

The background of the cover is a dark, textured surface with a subtle grid pattern. Scattered across this background are numerous 3D cubes and 2D squares in various colors including red, purple, yellow, green, and blue. Some of these shapes are solid and have a slight gradient, while others are transparent or outlined. They appear to be floating or falling, creating a sense of dynamic movement.

Windows Phone 7.5

Einstieg in die professionelle App-Entwicklung

Mario FraiB

Mit Ausblick auf Windows
Phone 8.0

Mario Fraiß

Windows Phone 7.5

Einstieg in die professionelle App-Entwicklung

Mario Frai
Windows Phone 7.5
ISBN: 978-3-86802-248-3

© 2012 entwickler.press
Ein Imprint der Software & Support Media GmbH

Bibliografische Information Der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen
Nationalbibliografie; detaillierte bibliografische Daten sind im Internet ber
<http://dnb.ddb.de> abrufbar.

Ihr Kontakt zum Verlag und Lektorat:
Software & Support Media GmbH
entwickler.press
Darmstdter Landstr. 108
60598 Frankfurt am Main
Tel.: +49 (0)69 630089-0
Fax: +49 (0)69 930089-89
lektorat@entwickler-press.de
<http://www.entwickler-press.de>

Lektorat: Stefan Knichel
Korrektorat: Barbara Decker
Satz: Laura Acker

Alle Rechte, auch fr bersetzungen, sind vorbehalten. Reproduktion jeglicher Art (Fotokopie, Nachdruck, Mikrofilm, Erfassung auf elektronischen Datentrgern oder anderen Verfahren) nur mit schriftlicher Genehmigung des Verlags. Jegliche Haftung fr die Richtigkeit des gesamten Werks kann, trotz sorgfltiger Prfung durch Autor und Verlag, nicht bernommen werden. Die im Buch genannten Produkte, Warenzeichen und Firmennamen sind in der Regel durch deren Inhaber geschtzt.

Für meine Eltern *Brigitte und Siegfried*,
meine Freundin *Rebekka*
und all meine *Freunde*.

Ihr seid es, die mich immer wieder aufs Neue zu Höchstleistungen motivieren und
jederzeit zu mir stehen.

Inhaltsverzeichnis

Vorwort	13
1 Einleitung	15
1.1 Apps für alle	15
1.2 Was Sie in diesem Buch erwartet	17
1.2.1 Website zum Buch	17
1.2.2 Benötigte Software	18
1.3 Auf die Plätze. Fertig. App!	20
2 Windows-Phone-Grundlagen	21
2.1 Windows-Phone-UI-Konzept „Metro“	21
2.2 Windows-Phone-Architektur	22
2.2.1 Hardware Foundation	23
2.2.2 Softwarearchitektur	24
2.2.3 App Model	25
2.2.4 UI Model	27
2.2.5 Cloud Integration Services	28
2.3 Die Entwicklungswerkzeuge	29
2.3.1 Entwicklungsumgebung	29
2.3.2 Windows Phone Emulator	31
2.4 Zusammenfassung	34
3 Grundlagen der Entwicklung	35
3.1 Hello Phone	35
3.2 Anwendungsarchitektur	39
3.2.1 Spezifische Assemblies für Windows Phone	40
3.2.2 Anwendungsgrundgerüst	40
3.2.3 Anwendungsmanifest	41
3.2.4 Windows Phone Application-Lifecycle	45
3.3 Zusammenfassung	46

