



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Master

Alewtina Schuman

Ein einfach benutzbares mobiles  
Navigationssystem für Fußgänger

Alewtina Schuman

Ein einfach benutzbares mobiles  
Navigationssystem für Fußgänger

Masterarbeit eingereicht im Rahmen der Masterprüfung  
im Studiengang Angewandte Informatik  
am Studiendepartment Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. rer. nat. Kai von Luck  
Zweitgutachter : Prof. Dr. rer. nat. Jörg Raasch

Abgegeben am 23. April 2008

**Alewtina Schumann**

**Thema der Diplomarbeit**

Ein einfach benutzbares mobiles Navigationssystem für Fußgänger

**Stichworte**

Fußgänger-Navigation, Landmarken, Wegsuche, mobile Systeme, Software-Ergonomie

**Abstract**

Ziel dieser Arbeit ist es, auf Basis der Untersuchungen der natürlichen menschlichen Orientierung herauszufinden, welchen Einfluss Landmarken auf die Benutzbarkeit eines mobilen Navigationssystem für Fußgänger nehmen.

**Alewtina Schumann**

**Title of the paper**

A simply useable mobile Pedestrian Navigation-System

**Keywords**

pedestrian navigation, landmarks, wayfinding, mobile systems, software-ergonomics

**Abstract**

The aim of this thesis is the research into importance of landmarks for the mobile pedestrian navigation-systems. This research is based on the investigations of the discipline of spatial cognition about the human wayfinding concepts.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>6</b>
<b>1. Einführung</b>	<b>7</b>
1.1. Motivation . . . . .	8
1.2. Ziel dieser Arbeit . . . . .	9
1.3. Gliederung der Arbeit . . . . .	9
<b>2. Grundlagen</b>	<b>11</b>
2.1. Menschliches Orientieren . . . . .	11
2.2. Landmarken . . . . .	13
2.2.1. Eigenschaften von Landmarken . . . . .	15
2.2.2. Bedeutung von Landmarken für die Navigation . . . . .	18
2.3. Linguistische Beschreibung des Weges . . . . .	21
2.4. Vergleichbare Arbeiten . . . . .	22
2.4.1. Active Landmarks . . . . .	22
2.4.2. Bum Bag Navigator . . . . .	23
2.4.3. Navigon 7100 . . . . .	27
2.4.4. City Maps . . . . .	28
2.5. Zusammenfassung . . . . .	29
<b>3. Analyse</b>	<b>30</b>
3.1. Beispielszenario Frankfurter Flughafen . . . . .	31
3.2. Ergonomie und ihre Anforderungen an die Fußgängerführung . . . . .	32
3.2.1. Multimodale Navigation . . . . .	33
3.3. Funktionale Anforderungen . . . . .	34
3.4. Nicht funktionale Anforderungen . . . . .	37
3.5. Zusammenfassung . . . . .	41
<b>4. Design</b>	<b>42</b>
4.1. Gebäudemodell . . . . .	42
4.1.1. CityGML . . . . .	43
4.2. Entwurf der Anwendungsarchitektur . . . . .	44
4.3. Entwurf der Client-Server-Architektur . . . . .	48

---

4.4. Framework für die Entwicklung mobiler Anwendungen . . . . .	49
4.5. Realisierung des Software-Prototyps für die Evaluation . . . . .	51
4.5.1. Hardware . . . . .	52
4.5.2. Software . . . . .	53
4.5.3. Realisierung der Karte . . . . .	53
4.6. Zusammenfassung . . . . .	55
<b>5. Evaluation</b>	<b>57</b>
5.1. Test-Szenario: Informationstage an der HAW Hamburg . . . . .	57
5.1.1. Durchführung des Tests . . . . .	58
5.2. Auswertung . . . . .	61
5.3. Zusammenfassung . . . . .	63
<b>6. Methodische Abstraktion</b>	<b>64</b>
<b>7. Zusammenfassung</b>	<b>65</b>
<b>Literaturverzeichnis</b>	<b>68</b>
<b>A. Anhang</b>	<b>71</b>
A.1. Anwendungsfälle . . . . .	71
A.2. Gebäudemodell in CityGML . . . . .	74

# Abbildungsverzeichnis

2.1. Eine Landmarke Quelle: [17]	14
2.2. Beide Zellen haben gleiche Größe, aber die rechte Zelle ist länger und bildet damit größere Barriere; Quelle: [29]	18
2.3. Sechzehn Richtungen bestimmen die Position eines Objektes; Quelle: [32]	21
2.4. Informationsfluss einer aktiven Verbindung; Quelle: [8]	23
2.5. Bum Bag Navigator; Quelle: [5]	24
2.6. Extra View; Quelle: [4]	25
2.7. Abstract View; Quelle: [4]	25
2.8. Das Navigon 7100 in der Handy-Version (3D)	27
2.9. Darstellung der Karte durch City Maps	28
2.10. 3D-Darstellung von Landmarken am Beispiel	29
4.1. Die fünf Levels of Detail, definiert von CityGML; Quelle: [15]	44
4.2. Drei-Schichten-Architektur	46
4.3. Eine Kompetenzverteilung nach Tanenbaum [34]	48
4.4. Android-Architektur; Quelle: [1]	50
4.5. Graphische Darstellung vom Android-Emulator	52
5.1. künstliche Landmarken	58
5.2. Darstellung der Karte mit Landmarken	59
5.3. Tobii Eye Tracker	59
5.4. Infrastruktur-Aufbau für den Evaluierungstest	60
5.5. Suchen der Landmarke auf dem Bildschirm des mobilen Gerätes (der rote Punkt zeigt die Fokussierung der Augen, die rote Linien zeigen Augenbewegungen)	61
5.6. Suche Landmarke in der realen Umgebung (a)	62
5.7. Suche Landmarke in der realen Umgebung	62
5.8. Interaktion zwischen dem Bildschirm und Realität	63

# 1. Einführung

Die Navigationssysteme für Autos werden seit mehreren Jahren in der Praxis eingesetzt. Sie sind keine Neuigkeit mehr. Die Technik ist ausgereift und Systeme sind stabil. Sie benutzen GPS für die Positionserkennung und Tracking sowie Straßenkarten für die Navigation selbst.

