



Meilenstein 2

ENTWICKLUNGSPROJEKT INTERAKTIVE SYSTEME

ausgearbeitet von

Burcu Özata und Selin Öztürk

vorgelegt an der

TECHNISCHE HOCHSCHULE KÖLN
CAMPUS GUMMERSBACH
FAKULTÄT FÜR INFORMATIK UND
INGENIEURWISSENSCHAFTEN

im Studiengang

MEDIENINFORMATIK

Prof. Dr. Kristian Fischer
Prof. Dr. Gerhard Hartmann

Betreut von: David Bellingroth
Franz-L. Jaspers
Daniela Reschke

Gummersbach, 26.10.2015

Inhaltsverzeichnis

1	Zielhierarchie	3
1.1	Strategische Ziele	3
1.2	Taktische Ziele	3
1.3	Operative Ziele	3
2	Marktrecherche	4
2.1	Parkplätze in der Nähe	4
2.2	Parkonout	4
2.3	Fazit	4
3	Domänenrecherche	5
4	Alleinstellungsmerkmale	6
4.1	Sensor an der Schranke	6
4.2	Navigation	6
5	Methodischer Rahmen - MCI	7
5.1	User Centered Design vs. Usage Centered Design	7
5.2	MCI - Vorgehensmodelle	7
5.3	MCI - Vorgehensmodelle	8
5.4	Identifizierung der Stakeholder	9
5.5	Benutzer und Interessengruppen	9
5.6	Stakeholderanalyse	9
5.6.1	Studenten / Professoren / TH-Mitarbeiter	9
5.6.2	Besucher	10
6	Kommunikationsmodell	11
6.1	Deskriptiven Kommunikationsdiagramm	11
6.2	Präskriptiven Kommunikationsdiagramm	12
7	Risiken	13
7.1	Keine genaue Lokalisierung	13
7.2	Zeitliches Nicht Erreichen des Ziels	13
7.3	Implementierung der Schnittstellen	13
8	Proof Of Concepts	14
8.1	GPS Lokalisierung gewährleisten	14
8.2	Parkdaten mit der GET-Methode abrufen	14
9	Architekturdiagramm	15
9.1	Anwendungslogik und Präsentationslogik	15
10	Architekturbegründung	16
11	Projektplan	17

1 Zielhierarchie

Folgend werden die Ziele des Projekts veranschaulicht/ernannt. Für die weitere Planung ist unbedingt eine konkrete Formulierung der Ziele erforderlich und somit führt es zum Erfolg des Projekts.

1.1 Strategische Ziele

Durch die folgenden Ziele sollte den Benutzern der Anwendung eine Möglichkeit gegeben werden, dass Studenten, Mitarbeiter oder Besucher am TH-Parkplatz oder in der Nähe der TH zu parken. Als primäres Ziel soll auf langfristiger Sicht die Applikation einfach und übersichtlich ermöglicht werden, sodass die Studenten, Mitarbeiter, Professoren und Besucher voraussehend planen/sicher gehen können ob sie zur TH mit dem Auto fahren oder doch noch mit der RB25 fahren. Weiterhin soll die Möglichkeit gegeben werden, wenn man am TH-Parkplatz ist und diese "belegt", dass man mit einer Navigation an den nächst nahliegenden, günstigen Parkplatz weitergeleitet wird.

1.2 Taktische Ziele

Aus strategischen Zielen werden die taktischen Ziele abgeleitet: Damit alle Erwartungen und Anforderungen der Stakeholder berücksichtigt werden, soll eine Benutzermodellierung durchgeführt werden. Unter anderem müssen Schnittstellen zur Ortung und einer Navigation (Stadtkarte) integriert werden, damit der Benutzer zum nächsten Parkplatz weitergeleitet werden kann. User Interfaces soll vorzugsweise bescheiden gestaltet werden, um den Benutzern der Anwendung eine komfortable Nutzung zu ermöglichen. Durch die Anwendung sollen die Benutzer auf dem aktuellsten Stand gehalten werden, indem die Belegtheit des TH-Parkplatzes jederzeit aktualisiert wird.