4	Projektvorlagen	47
4.1	Portrait-Seite	48
4.2	Pivot-Seite	54
4.3	Panoramaseite	57
4.4	Seitenausrichtung	62
4.5	Zusammenfassung	64
5	Windows-Phone-Steuerelemente	65
5.1	Basis-Steuerelemente	65
5.1.1	TextBlock	65
5.1.2	TextBox	65
5.1.3	CheckBox	66
5.1.4	RadioButton	66
5.1.5	PasswordBox	66
5.2	Standardformular erstellen	67
5.3	Erweiterte Steuerelemente	71
5.3.1	HyperLinkButton	71
5.3.2	ProgressBar	71
5.3.3	ScrollViewer	71
5.3.4	Image	72
5.3.5	Slider	72
5.3.6	Rectangle und Ellipse	72
5.4	Verwendung der erweiterten Steuerelemente	73
5.5	Zusammenfassung	78
6	Anwendungsmenüs und Styling	79
6.1	ApplicationBar	79
6.1.1	MenuItem und ApplicationBarIconButton	79
6.1.2	ApplicationBar verwenden	80
6.1.3	Grafische Elemente der ApplicationBar	83
6.2	Windows Phone Themes und Styling	85
6.2.1	Hintergrund	85
6.2.2	Akzentfarbe	86
6.2.3	Themenressourcen	90
6.3	Zusammenfassung	91

7	Anwendungslayout	93
7.1	Grid-Layouts	97
7.1.1	Das Grid-Steuerelement	97
7.1.2	Dynamische Zeilen und Spalten	98
7.1.3	Automatische Zeilenhöhe und Spaltenbreite	99
7.1.4	Inhaltspositionierung innerhalb eines Grid	99
7.1.5	Inhalt über mehrere Zeilen und Spalten	100
7.1.6	Taschenrechner-Layout mithilfe eines Grid	101
7.2	Stack-Panel-Layout	105
7.2.1	Das Stack-Panel-Steuerelement	105
7.2.2	Layout einer Quiz-Seite durch mehrere Stack-Panels	106
7.3	Canvas-Layout	108
7.3.1	Das Canvas-Steuerelement	108
7.3.2	Positionierung innerhalb eines Canvas	109
7.3.3	Beispiel eines Canvas-Layouts	110
7.4	Zusammenfassung	111
8	Animationen	113
8.1	Storyboard	113
8.2	Animationszustand steuern	114
8.3	Zusammenfassung	116
9	Multimedia	117
9.1	Wiedergabe von Audiodateien	117
9.2	Streaming von Videos	121
9.2.1	Media Element-Steuerelement	121
9.2.2	Videostream wiedergeben	122
9.3	Background Audio-Wiedergabe	125
9.3.1	Background Audio-Agent	126
9.3.2	Steuerung der Hintergrund-Wiedergabe	131
9.3.3	Abruf von Titelinformationen	133
9.4	Zusammenfassung	136

10 Windows Phone als Entwicklungsgerät registrieren	137
10.1 Windows Live ID anlegen	137
10.2 Microsoft App Hub	139
10.2.1 Entwicklerregistrierung	139
10.2.2 Kostenpflichtige Mitgliedschaft	140
10.3 Microsoft Zune installieren	141
10.4 Entwicklungsgerät registrieren	142
10.5 Zusammenfassung	145
11 Hardwarefunktionen nutzen	147
11.1 Integrierte Kamera	147
11.1.1 Kamera-Livebild abrufen	148
11.1.2 Ausführung auf dem Entwicklungsgerät	149
11.1.3 Autofokus	150
11.1.4 Kamera-Aufnahme	151
11.1.5 Benutzerdefinierten Sucher darstellen	154
11.2 GPS- und Location-Services	155
11.2.1 Erforderliche Bibliotheken	156
11.2.2 GPS-Sensor nutzen	156
11.3 Hardwaresensoren	163
11.3.1 Kompass	165
11.3.2 Gyroskop	167
11.3.3 Beschleunigungssensor	168
11.3.4 Motion API	169
11.4 Zusammenfassung	171
12 Erweiterte API-Funktionen nutzen	173
12.1 Gerätezustand auslesen	173
12.2 Background Notification Services	174
12.2.1 Erinnerung	174
12.2.2 Alarm	175
12.3 Zugriff auf Kontakte (Contacts API)	176
12.3.1 Kontaktdatenbank öffnen	177
12.3.2 Kontakte anzeigen und filtern	177
12.3.3 Kontakteigenschaften	178
12.4 Zugriff auf Kalender (Calendar API)	178