Seit kurzem gewinnt auch die Fußgänger-Navigation immer mehr Aufmerksamkeit der Forschung und der Wirtschaft. Das kommt einerseits daher, dass die Technologien sich weiterentwickelt haben und eine neue Generation von mobilen Geräten auf dem Markt erschienen ist. Die modernen Handys sind kleine Multimediageräte. Die Bildschirmauflösung wird immer höher, die Farbpalette größer und der Klang natürlicher. Des Weiteren werden mobile Geräte immer mächtiger, was die Verarbeitungsgeschwindigkeit und besonders den (Arbeits-) Speicher betrifft. Das alles erlaubt erst die Entwicklung mobiler Navigationssysteme für Fußgänger.

Andererseits gibt es immer mehr öffentliche Einrichtungen und Bauten, die große Flächen besitzen und von komplexer Struktur sind. Als Beispiel können Flughäfen, große Bahnhöfe oder Messehallen genannt werden. Um sich in solchen Einrichtungen zurecht zu finden, braucht man Hilfe eines Navigationssystems. Ein anderer Grund für die Entwicklung der Navigationssysteme für Fußgänger ist die wachsende Tourismusbranche. Immer mehr Menschen verreisen in andere Länder und besuchen fremde Städte. Ein Navigationssystem kann Menschen in einer unbekanntem Umgebung helfen, schnell die POIs (points of interest) zu finden, und Routen anbieten, die interessante Stellen mit einbinden und sogar zusätzliche Informationen dazu liefern.

Die ersten Fußgänger-Navigationssysteme basierten auf Auto-Navigation. Es gibt aber eine Menge Merkmale, die Fußgänger-Navigation von Auto-Navigation unterscheiden.

Als erstes ist GPS für die Positionsbestimmung von Personen ungeeignet, da es viel zu ungenau ist. Außerdem sind Fußgängerwege nicht klar definiert wie die Autostraßen. Es gibt eher große Plätze, Messehallen oder Flughäfen mit nur implizit vorhandenen Wegen und vielen Ein- und Ausgängen. Wenn man sich zu Fuß bewegt, gelten andere Regeln. Die Aussagen, wie „in 50m nach links“ gelten nicht, da man als Fußgänger nicht weiß, wann man die fünfzig Meter zurückgelegt hat. Fußgänger verhalten sich auch anders. Für sie gibt es keine strengen Vorgaben, wie auf der Straße (Verkehrsregeln). Eine optimale Route heißt

für die Fußgänger auch nicht immer die kürzeste Route. Es ist einerseits von dem Ziel der Navigation abhängig. Z.B wollen Touristen die interessanteste Route wählen, die vielleicht etwas länger ist, aber viele POIs einbezieht. Sicherheitsaspekte spielen auch eine wichtige Rolle. So würde man am Abend einen anderen Weg nehmen wollen, als am Tag (z.B. wenn dieser über einen Wald geht). Oft bevorzugen Fußgänger einfachere Routen mit wenigen Entscheidungspunkten den kürzeren, aber komplizierten Wegen [20].

Also stellt Fußgänger-Navigation mehr Anforderungen an ein System als die Auto-Navigation. Die wichtigsten Fragen dabei sind: Wie soll sich der Fußgänger orientieren und welche Informationen braucht er dabei.

## 1.1. Motivation

Um ein geeignetes Navigationssystem für Fußgänger zu erstellen, muss man erst feststellen, wie die Menschen den Weg finden, wie orientieren sie sich; also wie funktioniert die natürliche menschliche Navigation. Die Fußgänger-Navigation ist eher eine Interaktion zwischen einem Fußgänger und seiner Umgebung. Für die Navigation nutzen Menschen ihre persönlichen Fähigkeiten: räumliche Orientierung, kognitive und Verhaltensfähigkeiten. Außerdem ist ihr Verhalten von ihrem eigentlichen Ziel abhängig. So benutzen sie für das Finden eines Weges in einer Großstadt andere kognitive Fähigkeiten als bei der Orientierung in einem Gebäude.

Auf der Basis zahlreicher empirischer Studien wird belegt, dass die natürliche Navigation und Orientierung des Menschen zwischen zwei Punkten im Wesentlichen auf zwei Mechanismen beruht [22]:

- **routenbasierter Mechanismus:** Der Weg zwischen einem Start- und einem Zielpunkt wird in mehrere Etappenpunkte (Knoten) eingeteilt. Die Navigation eines Individuums entlang der Route basiert dabei auf der Richtung und dem Abstand zum jeweils nächsten Etappenpunkt.
- **ortsbasierter Mechanismus:** Die Navigation eines Individuums basiert dabei auf der Position und dem Abstand zu einer nahe gelegenen Landmarke, die auf dem Weg zwischen einem Start- und einem Zielpunkt identifiziert werden kann.

Weiter ergeben Untersuchungen der menschlichen Wahrnehmung und natürlichen Orientierung im Raum, dass Menschen, wenn sie einen Weg beschreiben oder finden müssen, sich auf die Richtung und die Orientierungspunkte sog. Landmarken stützen. Daher sind Landmarken unentbehrlich für die Fußgänger-Navigation. Ein Navigationssystem soll für Menschen eine natürliche Hilfe darstellen. Deswegen soll man bei der Entwicklung eines Navi-

gationssysteme für Fußgänger die Ergebnisse oben genannter Untersuchungen nutzen und Landmarken in die Routenbeschreibung einbauen.

Die Anweisungen und Landmarken sollen die Navigierten nicht überfordern, also nicht zu detailliert aber auch nicht zu unpräzise sein [12]. Das betrifft auch die Art, wie die Anweisungen und die Route dem Benutzer präsentiert werden: sprachliche Ausgabe, als Text, 3D-Route, Bilder von der Umgebung. Die Person befindet sich meist in einer unbekannteren Umgebung, wenn sie Navigationshilfe braucht, und kann nicht ständig auf den Bildschirm schauen. Außerdem ist die Navigation in den meisten Fällen eine von mehreren nebensächlichen Aufgaben [35]. Also darf das System die Person nicht von ihrer Hauptaufgabe ablenken. So könnte z.B. das Ausgabegerät (Handy oder PDA) vibrieren, wenn man an einem neuen Entscheidungspunkt angekommen ist, und damit dem Benutzer mitteilen, dass er sich von System Hilfe holen kann. Bis dahin bräuchte man die Ausgabe nicht unbedingt und könne sich auf andere Aufgaben konzentrieren.

