

**Berufsbegleitender Studiengang zum Master of Arts (M.A.)
IT Management**

2. Semester

**Seminararbeit Software-Engineering, IT-Projekt- und
Qualitätsmanagement**

**„Mobile Applikationen - Softwareentwicklung für den Android
Market: Die Entwicklung und Vermarktung von Android Apps“**

Betreuer: Professor Dr. Gregor Sandhaus

Autor: Andre Tschersich

Matrikelnummer: 269765

Marl, den 28.06.2011

Inhaltsverzeichnis

1. Einleitung – Ziel der Arbeit	2
2. Das Android OS, die Systemarchitektur und der Android Market	2
2.1. Das Android OS.....	2
2.2. Android Apps entwickeln – Tools und Zugangsvoraussetzungen	3
2.3. Die Android-Systemarchitektur.....	4
2.4. Der Android Market	5
2.5. Zahlen und Fakten zum Android Market	5
2.6. Pro und Contra Android Market.....	6
2.7. Alternativen und weitere Vertriebsplattformen (Secondary und Independent Markets)	7
3. Die Applikationstypen	8
3.1. Paid-Apps versus Free-Apps	8
3.2. Trial-Versionen oder: Wie animiere ich den User eine App zu kaufen?	9
4. Revenue Generierung über In-app Advertisement oder In-app Billing	10
4.1. In-app Advertisement	10
4.2. In-app Billing.....	11
5. Der Entwicklungsprozess von Android Apps – Planung und Projektmanagement.....	11
6. Marketing V2.0.....	13
7. Nach der App ist vor der App	14
8. Produktpflege und Qualitätsmanagement.....	14
8.1. Warum regelmäßige Updates so wichtig sind	14
8.2. Qualitätsmanagement.....	15
9. Checkliste Applikationsentwicklung.....	16
10. Ausblick und Quintessenz	16
11. Glossar.....	17
12. Abbildungsverzeichnis.....	18
13. Literaturverzeichnis.....	18

1. Einleitung – Ziel der Arbeit

Der Markt der Smartphones hat sich zu einem bedeutenden Wachstumsmarkt entwickelt. Während im vierten Quartal 2009 allein 470.000 Android Smartphones in Westeuropa verkauft wurden, waren es im vierten Quartal 2010 bereits 7,9 Millionen Endgeräte, was einem Wachstum von 1.580 Prozent entspricht. Erstmals setzte sich Android damit vor Branchenprimus Apple, der von seinem Smartphone iPhone im Vergleichszeitraum ebenfalls beachtliche 5,2 Millionen Geräte absetzen konnte. Der IDC schätzt die jährlichen Wachstumsraten bis 2015 allein für Android auf jährlich 37%¹. Wir erleben gerade einen Umbruch des Nutzungsverhaltens, weg vom stationären PC daheim zu kompakten, leistungsstarken, mobilen Endgeräten, die der Nutzer jederzeit verfügbar hat. Zu diesen Endgeräten zählen weitläufig Smartphones und Tablets, die ebenso wie der PC daheim über ein Betriebssystem verfügen. Die etablierten Smartphone Betriebssysteme sind: Android OS, Apple iOS, Bada OS, Blackberry OS, Symbian OS, Web OS und Windows Mobile. Die beiden attraktivsten Plattformen stellen zurzeit jedoch das Android und Apple iOS dar. Out of the Box kommen die meisten Smartphones bereits mit einem Grundstock an Applikationen, die die Grundfunktionen, wie Email, Surfen und Medienwiedergabe abdecken. Jedoch hat jeder Nutzer andere Ansprüche und das ist die Stelle, an der Apps (= Anwendungen für die jeweilige Plattform) ins Spiel kommen. Es sind Programme, die eine bestimmte Funktion für einen bestimmten Einsatzzweck zur Verfügung stellen – vergleichbar mit Programmen auf dem heimischen PC. Das war es dann allerdings auch schon mit den Gemeinsamkeiten. Wenn man sich mit der Entwicklung mobiler Apps beschäftigt, stellt man schnell fest, dass es von der ersten Planung bis hin zum Vertrieb eine im Vergleich zur Softwareentwicklung für den PC andere Welt mit anderen Anforderungen und Prozessen ist, die eine dementsprechend differenzierte Herangehensweise an das Thema mobile App Entwicklung erfordert. Ziel dieser Arbeit ist es, ein Grundverständnis für das Betriebssystem Android, die Entwicklungsumgebung und Vertriebsstrukturen zu bekommen und den Softwareentwicklungsprozess mit seinen Besonderheiten zu erörtern.


2. Das Android OS, die Systemarchitektur und der Android Market

2.1. Das Android OS

Google erwarb im Jahr 2005 das von Andy Rubin, Rich Miner, Nick Sears und Chris White gegründete Startup Android Inc.. Das Startup beschäftigte sich seinerzeit mit der Softwareentwicklung für Mobiltelefone, im speziellen standortbezogener Dienste. Gegen Ende 2007 gab Google bekannt, zusammen mit 33 anderen Mitgliedern des Verbundes der Open Handset Alliance (OHA) ein Betriebssystem für mobile Endgeräte namens Android zu entwickeln. Bei Android handelt es sich um ein quelloffenes Betriebssystem, welches auf Linux basiert.

¹ Vgl. (Francisco Jeronimo, IDC, 2011)

Das erste Endgerät mit Android erschien im September 2008 beim Netzbetreiber T-Mobile unter dem Namen G1 mit der Android Version 1.5 (Cupcake). Kurz darauf erschien das Android SDK in der Version 1.0 und Google veröffentlichte den Source Code der Android Plattform im Rahmen der Apache Open Source License². Mittlerweile existiert Android in der Version 2.3 (Gingerbread) für Smartphones und 3.0 (Honeycomb) für Tablets. Die Entwicklung und Funktionalitäten der einzelnen Versionen stellt die nachfolgende Tabelle dar:



Funktionen	1.5 Cupcake	1.6 Donut	2.0 Eclair	2.1 Eclair	2.2 Froyo	2.3 Gingerbread	3.0 Honeycomb
Exchange-Anbindung (Active Sync) *	○	○	○	○	○	○	○
Sicherheitseinstellungen für Exchange	○	○	○	○	○	○	○
Virtual Privat Network (VPN-Unterstützung)	○	○	○	○	○	○	○
Unterstützung von HTML 5	○	○	○	○	○	○	○
Unterstützung WebM	○	○	○	○	○	○	○
Multitouch-Bedienung	○	○	○	○	○	○	○
globale Suchfunktion (Gerät und Web)	○	○	○	○	○	○	○
Unterstützung für Foto-LED	○	○	○	○	○	○	○
Unterstützung für Flash *	○	○	○	○	○	○	○
Sprachsteuerung	○	○	○	○	○	○	○
Google Maps Navigation	○	○	○	○	○	○	○
Smartphone als Modem via USB *	○	○	○	○	○	○	○
Smartphone als WLAN-Hotspot *	○	○	○	○	○	○	○
App-Installation auf SD-Karte	○	○	○	○	○	○	k.A.
Unterstützung NFC-Chip	○	○	○	○	○	○	k.A.
Unterstützung VoIP-Telefonie	○	○	○	○	○	○	k.A.
Unterstützung Webstorage	○	○	○	○	○	○	k.A.
Unterstützung von Gyroskopen	○	○	○	○	○	○	○
integrierter Downloadmanager *	○	○	○	○	○	○	○
Dual-Core-Unterstützung	○	○	○	○	○	ab Android 2.3.3	○
Push-Funktion für Apps	○	○	○	○	○	○	○