12.5 Netz- und Netzwerkinformationen auslesen	180
12.5.1 Mobilnetz	180
12.5.2 Netzwerk	180
12.6 Standardaufgaben (Tasks)	181
12.6.1 Launcher-Tasks	182
12.6.2 Chooser-Tasks	184
12.6.3 Spezielle Tasks	186
12.7 Live-Tiles	188
12.7.1 Primäres Tile aktualisieren	190
12.7.2 Zusätzliche Tiles	191
12.8 Push-Notifications	192
12.8.1 Toast-Push-Notifications	193
12.8.2 Tile-Push-Notifications	198
12.9 Background-Agents	199
12.9.1 Periodic-Agent	199
12.9.2 Background-Transfer	202
12.10 Datenbanken	205
12.10.1 Data-Context und Entities	205
12.10.2 Datenbanknutzung	206
12.11 Zusammenfassung	209
13 Anwendungen im Marketplace veröffentlichen	211
13.1 Windows-Phone-Leistungsanalyse (Profiling)	211
13.1.1 Analyse der Ausführungsebene	212
13.1.2 Analyse der Speicherebene	216
13.1.3 Analyse der UI-Performance	218
13.2 Windows-Marketplace-Testkit	220
13.2.1 Anwendungsdetails	221
13.2.2 Automatisierte Tests	222
13.2.3 Überwachte Tests	223
13.2.4 Manuelle Tests	224
13.3 Windows-Marketplace-Einreichung	224
13.4 Zusammenfassung	232

14 Ausblick	233
14.1 Entwicklung des mobilen Marktes	233
14.2 Mobile Anwendungen: heute und morgen	234
14.3 Grenze zwischen Mobile und Desktop	234
14.4 Tanze Tango und fliege zum Apollo	235
14.4.1 Minor-Update Tango	235
14.4.2 Windows Phone 8: Codename Apollo	236
14.4.3 Fazit	238
Stichwortverzeichnis	241

Vorwort

Betrachtet man den heutigen Markt der sogenannten Smartphones, so stellt man als Entwickler schnell fest, dass es keinen anderen Bereich der Softwareindustrie gibt, der sich so rapide verändert wie jener der App-Entwicklung. In diesem Kontext spielt es auch keine Rolle, ob man sich auf Apple iOS, Google Android, Nokia Symbian oder auch Windows Phone konzentriert, denn eines haben all diese Plattformen gemeinsam: Sie werden kontinuierlich und sehr rasant erweitert und mit immer neuen Funktionen ausgestattet.

Gerade deswegen bietet der Smartphone-Markt eine interessante Möglichkeit für Softwarehersteller, entsprechende Produkte zu platzieren. Das Besondere dabei ist aber, dass auch Einzelpersonen Anwendungen auf einfachstem Wege über die zahlreichen App-Stores verteilen können und im Idealfall ein Millionenpublikum erreichen – zweifelsohne wäre dies ohne die Infrastruktur, die durch Apple, Google und andere Anbieter bereitgestellt wird, nicht ohne erheblichen Mehraufwand möglich. Erfolgs-Apps wie *Angry Birds* verzeichnen aktuell mehr als 700 Millionen Downloads (März 2012) und zählen zu den erfolgreichsten Apps, die je entwickelt wurden. Dieses Beispiel zeigt, dass auch ein simples Konzept mit kontinuierlich aktualisierten Inhalten ausreichen kann, um User zu begeistern.

Seit Ende 2010 ist auch Microsoft auf dem Smartphone-Markt vertreten. Das Unternehmen präsentierte im Oktober 2010 die mobile Plattform Windows Phone, eine auf Multimedia- und Multi-Touch-Anwendungen ausgerichtete Technologie. Damit positionierte sich auch Microsoft auf dem mobilen Sektor und stellte mit der damaligen Version Windows Phone 7.0 eine innovative Möglichkeit bereit, auf Basis von Microsoft Silverlight und dem .NET Framework professionelle Smartphone-Anwendungen zu realisieren.

Für wen wurde dieses Buch geschrieben?