## 1.2. Ziel dieser Arbeit

Damit Navigationssysteme für Fußgänger von diesen akzeptiert werden, sollen sie einfach und intuitiv in der Benutzung und Gestaltung sein. Die Entwicklung solcher Navigationssysteme erfordert kognitive Aspekte der Wegfindung beim Menschen [10]. Landmarken gehören zu solchen Aspekten und sind natürliche Orientierungshilfe.

Die zentralen Fragestellungen dieser Arbeit sind: Welche Landmarken sollen in die Routenbeschreibung eingebaut werden? Wie werden Landmarken durch das System präsentiert? Inwieweit machen Landmarken ein mobiles Navigationssystem einfacher und benutzergerechter?

Das Entwickeln eines vollständigen Navigationssystems ist nicht der Inhalt dieser Arbeit. Ebenfalls gehören Positionsbestimmung und Routenberechnungsalgorithmen nicht zu dem Themengebiet.

Es wird eine Software gebaut, die die Durchführung des Tests, der die Nützlichkeit von Landmarken für die Fußgänger-Navigation bestätigen soll, unterstützen wird.

## 1.3. Gliederung der Arbeit

Kapitel 2 erläutert die Grundlagen natürlicher menschlicher Navigation. Diese Grundlagen werden benötigt, um herauszufinden, worauf ein Navigationssystem für Fußgänger basieren soll, um von den Benutzern als einfach benutzbar betrachtet zu werden. Danach folgt

die Einführung des Begriffs „Landmarken“. Es werden Eigenschaften und die Bedeutung von Landmarken für die Navigation präsentiert. Weiter wird kurz auf die Linguistische Beschreibung des Weges eingegangen. Es werden außerdem bereits bekannte vergleichbare Projekte erläutert.

Im Kapitel 3 werden anhand eines Beispielszenarios die grundlegenden funktionalen sowie nicht funktionalen Anforderungen an ein einfach benutzbares Fußgänger-Navigationssystem ermittelt. Diese Anforderungen werden in den Unterkapiteln 3.3 und 3.4 festgehalten. Eine besondere Stellung haben Aspekte der Ergonomie und ihre Anforderungen an die Fußgängerführung bekommen, die das Kapitel 3.2 beschreibt.

Das Kapitel 4 stellt den groben Entwurf des Systems dar. Dabei wird das Modellierungsschema für den Gebäudeplan präsentiert (Unterkapitel 4.5.3). Des Weiteren wird im Kapitel 4.4 ein Framework zur Entwicklung der Software-Anwendungen für mobile Geräte und seine Architektur betrachtet. Dieser wird bei der Implementierung des Software-Prototyps eingesetzt. Der Software-Prototyp wird für den im Kapitel 5 beschriebenen Test entwickelt und im Kapitel 4.5 präsentiert. Es wird zuerst auf die Hard- und Software eingegangen, die für die Entwicklung eines mobilen Fußgänger-Navigationssystem gebraucht wird. Danach wird die Realisierung der Karte bzw. des Gebäudeplans betrachtet.

Das Kapitel 5 beschreibt den Evaluierungstest, der in Rahmen dieser Arbeit durchgeführt wurde. Der Test basiert auf einem Szenario, das im Kapitel 5.1 erläutert wird. Der Test sollte die Richtigkeit der Aussagen in den Kapiteln 2 und 3 bestätigen.

Vor dem Schluss wird durch die methodische Abstraktion im Kapitel 6, weitere Möglichkeiten für den Einsatz von den landmarkenbasierten Navigationssystemen vorgestellt.

Das letzte Kapitel 7 fasst zusammen, was in dieser Arbeit realisiert wurde und welche Erkenntnisse dabei erfolgten. Des Weiteren wird auf die möglichen Erweiterungen der Untersuchungen dieser Arbeit eingegangen.

## 2. Grundlagen

Dieses Kapitel beschäftigt sich mit den Grundlagen der menschlichen Wegfindung und der räumlichen Wahrnehmung. Das Unterkapitel 2.1 beschreibt, wie sich Menschen in einer fremden Umgebung orientieren. Weiter wird im Kapitel 2.2 der Begriff „Landmarke“ betrachtet. Es wird vorgestellt, welche Bedeutung Landmarken für die natürliche menschliche Navigation haben. Im Kapitel 2.3 wird auf die linguistische Beschreibung des Weges eingegangen. Kapitel 2.4 betrachtet vergleichbare Projekten und Navigationssysteme. Das Kapitel 2.5 fasst die aus den vorherigen Kapiteln gesammelten Erkenntnisse zusammen.

### 2.1. Menschliches Orientieren

Woher weiß man, wo man sich befindet oder wohin man gehen will? Man erkennt die Orientierungspunkte rundherum. Z.B. die Gebäude die man sieht, kennt man. Es kann eine Kirche oder ein Restaurant sein. Die Orientierungspunkte zeigen an, wo man sich auf seiner mentalen Karte der Umgebung befindet. Man hat also eine Karte im Kopf, die aber häufig ungenau ist und nur über Teilbereiche ohne exakte Entfernungangaben verfügt.

Die mentale Karte erstellt ein Bild von unserer Umgebung und enthält dabei nicht nur das, was wir „tatsächlich sehen können, sondern auch vertraute Bereiche außerhalb unseres Blickfeldes, aus der Sicht unseres momentanen Standorts, also von der Seite aus“ [16]. Diese Angaben werden durch zwei dafür notwendige Systeme unterstützt; das Berechnungssystem, das uns sagt, wo wir sind, und das Orientierungssystem, das uns angibt, in welche Richtung wir schauen. Menschliche mentale Karte enthält auch Objekte, die man mit den Augen nicht direkt sehen kann, z.B. verdeckte Gebäude. Man kann auf dieser Karte „um die Ecke schauen“ oder „durch die Häuser sehen“, sie zeigt also das ganze vertraute Gebiet, das man in Erinnerung hat. Was der Mensch sehen wird, wenn er sich bewegt hat, „steht in einer bereits definierten Beziehung ("hinter") zu dem, was er jetzt sieht. Die relative Lage der Objekte erkennt man bevor dem Auge spezifische Informationen über sie verfügbar werden“ [23].