1.3 Operative Ziele

Die operativen Ziele stellen fest, welche Schritte durchgeführt werden müssen, um das Endziel zu erreichen. Entscheidungen zu den Techniken der Vorgehensmodelle zur Mensch-Computer-Interaktion sollten getroffen und die technischen Anforderungen müssen für das System gefunden werden. Die benötigten Funktionen sollen technisch geprobt/geübt und durchgespielt werden, damit die Proof-Of-Concept erfüllt werden. In der Evaluierungsphase soll der Prototyp von realen Testpersonen getestet werden.

2 Marktrecherche

Es ist zwingend notwendig eine Marktrecherche durchzuführen, um das eigene Projekt von den Konkurrenzprodukten unterscheiden zu können. Es werden Vor- und Nachteile der folgenden Konkurrenzprodukte werden unten dargestellt:

2.1 Parkplätze in der Nähe

Durch die "Parkplätze in der Nähe" findet man die Parkplätze am aktuellen Standort und überall auf der ganzen Welt. Die APP zeigt Standorte in einer Liste und auf einer Karte und ermöglicht eine einfache One-Click-Navigation zu den Parkplätzen. Als Feedback ist zu erwähnen, dass es zwei Möglichkeiten zum anzeigen der Standorte gibt. Beim öffnen des App's den Standort selber durch die GPS - Signal feststellt. Im Gegensatz zum "Parkonout" ist diese App simple und benutzerfreundlich.

2.2 Parkonout

Bei der "Parkonaut" handelt es sich um eine Applikation, mit welcher man freie Parkplätze melden oder suchen kann. Die Darstellung des App bestehen aus Text und einer Karte. Informationen zur Parkplatz finden und Parkplatz melden, wie z.B: öffentliche Parkplatz suche, anbieten der Parkplätze und erhalten von Punkte bei der Parkplatz Übergabe sind vorhanden. Im Vergleich zur "Parkplätze in der Nähe" bietet diese Applikation mehrere Möglichkeiten, wie jeder Benutzer mit anderen Benutzern unter sich Kommunizieren können. Im Großen und Ganzen ist die Applikation ein gut geeignetes Community-Portal, womit es ermöglicht wird die freien Parkplätzen festzuhalten und mit anderen Benutzern zu teilen und neue Routen/Parkplätzen zu entdecken. Allerdings ist die App sehr kompliziert und zeitaufwändig um es ganze auf einmal zu verstehen.

2.3 Fazit

Die beiden Konkurrenzprodukten sind sehr gut, doch es entspricht keinerlei das Ziel unseres Projekts an. Die 1.APP umfasst eine sehr weiteren Radius, was zu unübersichtlich wird und bei der 2.APP steht das Kommunizieren und austauschen der Nutzer unter sich im Mittelpunkt. Doch beide APP's gibt zusätzliche Informationen wie Parkkosten, Parkdauer, Öffnungszeiten etc. nicht an. Was wir uns Vorstellen ist es eine spezielle Applikation für die TH - Köln, aber Planen erst mal für den Campus Gummersbach, in dem er uns zeigt, ob es einen freien Parkplätzen auf dem Campus Parkplatz gibt oder nicht. Hieraus leitet sich oftmals das Alleinstellungsmerkmal ab, welches im weiteren der Aufgaben dargestellt wird.

3 Domänenrecherche

4 Alleinstellungsmerkmale

Das Alleinstellungsmerkmal zeichnet sich dadurch aus, dass dem Benutzer eine Anwendung geschaffen werden soll, die es ermöglicht, zu erfahren, ob es freie Plätze auf dem Parkplatzgelände zu finden sind oder eine nächstmögliche Parkmöglichkeit gibt. Diese Anwendung soll den Nutzern das Suchen der Parkplätze erleichtern indem eine Benachrichtigung auf einem Board (in der Applikation) erscheint, ob freie Plätze existieren und gegebenenfalls zu neuen Parkmöglichkeiten in der Nähe navigiert.

4.1 Sensor an der Schranke

Die Sensoren sollen dem Benutzer die Möglichkeit geben, dass sie einerseits Vorort sehen können, wie viele Parkplätze frei oder belegt sind und andererseits diese durch die Anwendung im Voraus sehen.

4.2 Navigation

Die Navigation soll die Benutzer nach Antritt des Parkplatzes, wenn es belegt ist, an die nächst nah liegenden, Kostengünstigen Parkplatz weiterleiten. Die Funktion soll also das schnell Parken, nah Parken und das günstig Parken ermöglichen.