Dieses Buch richtet sich in erster Linie an Software-Entwickler und Entwicklerinnen, die sich mit der App-Entwicklung für Microsoft Windows Phone auseinandersetzen möchten. Die Inhalte wurden daher entsprechend für Einsteiger konzipiert und stellen eine optimale fachliche Grundlage dar, um mit der Entwicklung eigener Windows Phone Anwendungen zu beginnen. Zusätzlich vermittelt es Hintergrundwissen rund um das Thema der App-Entwicklung, veranschaulicht anhand vieler unterschiedlicher Beispielprojekte, welche Möglichkeiten sich im Rahmen der App-Entwicklung für Windows Phone bieten, und zeigt, wie Sie Ihre Apps über den Windows Marketplace (App Hub) veröffentlichen und verkaufen können. Um die zahlreichen Beispiele erfolgreich nachvollziehen zu können, werden lediglich Grundkenntnisse der Programmierung unter der .NET Plattform vorausgesetzt. *Ich wünsche Ihnen viel Vergnügen beim Lesen und Durcharbeiten dieses Buches und möchte mich abschließend herzlich für den Erwerb bedanken.*

Über den Autor



Mario Fraiß, DI (FH), geboren am 21. Juni 1985 in Graz, ist Gründer und Inhaber der FRAISS – IT Consulting und Mediendesign, ein auf Softwareentwicklung und Technologie-Innovation spezialisiertes Unternehmen mit Sitz in Graz, Österreich. Ergänzend dazu arbeitet er seit mehr als zehn Jahren als freiberuflicher Consultant, Trainer und Autor mit dem Schwerpunkt Softwareentwicklung unter .NET, bietet aber dazu auch eine Expertise im Java-Umfeld und in weiteren Programmiersprachen. Er betreut mittelgroße bis große Unternehmen und war unter anderem auch bereits als Technologieberater für Microsoft Österreich tätig. Sie erreichen ihn via Blog <http://www.mariofraiss.com> sowie unter <http://www.fraiss.IT>

Anmerkungen und Anregungen zu den Inhalten und Buchbeispielen sind erbeten und erwünscht. Sie können diese gerne per E-Mail an m.fraiss@fraiss.at senden.

Danksagung

An dieser Stelle möchte ich mich bei all jenen Personen bedanken, die mich im Rahmen der Bucherstellung unterstützt haben, und gleichzeitig bei jenen entschuldigen, die ich aufgrund dieses Projektes vernachlässigen musste. Gleichmaßen großen Dank aussprechen möchte ich auch gegenüber meinen beiden Lektoren Sebastian Burkart und Stefan Knichel, die während der Entstehung dieses Buches stets all meine Fragen beantwortet haben, sowie auch dem beteiligten entwickler.press-Verlagsteam.

1

Einleitung

Bevor es mit der tatsächlichen Entwicklung Ihrer ersten App für Windows Phone losgeht, vermittelt Ihnen dieses Kapitel noch einige Basisinformationen zum Thema der App-Entwicklung. Dabei werden die bekanntesten Technologien kurz beschrieben und aufgezeigt, zwischen welchen App-Typen unterschieden werden kann.

1.1 Apps für alle

Unter einer sogenannten App versteht man im Allgemeinen eine Applikation bzw. Anwendung, die für aktuelle Smartphones bzw. auch Tablet-Systeme entwickelt wurde. Ein ebenfalls wichtiges Merkmal von Apps ist, dass diese über einen, meist in das jeweilige Betriebssystem integrierten Online-Shop bezogen werden können. In der Zwischenzeit gibt es Millionen verschiedenster Apps für die unterschiedlichen Betriebssysteme. Man könnte durchaus behaupten, es gibt eine App für alle erdenklichen Aufgabenstellungen und Zielgruppen.