Die mentale Karte wird sofort im Gehirn angelegt, wenn man eine neue Umgebung betritt, ohne dass man dessen bewusst ist. Man legt Bilder von Orientierungspunkten in seinem Gehirn an. Dabei wird nicht nur das Aussehen der Objekte gespeichert, sondern ihre genaue

Position und Ausrichtung. Das ist sehr wichtig, da man auch gleiche Orientierungspunkte voneinander unterscheiden kann, wenn ihre Lage verschieden ist. Man kann dann diese Punkte für die Navigation nutzen. Nach Kevin Lynch entsteht das Bild der Umgebung durch einen Prozess, der zwischen dem Beobachter und seiner Umgebung stattfindet. „Die Umgebung bietet Unterscheidungen und Beziehungen, und der Beobachter wählt und fügt mit großer Anpassungsfähigkeit zusammen und gibt dem, was er sieht, eine Bedeutung“ [17]

Eine mentale Karte kann sehr lange im Gedächtnis erhalten bleiben. Wenn man aber sehr lange in einer Umgebung nicht gewesen war, erlöscht das genaue Bild dieser Umgebung. Man kann also zwischen zwei Arten von mentalen Karten unterscheiden [16]:

1. Die *aktive* mentale Karte, die immer präsent ist und die man benutzt, um einen Ort zu beschreiben.
1. Die *passive* mentale Karte, die wir noch im Gedächtnis haben und die Orientierungspunkte zwar enthält, die wir aber nur dann erkennen, wenn wir sie sehen.

Aber nicht nur durch ihre Präsenz im Gedächtnis unterscheiden sich mentale Karten. Wenn Menschen einen Weg beschreiben, nutzen sie eine ichbezogene(egozentrische) mentale Karte. Sie zeigt die Objekte auf dem Weg und deren Position aus der Perspektive der Person. Wenn man aber Objekte und ihre Lage relativ zu den anderen Objekten betrachtet, benutzt man eine allozentrische Karte. Diese Karte wird auch angewendet, wenn man ein Bild von einer bekannten Gegend anschaut. Die Objekte auf dem Bild erkennt man dank der allozentrischen mentalen Karte. Diese Unterschiede kann man für die verschiedenen Aspekte der Navigation nutzen. Graphische, Audio- oder Text-Beschreibung der Route sollen aus der egozentrischen Perspektive gemacht werden, da Menschen allozentrische Beschreibungen bei der Wegfindung unzureichend finden [7]. Jedoch könnte sie als grobe Übersicht der Gegend eingesetzt werden.

Man nutzt nicht eine und dieselbe Karte für den Hin- und Rückweg. Da auf dem Rückweg die Orientierungspunkte, die man gut kennt, aus einer ganz anderen Perspektive und in umgekehrter Reihenfolge erscheinen, hat man immer zwei Karten für eine Route, eine jedoch für den Hinweg, eine andere für den Rückweg. Das ist aber nicht alles. Im Outdoor-Bereich braucht man extra Karten für verschiedene Tageszeiten. In der Nacht ist die Beleuchtung anders als am hellen Tag und die Objekte sehen anders aus. Die Orientierungspunkte, die man am Tag genutzt hat, können im Dunkeln schwer erkennbar sein oder sie sind gar nicht zu sehen. Dagegen können andere Objekte, die man am Tag nicht bemerkt hat, in der Nacht durch eine besondere Beleuchtung zu Orientierungspunkten werden.

Im Outdoor-Bereich muss die mentale Karte auch die saisonalen Veränderungen berücksichtigen. Die Orientierungspunkte, die im Winter gut sichtbar sind, können im Sommer durch die Belaubung verdeckt werden. Oder ist ein prächtiger Baum im Sommer sehr gut bemerkbar. Dabei lässt er sich im Winter nicht deutlich von den anderen Bäumen unterscheiden. Man

braucht also eine Winter- und eine Sommerkarte. Und diese wiederum jeweils für die Nacht und für den Tag.

Die menschliche mentale Karte ist sehr gut auf die Außenwelt abgestimmt. Auf dieser Karte kann man die Orientierungspunkte von der Seite in Augenhöhe und von dem momentanen Standort aus sehen. Dadurch sind sie leicht erkennbar. Das bedeutet, dass die mentale Karte sich ändert, sobald man seine Position und Richtung ändert. Die Karte passt sich der neuen Perspektive an.

## 2.2. Landmarken

In der oberen Beschreibung der natürlichen menschlichen Orientierung und der mentalen Karte wird ständig der Begriff „Orientierungspunkt“ verwendet. Orientierungspunkte oder auch Landmarken genannt, können eindeutig identifiziert werden, da sie sich von den anderen Gegebenheiten in ihrer Umgebung deutlich unterscheiden. Für menschliche Orientierung sind Landmarken sehr wichtig, weil sie “einen fixen Punkt auf der mentalen Karte eines bekannten Gebietes angeben“ [16]. Einerseits hilft die menschliche mentale Karte, Landmarken zu identifizieren, andererseits gibt eine richtig identifizierte Landmarke den Standort einer Person auf ihrer mentalen Karte an und richtet diese danach aus.

Kevin Lynch hat auch eine andere Definition für Landmarken aus der Sicht der Elemente einer Stadt [17]. Als Elemente des Stadtbildes unterscheidet er unter anderem *Brennpunkte* sowie *Merk- und Wahrzeichen*.

- Brennpunkte (z.B. Wochenmarkt) - sind die strategischen (Knoten)Punkte einer Stadt, die einem Beobachter zugänglich sind; sie sind intensiv genutzte Zentralpunkte, Ziel und Ausgangspunkt seiner Wanderungen. Knotenpunkte sind ganz besonders wichtig: hier müssen Entscheidungen getroffen werden. Die Verkehrsteilnehmer sind an solchen Stellen besonders aufmerksam und nehmen die in der Nähe befindlichen Elemente mit mehr als der üblichen Deutlichkeit wahr. Theoretisch sind sogar ganz gewöhnliche Straßenkreuzungen Brennpunkte, aber im Allgemeinen fallen sie nicht genügend auf, um für mehr als eben rein zufällige Kreuzungen gehalten zu werden [17]. Viele Brennpunkte sind situationsabhängig. Sie entstehen in bestimmten Situationen und sind nur dann interessant. Man muss also die Umstände (z.B. Wochenende, Feiertag) berücksichtigen, wenn man solche Brennpunkte als Landmarken bezeichnen will.
- Merk- und Wahrzeichen(Landmarken) sind äußere Merkmale. Gewöhnlich sind das ganz einfache Objekte wie Gebäude, Schilder oder Warenhäuser. Diese Objekte werden aber von den anderen in ihrer Umgebung abgesondert. Ihr Hauptcharakteristikum

ist Einmaligkeit bzw. „in diesem Zusammenhang einzigartiger oder merkwürdiger Anblick“ [17]. Durch räumliches Hervorragen wird ganz besondere Aufmerksamkeit erzwungen.