5 Methodischer Rahmen - MCI

5.1 User Centered Design vs. Usage Centered Design

Unter User Centered Design versteht man die Benutzerorientierte Gestaltung eines Produktes um eine Zielorientierte Auswahl zu erreichen.

Der User Centered Design ist ein Ansatz zur Gestaltung von interaktiven Systemen, in dem die unterschiedliche Merkmale, Bedürfnisse, Fähigkeiten und Erfahrungen von Benutzer im Mittelpunkt stehen. Das Ziel hierbei ist es, leicht erlernbare und einfach zu bedienende interaktive Produkte zu gestalten, die dem Benutzer eine optimale Interaktion und ein angenehmes Nutzererlebnis zu ermöglichen, in dem das System sich an den Benutzer anpasst. Der Unterschied zwischen User Cetered Design und Usage Centered Design ist, dass der Nutzungskontext sich variieren kann. Beim USage Centered Design ist der Kernpunkt die Nutzungsanforderungen, worauf der Entwicklung des Systems stattfinden muss.

Aus diesem Grund wird das User Centered Deisgn zur Umsetzung des System erfolgen, um die Anforderungen und Bedürfnisse des Benutzer gerecht zu sein, damit der Benutzer seinen Ziel bestmöglich umsetzen kann.

5.2 MCI - Vorgehensmodelle

Laut Norm ISO 9241 Teil 210 sind die MCI-Vorgehensmodelle ein internationaler Standart, welches die Richtlinien der Mensch-Computer Interaktion enthält. Die Interaktion ist der wesentliche Vorteil der MCI-Vorgehensmodells. Sie findet so oft statt, bist die Gestaltungslösung den Nutzungsanforderungen entspricht. Außerdem ermöglicht die Allgemeinheit der Norm eine individuelle Gestaltung der Arbeitsweise, welches ein weiterer Vorteil darstellt. Abgesehen vom MCI-

Vorgehensmodell sind auch weitere Vorgehensmodelle wie das Scenariobased usability engineering nach Rosson und Carroll oder das Usability engineering lifecycle nach Mayhew vorhanden. Natürlich bringen diese Vorgehensmodelle neben der Vorteilen auch Nachteile mit sich. Auch wenn das Usability engineering lifecycle die parallele Planungs- und Entwicklungszeit ermöglicht, wird die Gebrauchstauglichkeit erst zu einem späteren Zeitpunkt gewährt leistet. Da dies für diese Abgesehen von dem USability engineering Vorgehensmodell erbringt das Scenariobased usability engineering den Benutzern eine kognitive Sicht. Der Nachteil hierbei ist jedoch der zeitliche Aufwand, welches bei der Wahl dieses Vorgehensmodell betrachter werden muss. Nach durchdenken der Vor- und Nachteile der Vorgehensmodelle wird deutlich, dass die Norm ISO 9241 Teil 210 sich Vorteilhaft ergibt. Laut ISO 9241 Teil 10 wird der Benutzer sowohl

konkret analysiert als auch in jedem Entwicklungsschritt mit eingebunden. Es werden in der Entwicklungsphase Papierbasierte Prototypen erstellt, um geeigneter Design Lösungen zu finden. Zwar stellt Computerbasierte Prototypen eine von selbst ablaufende Animationmöglichkeit dar, sind jedoch bei der Umsetzung einer Flash Animation aus zeitlichen Gründen nicht realisierbar. Um die bestmögliche Designlösung zu erzielen, sollten mehrere Lösungen zum Prototypen vorhanden sein. Diese kann man durch reine Skizzen, welches mit Hilfe von Papierbasierten Prototypen erstellt werden, erzielen.

5.3 MCI - Vorgehensmodelle

Die unten dargestellte Benutzerzentrierte Gestaltungprozess nach der Norm ISO 9241 Teil 210 (Abbildung 1) soll helfen um die präzisiere Entscheidungen treffen zu können.

Als erstes muss der Menschzentrierte Gestaltungprozess geplant werden, um die Identifizierung und Erstellung von Nutzungskontext zur Ermittlung von Anforderungen der Benutzer festzulegen. Um es genauer zu betrachten, muss eine detaillierte Benutzermodellierung erstellt werden, in dem erst die Stakeholder, dann User Profiles und anschließend die Personas erstellt werden, um die Besonderheit der Testpersonen darstellen zu können. Um die Zielsetzung des Benutzers erfassen zu können, muss die User Cases auch erstellt werden.

Als letztes wird die Evaluation durchgeführt. Da ein analytisches Verfahren zur Evaluation mit Usability-Experten zur hohen Kosten führen würde, wird ein empirisches Verfahren angewendet. Der Unterschied zwischen quantitatives und qualitatives Verfahren ist, dass beim quantitativen Verfahren geht es um die Evaluation der Ergebnisse, die zahlenbasiert vermessen werden sollen. Doch beim qualitativen Verfahren handelt es sich um die sprachliche Einschätzung von Testpersonen. Folgende Evaluationsmethoden werden angeboten: Heuristi-

sche Evaluation, Cognitive Walkthrough, AttrakDiff und Think Aloud. Als Evaluationsmethode haben wir die Think Aloud bevorzugt, weil die Evaluation von Benutzern durchgeführt werden, die Interesse an dem System haben. Dadurch käme die Cognitive Walkthrough der Benutzer zum Vorschein. Es gibt zwei Phasen: die Aufwärmphase und die Evaluationphase, in dem die Methoden von einem Moderator und dem Probanden durchgeführt werden. In der Evaluationphase soll der Proband gewisse Funktionalitäten, als Anweisung erhalten und es ausprobieren und dabei laut denken. Es besteht die Gefahr, dass die Probanden nicht genügend laut denken, deswegen soll die Aufwärmphase reichlich ermutigen. Der Cognitive Walkthrough von Polson Lewis ist nicht als Evaluationsmethode geeignet, da das System wenig Funktionalitäten und Komplexität beinhaltet.

5.4 Identifizierung der Stakeholder

Um eine genaue Benutzermodellierung zu haben, soll die Identifizierung von Stakeholdern erfasst werden, welches für die Gestaltung laut ISO 9241 Teil 210 notwendig ist. Stakeholder werden als „ Einzelpersonen oder Organisationen, die ein Anrecht, einen Anteil, einen Anspruch oder ein Interesse auf ein bzw. an einem System oder an dessen Merkmalen haben, die ihren Erfordernissen und Erwartungen entsprechen.“ [ISO/IEC 15288:2008] definiert. Um das Kommunikationsmodell gut/richtig gestalten und erfassen zu können, müssen demgemäß die notwendigen Stakeholder aufgelistet werden, die für den Meilenstein 2 erforderlich sind.

5.5 Benutzer und Interessengruppen

Bezeichnung	Beschreibung
Student	alle Studenten der TH-Köln
Professor	alle Professoren der TH-Köln
Wissenschaftliche Mitarbeiter	alle Mitarbeiter der TH-Köln
Besucher	eine bestimmte Anzahl von Besuchern, die für kurze Zeit die TH besuchen
Parkplatz	verschiedene Parkplätze in der Nähe der TH, die den TH-Studenten, Professoren, Mitarbeiter oder Besucher erlauben dort zu parken

5.6 Stakeholderanalyse

Um eine ordentliche Repräsentation von Stakeholder zu erhalten, ist eine Auflistung der Benutzermerkmale konstitutiv notwendig. Nähere Definition der Stakeholder sieht wie folgt aus:

5.6.1 Studenten / Professoren / TH-Mitarbeiter

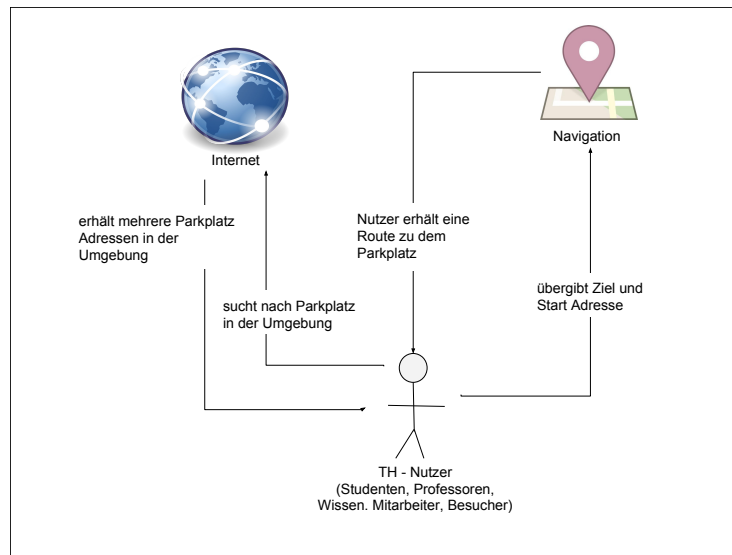
Die Stakeholder Student/Professor/Mitarbeiter schließen alle Studenten/Professoren/Mitarbeiter, die an der TH-Köln Campus Gummersbach studieren, unterrichten und arbeiten. Studenten/Professoren/Mitarbeiter sollen die primären Benutzer des Systems darstellen. Die Anwendung soll zu ihrem Nutzen sein, indem Sie sich mit dem System direkt interagieren. Durch die Gestaltung, was zu Ihrem Interesse geplant und vorgenommen werden, sollten sie alle Funktionen des System problemlos nutzen können.