Smartphonemarkt

Umso relevanter ist es, sich vor der Entwicklung einer App zu überlegen, welche Technologien in erster Linie unterstützt werden sollen. Dazu ist es sinnvoll, sich zu Beginn mit dem Smartphonemarkt und den jeweiligen Anteilen der einzelnen Hersteller auseinanderzusetzen:

Hersteller	Endgeräte Q3/2011
Nokia	105353,5
Samsung	78612,2
LG	21014,6
Apple	17295,3
ZTE	14107,8
RIM	12701,1
HTC	12099,9
Motorola	11182,7
Sony Ericsson	8475,9
Huawei Device	10668,2
Andere Hersteller	148990,9

Tabelle 1.1: Verkaufte Endgeräte nach Hersteller Q3/2011 [Quelle: Gartner, <http://www.gartner.com/it/page.jsp?id=1848514> vom 22.02.2012]

Technologie	Marktanteil Q3/2011 in %
Android	25,3
Symbian	36,3
iOS	16,6
Blackberry	15,4
Bada	1,1
Windows Phone	2,7
Weitere OS	2,5

Tabelle 1.2: Betriebssystem-Marktanteile Q3/2011 [Quelle: Gartner, <http://www.gartner.com/it/page.jsp?id=1848514> vom 22.02.2012]

Wie die beiden oben angeführten Tabellen zeigen, sollte man die Anwendung für das Betriebssystem Android oder Symbian entwickeln, sofern man die größte Benutzergruppen erreichen möchte (Stand Februar 2012). Natürlich hängt dies aber in der Praxis davon ab, welche Zielgruppe eine Anwendung tatsächlich verfolgt.

HINWEIS: Evaluieren Sie den Markt, den Sie mit Ihrer Anwendung erreichen möchten, und recherchieren Sie, welche Technologien von Ihrer Zielgruppe primär genutzt werden.

1.2 Was Sie in diesem Buch erwartet

Dieses Buch beschäftigt sich vorrangig mit der Entwicklung von Anwendungen unter Microsofts mobilem Betriebssystem Windows Phone 7 bzw. Windows Phone 7.5 Mango (verfügbar seit September 2011), den beiden direkten Nachfolgern von Windows Mobile, das in der Zwischenzeit in Windows Phone 6.5 umbenannt wurde. Die Technologie, mit der Sie Lösungen für Windows Phone 7.x entwickeln können, nennt sich Microsoft Silverlight und liegt seit Dezember 2011 in der aktuellen Version 5.0 vor.



Abbildung 1.1 Offizielles Windows Phone Logo © Microsoft

HINWEIS: Nützliche Informationen rund um Microsoft Silverlight finden Sie unter <http://www.silverlight.net/>. Zum jetzigen Zeitpunkt (Februar 2012) sind noch keine Informationen über einen möglichen Erscheinungstermin des nächsten Release verfügbar.

Im Laufe der unterschiedlichen Kapitel werden Sie Grundlegendes über das Betriebssystem Windows Phone 7 lernen und wie Sie mithilfe der unterschiedlichen Steuerelemente Ihre eigenen Benutzeroberflächen erzeugen. Die erworbenen Kenntnisse und Informationen werden Sie im Rahmen einiger praxisnaher Beispiele anwenden und Ihre ersten eigenen Windows Phone Apps entwickeln. Den Abschluss bildet ein kurzer Ausblick in die Zukunft und was die Entwicklergemeinde in diesem Zusammenhang mit Windows Phone erwartet.

1.2.1 Website zum Buch

Sobald dieses Buch auf dem Markt erhältlich ist, wird auch die entsprechende Website www.phonedevlopment.at freigeschaltet. Dort ist es möglich, bereits ergänzte Errata zu beziehen und mit dem Autor in Kontakt zu treten. Möglicherweise werden von Zeit zu Zeit auch einige Ergänzungen bereitgestellt und interessante Links zum Thema empfohlen. Eine Kontaktaufnahme mit dem Autor ist jederzeit über die Website sowie über die E-Mail-Adresse phonedevlopment@fraiss.at möglich.

1.2.2 Benötigte Software

Um die Funktionalität der Beispiele aus dem Buch zu garantieren, benötigen Sie verschiedene Softwarekomponenten, die Sie in diesem Abschnitt gelistet finden.