Es gibt Merkzeichen, die man aus der Ferne sieht. Sie wirken typisch von verschiedenen Standpunkten aus und in verschiedenen Abständen. Zu solchen Merkzeichen gehören Türme, goldene Kuppeln, ansehnliche Hügel. Ferne Merkzeichen werden nur zur sehr allgemeinen Richtungsorientierung benutzt. Sie geben die allgemeine Richtung vor oder bestätigen sie. Es gibt aber auch lokale Merkzeichen, die nur in einem bestimmten Umkreis sichtbar sind. Darunter fallen verschiedene Schilder, Plakate, Kaufhausfronten. Landmarken werden auch durch ihre Beziehung zu der Route unterschieden. Dann sind ferne Landmarken, solche, die nicht direkt auf der Route liegen. Die Landmarken, die sich in der unmittelbaren Nähe von Entscheidungspunkten mit oder ohne Richtungswechsel befinden, sind so genannte Knotenlandmarken [38]. Sie sind sehr wichtig für die Orientierung, da sie als Entscheidungshilfe dienen.

Lynch sagt, dass ein Merkzeichen nicht immer groß sein muss. Es kann ein Dom sein, aber auch ein Türgriff. Die Lage ist dabei entscheidend, wenn ein Objekt als ein Merk- oder Wahrzeichen bezeichnet werden soll. Z.B. sind alle Unterbrechungen im Verkehrsfluss - Knoten oder Entscheidungspunkte - Orte intensiver Wahrnehmung. So wirken an solchen Punkten ganz gewöhnliche Gebäude auffällender als sonst und bleiben deutlich im Gedächtnis haften, während an sich bemerkbare Bauwerke in einem langen geraden Straßenabschnitt verborgen bleiben. Ein Merkzeichen ist besonders wertvoll, wenn es die Blickrichtung des Beobachters nachprüfen lässt.



Abbildung 2.1.: Eine Landmarke Quelle: [17]

Nicht nur Lage und einmaliges Aussehen machen Objekte zu Merk- oder Wahrzeichen. Die Attraktivität einer Landmarke wächst, wenn sie gleichzeitig mit „einer Konzentration von Erinnerungen“ [17] verbunden ist. Wenn ein in seiner Form ungewöhnliches Gebäude gleichzeitig mit historischen Ereignissen verbunden ist, dann steigt seine Bedeutung als Merkzeichen,

da es sich einfacher im Gehirn einprägt. Auch wenn der Name eines Objekts allgemein bekannt ist, dann hat allein die Benennung dazu beigetragen, dass das Objekt ein Merkzeichen wird.

Wie schon erwähnt wurde, bestimmt die Struktur des Weges, welche Landmarken für die Navigation einbezogen werden sollen. Landmarken werden typischerweise an vier kritischen Stellen benutzt [32]:

1. am Anfang der Route
2. am Ende der Route
3. bei dem Wechsel der Richtung
4. an den Kreuzungen, wo man sich leicht verirren kann

### 2.2.1. Eigenschaften von Landmarken

Ein Objekt, sei es ein Gebäude, ein Denkmal oder ein Schild, wird zu einer Landmarke durch bestimmte Eigenschaften, die dieses Objekt von den anderen hervorheben. Landmarken können anhand ihrer individuellen Eigenschaften in visual, structural und cognitive unterteilt werden. Visual-Landmarken sind visuell auffällige Gegenstände und Einrichtungen wie z.B. ein großes Werbeplakat oder ein buntes Schild. Strukturelle-Landmarken sind an erster Stelle Bauwerke besonderer Bedeutung: Museen, Denkmäler, Brücken. Objekte, die leicht zu erkennen sind und allen bekannt sind, gehören in die Kategorie kognitiver Objekte. Beispiel ist: McDonalds. Einige Quellen beschreiben mehr als 40 Unterkategorien für Landmarken [19]. In [29] werden folgende Eigenschaften als Maße für das Hervorheben der Objekte als Landmarken genannt:

- **Visuelles Hervorheben:** Landmarken werden als visuell anziehend qualifiziert, wenn sie bestimmte visuelle Eigenschaften besitzen, wie starken Kontrast zu ihrer Umgebung, oder besondere Positionierung (z.B. am Hügel). Es gibt vier wichtige Merkmale, nach denen Landmarken als visuell attraktiv bestimmt werden.
  - **Größe der Fassade oder Fassadenumfang:** Fassadenumfang ist eine wichtige Eigenschaft für den Vergleich eines Objektes mit seinen Nachbarn. Objekte, die wesentlich größer oder kleiner sind als ihr Umfeld, werden von Personen leichter gemerkt. Im einfachsten Fall, wenn das Gebäude eine rechteckige Form hat, wird der Fassadenumfang durch Multiplikation der Höhe und Breite bestimmt.