5.6.2 Besucher

Der Stakeholder Besucher schließt alle Besucher, die die TH Köln Campus Gummersbach besuchen wollen. Besucher sollen die sekundären Benutzer des Systems darstellen. Sie können die Anwendung für Informationsquelle benutzen.

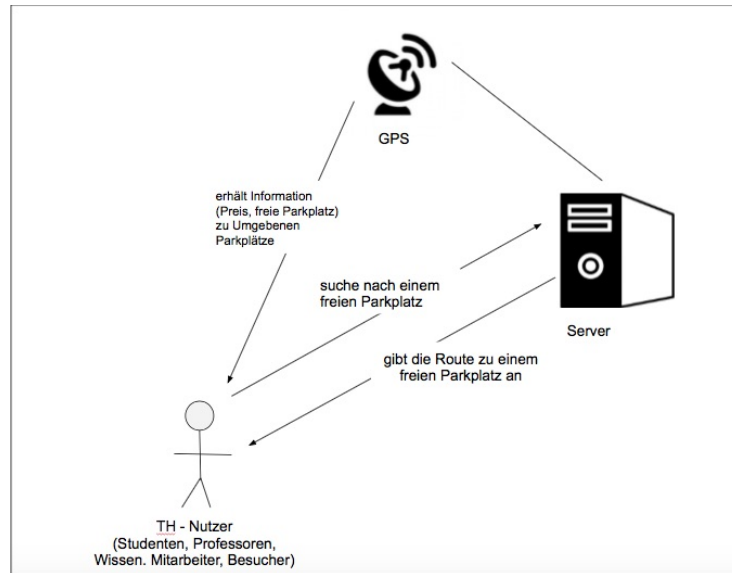
6 Kommunikationsmodell

6.1 Deskriptiven Kommunikationsdiagramm



Im deskriptiven Kommunikationsdiagramm sind die Stakeholder Student, Professoren, wissenschaftliche Mitarbeiter und die Besucher abgebildet. Im Ist-Zustand erhalten die Nutzer über das Internet Informationen, darüber welche Parkplätze in der Umgebung sich befinden. Die Nutzer erhalten keinerlei Informationen darüber ob sich auf dem Parkplatzgelände freie Plätze befinden. Der Nutzer muss selbst nach freien Plätzen suchen. Sofern der Nutzer eine Adresse zu einem Parkplatzgelände hat, kann der Nutzer sich über ein Navigationssystem (APP oder Autonavigation) zur gewünschtem Platz navigieren lassen ohne auskunft über freie Parkplätze.

6.2 Präskriptiven Kommunikationsdiagramm



Im präskriptiven Kommunikationsdiagramm sind die Nutzergruppen unverändert wie im deskriptiven Kommunikationsdiagramm. Anhand der Abbildung ist zu erkennen, dass alle Information wie freie Plätze, Parkgebühr und Route zu einem freien Parkplatz über die Parkapp gebündelt an den Nutzer übermittelt werden sollen. Der Nutzer soll automatisch über seine GPS Koordinaten Informationen über umgebene Parkplätze erhalten. Desweiteren soll dem Nutzer eine eigene suche nach einem freien Parkplatz bereitstehen. Dem Nutzer soll über die ermittelte Position eine entsprechende Route zu einem freien Parkplatz angeboten werden.

7 Risiken

7.1 Keine genaue Lokalisierung

Es kann sein, dass die Navigation/Karte/Ortungsdienst die Benutzer falsch weiterleitet, neue Straßen nicht erkennt oder nicht in der Karte gespeichert sind. Damit solche Probleme frühzeitig erkannt werden, muss die Anwendung mehrmals durchgeführt/getestet werden. Getestet, indem die Anwendung mehrmals zu den verschiedenen Parkplätzen navigiert wird, die in den Proof Of Concepts vorgenommen werden.

7.2 Zeitliches Nicht Erreichen des Ziels

Es könnte ein Risiko eintreten, indem das Ziel nicht erreicht werden kann, aufgrund der Tatsache, dass das Projekt zur verfügbaren Zeit sehr um weitreichend ist. Um dies zu vermeiden, sollten die wichtigsten Kernpunkte zu Realisierung der Anwendung möglichst genau beschrieben werden.

7.3 Implementierung der Schnittstellen

Viele unterschiedlich technische Funktionalitäten werden im Projekt aufgewiesen, die umgesetzt werden müssen. Es besteht die Gefahr, dass diese scheitert, da mehrere Schnittstellen implementiert werden müssen um die Funktionalität zu sichern. Wenn die Abfragen funktional nicht durchgeführt, kann es dazu führen das der Dienst nicht erfolgreich ablaufen kann. Zunächst sollten nur die als wichtig erachteten technischen Funktionalitäten des Systems realisiert werden, die dieses Risiko minimieren sollen.

8 Proof Of Concepts

Die Proof-Of-Concepts leiten sich aus den Risiken, die vorher ausgearbeitet wurden. Die Funktionen der Anwendung sollten zwingend getestet und realisiert werden, um Erfolg zu haben. Auf Grund dessen sollten die Funktionen sorgfältig ausgearbeitet werden, um die Umsetzung sicher zu stellen.

8.1 GPS Lokalisierung gewährleisten

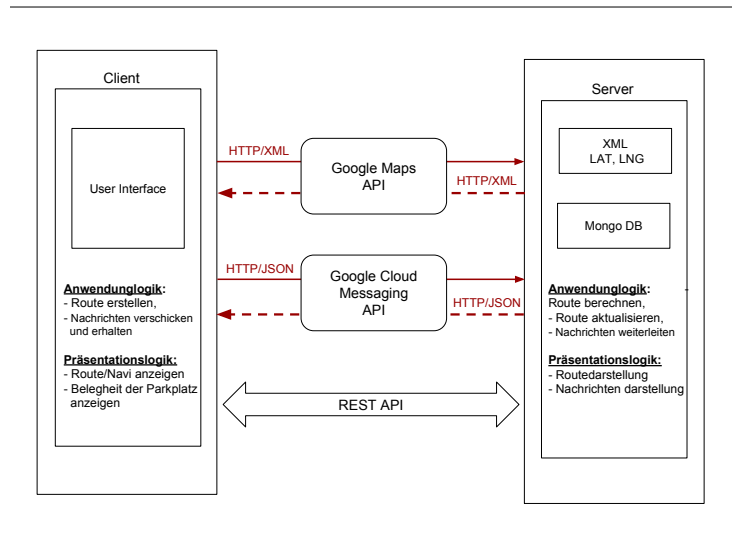
Die Lokalisierung ist notwendig, damit die Routenplanung in der Implementierungsphase fertiggestellt werden kann. Die Routen sollen mit Hilfe der Google Maps API ausgeführt werden. Die Genauigkeit der Lokalisierung führt zum Erfolg, die ebenfalls an verschiedenen Standorten getestet werden sollen. Genauso sollte die zeitliche Spanne des Ortungsdienstes gemessen werden, die nicht all zu lange dauern sollte. Außerdem sollte die Navigation an mehreren Routen navigiert werden, und somit überprüft werden, ob der Benutzer das Ziel erreicht. Auch hier müsste die Zeit gemessen werden. Hier könnte das Problem sein, dass kein Standort erkannt wird und/oder die Navigation, also die Google Maps API nicht verwendet werden kann. Als Notlösung könnte man den OpenStreetMap API mit einbinden.