- Microsoft .NET Framework 4.0
- Microsoft Windows Phone SDK 7.1
- Microsoft Visual Studio 2010 Express für Windows Phone (in Windows Phone SDK 7.1 enthalten)

Microsoft .NET Framework 4.0

Es ist erforderlich, dass Sie das *Microsoft .NET Framework 4.0* auf Ihrem System installieren. Sie können dieses unter der Adresse <http://www.microsoft.com/germany/net/downloadFile.aspx?file=.netframework> herunterladen.

Microsoft Windows Phone SDK 7.1

Ebenfalls erforderlich ist die Installation der *Microsoft Windows Phone Developer Tools 7.1*, die Sie unter <http://msdn.microsoft.com/de-de/windowsphone/hh442445> beziehen können. Diese stellen Ihnen ebenfalls eine auf *Microsoft Visual Studio 2010* basierende Entwicklungsumgebung inklusive Geräte-Emulator zur Verfügung, um Ihre Anwendungen zu testen.

Das Windows Phone SDK 7.1 stellt Ihnen eine Vielzahl nützlicher Tools und Software zur Verfügung, um für die Windows-Phone-Plattform entwickeln zu können. Im Detail sind dies folgende Komponenten:

- Microsoft Visual Studio 2010 Express for Windows Phone
- Windows Phone Emulator
- Windows Phone SDK 7.1 Assemblies
- Silverlight 4 SDK und DRT
- Erweiterungen für XNA Game Studio 4.0
- Microsoft Expression Blend SDK für Windows Phone 7
- Microsoft Expression Blend SDK für Windows Phone OS 7.1
- WCF Data Services Client für Windows Phone
- Microsoft Advertising SDK für Windows Phone

Um optimal mit der Entwicklungsumgebung und den verschiedenen Werkzeugen arbeiten zu können, empfiehlt es sich, folgende Mindest-Systemanforderungen zu erfüllen:

- Microsoft Windows Vista inkl. SP2 oder Windows 7 (jeweils x86 und x64) – alle Editionen außer der Starter Edition
- 4GB Festplattenspeicher
- 3GB Arbeitsspeicher
- Für Windows-Phone-Emulator: Grafikkarte mit Kompatibilität zu DirectX 10 oder höher

PROFITIPP: Sollte bereits *Visual Studio 2010* inkl. *Visual Studio 2010 Service Pack 1* auf Ihrem System installiert sein, so empfehle ich Ihnen die Deinstallation all Ihrer Visual-Studio-bezogenen Produkte bevor Sie mit dem eigentlichen Aufsetzen der Entwicklungsumgebung für Windows Phone beginnen. So vermeiden Sie Inkompatibilitätskonflikte.

ACHTUNG: Gehen Sie bei der Installation der erforderlichen Komponenten unbedingt in der nachfolgend gezeigten Reihenfolge vor, da es ansonsten zu Fehlfunktionalitäten im Rahmen der einzelnen Anwendungen kommen kann und etwaige Vorlagen für Windows Phone innerhalb von *Visual Studio 2010 Express für Windows Phone* nicht funktionieren.

- (optional) beliebige Visual Studio 2010
- Version Windows Phone SDK 7.1
- Visual Studio 2010 - Service Pack 1

Sollten Sie auf Ihrem System bereits eine Vollversion von Visual Studio 2010 installiert haben, integriert sich das Windows Phone SDK in diese Version und Sie erhalten keinen separaten Eintrag für *Visual Studio 2010 Express für Windows Phone* in Ihrem Startmenü. Sie können, falls gewünscht, *Visual Studio 2010 Express für Windows Phone*, über folgenden Anwendungspfad öffnen: `C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE\VPDEpress.exe`

1.3 Auf die Plätze. Fertig. App!

Jetzt bleibt mir nur noch, Ihnen viel Spaß und Erfolg beim Durcharbeiten der einzelnen Buchbeispiele und beim Erstellen Ihrer ersten Windows Phone Apps zu wünschen.

HINWEIS: Im Rahmen der Beispiele werden unterschiedliche freie Bibliotheken oder Softwareprodukte von Dritten eingesetzt. Die jeweiligen Bezugsmöglichkeiten werden innerhalb der einzelnen Kapitel angeführt.