- **Form:** Die visuelle Anziehungskraft eines Objektes wird auch durch seine Form bestimmt. Unorthodoxe Formen inmitten gewöhnlicher Bauten fallen schnell auf. Das Maß, inwieweit eine Form ungewöhnlich ist, kann durch verschiedene Parameter bestimmt werden. Das kann Verhältnis Höhe zu Breite sein. Wolkenkratzer sind z.B. sehr lang und relativ zu ihrer Länge nicht breit. Deswegen sind sie für die Fußgänger sehr auffällig. Man darf aber nicht vergessen, dass Objekte sich von ihrer Umgebung unterscheiden sollen, wenn sie als Landmarken bezeichnet werden. Ein Wolkenkratzer unter vielen ist schwer zu differenzieren.
- **Farbe:** Ein Objekt kann durch seine Farbe hervorstechen, wie z.B. ein rotes Haus zwischen grauen Gebäuden. Farbe ist aber eine Eigenschaft, die schwer zu messen und zu vergleichen ist. Außerdem ist die Wahrnehmung der Farbe von der Beleuchtung und vom Betrachter (mögliche Farbenblindheit) abhängig. So scheint eine Farbe in der Nacht anders zu sein als am Tag. Deswegen ist Farbe im Unterschied zu Fassadenumfang und Form als Parameter schwer zu messen.
- **Sichtbarkeit:** Ein weiteres Maß, das für die Bestimmung von Landmarken eine wichtige Rolle spielt, ist die Lage des Objekts. Hier betrachtet man aber seine Sichtbarkeit für den Navigierenden. Ein Objekt soll aus der Benutzerperspektive sehr gut sichtbar sein und nicht durch andere Objekte verdeckt werden. Das ist besonders in Outdoor-Umgebung relevant.

Auch andere visuelle Eigenschaften wie Textur oder allgemeiner Zustand eines Objekts können das Hervorstechen dieses Objekts von den anderen beeinflussen. Es fehlt aber an der Formalität und einem objektiven Maß, um diese Eigenschaften zu vergleichen. Die Textur ist für eine Person sowie in den Datensammlungen schwer zu bestimmen. Der Zustand eines Objektes wird anhand seines Alters und seiner Sauberkeit gemessen. Das Alter ist in der realen Welt schwer zu erraten, da ein Gebäude durch die Renovierungsarbeiten viel jünger aussehen mag, als es wirklich ist. Sauberkeit ist kein objektiver Parameter und ist auch schwer formal zu spezifizieren.

- **Semantische oder kognitive Attraktivität:** basiert auf der allgemeinen Bedeutung des Objektes. Es wird zwischen kulturellen und historischen Objekten und expliziten Merkzeichen unterschieden.
  - **Historische oder kulturelle Bedeutung** kann semantische Attraktivität von Objekten bestimmen. Solche Objekte sind Museen, Paläste, Kirchen. Das Rathaus in Hamburg ist z.B. berühmt für seinen Architekturstil. Diese Eigenschaft ist für das Identifizieren von Landmarken wichtig, da die Informationen über sie den Datenbanken und Sammlungen historischer und kulturelle Objekte entnommen werden können.

- **Explizite Landmarken:** Markierungen wie Schilder an Gebäuden, spezifizieren explizit Bedeutung dieser für Fußgänger. Z.B. Schild „Double Coffee“ sagt uns ohne weiteren, dass hier ein Cafe namens „Double Coffee“ befindet. Straßenschilder gehören ebenso zu dieser Kategorie von Landmarken. Die Entwicklung von Navigationssystemen für Fußgänger kann die Informationen über solche Objekte einfach aus den „gelben Seiten“ entnehmen.
- **Subjektive Landmarken:** Objekte können zur subjektiven Landmarken werden, wenn sie mit bedeutenden persönlichen Ereignissen assoziiert werden. Zum Beispiel, wenn man den Lebenspartner in einem Lokal kennenlernte, gewinnt dieses Lokal eine besondere Bedeutung für die Betroffenen. Solche Landmarken sind aber nicht in den Datenbanken zu finden.
- **Strukturelle oder räumliche Eigenschaften:** Wenn es keine Objekte gibt, die sich visuell deutlich von den anderen unterscheiden, wendet man sich an die räumliche Position der Objekte und untersucht diese [38]. Räumliche Eigenschaften sind Routenabhängig, trotzdem können sie als konstant angesehen werden und als Eigenschaft für Objekte gespeichert werden. Eine Landmarke ist strukturell attraktiv, wenn sie eine wichtige Rolle für die Route spielt, oder eine bedeutende Lage hat. Beispiele sind Straßenkreuzungen oder Plätze. Nach Lynch [17] sind Knoten, Grenzlinien und Bereiche Elemente, die die Struktur einer Stadt bilden. Diese Elemente sind sehr gut wahrnehmbar und Objekte, die sich in der Nähe dieser Elemente befinden, könnten abhängig von ihrer individuellen räumlichen Eigenschaften von den anderen Elementen hervorstechen und so zu Landmarken werden.
  - **Knotenpunkte oder Brennpunkte:** Knotenpunkte sind strategische Punkte einer Stadt. Sie sind Zentralpunkte, die intensiv genutzt werden. Dazu gehören Ziel und Ausgangspunkt einer Route. Für Autofahrer sind Verkehrsunterbrechungen, Straßenkreuzungen Knotenpunkte. Für Fußgänger können irgendwelche Plätze, Märkte oder größere Bahnstationen als Knoten- bzw. Brennpunkte dienen. Knotenpunkte sind für die Navigation sehr wichtig, da hier Entscheidungen getroffen werden müssen. Die Verkehrsteilnehmer sind an solchen Stellen besonders aufmerksam und nehmen die in der Nähe befindlichen Elemente viel intensiver wahr. Die zentrale strukturelle Eigenschaft eines Knotenpunktes ist der Grad seiner Verbindungsfähigkeit. Verbindungsfähigkeit wird durch die Anzahl und die Qualität der eingehenden und ausgehenden Kanten/Wege bestimmt.
  - **Grenzlinien:** Nach Lynch [17] sind Grenzlinien Linearelemente, die nicht als Straßen dienen, sondern sind gewöhnlich - aber nicht immer - die Grenzen zwischen zwei Gebieten, lineare Unterbrechungen des Zusammenhangs; Küsten, Eisenbahnstrecken, Baugebietsränder, Mauern. Sie dienen als seitliche Bezugslinien bzw. Leitmarken. Für viele Fußgänger sind diese „Randelemente“ wichtig

als Elemente der Gliederung. Besonders wichtig sind sie da, wo sie Bezirke zusammenfassen, z.B.: Wasser oder Mauer als Umriss einer Stadt.

Etwas anders wird der Begriff der „Grenzlinien“ bei Raubal [29] definiert. Er sagt, dass der Maß der Wahrnehmung von Grenzlinien mit dem Aufwand verbunden ist, den man bei der Durchquerung dieses Elements aufbringen muss. Raubal nimmt an, dass die Grenzlinie umso bedeutender ist, je größer ihr Durchgangswiderstand ist.