* bei älteren Versionen ggf. vom Hersteller nachgerüstet

Abbildung 1: Die einzelnen Android Versionen im Überblick (Kaliudis, 2011)

2.2. Android Apps entwickeln – Tools und Zugangsvoraussetzungen

Für die Entwicklung von Android Apps wird das Android Software Development Kit (Android SDK) benötigt, welches unter <http://developer.android.com/sdk/index.html> bezogen werden kann. Bevorzugt man als Entwicklungsumgebung die Plattform Eclipse, so ist ferner noch die Installation des Android Development Tools (ADT) Plugin notwendig³.

Um später Apps im Android Market publizieren zu können, sind zwei Voraussetzungen zu erfüllen. Als erstes ist ein Developer Account erforderlich, welcher unter <http://market.android.com/publish/> beantragt werden kann. Die einmalige Registrierungsgebühr beträgt 25 US\$. Zum Verkauf kostenpflichtiger Apps ist ferner ein Verkäuferkonto bei Google Checkout notwendig. Über Google Checkout werden sämtliche Transfers im Rahmen des Verkaufs von Apps verwaltet⁴.

² Vgl. (Sayed Y. Hashimi, 2010)

³ Vgl. (Google Inc., 2011)

⁴ Vgl. (Google Inc., 2011)

2.3. Die Android-Systemarchitektur

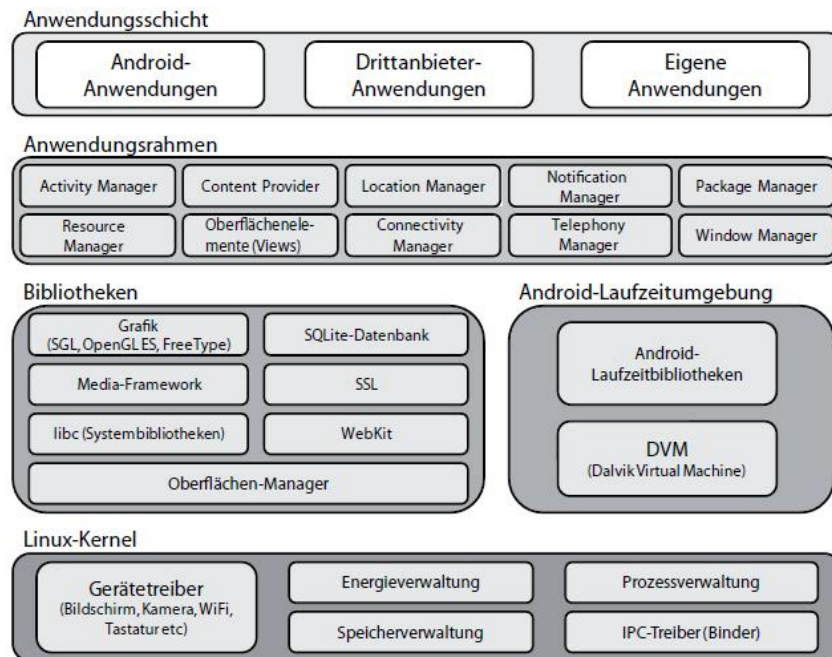


Abbildung 2: Die Android Systemarchitektur (Becker & Pant, 2009)

Die Android-Systemarchitektur setzt sich zusammen aus der Anwendungsschicht, dem Anwendungsrahmen, den Bibliotheken, der Android-Laufzeitumgebung und dem Linux-Kernel. Innerhalb der Anwendungsschicht findet die Kommunikation zwischen Nutzer und Endgerät, sowie den Anwendungen untereinander statt. Die Android-Anwendungen an sich lassen sich untergliedern in die von Google mitgelieferten Standardanwendungen und Eigenentwicklungen. Der Anwendungsrahmen ist die Schicht im Android-Systemaufbau, die den Unterbau für die Anwendungen bildet und stellt verschiedene Programmierschnittstellen bereit. Er ermöglicht somit die Realisierung der Kommunikation zwischen Anwendungen selbst, sowie der Anwendung und dem Benutzer. Anwendungsschicht und –rahmen greifen auf eine Gruppe von Standardbibliotheken zu. Diese Standardbibliotheken stellen alle zum Betrieb von Android-Anwendungen erforderlichen Funktionalitäten wie Datenbanken, 3D-Grafikbibliotheken, Webzugriffe, GUI-Design zur Verfügung. Die Android-Laufzeitumgebung besteht im Kern aus der sogenannten Dalvik Virtual Machine, abgekürzt DVM. Jede Android-Anwendung läuft in einem eigenen kleinen Betriebssystemprozess – einer sogenannten Sandbox - und verfügt über eine eigene DVM. Dieses Prinzip kostet zwar Ressourcen, bietet aber erhebliche Vorteile in Bezug auf Sicherheit und Verfügbarkeit. Die Anwendungen teilen sich keinen gemeinsamen Speicher und sollte ein Prozess stocken, so bringt das nicht gleich das gesamte System ins Trudeln. Die Sandbox regelt somit den Zugriff auf Systemfunktionen und Ressourcen außerhalb der Sandbox. Erfordert eine Anwendung Zugriff auf Ressourcen außerhalb der Sandbox, so müssen diese explizit über Berechtigungen angefordert werden. Je mehr Berechtigungen eine Anwendung einfordert, umso mehr weicht jedoch auch das

Sicherheitskonzept der Sandbox auf. Ferner ist es auch möglich, mehrere Anwendungen gemeinsam in einer Sandbox laufen zu lassen. Der Linux-Kernel bildet die Basis des Android-Systems und stellt neben den Hardware-Treibern auch die entsprechenden Verwaltungsfunktionalitäten für Prozesse, Speicher und Energie bereit.

2.4. Der Android Market

Der Android Market ist ein virtueller Marktplatz für Android Apps, welcher auf Google zertifizierten Geräten über eine eigenständige Applikation, als auch über das Web (<https://market.android.com/>) erreichbar ist. Eine besondere Funktionalität der Web Variante des Android Markets ist die Verwaltung der dem eigenen Google Account zugeordneten Geräte und die Remote-Installation von Apps.