HINWEIS: Sämtliche Beispiele, die Sie im Buch erarbeiten, sind als Fortbildungs- und Entwicklungsbeispiele gedacht und nicht für den produktiven Einsatz konzipiert worden. In erster Linie geht es im vorliegenden Buch darum, die Anwendung von Funktionen und das Implementieren von Window-Phone-Features zu erlernen.

2

Windows-Phone-Grundlagen

Dieses Kapitel vermittelt Ihnen das notwendige Basiswissen für das Betriebssystem Windows Phone 7.5. Behandelt werden dabei zum einen das Oberflächenkonzept *Metro* und zum anderen die grundlegende Windows Phone-Hard- und Softwarearchitektur.

Es ist wichtig, dass Sie die Arbeitsweise von Windows Phone und einer Windows Phone App verstehen, um auf dieses Wissen aufbauen und eigene Apps realisieren zu können.

2.1 Windows-Phone-UI-Konzept „Metro“



Abbildung 2.1: Windows-Phone-Startseite (© Microsoft)



Abbildung 2.2: Windows Phone People-Hub (© Microsoft)

Microsoft stellt mit dem Betriebssystem Windows Phone ein völlig neuartiges User-Interface-Konzept bereit, das den Namen „Metro“ trägt. Dabei steht Metro nicht nur für einen Codenamen, sondern vielmehr für eine eigene Designphilosophie, die sich durch das gesamte Windows-Phone-Konzept zieht. Angelehnt wurde Metro an eine visuelle Sprache, die sich zum Beispiel in Flughafen- und U-Bahn-Leit- und Informationssystemen wiederfinden lässt.

Bei der Realisierung des Metro-Designs wurden folgende primären Paradigmen verfolgt:

- Sauberkeit, Einfachheit und Schnelligkeit
- Fokussierung von Inhalt
- Nahtlose Integration von Hard- und Software
- „World-Class“-Motion User Experience
- Personalisiert, aktiv und live

Sie sollten die oben gelisteten Grundgedanken im Rahmen Ihres Designprozesses stets im Hinterkopf behalten, da die Beliebtheit Ihrer Anwendungen unter anderem auch von der Art und Weise der Integration in das Betriebssystem und von der Einfachheit der Benutzeroberfläche abhängig sein kann.

2.2 Windows-Phone-Architektur

Die Architektur des gesamten Windows-Phone-Konzeptes setzt sich aus insgesamt fünf Teilen zusammen, dazu zählen:

- Hardware Foundation
- Software Architecture
- App Model
- UI Model
- Cloud Integration Services

Die nachfolgenden Abschnitte beschäftigen sich mit den einzelnen Architekturebenen und zeigen deren Verantwortungsbereiche.

2.2.1 Hardware Foundation

Unter dem Begriff *Hardware Foundation* fasst Microsoft alle hardwareseitigen Anforderungen zusammen, die an sämtliche Windows Phone Geräte gestellt werden und von diesen erfüllt werden müssen. Im Wesentlichen sind dies die folgenden Eigenschaften:

Feature	Mindestanforderung
Bildschirmauflösung	WVGA (800 x 480)
Hardwareeschaltflächen	3 Stück: Zurück, Start, Suchen
Touchscreen	Kapazitiv mit vier oder mehr Kontaktpunkten
Sensoren	A-GPS Beschleunigung Kompass Licht Annäherung Gyrometer (ab Mango)
Kamera	5 MP oder mehr
Multimedia	Unterstützung diverser Multimedia-Features Codec-Beschleunigung
Speicher	256 MB RAM oder mehr 8 GB Flash-Speicher oder mehr
GPU	Microsoft DirectX 9 Hardwarebeschleunigung
CPU	Qualcomm MSM8x55 mit 800 MHz oder mehr Qualcomm MSM7x30 (ab Mango)

Tabelle 2.1: Übersicht über die Windows-Phone-Hardwareanforderungen

Ergänzend zur technischen Spezifikation sind auch Aussehen sowie Look & Feel aller Windows Phone Geräte einigermaßen standardisiert. Das zeigt die nachfolgende Abbildung:



Abbildung 2.3: Windows-Phone-Hardwaredesign

Nach der Hardware Foundation folgt als nächste Ebene im Windows-Phone-Architektur-Stack die Softwareebene, zusammengefasst als *Software Architecture*.