8.2 Parkdaten mit der GET-Methode abrufen

Das Erhalt der Parkdaten sollten getestet werden, indem es überprüft wird, ob es mit der GET-Methode realisiert werden kann. Dafür sollte eine Anwendung programmiert werden, welches die Daten im XML-Format ausgibt. Um das durchgeführte, in Erfolg beweisen zu können, soll die Anwendung mehrmals an verschiedenen Smartphones getestet werden. Das Testen gilt als erfolgreich, wenn die Anwendung, die richtigen belegten und freien Parkplätze anzeigen kann. Das einzige, was als Fehler auftauchen könnte, wäre, dass der Abruf der GET-Methode fehlgeschlagen wird.

9 Architekturdiagramm

Die Darstellung des Architekturdiagramms soll die Verteiltheit und die Kommunikationsabläufe des Systems präsentieren.



9.1 Anwendungslogik und Präsentationslogik

Mit der Eröffnung der Anwendung schickt der Server dem Client die Anzahl der freien und belegten Parkplätze an. Wenn die Plätze belegt sind und der Benutzer auf das Button 'Navigation' klickt, bittet der Client dem Server das er einer der Routen berechnet und dem Client wieder zurück schickt damit der Benutzer navigiert werden kann. Durch die Anzeige der Belegtheit der Parkplätze und erscheinen der Karte der Route, macht sich die Präsentationslogik bemerkbar.

10 Architekturbegründung

Google Produkte können mit Android perfekt eingebunden werden, weil Android mit der Programmiersprache JAVA programmiert wird. Da JAVA die Voraussetzung ist das Projekt umzusetzen, ist es die beste Möglichkeit die Anwendung mit JAVA und XML, was in dem Modul Web basierte Anwendung schon umgesetzt wurde. Um den Datenverlust zu vermeiden, soll die Datenhaltung auf dem Server ablaufen. Damit die Anwendung ihre Daten publizieren kann, sollte das Architekturmodell „Representational State Transfer“ für verteilte Systeme benutzt werden. Durch das Transportprotokoll HTTP, sollte der Client auf die vom Server bereitgestellten Ressourcen zugreifen können. Da Google Messaging ein kostenloser Dienst ist und ausreichende Informationen zur Implementierung zur Verfügung stellt, kann es als Push-Dienste verwendet werden. Die synchrone und asynchrone Kommunikation zwischen Serverinstanzen und der mobilen Endgeräte wird durch den Google Cloud Messaging ermöglicht. Die Routenplanung wird mit Google Maps implementiert. Google Maps API wird kostenlos zur Verfügung gestellt und ist sehr flexibel, d.h man kann das Kartenmaterial zur persönlichen Routen visualisieren.

11 Projektplan

Datum / KW	Aktivität	1. Unteraktivität	2. Unteraktivität	Workload geplant /	Workload gesamt	Workload tatsächlich	
						Burcu Özata	Selin Öztürk
	Exposé	Ideenfindung	Brainstorming		3h	3h	0h
	Dokumentaufbau	Layout	Microsoft-Word	2h		2.5h	0h
					5h	5.5h	0h
12.10.15	Meilenstein 1	Nutzungsproblem		1h		1h	0h
		Zielsetzungen des Projektes		1h		1h	0h
		Verteiltheit der Anwendungslogik		1h		1h	0h
		Wirtschaftliche/Gesellschaftliche Aspekte		1h		1h	0h
					4h	4h	0h
26.10.15	Meilenstein 2	Zielhierarchie	Strategische Ziele	1h		2h	0h
			Taktische Ziele	1h		2h	0h
			Operative Ziele	1h		1h	0h
		Marktrecherche		3h		0h	4h
		Domänenrecherche		3h		0h	1h
		Aktinstellungsmerkmale		1h		1h	0h
		Methodischer Rahmen (MCI)		10h		3h	7h
		Kommunikationsmodell		4h		0h	5h
		Risiken		3h		3h	0h
		Spezifikation der POCs		1h		1.5h	0h
		Architekturdiagramm		1h		2h	0h
		Architekturbegründung		2h		2h	0h
		Projektplan		2h		2h	0h
		Projektbegründungen					
					33h	19.5h	17h
			Insgesamt	Soll: 42h		Ist: 29h	Ist: 17h

Abbildungsverzeichnis

Tabellenverzeichnis

1	Abkürzungsverzeichnis	20
---	---------------------------------	----

12 Literaturverzeichnis

Abkürzungsverzeichnis

Abkürzung	Bedeutung
etc.	et cetera (und die Übrigen)
APP	Application Software
API	Application-Programming-Interface

Tabelle 1: Abkürzungsverzeichnis