
27.3.2.1	NSPersistentStore	847
27.3.2.2	NSManagedObjectContext	848
27.3.2.3	NSManagedObject	848
27.3.2.4	NSManagedObjectContext	849
27.3.2.5	NSPersistentStoreCoordinator	849
27.3.2.6	NSPersistentContainer	849
27.3.3	Einen Core Data Stack erstellen	850
27.3.4	Ein Managed Object Model erstellen	852
27.3.4.1	Entitäten und Eigenschaften erstellen	853
27.3.4.2	Relationships zwischen Entitäten	855
27.3.4.3	Entitäten und Attribute als Managed Objects	860
27.3.5	Grundlegende Core-Data-Operationen	864
27.3.5.1	Managed Object erstellen und konfigurieren	865
27.3.5.2	Gespeicherte Managed Objects laden	865
27.3.5.3	Bestehende Managed Objects löschen	866
27.3.6	Core Data mit SwiftUI	867
27.3.6.1	NSManagedObject als Status	867
27.3.6.2	Zugriff auf Managed Object Context	868
27.3.6.3	Auslesen persistent gespeicherter Elemente	870
28	Weitere Projektkonfigurationen	874
28.1	Cross-Platform-Entwicklung	874
28.1.1	Neue Targets hinzufügen	875

28.1.2	Target-Zuweisung	878
28.1.3	Plattform im Code prüfen	879
28.1.4	Funktionen auf Verfügbarkeit prüfen	880
28.2	Mehrsprachigkeit	883
28.2.1	Grundlagen	883
28.2.1.1	Erstellen eines String Catalogs	883
28.2.1.2	Übersetzungen im Code kennzeichnen	886
28.2.1.3	Auf Übersetzungen im Code zugreifen.....	887
28.2.1.4	Übersetzungen mit dynamischem Parameter konfigurieren	887
28.2.2	Verschiedene Sprachen einer App testen.....	891
28.3	Asset Catalogs.....	892
29	Preview und Library	896
29.1	Preview	896
29.1.1	Funktionsweise der Preview.....	898
29.1.2	Konfiguration der Preview	898
29.1.3	Preview ausführen und anhalten	900
29.2	Library.....	902
29.3	Attributes Inspector	903
	Teil IV: Source Control und Testing	905
30	Source Control	907
30.1	Basisfunktionen und -begriffe der Source Control.....	907
30.2	Source Control in Xcode	909
30.2.1	Bestehendes Projekt klonen	911
30.2.2	Lokale Änderungen committen	913
30.2.3	Lokale Änderungen verwerfen	916
30.2.4	Pull und Push	916
30.2.5	Aktuelle Branches vom Repository laden	917
30.2.6	Git-Repository mit neuem Xcode-Projekt erzeugen.....	917
30.2.7	Optische Source-Control-Hervorhebungen im Editor.....	918
30.2.8	Zugriff auf GitHub, GitLab und Bitbucket	920
30.3	Source Control Navigator	921
30.4	Code Review-Mode	922

31	Testing	925
31.1	Unit-Tests	925
31.1.1	Unit-Tests mit Swift Testing	930
31.1.2	Unit-Tests mit XCTest	931
31.1.3	Neue Test-Datei erstellen	936
31.1.4	Ausführen von Unit-Tests	938
31.2	Performancetests	941
31.3	UI-Tests	942
31.3.1	Klassen für UI-Tests	944
31.3.2	Aufbau von UI-Test-Klassen	947
31.3.3	Automatisches Erstellen von UI-Tests	947
31.3.4	Einsatz von UI-Tests	948
Teil V: Veröffentlichung von Apps		949
32	Veröffentlichung im App Store	951
32.1	Das Apple Developer Portal	953
32.1.1	Zertifikate, App IDs und Provisioning Profiles	956
32.1.1.1	Erstellen von Entwicklerzertifikaten	958
32.1.1.2	Erstellen von App IDs	962
32.1.1.3	Hinzufügen von Devices	964
32.1.1.4	Erstellen von Provisioning Profiles	966
32.1.2	Code Signing	972
32.1.2.1	Automatic Code Signing	972
32.1.2.2	Manual Code Signing	974
32.1.2.3	Code Signing in den Build Settings	975
32.2	App Store Connect	976
32.2.1	Apps für den App Store vorbereiten und verwalten	978
32.2.2	Apps erstellen, hochladen und einreichen	982
32.3	App Store Review Guidelines	985
33	Das Business Model für Ihre App	987
33.1	Geschäftsmodelle	987
33.1.1	Free Model	988
33.1.2	Freemium Model	988
33.1.3	Subscription Model	988

33.1.4	Paid Model.....	989
33.1.5	Paymium Model.....	990
33.2	App Bundles.....	990
33.3	Veröffentlichung außerhalb des App Store.....	992
33.3.1	Das Apple Developer Enterprise Program.....	993
34	TestFlight.....	995
34.1	TestFlight in App Store Connect.....	995
34.2	TestFlight im App Store.....	997
	Index.....	1001