2.2.2 Softwarearchitektur

Innerhalb der Architekturstufe „Software Architecture“ wurde spezifiziert, wie alle Softwareanwendungen im Rahmen von Windows Phone funktionieren. Die unten stehende Abbildung 2.4 zeigt vereinfacht, aus welchen Kernkomponenten sich das Windows-Phone-Softwarekonzept zusammensetzt:

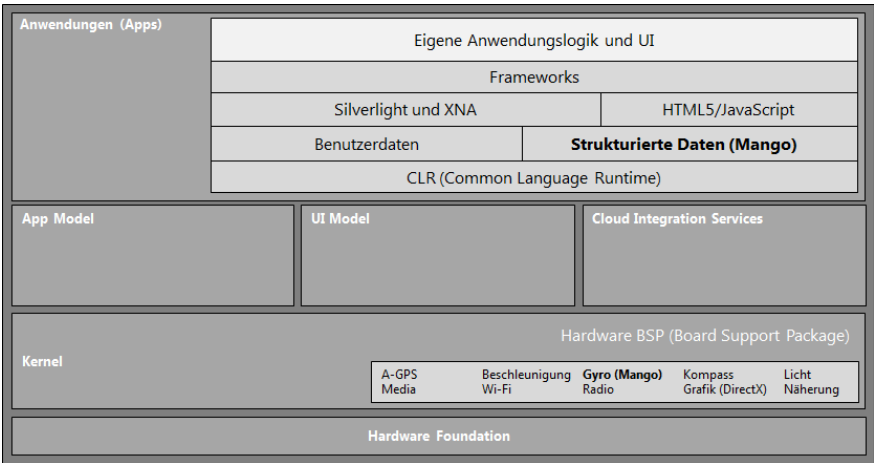


Abbildung 2.4: Überblick über die Windows-Phone-Softwarearchitektur (Quelle: Microsoft)

HINWEIS: Die Anwendungen, die Sie implementieren, also Ihre Apps, werden auf oberster Ebene ausgeführt. Dadurch ist es möglich, dass diese Zugriff auf die meisten der untergeordneten Features erhalten, sofern er nicht aus Sicherheitsgründen hardwareseitig untersagt wurde.

Die nächste Ebene der Windows Phone Architektur ist das App Model, mit dem wir uns im nachfolgenden Abschnitt beschäftigen.

2.2.3 App Model

Unter dem sogenannten *App Model* verstehen wir im Rahmen von Windows Phone den Aufbau, also die grundlegende Struktur einer jeden Windows Phone Anwendung. Dementsprechend hat das Verständnis des App Models eine hohe Relevanz, um erfolgreich Windows Phone Apps entwickeln zu können. Das App Model dient dazu, folgende Aufgaben abzudecken:

- Anwendungsmanagement
- Lizenzierung
- Anwendungsisolation
- Softwareaktualisierungen
- Gemeinsame Datennutzung (Sharing)

Durch das App Model wird auch die Ausführung aller Windows-Phone-Anwendungen gesteuert. Diese befolgt immer die nachfolgend gelisteten Regeln:

- Bereitstellung eines Host-Prozesses durch das System
- Bereitstellung einer Sandbox für den Host-Prozess, basierend auf festgelegten Berechtigungen
- Anwendungsausführung wird innerhalb eines isolierten, mit weniger Rechten versehenen Host-Prozesses ausgeführt
- Offenlegung des gesamten Anwendungscodes, sodass dieser hinsichtlich der Common Language Specification (CLS) überprüfbar bleibt, wodurch sich die Auswirkungen möglicher Angriffe minimieren lassen

Abbildung 2.5 zeigt die Zusammensetzung des Windows Phone App Models: