

Masterarbeit

Florian Miess

26. September 2009



Hochschule Darmstadt
- Fachbereich Informatik -

Thema der Masterarbeit

Ein generisches Framework für mobile Umfragen: Konzept und Realisierung, Distributionsvarianten und Nutzerakzeptanz

Abschlussarbeit zur Erlangung des akademischen Grades
Master of Science (M.Sc.)

vorgelegt von
Florian Miess

Referent: Prof. Dr. Erbs
Korreferent: Prof. Dr. del Pino

Ausgabedatum: 31.03.2009
Abgabedatum: 30.09.2009

Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die im Literaturverzeichnis angegebenen Quellen benutzt habe.

Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder noch nicht veröffentlichten Quellen entnommen sind, sind als solche kenntlich gemacht. Die Zeichnungen oder Abbildungen in dieser Arbeit sind von mir selbst erstellt worden oder mit einem entsprechenden Quellennachweis versehen.

Diese Arbeit ist in gleicher oder ähnlicher Form noch bei keiner anderen Prüfungsbehörde eingereicht worden.

Darmstadt, den 26. September 2009

Florian Miess¹

¹Florian.Miess@sinnix.de

Abstrakt

Die Arbeit beschreibt die Entwicklung eines Frameworks, um Umfragen mittels Mobiltelefonen vornehmen zu können. Darüber hinaus beleuchtet sie verschiedene Distributionsvarianten und geht auf die Akzeptanz mobiler Umfragen ein. Mobile Umfragen bieten eine Reihe von Vorteilen gegenüber analogen Methoden wie eine Vermeidung von Medienbrüchen oder die orts- und zeitunabhängige Beantwortung von Fragen. Das Framework erlaubt die generische Erstellung und Bearbeitung von Umfrage-Beschreibungen. Dabei werden Umfragen an einem Desktop-Rechner erstellt und an Mobiltelefone verteilt. Auf den Mobiltelefonen können dann die Fragen beantwortet werden und werden danach zurück geschickt. Zur Verteilung wird Bluetooth eingesetzt, welches ein Push-Verfahren nutzt. Rückantworten werden über eine Datenverbindung als SQL-Befehl verschickt. Falls keine derartige Verbindung möglich ist, kann auf SMS zurückgegriffen werden. Die Nutzerakzeptanz ist verglichen mit analogen Methoden geringer, jedoch ist die Qualität der Kommunikation besser, außerdem ist eine Superdistribution möglich. Insbesondere bei amtlichen Befragungen können die Vorteile von mobilen Umfragen eine Erleichterung für die Teilnehmer darstellen.

This thesis describes the design of a framework for developing questionnaires and filling out them on a mobile phone. Mobile questionnaires have some benefits compared with analog polls: They avoid media disruptions and allows location- and time-independet answering. The chosen implementation has some advantages compared to existing applications like saving of costs. The framework enables generic creationing and editing of the questionnaire descriptions. New questionnaires are build within a desktop application and then are shared to mobile phones. Answers are given in through the mobile phone and then the answers are send back. Sharing is done through bluetooth wich uses a push mode. Replys from the questionnaire are send as a SQL command through or data connection if available, elsewise as SMS. The user acceptance is lower compared to analog methods, but the quality of the communication is better and a superdistribution is possible. Particularly with regard to official polls are mobile questionnaires a benefit for the interviewee.

Inhaltsverzeichnis

Abbildungsverzeichnis	6
Tabellenverzeichnis	7
1. Einleitung	8
1.1. Vorwort	8
1.2. Motivation	9
1.3. Aufgabenstellung	10
2. Grundlagen	12
2.1. Mobile Marketing	12
2.2. Charakteristiken heutiger Mobiltelefone	19
2.3. JavaME	23
2.4. Bluetooth	28
3. Anforderungen	31
4. Stand der Forschung	34
5. Entwurf eines generischen Frameworks für mobile Umfragen	37
5.1. Generik des Frameworks	38
5.2. Use-Cases	38
5.3. Architekturentwurf des Framework	41
5.3.1. Entwurf des Clientsystems	41
5.3.2. Entwurf des Serversystems	44
5.3.3. Subsysteme des Frameworks	45
5.4. Spezifikation des Framework	46
5.4.1. Client-Software	49

5.4.2. Designer-Frontend	50
5.4.3. Empfangs-Server	54
5.4.4. SMS-Hub	58
5.5. Entwurf des Datenmodells	59
6. Implementierung des generischen Frameworks	62
6.1. Client-Software	62
6.2. Designer-Frontend	65
6.3. Empfangs-Server	69
6.4. SMS-Hub	70
7. Validierung des Framework	73
8. Distributionsvarianten und Nutzerakzeptanz	76
8.1. Rechtliche Grundlagen zu Distributionsvarianten und dem Datenschutz .	76
8.2. Fallstudien	80
8.2.1. Fallstudie Hobit 2009	80
8.2.2. Fallstudie intern am Fachbereich	82
8.2.3. Fallstudie IT-Forum 2009	84
8.3. Distributionsvarianten	87
8.4. Nutzerakzeptanz	93
8.5. Zusammenfassung und Konsequenzen der Distributionsvarianten und Nutzerakzeptanz	97
8.6. Ethische Betrachtung	99
9. Zusammenfassung und Ausblick	101
9.1. Zusammenfassung	101
9.2. Fazit	103
9.3. Ausblick	104
A. Anhang	105
A.1. Auswertung Fallstudie FBI	106
A.2. Statistiken h_da SS09	110
Literaturverzeichnis	111

Abbildungsverzeichnis

1.	Bluetooth-Hotspot mit integrierter Sendestation	15
2.	Bluetooth-Hotspot in einem Autohaus	16
3.	Weltweite Marktanteile von Mobiltelefonen	20
4.	Weltweite Marktanteile von Smartphones	20
5.	Beziehung zwischen JavaME und JavaSE	24
6.	Bluetooth-Piconetz	29
7.	Use-Case des Clients	39
8.	Use-Case des Servers	40
9.	Die einzelnen Systeme des Framework	47
10.	Zu erstellende Subsysteme des Frameworks	48
11.	Klassen der Client-Software Teil 1	51
12.	Klassen der Client-Software Teil 2	52
13.	Klassen zur Datenübertragung in der Client-Software	53
14.	Klassen des Designer-Frontends Teil 1	55
15.	Klassen des Designer-Frontends Teil 2	56
16.	Klassen des Empfangs-Server	57
17.	Klassen des SMS-Hub	59
18.	Datenmodell	60
19.	Aufbau einer Beispiel-Umfrage	61
20.	Beispiel-Form mit einer Choicegroup	63
21.	Geöffnete Umfrage im Designer-Frontend	67
22.	Bluetooth-Sendestation für den Nahbereich	80
23.	Als Bluetooth-Sendestation eingesetztes ehemaliges Webkiosk	83
24.	Superdistribution der Umfrage-Anwendung	91

Tabellenverzeichnis

1.	Zusammensetzung des Fachbereichs	83
2.	Angenommene und abgelehnte Verbindungsversuche	84
3.	Ausgelieferte Dateien und die erhaltenen Antwort-Codes	85
4.	Ergebnisse der Bachelor-Master-Absolventen Umfragen	87
5.	Nutzerakzeptanz verschiedener Werbekonzepte	95
6.	Weltweite Marktanteile von Mobiltelefonen	105
7.	Weltweite Marktanteile von Smartphones	105

1. Einleitung

1.1. Vorwort

Die vorliegende Masterarbeit gliedert sich grob in zwei verschiedene Bereiche. Im ersten Bereich wird die Entwicklung eines Frameworks beschrieben. Der zweite Bereich beschäftigt sich mit den Distributionsvarianten und der Nutzerakzeptanz von mobilen Umfragen.

Zunächst werden im ersten Kapitel die wichtigsten Grundlagen zum Verständnis der Masterarbeit beschrieben. Danach werden die Anforderungen genau spezifiziert und es wird ein Überblick über angrenzende Arbeiten und den aktuellen Forschungsstand gegeben. Die Kapitel 5 und 6 widmen sich dem Entwurf und der Implementierung. Abgeschlossen wird dies mit der Validierung des erstellten Frameworks, indem die gestellten Anforderungen mit den implementierten Funktionen verglichen werden.

Zur Betrachtung der Distributionsvarianten und der Nutzerakzeptanz werden zunächst rechtliche Aspekte betrachtet und auf die Anforderungen des Datenschutzes eingegangen, die bei Umfragen und der Verwendung von sehr persönlichen Gegenständen, wie Mobiltelefonen, bestehen. Anschließend werden drei selbst durchgeführte Fallstudien und deren Ergebnisse beschrieben. Danach werden Distributionsvarianten vorgestellt. Anhand der Ergebnisse der Fallstudien wird schließlich die Nutzerakzeptanz betrachtet und ein Fazit der Arbeit gezogen. Abgeschlossen wird dieser Bereich der Masterarbeit mit einer Betrachtung der ethischen Implikationen.

Zur Benutzung der Arbeit empfiehlt es sich zunächst Kapitel [→ 1.3, Seite 10] „Aufgabenstellung“ heranzuziehen. Danach kann man sich das Kapitel [→ 5.1, Seite 38] „Generik des Frameworks“ anschauen, wo der Grundgedanke für ein generisches Framework beschrieben wird. Unterkapitel [→ 5.3.3, Seite 45] „Subsysteme des Frameworks“ und Abbildung

[→ 10, Seite 48] geben einen Überblick aus welchen Teilen das System besteht und wie diese zusammenarbeiten.

Für die nicht-technischen Aspekte der Arbeit empfiehlt es sich das Kapitel [→ 8.5, Seite 97] „Zusammenfassung und Konsequenzen der Distributionsvarianten und Nutzerakzeptanz“ zu lesen.

Für entsprechende Vertiefung verweise ich im Text auf den entsprechenden Abschnitt, z.B. Motivation [→ 1.2, Seite 9] oder auf zugrunde liegende Literatur [Autor Jahr]. Details zur Literatur finden Sie im Literaturverzeichnis (Seite 111 ff.).

1.2. Motivation

Umfragen sind heutzutage allgegenwärtig. Sie sind wichtige Formen der Marktforschung und bestimmen das Handeln von Unternehmen.

Der Brockhaus beschreibt die Marktforschung als: „... die systematische Beschaffung und Auswertung von Informationen über sämtliche ein Unternehmen relevanten Märkte“ [Brockhaus 2005]. Zu diesen Märkten zählen Absatz-, Beschaffungs-, Finanz- und Arbeitsmärkte. Als Gründe für die Informationsbeschaffung werden Entscheidungshilfen zur Beschaffungs- und Absatzpolitik genannt, ebenso wie die Untersuchung der Wirksamkeit der getroffenen Entscheidungen.

Die Verwendung von Mobiltelefonen als Umfragewerkzeug bietet dabei, gegenüber den herkömmlichen Papierfragebögen, mehrere Vorteile:

- Die durchgehend digitale Umfrage vermeidet einen Medienbruch. Die Antworten gelangen direkt in eine zentrale Datenbank und können unmittelbar ausgewertet werden. Erfassungsfehler werden damit vermieden.
- Es entstehen keine Druck- und Papierkosten, da auf Ausdrucke völlig verzichtet werden kann.
- Die Möglichkeit von Verzweigungen von Antworten zu den dazu abhängigen Fragen.

- Die Möglichkeit direkt ausgewertete Umfrageergebnisse an die Teilnehmer zurück liefern zu können.
- Der Teilnehmer der Umfrage ist sowohl ortsungebunden, als auch zeitlich unabhängig: Der Befragte kann selbst entscheiden wann und wo er die Umfrage beantwortet.

Eine normale papiergebundene Umfrage zeichnet sich durch folgenden Ablauf aus:

1. Eine Umfrage wird erstellt.
2. Die Umfrage wird an mögliche Teilnehmer verteilt.
3. Die Teilnehmer füllen die Umfrage aus.
4. Die Antwort zur Umfrage wird an die Fragenden zurück gegeben.
5. Die Antworten werden in einer Datenbank erfasst und gegebenenfalls anonymisiert.
6. Die Fragenden werten die Antworten zur Umfrage aus.

Auch eine mobile Umfrage folgt diesem prinzipiellen Ablauf, kann jedoch einen Teil dieser Punkte vereinfachen. Je nach Art der Implementierung ist es z.B. möglich die Auswertung der Umfrage oder die Verteilung der Umfrage zu automatisieren

Gegenwärtig existieren verschiedene Plattformen für mobile Umfragen. Gemeinsam ist diesen Plattformen, dass sie das Internet nutzen oder SMS verwenden. So muss der Anwender die Umfrage entweder mit einem Browser durchführen, oder er erhält einzelne Fragen als SMS auf sein Mobiltelefon zugestellt und beantwortet diese nacheinander. Die Verwendung einer dedizierten Software auf mobile Geräte wie Telefon, MDA oder Laptop bietet einen neuen Ansatz gegenüber bisherigen Lösungen.

1.3. Aufgabenstellung

Diese Arbeit untersucht die Möglichkeiten der Durchführung von Umfragen mittels dedizierter Software für Mobiltelefone und prüft die Akzeptanz dieser neuen Technik.

Die Umfragen sollen einfach zu erstellen sein. Damit keine Programmierkenntnisse zur Erstellung von Umfragen benötigt werden, soll ein generisches Framework entwickelt werden. Ein Framework ist kein eigenständiges Programm, sondern stellt ein Gerüst

zur Entwicklung von Programmen bereit. In der vorliegenden Arbeit wird der Begriff synonym für das Gesamtsystem mit seinen verschiedenen Subsystemen verwendet. Das Framework soll eine Beschreibungssprache nutzen, welches aus den Beschreibungen die Umfragen generiert.

Dazu soll eine Anwendung neue Umfragebeschreibungen generieren und eine weitere Anwendung diese Umfragen auf Mobiltelefonen darstellen.

Damit erstellte Beispiel-Umfragen werden zur Betrachtung der Akzeptanz der Nutzer zur Teilnahme an mobilen Umfragen genutzt.

Innovativ an dieser Arbeit ist die explizite Nutzung von JavaME und Bluetooth zur Durchführung von Umfragen auf Mobiltelefonen. Dabei werden die Umfragen mittels Bluetooth an alle vorbei laufenden Personen rund um eine Bluetooth-Sendestation verteilt. Bisherige mobile Umfragen wurden dagegen entweder über den Browser der Mobiltelefons durchgeführt, oder eine Reihe von Einzelfragen wurde als SMS an die Teilnehmer einer Umfrage verschickt.

2. Grundlagen

Dieses Kapitel beschreibt die Grundlagen, die zum Verständnis der Arbeit wichtig sind. Zunächst wird der Begriff des „Mobile Marketing“ erklärt, da mobile Umfragen als eine Untermenge des Mobile Marketing gesehen werden können. Darüber hinaus wird die aktuelle Verbreitung des Mobile Marketing und die damit einhergehende steigende Verwendung von Mobiltelefonen für weitergehende Dienste geklärt. Anschließend werden die speziellen Eigenschaften von Mobiltelefonen untersucht, da ihre Charakteristiken als Zielgeräte entscheidend für das zu erstellende Framework sind. Hier werden auch die unterschiedlichen Kommunikationsmöglichkeiten von Mobiltelefonen vorgestellt. Danach wird ein kurzer Überblick über JavaME als Programmiersprache für Mobiltelefone gegeben. Abgeschlossen wird das Grundlagen-Kapitel mit einem Überblick über Bluetooth als spezielle Kommunikationsmöglichkeit, da diese Technik als Grundlage benötigt wird.

2.1. Mobile Marketing

Zum Begriff des Mobile Marketing sagen Pousttchi und Wiedemann: „Mobile Marketing [ist] ein operatives Instrument der Marketing-Kommunikation, das mobile elektronische Kommunikationstechnologien (in Verbindung mit mobilen Endgeräten) nutzt, um Güter, Dienstleistungen oder Ideen zu fördern.“ [Dr. Pousttchi und Wiedemann 2006]. Mobile Marketing beschreibt also in erster Linie das Werben auf Mobiltelefonen mittels verschiedener Konzepte. Als synonyme Begriff wird oft „Mobile Advertising“ gebraucht, obwohl dies nur eine Teilmenge des Mobile Marketing ist. So beschreibt der Begriff des Mobile Marketing nämlich zusätzlich die Möglichkeit von Werbung in Bewegung - also z.B. mobile Plakatwände. In dieser Arbeit werden Mobile Marketing und Mobile Advertising im Sinne von mobilen Empfangsgeräten synonym verwendet.

Das Mobile Marketing untergliedert sich in mehrere Konzepte [Kizilok 2007, S. 22ff.]:

SMS und MMS Die Verwendung von SMS¹ und MMS², um direkt Werbung an die Endgeräte der Nutzer zu senden.

Werbung in mobilen Portalen und Communities Hier definiert das Portal oder die Community die Zielgruppe für die Werbung. Z.B. werden dabei Nutzer über die Portale der Diensteanbieter geleitet, wenn sie im Internet Seiten aufrufen.

Cross-Media-Maßnahmen Hierunter versteht man Maßnahmen, bei denen Konsumenten mittels anderer Werbeträger wie Radio und TV auf die Online-Werbung aufmerksam gemacht werden.

Stand-Alone-Maßnahmen Das sind eigenständige Inhalte die durch Cross-Media-Maßnahmen beworben werden.

Werbung auf mobilen Suchmaschinen Werbung innerhalb von Suchmaschinen durch angepasste Zusatzinformationen oder Suchergebnisse (z.B. Google Adds).

Location Based Services Die Werbung ist hier abhängig vom Ort des Benutzers. Eine Möglichkeit hierfür bietet Bluetooth durch seine Nahstrecken-Funktechnik oder die netzbezogene Aufbereitung von Informationen (IP, Netz-Zelle, ...).

Ein Hauptaugenmerk liegt heute auf der Verwendung von SMS und MMS, da diese eine hohe Reichweite und starke Mobilität bieten. Die hohe Reichweite ist damit begründet, dass SMS bzw. MMS an nahezu jedem Ort verfügbar ist, da es die bestehenden Mobilfunknetze nutzt, die eine hohe Abdeckung bieten. Die starke Mobilität ergibt sich durch die Möglichkeit den eigenen Standort beliebig wechseln zu können, ohne sich um die Mobilfunkverbindung kümmern zu müssen. So besteht bei der Verwendung anderer Technologien wie WLAN (Wireless Local Area Network) die Schwierigkeit beim Wechseln von einem Access-Point³ zum nächsten darin, dass man sich in das neue Netz erst einwählen muss - der *Seamless*-Übergang funktioniert hier nicht.

Ein weiteres Hauptaugenmerk liegt auf der Verwendung von Bluetooth [→ 2.4, Seite 28], um Marketing zu betreiben. Diese spezielle Form wird manchmal auch als „Bluetooth Marketing“ bezeichnet [Hammel u. a. 2007]. Bluetooth bietet dabei den Vorteil, dass es

¹SMS: Abk. f. Short Message Service

²MMS: Abk. f. Multimedia Messaging Service

³Access-Point - Ein Zugangspunkt um sich mit dem Internet verbinden zu können.

einen hohen Verbreitungsgrad auf Mobiltelefonen hat [→ 2.4, Seite 28] und beliebige Inhalte übertragen werden können. Die Reichweite einfacher Sender bis 10 Meter bietet den Vorteil standortabhängige Inhalte zu ermöglichen (Location Based Services, siehe oben).

Die Inhalte werden dabei mittels Bluetooth-Sender (Hotspots) verteilt. So können Bluetooth-Sender zum Beispiel in Postervitrinen an Bushaltestellen, Einkaufszentren oder Kinos eingebaut sein. Diese suchen nach Geräten in der unmittelbaren Umgebung, die über eine angeschaltete und als „sichtbar gesetzte“ Bluetooth-Schnittstelle verfügen. Werden die Sender fündig, bieten sie diesen Geräten automatisch zumeist Werbeinhalte an. Als Inhalte kommen alle möglichen Formate in Frage: Bilder, Klingeltöne, Videos oder auch Handyspiele. Der „angesprochene“ Gerätebesitzer muss dann das Empfangen dieser Inhalte bestätigen oder ablehnen. Zielgeräte sind dabei primär Mobiltelefone in der Zielgruppe der 14- bis 39-jährigen. [Hammel u. a. 2007]

Bluetooth-Marketing kann eingesetzt werden, um Werbung an potentielle neue Kunden zu verteilen oder um Zusatzinformationen zu bestimmten Themen anzubieten. So verteilen bereits heute Bluetooth-Hotspots in vielen Kinos Werbung, indem sie aktuelle Trailer neuer Kinofilme zum Herunterladen anbieten. Zusatzinformationen hingegen werden z.B. in Autosalons eingesetzt, um interessierten Betrachtern eines Objektes weitergehende Informationen anzubieten, z.B. zusätzliche Bilder des Fahrzeugs in anderen Farben oder Videos über die Fahreigenschaften (Bild [→ 2, Seite 16]).

Um Passanten auf die Bluetooth-Werbung aufmerksam zu machen, sind häufig Cross-Media-Maßnahmen erforderlich. Dies geschieht z.B. durch Plakate oder Leuchtsäulen, die auf einen Bluetooth-Hotspot hinweisen. Dabei kann der eigentliche Bluetooth-Baustein bereits in die entsprechende Werbesäule integriert sein. Abbildung [→ 1, Seite 15] zeigt einen solchen Bluetooth-Hotspot des Unternehmens Blue Cell Networks⁴.

Laut dem „Cluetrain Manifesto“ existieren verschiedene Erfolgsfaktoren im Umgang mit dem Internet [Levine u. a. 2002]. Toma und Schäfer nehmen sich hierbei mehrere heraus und beleuchten diese im Zusammenhang mit dem Mobile Marketing [Schäfer und Toma 2008]. Insbesondere die These 73 „You’re invited, but it’s our world. Take your shoes off at the door.“ ist beim Mobile Marketing von entscheidender Bedeutung: Un-

⁴Unternehmen Blue Cell Networks GmbH, Bamberg: <http://www.bluecellnetworks.com>

2. Grundlagen



Abbildung 1.: Bluetooth-Werbung mit integrierter Sendestation der Blue Cell Networks (Quelle: Blue Cell Networks GmbH)

aufgefordert zugesandte Daten werden von allen Nutzern als unhöflich und negativ aufgefasst.

Bluetooth Marketing nutzt im weitesten Sinne eine Pull-Push-Kombination, um Nutzer anzusprechen [→ 2.4, Seite 28]. Da die Verwendung des Push-Verfahrens in Deutschland unzulässig ist, werden Passanten nur innerhalb eines bestimmten Umkreises angesprochen [→ 8.1, Seite 76].

Die Verwendung des Opt-in-Verfahrens bietet neben dem Nachteil weniger Menschen zu erreichen einen entscheidenden Vorteil: Nimmt eine Person das Angebot, Daten zu empfangen, an, kann man fast sicher sein, dass die Person auch am Produkt interessiert ist. Bluetooth Marketing bedeutet also Kommunikation auf Wunsch, wodurch eine sehr hohe Kontaktqualität erreicht wird und Streuverluste extrem minimiert werden. Hammel, Sassenberg und Scholz haben dazu eine Tabelle mit weiteren Vor- und Nachteilen des Bluetooth-Marketing erstellt [Hammel u. a. 2007, S. 10].

Im Bereich des Mobile Marketing ist die MMA⁵ maßgebend. Sie erstellt Richtlinien zur Vorgehensweise bei Mobile Marketing-Kampagnen. In Deutschland sind bisher zehn Unternehmen Mitglieder des Verbandes.

⁵MMA: Abk. f. Mobile Marketing Association: <http://mmaglobal.com/main>

2. Grundlagen



Abbildung 2.: Bluetooth-Hotspot der Hiwave in einem Autohaus (Quelle: <http://www.newsmax.de>)

Um erfolgreich Mobile Marketing betreiben zu können, stellten Bauer, Neumann und Reichardt neun Punkte auf, die für eine Akzeptanz von Mobile Marketing zutreffen müssen [Bauer u. a. 2008]:

1. Je positiver die persönliche Einstellung zu Mobile Marketing, desto höher die Verhaltensabsicht zur Nutzung von Mobile Marketing.
2. Je positiver die subjektive Wahrnehmung von sozialen Normen hinsichtlich der Nutzung von Mobile Marketing, desto höher die Verhaltensabsicht zur Nutzung von Mobile Marketing.
3. Je positiver die subjektive Wahrnehmung von sozialen Normen hinsichtlich der Nutzung von Mobile Marketing, desto positiver die Einstellung zum Mobile Marketing.
4. Je größer die Innovationsbereitschaft, desto größer das vorhandene Wissen über Mobilkommunikation.

5. Je größer das vorhandene Wissen über Mobilkommunikation, desto positiver die Einstellung zum Mobile Marketing.
6. Je stärker ausgeprägt das „Information-Seeker“-Verhalten, desto positiver die Einstellung zur Werbung im Allgemeinen.
7. Je positiver die Einstellung zur Werbung allgemein, desto positiver die Einstellung zum Mobile Marketing.
8. Je größer der wahrgenommene Informations-, Unterhaltungs- und soziale Nutzen, desto positiver die Einstellung zum Mobile Marketing.
9. Je höher das wahrgenommene Risiko, desto negativer die Einstellung zum Mobile Marketing.

Die Akzeptanz von Mobile Marketing ist also abhängig von der persönlichen Einstellung jedes Einzelnen gegenüber dieser Form der Werbung.

Laut Hammel, Sassenberg, Scholz und Rösch wird das Mobile Marketing gegenwärtig am häufigsten eingesetzt, um Kunden weitere Informationen zu bestimmten Produkten bieten zu können [Hammel u. a. 2007] [Rösch 2006].

Zur Verteilung von Zusatzinformationen unterscheidet man zwei Benutzergruppen: Die geschlossene Benutzergruppe und die nicht-geschlossene Benutzergruppe. Geschlossene Benutzergruppen sind Gruppen innerhalb eines eng umgrenzten Bereichs, z.B. nicht-öffentliche Veranstaltungen, also Veranstaltungen für Mitglieder oder Teilnehmer. In diesem Fall kann die Gruppe mit spezifischen Informationen versorgt werden. Kinobesucher und Messeteilnehmer gehören zu solchen geschlossenen Gruppen. Hier kommt mobile Marketing mittels Bluetooth zum Einsatz, da es die Möglichkeit bietet, zusätzliche Informationen zu Produkten an interessierte Besucher vermitteln zu können, z.B. weitergehende Informationen zu Kinofilmen wie Trailer oder Klingeltöne.

Die nicht-geschlossene Benutzergruppe hingegen bezeichnet Nutzer, die überall angesprochen werden können - z.B. auf Flughäfen, Bahnhöfen oder in Einkaufszentren - und die keiner speziellen Gruppe angehören. An diese werden häufig Werbe-SMS oder mobile Gewinnspiele geschickt, um allgemein auf ein Produkt aufmerksam zu machen. Der Grund hierfür ist, dass keine Informationen über spezielle Interessen der Personen vorliegen und deswegen auch keine besonderen Informationen angeboten werden können [Dufft 2003,

S. 25f.]. Auch Massenversand-SMS, bei denen eine SMS an alle Mobiltelefone in einer bestimmten Zelle versendet werden, zählen zum Mobile Marketing für nicht-geschlossene Benutzergruppen.

Die gesamten Investitionen für Mobile Marketing in Deutschland betragen laut einer Studie der Post AG 30 Mio. Euro im Jahr 2007. Die Investitionen stiegen damit von 10 Mio. Euro im Jahre 2005 um 300% und beweisen die zunehmende Bedeutung von Mobiltelefonen im Marketing-Bereich. Allerdings nutzen bisher erst etwa 8000 Unternehmen in Deutschland das Mobile Marketing, das sind knapp 1% aller Unternehmen in Deutschland. [Post AG 2007]

Laut Ahonen existieren sieben Aspekte bei Mobiltelefonen, die das Mobile Marketing einzigartig machen [Ahonen 2008]:

1. Mobiltelefone sind das erste „persönliche“ Massenmedium überhaupt.
2. Es ist das einzige Medium, welches „immer“ mitgenommen wird.
3. Mobiltelefone sind „immer“ eingeschaltet.
4. Kein anderes Medium hat eine eingebaute Bezahl-Funktion.
5. Nur das Mobiltelefon ist „immer“ zur Stelle, wenn man eine kreative Inspiration hat.
6. Mit keinem anderen Medium kann eine Zielgruppe so exakt bestimmt werden.
7. Nur das Mobiltelefon kann zusätzlich den sozialen Kontext des Medienkonsums messen.

Mobile Marketing ist ein stark wachsendes Segment im Marketing. Es verdeutlicht, dass bereits heute die meisten Menschen ihr Mobiltelefon für mehr Verwenden, als nur zum Telefonieren und SMS-Versenden.

2.2. Charakteristiken heutiger Mobiltelefone

Der Brockhaus definiert ein Mobiltelefon als ein „ortsungebundenen, batteriebetriebenes Funktelefon, das zur Sprach- und Datenkommunikation ein entsprechendes Mobilfunknetz nutzt, allgemeinsprachlich Handy. . . . sie können u.a. mit Sprachsystemen, mehrzeiligen und farbigen Displays bzw. Touchscreens ausgestattet sein“ [Brockhaus 2005]. Mobiltelefone die dem Benutzer neben dem reinen Telefonieren und SMS-Versenden PIM-Anwendungen⁶ zur Verfügung stellen, werden als Smartphones bezeichnet. Diese Smartphones bieten oftmals weitere Funktionen, wie das Abspielen von Multimedia-Inhalten und das Surfen im Internet. Allen Mobiltelefonen gemeinsam ist die SIM-Karte⁷, die zur Identifizierung des Teilnehmers im Mobilfunknetz dient [Brockhaus 2005, SIM-Karte]. Auf ihr sind alle relevanten Daten des Teilnehmers gespeichert wie Rufnummer, PIN, Geräteerkennung etc.. Mittels der IMSI-Nummer⁸ ist die SIM-Karte in einem Mobilfunknetz eindeutig identifizierbar, wodurch auch eine eindeutige Zuordnung zu einem bestimmten Teilnehmer ermöglicht wird. Aus dieser Bindung an einen spezifischen Nutzer resultiert, dass Mobiltelefone sehr persönliche Gegenstände sind, die für jedwede private Kommunikation genutzt werden.

Zurzeit existieren viele verschiedene Geräte auf dem Markt. Die Anteile der jeweiligen Geräte einzelner Hersteller auf dem weltweiten Markt sind dem Diagramm [→ 3, Seite 20] zu entnehmen (Tabelle siehe Anhang [→ 6, Seite 105]).

Die Marktanteile bei den Smartphones zeigt Diagramm [→ 4, Seite 20] (Tabelle siehe Anhang [→ 7, Seite 105]). Aus beiden Diagrammen ist ersichtlich, dass Nokia den mit Abstand größten Marktanteil auf der Welt besitzt. Alle Daten beruhen auf einer Studie der Gartner Incorporated von 2009 [Stevens und Pettey 2009].

2008 lag die Penetrationsrate von Teilnehmer in den Mobilfunknetzen, also die Zahl der Mobiltelefonbesitzer, in Deutschland bei über 130%, wobei ein Nutzer auch mehrere Vertragsverhältnisse haben kann [Henn u. a. 2008, S. 77ff.]. Auf Mobiltelefone als Zielmedium

⁶PIM: Abk. f. Personal Information Management: Klassische Anwendungen eines PDA (Personal Digital Assistend) bestehend aus Kalendersoftware, ToDo-Listen und Adressbuch

⁷SIM: Abk. f. Subscriber Identity Module

⁸IMSI: Abk. f. International Mobile Subscriber Identity

2. Grundlagen

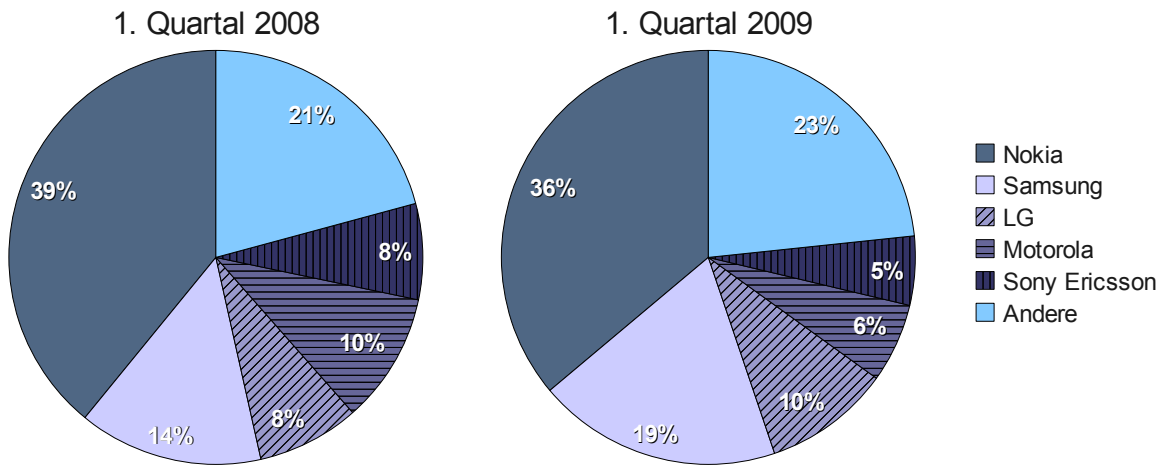


Abbildung 3.: Weltweite Marktanteile von Mobiltelefonen (Quelle: Gartner)

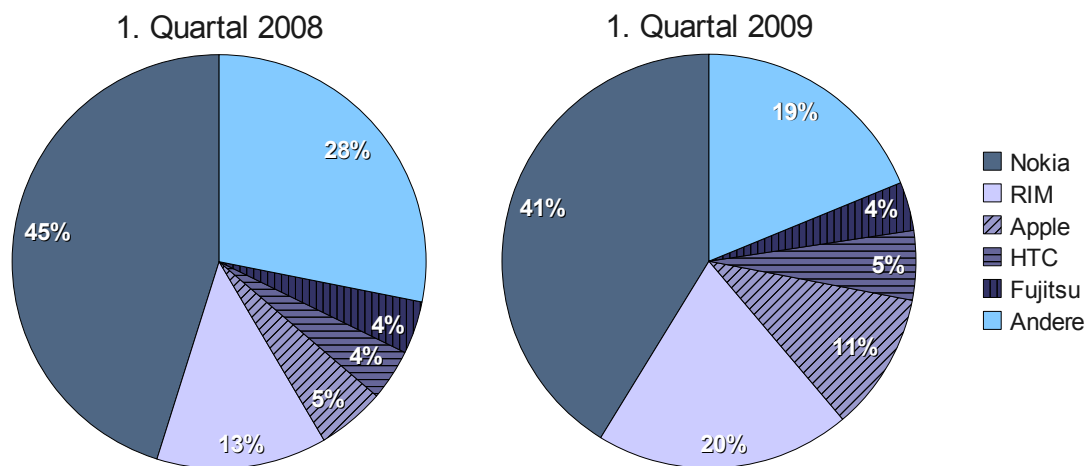


Abbildung 4.: Weltweite Marktanteile von Smartphones (Quelle: Gartner)

zu setzen erscheint daher sehr sinnvoll. Alle Mobiltelefone unterliegen jedoch einigen Einschränkungen. Obwohl die Geräte immer leistungsfähiger werden, sind die Ressourcen, im Gegensatz zu stationären Geräten wie PCs, gering. Schwache Hardware-Plattformen, kleine Displays, geringer Speicher und begrenzte Eingabemöglichkeiten sind nur einige dieser Beschränkungen. Hierzu kommen eingeschränkte Übertragungskapazitäten, da die Kommunikation der Endgeräte derzeit hauptsächlich durch die GSM-Netze der Mobilfunkanbieter erfolgt. Erst UMTS stellt wesentlich höhere Datenraten zur Verfügung. [Miess und Seifert 2007]

Zu den Charakteristiken von Mobiltelefonen zählen daher:

- Geringe Displaygrößen. Die Größen variieren zwischen 240x320 Pixeln und 360x640 Pixeln.
- Geringe Rechenleistung. Verglichen mit handelsüblichen Computern besitzen Mobiltelefone noch eine geringere Rechenleistung, auch wenn diese von Gerät zu Gerät stark schwanken kann.
- Unterschiedliche Eingabemethoden. Neben der klassischen Telefon-Tastatur mit 12 Tasten existieren auch vollwertige QWERTY-Lösungen. Darüber hinaus gibt es Geräte mit Touchscreens die z.T. mit Stiften, z.T. mittels Fingern oder beidem bedient werden können. Einige Geräte lassen sich zudem mittels Stimme steuern.
- Verschiedenste Übertragungstechniken. Angefangen bei GSM⁹ über Bluetooth bis WLAN.

Die bei allen Mobiltelefonen in Europa eingesetzten Funktechniken sind GSM oder UMTS¹⁰. Diese basieren auf der Einteilung von Gebieten in Funkzellen. In diesen Zellen werden einzelne Teilnehmer von Richtantennen angesprochen. Um mehrere Teilnehmer gleichzeitig in einer Zelle zu ermöglichen, existieren verschiedene Multiplex-Techniken [Schiller 2003, S. 63-69]:

Raummultiplexing Bei dieser Technik sind die Empfänger räumlich weit voneinander getrennt, so dass keine Interferenzen entstehen.

⁹GSM: Abk. f. Global System for Mobile Communications

¹⁰UMTS: Abk. f. Universal Mobile Telecommunications System

Frequenzmultiplexing Die verschiedene Empfänger werden mit unterschiedlichen Frequenzen angesprochen, können sich also im selben Gebiet befinden.

Zeitmultiplexing Jeder Empfänger bekommt hier einen Zeit-Slot zum Senden und Empfangen zugeteilt. Die einzelnen Geräte werden dazu miteinander synchronisiert, wobei unterschiedliche Entfernung eine unterschiedliche Signallaufzeit zur Folge haben.

Codemultiplexing Hier werden für die einzelnen Empfänger orthogonale Codes erzeugt mit denen diese angesprochen werden. Das bedeutet, dass ein gesendeter Datenstrom gleichzeitig mehrere Nachrichten für unterschiedliche Teilnehmer enthält und diese Nachrichten jeweils von den Empfängern mit dem passenden Code herausdecodiert werden können.

Um die eigentlich digitalen Informationen mittels analoger Funkwellen übertragen zu können, existieren verschiedene Modulationstechniken: Die Amplitudenumtastung, Frequenzumtastung und Phasenumtastung, sowie Kombinationen davon. Hinzu kommen Bandspreizverfahren, um den Datendurchsatz zu erhöhen und die Fehleranfälligkeit zu senken [Schiller 2003, S. 69-87]:

Direkt-Sequenz-Spreizspektrum (DSSS) Hierbei werden die zu übertragenden Datenbits mit einer speziellen Bit-Sequenz mittels XOR verknüpft.

Frequenzsprung-Spreizspektrum (FHSS) Hier wird ein verfügbarer Frequenzbereich in viele kleine Bereich aufgeteilt.

Gegenwärtig werden in Deutschland Mobilfunknetze der zweiten und dritten Generation betrieben. Zur zweiten Generation gehört das GSM mit seinen D- und E-Netzen. Es ist gegenüber der ersten Netzgeneration nicht mehr analog, sondern digital. Allerdings unterstützte es anfangs nur Übertragungsbandbreiten bis 9,6 Kbit/s, weswegen mehrere Erweiterungen entwickelt wurden: Z.B. GPRS¹¹ mit einer Übertragungsrate von 21,4 Kbit/s pro Zeitschlitz. Die dritte Generation ist das UMTS-Netz. Auch dieses hat mittlerweile Erweiterungen erfahren, so dass mit HSDPA¹² nun Übertragungsraten bis 7,2 Mbit/s erreicht werden, statt anfangs 384 Kbit/s.

¹¹GPRS: Abk. f. General Packet Radio Service

¹²HSDPA: Abk. f. High Speed Downlink Packet Access

Neben diesen Funknetzen unterstützen viele Mobiltelefone weitere Übertragungstechniken, angefangen bei Infrarot (IrDA) über Bluetooth bis WLAN.

Es ist schwierig Gemeinsamkeiten aller Mobiltelefone zu finden, bereits beim Punkt der Eingabemöglichkeiten existieren sehr unterschiedliche Ansätze. Mobiltelefone sind unterschiedlich aufgebaut und besitzen unterschiedliche Betriebssysteme. Sie bieten damit auch unterschiedliche Möglichkeiten der Programmierung:

- C++. Für Symbian-Geräte, Palms etc.. Die Bibliotheken für diese Programmiersprache sind entsprechend an die Geräte angepasst und unterstützen - je nach Endgerät - unterschiedliche Funktionen. Dementsprechend ist ein Programm für jedes Gerät und dessen Betriebssystem speziell anzupassen und neu zu übersetzen. Die unterstützten APIs¹³ und Funktionen können sich selbst zwischen Geräten eines Herstellers unterscheiden.
- ObjectiveC. Diese Programmiersprache wird ausschließlich von Geräten mit MacOS unterstützt.
- JavaME. Eine abgespeckte Variante der Java-Programmiersprache, wird von den meisten Mobiltelefonen unterstützt (Ausnahmen sind das iPhone und Geräte mit Android als Betriebssystem).

2.3. JavaME

Daneben bietet eine Verwendung von JavaME noch weitere Vorteile [[Schmartz 2007](#), S. 4f.]:

- Öffentliche und dokumentierte Schnittstellen.
- Erweiterbar durch neue Klassenbibliotheken.
- Höheres Abstraktionsniveau als C++ und damit geringere Fehleranfälligkeit.
- Einfache Umstellung von Java-Entwicklern auf JavaME.

¹³API: Abk. f. Application Programming Interface

JavaME (ehemals J2ME) steht dabei für Java Micro Edition und ist eine Programmiersprache von SUN Microsystems. Sie ist eine speziell für Embedded-Geräte angepasste Version von Java und bietet erweiterte APIs zur Verwendung von Mobiltelefonen. Abbildung [→ 5, Seite 24] zeigt die Beziehungen zwischen der Java Standard Edition (JavaSE, ehemals J2SE) und der JavaME.

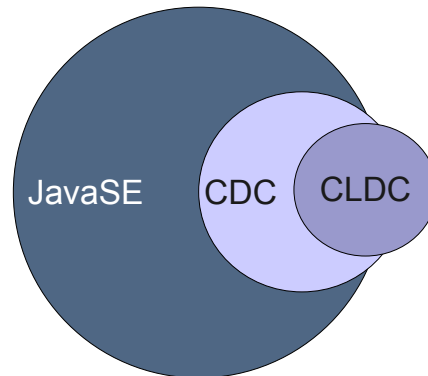


Abbildung 5.: Beziehung zwischen der JavaME CDC und CLDC und JavaSE (Quelle: SUN microsystems)

Dabei besteht JavaME aus zwei Teilen: Der CDC und der CLDC. CDC bedeutet *Connected Device Configuration* und ist eine Spezifikation und Bibliothekensammlung, um grundlegende Funktionen bereit zu stellen. CLDC steht für *Connected Limited Device Configuration* und ist ebenfalls eine Spezifikation und Bibliothekensammlung, um grundlegende Funktionen auf allen Mobiltelefonen zur Verfügung zu stellen. In den *Configurations* sind auch die Virtuellen Maschinen enthalten: Die KVM - Kilobyte Virtual Machine auf Seiten von CLDC und die Connected Device Configuration HotSpot Implementation (früher CVM) auf Seiten von CDC. Eine Virtuelle Maschine ist diejenige Umgebung, in der die JavaME-Programme ablaufen. Sie ist eine Zwischenschicht zwischen Anwendung und Betriebssystem und führt den Bytecode des Programms aus [SUN 2009b].

Gegenwärtig ist Sony-Ericsson der einzige Hersteller von Mobiltelefonen die CDC in einigen wenigen Geräten (P1i und m600) implementieren. Aufgrund der geringen Anzahl an Unterstützern unter den Mobiltelefon-Herstellern muss davon ausgegangen werden, dass CDC keine Zukunft hat. Darüber hinaus werden Mobiltelefone immer leistungsfähiger, so

dass es absehbar ist, dass auch normale JavaSE-Programme irgendwann auf diesen lauffähig sind. Ein erster Schritt in diese Richtung ist das Android-Betriebssystem¹⁴, das eine ähnliche Funktionalität der Java-Programmierung bietet, wie sie JavaSE-Programme benötigen.

Der Unterschied zwischen CDC und CLDC ist der bereitgestellte Funktionsumfang. Damit einher gehen auch unterschiedliche Anforderungen an die verwendeten Geräte, so dass im allgemeinen CDC auf leistungsstärkere Endgeräte wie Settop-Boxen etc. abzielt, während CLDC leistungsschwächere Endgeräte wie einfache Mobiltelefone adressiert. Beiden *Configurations* gemein ist, dass sie allein noch nicht ausreichen, um Java-Programme erstellen zu können. Hinzu kommen speziell an ein entsprechendes Mobiltelefon angepasste Profile, die zusätzliche Funktionen bereit stellen. Auf Seiten von CLDC ist dies MIDP. MIDP steht für *Mobile Information Device Profile* und ist eine Sammlung höherer APIs für spezielle, geräteabhängige Funktionen, um Interfaces und Netzzugänge zu gewährleisten. Die MIDP bietet Java-Klassen an, um z.B. auf das Telefonbuch des Mobiltelefons zugreifen zu können. Ebenso ermöglicht sie den Versand von SMS oder die generelle Kommunikation mit dem Internet. Um derart tiefgreifende Funktionen nutzen zu dürfen, bedarf es der Zustimmung des Betriebssystems. Diese Zustimmung kann entweder händisch durch den Benutzer erfolgen oder automatisch durch eine unabhängige Instanz, die die Korrektheit eines Programms prüft, bevor es an potentielle Nutzer verteilt wird (siehe weiter unten)

Eine mit CLDC/MIDP erstellte Anwendung wird als MIDlet bezeichnet. Ein MIDlet ist eine einzelne Anwendung die von der Oberklasse *MIDlet* erbt und mehrere weitere Java-Klassen besitzen kann. Mehrere MIDlets zusammengefasst bezeichnet man als eine MIDlet-Suite. Diese MIDlet-Suite ist eine in ein *JAR*-Archiv¹⁵ zusammengefasste Sammlung aller Programmklassen und stellt die kleinste installierbare Einheit dar. Das JAR-Archiv besitzt eine Dateisystemstruktur und enthält neben den Klassen der MIDlets auch Ressourcen die die MIDlets benötigen, z.B. Bilder oder Töne. Darüber hinaus sind in ihr Informationen bezüglich der MIDlet-Suite vorhanden wie Name der Suite, Größe und Hersteller. Neben dem JAR-Archiv wird beim kompilieren einer MIDlet-Suite auch eine JAD-Datei¹⁶ erstellt. In dieser sind ebenfalls Informationen bezüglich der MIDlet-Suite

¹⁴<http://www.android.com/about/>

¹⁵JAR: Abk. f. Java Archive

¹⁶JAD: Abk. f. Java Application Descriptor

enthalten. Der Grund für die zusätzliche Erstellung einer Informationsdatei liegt darin, dass mit den darin enthaltenen Informationen meist schon entschieden werden kann, ob die Anwendung überhaupt für den Anwender interessant ist. Der Anwender spart sich so mit das Herunterladen unnötiger oder ungewollter Anwendungen, was sowohl Kosten, als auch Bandbreite und Zeit spart. Darüber hinaus wird in der JAR-Datei eine mögliche Signatur der MIDlet-Suite und das zugehörige Zertifikat gespeichert.

Die Darstellung von Daten innerhalb eines MIDlet kann auf zwei Arten erfolgen: Entweder über eine High-Level-API oder über eine Low-Level-API. Die High-Level-API bietet vordefinierte Darstellungsarten wie Textboxen, Multiple-Choice-Fragen usw., mit denen auf einfache Weise Oberflächen zusammengestellt werden können. Hierbei erbt die darstellende Oberflächen-Klasse von der Klasse *Screen*. Damit können dann Listen, Textboxen, Alarmmeldungen oder *Forms* dargestellt werden. Eine Form stellt einen leeren Bildschirm dar, der weitere Komponenten, sogenannte *Items*, beinhalten kann, z.B. Textfelder, Datumsfelder oder Bilder. Sind diese vorgegebenen Darstellungsmöglichkeiten nicht ausreichend, was z.B. bei der Entwicklung von Spielen häufig der Fall ist, wo besondere grafische Elemente benötigt werden, kann auf die Low-Level-API zurück gegriffen werden. Dann wird von der Klasse *Canvas* geerbt, statt von der Klasse *Screen*. In diesem Fall ist der Entwickler in den Darstellungsarten völlig frei.

Neben der Darstellung von Informationen ist auch die Datenhaltung ein wichtiger Punkt bei vielen Programmen. In JavaME werden zur Speicherung von Daten *RecordStores* verwendet. Das RMS (Recordstore Management System) stellt die Programmierschnittstelle dafür dar. Das RMS arbeitet dabei mit einfachen serialisierten Datensätzen, die in ein RecordStore eingetragen werden. Einen wahlfreien Zugriff erlaubt das RMS jedoch nicht.

JavaME unterstützt auch verschiedene Protokolle, um eine Kommunikation zwischen verschiedenen Geräten und dem Internet ermöglichen zu können. Dazu zählt z.B. HTTP ebenso, wie die Möglichkeit SMS zu schicken und zu empfangen. Ob die benötigten APIs allerdings vorhanden sind, hängt vom Hersteller des entsprechenden Mobiltelefons und dessen CLDC/MIDP-Implementierung ab.

Drüber hinaus ermöglicht JavaME das parallele Abarbeiten verschiedener Aufgaben. Dazu kann der Entwickler sogenannte Threads erstellen die eigenständig laufen und ein Blockierung des Systems verhindern [Tanenbaum 2002, S. 87ff.].

Hat ein Anwender eine MIDlet-Suite (also das JAR-Archiv) auf seinem Mobiltelefon herunter geladen, wird er vom Betriebssystem des Gerätes gefragt, ob er die Anwendung installieren möchte. Stimmt er dem zu, wird der Installationsprozess gestartet. Dazu überprüft das System welche Funktionen des Mobiltelefons vom MIDlet benötigt werden. Diese Informationen sind entweder in der MIDlet-Suite enthalten, oder werden dynamisch durch das System festgestellt - nämlich dann, wenn die Anwendung eine entsprechende Funktion aufruft. Entweder fragt also die Installationsroutine beim Nutzer nach, ob er die nötigen Genehmigungen erteilen möchte, oder die Anwendung erfragt die Erlaubnis dynamisch wenn sie benötigt wird. Dafür leitet die KVM die Anfrage zum Betriebssystem des Mobiltelefons weiter und dieses fragt dann beim Benutzer nach. Die Berechtigungen gelten dabei immer nur für die aktuelle Sitzung, müssen also bei jedem Programmstart erneut gegeben werden.

Eine andere Möglichkeit, um MIDlets den Zugriff auf spezifische Gerätefunktionen zu ermöglichen, besteht darin, die Anwendung als vertrauenswürdig einstufen zu lassen. Dies ist darüber hinaus sinnvoll, weil hierdurch auch der Urheber ersichtlich wird. Außerdem werden dadurch wiederholende Nachfragen zu Berechtigungen vermieden - z.B. ist es wenig sinnvoll einer Browser-Applikation jedes mal die Erlaubnis zur Kommunikation zu geben, schließlich ist dies der Anwendungszweck der Applikation. Zur Einstufung einer MIDlet-Suite als vertrauenswürdig, sind zwei Schritte nötig: Erstens wird der Entwickler authentifiziert, seine Identität also zweifelsfrei nachgewiesen. Zweitens wird die Vertrauenswürdigkeit des Entwicklers, also seine Seriosität, nachgewiesen. Diese Informationen werden dann der MIDlet-Suite mitgegeben.

Hierzu wird im ersten Schritt die MIDlet-Suite mit einer Signatur des Entwicklers versehen. Dazu wird eine Prüfsumme über die JAR-Datei erstellt, diese Prüfsumme danach mit dem privaten Schlüssel des Entwicklers codiert und das Ergebnis in der JAD-Datei eingetragen. Ebenso wird das Zertifikat (Teil des öffentlichen Schlüssels) in der JAD-Datei eingetragen, um die Signatur überprüfen zu können - die JAR-Datei bleibt unverändert. Der private Schlüssel gehört dabei zu einer PKI-Infrastruktur (Public Key Infrastructure), bei der eine unabhängige Instanz Personen als vertrauenswürdig einstuft, z.B. VeriSign¹⁷. Das Verfahren arbeitet dabei mit zwei Schlüsselpaaren: Einem privaten Schlüssel, um Nachrichten zu entschlüsseln und einem öffentlichen Schlüssel, um Nachrichten zu verschlüsseln und die Herkunft zweifelsfrei fest zu stellen (indem das Zertifikat

¹⁷Unternehmen VeriSign, Kalifornien: <http://www.verisign.de/>

im öffentlichen Schlüssel auf eine encodierte Nachricht angewandt wird) [Eckert 2008, S. 385ff.].

Im zweiten Schritt muss der authentifizierte Entwickler als seriös eingestuft werden. Dies geschieht, indem die Gerätehersteller die Signaturen bestimmter Unternehmen als vertrauenswürdig einstufen. Eine andere Möglichkeit besteht darin, die MIDlet-Suite durch das *Java-Verified-Programm* testen und signieren zu lassen. Dabei wird die Anwendung durch diese Instanz verschiedensten Tests unterzogen, um eine einwandfreie Funktion zu überprüfen und versteckte Programmaktivitäten (wie das heimliche Auslesen und Weitergeben privater Informationen) auszuschließen [SUN 2009c].

[Schmartz 2007] [SUN 2009a]

2.4. Bluetooth

Bluetooth ist eine Funktechnologie für kurze Reichweiten bis 100 Meter. Entwickelt wurde es von Ericsson, um Kabelverbindungen zwischen verschiedenen Gerätetypen zu ersetzen und mehrere Geräte miteinander zu vernetzen. Die Spezifikation der Bluetooth-Technik und der Protokolle wurde von der Bluetooth SIG (Special Interest Group) entwickelt. Diese kümmert sich auch um die Weiterentwicklung von Bluetooth und ist ein Zusammenschluss mehrerer großer Unternehmen, darunter Ericsson, Intel, Lenovo, Microsoft, Motorola, Nokia und Toshiba.

Bluetooth basiert auf dem Erstellen von sogenannten Ad-hoc-Netzen (auch Piconetze genannt), also vom generieren eines Netzwerks durch die Geräte selbst, ohne dass der Anwender Konfigurationen vornehmen muss. Abbildung [→ 6, Seite 29] zeigt ein Piconetz, bei dem mehrere Geräte miteinander verbunden sind. Dabei übernimmt ein Gerät die Rolle des Masters, alle anderen sind Slaves. Der Master teilt den Slaves - und sich selbst - Sende-Slots zu, damit keine gleichzeitigen Übertragungen stattfinden.

Die Funktechnik basiert auf einem Frequenzsprungverfahren mit 1600 Sprüngen pro Sekunde. Gesendet wird im Bereich um 2,4 GHz. Dieser Bereich wird auch von WLAN genutzt und ist in den meisten Ländern lizenzfrei. Er wird deshalb auch als ISM-Band (Industrial, Scientific and Medical-Band) bezeichnet.

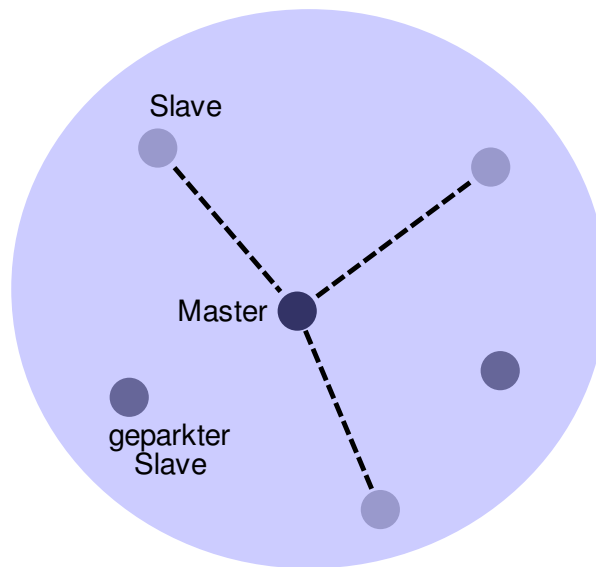


Abbildung 6.: Bluetooth-Piconetz (Quelle: Wikipedia)

Für Bluetooth gibt es mittlerweile mehrere Spezifikationen, die, neben neuen Sicherheitsstandards, auch immer die Übertragungsgeschwindigkeit erhöht haben. Aktuell ist die Version 2.1 mit Übertragungsraten von 3Mbit/s, wobei die Versionen abwärtskompatibel sind. Zur Zeit ist Version 3 in der Planung, das die Übertragungsgeschwindigkeit auf 54Mbit/s anheben soll und eine zusätzliche Kollisionsvermeidung beinhaltet (CSMA/-CA¹⁸). [SIG 2009]

Bluetooth unterstützt unterschiedliche Protokolle, auch Profile genannt. Sie regeln die Übertragungsart je nach Einsatzbereich. Dadurch ist es möglich z.B. Eingabegeräte wie Tastaturen anzusprechen (mittels dem Human Interface Device (HID) Profile) oder Audiosignale an ein Headset zu übertragen (mit dem Advanced Audio Distribution Profile (A2DP) Profile). Diese Protokolle besitzen eine eindeutige Kennung, die UUID (Universally Unique Identifier). Wenn zwei Geräte in einem Piconetz in Kontakt treten wollen tauschen sie vorher Informationen über die unterstützten Protokolle aus, um sich auf ein gemeinsames zu einigen. Dadurch wird z.B. vermieden, dass eine Tastatur mit einem Audio-Protokoll angesprochen wird.

¹⁸CSMA/CA: Abk. f. Carrier Sense Multiple Access / Collision Avoidance

2. Grundlagen

Neben den Mobiltelefonen können alle Endgeräte - wie PDAs oder Laptops - Inhalte per Bluetooth empfangen, die über das Bluetooth Object Push Profile (OPP bzw. OBEX) verfügen.

Zur Zeit besitzen etwa 96% aller Mobiltelefone in Deutschland Bluetooth, wobei der Bekanntheitsgrad in der Bevölkerung bereits 2006 bei 88% lag [Haase und Martin 2009] [Devriendt und Novelli 2006].

Besonders für Mobiltelefone interessant ist die Möglichkeit ein Gerät in einem *nicht-sichtbar*-Modus (non-discoverable mode) zu betreiben. Durch diese Einstellung sollen unerwünschte Ad-hoc-Netze verhindert werden, da das Gerät nicht auf Anfragen von Master-Geräten reagiert. Dies bedeutet jedoch nicht, dass das Gerät gänzlich unentdeckt bleibt: Zoller zeigt wie auch solche Geräte gefunden werden können, indem Sniffer¹⁹ den Datenverkehr überwachen [Zoller 2007].

Obwohl Bluetooth-Sender zum Teil Reichweiten bis 100m unterstützen, wird der Radius durch die in den Mobiltelefonen verbauten schwachen Sender auf etwa 10m beschränkt. Der Vorteil darin besteht aber, dass durch die begrenzte Reichweite von Bluetooth eine Art Location Based Service emuliert werden kann, also dass Inhalte bzw. Umfragen abhängig vom Standort des Teilnehmers sind. Insbesondere auf Messen ist dieser Punkt ausschlaggebend, so erlaubt er die Verteilung von Informationen zu Produkten direkt in der Nähe dieser Produkte.

¹⁹Sniffer: Software, die den Datenverkehr eines Netzwerks überwachen kann. Wikipedia [2009h]

3. Anforderungen

Dieses Kapitel beschreibt die Anforderungen an das zu erstellende Framework. Dazu unterscheidet man zwischen funktionalen und nicht-funktionalen Anforderungen [Somerville 2001, Kapitel 3 und 5]. Funktionale Anforderungen beschreiben die konkreten Anforderungen an das System, also welche Probleme es lösen können soll und welche Funktionen es bietet. Die nicht-funktionalen Anforderungen sind allgemeine Ansprüche an das System, die dieses nicht konkret betreffen, z.B. Anforderungen an die Zuverlässigkeit oder zum ethischen Hintergrund usw.. Die funktionalen Anforderungen wurden durch Gespräche mit Kunden und dem Auftraggeber gesammelt, außerdem durch Analyse der Fallstudie „Hobit“.

Zunächst wurde eine Machbarkeitsstudie durchgeführt: Dabei wurde von zwei Bachelor-Studentinnen der JavaME-Prototyp eines Informatik-Quiz für die Hobit entwickelt. Der Prototyp funktionierte wie erwartet [→ 8.2.1, Seite 80]. Daneben wurde eine Server-Anwendung entwickelt, um für verschiedene, vorangegangene Praktika am Fachbereich Informatik SQL-Befehle entgegennehmen und auf einer Datenbank ausführen zu können. Da alle Komponenten die Erwartungen erfüllen konnten, wird davon ausgegangen, dass eine Durchführbarkeit der Ausgabe gewährleistet werden kann.

Das Framework soll später die papiergebundenen Fragebögen möglichst komplett ersetzen und überflüssig machen. Die mobilen Umfragen müssen also mindestens die Funktionen dieser Umfragetechnik unterstützen, also einfach an potentielle Kunden verteilbar sein und durch diese einfach beantwortet werden können. Außerdem muss es relativ einfach sein neue Umfragen zu erstellen. Darüber hinaus sind folgende Punkte als Anforderungen zu beachten:

- **Ein generisches Framework** Die Anwendung soll beliebige Umfragen darstellen und ausführen können, ohne selbst angepasst werden zu müssen.

- **Einfache Verteilung** Umfragen sollen möglichst einfach an potentielle Teilnehmer verteilt werden. Die spezielle Problematik hierbei ist das Fehlen von Personal: Ziel ist das völlig selbstständige Verteilen von Umfragen, ohne Menschen die die Umfragen bewerben. Dies kann mittels Bluetooth geschehen.
- **Drag&Drop** Einfache Erstellung von neuen Umfragen mittels Drag & Drop auf einem normalen Arbeitsplatzrechner.
- **Superdistribution** Die Möglichkeit der Superdistribution, d.h. Teilnehmer der Umfrage können gleichzeitig wieder Verteiler der Umfrage werden, indem sie die Umfrage-Software an andere weitergeben. Dieser Punkt ist ein immenser Vorteil gegenüber „normalen“ Umfragen, da hier die Teilnehmer nicht die Möglichkeit haben, Fragebögen für andere zu vervielfältigen.
- **Viele Geräte** Die mobilen Umfragen sollen auf möglichst vielen Mobiltelefonen genutzt werden können.
- **Verzweigung von Umfragen** Das bedeutet, das Umfrage-Framework soll in der Lage sein, bei entsprechenden Antworten auch nur die zugehörigen weiterführenden Fragen darzustellen, also eine Art Baumstruktur von Umfragen ermöglichen.
- **Direktes Antworten** Nachdem an einer Umfrage teilgenommen wurde, soll direkt eine Rückmeldung an die Teilnehmer erfolgen können. Das kann eine einfache Bestätigung der Teilnahme sein, oder komplexe aggregierte Daten aus den gesammelten Umfrage-Antworten.

Neben diesen projektspezifischen, funktionalen Anforderungen gelten die nicht-funktionalen Anforderungen für jede Software [Sommerville 2001]: Wartbarkeit, Zuverlässigkeit, Effizienz und Benutzerfreundlichkeit. Darüber hinaus gelten gewisse ethische Anforderungen. Diese werden in [→ 8.6, Seite 99] genauere betrachtet. Außerdem sind Sicherheitsaspekte zu beachten: Die Sicherheit der Kommunikation gegen Manipulation und die Sicherheit der Daten gegen Veränderungen.

Zusammengefasst können daraus nun folgende Aufgaben abgeleitet werden:

1. Erstellung einer Software, um Umfragen generieren zu können.
2. Erstellung einer Software, um Umfragen auf dem Handy ausführen zu können.
3. Verteilung der Umfragen.

3. Anforderungen

4. Speicherung der Umfrage-Antworten in einer Datenbank.
5. Rückmeldung an den Umfrage-Teilnehmer.

4. Stand der Forschung

Dieses Kapitel beleuchtet andere Forschungsarbeiten die sich mit gleichen oder ähnlichen Themen beschäftigen. Da diese Arbeit aus zwei Teilen besteht - der Entwicklung eines Frameworks für mobile Umfragen und der Untersuchung von Distributionsvarianten und der Nutzerakzeptanz - werden auch zwei verschiedene Forschungsbereiche betrachtet.

Das Framework wird aus verschiedenen Komponenten bestehen, um Umfragen mit dem Mobiltelefon zu ermöglichen. Die zugrunde liegende Technik ist nicht neu, sie wird stattdessen nur in neuer Art verwendet. Aus diesem Grund werden nur ähnliche Ansätze beschrieben, um Umfragen mit Mobiltelefonen durchführen zu können, jedoch keine neuen Techniken beschrieben. Allgemein lassen sich die vorhandenen Ansätze in zwei Typen einteilen: Umfragen durch SMS und Umfragen im Browser mittels WAP¹. Stellvertretend für beide Möglichkeiten können folgende Produkte gesehen werden:

Xquest

Beispiel für eine Software, um Umfragen mit Mobiltelefonen per SMS durchführen zu können. Hersteller ist die Pixip.net GmbH². Diese Lösung besteht aus einer Webanwendung, um Umfragen zu erstellen und empfangene Antworten darstellen zu können. Die Anwendung generiert dabei aus jeder Frage eine einzelne SMS mit zugehörigen Antwortmöglichkeiten. Der Kunde kann dann auf diese SMS mit einer Antwort-SMS reagieren, in die er per Hand eine der möglichen Antworten eingibt. Für jede einzelne Frage wird somit eine SMS als Antwort benötigt. Ein Empfänger nimmt dann die empfangene SMS entgegen und trägt die Antwort in einer Datenbank ein.

¹WAP: Abk. f. Wireless Application Protocol: Techniken, um Internetseiten für Mobiltelefone optimiert zu übertragen und darzustellen

²Unternehmen Pixip.net GmbH, Garching: <http://www.pixip.net/>

mobiode

Beispiel einer Software, um Umfragen mittels WAP auf Mobiltelefonen durchführen zu können. Entwickler ist Wirenode s.r.o.³. Die Software erlaubt ebenfalls das Erstellen von Umfragen in einer Webanwendung, sowie die Analyse erhaltener Antworten. Dazu muss der Nutzer eine bestimmte URL mit dem Browser seines Mobiltelefons aufrufen und an der Umfrage teilnehmen. Während der gesamten Dauer der Umfrage ist der Nutzer mit dem Internet verbunden.

Zur Betrachtung der Distributionsvarianten und der Nutzerakzeptanz im Mobile Marketing existieren mehrere Studien, hier werden stellvertretend drei aufgelistet:

Konsumentenverhalten gegenüber Mobile Marketing

Akzeptanz und Reaktanz, Kizilok [Kizilok 2007].

Die Schrift fasst alle Konzepte des Mobile Marketing in einem Morphologischen Kasten zusammen und analysiert diese anschließend (z.B. Formen der Werbe-SMS, Banner und Location Based Services). Dann werden konsumentenseitige Erfolgsfaktoren analysiert und es wird auf die Reaktanztheorie eingegangen, also negative Effekte des Mobile Marketing beschrieben. Schließlich werden die aus den vorhergehenden theoretischen Untersuchungen gewonnenen Theorien in einer empirischen Studie erforscht.

Wann werden Mobile Marketing-Kampagnen akzeptiert?

Bauer, Neumann und Reichardt [Bauer u. a. 2008].

Die Schrift beschreibt die Akzeptanz des Mobile Marketing anhand einer empirischen Studie. Dazu wird vorher ein Strukturgleichungsmodell aufgestellt, basierend auf Hypothesen, die die Akzeptanz von Mobile Marketing verbessern sollen. Die Studie kommt zu dem Ergebnis, dass die persönliche Einstellung zum Mobile Marketing entscheidend für die Verhaltensabsicht ist. Andere Faktoren, wie soziale Normen etc., sind dabei nebensächlich.

Mobile-Marketing-Instrumente im Kundenmanagement

Erfolgsfaktoren und Erfolgspotential - Eine State of the Art Analyse, Miller [Miller 2009].

Die Studie analysiert die Potentiale innerhalb der Kundenmanagement-Prozesse und die Rolle des Mobiltelefons im Kundenmanagement. Dies geschieht anhand

³Unternehmen Wirenode s.r.o., Prag: <http://www.mobiode.com/>

verschiedener Fallbeispiele der Mobile Marketing Instrumente. Anschließend werden die Erfolgsfaktoren für das Kundenmanagement abgeleitet. Als Schluß wird das Fazit gezogen, dass die Nutzerakzeptanz entscheidend für alle Mobile Marketing Instrumente ist und die Kundenmanagement-Prozesse direkt davon abhängen. Darüber hinaus gibt die Studie einen umfassenden Überblick über wissenschaftliche, empirische Untersuchungen zur Akzeptanz bzw. Nutzungsabsicht Mobiler Marketing-Instrumente.

5. Entwurf eines generischen Frameworks für mobile Umfragen

Dieses Kapitel beschreibt den Entwurf des generischen Frameworks und die zugrunde liegenden Entscheidungen. Zunächst wird der Grundgedanke eines generischen Ansatzes beschrieben. Dann werden die Use-Cases anhand der Anforderungen erstellt. Anschließend werden darauf aufbauend im Architekturentwurf die Subsysteme des Frameworks entworfen. Falls zur Erstellung der Subsysteme mehrere Möglichkeiten existieren, so werden diese verglichen und die am besten passende ausgewählt.

Danach wird in der Spezifikation der Feinentwurf der Subsysteme beschrieben und die Schnittstellen zwischen diesen entworfen. Darauf aufbauend werden dann die einzelnen Komponenten der Subsysteme bzw. deren Klassenmodelle abgeleitet und ein Datenmodell erstellt.

Der hier gewählte Ansatz bietet gegenüber den existierenden Lösungen für mobile Umfragen zwei Vorteile:

1. Es ist nicht nötig, während der Teilnahme an einer Umfrage ständig online sein zu müssen.
2. Es wird nicht für jede einzelne Frage eines Fragen-Katalogs eine eigene SMS benötigt, vielmehr werden Antworten zusammengefasst oder gleich über andere Funktechniken übertragen.

5.1. Generik des Frameworks

Das zu erstellende Framework soll generisch sein. Unter generischer Programmierung versteht man: „... ein Verfahren zur Entwicklung wiederverwendbarer Software-Bibliotheken“ [Wikipedia 2009b]. Das bedeutet im vorliegenden Fall, dass das Framework möglichst universell eingesetzt werden kann zur Erstellung unterschiedlichster Umfragen und nicht speziell auf eine Umfrage bzw. einen Umfragetyp festgelegt ist.

Erreicht werden kann diese Generik durch die Verwendung einer Beschreibungssprache. Das bedeutet, Umfragen werden durch eine Beschreibungssprache beschrieben und können beliebig aufgebaut sein: Das erstellende Subsystem erzeugt eine Beschreibung, das darstellende Subsystem auf einem Mobiltelefon wertet diese Beschreibungen dann aus und stellt die Umfrage entsprechend dar. Als Beschreibungssprache kann ein proprietäres, selbst entwickeltes Format verwendet werden, oder ein existierendes Format wie XML.

Die Verwendung einer Beschreibungssprache bietet mehrere Vorteile:

1. Ein reiner Beschreibungstext kann sehr kompakt sein. Ein Textdokument braucht meist wesentlich weniger Speicherplatz, als ein fertig kompiliertes Programm.
2. Eine Beschreibungssprache ermöglicht die Wiederverwendung und Anpassung existierender Umfragen, da ein Textdokument einfach geparsed werden kann.
3. Textdokumente sind menschenlesbar und können demnach auch ohne Werkzeuge angepasst werden.

5.2. Use-Cases

Hier werden aufgrund der gestellten Anforderungen die Use-Cases entworfen.

Abbildung [→ 7, Seite 39] zeigt das Client-Subsystem des Umfrage-Frameworks. Der Akteur ist der Befragte, d.h. der Teilnehmer der Umfrage. Dieser ist gleichbedeutend mit dem Eigentümer des Mobiltelefons. Er muss in der Lage sein, sich die einzelnen

Fragen einer Umfrage anzeigen zu lassen und er muss diese beantworten können. Darüber hinaus soll er die gegebenen Antworten jederzeit und von überall versenden können. Außerdem soll er sich die Antworten vom Server (falls vorhanden) ansehen können.

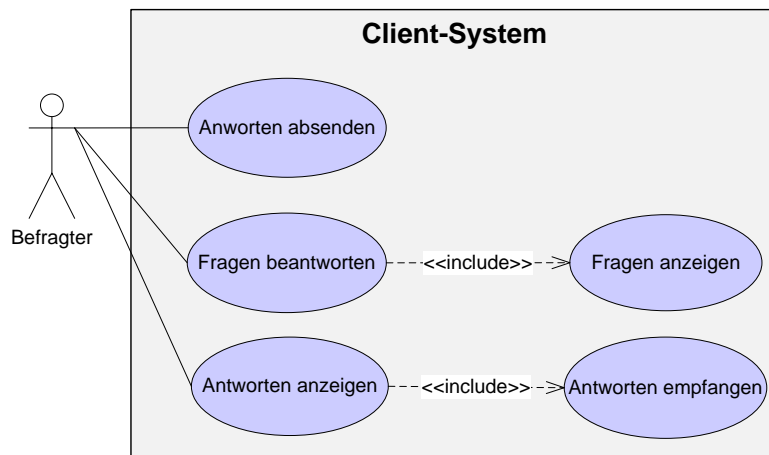


Abbildung 7.: Use-Case des Clients

Abbildung [→ 8, Seite 40] zeigt das Server-Subsystem des Frameworks. Hier existieren zwei Akteure: Ein Befragter, der neue Umfragen mit Hilfe des Frameworks erstellen kann und ein Administrator, der das Framework mit einer Datenbank verbindet. In dieser Datenbank werden dann die Antworten zu einer Umfrage in einer einzelnen Tabelle gespeichert, dabei gehört zu jeder Umfrage eine eigene Tabelle. Der Befragter selbst muss in der Lage sein, neue Umfragen zu erstellen und alte zu bearbeiten bzw. zu löschen. Aus den erstellten Umfragen muss außerdem die Client-Software generiert werden, ebenso ein SQL-Befehl, um die Datenbank entsprechend den Anforderungen einer Umfrage anpassen zu können. Die Datenbank selbst wird über eine Schnittstelle vom Server-Subsystem angesprochen, welches Antworten in diese einträgt - zu sehen am *Antworten erhalten*. Sie gehört nicht zum eigentlichen Framework, deshalb kann jedwede Datenbank verwendet werden, für die Treiber existieren und die SQL-Befehle verarbeiten kann.

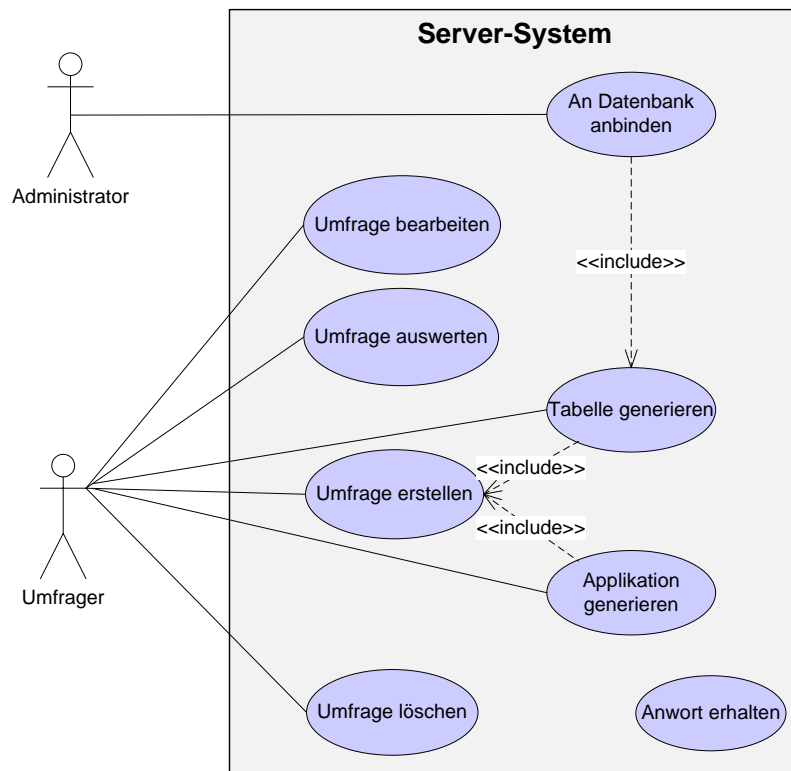


Abbildung 8.: Use-Case des Servers

5.3. Architekturentwurf des Framework

Laut den Use-Cases muss das Framework verschiedene Aufgaben erfüllen können: Erstens müssen die Umfragen auf einem Mobiltelefon dargestellt und beantwortet werden können. Zweitens wird eine Software für einen Arbeitsplatz-Rechner benötigt, um neue Umfragen erstellen zu können. Drittens ist es nötig, eintreffende Umfrage-Antworten entgegen zu nehmen und in eine Datenbank einzutragen und eventuell Antworten an die Teilnehmer zurück zu liefern.

5.3.1. Entwurf des Clientsystems

Das Clientsystem ist dafür zuständig, Nutzern eine Umfrage darzustellen und sie Eingaben tätigen zu lassen. Außerdem soll das Subsystem getätigte Eingaben speichern und an den entsprechenden Server weiterleiten. Darüber hinaus soll es die Möglichkeit bieten, Antworten seitens des Servers darstellen zu können, damit z.B. aggregierte Ergebnisse einer Umfrage angezeigt werden können.

Für die Darstellung von Umfragen auf Mobiltelefonen sind drei verschiedene Ansätze denkbar:

Die Umfrage wird in einem bereits installierten Programm auf dem Mobiltelefon dargestellt, z.B. einem Browser.

Vorteil:

- Es wird keine zusätzliche Software benötigt.
- Es existieren bereits ausgereifte Werkzeuge, um Umfragen für Webbrowser erstellen und auswerten zu können.
- Eine Teilnahme ist auch ohne Mobiltelefon möglich.
- Die Darstellung ist, bei Verwendung von Standards, auf allen Geräten gleich.

Nachteil:

- Nicht alle Geräte besitzen einen Browser.
- Es ist schwierig die Leute auf eine Umfrage aufmerksam zu machen.

- Kein Location Based Service möglich: Möglicherweise fehlender Bezug der Umfrage zur momentanen Umgebung des Teilnehmers bzw. des Mobiltelefons.
- Eine Internetverbindung ist zwingend nötig.
- Schwierige Navigation innerhalb der Umfrage-Seite bei Geräten ohne Touchscreen.

Die Umfrage wird als einzelne SMS je Frage verschickt:

Vorteil:

- Es wird keine zusätzliche Software benötigt.
- Es existieren Werkzeuge hierfür.
- Die Antworten erfolgen unmittelbar.
- Nutzer können direkt „angesprochen“ werden.

Nachteil:

- Kein Location Based Service möglich: Möglicherweise fehlender Bezug der Umfrage zur momentanen Umgebung des Teilnehmers bzw. des Mobiltelefons.
- Antworten können nur als SMS gegeben werden und müssen als richtiger Text eingegeben werden (trotz vorhandener Antworten zur Auswahl).
- Jede Antwort benötigt eine eigene SMS.
- Der Teilnehmer kann Antworten nicht mehr ändern.
- Eine Navigation zwischen den Fragen ist schwierig.

Die Umfrage nutzt ein eigenständiges Programm zur Darstellung:

Vorteil:

- Ein eigenständiges Programm kann an Kunden über Mobile Marketing verteilt werden und diese so auf die Umfrage aufmerksam machen.
- Location Based Service bei Bluetooth möglich [→ 2.1, Seite 12][→ 2.4, Seite 28].

- Eine dauerhafte Internetverbindung ist nicht nötig, um an einer Umfrage teilzunehmen.
- Nicht jede Frage benötigt eine Antwort-SMS.
- Orts- und Zeitunabhängigkeit bei der Beantwortung der Fragen.

Nachteil:

- Es existiert bisher keine Software.
- Unterschiedliche Betriebssysteme auf Mobiltelefonen schränken die Programmiersprache ein.
- An der Umfrage kann nur mittels eines Mobiltelefons teilgenommen werden.
- Übertragung der Daten erfordert irgendeine Datenverbindung.

Zusätzlich zu den genannten Vor- und Nachteilen geht aus den Anforderungen in Kapitel [→ 3, Seite 31] hervor, dass möglichst viele Endgeräte bedient werden sollen, also ein möglichst großer Nutzerkreis vorhanden ist. Außerdem muss eine Superdistribution (also das Weitergeben der Umfrage an andere Nutzer durch den Teilnehmer [→ 8.3, Seite 87]) möglich sein. Aus diesen Gründen wird ein eigenständiges Programm zur Darstellung der Umfragen auf den Mobiltelefonen ausgewählt.

Zur Weiterleitung der Antworten ist eine Datenverbindung nötig, über die die Antworten an den Server übermittelt werden können. Hierfür soll, falls vorhanden, eine bestehende Internetverbindung genutzt werden. Als Alternative hierzu sollen die Antworten als SMS versandt werden.

Wie in [→ 2.2, Seite 19] beschrieben, existieren für Mobiltelefone verschiedene Programmiersprachen, abhängig vom verwendeten Betriebssystem auf dem Gerät. Da JavaME von den meisten Geräten und Herstellern unterstützt wird, wird diese Programmiersprache zur Implementierung des Client-Subsystems gewählt [→ 2.3, Seite 23].

5.3.2. Entwurf des Serversystems

Das Serversystem muss mehrere Funktionen erfüllen: Erstens dient es zur Erstellung neuer Umfragen. Zweitens muss es eingehende, beantwortete Umfragen in einer Datenbank speichern. Und drittens soll es Rückantworten an die Teilnehmer zurückliefern können.

Wegen der unterschiedlichen Anforderungen an das Serversystem ist es zweckmäßig, es in zwei getrennte Subsysteme aufzuspalten: Ein Subsystem ist für die Erstellung neuer Umfragen zuständig und generiert Umfrage-Beschreibungen für das Clientsystem. Darüber hinaus erstellt es die entsprechenden Tabellen in der Datenbank. Das zweite Subsystem ist für den Datenaustausch zuständig und nimmt ankommende Umfrage-Antworten entgegen und trägt diese in die Datenbank ein, bzw. beantwortet eintreffende Anfragen.

Zur Erstellung von Umfragen existieren verschiedene Anwendungen, diese können jedoch nicht die erforderliche Umfrage-Beschreibung generieren, um den generischen Ansatz zu ermöglichen. Außerdem existiert momentan keine Software für die Darstellung von Umfragen auf Mobiltelefonen. Es gibt daher auch keine Software, um Umfragen für diese zu erstellen und das Server-Subsystem muss neu entwickelt werden.

Da das Clientsystem als eigenständiges Umfrage-Programm entwickelt wird, das anhand einer Beschreibungssprache Umfragen darstellt, gibt es zwei Möglichkeiten neu erstellte Umfragen mit diesem zu nutzen:

- Neu erstellte Umfrage-Beschreibungen werden an das Clientsystem übermittelt und von diesem ausgewertet.
- Eine neu erstellte Umfrage-Beschreibung wird direkt mit dem Clientsystem kombiniert und gemeinsam verteilt.

Beide Möglichkeiten unterscheiden sich in der Implementierung nur marginal und können einfach durch die andere ersetzt werden. Aufgrund einer einfacheren Verteilung später, wird hier der zweite Ansatz gewählt und die Beschreibung mit dem Clientsystem gekoppelt. Zur genauen Begründen und weiteren Verteilungsmöglichkeiten siehe Distributionsvarianten [→ 8.3, Seite 87].

Als Beschreibungssprache kommen proprietäre Ansätze in Frage oder die Verwendung von vorhandenen Standards wie XML. XML bietet den Vorteil, dass es viele Implementierungen zum Parsen und Erstellen von XML-Dokumenten gibt, was deren Handhabung sehr erleichtert. Darüber hinaus ist XML weit verbreitet und einfach zu verstehen. Deswegen wird als Beschreibungssprache eine angepasste Variante von XML verwendet.

Die weiteren Anforderungen an das Serversystem sind der Empfang von Umfrageergebnissen und das Eintragen der empfangenen Daten in eine Datenbank. Außerdem ist die Erstellung von Antworten für die Teilnehmer wichtig. Für diese Aufgaben ist ein Zugang zum Internet zwingend nötig. Dazu kann entweder ein vollständig eigenständiges Programm entwickelt werden, das auf entsprechende Nachrichten wartet, oder die Anwendung kann als Modul in eine vorhandene Server-Farm integriert werden. Da in Unternehmensnetzwerken meistens rigide Sicherheitsanforderungen für die Netzwerkkommunikation gelten und da sowohl das Einrichten neuer Rechner, als auch das Erstellen neuer Netzwerkfreigaben erheblichen Aufwand bedeuten kann, sollte ein bestehendes System erweitert werden. Eine Webapplikation innerhalb eines Applikations-Servers ist das zweite Subsystem des Serversystem und nimmt eintreffende Anfragen entgegen und wertet sie aus.

5.3.3. Subsysteme des Frameworks

Aus den vorher angeführten Möglichkeiten wird das Framework aus folgenden vier Subsystemen bestehen (Begründungen siehe im vorherigen Abschnitt):

1. Die Client-Software. Dies ist die Umfrageanwendung für das Mobiltelefon. Sie stellt die Umfragen auf dem Mobiltelefon dar, indem sie die Beschreibung einer Umfrage auswertet. Außerdem speichert sie die Antworten und leitet sie an den Empfangs-Server weiter. Zusätzlich stellt sie Antworten vom Server, z.B. aggregierte Daten, dar.
2. Ein Designer-Frontend. Dieses Frontend dient zur Erstellung neuer Umfragen auf einem Computer. Es erstellt auf Knopfdruck aus einer Umfrage die zugehörige Umfrage-Beschreibung. Außerdem erstellt es die entsprechenden SQL-Anweisungen zur Anpassung der Datenbanken eines Datenbank-Servers an eine neue Umfrage.

3. Ein Empfangs-Server. Er nimmt Antworten zu Umfragen von den Clients entgegen und trägt diese in eine Datenbank ein. Außerdem kann er die Datenbestände eines Datenbank-Servers aggregieren und diese an die Clients zurück liefern.
4. Ein SMS-Hub. Diese Anwendung dient als Zwischenstation zur Weiterleitung von Umfrage-Antworten an den Empfangs-Server. Sie kommt zum Einsatz, wenn die Antworten vom Mobiltelefon mittels SMS gesendet werden sollen. Außerdem kann sie aggregierte Daten vom Empfangs-Server an die Client-Software mittels SMS weiterleiten.

5.4. Spezifikation des Framework

Hier wird der Feinentwurf des Frameworks spezifiziert und damit die konkrete Struktur der Subsysteme und ihrer Komponenten, sowie das Zusammenspiel zwischen diesen. Die Entscheidungen, die zu diesem Ansatz geführt haben, werden im vorigen Kapitel beschrieben.

Abbildung [→ 9, Seite 47] zeigt eine Gesamtübersicht der einzelnen Subsysteme des Frameworks. Das Framework selbst besteht zunächst aus einem Server-Teil und einem Client-Teil. Abbildung [→ 10, Seite 48] zeigt dabei, welche Teile als eigenständige Subsysteme erstellt werden.

Als Beschreibungssprache kommt XML zum Einsatz. Die Beschreibung einer Umfrage muss mehrere Informationen enthalten. Zum einen sind das Informationen über die Umfrage selbst: Also Ersteller der Umfrage und Inhalt derselben. Dazu kommen Informationen zum Zielort, d.h. wohin die Antworten gesendet werden sollen, also eine URL oder eine Rufnummer bei der Verwendung von SMS. Hinzu kommen Zugangsdaten zur Authentisierung bei der Empfangskomponente des Servers. Schließlich kommen noch die einzelnen Fragen der Umfrage selbst hinzu, ebenso wie eventuell vorgegebene Antwortmöglichkeiten. Die Fragen erhalten außerdem eine Beschreibung über den Fragetyp, also ob es sich um eine Multiple-Choice-Frage, offene Frage etc. handelt.

5. Entwurf eines generischen Frameworks für mobile Umfragen

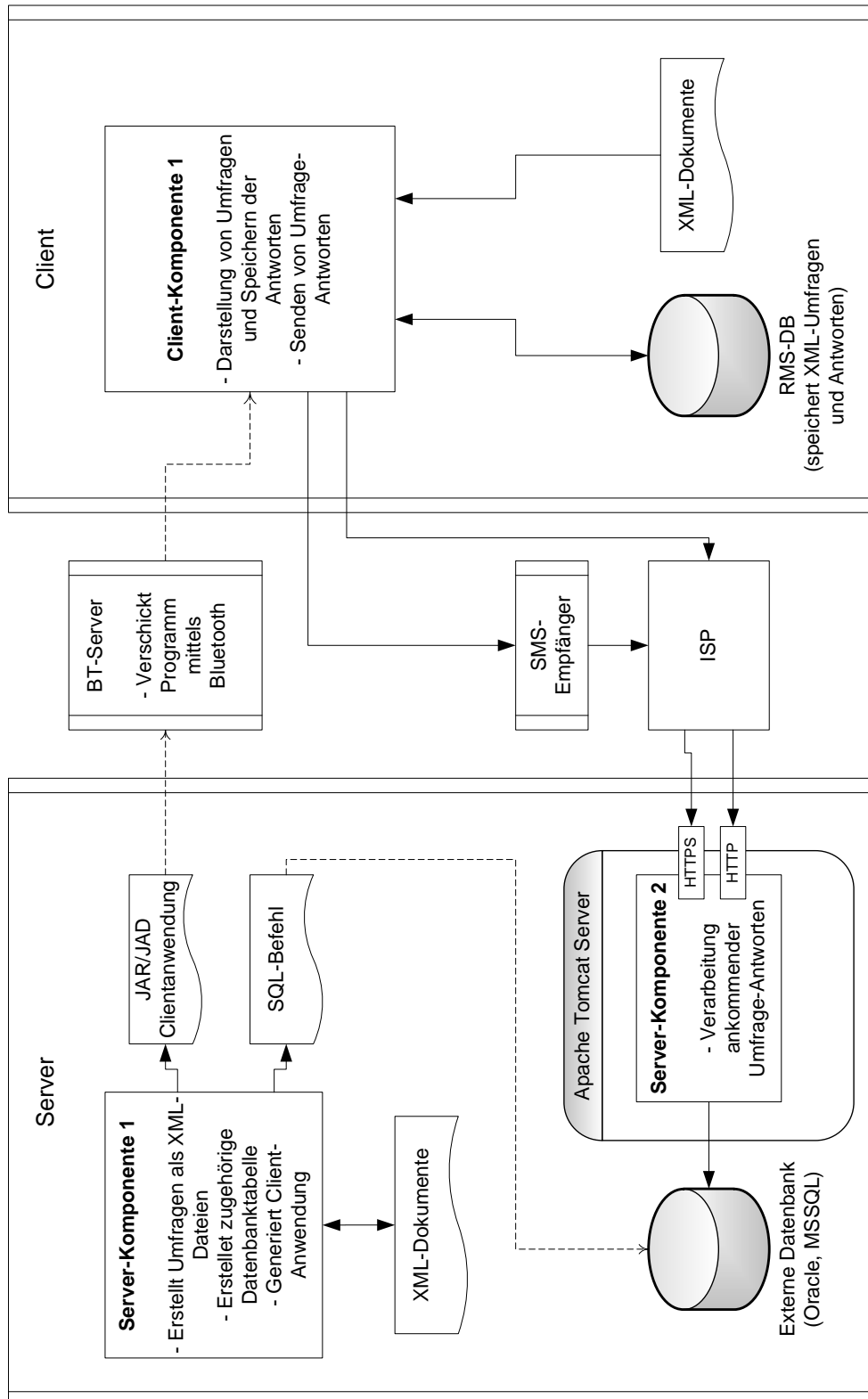


Abbildung 9.: Die einzelnen Systeme des Framework

5. Entwurf eines generischen Frameworks für mobile Umfragen

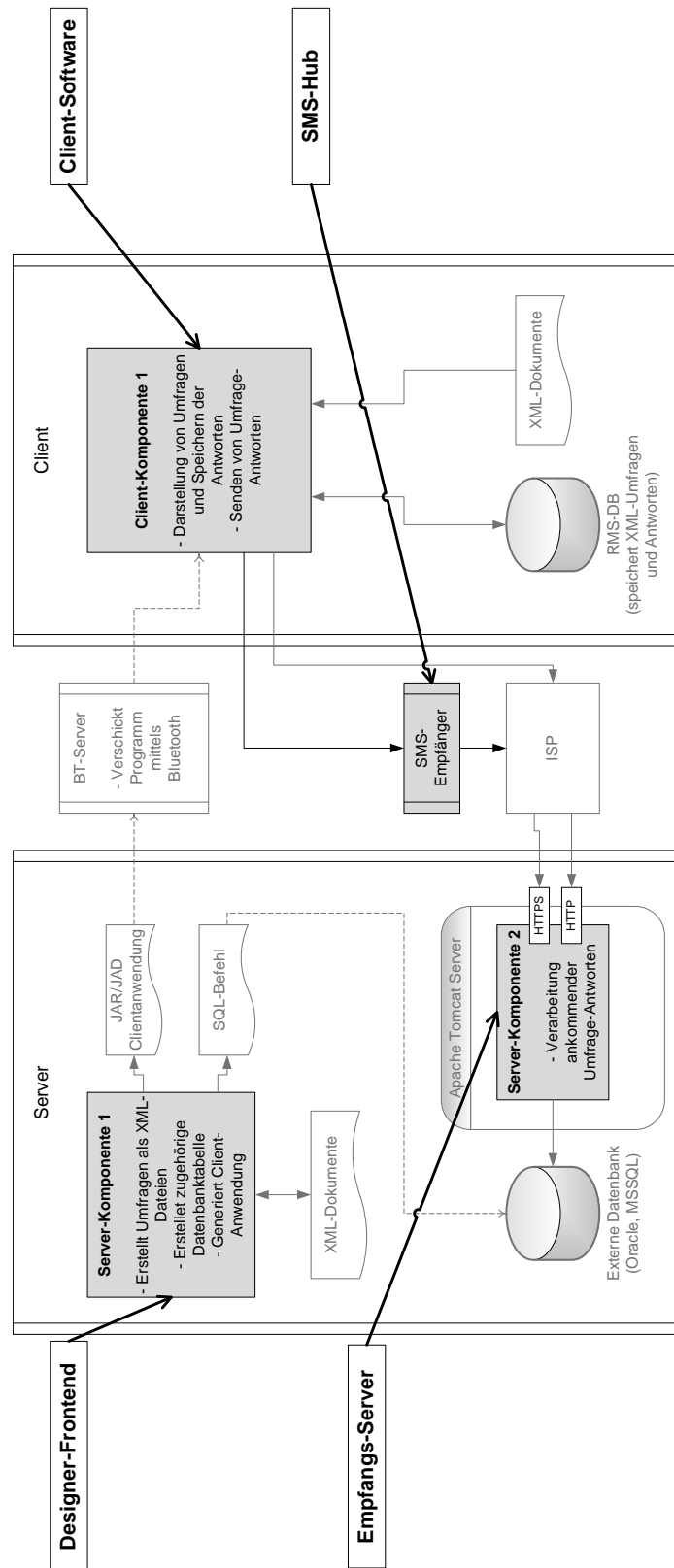


Abbildung 10.: Zu erstellende Subsysteme des Frameworks

5.4.1. Client-Software

Die Darstellung der Fragen und zugehörigen Antworten durch die Client-Software erfolgt generisch. Das bedeutet, das Subsystem generiert aus der Beschreibung einer Umfrage die passenden Oberflächen und stellt diese dar. Dazu muss die XML-Beschreibung eingelesen werden, was die Komponente *XMLReader* übernimmt. Sie generiert dann die Umfrage-Objekte *Questionnaire*, *QuestionsCollection*, *Question* und *Answer*. Diese werden in [→ 5.5, Seite 59] genauer beschrieben. Die Darstellung der Oberflächen erfolgt durch die Komponente *QuestionForm*.

Die Antworten zu den Fragen müssen lokal auf dem Gerät gespeichert werden, damit der Versandzeitpunkt dem Nutzer überlassen werden kann und der Versand nicht zwangsweise direkt nach Beantwortung der Umfrage erfolgen muss. Dieser variable Versandzeitpunkt ist zwingend nötig, da Mobiltelefone vom momentanen Netz abhängig sind und es nicht sicher gewährleistet ist, dass jederzeit eine Datenverbindung hergestellt werden kann. Außerdem wird durch die Speicherung Datenverlust vermieden, sollte die Anwendung plötzlich abstürzen. Darüber hinaus dient die Speicherung der Umfrage-Antworten als Kontrolle, ob die Umfrage bereits beantwortet wurde. Nur dadurch ist es der Client-Software überhaupt möglich die mehrfache Teilnahme an einer Umfrage zu verhindern, ohne dass Daten des Teilnehmers auf dem Server benötigt werden. Für die Handhabung der Daten ist die Komponente *QuestionnaireDB* zuständig.

Die gegebenen Antworten werden als fertiger SQL-Befehl an den Empfangs-Server verschickt, der diesen auf einer Datenbank ausführen kann. Dies bietet den Vorteil, dass ein Parser auf Seiten des Servers entfällt und keine zusätzlichen Funktionen zur Umwandlung im Empfangs-Server integriert werden müssen. Zur Erstellung der korrekten SQL-Anweisung dient die Komponente *SQLPreparer*.

Als Schnittstellen für die Client-Software werden somit jene benötigt, die dem Austausch von Antworten mit dem Empfangs-Server dienen. Die Komponente *ConnectToServer* ist dafür zuständig. Die Verbindung mit dem Empfangs-Server wird als HTTP-Verbindung implementiert, da damit eine einfache Erweiterung auf HTTPS¹ möglich ist. Das erlaubt eine gesicherte Verbindung zum Empfangs-Server. Darüber hinaus ist HTTP ein

¹HTTPS: Abk. f. Hypertext Transfer Protocol Secure: Übertragungsprotokoll, um Verschlüsselung und Authentifizierung zwischen Kommunikationspartnern zu ermöglichen.

Internet-Standard und ermöglicht die Integration des Empfangs-Servers in einem Web-/Applikationsserver. Sollte keine direkte Datenverbindung möglich sein, z.B. durch einen fehlenden Datentarif, müssen die Antworten mittels SMS abgeschickt werden. SMS wird deswegen gewählt, da dies das einzige Medium bei Mobiltelefonen ist, das immer vorhanden ist.

Die Diagramme [→ 11, Seite 51] und [→ 12, Seite 52] zeigen das Klassenmodell der Client-Software. Das Diagramm [→ 13, Seite 53] zeigt die Klassen die zur Datenübertragung genutzt werden.

5.4.2. Designer-Frontend

Das Model, also die Datenstruktur ist gleich dem in der Client-Software und wird in [→ 5.5, Seite 59] beschrieben. Die Oberfläche (View) ermöglicht ein Erstellen von neuen Umfragen mittels Drag&Drop. Zur Darstellung der Umfrage-Struktur wird eine Baumansicht gewählt, um die möglichen Verzweigungen bei unterschiedlichen Antworten zu zeigen. Die Komponenten *View_EditNewDialog* und *View_PanelTreeview* stellen die entsprechende Oberfläche dar. Darüber hinaus ist es möglich, mehrere Fragen gleichzeitig zu bearbeiten. Hierzu wird eine Tabulator-Ansicht gewählt. Die Verwendung einer Plattform als Grundlage hat mehrere Vorteile:

- Mehrere gleichzeitig geöffnete Umfragen, durch automatische Nebenläufigkeit der Threads durch die Plattform.
- Nutzung von Standard-Dialogen wie Speichern, Öffnen etc..
- Schnellere Entwicklung der Anwendung durch die Nutzung von bestehenden Modulen zur Dateimanipulation und Darstellung.
- Bewährte Plattform als Unterbau verhindert unerwartete Programmabstürze.

Die *Action*- Module nehmen dabei die Aktionen der Oberfläche entgegen und leiten diese an die entsprechenden Stellen weiter.

Erstellte Umfrage können vom Designer-Frontend gespeichert und anschließend wieder geöffnet werden. Dazu wird die Umfrage-Beschreibung als XML-Dokument gespeichert. Per Knopfdruck wird dann aus einer Umfrage die Umfrage-Beschreibung generiert. Diese

5. Entwurf eines generischen Frameworks für mobile Umfragen

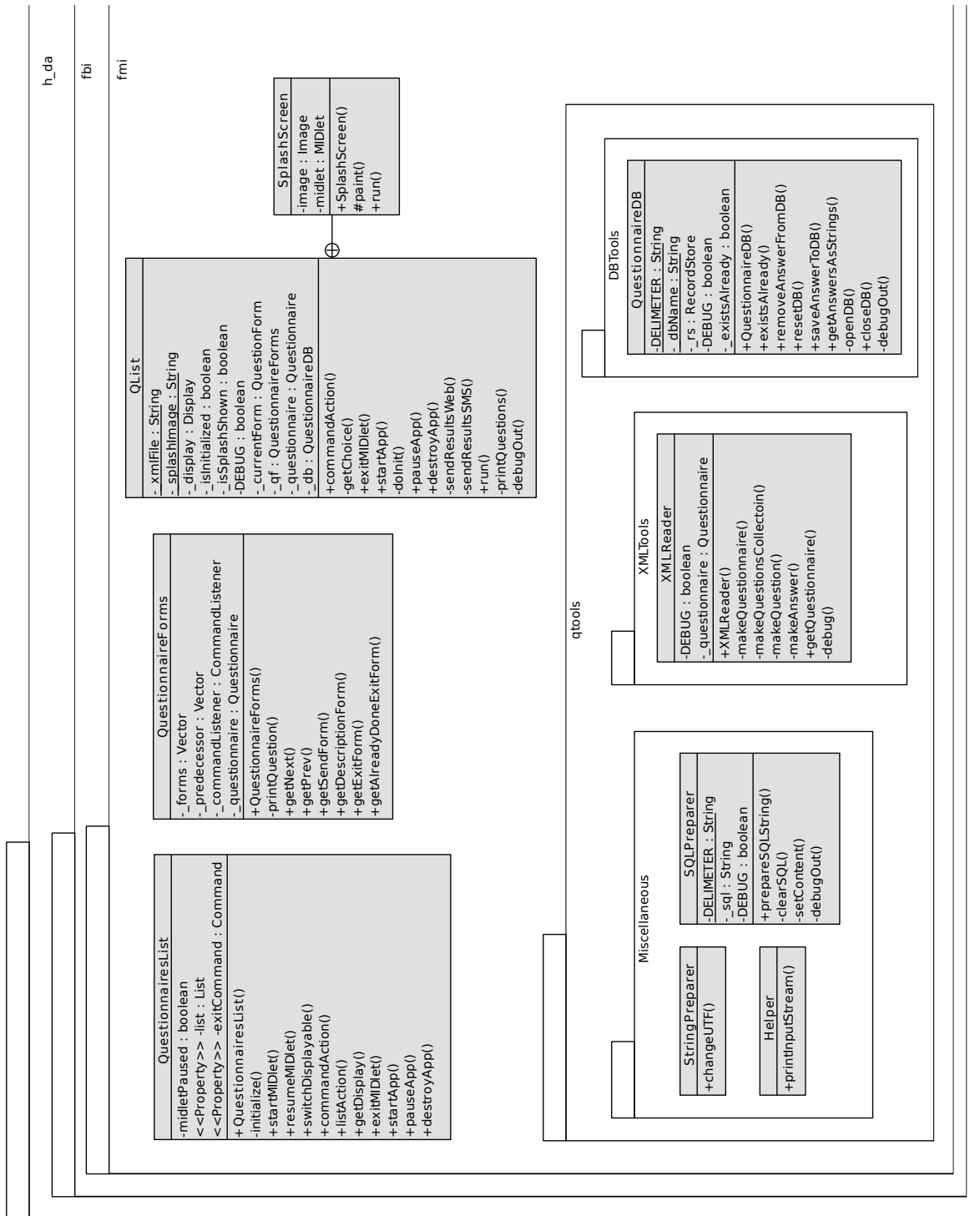


Abbildung 11.: Klassen der Client-Software Teil 1

5. Entwurf eines generischen Frameworks für mobile Umfragen

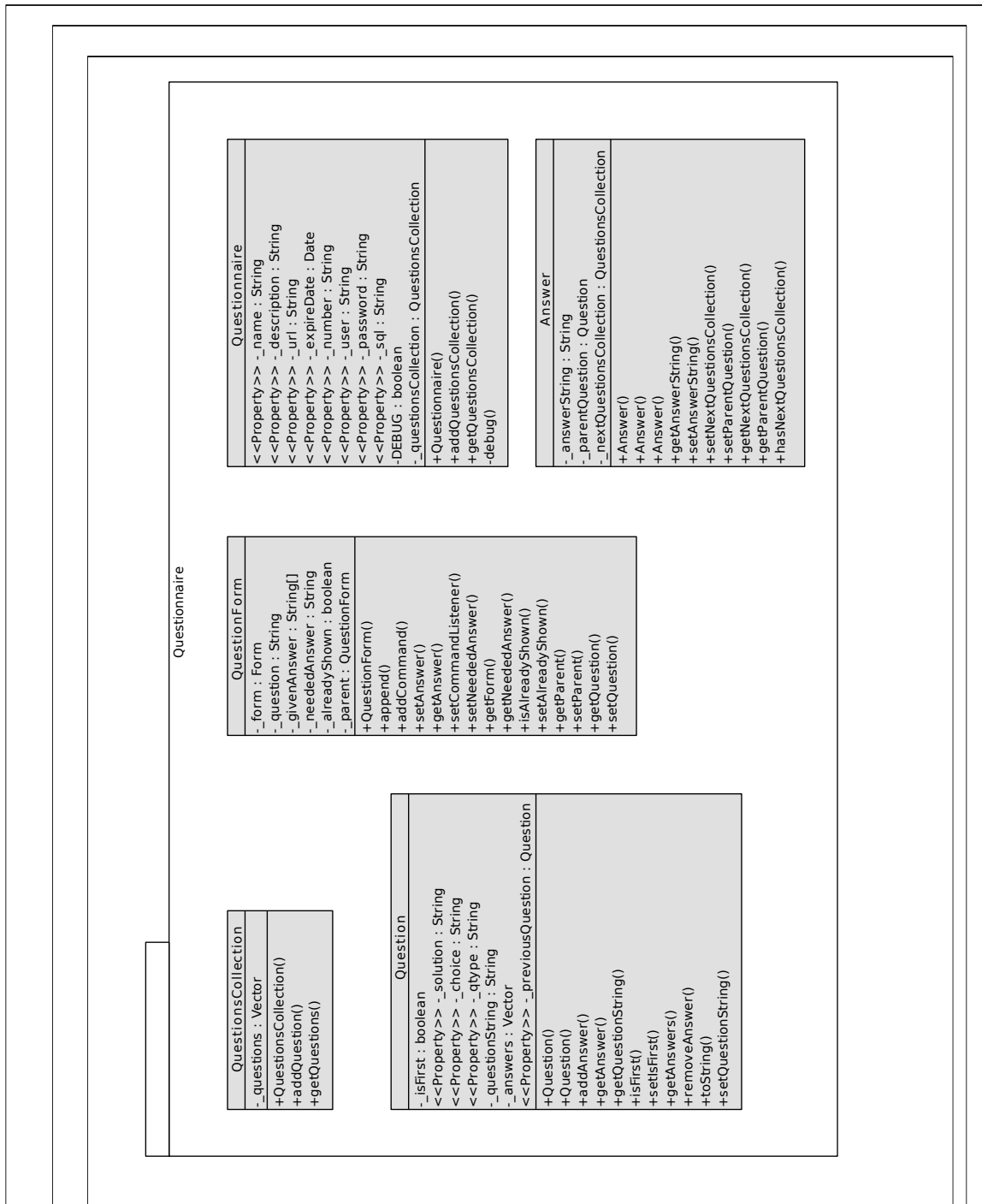


Abbildung 12.: Klassen der Client-Software Teil 2

5. Entwurf eines generischen Frameworks für mobile Umfragen

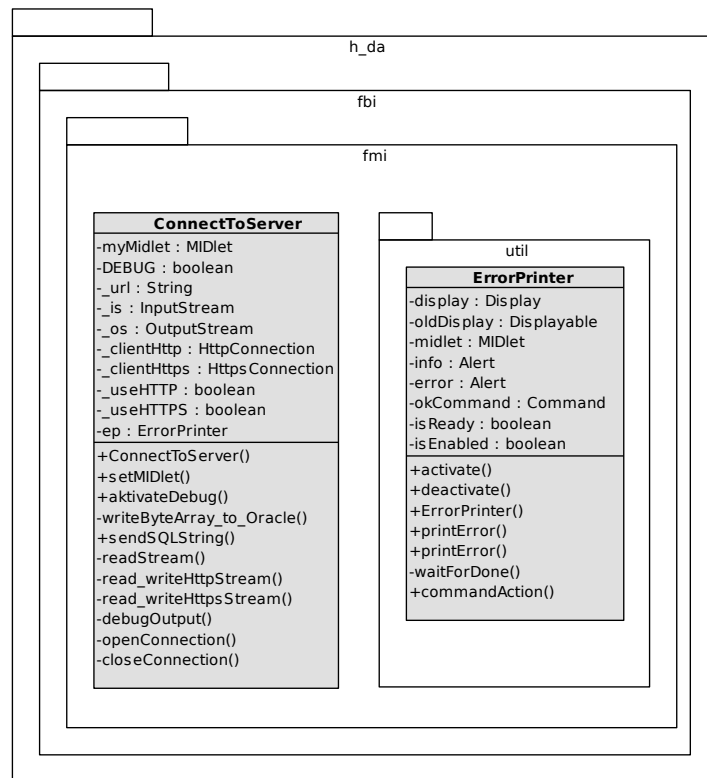


Abbildung 13.: Klassen zur Datenübertragung in der Client-Software

wird dann in das Archiv der Client-Software integriert, da so eine einfache Verteilung neuer Umfragen ermöglicht wird [→ 8.3, Seite 87]. Die Client-Software kann dann die Umfrage darstellen. Der Befehl generiert darüber hinaus einen zur Umfrage passenden SQL-Befehl, um eine Tabelle in einer Datenbank anzulegen. Die Tabelle dient dazu, die entsprechenden Antworten der Umfrage speichern zu können. Diese Funktionen übernimmt die *Controller_QFrameModul* Komponente.

Schnittstellen zu anderen Subsystemen besitzt das Designer-Frontend nicht, es werden lediglich JAR-Dateien mit der Client-Software [→ 2.3, Seite 23] und Textdateien mit den SQL-Befehlen erstellt. Außerdem werden XML-Dateien mit den Beschreibungen der Umfragen gelesen und geschrieben.

Die Diagramme [→ 14, Seite 55] und [→ 15, Seite 56] zeigen das Klassenmodell des Designer-Frontends.

5.4.3. Empfangs-Server

Der Empfangs-Server nimmt SQL-Anweisungen von den einzelnen Umfrage-Clients entgegen und führt diese auf einer Datenbank aus (Komponente *ExchangeRMS_Oracle*). Dafür verbindet sich der Server durch die JDBC-Schnittstelle. JDBC steht für Java Database Connectivity und ist eine Datenbankschnittstelle für die Java-Plattform, um relationale Datenbanken ansprechen zu können [Wikipedia 2009d]. Anschließend wird die Antwort der Datenbank auf die SQL-Anweisung durch die Komponente *ProcessSQL* zurück geliefert. Bis zum Erhalt der Antwort von der Datenbank wird die Datenverbindung offen gehalten, um Fehlermeldungen oder aggregierte Daten zurückliefern zu können.

Der Empfangs-Server läuft als Java-Servlet in einem Servlet-Container auf einem Webserver. Dafür wird der Apache Tomcat-Webserver verwendet. Darüber hinaus bietet der Empfangs-Server auch die Möglichkeit die aktuellen Inhalte der Tabellen durch eine JSP-Seite darstellen zu lassen. JSP steht für Java Server Pages und ist eine Technik, um Webseiten dynamisch erstellen zu können. Die Komponente *PrintDBRows* dient als unterstützende Klasse für die JSP-Seite, um die Inhalte darstellen zu lassen.

Diagramm [→ 16, Seite 57] zeigt das gesamte Klassenmodell des Empfangs-Servers.

5. Entwurf eines generischen Frameworks für mobile Umfragen

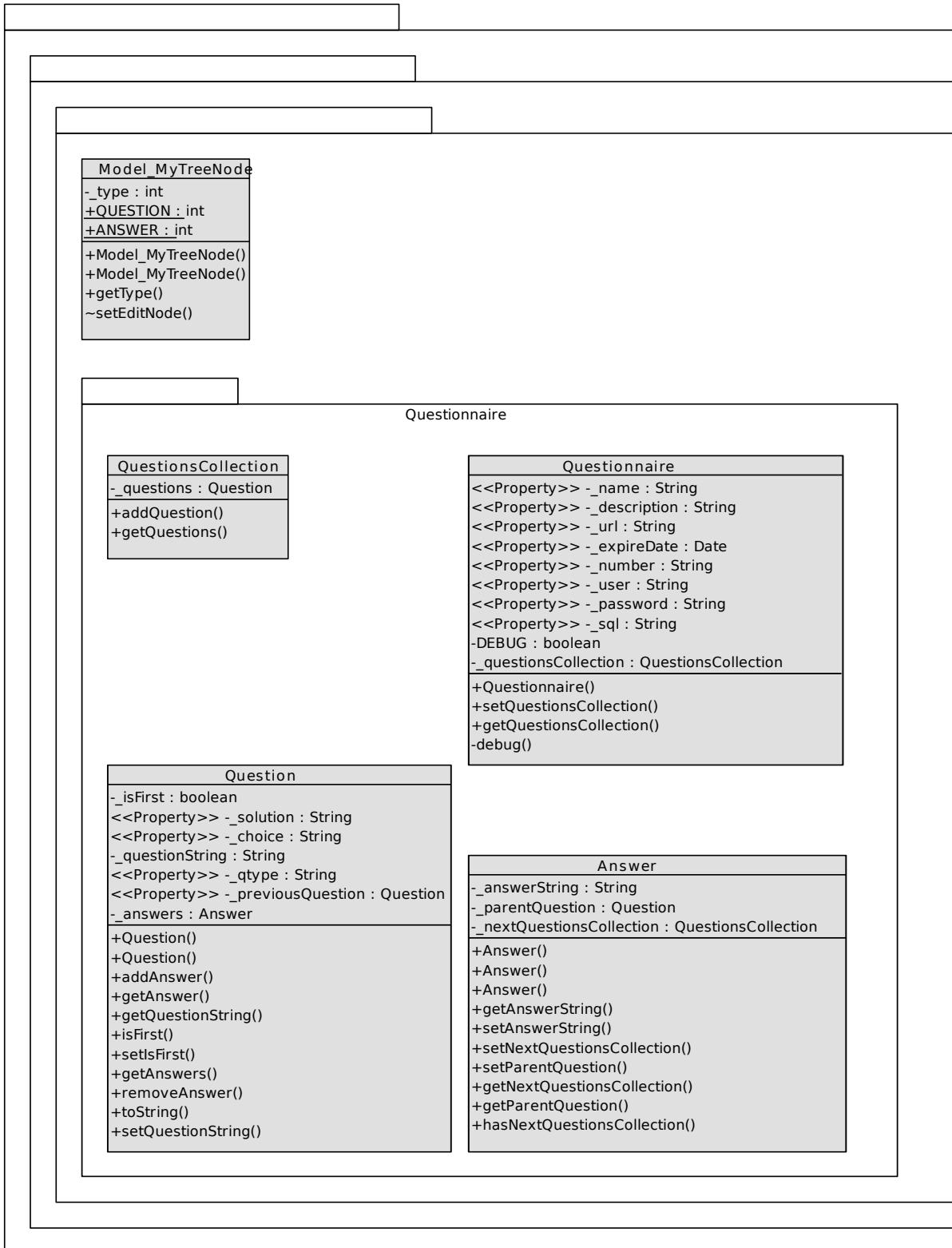


Abbildung 14.: Klassen des Designer-Frontends Teil 1

5. Entwurf eines generischen Frameworks für mobile Umfragen

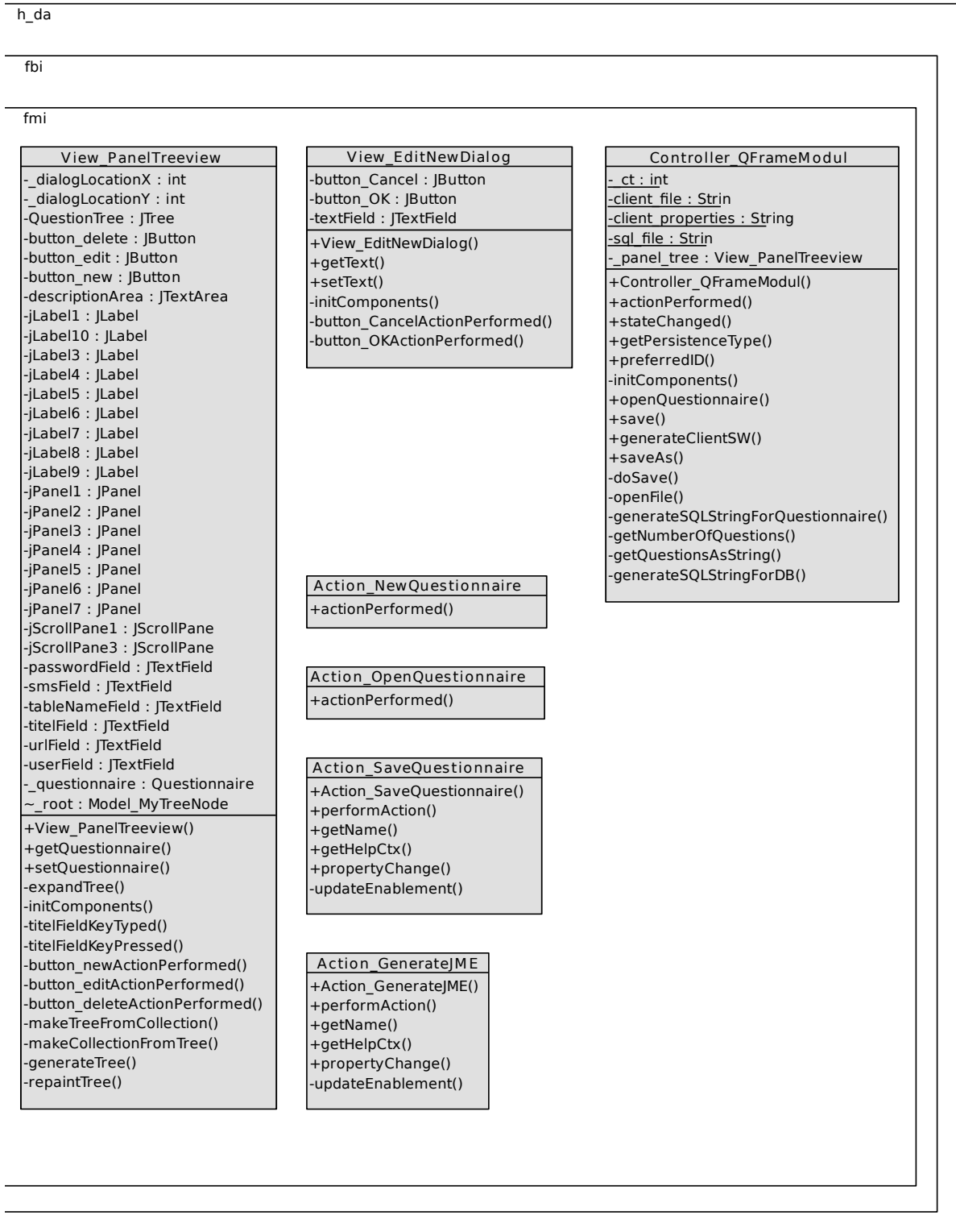


Abbildung 15.: Klassen des Designer-Frontends Teil 2

5. Entwurf eines generischen Frameworks für mobile Umfragen

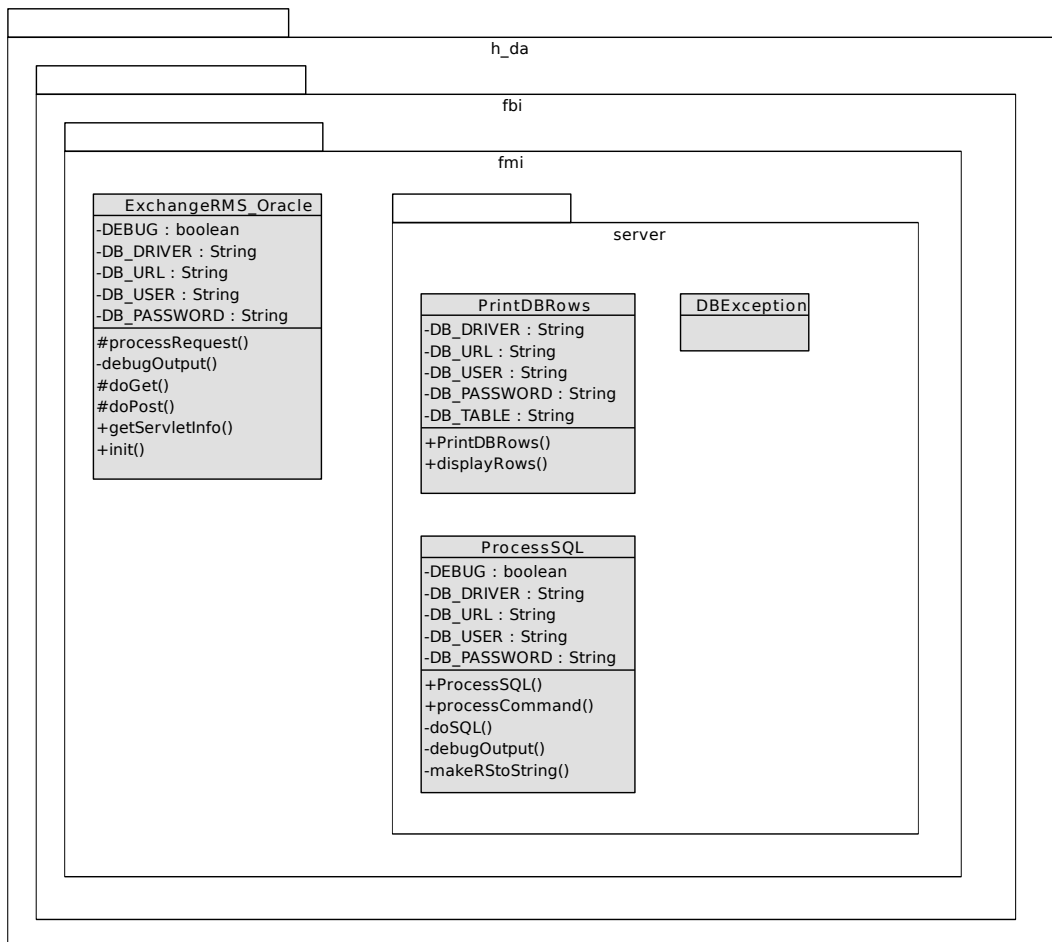


Abbildung 16.: Klassen des Empfangs-Server

5.4.4. SMS-Hub

Das SMS-Hub-Subsystem dient als Zwischenstation für das Weiterleiten von Umfrage-Antworten an den Empfangs-Server. Es kommt zum Einsatz, wenn der Teilnehmer der Umfrage nicht bereit ist oder nicht die Möglichkeit hat, eine Internetverbindung mit seinem Mobiltelefon herzustellen und somit keine Direktverbindung zum Empfangsserver möglich ist. Aufgrund dessen muss der SMS-Hub auf eintreffende SMS von der Client-Software warten. Dafür können SMS neben einer Empfangsnummer zusätzlich in JavaME mit einem Ziel-Port versehen werden, an den diese geschickt werden. Dazu registriert sich eine Anwendung beim Betriebssystem als Empfänger von Nachrichten auf diesem Port. Das Betriebssystem leitet dann SMS auf diesem Port an die Anwendung weiter. Ist kein Port angegeben oder keine Anwendung lauscht am entsprechenden Port, wird die SMS ganz normal durch das Betriebssystem des empfangenden Mobiltelefons ausgewertet, wie andere SMS auch.

Treffen SMS am entsprechenden Port beim SMS-Hub ein, müssen diese ausgewertet und der Inhalt der SMS an den Empfangs-Server weitervermittelt werden. Dabei ist es weiterhin möglich eintreffende SMS entgegen nehmen zu können, während eine Antwort an den Empfangs-Server vermittelt wird. Der SMS-Hub blockiert nicht, sondern verteilt die Aufgaben an einzelne Threads.

Der SMS-Hub benötigt für diese Aufgaben zwei Schnittstellen: Erstens müssen SMS entgegen genommen werden können, dazu dient die Komponente *SMSListener*. Die Möglichkeit Antworten mittels SMS zurück an den Absender zu senden, wird nicht implementiert. Zweitens ist zur Seite des Empfangs-Servers hin eine HTTP-Schnittstelle erforderlich, um die eingetroffenen SMS-Inhalte weiterleiten zu können. Diese Schnittstelle muss einen Hin- und Rückkanal bieten, um Antworten vom Empfangs-Server entgegen nehmen zu können. Die HTTP-Schnittstelle ist dabei identisch mit jener der Client-Software. Implementiert wird sie durch die *SMSReader*-Komponente.

Diagramm [→ 17, Seite 59] zeigt das Klassenmodell des SMS-Hub.

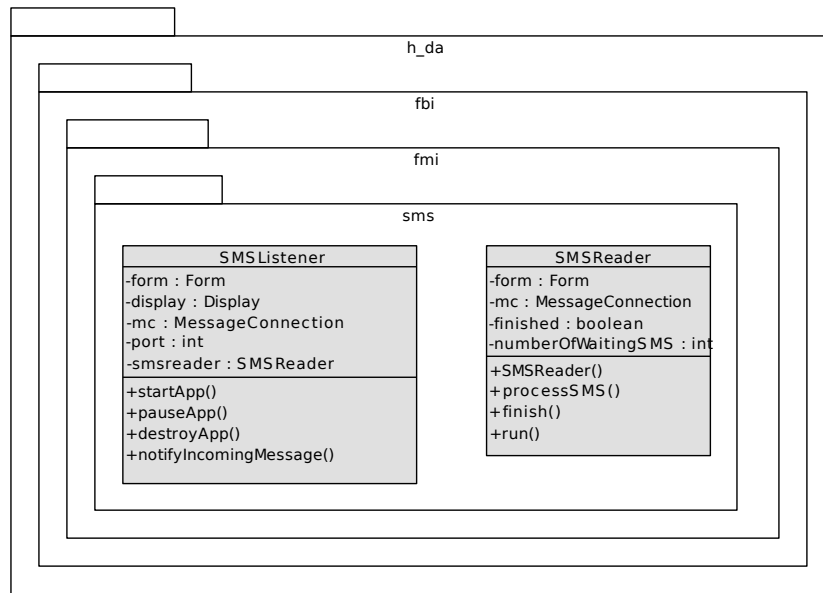


Abbildung 17.: Klassen des SMS-Hub

5.5. Entwurf des Datenmodells

Beim Entwurf des Datenmodells ist es wichtig, die Möglichkeit von Verzweigungen zu implementieren. Es muss also möglich sein, abhängig von einer Antwort, wieder zu bestimmten Fragen zu gelangen. Dazu wird eine Baumstruktur generiert, die aus *QuestionsCollections*, *Questions* und *Answers* besteht und das Traversieren über alle Elemente ermöglicht (siehe Diagramm [→ 18, Seite 60]). Die Struktur ist ein echter Baum, d.h. eine Zusammenführung von Verzweigungen ist nicht mehr möglich.

Zum Traversieren von Bäumen existieren verschiedene Vorgehensweisen: Pre-order, post-order, in-order und level-order [Wikipedia 2009a]. Die hier gewählte Traversierung findet in pre-order statt. Somit ist der generelle Ablauf wie folgt: Es wird über alle Fragen einer ausgewählten *QuestionsCollection* iteriert, angefangen bei der „QuestionsCollection0“. Dabei werden zuerst, an der Wurzel startend, die Fragen und zugehörigen Antworten bis zu den Blättern durchlaufen. Anschließend springt das System zurück zum nächst höheren Knoten - der übergeordneten *QuestionsCollection* -, also zur letzten Verzweigung und geht von dort wiederum bis zu den Blättern durch. So werden nach und nach die

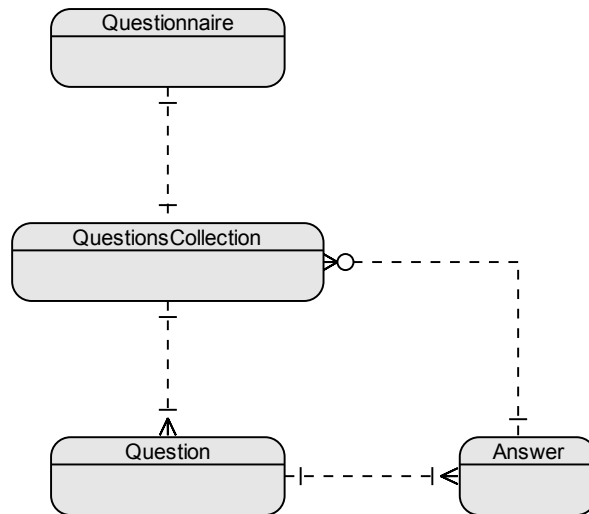


Abbildung 18.: Datenmodell

Teilbäume abgearbeitet. Die Fragen der „QuestionsCollection0“ sind dabei der Wurzelknoten und werden jeweils zwischen den Teilbäumen bearbeitet.

Ein Beispielaufbau einer Umfrage wird in Abbildung [→ 19, Seite 61] gezeigt. Dabei wird als erste Frage „Question0: Besitzen Sie ein Auto?“ angezeigt. Sollte hier „Ja“ ausgewählt werden, werden als nächstes die Fragen „Question2: Welche Marke?“ und „Question3: Welche Farbe?“ angezeigt. Danach wird die Frage „Question1: Besitzen Sie ein Fahrrad?“ angezeigt. Dies geschieht auch, sollte bei der ersten Frage „Question0“ als Antwort „Nein“ gewählt worden sein. Abhängig von der Antwort hierauf wird entweder die Frage „Question4: Wie viele Gänge hat es?“ bei „Ja“ oder die die Frage „Question5: Warum nicht?“ bei „Nein“ gezeigt.

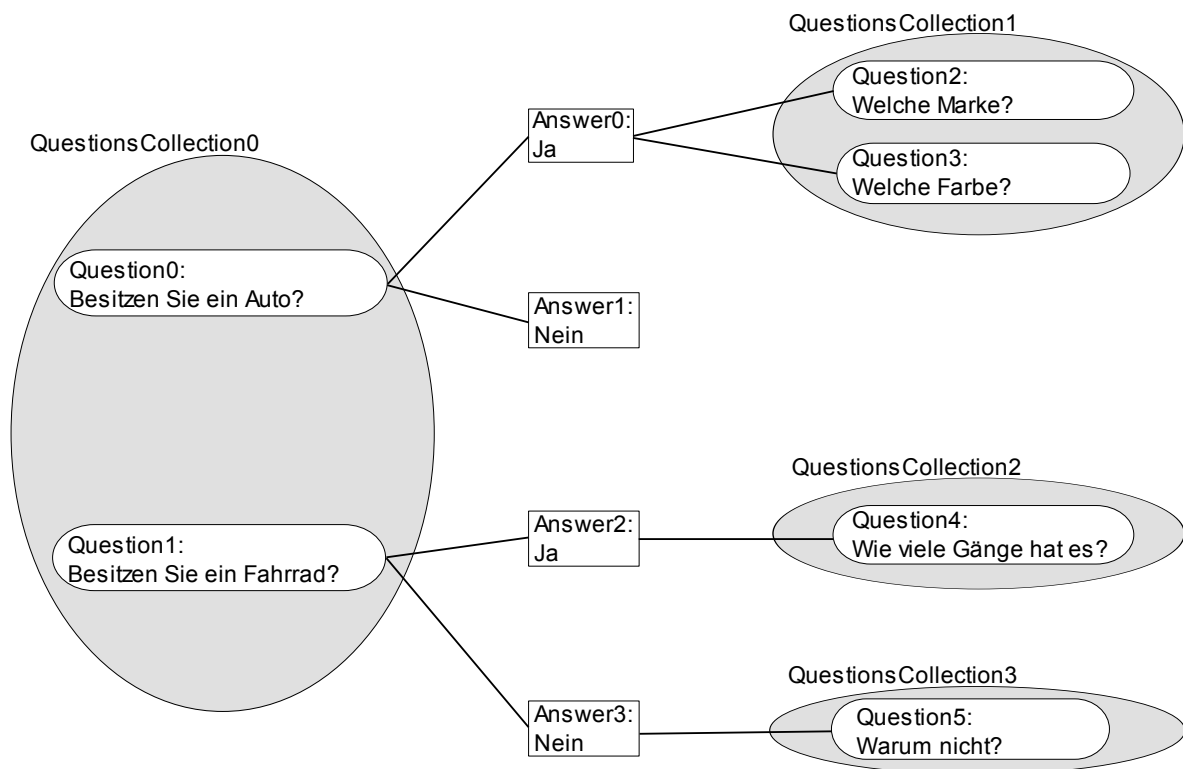


Abbildung 19.: Aufbau einer Beispiel-Umfrage

6. Implementierung des generischen Frameworks

6.1. Client-Software

Die Client-Software ist eine MIDlet-Suite [→ 2.3, Seite 23], also eine JavaME-Anwendung. Sie ist so konzipiert, dass beliebige Umfragen dargestellt werden können. Dazu liest das MIDlet beim Programmstart die Datei *cf.xml* im Archiv der MIDlet-Suite ein und erstellt daraus die entsprechenden Fragen. Die Darstellungsbildschirme dieser Fragen werden als „Forms“ bezeichnet, wobei eine Form wiederum von der JavaME-Klasse *Screen* erbt. Die Darstellungsbildschirme nutzen also die High-Level-API der JavaME.

Listing [→ 6.1, Seite 63] zeigt die XML-Repräsentation einer Umfrage auf dem Client. Zwischen der XML-Repräsentation einer gespeicherten Umfrage durch das Designer-Frontend und der Umfrage-Beschreibung für die Client-Software bestehen geringe Unterschiede: Die XML-Tags wurden gekürzt, weil die Beschreibungs-Datei durch die Client-Software selbst geparsed werden muss und die verwendeten Hilfsbibliotheken Probleme mit den ursprünglichen Tags haben. Darüber hinaus enthält die Umfrage-Beschreibung eine zusätzliche SQL-Sequenz.

Da keine Bibliotheken für JavaME existieren, um aus einer XML-Repräsentation direkt wieder Objekte zu erstellen, wurde eine rekursive Funktion zum Parsen entworfen (siehe Listing [→ 6.2, Seite 64]). Zur Erleichterung wird auf die freie Bibliothek kXML2¹ zurück gegriffen. Sie ist ein SAX-Parser² und erlaubt ein lineares Parsen von XML-Dokumenten. Aus den geparsen Antworten werden die Forms-Objekte erstellt, die eine

¹kXML2 <http://kxml.sourceforge.net/>

²SAX: Abk. f. Simple API for XML. Einfache API um XML-Dokumente sequentiell verarbeiten zu können. [Wikipedia \[2009g\]](#)

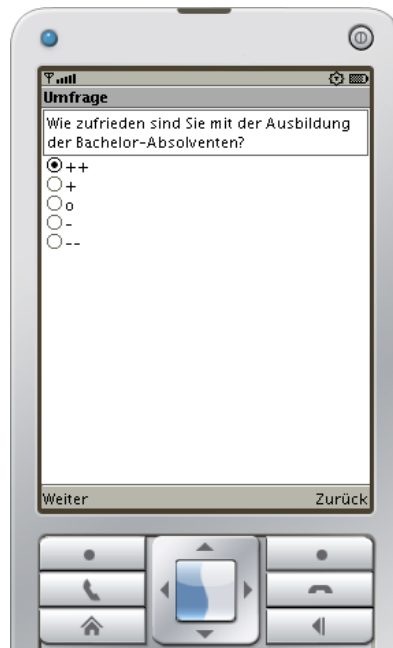


Abbildung 20.: Beispiel-Form mit einer Choicegroup

Repräsentation der Darstellung der Fragen samt Antworten sind. Jede Frage ist ein eigenes Form-Objekt. Alle Form-Objekte werden einem Vector hinzugefügt, um über diese iterieren zu können.

```
1 <questionsCollection>
2   <questions>
3     <Question>
4       <questionString>Besitzen Sie ein Auto?</questionString>
5       <answers>
6         <Answer>
7           <answerString>Ja</answerString>
8         <nextQuestionsCollection>
9           <questions>
10            <Question>
11              <questionString>Welche Marke?</questionString>
12              <answers>
13                ...
```

Listing 6.1: Aufbereitetes XML für den Client

Bei jedem Wechsel eines Form-Objekts wird die zuvor ausgewählte Antwort geparsed und, abhängig davon, das nächste Form-Objekt dargestellt. Abbildung [→ 20, Seite 63] zeigt ein solches Form-Objekt. Um immer das richtige Form-Objekt innerhalb des Vektors zu finden und ein Vor- und Zurückspringen zwischen diesen zu ermöglichen, wird ein zweiter Vektor genutzt, an den das zuletzt angezeigte Form-Objekt angefügt wird.

```
1 private void printQuestion(QuestionsCollection questionCollection
2     , String prevAnswer, QuestionForm parent) {
3
4     // iterate through all Questions of this QuestionsCollection
5     Enumeration questionEnum = questionCollection.getQuestions().
6         elements();
7     while (questionEnum.hasMoreElements()) {
8         form = new QuestionForm("Questionnaire ", parent);
9         form.setNeededAnswer(prevAnswer);
10
11        Question q = (Question) questionEnum.nextElement();
12        StringItem si = new StringItem("", q.getQuestionString());
13
14        // generate ChoiceGroup
15        ChoiceGroup choiceGroup = new ChoiceGroup("", ChoiceGroup.
16            EXCLUSIVE);
17        choiceGroup.setFitPolicy(ChoiceGroup.TEXT_WRAP_ON);
18
19        // iterate through all Answers of this Question
20        Enumeration answerEnum = q.getAnswers().elements();
21        while (answerEnum.hasMoreElements()) {
22            Answer a = (Answer) answerEnum.nextElement();
23            choiceGroup.append(a.getAnswerString(), null);
24            if (a.hasNextQuestionsCollection()) {
25                // branch from an answer to a new QuestionsCollection
26                printQuestion(a.getNextQuestionsCollection(), a.
27                    getAnswerString(), form);
28            }
29        }
30        // generate Form
31        ...
32
33        // add Form to Vector
34        _forms.addElement(form);
```

```
32 }  
33 }
```

Listing 6.2: Rekursive Erstellung der Forms aus der XML-Beschreibung der Umfrage

Nach Beantwortung einer Frage wird die gepasste Antwort in einem RecordStore lokal gespeichert. Nachdem alle Fragen beantwortet wurden, wird anhand der gegebenen Antworten ein SQL-Befehl generiert. Dazu ist in der Umfrage ein vorbereiteter SQL-Befehl enthalten, der nun angepasst wird. Der Befehl beinhaltet dabei die Fragen der Umfrage als Werte im SQL-String. Diese Fragen werden durch die gegebenen Antworten ersetzt und der Befehl ist fertig. Wichtig hierbei ist, dass keine Fragen mehrfach vorkommen dürfen, da sonst der Ersetzungs-Vorgang möglicherweise nicht korrekt funktioniert. Eine Überprüfung auf gleiche Fragen findet deshalb im Designer-Frontend bei der Erstellung der Umfrage statt.

Für die Zukunft sind weitere Verbesserungen vorstellbar, wurden aber in der aktuellen Version nicht implementiert:

- Die Auswahl mehrerer Umfragen: Statt eine statische einzelne Umfrage könnte als Startbild eine Liste mit vorhandenen Umfragen angezeigt werden. So könnten die Beschreibungen der Umfragen, anstatt der gesamten Anwendung, an die Geräte verteilt werden. Es könnte dazu ein Abonnement-System verwendet werden. Die Umfrage-Beschreibungen könnten dabei mittels SMS an registrierte Geräte gesendet werden, wo die Client-Software auf diese speziellen SMS reagiert und in die Liste einfügt. Problematisch hierbei ist aber die Identifikation der Nutzer und ihrer Geräte und ob die nötige Client-Software bereits installiert ist.
- Die Client-Software könnte auf eintreffende Antwort-SMS vom Empfangs-Server lauschen und diese darstellen.
- Neben den implementierten Auswahl-Gruppen sind weitere Antwort-Typen für die Forms möglich, z.B. freie Texte.

6.2. Designer-Frontend

Um eine Verzweigung von Umfragen zu ermöglichen, ist eine baumartige Darstellung der Fragen/Antworten einer Umfrage gewählt worden. Dies ermöglicht ein intuitives Navigie-

ren innerhalb der Fragen/Antworten und erlaubt ein schnelles Erfassen der Reihenfolge der Fragen durch den Ersteller. Abbildung [→ 21, Seite 67] zeigt das Hauptfenster des Designer-Frontends mit einer geöffneten Beispielumfrage. Dieses ermöglicht das Anlegen, Öffnen und Speichern von Umfragen in separaten Tabs. Die Anwendung nutzt die Netbeans-Plattform³ als Basis und ist in Java implementiert.

Die geöffnete Beispielumfrage zeigt im Vordergrund ein Eingabefenster und im Hintergrund die Eingabemaske zur aktuellen Umfrage. Diese Eingabemaske gliedert sich in einen oberen und einen unteren Bereich. Im oberen Bereich können allgemeine Angaben zur Umfrage eingegeben werden wie der Name der Umfrage, der Name der anzulegenden Tabelle in der Datenbank, die URL des Empfangs-Servers, Nutzernamen und Passwort zum Authentifizieren am Empfangs-Server und eine Mobilfunknummer als Adresse des SMS-Hub. Der untere Bereich zeigt die Fragen und Antworten der Umfrage in einer Baumstruktur. Damit ist ein einfacher Überblick über die Struktur der Umfrage gewährleistet und somit die Abfolge der Antworten und Fragen ersichtlich.

Das Eingabefenster im Vordergrund ermöglicht es, neue Fragen/Antworten zu erstellen bzw. zu editieren (durch Mehrfach-Klicken in der Baumstruktur oder den entsprechenden Knopf). Der Eingabedialog passt sich dabei dynamisch dem ausgewählten Eintrag in der Baumstruktur an, d.h., sollte in der Baumstruktur eine Frage ausgewählt sein, so ist es nur möglich zugehörige Antworten anzulegen. Bei einer Antwort als Auswahl, können entsprechend nur neue Fragen als Unterpunkte angelegt werden.

So entsteht bei der Erstellung einer Umfrage eine ständige Abfolge von Fragen und Antworten. Fragen/Antworten der gleichen Ebene gehören dabei zu einer *QuestionsCollection* (siehe Datenmodell [→ 5.5, Seite 59]) und werden anhand ihrer Anordnung in der Baumstruktur von oben nach unten nacheinander dargestellt. Dabei dürfen keine Fragen mehrfach vorkommen - auch nicht in verschiedenen Ebenen. Hintergrund ist die Schwierigkeit in der Client-Software, die Antworten der richtigen Position im SQL-Befehl zuzuordnen. Da die Reihenfolge von Fragen und Antworten aufgrund der Möglichkeit von Verzweigungen und dem dynamischen Generieren der zugehörigen Oberflächen nicht gesichert ist, erfolgt eine Zuordnung anhand der Fragetexte. Eine gegebene Antwort er-

³NetBeans IDE and NetBeans Platform are based on software from netbeans.org, which has been dual licensed under the Common Development and Distribution License (CDDL) and the GNU General Public License version 2 with Classpath exception. For more information, please visit www.netbeans.org.

6. Implementierung des generischen Frameworks

setzt hierbei immer eine spezifische Frage im SQL-Befehl, deswegen sind gleiche Fragen nicht erlaubt (näheres hierzu siehe im Kapitel [→ 6.1, Seite 62]).

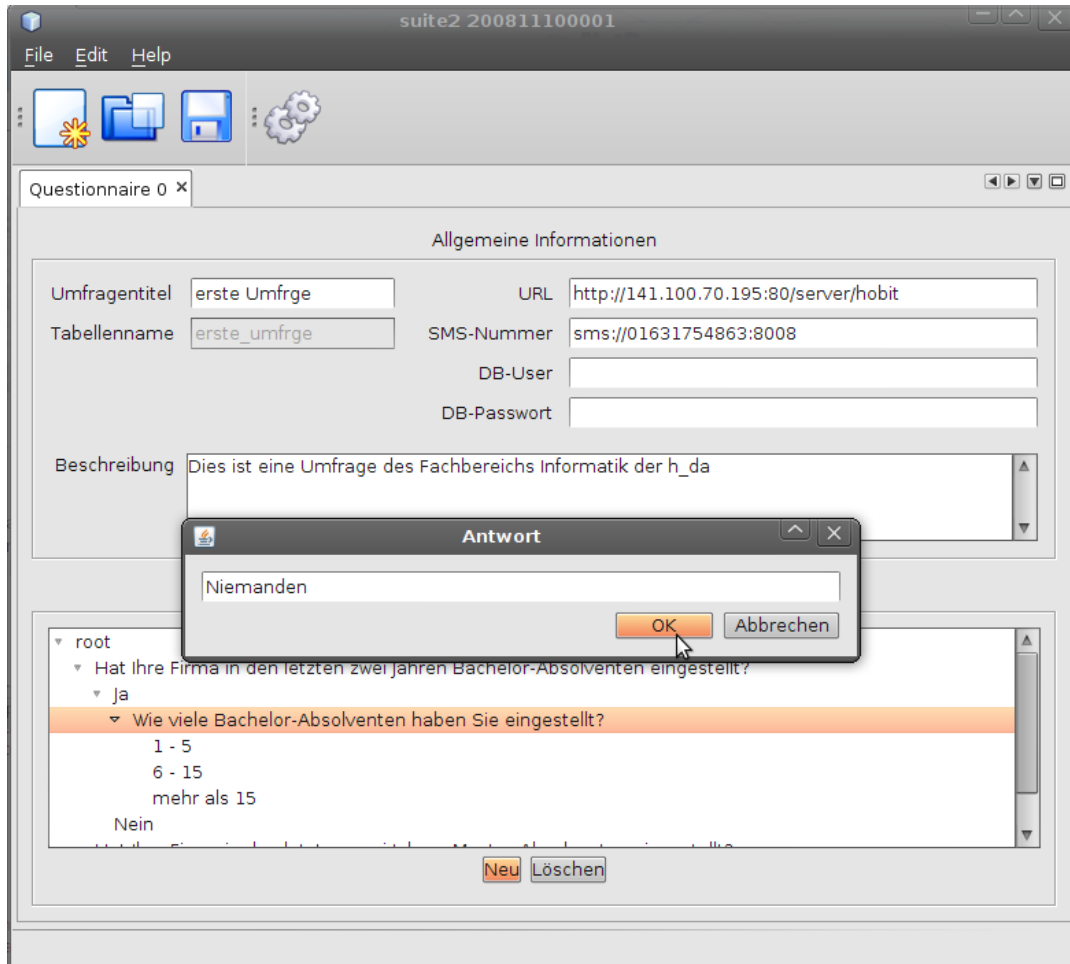


Abbildung 21.: Geöffnete Umfrage im Designer-Frontend

Durch den *Erstellen*-Befehl wird im Hintergrund ein passender SQL-Befehl generiert, um eine entsprechende Tabelle in einer Datenbank anlegen zu können. Listing [→ 6.3, Seite 67] zeigt ein Beispiel für einen derartigen SQL-Befehl.

```
1 CREATE TABLE erste_umfrage(id INTEGER NOT NULL, frage1 INTEGER,  
    frage2 INTEGER, frage3 INTEGER, PRIMARY KEY(id))
```

Listing 6.3: Aufbau des SQL-Befehls

Außerdem wird die Client-Software generiert. Dazu steht die gesamte Client-Software samt zugehöriger Bibliotheken als Sourcecode im Programmverzeichnis des Designer-Frontends zur Verfügung. Der Befehl ändert nun die Beschreibung und den Namen des zu Erstellenden MIDlets in der Datei *MIDlet/nbproject/project.properties* und ersetzt die Datei *MIDlet/ressources/cf.xml*. Letztere enthält die vollständige Beschreibung der Umfrage. Sie ist eine leicht abgewandelte XML-Repräsentation des Datenobjekts *Questionnaire* aus dem Designer-Frontend. Sie wird durch die freie Bibliothek *XStream* erstellt, die beliebige Objekte zu XML serialisieren kann, bzw. aus diesen XML wieder Objekte erstellen kann. Die MIDlet-Suite wird jedesmal neu übersetzt, um sie, nach einer Veränderung der Umfrage-Beschreibung, neu signieren zu können [→ 2.3, Seite 23].

Gespeicherte Umfragen im Designer-Frontend sind als XML-Dateien hinterlegt. Generiert werden diese automatisch durch das Serialisieren des Datenobjekts *Questionnaire* (siehe Datenmodell [→ 5.5, Seite 59]) durch die freie Bibliothek *XStream*⁴ zu einer XML-Repräsentation. Listing [→ 6.4, Seite 68] zeigt die baumartige Struktur der XML-Datei mit den Fragen und Antworten. Beim Laden gespeicherter Umfragen wird der umgekehrte Weg gegangen: Das Datenobjekt wird durch de-serialisieren der XML-Datei mittels *XStream* erstellt und anschließend dargestellt. Das Subsystem wurde nach dem Model-View-Controller Prinzip entworfen.

```
1 <__questionsCollection>
2 <__questions>
3   <org.h__da.fbi.fmi.Questionnaire.Question>
4     <__questionString>Besitzen Sie ein Auto?</__questionString>
5     <__answers>
6       <org.h__da.fbi.fmi.Questionnaire.Answer>
7         <__answerString>Ja</__answerString>
8         <__nextQuestionsCollection>
9           <__questions>
10            <org.h__da.fbi.fmi.Questionnaire.Question>
11              <__questionString>Welche Marke?</__questionString>
12              <__answers>
13 ...
```

Listing 6.4: XML-Beschreibung einer Umfrage

⁴XStream <http://xstream.codehaus.org/index.html>

Folgende Verbesserungen sind vorstellbar:

- Die automatische Erstellung der Tabellen in der Datenbank. So würde der Umweg über eine SQL-Anweisung entfallen, die der Anwender zusätzlich noch ausführen muss.
- Die vollständige Umstellung auf eine Implementierung im Web, wie auch andere Lösungen, oder:
- die Anbindung an existierende Werkzeuge zur Erstellung von Umfragen.

6.3. Empfangs-Server

Der Empfangs-Server ist als Java-Servlet implementiert und läuft innerhalb eines Tomcat-Servers. Er reagiert auf HTTP-Anfragen auf einer bestimmten URL und parsed die darauf eintreffende HTTP-Nachricht (zum prinzipiellen Aufbau von HTTP-Nachrichten siehe [Wikipedia 2009c]). Beginnt der Inhalt der Nachricht mit *SQL*, so nimmt der Server an, es handelt sich um eine SQL-Anweisung eines Clients und führt diese aus. Listing [→ 6.5, Seite 69] zeigt eine typische HTTP-Nachricht.

```
1 POST /server/hobit HTTP/1.1
2 enctype: MULTIPART/FORM-DATA
3 Content-Length: 130
4 Host: 141.100.70.195:80
5
6 SQL Insert into fbi_umfrage(frage1, frage2, frage3, frage4,
   frage5, frage6, frage7, frage8) values
   ('0','0','0','','0','0','0','')
```

Listing 6.5: Beispiel einer eintreffenden HTTP-Nachricht

Clients können sowohl einzelne Instanzen der Client-Software (auf verschiedenen Mobiltelefonen) sein, oder SMS-Hubs, die Antworten weiterleiten. Zurzeit findet weder eine Authorisation noch eine Authentifizierung der Clients statt. Auch die Überprüfung auf Korrektheit, der im HTTP-Body enthaltenen SQL-Anweisung, findet nicht statt. Statt dessen wird die Anweisung an eine Datenbank-Klasse weiter gegeben, die sie auf einer Datenbank ausführt. Während dieser Zeit wird die HTTP-Verbindung zum Client offen

gehalten. Liefert die Datenbank auf den ausgeführten Befehl eine Antwort, so wird diese von der Datenbank-Klasse an das Servlet zurück gegeben, ansonsten wird eine Bestätigungsmeldung zurück geliefert. Das Servlet sendet nun diese Antwort, über die noch offene Verbindung, an den Client zurück.

Für die Kommunikation mit der Datenbank wird JDBC⁵ verwendet. Die nötigen Informationen, um sich mit der entsprechenden Datenbank verbinden zu dürfen, liegen dem Servlet in einer Konfigurationsdatei vor und werden bei jedem Aufruf des Servlets eingelesen und ausgewertet.

Die Implementierung verwendet zur Speicherung von Umfrage-Antworten eine Oracle-Datenbank, eine Microsoft SQL-Datenbank wurde aber ebenso erfolgreich als Empfänger getestet.

Neben der eigentlichen Empfangs-Funktionalität existiert eine JSP-Seite, um die aktuellen Inhalte der Datenbank-Tabellen in einer Webseite anzeigen zu können. Dabei wird lediglich der Tabelleninhalt aufgelistet, eine Analyse und Auswertung ist nicht möglich.

Für dieses Subsystem sind weitere Verbesserungen vorstellbar:

- Die Unterstützung von HTTPS zur Sicherung der Datenübertragung.
- Eine Prüfung der Berechtigung des übermittelnden Clients.
- Eine Prüfung der SQL-Anweisung, ob diese schädlichen Code enthält.
- Funktionen, um Auswertungen von Datenbeständen durchführen und darstellen zu können.

6.4. SMS-Hub

Die vorliegende Version des SMS-Hub ist als JavaME-Programm implementiert. Dabei lauscht ein Thread auf eintreffende SMS an Port 8008. Ankommende SMS werden geparsed und die SQL-Anweisung ausgelesen. Anschließend baut der Hub eine Verbindung zum Internet auf (z.B. mittels GPRS) und übermittelt die SQL-Anweisung an die

⁵JDBC: Abk. f. Java Database Connectivity. [Wikipedia \[2009d\]](#)

Empfangs-URL. Er übernimmt dabei die Aufgabe der Client-Software: Das Übermitteln der Umfrage-Antworten. Dabei werden die Daten nicht weiter verändert und für den Empfangs-Server besteht kein Unterschied zur direkten Variante des Versendens durch die Client-Software. Die empfangene SMS ist dabei folgendermaßen aufgebaut: Ein Header, mit der Mobilfunknummer als Absender-Information, und ein Body, mit der URL des Empfangs-Server und der SQL-Anweisung. Der Header wird automatisch durch die JavaME-APIs erstellt, die Rufnummer des Absenders wird dabei automatisch der SMS hinzu gefügt. Um die Telefonnummer des Absenders zu verbergen, muss dieser generell die Rufnummernübermittlung deaktivieren. Listing [→ 6.6, Seite 71] zeigt ein Beispiel für einen SMS-Body.

```
1 http://141.100.70.195:80/SQLServer/SQLReceiver|SELECT * FROM  
   table1
```

Listing 6.6: Aufbau eines SMS-Body

Sollte eine SMS mehr als die erlaubten 160 Zeichen enthalten, z.B. durch lange SQL-Anweisungen, wird sie vom Betriebssystem automatisch in mehrere SMS zerlegt, übertragen und am Ziel wieder zu einer langen SMS zusammengesetzt. So muss der SMS-Hub sich nicht um diese Problematik kümmern.

Die geöffnete HTTP-Verbindung dient gleichzeitig auch als Rückkanal, um Antworten vom Empfangs-Server erhalten zu können.

Als weitere Verbesserungen des SMS-Hubs sind folgende Punkte vorstellbar, aber noch nicht implementiert:

- Ein automatischer Programmstart beim Eintreffen von Nachrichten am korrekten Port. Dazu könnte sich die Anwendung mittels dem *PushRegistry*-Befehl beim Betriebssystem als zu benachrichtigende Anwendung registrieren.
- Versenden einer Rückantwort-SMS vom Empfangs-Server an die Client-Software. Sollte die gesendete SQL-Anweisung eine Antwort generieren, so wird diese vom Empfangs-Server an den SMS-Hub zurück geliefert. Der SMS-Hub könnte diese dann an das entsprechende Mobiltelefon mittels SMS weiterleiten. Das Empfangen und Senden der SQL-Anweisung müsste dabei in einem gemeinsamen Thread

geschehen, da nur so eine Zuordnung von Rückantworten zu den richtigen Mobilnummern möglich ist. Die entsprechende Telefonnummer ist im Header der empfangenen SMS enthalten.

- Speichern aller empfangenen SMS. Der SMS-Hub könnte alle eintreffenden SMS in einem RecordSet speichern, so dass keine Informationen bei einem Programmabsturz verloren gehen.
- Sichern aller empfangenen SMS in einer Datei: Automatisch oder durch eine Nutzeraktion könnte eine Datei angelegt werden, die alle SMS in Textversion enthält. Diese Datei kann dann einfach als Backup vom Mobiltelefon kopiert werden.

7. Validierung des Framework

Dieses Kapitel beschreibt die Validierung des gesamten Frameworks. Dazu wird zuerst der Ablauf der einzelnen Tests beschrieben, anschließend wird das Framework evaluiert, indem die erstellten Funktionen mit den Anforderungen verglichen werden.

Für das Designer-Frontend wurden Modultests mittels JUNIT¹ als Test-Framework durchgeführt. Dabei wurden die einzelnen Komponenten mittels automatisch generierter und dann speziell angepasster Test-Funktionen geprüft.

Für die JavaME-Anwendungen existiert keine derartige automatische Testumgebung, so dass die Komponenten sowohl für die Client-Software, als auch für den SMS-Hub mittels selbst geschriebener Testfunktionen geprüft wurden. Das Subsystem SMS-Hub wurde durch die Fallstudie „Hobit“ getestet, wo es erfolgreich alle Antworten der Quiz-Teilnehmer an den Empfangs-Server weiterleitete.

Die Komponenten des Empfangs-Server Subsystems wurden nicht speziell geprüft, sondern im Laufe der Zeit getestet und verbessert, da dieses Subsystem bereits vor Erstellung der Masterarbeit für ein Jahr im Einsatz war. Der Empfangs-Server wurde dabei während zweier Computer-Praktika getestet, wo es SQL-Anfragen bearbeitete und erfolgreich auf einer Datenbank ausführte.

Der Systemtest - also ein vollständiger Test aller Komponenten - wurde mit Hilfe der Fallstudie „IT-Forum“ durchgeführt. Dabei musste sich das gesamte System unter realen Bedingungen bewähren. Dafür wurden zunächst mit dem Designer-Frontend zwei Umfragen erstellt und daraus die Client-Software generiert. Danach wurden die beiden Client-Anwendungen verteilt. Die Antworten der Teilnehmer wurden dabei erfolgreich

¹„... a simple framework to write repeatable tests. It is an instance of the xUnit architecture for unit testing frameworks.“: <http://junit.sourceforge.net>

über den SMS-Hub an den Empfangs-Server weitergeleitet und von diesem in eine Datenbank eingetragen. Es traten bei diesem Test keine Probleme auf.

Die implementierten und getesteten funktionalen Anforderungen an das System sind:

1. Generische Framework: Die Generik des Framework wurde erreicht, indem eine Beschreibungssprache für die Umfragen entwickelt und genutzt wurde.
2. Die einfache Verteilung der Umfragen ist durch die Verwendung von eigenständigen Umfrage-Programmen (also der Client-Anwendung) gewährleistet. Die Verwendung von JavaME als Grundlage ermöglicht das Verteilen der Umfrage mittels Internet, Bluetooth - wie in den Fallstudien - und so weiter.
3. Erstellung neuer Umfrage mittels Drag&Drop: Diese Anforderung wird durch das Designer-Frontend gewährleistet. Die Baumstruktur zur Darstellung der Fragen und Antworten erlaubt dabei auch Drag&Drop.
4. Die Superdistribution ist ebenfalls durch die Verwendung von separaten JavaME-Programmen gewährleistet, da Teilnehmer die Client-Software einfach an andere Personen weiter verschicken können, z.B. über Bluetooth oder Infrarot [→ 8.3, Seite 87].
5. Die Umfrage ist auf vielen Geräten lauffähig, da JavaME als Programmiersprache verwendet wurde, welches von den meisten Mobiltelefonen unterstützt wird [→ 2.3, Seite 23].
6. Die Verzweigung von Umfragen wird durch das Datenmodell gewährleistet und durch die Baum-Darstellung im Designer-Frontend grafisch verdeutlicht.

Eine funktionale Anforderung wurde aus Zeitmangel nicht vollständig im System implementiert und konnte daher auch nicht getestet werden:

1. Das Zurück-Liefern von Antworten an die Teilnehmer. Dabei fehlt die Möglichkeit Antworten vom Empfangs-Server über den SMS-Hub zurück zu liefern - wie dies aussehen könnte und was dabei zu beachten ist, wird in [→ 6.4, Seite 70] angesprochen. Außerdem kann die Client-Software zwar Antworten vom Empfangs-Server entgegen nehmen (bei einer Datenverbindung, nicht bei Verwendung von SMS), allerdings ist die anschließende Darstellung der Antwort auf dem Gerät nicht implementiert.

Nichtfunktionale Anforderungen wie Effizienz, Wartbarkeit usw. wurden während der Entwicklung weitestmöglich beachtet. Die Sicherheitsaspekte bezüglich dem Schutz der Kommunikation konnten noch nicht umgesetzt werden, so ist momentan nur eine Datenverbindung über das ungesicherte HTTP vorgesehen, kann allerdings recht einfach auf HTTPS erweitert werden. Ebenso ist der Schutz der Daten gegen Manipulation nicht gewährleistet. Dies könnte durch eine Authentifizierung am Empfangs-Server erreicht werden (ein derartiges Feld ist im Designer-Frontend bereits vorgesehen). Näheres zu den Möglichkeiten diese Anforderungen umzusetzen, steht in den Unterkapiteln [→ 6.1, Seite 62], [→ 6.2, Seite 65] und [→ 6.3, Seite 69].

Ob die erstellte Software ethisch korrekt ist, hängt vom der Einsatzart und dem Einsatzzweck ab. In Unterkapitel [→ 8.6, Seite 99] werden hierzu genauere Aussagen getroffen.

Für das gesamte System wurde abschließend ein Abnahmetest gemacht. Dazu wurde die Client-Software von mehreren unabhängigen Personen getestet und für gut befunden.

8. Distributionsvarianten und Nutzerakzeptanz

Diese Kapitel betrachten die verschiedenen Möglichkeiten der Verteilung (also der Distribution), ebenso wie die Akzeptanz dieser neuen Konzepte durch die Benutzer. Dazu wird zunächst auf die rechtlichen Aspekte eingegangen, die zu beachten sind. Danach werden drei Fallstudien vorgestellt, um Daten für die Nutzerakzeptanz und die Distributionsvarianten zu besitzen. Abgeschlossen wird das Kapitel mit einer ethischen Betrachtung der Umfrage-Problematik.

8.1. Rechtliche Grundlagen zu Distributionsvarianten und dem Datenschutz

Bei der Bereitstellung von mobilen Dienstleistung sind mehrere Richtlinien bzw. Gesetze zu beachten:

1. Die Datenschutzrichtlinie für elektronische Kommunikation¹ der EU.
2. Das Telekommunikationsgesetz in Deutschland².
3. Das Gesetz gegen unlauteren Wettbewerb³.

¹Richtlinie 2002/58/EG des Europäischen Parlaments und des Rates vom 12. Juli 2002

²Telekommunikationsgesetz vom 22. Juni 2004 (BGBl. I S. 1190), zuletzt geändert durch Artikel 16 des Gesetzes vom 17. März 2009 (BGBl. I S. 550)

³Gesetz gegen den unlauteren Wettbewerb vom 3. Juli 2004 ((BGBl. I S. 1414)), zuletzt geändert durch den Artikel 2 des Gesetzes vom 29. Juli 2009 ((BGBl. I S. 2413, 2414))

Das Telekommunikationsgesetz ist die Bundesdeutsche Umsetzung der Vorgaben der EU durch die Datenschutzrichtlinie [Ranke 2004]. Das Gesetz legt den Anbietern von Dienstleistungen einige Hürden in den Weg, soll damit aber den Nutzer bzw. dessen Privatsphäre vor Missbrauch schützen. Es regelt den Umgang mit standortbezogenen Daten. Diese dürfen nur anonymisiert erhoben werden und für eine weitere Verwendung wird zusätzlich die Zustimmung des Nutzers benötigt. Darüber hinaus wird die Verwendung von Telekommunikationseinheiten an bestimmten Orten geregelt - so gelten etwa in Krankenhäusern oder Flugzeugen andere Regeln, als in privaten Wohnungen [Schäfer und Toma 2008, S. 26f.].

Hinzu kommt das Gesetz gegen unlauteren Wettbewerb. Es besagt, dass derjenige unlauter handelt, der einen Marktteilnehmer auf unzumutbare Weise belästigt. Das bedeutet, dass Inhalte an Nutzer nur gesendet werden dürfen, wenn diese vorher willentlich oder mutmaßlich ihre Einwilligung gegeben haben. Eine erklärte Einwilligung wird auch als *Opt-In-Verfahren* bezeichnet. Unter *Opt-In* versteht man ein Verfahren, das angewendet wird, um Nutzern Inhalte zukommen zu lassen bzw. allgemein Kontakt aufzunehmen. Dabei muss der Kunde aktiv Daten anfordern, damit er diese erhält. Dies geschieht z.B. mittels eines Eintrags in einer Liste. Problematisch hierbei kann sein, dass diese Liste mit fremden Einträgen manipuliert werden kann, also vermeintlich empfangswillige Kunden von anderen eingetragen werden. Hierfür existieren zum *Opt-In* zwei Erweiterungen: Das *Confirmed Opt-In*, bei dem eine Benachrichtigung über die Eintragung in einer solchen Liste an den Empfänger geschickt wird, worauf dieser entsprechend reagieren kann. Die zweite Erweiterung ist das *Double Opt-In-Verfahren*, bei dem der Empfänger auf die Benachrichtigung zusätzlich mit einer Bestätigung reagieren muss, um wirklich als neuer Kontakt aufgenommen zu werden [Wikipedia 2009e].

Daneben existiert das *Opt-Out-Verfahren*, bei welchem Nutzer, ohne ihr Zutun, in Verteilerlisten eingetragen und angesprochen werden. Der Nutzer muss sich anschließend aus diesen Listen entfernen, möchte er keine weiteren Informationen erhalten. [Wikipedia 2009f]

Neben der erklärten Einwilligung kann auch eine mutmaßliche Einwilligung vorliegen. Dies geschieht, wenn ein Nutzer willentlich Informationen anfordert bzw. seine Zustimmung zum Erhalt von Informationen gibt, indem er sich z.B. in unmittelbare Nähe eines speziellen Ortes begibt oder bestimmte Handlungen durchführt.

Für die eigentliche Nutzung von Diensten, z.B. dem Empfang von Werbebotschaften, unterscheidet man zwischen dem *Push*-Verfahren und dem *Pull*-Verfahren. Beim *Pull*-Verfahren werden Nachrichten/Dienste erst nach der Diensteanforderung zugestellt. Die *Opt-In*-Verfahren gehören zu dieser Kategorie. Beim *Push*-Verfahren hingegen werden dem Kunden Nachrichten/Dienste, ohne dessen vorheriges Zutun, zugesandt (z.B. mittels Massen-SMS). Es gilt als unseriös und der Versand von Werbe-E-Mails (Spam) ist in Deutschland nicht länger gesetzeskonform. Das *Opt-Out* zählt zu diesen Verfahren.

Jedwedes Versenden von Daten wie Bildern, ohne vorherige Zustimmung durch den Benutzer, ist somit nach dem Gesetz gegen den unlauteren Wettbewerb nicht erlaubt. Dies gilt es beim Bluetooth-Versand zu beachten.

Die Verwendung von Bluetooth erfolgt nach folgendem grundsätzlichen Schema:

1. Der Kunde aktiviert Bluetooth auf seinem Mobiltelefon.
2. Der Kunde begibt sich in Sendereichweite eines Bluetooth-Hotspots.
3. Der Kunde wird von diesem Bluetooth-Sender gefragt, ob er eine Anwendung erhalten möchte.
4. Stimmt der Kunde zu, wird die Anwendung an ihn übertragen.
5. Nachdem die Anwendung empfangen wurde, wird der Kunde durch das Betriebssysteme seines Mobiltelefons gefragt, ob er diese auch installieren möchte.
6. Danach kann der Kunde die Anwendung starten.
7. Lehnt der Kunde ab, wird die Kommunikation beendet und keine weiteren Daten übertragen.

Die erste Anfrage an den Kunden basiert dabei scheinbar auf dem *Push*-Verfahren. Das würde bedeuten, dass bereits die erste Anfrage nicht rechtens ist. Man könnte jedoch damit argumentieren, dass der Kunde durch zwei Handlungen eine mutmaßliche Einwilligung zum Erhalt von Daten gibt: Erstens ist das Bluetooth-Modul im Mobiltelefon aktiviert und das Gerät als sichtbar deklariert. Zweitens begibt der Nutzer sich willentlich in den Sendebereich des Bluetooth-Senders. Die Rechtslage ist hier nicht eindeutig, jedoch werden speziell für den letzten Fall bereits besondere Sendestationen

angeboten (siehe Bild [→ 22, Seite 80]). Allgemein geht man davon aus, dass die Unterschreitung einer Entfernung von 50cm zum Bluetooth-Hotspot als Einwilligung zum Empfang von Informationen gelten kann. Dieses Verfahren wird dann als *Touch-In* bezeichnet.

Die Tatsache, dass im Falle von mobilen Umfragen nicht direkt Werbung für ein bestimmtes Produkt gemacht wird, zählt dagegen nicht. Bereits die Aufforderung seine Meinung zu etwas zu äußern, fällt unter das Gesetz gegen den unlauteren Wettbewerb [Sassenberg und Berger 2007, S. 500].

Daher stellt sich die Frage, ob der Versand von Inhalten mittels Bluetooth, speziell wenn es sich um Werbung handelt, nicht als *Bluespamming* zu bezeichnen ist. Sassenberg und Berger gehen in einem Artikel auf die rechtliche Zulässigkeit speziell von Werbung via Bluetooth ein. Sie untersuchen dabei die Fragestellung, ob Bluetooth-Marketing als Spam angesehen werden kann und kommen zu folgendem Schluss: Ohne vorherige Einwilligung durch den Benutzer stellt das Bluetooth-Marketing eine unlautere Wettbewerbshandlung dar und kann zu Unterlassungs- und Schadenersatzansprüchen führen. Ohne ein Opt-In-Verfahren handelt es sich bereits bei der Anfrage, ob man Daten erhalten möchte, um eine unzumutbare Belästigung. Aktivierte und als sichtbar eingestellte Empfangsgeräte gelten noch nicht als Einwilligung, erst die Verwendung des *Touch-In*-Verfahrens darf als Zustimmung zum Erhalt von Daten gewertet werden. [Sassenberg und Berger 2007]

Ein weiteres Problem betrifft das Speichern von Informationen über Nutzer, da hier das Persönlichkeitsrecht verletzt wird. Damit ein angefragter Nutzer nicht erneut gefragt wird, ob er die Umfrage erhalten möchte, muss festgehalten werden, wer bereits bearbeitet wurde. Dazu werden die MAC-Adressen⁴ der Bluetooth-Empfänger in einer Datenbank gespeichert. Da derartige MAC-Adressen jedoch keine Rückschlüsse auf die zugehörigen Besitzer erlauben, und darüber hinaus recht einfach geändert werden können, scheint keine Verletzung des Persönlichkeitsrechts vorzuliegen. Außerdem kann durch das Opt-In- bzw. Touch-In-Verfahren davon ausgegangen werden, dass der Nutzer bereits seine Zustimmung zur Speicherung einfacher kontextbezogener und nicht-personalisierter Daten gegeben hat.

⁴MAC: Abk. f. Media-Access-Control Adresse: Eineindeutige, 48 bit lange Adresse eines Netzwerkelements, z.B. eines Bluetooth-Bausteins



Abbildung 22.: Bluetooth-Sendestation für den Nahbereich (Quelle: <http://www.pressbox.co.uk/>)

8.2. Fallstudien

Im Folgenden werden drei Fallstudien beschrieben die selbst durchgeführt wurden, um Verteilungsmöglichkeiten zu untersuchen, die erstellte Software zu testen und Daten zur Nutzerakzeptanz zu sammeln. Dazu wird zunächst der Inhalt der Fallstudie und das Vorgehen beschrieben. Anschließend wird das Teilnehmerfeld beschrieben und die erhaltenen Ergebnisse analysiert.

8.2.1. Fallstudie Hobit 2009

In dieser Fallstudie wurde ein Informatik-Quiz mittels Bluetooth an Schüler verteilt. Ziel der Fallstudie war es die Machbarkeit von mobilen Umfragen zu testen. Durchgeführt wurde diese Fallstudie während der Hobit 2009. Hobit steht für „Hochschul- und Berufsinformationstage“ und fand im Frühjahr zum 13. mal im Darmstadtium in Darmstadt statt [Center 2009]. Sie dient als Informationsmesse für Schüler und Schulabsolventen, bei welcher sich die Fachbereiche der verschiedenen Darmstädter Hochschulen präsentieren. Auf der Hobit waren über 70 Messestände einzelner Fachbereiche vertreten, über 12.000 Schüler besuchten die Hobit während der drei Messtage.

Am Stand des Fachbereichs Informatik der h_da wurde ein Informatik-Quiz durchgeführt. Dazu wurde ein JavaME-Prototyp von den Studentinnen Frau Tischler und Frau Oberle entwickelt und präsentiert. Das Quiz war mit der Teilnahme an einem Gewinnspiel verbunden, bei dem es ein Navigationsgerät zu gewinnen gab. Die Anwendung zeigte dazu drei zufällige Fragen mit Informatik-Hintergrund und mögliche Antworten dazu an. Das Gewinnspiel war unabhängig von den Quizfragen - diese konnten auch für sich allein beantwortet werden -, teilnehmen konnte man jedoch ausschließlich über die Quiz-Anwendung.

Die Software war vorher nicht zertifiziert worden, es konnte seitens der Teilnehmer also nicht ausgeschlossen werden, dass die Software ihre Mobiltelefone penetriere und private Daten auslesen könnte. Allerdings war der Rahmen der Veranstaltung seriös und das Auftreten des Standpersonals im Corporate Design der Hochschule vermittelte weitere Seriosität.

Mittels bluetoothfähigen Mobiltelefonen wurde diese Software dann an Schüler gesendet, wo diese das Quiz auf ihren eigenen Geräten beantworten konnten. Falls gewollt, konnten diese dann an der Verlosung teilnehmen. Dazu mussten die Schüler innerhalb der Quiz-Software ihre Adresse eingeben und die Quiz-Antworten mittels SMS oder einer Datenverbindung abschicken. Das jeweilige Standpersonal ermutigte vorbeikommende Schüler am Quiz teilzunehmen.

An den Besuchertagen kamen schätzungsweise 2000 Schüler am Stand des Fachbereichs vorbei, zumeist in kleinen Gruppen von zwei bis vier Personen. Davon blieben ca. 30 Schüler pro Stunde auch am Stand stehen, weil sie sich direkt für den Fachbereich und seine Themen interessierten oder durch das Gewinnspiel neugierig geworden waren. Auf die Dauer der Messe umgerechnet sind das etwa 900 Schüler.

Dabei zeigte sich, dass alle am Quiz teilnehmenden Personen, auch die Möglichkeit zur Teilnahme an der Verlosung nutzen - obwohl die möglichen Kosten für eine SMS explizit genannt wurden. Außerdem konnte ein Gruppen-dynamischer Effekt beobachtet werden: Sobald eine Person einer Gruppe an dem Quiz teilnahm, wollten auch alle anderen Mitglieder der Gruppe teilnehmen. Insgesamt nahmen bei der Verlosung 104 Personen teil, was einem Prozentsatz von 11,5% entspricht, bezogen auf alle am Stand vorbei gekommenen Besucher. Eine Aufteilung zwischen gefundenen Bluetooth-Geräte und Personen die teilgenommen haben, wird nicht gemacht, da aktiv auf die Besucher zugegangen

wurde anstatt durch eine Software potentielle Empfänger zu ermitteln. Darüber hinaus ist es auch nicht möglich Aussagen bezüglich einer möglichen Superdistribution [→ 8.3, Seite 87] zu treffen. Der Zeitraum in dem Quiz-Teilnehmer Daten an den Server gesendet haben, war ausschließlich während der Messezeiten. Von der Möglichkeit der zeitlichen Unabhängigkeit beim Beantworten des Quiz wurde demnach wahrscheinlich kaum Gebrauch gemacht, ebenso wenig von der Ortsunabhängigkeit.

8.2.2. Fallstudie intern am Fachbereich

Diese Fallstudie untersuchte das Annehmen von Anfragen bezüglich einer Bluetooth-Übertragung. Sie wurde durchgeführt am Fachbereich Informatik der Hochschule Darmstadt. In einem ersten Schritt wurde die Bereitschaft zur Annahme eines JavaME-Programms geprüft. Dazu wurde der existierende Prototyp des mobilen Umfrage-Quiz - siehe Fallstudie Hobit [→ 8.2.1, Seite 80] - an die Studenten verteilt. Die Studenten konnten das Quiz durchführen, jedoch keine Antworten absenden. Die Quiz-Anwendung wurde im Vorfeld nicht signiert. Allerdings benötigte diese auch keine besonderen Berechtigungen durch das Betriebssystem des Mobiltelefons, da keine Kommunikationsverbindung aufgebaut werden mußte [→ 2.3, Seite 23]. In einem zweiten Testlauf wurde ein einfacher Text mit Informationen zu einer Fachbereichsveranstaltung an die Studenten versandt.

Als Verteilerstation diente ein umgerüstetes Webkiosk [→ 23, Seite 83]. Dazu wurden zwei Rechner integriert: Nummer Eins diente zur Darstellung von Präsentationen auf dem integrierten Monitor, um Besucher auf das Gerät aufmerksam zu machen und Informationen zum Installationsvorgang der Software bereit zu stellen. Rechner Nummer Zwei wurde mit vier Bluetooth-Dongles ausgerüstet und verteilte die Umfrage-Software und später den Info-Text mittels einer Verteilungs-Software. Die eingesetzte Verteilungs-Software war die *MagicBeamer-Software* des Unternehmens *CheckCom Deutschland GmbH*⁵.

Die Dauer der Fallstudie betrug rund 14 Tage, an denen das Webkiosk im Foyer des Fachbereichs stand. Auf die Anwendung aufmerksam gemacht wurde durch eine Präsentation, die direkt auf dem Bildschirm des Webkiosk ablief und eine kurze Erklärung zum

⁵Unternehmen CheckCom Deutschland GmbH, Eschborn: www.BlueBlitz.com

Vorgehen der Installation und zum Inhalt des Programms zeigte. Außerdem wurde ein kurzer Artikel in der internen Fachbereichszeitung geschrieben, der das Webkiosk und die Hintergründe beschrieb.



Abbildung 23.: Als Bluetooth-Sendestation eingesetztes ehemaliges Webkiosk

Das Teilnehmerfeld bestand hauptsächlich aus Studenten des Fachbereichs Informatik der h_da. Tabelle [→ 1, Seite 83] gibt einen Überblick über die Zusammensetzung des Fachbereichs (genauere Angaben siehe im Anhang [→ A.2, Seite 110]). Dabei sind die potentiellen Teilnehmer zwischen 18 und 30 Jahren und als sehr technikaffin zu beschreiben.

Tabelle 1.: Zusammensetzung des Fachbereichs

	Bachelor	Master	Gesamt
männlich	738	166	904
weiblich	133	20	153
Gesamt	871	186	1057

Die Ergebnisse der Fallstudie sind in Tabelle [→ 2, Seite 84] zu sehen (vollständige Tabelle im Anhang [→ A.1, Seite 106]). Wie viele Studenten im genannten Zeitraum

das Webkiosk passierten ist unbekannt, jedoch kann von maximal ca. 1100 Personen ausgegangen werden: Da alle erfassten Geräte nur ein einziges Mal in der Statistik auftauchen und der Fachbereich aus ca. 1000 Studenten plus Professoren und Mitarbeitern besteht. In dieser Zeit wurden 173 aktivierte Bluetooth-Geräte gefunden, 114 davon Mobiltelefone. Dies entspricht etwa 10% des Fachbereichs. Dabei zeigte sich, dass von den Mobiltelefon-Besitzern mit aktiviertem und sichtbarem Bluetooth-Modul knapp 38% einem eintreffenden Verbindungsversuch zustimmten. Über 62% lehnten den Empfang von Daten vollständig ab, trotz mehrerer fördernder Faktoren: Erstens erregte das verwendete Webkiosk als Werbefläche und Informationsterminal Aufmerksamkeit, da es im Foyer des Fachbereichs untergebracht wurde und nahezu unübersehbar war. Zweitens machte sowohl der Charakter der Präsentation, als auch der Ort die Seriosität der Aktion deutlich. Darüber hinaus wurde die Aktion mit der Bitte um Teilnahme auch in der internen Zeitung des Fachbereich bekannt gegeben [Erbs 2009].

Umgerechnet auf alle Personen führt das zu einer Nutzerakzeptanz von 3,9%.

Tabelle 2.: Angenommene und abgelehnte Verbindungsversuche

Angenommen:	Typ:				Ergebnis
	Mobiltelefon	Notebook	Sonstiges	Unbekannt	
ja	41	3	0	8	52
nein	73	25	1	22	121
Ergebnis	114	28	1	30	173

8.2.3. Fallstudie IT-Forum 2009

Diese Fallstudie betrachtete die Verteilung von Umfragen an Teilnehmer eines IT-Forums. Das IT-Forum fand am 25.06.2009 im Darmstadtium in Darmstadt statt. Hierbei wurden ca. 12 Präsentations-Stände von verschiedenen Unternehmen und Institutionen aufgestellt, an welchen Besucher nähere Informationen zu den entsprechenden Themen erhalten konnten. Für die Umfrage wurde das gleiche, umfunktionierte Gehäuse eines Webkiosk verwendet wie in der Fachbereichs-internen Fallstudie [→ 8.2.2, Seite 82], mit dem Unterschied, dass der darin arbeitende Rechner und die darauf laufende Software ersetzt wurden. Die Versand-Software war eine Entwicklung von Herrn Stephan Räder vom Fachbereich Informatik der h_da.

Als Umfrage wurden zwei verschiedene Fragebögen verwendet, einmal fünf Fragen zum Thema „Berufsbegleitende wissenschaftliche Weiterbildung in der IT“ und einmal vier bis sechs Fragen zum Thema „Bachelor- und Master-Absolventen in Unternehmen“. Die Umfrage-Anwendungen wurden vorher nicht zertifiziert (d.h. die Teilnehmer wurden bei der Installation der Software - nach dem Empfang - vom Betriebssystem des jeweiligen Mobiltelefons auf diese Tatsache aufmerksam gemacht).

Die Umfragedauer betrug von 9:00 Uhr bis 16:00 Uhr und damit 7 Stunden, wobei jeder Fragebogen jeweils 3,5 Stunden verteilt wurde. Das Webkiosk stand während dessen an einer Ecke zwischen mehreren Messeständen. Statt aktiv werbendem Standpersonal (Cross-Media-Maßnahmen, siehe Mobile Marketing [→ 2.1, Seite 12]) wurde sich lediglich auf die Signalwirkung des im Webkiosk untergebrachten Bildschirms und einer darauf laufenden Präsentation verlassen.

Am Forum teilgenommen haben, laut Pressemitteilung der Veranstalter, rund 450 Personen überwiegend aus dem IT-Bereich [Wissenschaftszentrum 2009]. Damit bestand das Teilnehmerfeld hauptsächlich aus technikaffinen Personen im mittleren Alter, wovon schätzungsweise die Hälfte der Teilnehmer (225 Personen) die Präsentations-Stände besucht haben.

Die Umfrage lässt dabei eine Analyse in zweifacher Hinsicht zu: Erstens die Bereitschaft eine Bluetooth-Übertragung zuzulassen und ein unbekanntes Programm zu empfangen. Zweitens die Bereitschaft die Umfrage zu beantworten und die Antworten zurück zu senden.

Tabelle 3.: Ausgelieferte Dateien und die erhaltenen Antwort-Codes

File ID:	Response Code:					Ergebnis
	0	104	111	112	160	
10	11	1	2	6	2	22
11	12	0	4	3	5	24
Ergebnis	23	1	6	9	7	46

Tabelle [→ 3, Seite 85] zeigt die gesammelten Ergebnisse zum Versand von Software mittels Bluetooth. Es zeigt sich, dass 46 Teilnehmer durch das Webkiosk erfasst werden konnten. Bei angenommenen 225 Personen entspricht dies 20%. Trotz schlechter Positionierung und damit geringer Reichweite der Bluetooth-Dongles, ist diese Zahl recht

hoch. So sind in ihr nicht enthalten alle Geräte, die als „unsichtbar“ deklariert sind und natürlich kein Gerät, das Bluetooth gänzlich deaktiviert hat. Bei 23 Teilnehmern wurde die Verbindung ohne Reaktion wieder getrennt (Response-Code 0) - dies passiert, wenn der gefundene Teilnehmer den Empfangsbereich wieder verlässt, noch bevor das System versuchen kann eine Verbindung aufzubauen. Von den restlichen gefundenen Bluetooth-Geräten haben sieben dem Empfang einer Datei zugestimmt (Response-Code 160). Die restlichen verweigerten entweder die Annahmen (Response-Code 111), oder es kam zu sonstigen Übertragung-Fehlern (Code 104 und 112), z.B. bewegte sich die Person aus dem Empfangsbereich. Damit lag die Teilnehmerzahl auf Personen mit aktiviertem und sichtbarem Bluetooth bei 15,2%. Auf alle 225 Personen hochgerechnet ergibt sich somit eine Akzeptanz von 3,1%.

Während für die erste Umfrage (Berufsbegleitende wissenschaftliche Weiterbildung in der IT) keinerlei Rücklauf zu verzeichnen war, zeigt Tabelle [→ 4, Seite 87] die empfangenen Antworten für die zweite Umfrage (Bachelor- und Master-Absolventen in Unternehmen). Bei dieser zweiten Umfrage haben alle Empfänger der Anwendung auch teilgenommen und die Ergebnisse zurück gesandt. Dies entspricht einer Beteiligung von etwa 70% bei den Teilnehmern, die eine Umfrage empfangen haben und lediglich 11% auf alle Personen gerechnet, die vom Bluetooth-Hub erfasst wurden.

Die Antworten der Umfragen sind dabei alle im Zeitraum der Messe eingegangen, keine Antwort wurde später empfangen. Die Möglichkeit der zeitlichen Unabhängigkeit gegenüber papiergebundenen Umfragen scheint also nicht genutzt zu werden. Es erscheint damit auch nicht wahrscheinlich, dass eine Superdistribution [→ 8.3, Seite 87] stattgefunden hat.

Außerdem haben alle Teilnehmer die Antworten als SMS verschickt, niemand hat die Möglichkeit des direkten Versands über eine Datenverbindung genutzt.

Aufgrund der geringen Teilnehmerzahl können die gewonnenen Ergebnisse nicht als repräsentativ angesehen werden.

Ursachen für die schwache Teilnahme und Möglichkeiten genauerer Evaluation werden in den Kapiteln Distributionsvarianten [→ 8.3, Seite 87] und Nutzerakzeptanz [→ 8.4, Seite 93] besprochen.

Tabelle 4.: Ergebnisse der Bachelor-Master-Absolventen Umfragen

ID	F1	F2	F3	F4	F5	F6	F7	F8	Herkunft	IP
24	1			3	1			0	eplus	212.23.97.178
25	1			0	1			0	eplus	212.23.97.178
26	0	2	0		0	1	0		eplus	212.23.97.178
27	0	0	0		1			2	eplus	212.23.97.178
28	1			0	1			0	eplus	212.23.97.178

8.3. Distributionsvarianten

Es werden nun die verschiedenen Möglichkeiten beschrieben, um Software - speziell die Umfrage-Software - an mögliche Nutzer zu verteilen. Dabei werden zuerst verschiedene denkbare Varianten aufgezeigt, anschließend auf die gewählte Variante (Bluetooth) näher eingegangen. Abgeschlossen wird mit der Betrachtung der Superdistribution, also dem viralen Weiterverteilen der Umfrage durch die Empfänger.

Für die Kommunikation der Umfrage-Software sind zunächst zwei Bereiche voneinander zu trennen: Erstens braucht es eine Möglichkeit die Umfragen an die potentiellen Teilnehmer zu verteilen. Zweitens müssen die beantworteten Fragen an den Empfangs-Server zurückgeliefert werden.

Einhergehend mit der Art wie die Umfragesoftware funktioniert, sind verschiedene Varianten mit verschiedenen Vor- und Nachteilen vorstellbar. Die gewählte Realisierung setzt auf einen generischen Ansatz, um Umfragen darzustellen. Dies ermöglicht das reine Verteilen der Umfrage mittels der Beschreibungssprache [→ 5.1, Seite 38] [→ 5.5, Seite 59]. Der Vorteil dieser Variante besteht zum Einen in der geringen Datenmenge die übertragen wird, zum Anderen wird keine erneute Installation der immer gleichen Software (des Frameworks) nötig. Die Beschreibung ist eine reine Textdatei, wodurch eine breite Vielfalt von Verteilungsarten ermöglicht wird: Alle Datenfunk-Varianten unterstützen den Versand von Texten, angefangen bei Bluetooth bis hin zu den WLAN und den normalen Funknetzen GSM und UMTS [→ 2.2, Seite 19]. Darüber hinaus sind auch die etablierten Kommunikationsformen möglich wie SMS oder MMS.

Beim reinen Versand von Umfrage-Beschreibungen ist außerdem ein Abonnementen-System vorstellbar, bei dem registrierte Teilnehmer immer aktuelle Umfragen zugesandt bekommen, z.B. mittels SMS. Der ausschließliche Versand der Umfrage-Beschreibung

generiert allerdings eine Reihe weiterer Probleme, die es zu adressieren gilt: So ist es schwierig zu entscheiden, ob ein Mobiltelefon bereits die notwendige Software besitzt, um aus der empfangenen Beschreibung die Umfrage darzustellen. Es muss in diesem Fall genau mitprotokolliert werden, wer welche Umfrage bereits erhalten hat, wodurch datenschutzrechtliche Probleme auftreten können. Der Nutzer kann nicht mehr sicher sein, anonym an einer Umfrage teilzunehmen. Ein weiteres Problem ist die Schwierigkeit eine Superdistribution zu erreichen, da neu empfangene Umfragen nicht im Java-Archiv der Umfrage-Software enthalten sind (näheres zur Superdistribution siehe weiter unten). Um die Umfragen an andere, neue Nutzer weiter zu verteilen, muss die Client-Software, also die MIDlet-Suite, verteilt werden. Es ist jedoch nicht möglich eine MIDlet-Suite auf einem Mobiltelefon nachträglich zu verändern, also die enthaltene Umfrage-Beschreibung auszutauschen. Zu guter Letzt existiert auch noch das Problem die Client-Software zum ersten Mal an die Nutzer zu verteilen. Es macht keinen Sinn die Umfrage-Beschreibungen zu versenden, wenn die Empfänger noch nicht die notwendige Software besitzen, um diese auch darstellen zu können. Ein weiterer Punkt ist die fehlende Möglichkeit das Framework einfach zu verändern: Sobald verschiedene Versionen des Frameworks existieren - z.B. wegen Sicherheits-Updates oder neuen Funktionen - müssen bereits installierte Versionen des Subsystems Client-Software aktualisiert werden.

Im Hinblick der Unmöglichkeit eine Superdistribution mit immer neu angepassten Umfrage-Beschreibungen zu ermöglichen und die Notwendigkeit, die Umfragen in zwei Schritten zu verteilen (erstens die Client-Software, zweitens die Umfrage-Beschreibung), muss ein anderer Ansatz gewählt werden.

Jede Umfrage die mit dem Designer-Frontend erstellt wird, generiert eine neue, eigenständige Version der Client-Software. Die Client-Software besteht dabei aus der unveränderten MIDlet-Suite und der damit gekoppelten Umfrage-Beschreibung, die sich von Umfrage zu Umfrage ändert. Diese MIDlet-Suite kann dann an die potentiellen Nutzer verteilt werden. Die Vorteile dieses Vorgehens sind die einfache Update-Möglichkeit und das Wegfallen aller Problematiken bezüglich der Informationen, ob Nutzer bereits eine Version des Frameworks nutzen. Darüber hinaus kann die so erstellte Umfrage mittels Superdistribution an andere Mobiltelefone weitergegeben werden, ohne dass Informationen (z.B. über neu hinzu gekommene Umfragen) verloren gehen. Nachteile dieses Ansatzes sind ein möglicherweise unnötiges Datenaufkommen, da jedes mal die gesamte Client-Software (die komplette MIDlet-Suite) an das Mobiltelefon gesendet wird. Darüber hin-

aus könnten auf dem Mobiltelefon bereits existierende Umfrage überschrieben werden. Dies kann ältere, noch nicht beantwortete Umfragen betreffen, die dadurch gelöscht werden. Ein weiterer wesentlicher Nachteil ist die Einschränkung der Verteilungsarten: Um Java-Software übermitteln zu können, muss zwingend eine Datenkommunikation vorhanden sein, d.h. SMS als Übertragungsweg scheidet aus.

Neben der Verteilung der Umfrage-Software müssen die verschiedenen Möglichkeiten des Empfangs von beantworteten Umfragen durch den Empfangs-Server berücksichtigt werden. Dies ist gleichbedeutend mit der Art, wie Antworten durch die Client-Software vom Mobiltelefon versendet werden. Die Antwort ist dabei eine SQL-Anweisung, die alle Informationen bezüglich der gegebenen Antworten enthält. Eine derartige SQL-Anweisung ist ein menschenlesbarer Text und kann auch so behandelt werden. D.h., es ist möglich - wie in der oben beschriebenen Variante des reinen Versands von Umfrage-Beschreibungen - neben einer normalen Datenverbindung auch SMS oder MMS zu verwenden. Im Fall von SMS ist eine zusätzliche Empfänger-Station nötig, welche eintreffende Nachrichten entgegen nimmt und weiterleitet. Die Zieladresse (URL) des Empfangs-Servers muss hierzu der Empfänger-Station bekannt sein (z.B. kann dies durch SMS mit Konfigurationsinformationen geschehen). Diese Zwischenstation muss sich über einen ISP⁶ mit dem Internet verbinden, um die Daten weiterleiten zu können.

Wird stattdessen eine Datenverbindung genutzt, entfällt eine Zwischenstation. Bei dieser Variante ist die genaue Art der Kommunikation für die Client-Software nebensächlich, da sie von der Virtuellen Maschine bzw. vom Betriebssystem des Mobiltelefons geregelt wird. Die Zieladresse des Empfangs-Servers ist der Anwendung aus der Umfrage-Beschreibung bekannt, die Verbindungstechnik ist dabei abhängig vom Mobiltelefon und kann WLAN, GSM/EDGE, UMTS usw. sein. Dabei stellt der Service Provider den Zugang zum Internet zur Verfügung und leitet die Datenpakete an die entsprechenden Stellen weiter. Somit ist es möglich eine Socket- oder Datagrammverbindung zwischen der Client-Komponente und der Server-Komponente herzustellen und beliebige Daten zu übertragen, wenn eine Datenverbindung angefordert und vom Betriebssystem bereit gestellt wird [Miess und Seifert 2007, S. 11ff.].

Daneben ist Bluetooth als Kommunikationstechnik vorstellbar. Viele Mobiltelefone unterstützen diese Übertragungstechnik, da mit ihr viele unterschiedliche Geräte an ein

⁶ISP: Abk. f. Internet Service Provider: Zugangsanbieter zum Internet. Im Falle von Mobilfunk meist die Mobilfunkanbieter.

Mobiltelefon gekoppelt werden können [→ 2.4, Seite 28]. Neben dem Vorteil Location Based Services realisieren zu können, verursacht es darüber hinaus keine Übertragungskosten.

Bluetooth kann dabei sowohl zur Verteilung der Client-Software genutzt werden, als auch eine entsprechende Empfänger-Station bieten, die eintreffende Daten entgegen nehmen und über das Internet an den Empfangs-Server weiterleiten kann.

Um Daten mittels Bluetooth an Empfangsgeräte verteilen zu können, sind zwei Voraussetzungen zu erfüllen: Erstens muss das Empfangsgerät (natürlich) Bluetooth aktiviert haben und sichtbar sein, d.h. es muss durch die Sender such- und auffindbar sein. Zweitens muss der implementierte Bluetooth-Stack des Gerätes das OBEX-Protokoll⁷ unterstützen. Speziell wird der OBEX-File-Transfer benötigt, der ein Pushen von Daten mittels Bluetooth auf empfangswillige Geräte unterstützt.

Um Daten von Umfrage-Teilnehmer entgegen nehmen und weiterleiten zu können, muss der Bluetooth-Sender auch als Hotspot⁸ dienen. Allerdings macht die Verwendung der Bluetooth-Stationen als Empfänger einen Vorteil von mobilen Umfragen zunichte: Die Orts- und Zeitunabhängigkeit der Teilnahme an einer Umfrage geht verloren, da der Nutzer in der Nähe eines Bluetooth-Hotspots verweilen müsste, um die Antworten versenden zu können.

Die möglichen Distributionsvarianten variieren also mit der Art der Implementierung:

- Wird nur die reine Umfragen-Beschreibung verteilt, sind alle Arten der Kommunikation vorstellbar: Entweder werden neue Umfragen über ein Push-Verfahren auf das Gerät geschickt - mittels SMS, MMS oder Bluetooth. Oder durch ein Pull-Verfahren holt sich der Nutzer - oder die Anwendung - neue Umfragen selbstständig ab. Dies kann über alle Datenverbindungen geschehen, z.B. mittels USB-Kabel.
- Wird die Umfrage-Beschreibung mit der Client-Software zusammen ausgeliefert, sind nur Push-Verfahren möglich, die die Übertragung von Binärdaten (wie die MIDlet-Suite) erlauben. Bluetooth ist hier die einzige Möglichkeit, da SMS rein textbasiert ist und MMS das Anhängen von JavaME-Programmen nicht erlaubt. Beim Pull-Verfahren wiederum sind auch wieder alle Möglichkeiten des jeweiligen

⁷OBEX: abk. f. Object Exchange: Ein Protokoll, um zwei Geräten ein Datenübertragung zu ermöglichen.

⁸Hotspot: Öffentliche drahtlose Internetzugriffspunkte

Mobiltelefons vorstellbar, die eine Datenverbindung erlauben. So könnte die Client-Software auf einer Webseite im Internet liegen und von dort herunter geladen werden, ebenso ist eine Installation über ein USB-Kabel möglich.

Da in der gewählten Implementierung die Umfrage-Beschreibung mit der Client-Software gekoppelt wurde, wird in der vorliegenden Implementierung Bluetooth zur Verteilung genutzt. Dabei ist keine automatische Softwarelösung für die Verteilung neuer Umfragen an die Hotspots vorgesehen, vielmehr wird die erstellte Umfrage händisch an den Bluetooth-Server übermittelt und von diesem weiter verteilt.

Die zusätzliche Entwicklung eines Systems zur Verteilung von Software mittels Bluetooth übersteigt den Rahmen dieser Masterarbeit, außerdem existieren bereits funktionierende Programme für diesen Zweck. Aus diesem Grund wird für die Verteilung der Umfragen bzw. der Umfragesoftware auf zwei fremde Lösungen zurück gegriffen. Zum Einen wird die *MagicBeamer-Software* des Unternehmens *CheckCom Deutschland GmbH BlueBlitz Mobile* eingesetzt [→ 8.2.2, Seite 82]. Zum Zweiten wird eine von Herrn Räder am Fachbereich Informatik entwickelte Lösung eingesetzt, die ebenfalls die Verteilung von Inhalten über Bluetooth erlaubt, aber eine bessere Analyse ermöglicht [→ 8.2.3, Seite 84]. Das Bluetooth-Subsystem ist in Abbildung [→ 9, Seite 47] und [→ 10, Seite 48] als „BT-Server“ zu sehen (Unterkapitel [→ 5.4, Seite 46]).

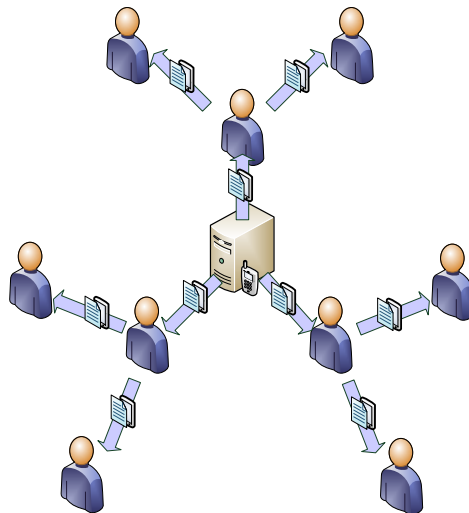


Abbildung 24.: Superdistribution der Umfrage-Anwendung

Neben diesen klassischen Möglichkeiten die Umfragen an Nutzer zu verteilen, ermöglichen mobilen Umfragen eine weitere Distributionsvariante: Die Superdistribution.

Unter Superdistribution versteht man „... an approach to distributing digital products such as software, videos, and recorded music in which the products are made publicly available and distributed in encrypted form instead of being sold in retail outlets or online shops.“[[Wikipedia 2009i](#)].

Die Telekom-Labore verstehen unter der Superdistribution „... ein innovatives Geschäftsmodell, das die Superdistribution ungeschützter und rechtgeschützter digitaler Inhalte ermöglicht. Basierend auf Web 2.0-Technologien können Nutzer selbst erstellte Inhalte über Festnetz, Mobilfunk und zwischen verschiedenen Endgeräten einfach, schnell und legal an andere Nutzer weitergeben und erhalten durch die Community-Plattform Feedback zu ihren Inhalten.“[[Telekom 2008](#)].

Superdistribution beschreibt also die Möglichkeit, dass normale Teilnehmer bzw. Empfänger ihrerseits wieder zu Sendern werden und die Umfragen weiter verteilen [→ [24](#), Seite [91](#)].

Ein anderer Begriff für die Superdistribution ist im Zusammenhang von mobile Marketing das Mobile Viral Marketing. Es bezeichnet laut Pousttchi, Wiedemann und Turowski die digitale Version der Mundwerbung. Dabei werden empfangene Inhalte von den Nutzern an andere Nutzer weitergeleitet [[Pousttchi u. a. 2008](#)]. Allerdings gehen sie auch davon aus, dass der versendete Inhalt sehr stark nutzerspezifisch ist [[Pousttchi und Wiedemann 2007](#)]. Das wiederum sorgt dafür, dass die Verteilung nur in ähnlichen Milieus stattfindet. Was für Werbetreibende somit von Vorteil ist - das er speziell auf eine Nutzerschicht angepasste Inhalte innerhalb dieser Schicht verteilt bekommt - ist in unserem Fall der Mobilen Umfragen eher ein Nachteil: Unser Ziel ist es, ein möglichst breites Spektrum an unterschiedlichen Personengruppen anzusprechen und so bessere Ergebnisse der Umfragen zu erhalten. Wenn wir eine derartige Umfrage jedoch nicht gezielt auf eine bestimmte Gruppe zuschneiden, sinkt die Bereitschaft der Nutzer an solch einer Umfrage teilzunehmen zwangsläufig, weil sie keine für sich angepassten Inhalte erwarten können. Dieser Nachteil ist allerdings durch die generelle Möglichkeit der Superdistribution zu vernachlässigen: Eine Verteilung innerhalb eines Milieus kann mittels Superdistribution stattfinden, zwischen den Milieus wird die Umfrage durch die Hotspots verteilt.

8.4. Nutzerakzeptanz

Dieses Unterkapitel beschreibt die Akzeptanz zur Nutzung mobiler Umfragen. Die Daten auf die sich hierbei gestützt wird, basieren auf den Ergebnissen der Fallstudien [→ 8.2, Seite 80].

Die Ergebnisse der Fallstudien ermöglichen verschiedene Erkenntnisse: Aussagen über die Bereitschaft von Personen unbekannte Daten über Bluetooth zu empfangen und eine Software zu installieren und Aussagen über den Willen, Zeit und Energie in die Beantwortung einer Umfrage zu investieren.

Dabei hat sich gezeigt, dass die Akzeptanz zur Teilnahme an mobilen Umfragen hauptsächlich von zwei Faktoren abhängig ist: Erstens der Zielgruppe und zweitens dem spezifischen Anreiz. Zusätzlich kommen weitere Faktoren hinzu, die allgemein für Mobile Marketing wichtig sind [→ 2.1, Seite 12].

Zu Zielgruppen im Speziellen lässt sich sagen: Je jünger und damit technikaffiner die Personen sind, desto eher akzeptieren sie Inhalte über die Bluetooth-Schnittstelle.

Bei der ersten Fallstudie [→ 8.2.1, Seite 80] auf der Informationsmesse Hobit nahmen rund 100 Personen teil und installierten sich eine Java-Anwendung. Obwohl die Quiz-Software nicht signiert worden war, vertrauten alle Teilnehmer den Beteuerungen des Standpersonals und installierten die Anwendung auf ihrem eigenen Gerät (sofern dies möglich war). Die Akzeptanz lag bei 11,5%. Inwiefern die Quiz-Anwendung durch die Teilnehmer an andere weiterverteilt wurde, lässt sich nicht sagen. Somit ist keine Aussage zur Superdistribution möglich.

In der zweiten Fallstudie [→ 8.2.2, Seite 82] am Fachbereich Informatik wurden neben dem Quiz (ohne Gewinnspiel) auch Zusatz-Informationen zu Veranstaltungen verteilt. Die Akzeptanz lag hier bei 3,7% aller Nutzer und bei ganzen 35% Akzeptanz von Personen mit aktiviertem Bluetooth, obwohl insbesondere die Studenten am Fachbereich Informatik die Gefahren von unbekannter Software kennen. Dass trotzdem 35% den Empfang von Daten erlaubten, kann mit der Menge an Informationen bezüglich der versendeten Inhalte erklärt werden. Die Zahlen decken sich auch mit einer Aussage von Bauer, Neumann und Reichardt, wonach die Einstellung zum Mobile Marketing höher

ist, je mehr Sachverstand die Personen zu diesem Thema haben [→ 2.1, Seite 12][Bauer u. a. 2008].

Die dritte Fallstudie [→ 8.2.3, Seite 84] hatte eine Benutzerakzeptanz von 3,1% auf alle Teilnehmer gerechnet. Davon haben aber wiederum 70% die Fragen beantwortet und die Daten abgeschickt. Zu beachten ist, dass die Zahlen auf nur wenigen Teilnehmern beruhen und nicht als repräsentativ angesehen werden können.

Der erstaunlich hohe Prozentsatz zur Akzeptanz bei der ersten Fallstudie kann, wie in [→ 8.2.1, Seite 80] zum Teil bereits angesprochen, an mehreren Faktoren liegen:

- Der offizielle Rahmen der Veranstaltung.
- Das seriöse Auftreten im Corporate Design der Hochschule Darmstadt durch das Standpersonal.
- Die Teilnahme an einem Gewinnspiel.
- Der alltägliche Umgang mit Mobiltelefonen durch das junge Alter der Besucher.
- Das Unwissen der Teilnehmer über mögliche Gefahren unbekannter Software.
- Die Besucher waren interessiert an repräsentativen Inhalten, die die verschiedenen Studiengänge vermitteln.

Es lässt sich hieraus schließen, dass junge Personen die Installation unbekannter Anwendungen durchführen, wenn die Anreize dies zu tun entsprechend vorhanden sind. Anreize können z.B. mögliche Gewinne bei der Teilnahme sein, oder das aktive Werben durch Standpersonal. Zur Erhöhung der Anreize können außerdem Cross-Media-Maßnahmen genutzt werden [→ 2.1, Seite 12].

Darüber hinaus sind die Standorte der Bluetooth-Sendestationen wichtig: Je mehr Aufmerksamkeit eine Sendestation erregt und je höher der persönliche Nutzen für den einzelnen ist, desto mehr Personen nutzen das Angebot.

Außerdem muss der Kunde rechtzeitig und vorher auf den Bluetooth-Stand aufmerksam gemacht werden. Denn nicht alle haben Bluetooth ständig aktiviert, weil es Energie kostet und nicht unbedingt benötigt wird. So wird es standardmäßig nur zur Kopplung mit anderen Geräte, z.B. Headsets oder als Modems für Notebooks verwendet. In diesem Fall ist es jedoch für den Nutzer ausreichend und sogar vorteilhaft, das eigene

Mobiltelefon als „versteckt“ zu deklarieren. Das ermöglicht ihm selbst die Nutzung der Bluetooth-Funktionen, ohne von anderen Sendern - wie den Bluetooth-Hotspots - entdeckt werden zu können (wobei eine Entdeckung prinzipielle nicht auszuschließen ist [Zoller 2007]).

Zu guter Letzt muss der Standort eines Bluetooth-Hotspots zum Verweilen einladen: Erstens kann es bis zum Eintreffen einer Benachrichtigung, dass Inhalte bereit stehen einige Sekunden dauern, zweitens kann die Übertragung der Inhalte selbst einige Sekunden beanspruchen.

Insgesamt erscheint die Akzeptanz von Mobile Marketing und speziell von mobilen Umfragen verglichen mit herkömmlichen Werbemaßnahmen eher gering, besonders im Hinblick darauf, dass weitere Verluste entstehen, wenn die Personen auch Umfragen beantworten sollen. So steht eine Akzeptanzrate von ca. 4% für das reine Empfangen von Bluetooth-Inhalten einer Rate zwischen 24% und 44% für herkömmliche Werbung gegenüber. Die Tabelle [→ 5, Seite 95]) listet die verschiedenen Akzeptanzwerte zwischen den Werbearten auf.

Tabelle 5.: Nutzerakzeptanz verschiedener Werbekonzepte (Quelle: [Zettel 2007])

Art	Akzeptanz
Zeitungswerbung	44%
Zeitschriftenwerbung	41%
Fernsehwerbung	35%
Plakatwerbung	32%
Radiowerbung	24%
Online-Werbung	10%

Dennoch brauchen diese Zahlen nicht abzuschrecken: Zwar sind 4% Nutzerakzeptanz gegenüber herkömmlichen Werbemaßnahmen sehr klein, aber die Zahl von aktivierten bluetoothfähigen Geräten wird weiter zunehmen, ebenso wie der Bekanntheitsgrad von Bluetooth und seine Verwendung wachsen wird [→ 2.4, Seite 28].

Ein weiterer Punkt ist die kostengünstige Möglichkeit nahezu beliebig viele Leute zu erreichen. So wurde z.B. im März 2007 eine Fremdstudie durchgeführt, bei der innerhalb von drei Wochen 26.000 Personen mit aktivierten und sichtbaren Bluetooth-Geräten von den Bluetooth-Sendestationen gefunden wurden. Von den aufgeforderten Nutzern

eine unbekannte Datei zu empfangen, nahmen etwa 1200 Personen diese Offerte an [Kremp 2007]. Es zeigte sich hier eine Akzeptanzrate von 4,6% bezogen auf Personen mit aktiviertem und sichtbarem Bluetooth. Die tatsächliche Anzahl an Personen die diese Plakate passiert haben ist unbekannt, weswegen die Nutzerakzeptanz noch weit geringer sein muss, als die 4% der eigenen Fallstudien. Dennoch ist eine reale Zahl von 1200 Personen beeindruckend: Wenn man bedenkt, dass davon wiederum vielleicht 70% (wie in Fallstudie 3) eine Umfrage beantworten würden, wären dies 840 Personen - ohne dass Personal eingesetzt worden wäre!

Außerdem gelten Mobiltelefone als sehr persönliche Kommunikationsgegenstände und sind sehr stark an den Besitzer gebunden [→ 2.2, Seite 19]. Das bedeutet aber, sobald eine Kommunikation zwischen dem Bluetooth-Hotspot und dem Kunden zustande kommt, ist die Kontaktqualität sehr hoch. Darüber hinaus werden Streuverluste extrem minimiert.

Anhand der Fallstudien ließ sich ein Punkt feststellen: Es herrscht zum Teil ein naiver Umgang mit via Bluetooth versandten Daten. Gerade bei der ersten Fallstudie auf der Hobit stand die Vertrauenswürdigkeit der Software nicht im Vordergrund. Angesichts der Gefahren durch Viren und Datendiebstahl ist diese Tatsache bedenklich [Rogge 2006].

Daneben ließ sich ein weiterer Punkt beobachten: Die Inhalte der zu versendenden Daten sind am Anfang scheinbar nebensächlich. So erhält der Nutzer am Anfang prinzipiell nur die Information bzw. Frage, ob er Daten von einem anderen Gerät empfangen möchte. Die Art der Daten bzw. der Datentyp wird erst nach dem Empfang vom Betriebssystem analysiert. Es wird also bei keinem mit Bluetooth gesendeten Objekt vorher klar, um was es sich genau handelt, der Nutzer kann nicht zwischen Bildern, Musik und Spielen unterscheiden.

8.5. Zusammenfassung und Konsequenzen der Distributionsvarianten und Nutzerakzeptanz

Gegenüber herkömmlichen, papiergebundenen Umfragen bieten mobile Umfragen eine Orts- und Zeitunabhängigkeit beim Beantworten der Fragen. Von dieser Unabhängigkeit wurde jedoch kaum Gebrauch gemacht. Dies kann allerdings damit zusammen hängen, dass die Nutzer die Einschränkungen normaler Umfragen gewohnt sind und automatisch annehmen, diese gelten auch für mobile Umfragen. Dieser Punkt sollte dahingehend untersucht werden, ob die Zeitunabhängigkeit mehr genutzt wird, wenn die Leute besser informiert sind. Z.B. kann gezielt auf die Vorteile der Unabhängigkeit beim Beantworten hingewiesen werden. Ergibt sich dann eine größere zeitliche Spreizung für den Rücklauf von Antworten, war den Leuten diese Möglichkeit wahrscheinlich vorher nicht bewusst.

Bezüglich einer Superdistribution lassen sich nur eingeschränkt Aussagen treffen: Sie erscheint nicht sehr wahrscheinlich, aufgrund des Zeitraums in dem Antworten eintrafen. Um hierzu Aussagen treffen zu können, ist folgendes Vorgehen denkbar: Erstens die Leute explizit auffordern, die Anwendung an andere weiter zu verteilen. Dann müssen die empfangenen Nachrichten eindeutig gekennzeichnet werden, z.B. könnte die MAC-Adresse des jeweiligen Sende-Gerätes mit übertragen werden. Da die MAC-Adressen der ursprünglichen Empfangs-Geräte vom Bluetooth-Hub gespeichert werden, kann durch einen anschließende Vergleich dieser Datensätze auf die Größe der Superdistribution geschlossen werden (zu Datenschutzbestimmungen siehe Kapitel [→ 8.1, Seite 76]).

Um genug Aufmerksamkeit für Bluetooth-Umfragen zu erregen, empfehlen sich Cross-Media-Maßnahmen, da Push-Verfahren als unlauterer Wettbewerb zählen und nur nach Unterschreitung einer bestimmten Distanz erfolgen dürfen. Diese Maßnahmen können auch durch Personal geschehen, das aktiv die Umfragen bewirbt. So wäre auch eine Hilfestellung bei eventuellen Problemen der Kunden möglich.

Die Fallstudien haben gezeigt, dass nur etwa 3-4% aller Personen sowohl Bluetooth aktiviert haben, als auch darüber offerierte Inhalte akzeptieren. Andere Studien ge-

hen sogar von nur 0,4% aus, bezogen auf Personen mit aktiviertem Bluetooth [Kremp 2007].

Etwa 70% dieser Personen nehmen dann auch an einer Umfrage teil. Die Zahlen basieren auf der Fallstudie „IT-Forum“ [→ 8.2.3, Seite 84], die nur eine geringe Teilnehmerzahl hatte. Deswegen sollten diese in einer längeren Versuchsreihe auf anderen Messen und mit anderen Personengruppen wiederholt werden. Ausgehend von diesen 70% Teilnehmern bezogen auf die Gesamtmenge, kann recht schnell - einige, wenige Wochen - eine statistisch relevante Anzahl an Antworten gewonnen werden, um repräsentative Umfrageergebnisse zu erhalten.

8.6. Ethische Betrachtung

Neben der Beschaffung von statistischen Daten und Meinungsumfragen durch bestimmte Ämter, dienen Umfragen im Allgemeinen der Marktforschung.

Die Informationen kommen in erster Linie den erhebenden Unternehmen zu Gute, den befragten Personen entsteht dabei kein unmittelbarer Nutzen. Langfristig gesehen können Umfragen bzw. deren Ergebnisse jedoch auch positive Auswirkungen für die Kunden haben, z.B. durch die Verbesserung von Produkten aufgrund des erhaltenen Feedbacks. Die Kundenbewertungen von Artikeln innerhalb vieler Online-Versandhäuser sind ein Beispiel für Informationen durch Kunden und stellen Meinungsäußerungen dieser dar.

Ist eine Person jedoch nicht gewillt an einer Umfrage teilzunehmen, so ist die Art der Umfrage nebensächlich. Die Verwendung von mobilen Umfragen ändert an diesem Sachverhalt wenig. Allerdings können diese eine Erleichterung gegenüber herkömmlichen Umfragen bieten, z.B. durch eine automatische Benutzerführung bei sich verzweigenden Fragen. Wenn Nutzer sogar gezwungen sind bestimmte Befragungen über sich ergehen zu lassen - z.B. den Mikrozensus⁹ des Statistischen Bundesamtes - so ermöglichen Mobile Umfragen eine zeit- und ortsunabhängige Beantwortung.

Entscheidend ist also der Willen der Person, Umfragen an sich sind weder gut noch verwerflich. Wichtig ist, wie Personen zur Teilnahme an Umfragen bewegt werden können: Werden ihnen, ohne vorherige Einwilligung, mittels Push-Verfahren Aufforderungen zur Teilnahme geschickt, so ist dies ärgerlich. Die Verwendung von Bluetooth ohne Opt-In- oder Touch-In-Verfahren ist sogar unzumutbar, da Mobiltelefone als sehr persönliche Kommunikationsgegenstände zählen. Derartiger Versand von mobilen Umfragen, bzw. die Aufforderung an einer teilzunehmen, muss als Spam angesehen werden und ist in Deutschland verboten.

Ein weiterer Punkt der kritisch betrachtet werden muss, ist die allgemeine Verwendung von Mobile Marketing: Nutzer werden durch ständige Werbebotschaften über Bluetooth desensibilisiert. Neben überall zu findenden Hotspots, die kostenlose Bilder, Spiele und Musik anpreisen, können gefährliche Inhalte wie Viren eingeschmuggelt werden [Rogge

⁹Mikrozensus: <http://www.destatis.de/jetspeed/portal/cms/Sites/destatis/Internet/DE/Presse/abisz/Mikrozensus>

2006]. Die Nutzer können dabei nicht zwischen gefährlichen und ungefährlichen Inhalten unterscheiden, ein gesundes Misstrauen gegenüber unbekanntem Anbietern existiert hier noch nicht.

Bei der Entwicklung von Software sollten sich Softwareentwickler deshalb an bestimmte ethische Regeln halten. Im „Software Engineering Code“ werden acht Punkte genannt, die beachtet werden sollten [ACM/IEEE-CS 1999]:

Öffentlichkeit Der Entwickler sollte immer im öffentlichen Interesse agieren.

Kunde und Arbeitgeber Der Entwickler sollte so handeln, dass - vereinbart mit öffentlichen Interessen - das Interesse des Kunden und Arbeitgebers gewahrt wird.

Produkt Die Anwendung soll den höchsten vorhandenen Standards entsprechen.

Beurteilung Der Entwickler soll integer und unabhängig professionelle Entscheidungen treffen.

Management Manager sollen ethische Grundsätze achten und wahren.

Beruf Ansehen und Unversehrtheit des eigenen Berufs soll gewährleistet sein.

Kollegen Fairness gegenüber Kollegen.

Selbst Lebenslange Fortbildung und Weitergeben der ethischen Grundsätze.

9. Zusammenfassung und Ausblick

9.1. Zusammenfassung

Die vorliegende Arbeit beschäftigt sich mit zwei Themen: Erstens der Erstellung eines generischen Frameworks, um Umfragen mit Mobiltelefonen durchführen zu können, zweitens der Betrachtung von Distributionsvarianten und der Nutzerakzeptanz zu mobilen Umfragen.

Ersteres wurde erreicht, indem das Framework in Subsysteme aufgeteilt wurde: Ein Designer-Frontend mit Netbeans als Unterbau, das es erlaubt Umfragen zu erstellen. Eine Baumstruktur stellt die Fragen samt zugehöriger Antworten dar und ermöglicht eine Verzweigung von Fragen. Drag&Drop sorgen für das einfache Herstellen und Ändern von Umfragen. Auf Knopfdruck kann jederzeit die Umfrage-Anwendung - eine MIDlet-Suite - für Mobiltelefone erstellt werden, ebenso wie die zugehörige SQL-Anweisung, um eine Datenbank entsprechend der Umfrage vorzubereiten. Diese MIDlet-Suite - als Client-Software bezeichnet - stellt die Umfragen auf einem Mobiltelefon dar und erlaubt das Auswählen und Zwischenspeichern von Antworten zu den einzelnen Fragen. Außerdem unterstützt es die Übertragung der Antworten über eine direkte Datenverbindung, oder als SMS. Die Antworten werden dabei in eine SQL-Anweisung codiert, die auf der Empfänger-Datenbank ausgeführt wird. Weiterhin wurde ein Empfangs-Server erstellt, um diese SQL-Anweisungen entgegen nehmen und auf der Ziel-Datenbank ausführen zu können. Dazu wurde dieses Subsystem als Java-Servlet entworfen, dass in einem beliebigen Servlet-Container (z.B. Apache Tomcat) lauffähig ist. Das vierte Subsystem ist ein SMS-Hub der auf einem bestimmten Port eintreffenden SMS entgegen nimmt und seinerseits eine Datenverbindung zum Empfangs-Server aufbaut. Der Inhalt einer empfangenen SMS ist dabei der gleiche wie bei einer direkten Datenverbindung und besteht

aus einer SQL-Anweisung. Dieses Vorgehen wurde als Alternative zur direkten Rücklieferung der Umfrageantworten implementiert, da nicht alle teilnehmenden Mobiltelefone in der Lage sind, eine Internetverbindung aufzubauen. SMS hingegen wird von allen Geräten unterstützt. Um die Generik des Frameworks zu erreichen, wurde eine Beschreibungssprache für Umfragen entwickelt, die durch die Client-Software analysiert wird, um entsprechende Umfragen darzustellen. Sie ermöglicht auch die Verzweigung von Fragen. Die Beschreibungssprache ist dabei XML, das Gesamtsystem selbst wurde in Java und seinen Derivaten (JavaME) erstellt.

Gegenüber papiergebundenen Umfragen bieten mobile Umfragen mehrere Vorteile: Die Vermeidung des Medienbruchs, die Möglichkeit von Verzweigungen der Fragen, Orts- und Zeitunabhängigkeit, sowie die Option, Rückantworten an die Teilnehmer geben zu können.

Gegenüber anderen mobilen Umfragen hat der gewählte Ansatz zwei Vorteile: Erstens ist es nicht nötig während der Teilnahme an einer Umfrage ständig online zu sein. Zweitens wird für den Versand der Antworten nicht für jede Antwort eine einzelne SMS benötigt, stattdessen werden alle Antworten einer Umfrage gesammelt und in möglichst wenigen SMS verschickt (im Normalfall ist eine SMS ausreichend).

Durch die Verwendung eines selbstständigen JavaME-Programms wird außerdem eine Superdistribution von Umfragen ermöglicht.

Die möglichen Distributionsvarianten variieren mit der Art der Implementierung: Beim reinen Versand der Umfragen-Beschreibung sind alle Kommunikationsarten der Mobiltelefone vorstellbar, z.B. SMS, MMS und Bluetooth zum Pushen. Beim Pullen neuer Umfragen dagegen WLAN, UMTS usw.. Hierfür wird aber eine bereits installierte Client-Software benötigt. Aufgrund der Schwierigkeit zu entscheiden, ob die Client-Software schon vorhanden ist, wird diese mit der Umfrage-Beschreibung in einem JAR-Archiv verbunden und ausgeliefert. Bei dieser (gewählten) Implementierung kommt zum Pushen nur Bluetooth in Frage, beim Pullen existieren hingegen die gleichen Möglichkeiten wie beim reinen Versand der Umfrage-Beschreibung. Die deshalb implementierte Variante setzt auf die Verwendung von Bluetooth. Als Bluetooth-Sender (hier auch BT-Server genannt) wurden zwei verschiedene externe Lösungen benutzt. Die Client-Software wurde dabei immer händisch auf die entsprechenden BT-Server aufgespielt.

Die Nutzerakzeptanz, zum Erhalt unbekannter Daten mittels Bluetooth und zur Teilnahme an mobilen Umfragen im Speziellen, wurde durch drei Fallstudien untersucht. Dabei stellte sich heraus, dass etwa 4% aller Personen Inhalte akzeptieren und annehmen, die per Bluetooth angeboten werden. Andere Studien gehen sogar von noch geringeren Zahlen aus (nur 0,4% der Personen mit aktiviertem Bluetooth nehmen die Inhalte an). Gegenüber anderen Werbemaßnahmen sind diese Zahlen gering, wobei allerdings die Qualität der Zustande gekommenen Kommunikation zwischen Anbieter und Kunde viel höher als bei klassischen Varianten ist. Dies liegt daran, dass Mobiltelefone als sehr persönliche Gegenstände gelten. Von allen Personen, die erfolgreich mit Bluetooth angesprochen werden konnten, sind etwa 70% dann auch zur Teilnahme an den mobilen Umfragen bereit gewesen, wobei die Teilnehmerzahl zu gering war, um repräsentative Aussagen treffen zu können. Dafür sind weitere Untersuchungen notwendig.

9.2. Fazit

Es wurde gezeigt, dass mobile Umfragen und der hierfür gewählte Ansatz funktionieren können. Mobile Umfragen bieten einige Vorteile gegenüber herkömmlichen Umfragen, ebenso wie der generische Ansatz des Frameworks Vorteile gegenüber ähnlichen Ansätzen bietet. Als Distributionsvariante kommt nur Bluetooth in Frage.

Das gesamte Konzept besitzt jedoch auch bestimmte Nachteile: Bei der Verwendung von Bluetooth sind zusätzliche Cross-Media-Maßnahmen nötig, um Kunden auf das Vorhandensein von Umfragen aufmerksam zu machen. Es stellt sich hierbei die Frage, ob z.B. Hostessen, die eine Umfrage anpreisen, diese nicht auch gleich selbst durchführen können. Eine Verteilung von Umfragen an Personen außerhalb eines Umkreises von 50cm um den Sender ist nicht erlaubt und gilt als Spam. Darüber hinaus steigt die Gefahr der Verbreitung von Viren, je mehr Hotspots in der Öffentlichkeit eingesetzt werden. Da die Übertragung einer Umfrage außerdem einige Sekunden Zeit benötigt und vorher empfangsbereite Geräte noch gefunden werden müssen, empfehlen sich nur Standorte für Bluetooth-Hotspots an denen Menschen verweilen, z.B. Cafés oder Haltestellen. In den Fallstudien konnte darüber hinaus kaum die Nutzung der Ortsunabhängigkeit und Zeitunabhängigkeit beobachtet werden. Gleiches gilt für eine Superdistribution.

9.3. Ausblick

Ein entscheidender Vorteil wäre die Möglichkeit, direkt die komplette Umfrage-Anwendung an Kunden - auf deren Wunsch hin - schicken zu können. Der Versand von Programmen durch MMS-ähnliche Techniken ist jedoch nicht in Planung. So kann auch in Zukunft nur Nahbereichs-Funk zum Pushen verwendet werden, oder interessierte Nutzer müssen sich die Umfragen eigenhändig herunterladen.

Wahrscheinlich werden Mobiltelefone später immer öfter, vielleicht ständig, online im Internet sein, womit eine eigenständige Software für Umfragen unnötig werden würde: Umfragen könnten direkt im Browser auf speziellen Webseiten durchgeführt werden. Bereits heute zeichnet sich der Trend ab, immer mehr Anwendungen direkt im Browser laufen zu lassen (Stichwort Cloud-Computing, z.B. Google Apps).

Angesichts dessen erscheint es fraglich, inwiefern die erstellte Software eine Zukunft hat. Aus diesem Grund wird die Entwicklung der Software vorerst nicht weiter verfolgt. Um eine weitere Nutzung durch Dritte zu ermöglichen, wird der gesamte Sourcecode unter der GPLv3¹ veröffentlicht. Zu finden ist das Projekt mit allen zugehörigen Dateien bei SourceForge.net² als Mobile Questionnaire <https://sourceforge.net/projects/mobilequestion/>. Er kann somit weiterverwendet und beliebig modifiziert werden.

¹GNU General Public License Version 3, <http://www.gnu.de/documents/gpl.de.html>

²SourceForge: Opensource software directory, <http://sourceforge.net/>

A. Anhang

Tabelle 6.: Weltweite Marktanteile von Mobiltelefonen (Quelle: Gartner)

Anbieter	Marktanteil	
	Q1 2009	Q1 2008
Nokia	36,20%	39,10%
Samsung	19,10%	14,40%
LG	9,90%	8,00%
Motorola	6,20%	10,20%
Sony Ericsson	5,40%	7,50%
Andere	23,40%	20,80%

Tabelle 7.: Weltweite Marktanteile von Smartphones (Quelle: Gartner)

Anbieter	Marktanteil	
	Q1 2009	Q1 2008
Nokia	41,20%	45,10%
RIM	19,90%	13,30%
Apple	10,80%	5,30%
HTC	5,40%	4,00%
Fujitsu	3,80%	4,10%
Andere	18,80%	28,10%

A. Anhang

Florian Miess

Auswertung Fallstudie FBI

28.07.2009

<u>MAC</u>	<u>Typ</u>	<u>Angenommen</u>	<u>Hersteller</u>
0001e33f3687	Handy	ja	Siemens
00027875293c	Notebook	nein	
00027875293d	Handy	nein	
0002787660a6	Unbekannt	nein	Samsung Electronics
0002c7e4dfdc	Unbekannt	nein	ALPs Electronic
00031901568c	Unbekannt	nein	Infineon
00037a752079	Unbekannt	nein	Taiyo Yuden
00037aeb8211	Notebook	ja	Taiyo Yuden
0005c94ecfdb	Handy	nein	
0005c96be8da	Handy	ja	
000a2819497d	Handy	nein	Motorola
000a286ba7e6	Handy	nein	
000b0d30d04a	Unbekannt	ja	Air2U
000d411ca145	Unbekannt	ja	Siemens
000df05bdd9f	Notebook	nein	
000e07d0a2e3	Notebook	nein	
000e6d963dbc	Handy	ja	
000eed4d6e84	Handy	ja	Nokia
000fde2550a0	Handy	nein	
000fde882ee3	Handy	nein	
0010c652aaab	Notebook	nein	
001237f0cd38	Unbekannt	nein	Texas Instruments
001237f3f9cb	Handy	nein	Texas Instruments
00124724ade3	Unbekannt	nein	Samsung Electronics
00124744eebf	Handy	nein	Samsung Electronics
001247c9ef7d	Handy	nein	Samsung Electronics
001262df139a	Handy	nein	Nokia
0012d10b07f9	Handy	ja	
0012d1ac6f98	Unbekannt	nein	Texas Instruments
0012d2a37a9d	Unbekannt	nein	Texas Instruments
0012ee886ffe	Handy	nein	Sony Ericsson
001370024614	Handy	nein	Nokia
0014a75f90b1	Handy	nein	Nokia
00158313e4b4	Unbekannt	ja	IVT corporation
00158313e6ba	Unbekannt	ja	IVT corporation
00158313e729	Unbekannt	ja	IVT corporation
0015b918f958	Unbekannt	nein	Samsung Electronics
0015b99c4639	Handy	nein	Samsung Electronics
0015b9c22334	Handy	nein	Samsung Electronics
0016205b401d	Handy	ja	Sony Ericsson
001620c83c4a	Handy	nein	Sony Ericsson
0016db2caf75	Unbekannt	nein	Samsung Electronics
0016fe981e48	Notebook	nein	
0016fe963c??	Unbekannt	ja	Alps Electronic
00174b019a6c	Handy	nein	Nokia
00174b8d????	Handy	ja	Nokia
0017e5eef079	Handy	nein	Texas Instruments
0017e5fc71e5	Unbekannt	nein	Texas Instruments
0017e828d7fe	Handy	nein	Texas Instruments
0017e8823b48	Unbekannt	nein	Texas Instruments
0017e8a7896b	Unbekannt	nein	Texas Instruments
0018130bb3ed	Handy	ja	
0018130fe2bd	Handy	nein	
001813c6360e	Handy	nein	
0018afccee3f	Handy	nein	

A. Anhang

Florian Miess			Auswertung Fallstudie FBI	28.07.2009
0018c51c9d67	Handy	nein		
0018c5e780bd	Handy	nein	Nokia	
0018e41d604e	Unbekannt	nein	Yiguang	
00191d949300	Sonstiges	nein	Nintendo	
0019630f234e	Handy	ja		
001963925b1f	Handy	nein	Sony Ericsson	
0019639c6f73	Handy	nein	Sony Ericsson	
001963d5a684	Handy	nein	Sony Ericsson	
001963fed27e	Handy	nein	Sony Ericsson	
00197de37aa7	Notebook	nein	Hon Hai	
0019c18c2285	Notebook	nein		
001a1630d96b	Handy	ja		
001a75157901	Handy	ja		
001a89bdaeea	Handy	ja	Nokia	
001a8a973404	Handy	ja		
001b5917fc10	Handy	ja	Sony Ericsson	
001baf001e3b	Handy	nein	Nokia	
001bee7f862b	Handy	nein	Nokia	
001beebdf33e	Handy	nein	Nokia	
001c356c40ca	Handy	ja		
001c356c444f	Handy	nein		
001c356e295e	Handy	nein	Nokia	
001c9a2978bf	Handy	nein	Nokia	
001ca4c07563	Handy	nein		
001ch32aeaaa	Handy	nein		
001cd41c958f	Handy	nein		
001cd614a75e	Handy	nein		
001d2821187f	Handy	ja	Sony Ericsson	
001d285415e4	Handy	nein	Sony Ericsson	
001d28776273	Handy	ja	Sony Ericsson	
001d28837e25	Handy	ja	Sony Ericsson	
001d3bb3199f	Handy	nein		
001d6eba1dbc	Handy	nein		
001d6ebb83a1	Handy	nein	Nokia	
001d985cf111	Handy	ja	Nokia	
001dbee9b39b	Handy	nein	Motorola	
001dd9f61f9b	Notebook	nein	Hon Hai	
001dd9f62c68	Notebook	nein	Hon Hai	
001dd9f62eba	Notebook	nein	Hon Hai	
001dd9f632e7	Notebook	nein	Hon Hai	
001dd9f634da	Notebook	nein	Hon Hai	
001de90e4e43	Handy	nein	Nokia	
001de97f7c17	Handy	nein	Nokia	
001df6ab610d	Handy	nein		
001df6c0b019	Unbekannt	nein	Samsung Electronics	
001dfd724f41	Handy	ja		
001e37b8818a	Notebook	nein		
001e3a29f153	Handy	ja	Nokia	
001e3b3101c3	Handy	nein	Nokia	
001e3d8893b3	Unbekannt	nein	Alps Electronic	
001e45665767	Handy	nein	Sony Ericsson	
001e45bf799e	Handy	ja	Sony Ericsson	
001e4cd75b46	Notebook	nein	Hon Hai	
001e52d775a8	Unbekannt	nein	Apple	
001e7d8318c8	Handy	nein		
001e8c045eb9	Notebook	nein		

A. Anhang

Florian Miess

Auswertung Fallstudie FBI

28.07.2009

001edc009a91	Handy	ja	
001edc72b597	Handy	nein	
001edc7bec1a	Handy	ja	
001ee29e5f57	Unbekannt	ja	Samsung Electronics
001f00573d40	Handy	ja	
001f00b9182a	Handy	nein	Nokia
001f5bd2dfa3	Unbekannt	nein	Apple
001fde36bb28	Handy	nein	Nokia
001fdf7390e9	Handy	nein	
001fe1fe3b27	Notebook	ja	
001fe3701698	Handy	nein	
001fe4128dd6	Handy	ja	Sony Ericsson
001fe4182fc3	Handy	ja	
001fe4e61eaa	Handy	ja	
001fe4f8eada	Handy	ja	Sony Ericsson
001fe4fb2732	Handy	ja	
0020751324e3	Handy	nein	Motorola
002108338fa4	Handy	nein	Nokia
0021864afbb0	Notebook	ja	
0021d2970781	Unbekannt	nein	Samsung Electronics
0021e98cc578	Handy	nein	
0021fbe6a7ba	Handy	nein	LG Electronics
0021fced3855	Handy	nein	Nokia
0021fe02be28	Handy	nein	Nokia
0021fead1e29	Handy	nein	Nokia
0021fedb73f7	Handy	ja	Nokia
0021fed14228	Handy	nein	Nokia
0021fef9d732	Handy	ja	Nokia
002243a13178	Handy	nein	
002243b7e5eb	Unbekannt	nein	AzureWave Technologies
002243bcea73	Unbekannt	nein	AzureWave Technologies
002243bd10b1	Notebook	nein	AzureWave Technologies
00225f209483	Notebook	nein	Liteon Technology
00225f4bfa59	Notebook	nein	Liteon Technology
0022658d85f2	Handy	nein	Nokia
002268e93795	Notebook	nein	
002269bf7cd5	Notebook	nein	
002269e25d82	Notebook	nein	
002269f3286c	Notebook	nein	Hon Hai
002269f752d8	Notebook	nein	
002298200aef	Handy	nein	
002298d06b06	Handy	nein	Sony Ericsson
0022fc4e21df	Handy	ja	Nokia
0022fc7baaf8	Handy	nein	Nokia
0022fcee178b	Handy	nein	Nokia
0022fd0ae7a0	Handy	nein	Nokia
0022fd9e0623	Handy	ja	Nokia
0022fdb71bbc	Handy	nein	Nokia
00233a4002df	Unbekannt	ja	Samsung Electronics
00234df11edd	Notebook	nein	
00234ee3d319	Handy	nein	Alcatel-Lucent
00234ef9e20b	Notebook	nein	
0023d632c75a	Handy	ja	
0023f104827d	Handy	ja	
0023f128f104	Handy	ja	
002404ab7376	Handy	ja	Nokia

A. Anhang

Florian Miess

Auswertung Fallstudie FBI

28.07.2009

00247d122021	Handy	nein	Nokia
00247e4b4527	Unbekannt	nein	USI
002490c75dac	Handy	ja	
0024ef161293	Handy	nein	
0024ef190c98	Handy	ja	
0025473cee80	Handy	ja	Nokia

Filter

Anzahl - Angenommen	Typ				Gesamt Ergebnis
Angenommen	Handy	Notebook	Sonstiges	Unbekannt	
ja	41		3	8	52
nein	73		25	1	121
Gesamt Ergebnis	114		28	1	30

A. Anhang



Semesterstatistik Sommersemester 2009

Stand 14.5.2009

Fachbereich	I			
	Informatik	Informatik	Kooperativer Studieng.Informatik	oint Intern.Mast.in Comp.Science
Studiengang				
Abschluss/ Grad	B	M	B	M
Regelstudienzeit	6	4	7	4
Cluster	VII	VII	VII	VII
Fächergruppe	MN	MN	MN	MN
1. Fachsemester	180	43	11	0
2. Fachsemester	177	43	38	6
3. Fachsemester	72	35	5	0
4. Fachsemester	105	31	17	7
5. Fachsemester	34	14	4	3
6. Fachsemester	92	15	19	1
7. Fachsemester	22	0	6	0
8. Fachsemester	57	3	1	0
9. Fachsemester	6	1	0	0
10. Fachsemester	35	0	0	0
11. Fachsemester	1	0	0	0
12. Fachsemester	38	0	0	0
13. Fachsemester	3	1	0	0
14. Fachsemester	17	0	0	0
15. Fachsemester	1	0	0	0
16. Fachsemester	15	0	0	0
17. Fachsemester	0	0	0	0
18. Fachsemester	5	0	0	0
19. Fachsemester	1	0	1	0
20. Fachsem. u. höher	10	0	0	0
Summe Studierende	871	186	102	17
zusätzlich Beurlaubte	4	1		
Insgesamt	875	187	102	17
davon Int.Austauschstud.	7			

In der Regelstudienzeit	660	152	100	13
davon weiblich	103	12	9	0
Über der Regelstudienzeit	211	34	2	4
davon weiblich	30	8	1	1
Teilzeitstud. Insgesamt	1	4		
davon weiblich	0	0		

durchschn. Jahrgangsbz.	220	76	29	7
-------------------------	-----	----	----	---

Deutsche männlich	605	150	90	14
Deutsche weiblich	87	10	9	0
Ausländer männlich	133	16	2	2
Ausländer weiblich	46	10	1	1
Deutsche insgesamt	692	160	99	14
Ausländer insgesamt	179	26	3	3
männlich insgesamt	738	166	92	16
weiblich insgesamt	133	20	10	1

Fachbereich	I			
	Informatik	Informatik	Kooperativer Studieng.Informatik	oint Intern.Mast.in Comp.Science
Studiengang				
Abschluss/ Grad	B	M	B	M
Regelstudienzeit	6	4	7	4
Cluster	VII	VII	VII	VII
Fächergruppe	MN	MN	MN	MN

Ausländische Studierende				
Bildungsinländer/innen 1. FS	12	5	0	0
Bildungsausländer/innen 1. FS	24	2	0	0
Bildungsinländer/innen insg.	82	19	2	2
Bildungsausländer/innen insg.	97	7	1	1

Schul.Vorbildung Insges.				
allgem.HS-Reife männlich	371	134	41	15
allgem.HS-Reife weiblich	84	17	5	1
FH-Reife männlich	364	32	50	1
FH-Reife weiblich	49	3	5	0
Sonstige männlich	3	0	1	0
Sonstige weiblich	0	0	0	0
Schul.Vorbildung 1. FS				
allgem.HS-Reife männlich	75	19	1	0
allgem.HS-Reife weiblich	37	4	1	0
FH-Reife männlich	50	19	6	0
FH-Reife weiblich	17	1	2	0
Sonstige männlich	1	0	1	0
Sonstige weiblich	0	0	0	0

Beruf.Vorbildung Insges.				
abgeschl.Berufsausbild.männl.	11	4	2	0
abgeschl.Berufsausbild.weibl.	1	0	1	0
abgeschl.Studium männl.	31	0	1	0
abgeschl.Studium weibl.	28	0	1	0
Beruf.Vorbildung 1. FS				
abgeschl.Berufsausbild.männl.	0	0	0	0
abgeschl.Berufsausbild.weibl.	0	0	0	0
abgeschl.Studium männl.	24	0	0	0
abgeschl.Studium weibl.	23	0	0	0

Literaturverzeichnis

ACM/IEEE-CS 1999

ACM/IEEE-CS: *Software Engineering Code of Ethics and Professional Practice*. 1999. – URL <http://www.acm.org/about/se-code>. – [Online; Aktualisierungsdatum 12 Mai 2009]

Ahonen 2008

AHONEN, Tomi T.: Deeper insights into the 7th Mass Media channel, mobile is to the internet, what TV is to radio. In: *Communities Dominate Brands* (2008). – URL http://communities_dominate_blogs.com/brands/2008/05/deeper-insights.html. – [Online; Aktualisierungsdatum 27. August 2009]

Bauer u. a. 2008

BAUER, H. H. ; NEUMANN, M. M. ; REICHARDT, T.: Wann werden Mobile Marketing-Kampagnen akzeptiert? In: *Erfolfsfaktoren des mobile Marketing - Strategien, Konzepte und Instruments*, 2008

Brockhaus 2005

BROCKHAUS ; F.A. BROCKHAUS AG MANNHEIM F.A. BROCKHAUS GMBH, Bibliografisches I. und (Hrsg.): *Brockhaus Enzyklopädie in 30 Bänden*. 21. F.A. Brockhaus GmbH, Leipzig, Bibliografisches Institut und F.A. Brockhaus AG, Mannheim, 2005

Center 2009

CENTER, Hochschule Darmstadt Student S.: *hobit - Informationen zu Studien- und Ausbildungsmöglichkeiten in Darmstadt und der Region*. 2009. – URL <http://www.hobit.de/>. – [Online; Aktualisierungsdatum 27. Juli 2009]

Devriendt und Novelli 2006

DEVRIENDT, Danny ; NOVELLI, Porter: Bluetooth wireless awareness soars among consumers. In: *Bluetooth SIG* (2006), 4. – URL

http://bluetooth.com/Bluetooth/Press/SIG/IBLUETOOTHI_WIRELESS_AWARENESS_SOARS_AMONG_CONSUMERS.htm. – [Online; Aktualisierungsdatum 26. August 2009]

Dr. Pousttchi und Wiedemann 2006

DR. POUSTTCHI, Key ; WIEDEMANN, Dietmar G.: Mobile Marketing - Begriffserklärung und Kategorisierung / Arbeitsgruppe Mobile Commerce (wi-mobile), Universität Augsburg. 2006. – Forschungsbericht

Dufft 2003

DUFFT, Nicole: Basisreport Mobile Marketing - Einsatz, Erfolgsfaktoren, Dienstleister / BERLECON RESEARCH. 2003. – Forschungsbericht

Eckert 2008

ECKERT, Claudia: *IT-Sicherheit: Konzepte - Verfahren - Protokolle*. 2008

Erbs 2009

ERBS, Heinz-Erich: 02.09 i-news / Fachbereich Informatik, Hochschule Darmstadt. 2009. – Forschungsbericht

Haase und Martin 2009

HAASE, Matthias ; MARTIN, Alexander: Die Perspektiven des Bluetooth Marketing im Vergleich zu W-LAN und Over-the-Air. In: *mobile Zeitgeist* (2009), 1. – URL <http://www.mobile-zeitgeist.com/download/7/>. – Online Magazin, Ausgabe 1

Hammel u. a. 2007

HAMMEL, Holger ; DR. SASSENBERG, Thomas ; SCHOLZ, Heike: Bluetooth Marketing als Bestandteil moderner Kommunikationsstrategien / Whitepaper. 2007. – Forschungsbericht

Henn u. a. 2008

HENN, René ; HICHERT, Renate ; LÜDEMANN, Cord ; SYDOW, Linda ; WELLER, Ulrike: *Jahresbericht 2008*. 2008

Kizilok 2007

KIZILOK, Taner: *Akzeptanz und Reaktanz gegenüber Mobile Marketing*, Technische Universität Berlin, Diplomarbeit, 2007

Kremp 2007

KREMP, Matthias: Bluetooth-Werbung - Gratis-MP3 aus der Plakatwand. In: *Spiegel-Online* (2007). – URL <http://www.spiegel.de/netzwelt/mobil/0,1518,469554,00.html>. – [Online; Aktualisierungsdatum 26. August 2009]

Levine u. a. 2002

LEVINE, Rick ; LOCKE, Christopher ; SEARLS, Doc: *Das Cluetrain Manifest. 95 Thesen für die neue Unternehmenskultur im digitalen Zeitalter*. 2002

Miess und Seifert 2007

MIESS, Florian ; SEIFERT, Jörg: *Planung und Realisierung einer Middleware zur asynchronen Kommunikation zwischen mobilen Anwendungen und Webanwendungen*, Hochschule Darmstadt, Diplomarbeit, 2007

Miller 2009

MILLER, Andrea: *Mobile-Marketing-Instrumente im Kundenmanagement. Erfolgsfaktoren und Erfolgspotential - Eine State of the Art Analyse*. VDM, 2009

Post AG 2007

POST AG, Deutsche: *Direktmarketing Deutschland 2007 / Deutsche Post*. 2007. – Forschungsbericht

Pousttchi u. a. 2008

POUSTTCHI, K. ; TUROWSKI, K. ; WIEDEMANN, D. G.: *Mobile Viral Marketing - Ein State of the Art*. In: *Erfolgsfaktoren des mobile Marketing - Strategien, Konzepte und Instruments*, 2008

Pousttchi und Wiedemann 2007

POUSTTCHI, K. ; WIEDEMANN, D. G.: *Success Factors in Mobile Viral Marketing: A Multi-Case Study Approach*. In: *Proceedings of the 6th International Conference on Mobile Business*, 2007

Ranke 2004

RANKE, Johannes S.: *M-Commerce und seine rechtsadäquate Gestaltung*. Nomos Verlagsgesellschaft, 2004

Rogge 2006

ROGGE, Marko: *Bluetooth als Einfallstor - Wie Bluetooth-Marketing Anwender für*

mobile Viren desensibilisiert. In: *heise* (2006), 12. – URL <http://www.heise.de/security/Bluetooth-als-Einfallstor--/artikel/81447>. – [Online; Stand 28. August 2009]

Rösch 2006

RÖSCH, Bert: Auf dem Weg zu Mobile Marketing 2.0. In: *ONEtoONE* (2006), S. 8–13

Sassenberg und Berger 2007

SASSENBERG, Thomas ; BERGER, Ernst G.: Rechtliche Zulässigkeit von Werbung via Bluetooth. In: *Kommunikation und Recht* Bd. 10. Verlag Recht und Wirtschaft, Frankfurt, 2007, S. 499–503

Schiller 2003

SCHILLER, Jochen: *Mobilkommunikation*. 2003

Schmazit 2007

SCHMAZT, Klaus-Dieter: *Java Micro Edition*. 2. 2007

Schäfer und Toma 2008

SCHÄFER, J. ; TOMA, D.: Trends und Strategien im Mobile Marketing. In: *Erfolgsfaktoren des mobile Marketing - Strategien, Konzepte und Instruments*, 2008

SIG 2009

SIG, Bluetooth: Bluetooth / Bluetooth Special Interest Group (SIG). URL <http://german.bluetooth.com/Bluetooth/>, 2009. – Forschungsbericht. [Online; Aktualisierungsdatum 26. August 2009]

Sommerville 2001

SOMMERVILLE, Ian: *Software Engineering*. Pearson Studium Verlag, 2001

Stevens und Pettey 2009

STEVENS, Holly ; PETTEY, Christy: *Gartner Says Worldwide Mobile Phone Sales Declined 6 Per Cent and Smartphones Grew 27 Per Cent in Second Quarter of 2009*. 8 2009. – URL <http://www.gartner.com/it/page.jsp?id=1126812>. – [Online; Aktualisierungsdatum 14. August 2009]

SUN 2009a

SUN: *J2ME CLDC and K virtual machine: Frequently Asked Questions*. 2009. – URL <http://java.sun.com/products/cldc/faqs.html>. – [Online; Stand 14. August 2009]

SUN 2009b

SUN: *Java FAQ*. 2009. – URL http://www.java.com/de/download/faq/index_general.xml. – [Online; Aktualisierungsdatum 29. August 2009]

SUN 2009c

SUN (Veranst.): *Java Verified Manual*. 2009. – URL <http://javaverified.com/graphics/PDF/JV-manual-010809.pdf>

Tanenbaum 2002

TANENBAUM, Andrew S.: *Moderne Betriebssysteme*. 2002

Telekom 2008

TELEKOM, Detusche: *Superdistribution 2.0*. 2008. – URL <http://www.laboratories.telekom.com/ipws/Deutsch/News/Archiv/2008/Pages/Superdistribution20.aspx>. – [Online; Stand 24. August 2009]

Wikipedia 2009a

WIKIPEDIA: *Binärbaum* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL <http://de.wikipedia.org/w/index.php?title=Bin%C3%A4rbaum&oldid=62529706>. – [Online; Stand 3. August 2009]

Wikipedia 2009b

WIKIPEDIA: *Generische Programmierung* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL http://de.wikipedia.org/w/index.php?title=Generische_Programmierung&oldid=61218421. – [Online; Stand 4. August 2009]

Wikipedia 2009c

WIKIPEDIA: *Hypertext Transfer Protocol* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL http://de.wikipedia.org/w/index.php?title=Hypertext_Transfer_Protocol&oldid=62156875. – [Online; Stand 28. Juli 2009]

Wikipedia 2009d

WIKIPEDIA: *Java Database Connectivity* — *Wikipedia, Die freie Enzyklopädie*.

2009. – URL http://de.wikipedia.org/w/index.php?title=Java_Database_Connectivity&oldid=63527144. – [Online; Stand 30. August 2009]

Wikipedia 2009e

WIKIPEDIA: *Opt-in* — *Wikipedia, The Free Encyclopedia*. 2009. – URL <http://de.wikipedia.org/w/index.php?title=Opt-in&oldid=63708230>. – [Online; Stand 25. August 2009]

Wikipedia 2009f

WIKIPEDIA: *Opt-out (Permission Marketing)* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL [http://de.wikipedia.org/w/index.php?title=Opt-out_\(Permission_Marketing\)&oldid=59353045](http://de.wikipedia.org/w/index.php?title=Opt-out_(Permission_Marketing)&oldid=59353045). – [Online; Stand 25. August 2009]

Wikipedia 2009g

WIKIPEDIA: *Simple API for XML* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL http://de.wikipedia.org/w/index.php?title=Simple_API_for_XML&oldid=59008420. – [Online; Stand 26. September 2009]

Wikipedia 2009h

WIKIPEDIA: *Sniffer* — *Wikipedia, Die freie Enzyklopädie*. 2009. – URL <http://de.wikipedia.org/w/index.php?title=Sniffer&oldid=63950467>. – [Online; Stand 26. September 2009]

Wikipedia 2009i

WIKIPEDIA: *Superdistribution* — *Wikipedia, The Free Encyclopedia*. 2009. – URL <http://en.wikipedia.org/w/index.php?title=Superdistribution&oldid=293070280>. – [Online; Stand 29. Mai 2009]

Wissenschaftszentrum 2009

WISSENSCHAFTSZENTRUM, Kongresszentrum D.: *Exzellente Show-Bühne für das Business*. Pressemitteilung. 2009. – URL http://www.darmstadtium.de/index.cfm/pa_id/216/pr_id/193. – [Online; Aktualisierungsdatum 14 Juli 2009]

Zettel 2007

ZETTEL, Claudia: Konsumenten stehen auf Werbung. In: *Innovations-Report* (2007), 2. – URL <http://www.innovations-report.de/html/berichte/studien/bericht-78165.html>. – [Online; Aktualisierungsdatum 28. August 2009]

Zoller 2007

ZOLLER, Thierry: *Scheunentor Bluetooth?* 2007. – Vortrag auf der Heise Security Konferenz 2007