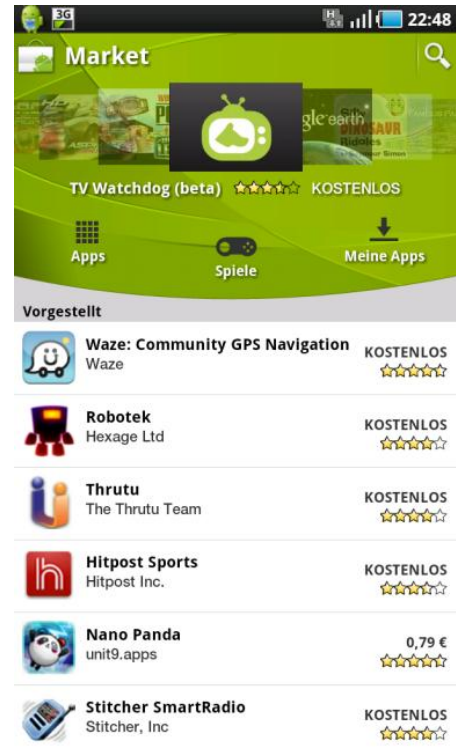


Abbildung 3: Der Android Market

2.5. Zahlen und Fakten zum Android Market

- Der Android Market startete am 22. Oktober 2008 und erfuhr seitdem 2 Redesigns
- Stand 1. April 2011 umfasste der Android Market ca. 380.000 Apps⁵
- Täglich finden zurzeit zwischen 800 und 1400 neue Apps Eingang in den Android Market; das entspricht im Monat mehr als 30.000 Apps⁶
- 63,2% aller Downloads entfallen auf kostenlose Apps, 37,7% auf Bezahl-Apps⁷
- 83,7% der Downloads sind Anwendungen, 16,3% Spiele⁸
- Nur 25,8% aller Apps erreichen mehr als 1000 Downloads⁹
- Die Top 10 Kategorien im Android Market sind: Unterhaltung, Personalisierung, Bücher & Nachschlagewerke, Tools, Lifestyle, Reisen, Rätsel & Denksport, Sport, Lernen, Effizienz-Tools¹⁰

⁵ Vgl. (AndroLib.com, 2011)

⁶ Vgl. (AndroLib.com, 2011)

⁷ Vgl. (AndroLib.com, 2011)

⁸ Vgl. (AndroLib.com, 2011)

⁹ Vgl. (AndroLib.com, 2011)

¹⁰ Vgl. (AppBrain.com, 2011)

2.6. Pro und Contra Android Market

Ein großer Vorteil des Android Market ist seine hohe Verfügbarkeit, da er auf allen von Google zertifizierten Endgeräten vorinstalliert ist und für den Nutzer des Endgerätes bereits ab der ersten Nutzung zur Verfügung steht. Weiterhin sprechen für den Android Market die reichhaltige und täglich wachsende Anzahl an Apps. Direkte Vorteile für Entwickler sind die im Vergleich zu Apple sehr kurzen Approval-Prozesse für neu eingestellte Apps und Updates, die oftmals binnen weniger Stunden erfolgen. Ein komfortables Upload-Interface unterstützt den Entwickler beim Einstellen seiner App in den Market, im späteren Verlauf zeigen umfangreiche Statistiken unter anderem den Verlauf der Downloads und die Aufteilung auf verschiedene Android-Revisionen. Ebenfalls kann der Entwickler sich über mögliche Fehler, die auf Geräten der Benutzer aufgetreten sind, informieren und diese zeitnah beheben. Für sein Handling beansprucht Google eine Transaktionskostengebühr in Höhe von 30% vom Verkaufspreis der Apps (70/30 Modell)¹¹. Das Thema Geld führt gleichzeitig zu einem der größten Nachteile des Android Market, nämlich die eingeschränkten Möglichkeiten bezüglich der Bezahlverfahren. Hier gibt es einzig und allein nur die Möglichkeit, Apps über Google Checkout zu erwerben, was eine Kreditkarte erfordert. Dass man in den USA schon ein Stück weiter ist und das Carrier Billing für T-Mobile Kunden ermöglicht hat, ist zurzeit für Kunden in anderen Ländern nur wenig tröstlich. Während der Android Market bereits in 56 Ländern der Welt genutzt werden kann, ist der Kauf von Apps jedoch nur in 32 Ländern möglich¹². Aus marketing-technischer Sicht nachteilig sind mangelnde Social Sharing Möglichkeiten im Android Market. So ist es nicht möglich Apps, die man gut findet mit Freunden zu „teilen“ oder gar die Liste seiner Lieblings-Apps Freunden zu zeigen. Auch eine Integration sozialer Netzwerke (Facebook-„Share“) ist zurzeit nicht möglich.

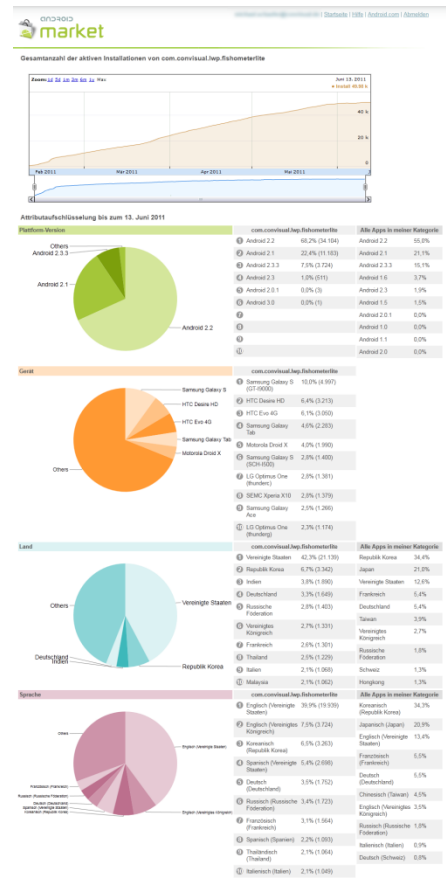


Abbildung 4: Statistikfunktion des Android Market

¹¹ Vgl. (Google Inc., 2011)

¹² Vgl. (Google Inc., 2011)

2.7. Alternativen und weitere Vertriebsplattformen (Secondary und Independent Markets)

Die mit Sicherheit spektakulärste Alternative zum Android Market hat sich im März 2011 mit Amazons Appstore für Android aufgetan. Mit interessanten Ideen macht Amazon die Kunden auf das Angebot aufmerksam, so verschenkt Amazon täglich eine sonst kostenpflichtige App und bietet eine sogenannte „Test Drive“-Funktion, bei der kostenpflichtige Apps 30 Minuten lang getestet werden können. Leider erfolgt dieser Test nicht auf dem Endgerät, sondern per Emulator oder im Browser. Dadurch ergibt der „Test Drive“ nur bei Apps Sinn, die weder Kamera, noch Gyroskop oder GPS nutzen. Auch der Amazon Appstore läuft entweder als Client auf dem Endgerät oder am heimischen PC via Webbrowser. Als Besonderheit und um die Attraktivität des Stores zu steigern, verschenkt Amazon jeden Tag eine Kauf-App (a paid app for free every day). Bislang ist das Shoppen im Amazon Appstore jedoch nur für Kunden in den USA möglich. Die Transaktionskostengebühr beträgt bei Amazon übrigens wie bei Google 30%¹³.

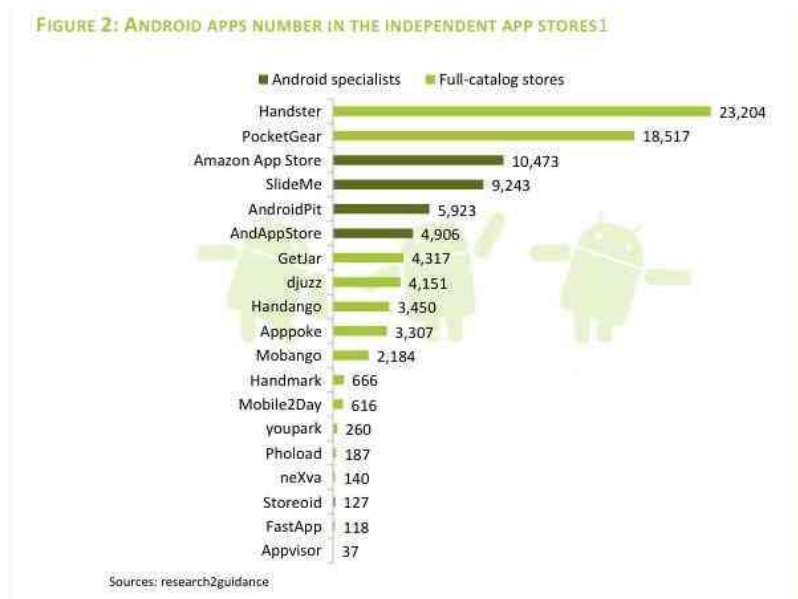


Abbildung 5: Ranking der Independent Appstores (Roehlinger, 2011)

Zurzeit existieren ca. 30 weitere Secondary Markets, wobei der größte von Ihnen Handster ist. Handster ist ein sogenannter „Full-Catalog Store“, welcher nicht nur die Android Plattform bedient. Weitere namhafte Markets sind u.a.

- Pocketgear (<http://www.pocketgear.com/de,de,eur/android/index.html>),
- SlideME (<http://slideme.org/>),
- AndroidPit (<http://www.androidpit.de/de/android/market/app-center>),
- AndAppStore (<http://andappstore.com/AndroidApplications/apps/>)

¹³ Vgl. (Janssen, 2011)

- GetJar (<http://www.getjar.com/mobile-all-applications-for-android-os-all/>),
- Handango (<http://www.handango.com/content/Android-Software>),
- Appoke (<http://appoke.com/>),
- Mobango
(http://www.mobango.com/swarea/index.php/home?platform=Android&area=apps&standardFilter=most_downloaded&idSubcategory=0),
- Handmark (<http://store.handmark.com/c/280/android-software>),
- Mobile2Day (<http://www.mobile2day.com/de,de,eur/android/index.html>),
- OnlyAndroid (<http://onlyandroid.mobihand.com/appstore.asp>),
- Pdassi (<http://android.pdassi.de/>).

Diese einzelnen Plattformen unterscheiden sich in App Angebot, Geschäftsmodellen, Reichweite und Zahlungsmöglichkeiten. Ebenso haben Netzbetreiber in Ihren Content-Shopsystemen Android Apps integriert und auch der Hersteller Sony Ericsson bietet in seiner PlayNow Arena mittlerweile Android Apps an. Die Nutzung weiterer Vertriebsplattformen unterscheidet die Android Welt von der Apple iOS Welt, in der als einziger Vertriebskanal der Apple App Store vorgesehen ist.

3. Die Applikationstypen

3.1. Paid-Apps versus Free-Apps

Neben der Typisierung nach App Kategorien wie Tools oder Spiele unterscheidet man Apps auch noch nach deren Verfügbarkeit in Free- und Paid-Applications. Im Android Market erstrecken sich 62,3% aller Downloads auf kostenfreie Apps und 37,7% der Downloads auf Paid- Apps. Dies ist eine niedrige Quote, wenn man betrachtet, dass die Aufteilung zwischen Paid- und Free-Apps in anderen Markets im Verhältnis 60 zu 40 steht. Im Apple App Store beträgt der Anteil der Kaufapplikationen beispielsweise 67%¹⁴. Den größten Anteil daran, dass dieses Verhältnis im Android Market nahezu umgekehrt verläuft, tragen sicherlich die eingeschränkten Zahlungsmöglichkeiten für die Nutzer. In einer Studie von research2guidance stellt man jedoch fest, dass sich der Trend im Android Market zugunsten der Paid-Apps verschiebt. Dort stellte man auch fest, dass die App Durchschnittspreise zwischen den Kategorien sehr stark schwanken. Beispielsweise kostete eine App aus der Kategorie „Personalisierung“ laut dieser Studie im April 2011 in Schnitt 1,31 US\$, während der Durchschnittspreis für Apps aus der Kategorie „Medizin“ 19,00 US\$ betrug¹⁵.

¹⁴ Vgl. (Oschatz, 2010)

¹⁵ Vgl. (research2guidance, 2011)

3.2. Trial-Versionen oder: Wie animiere ich den User eine App zu kaufen?

Wie im vorherigen Abschnitt erörtert worden ist, gestaltet es sich derzeit recht diffizil, Bezahl-Apps im Android Market erfolgreich zu platzieren. Es gibt verschiedene Möglichkeiten, bei Nutzern über das Modell der Trial-Versionen einen Anreiz zum Kauf einer App zu schaffen. Einige Modelle sollen nachfolgend vorgestellt werden:

Trial-Version mit begrenzter Ausführungsdauer – Modell 1: Die App läuft nach ihrem Start nur für eine bestimmte Dauer. Es stehen alle Funktionalitäten der Vollversion zur Verfügung. Nach Ablauf dieser Zeit erscheint ein Hinweis auf die unbeschränkt nutzbare Vollversion und die Trial-Version kann nur noch beendet werden.

Trial-Version mit begrenzter Ausführungsdauer – Modell 2: Im Gegensatz zum Modell 1 ist hier nicht die Laufzeit der App je Ausführung begrenzt, sondern die App startet nach Ablauf des Testzeitraums, zum Beispiel 2 Wochen, nicht mehr. Stattdessen wird bei Ausführung der App dann wie bei Modell 1 ein Hinweis auf den abgelaufenen Test-Zeitraum eingeblendet und ein Verweis auf die Vollversion gegeben. Auch hier stehen während des Testzeitraums alle Funktionalitäten, die die Vollversion bietet, zur Verfügung.

Trial-Version mit beschränkter Zahl an Starts: Hierbei sind die Starts der App limitiert. Ist eine bestimmte Anzahl von Starts der App erreicht, startet diese nicht mehr wie gewohnt, sondern weist auf das Ende des Testzeitraums und die Vollversion hin.

Trial-Version mit eingeschränktem Funktionsumfang: Hier ist nicht der Zeitraum, in welchem die App lauffähig ist, limitiert, sondern der Funktionsumfang im Vergleich zur Vollversion. Die Abgrenzung, wie viele Funktionen eine Trial-Variante umfassen sollte ist jedoch ein schwieriges Kapitel. Hier muss man einen Kompromiss finden, der einerseits so viele Funktionen bietet, damit die App den Nutzer überzeugt, darf andererseits aber auch nicht zu viele Funktionen bieten, die der Vollversion somit die Daseinsberechtigung entzieht.

Trial-Version mit verzögertem Start/Beenden der App: Der Vorteil vieler Android Apps ist, dass diese meist unmittelbar nach dem Anwählen zur Verfügung stehen und keine längeren Ladezeiten haben. Dieses Komfortmerkmal unterbindet man mit diesem Modell, indem man beispielsweise die Startphase um 15 Sekunden verzögert und während dieser Zeit einen Hinweis auf die Vollversion einblendet. Gleiches kann ebenfalls bei Beendigung der App geschehen.

Trial-Version mit eingeschränkter Nutzung: Bei dieser Variante der Trial-Version wird der Nutzer daran gehindert, die App ohne künstliche Einschränkungen zu nutzen. Solche künstliche Einschränkung könnte beispielsweise ein Tutorial sein, das immer zum Start der Anwendung durchlaufen werden muss und nicht übersprungen werden kann. Oder bei der

Anwahl bestimmter Funktionalitäten werden immer Hinweise zur Nutzung der Funktion nachgeschaltet, welche nicht übersprungen werden können.

Trial-Version mit eingeschränktem Content (Sonderform): Diese Sonderform erstreckt sich auf den Bereich der Spiele. Hier ist es bei der Trial-Version beispielsweise nur möglich, die ersten 2 oder 3 Levels des Spieles zu spielen. Hat der Nutzer diese Levels gemeistert, erfolgt ein Hinweis auf die Vollversion.

Es bleibt dem Entwickler und dem konkreten Anwendungsfall überlassen, ob man ein oder auch mehrere Trial-Merkmale, dann in abgeschwächter Form, kombiniert.

4. Revenue Generierung über In-app Advertisement oder In-app Billing

4.1. In-app Advertisement

Das In-app Advertisement ist ein häufig genutzter Service, um auch bei kostenlosen Varianten einer App einen Beitrag zur Deckung der Kosten zu erwirken. Dieser Service kann für Android Apps über den Mobile Advertising Spezialisten AdMob realisiert werden. AdMob gehört seit 2009 zu Google und dementsprechend integriert ist der Service. Zur Nutzung ist eine Registrierung auf <http://www.admob.com/> und die Installation des AdMob SDK notwendig¹⁶. Die letztendliche Integration ist mit wenigen Zeilen Code erledigt¹⁷. Relevant für die Abrechnung sind die Klicks, die auf die Werbebanner erfolgten.



Abbildung 6: Beispiel In-app Advertisement bei dem Spiel „Air Control Lite“

¹⁶ Vgl. (AdMob Google Inc, 2011)

¹⁷ Vgl. (Google Inc., 2011)

4.2. In-app Billing

Das In-app Billing ist ein relativ neuer Service und kann seit Ende März 2011 implementiert werden. Hintergrund des In-app Billing ist die Optimierung der Monetarisierung der App. Via In-app Billing erhält der Entwickler die Möglichkeit, Käufe innerhalb seiner App vorzusehen. Durch diese Käufe können beispielsweise Erweiterungen der App oder entsprechende Nutzungsberechtigungen erworben werden. Somit lassen sich über In-app Billing nun Abo-Modelle zur Nutzung einer App erstellen. Ferner ist es auch möglich, sogenannte Try-and-Buy Modelle einer App zu realisieren ohne gesonderte Trial-Versionen erstellen zu müssen. Von allen getätigten In-app Billing Umsätzen erhebt Google ebenfalls eine Transaktionskostengebühr von 30%¹⁸. Man zählt das In-app Billing zu den sogenannten Freemium Modellen, die sich bei Nutzern wachsender Beliebtheit erfreuen.

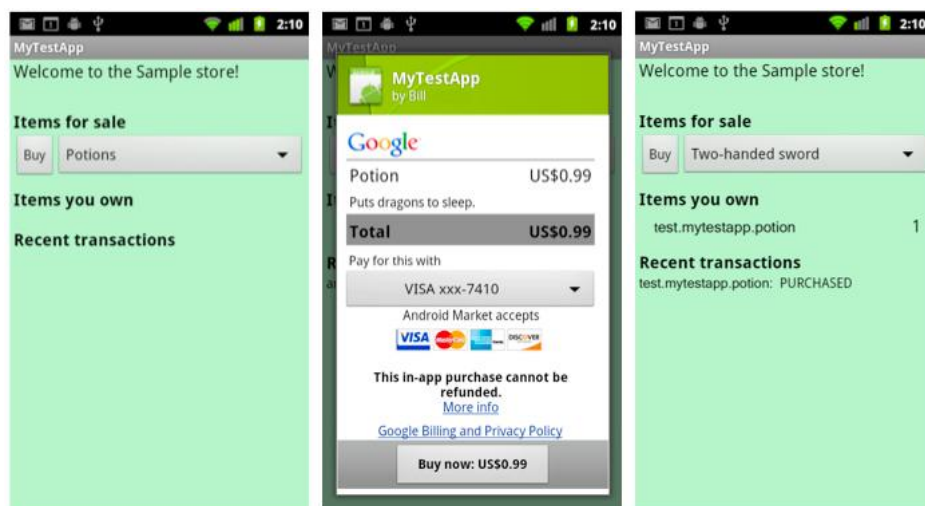


Abbildung 7: Google Demo-Case für In-app Billing (Google Inc, 2011)

5. Der Entwicklungsprozess von Android Apps – Planung und Projektmanagement

Die Entwicklung einer jeden App ist ein Projekt, das je nach Umfang der App mehr oder minder umfangreich ausfällt. Je umfangreicher eine App jedoch ist, desto wichtiger und tragender wird eine sorgfältige Planung des Gesamtprozesses und der involvierten Ressourcen.

Bei der folgenden Betrachtung wird davon ausgegangen, dass die App für den Eigenvertrieb und nicht im Kundenauftrag entwickelt wird. Der Entwicklungsprozess an sich lässt sich in 4 Grobphasen unterteilen: Definitionsphase, Planungsphase, Realisationsphase und Vermarktungsphase.

¹⁸ Vgl. (Google Inc., 2011)



Abbildung 8: Die Phasen des Entwicklungsprozess

In der Definitionsphase werden grundlegende Fragestellungen zur App erörtert, wie zum Beispiel, welcher App-Typ entwickelt werden soll. Hierbei ist die Wahl zwischen Spielen und Anwendungen. Spiele sind zwar eine lukrative Angelegenheit, aber auch dementsprechend aufwändiger in der Programmierung (3D Grafik, usw.). Anwendungen stellen wesentlich einfachere Anforderungen an die Programmierung und sind zudem laut Statistik auch der gefragtere App-Typ. Diese Punkte sollten bei der Entscheidung und anschließenden Aufwandsabschätzung berücksichtigt werden. Als nächstes sollte die Frage beantwortet werden, ob eine Free- oder Paid-App entwickelt werden soll. Im Falle einer Paid-App ist zu prüfen, welcher Sub-Kategorie des Markets diese später zugeordnet werden kann und welche Durchschnittspreise in dieser Kategorie pro App realisiert werden. Dies ist ein wichtiger Anhaltspunkt für die spätere Preisfindung. Sowohl für Paid- als auch für Free-Apps ist alsdann zu prüfen, welche Modelle zur Umsatzgenerierung genutzt werden sollen, als da wären In-app Advertisement und In-app Billing. Bei Paid-Apps ist die Frage zu beantworten, ob eine kostenfreie Trial-Variante erstellt werden soll und welches Trial-Modell zum Zuge kommen soll.

Sind diese grundlegenden Fragestellungen erörtert, kann die Detailplanung der App im Rahmen der Planungsphase begonnen werden: Welche Funktionalitäten soll die App abdecken und beanspruchen diese Funktionalitäten besondere Schnittstellen oder Features, die ältere Versionsstände oder bestimmte Gerätegruppen ausschließen (Beispiel: Live Wallpaper - Unterstützung erst seit Android 2.1). Hier sind eventuelle Konflikte mit der der Planung der zu unterstützenden Geräteklassen und Android Versionsstände zu prüfen und gegebenenfalls zu lösen. Umgekehrt können sich durch Maßgaben wie der Unterstützung von Geräten der Einsteigerklasse auch Beschränkungen nach oben in Form der begrenzten

Nutzung von Hardware-Ressourcen ergeben. Bei der Planung der zu unterstützenden Geräteklassen und Versionsstände darf freilich nicht der Testaufwand außer Acht gelassen werden. Je mehr Endgeräte und Versionsstände die App unterstützt, desto aufwändiger fällt die Testphase aus.

Das eigene App-Konzept sollte freilich auch auf ähnliche, bereits existierende Konzepte geprüft werden. Sind diese Konzepte erfolgreich? Was kann verbessert werden? In diesem Schritt sollten auch die USPs der App definiert werden: Was, respektive welche Funktionalitäten sollen die eigene App später einmal auszeichnen?

Nach diesen Schritten bietet es sich an, ein Mock-up der App zu erstellen. Auch der Projektplan einschließlich der Zeitplanung kann nun erstellt werden. Es müssen die erforderlichen Ressourcen definiert und geprüft werden, weiterhin ob alle Ressourcen intern abgedeckt werden können. Falls nicht, müssen entsprechende externe Ressourcen geprüft werden. Bei umfangreicheren Apps bietet es sich an, Meilensteine zu definieren, zu denen jeweils Zwischenstände der App für erste Tests erstellt werden und geprüft werden können, ferner, ob die App wie geplant umgesetzt werden kann oder sich Probleme ergeben, die bei der Planung nicht zu erkennen waren. Auch die Testszenarien sollten nun definiert werden.

Nach abgeschlossener Planung kann mit der Umsetzung der App im Rahmen der Realisationsphase begonnen werden. Die Realisationsphase umfasst auch die Durchführung der Tests der App.

Im Zuge der Vermarktungsphase müssen die Vertriebskanäle erörtert und eine passende Vermarktungsstrategie für die App entwickelt werden. In Abstimmung mit dem Entwicklungsteam sollte bereits rechtzeitig mit der Planung der Vermarktungsstrategie und Anfertigung des Marketing-Materials begonnen werden.

6. Marketing V2.0

Eine tolle App allein garantiert noch keine hohen Downloadzahlen. Die herausragenden Merkmale bzw. Alleinstellungsmerkmale der eigenen App müssen herausgearbeitet und kommuniziert werden. Das heißt: Mindestens aussagekräftige Screenshots und Beschreibungstexte müssen erstellt werden. Sofern möglich sollte auch von der Möglichkeit Gebrauch gemacht und ein Demonstrationsvideo der App erstellt werden, da der Android Market die Möglichkeit zur Einbindung von Videos der Youtube-Plattform bietet. Leider bietet der Android Market direkt keine Integration von Social Media Komponenten wie Twitter oder Facebook. Dennoch sind auch diese Kanäle bei der Vermarktung heutzutage hilfreich und relevant. Kanäle wie Social Media sind deshalb auch ein Hilfsmittel, um nicht in der Masse der Apps unterzugehen, da die Anzahl der neuen Apps pro Tag – wie in den einleitenden Kapiteln dargestellt – sehr hoch ist. Die eigene App verschwindet demnach

zwangsläufig recht schnell aus der Liste der Neuerscheinungen. Auch ein guter Review der eigenen App in einschlägigen Blogs steigert deren Bekanntheit und somit deren Downloadzahlen. Der Android Market ermöglicht seinen Nutzern, die angebotenen Apps über ein 5-Sterne System zu bewerten und zu kommentieren. Das Rating über das Sterne-System wird von den Nutzern sehr stark wahrgenommen, da diese Bewertung sehr einfach abgegeben werden kann. Kommentare erfolgen erfahrungsgemäß seltener, da der Aufwand zum Verfassen eines Kommentars natürlich höher ist. Einen sehr hohen Stellenwert hat das Sterne-Bewertungssystem, da sich viele Nutzer bei den Suchergebnissen an diesem Rating orientieren. Ein hohes Rating bringt die App einen Schritt näher auf das Endgerät, da das Rating eines der Motivationsmerkmale für den User ist, die Detailseite der App im Market zu besuchen.

7. Nach der App ist vor der App

Nach der App ist vor der App – dies darf wirklich wörtlich genommen werden, denn mit der Einstellung der App in den Market endet das Projekt nicht. Wie im vorhergehenden Abschnitt erläutert, haben die Nutzer die Möglichkeit Feedbacks zur App in Form von Kommentaren abzugeben. Diese Kommentare liefern einerseits Hinweise zu Endgeräten, auf denen die App nicht wie vorgesehen läuft, andererseits aber auch den ein oder anderen guten Hinweis, was an der App verbessert werden kann. Ferner liefert auch die Entwicklerkonsole des Android Market selbst relevante Informationen, da diese detaillierte Berichte zu Anwendungsfehlern bereithält. Sollen die Downloadzahlen der App von einer gewissen Nachhaltigkeit geprägt sein, tut der Entwickler gut daran, die User Feedbacks und die Berichte zu Anwendungsfehlern regelmäßig auszuwerten.

Neu	NullPointerException in LiveWpServiceFishOMeterLite.onCreate()	15 Berichte	2 Berichte/Woche
Neu	ActivityNotFoundException in Instrumentation.checkStartActivityResult()	11 Berichte	1 Berichte/Woche
Neu	IllegalArgumentException in Surface.unlockCanvasAndPost()	1 Berichte	0 Berichte/Woche
Neu	RuntimeException in InputQueue.nativeRegisterInputChannel()	2 Berichte	0 Berichte/Woche

Abbildung 9: Entwicklerkonsole Android Market - Übersicht zu Anwendungsfehlern

8. Produktpflege und Qualitätsmanagement

8.1. Warum regelmäßige Updates so wichtig sind

Wie im vorherigen Abschnitt dargestellt wurde, ist der Android Market nicht nur Vertriebskanal, sondern auch Feedbackplattform für den Entwickler und liefert über Fehlerberichte und User-Kommentare die Grundlage für Bug-Fixings und Updates der App. Regelmäßige Updates sind aber auch aus anderen Gründen wichtig. Die Android Welt wächst schnell und

dynamisch - diese Dynamik und Offenheit des Systems birgt jedoch nicht nur Vorteile in sich, sondern bringt auch Nachteile mit sich, die wir unter dem Fachbegriff Fragmentierung zusammenfassen. So kann das Aufkommen neuer Geräteklassen, wie zum Beispiel die zurzeit an Popularität gewinnenden Tablets dafür sorgen, dass die App weitere grafische Auflösungen für diese Geräteklasse unterstützen muss, um auf diesen Geräten korrekt dargestellt zu werden. Ferner haben Hersteller zum Beispiel die Möglichkeit, dem Basissystem eine eigene Bedienoberfläche überzustülpen (beispielsweise Sense bei HTC oder TouchWiz bei Samsung). Auch Netzbetreiber machen von diesen Anpassungsmöglichkeiten für die von ihnen vertriebenen Endgeräte regen Gebrauch, was man unter dem Stichwort Branding zusammenfasst. All diese Anpassungen können zu Seiteneffekten führen, die die ordnungsgemäße Funktion der eigenen App beeinträchtigen oder im Extremfall nicht mehr ermöglichen. Der Android-Emulator ermöglicht den Test der eigenen App unter den einzelnen Android Versionsständen – die Modifikationen von Herstellern und Netzbetreibern berücksichtigen kann er jedoch nicht. Es gibt also eine ganze Reihe guter Gründe, die für regelmäßige Updates der Anwendung sprechen.

8.2. Qualitätsmanagement

Das Qualitätsmanagement ist ein besonders wichtiger Aspekt im Rahmen des Entwicklungs- und Lebenszyklusses einer App. Wie bereits erläutert wurde, können im Emulator eine ganze Reihe von Testszenarien für die einzelnen Android-Versionen gefahren werden, jedoch nicht alle erforderlichen Tests können virtuell abgedeckt werden. Unter dem Stichwort Fragmentierung wurden eben die Möglichkeiten zur Anpassung des Basissystems erläutert. Dies macht einen Test auf einer Auswahl entsprechender Endgeräte zwingend erforderlich. Tests sollten sich mindestens auf einige ausgewählte, besonders kritische Geräte aller unterstützten Geräteklassen erstrecken. Es ist wichtig, die App zum Beispiel auf Geräten der Einsteigerklasse zu testen, da diese zumeist über eingeschränkte und knappe Hardwareressourcen verfügen (schwache CPU (<600MHz, knapper Hauptspeicherausbau (<384 MB)). Auch sollte ein besonderes Augenmerk auf den Speicherverbrauch und die CPU-Auslastung der App gelegt werden. Je mehr Geräteklassen die App unterstützt und je umfangreicher die Funktionalitäten der App sind, desto höher ist der resultierende Testaufwand. Dieser Testaufwand bedingt Investitionen in Form benötigter Hardware (Endgeräte), Organisationsaufwand (Testplanung), Zeit (Testdurchführung und Auswertung) und Personalaufwand zwecks Durchführung.

9. Checkliste Applikationsentwicklung

Fassen wir abschließend noch einmal die wichtigsten Punkte, die sich im Rahmen der App-Entwicklung ergeben, zusammen:

- Entscheidung für einen App-Typ: Spiel oder Anwendung
- Erörterung der Funktionalitäten der App. Eventuelle Einschränkungen hinsichtlich Kompatibilität prüfen.
- Paid-App: Preis Definition – Als Anhaltspunkt den Durchschnittspreis der Kategorie ermitteln
- Paid-App: Soll eine kostenfreie Trial-Version entwickelt werden? Wenn ja, welches Trial-Modell kommt zum Zuge?
- Revenue Generierung: In-app Billing oder In-app Advertising?
- Vermarktungsstrategie: Wie und welche Vertriebskanäle sollen genutzt werden?

10. Ausblick und Quintessenz

Es steht vollkommen außer Zweifel, dass Android eines der bedeutenden Smartphone Betriebssysteme der Zukunft sein wird. Die Wachstumsraten und die Entwicklung der Marktanteile der Smartphones mit Android Betriebssystem unterstreichen dies. Zudem bietet mittlerweile jeder namhafte Hersteller Android Endgeräte an und neben Top- und Mittelklasse Geräten findet Android zunehmend Eingang auf Geräten der Einstiegsklasse. Hinzu kommen neue Gerätegattungen wie Tablets, die neue Anwendungsbereiche für das Betriebssystem eröffnen und sogar Settop-Boxen mit Android sind bereits am Markt erhältlich. Die Open Handset Alliance tut ein Übriges, um das Betriebssystem weiter voranzutreiben. Neuerungen wie In-app Billing oder NFC (Near Field Communication) erhöhen die Attraktivität von Android und eröffnen immer wieder neue Anwendungsfelder. Auch die Hardware wird zunehmend performanter. Kommen zurzeit die ersten Endgeräte mit Dual-Core Prozessoren und dediziertem Grafikchip auf den Markt, so ist der Weg der nächsten Hardware Generation schon vorgezeichnet. Nvidia demonstrierte bereits den Nachfolger des aktuellen Tegra 2 Chipsatzes, der unter dem Codenamen Kal-El entwickelt wird und neben dedizierter Grafik über vier Prozessorkerne verfügen wird. Android bleibt also spannend, sowohl für Nutzer, Vermarkter, Hersteller als auch Entwickler!

11. Glossar

Dalvik Virtual Machine (DVM): Von dem Google Mitarbeiter Dan Bornstein entwickelte Android-Laufzeitumgebung.

Digitales Mock-up: Ein computergeneriertes Modell der App, das die verschiedenen Ein- und Ausgabemasken und Zustände visualisiert.

Eclipse (IDE): Eclipse ist ein quelloffenes Programmierwerkzeug zur Softwareentwicklung in der Programmiersprache JAVA und wird von der Eclipse Foundation herausgegeben.

Freemium: Kunstwort aus den Worten Free und Premium. Es handelt sich um ein Geschäftsmodell, bei dem ein Basis-Setup an Funktionen/Diensten gratis angeboten wird und für weitere Funktionen/Dienste ein Entgelt beispielsweise über In-app Käufe erhoben wird.

Google Checkout: Ein Online-Bezahlsystem von Google Inc.. Zur Anmeldung ist ein Google Konto und eine gültige Kreditkarte notwendig. Mit Checkout können in diversen Online-Shops Einkäufe getätigt werden, so auch im Android Market.

Nearfield Communication (NFC): NFC ist ein Übertragungsstandard zum kontaktlosen Austausch von Daten über kurze Strecken. Über NFC können zum Beispiel elektronische Ticketing-Systeme realisiert werden. NFC wird von Android seit der Version 2.3 (Gingerbread) unterstützt.

Open Handset Alliance (OHA): Ein Konsortium von zurzeit insgesamt 80 Unternehmen der Telekommunikations-Industrie. Sie wurde am 05.11.2007 unter der Federführung von Google Inc. gegründet. Ziel der OHA ist die Schaffung offener Standards von Mobilgeräten.

Smartphone: Mobiltelefon mit erweiterten Funktionalitäten gegenüber sogenannten Featurephones. Smartphones verfügen über Betriebssysteme, die sich mit Hilfe von mobilen Applikationen – sogenannten Apps erweitern lassen. Bezüglich der Bedienung hat sich bei Smartphones die Eingabemethode per Touchscreen etabliert.

Tablet: Gerätegattung, die meist auf Betriebssystemen und Hardware von Smartphones basiert, jedoch im Vergleich zu Smartphones einen wesentlich größeren Touchscreen besitzt und somit eine komfortablere Nutzung von beispielsweise Internet und eBooks ermöglicht.

Unique Selling Proposition (USP): Ein Begriff aus dem Marketing. Dabei handelt es sich um die Eigenschaften des Produktes, die es gegenüber Produkten der Mitbewerber abgrenzt (Alleinstellungsmerkmale)

12. Abbildungsverzeichnis

Abbildung 1: Die einzelnen Android Versionen im Überblick (Kaliudis, 2011)	3
Abbildung 2: Die Android Systemarchitektur (Becker & Pant, 2009)	4
Abbildung 3: Der Android Market	5
Abbildung 4: Statistikfunktion des Android Market.....	6
Abbildung 5: Ranking der Independent Appstores (Roehlinger, 2011)	7
Abbildung 6: Beispiel In-app Advertisement bei dem Spiel „Air Control Lite“	10
Abbildung 7: Google Demo-Case für In-app Billing (Google Inc, 2011)	11
Abbildung 8: Die Phasen des Entwicklungsprozess	12
Abbildung 9: Entwicklerkonsole Android Market - Übersicht zu Anwendungsfehlern	14

13. Literaturverzeichnis

- AdMob Google Inc. (Juni 2011). *AdMob Google Inc.* Abgerufen am 17. Juni 2011 von <http://developer.admob.com/>: <http://developer.admob.com/wiki/Android>
- AndroLib.com. (13. Juni 2011). *AndroLib.com*. Abgerufen am 13. Juni 2011 von <http://www.androlib.com/>: <http://www.androlib.com/appstatsfreepaid.aspx>
- AndroLib.com. (13. Juni 2011). *AndroLib.com*. Abgerufen am 13. Juni 2011 von <http://www.androlib.com/>: <http://www.androlib.com/appstats.aspx>
- AndroLib.com. (13. Juni 2011). *AndroLib.com*. Abgerufen am 13. Juni 2011 von <http://www.androlib.com/>: <http://www.androlib.com/appstatstype.aspx>
- AndroLib.com. (17. Juni 2011). *AndroLib.com*. Abgerufen am 17. Juni 2011 von <http://www.androlib.com/>: <http://www.androlib.com/appstatsdownloads.aspx>
- AppBrain.com. (Juni 2011). *AppBrain.com*. Abgerufen am 13. Juni 2011 von <http://www.appbrain.com/>: <http://www.appbrain.com/stats/android-market-app-categories>
- Becker, A., & Pant, M. (2009). Android - Grundlagen der Programmierung. In A. Becker, & M. Pant, *Android - Grundlagen der Programmierung* (S. 15-16). Heidelberg: dpunkt.verlag GmbH.
- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 13. Juni 2011 von <http://developer.android.com/>: <http://developer.android.com/sdk/installing.html>
- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 17. Juni 2011 von <http://code.google.com/>: <http://code.google.com/intl/de-DE/mobile/ads/docs/android/fundamentals.html>
- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 17. Juni 2011 von <http://developer.android.com/>: <http://developer.android.com/guide/market/billing/index.html>

- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 13. Juni 2011 von <http://www.google.com/http://www.google.com/support/androidmarket/developer/bin/answer.py?hl=de&answer=113468>
- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 13. Juni 2011 von <http://www.google.com/http://www.google.com/support/androidmarket/developer/bin/answer.py?hl=de&answer=138294>
- Google Inc. (Juni 2011). *Google Inc.* Abgerufen am 13. Juni 2011 von <http://www.google.com/http://www.google.com/support/androidmarket/developer/bin/answer.py?hl=de&answer=112622>
- Janssen, J.-K. (Mai 2011). Amazons Appstore für Android. *c't Praxis Android*, S. 54-55.
- Jeronimo, F. (07. März 2011). *IDC*. Abgerufen am 13. Juni 2011 von <http://www.idc.com/http://www.idc.com/about/viewpressrelease.jsp?containerId=prUK22729011§ionId=null&elementId=null&pageType=SYNOPSIS>
- Kaliudis, A. (2011). Das Wesen der Androiden. *connect 6/2011*, 34.
- Oschatz, A. (17. Dezember 2010). *Mobile Metrics, Alexander Oschatz*. Abgerufen am 15. Juni 2011 von <http://mobilemetrics.de/http://mobilemetrics.de/2010/12/17/infografik-neue-kennzahlen-zum-app-store/>
- research2guidance. (08. Juni 2011). *research2guidance*. Abgerufen am 15. Juni 2011 von <http://www.research2guidance.com/http://www.research2guidance.com/shop/index.php/android-market-insights-april-2011>
- Roehlinger, F. (10. Juni 2011). *Fonpit AG*. Abgerufen am 15. Juni 2011 von <http://www.androidpit.de/http://www.androidpit.de/de/android/blog/395737/Android-Market-Insights-AndroidPIT-ist-fuenftgrosster-Independent-App-Store-auf-der-Welt>
- Sayed Y. Hashimi, S. K. (2010). Pro Android 2. In S. K. Sayed Y. Hashimi, *Pro Android 2* (S. 3-5). New York: Springer-Verlag New York.