Solche Barrieren gestalten wichtige Umriss des Stadtbildes: das Routennetz weist durch die großen Grenzlinien eingeschlossene Zellen und Zellen mit kleinen Abständen zwischen den gegenüberliegenden Grenzlinien auf (Abbildung 2.2). Solche Maße wie das Produkt der Zellengrößen und ihre Form könnten strukturelle Attraktivität der Objekte als Landmarken in diesen Zellen bestimmen.

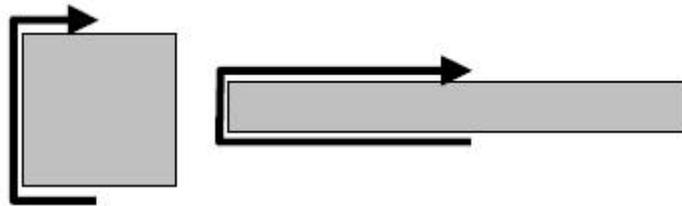


Abbildung 2.2.: Beide Zellen haben gleiche Größe, aber die rechte Zelle ist länger und bildet damit größere Barriere; Quelle: [29]

Alle oben genannten Eigenschaften kann man situationsabhängig messen und kombinieren [39].

### 2.2.2. Bedeutung von Landmarken für die Navigation

Landmarken sind für die menschliche Navigation sehr wichtig. Sie geben einen fixen Punkt auf der mentalen Karte eines bekannten Gebietes an [16]. Die übliche Methode, in der sich Personen durch die Stadt bewegen, besteht darin, dass sie einer Reihenfolge von Merkzeichen folgen, in der ein Detail jeweils auf das nächste vorbereitet und in der gewisse Schlüsselpunkte bestimmte Reaktionen auslösen. „Es gab in diesen Folgen „auslösende“ Hinweise dort, wo Richtungsänderungen vorgenommen werden mussten, und „versichernde“ Zeichen, die den Fußgänger in der getroffenen Entscheidung bestätigen“ [17].

In [36] wird festgestellt, dass Wegbeschreibungen mit Hilfe von Textanweisungen und in Kartenform in ihrer Struktur identisch sind: Start- und Zielpunkte sind Landmarken.

Eine gute Navigationsroute entsteht, wenn der Nutzwert eines Landmarks eng mit dem Nutzwert des nächsten Landmarks verbunden ist [7]. Die Wahl der Landmarken und die Navigationsstaktik sind stark aufgabenabhängig. Eine Route kann als Abfolge von Objekten oder Ereignissen betrachtet werden. Dabei kann man zwischen Bildsequenzen, Entscheidungsabfolgen, Blicksequenzen und Bewegungssequenzen unterscheiden. Folglich kann der Weg als Bewegungssequenz angesehen werden, und in Betrachtung des Kontextes bestimmter Orte auch als Bildsequenz. Entscheidungsabfolgen können aus der Abfolgen von „links-“ und „rechts-“ Anweisungen bestehen. Landmarken sind dabei eine notwendige Bedingung für die Gestaltung von Entscheidungs- und Bildabfolgen [37].

Wegfindung umfasst die Bewegung zu bekannten sowie unbekanntem Zielen und die Erforschung unbekannter Gebiete. Landmarken sind hervorstechende Hinweise, die Informationen über den Ort und die aktuelle Position der Person liefern. Menschen benutzen Landmarken, um die Wege zwischen den Orten ausfindig zu machen und um diese Informationen dann anzuwenden oder weiter zu übermitteln. Wie schon gesagt wurde, bilden Landmarken eine Abfolge von Entscheidungspunkten entlang der Route. Wie wichtig die menschliche Vorstellung des Raumes als eine Menge von bekannten Landmarken für die Wegfindung ist, wird durch die Wahrnehmung und Verhalten bestätigt [7]. Landmarken tragen zum Organisieren des Raumes in der kognitiven Wahrnehmung bei, indem die als Hinweise und Referenzpunkte in der Umgebung agieren.

Man könnte denken, dass die Landmarken, die man für die Navigation nutzt, solche sind, die man sich am besten merken kann und diese Information bei der Entwicklung des Navigationssystems nutzen. Das stimmt aber nicht. In [7] wird beschrieben, dass wenn Menschen versuchen Landmarken mittels Bildern zu identifizieren und zu lokalisieren, sie tendieren dazu, *die* Landmarken viel leichter zu finden und wieder zu erkennen, die von einer Freifläche umgeben sind. Sie können viel schwieriger die Landmarken identifizieren, die ungewöhnlich, klein, zu normal oder nicht eigenständig (z.B. ein Teil vom ganzen Gebäude) sind. Das kann Unterschiede zwischen Landmarken für die Navigation vor Ort und für die Navigation mittels Gedächtnis erklären. Bekannte Objekte unterstützen die Navigation mittels textlicher Beschreibung in einer fremden Umgebung. Diese Objekte sind aber schwieriger aus dem Gedächtnis zu rufen als ungewöhnliche Objekte, wenn man sich in einem vertrauten Ort befindet. Auch ist es viel schwieriger für Menschen, Landmarken in dicht bebauten Gebieten mit gleichen Wegen und Gebäuden zu finden und wieder zu erkennen.

Auch [6] haben festgestellt, dass die herrschende Strategie bei der natürlichen menschlichen Wegfindung die Wiedererkennung von Landmarken- und Richtungsabfolgen und nicht der Aufbau der mentalen Karte ist; besonders in einer völlig fremden Umgebung.

Um für die Navigation geeignet zu sein, sollen Landmarken visuell eindeutig und unverwechselbar sein. Außerdem sollen Sie klar definiert und sichtbar sein. Klar definiert heißt, dass man die Landmarken mit sichtbaren Namen über diese auch referenzieren soll [19]. So

sollte ein Geschäft, wenn es allgemein bekannt ist, per Namen genannt werden wie z.B. „Aldi“, statt „Supermarkt“. Die Landmarken sollen auch gut sichtbar sein, das System soll diese immer aus der Benutzerperspektive betrachten. Während der Navigation sollten richtige Entscheidungen durch Landmarken bestätigt werden, z.B. auch wenn man lange nur geradeaus geht, sollte das System es bestätigen („Am McDonalds vorbei weiter laufen“). Landmarken sollen in der unmittelbaren Nähe zu der vorgesehenen Route und beständig im Bezug zur Abweichungen von der Route sein.

Die Art, wie Landmarken verwendet werden, ist von den verschiedenen Einflussfaktoren sowie der Situation abhängig. Solche Faktoren können Bildungsstand und Alter des Nutzers sein (Umgang mit der Karte), oder der Zeitpunkt für den die Wegbeschreibung gelten soll (Tag/Nacht- Beleuchtung, Winter/Sommer wegen Bepflanzung), der Zweck der Wegbeschreibung (Tourist, oder schnellstes Erreichen des Ziels, Wanderung) sowie die Art der Fortbewegung (besonders für die Rollstuhlfahrer relevant).

Einige Fakten, die aus verschiedenen Untersuchungen abgeleitet sind, fassen den Begriff „Landmarke“ einmal zusammen:

- Jedes Objekt, das von den anderen in der Umgebung hervorsteht, kann eine Landmarke sein.
- Im bestimmten Kontext können Straßenkreuzungen zu Landmarken werden.
- Landmarken helfen einen bestimmten Weg zu finden
- Landmarken sind bekannte Objekte oder Begriffe, die man gut wiedererkennen kann.
- Landmarken strukturieren die Umgebung, sie sind so genannte Schlüsselpunkte.
- Landmarken werden benutzt, um die Route verbal oder graphisch zu übermitteln.
- Landmarken werden mit unterschiedlichem Anteil in die Weganweisungen integriert. Die größere Anzahl von Landmarken wird an den Start-, Ziel- und Entscheidungspunkten benötigt.
- An den Straßenkreuzungen sind Landmarken besonders notwendig, wenn der Richtungswechsel stattfinden soll.
- Für die Wegfindung sind Landmarken besser geeignet als Straßenschilder.

### 2.3. Linguistische Beschreibung des Weges

Zu der Fußgänger-Navigation gehört auch verbale Beschreibung der Route. Obwohl es hier nicht erforderlich ist, den Weg mit genauen Angaben der Entfernungen und Wendungen zu beschreiben, wie es bei Auto-Navigation üblich ist, muss der Fußgänger sich in der Umgebung orientieren können. So soll die Position einer Landmarke so übermittelt werden, dass die navigierende Person diese leicht finden kann.

Die eigentliche Richtung, wo sich ein Objekt bezüglich der Route befindet, kann durch eine der 16 Richtungen (Abbildung 2.3) definiert und als Position dieses Objektes benutzt werden. Durch diese 16 Richtungen kann man sehr genau die Lage eines Landmarks beschreiben. Wenn man die Lage des Objektes relativ zu der Position der Person betrachtet, lässt sich die Uhr-Metapher einsetzen. Bei der Uhr-Metapher gibt es zwar nur 12 Richtungen, sie ist aber allgemein verständlich.

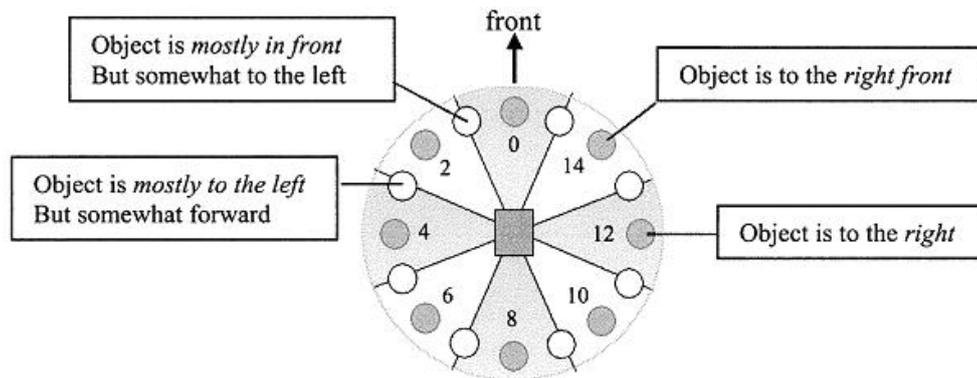


Abbildung 2.3.: Sechzehn Richtungen bestimmen die Position eines Objektes; Quelle: [32]

Eine verbale Beschreibung der Route umfasst aber nicht alles, was man im Raumkontext für die Navigation nutzen kann. Homogene Umgebungen wie Stadtränder oder Vororte sind schwierig zu beschreiben, weil die Objekte dort sich schwach von einander unterscheiden lassen. In solchen Fällen bleibt nichts anderes als die Straßennamen zu verwenden.

Wenn man Landmarken beschreibt, nutzt man ihre Namen oder den Zweck, wofür die Landmarken dienen. Stadtteile, Straßen und bekannte Geschäfte werden mit Namen genannt. Landmarken, die nach ihrem Zweck beschrieben werden, umfassen Gebäude (z.B. Bibliothek), Plätze (z.B. Golfplatz oder Marktplatz) und andere besondere Objekte (z.B. Wasserturm, Fernsehturm).

## 2.4. Vergleichbare Arbeiten

In diesem Kapitel werden bestehende Navigationssysteme für Fußgänger und vergleichbare Ansätze vorgestellt. Diese Vorstellung sollte helfen, anhand bestehender Erfahrungen Schlußfolgerungen zu ziehen und sie in dieser Arbeit zu verwenden.

### 2.4.1. Active Landmarks

*Active Landmarks* ist ein Projekt vom Institut für Geoinformation und Kartographie an der Technischen Universität Wien. Die Idee dabei ist eine Umkehrung des bisherigen Ansatzes (Lokalisierung des mobilen Gerätes) und einer Positionsangabe der Landmarke selber. Der Fußgänger wird also, sobald er in die Nähe einer Landmarke gelangt, von dieser quasi eingefangen und erhält nähere Informationen (Bestätigung seiner Wegwahl/Positionierung, aktuelle Informationen). Aktive Landmarken sollen also in der Lage sein mit dem Navigationsgerät des Benutzers eine spontane Funkverbindung über eine Luftschnittstelle aufzubauen und sich bei diesem zu identifizieren [8].

Die Objekte, die als Landmarken bezeichnet werden, werden mit einem Nahbereichsender (Reichweite ca. 10-50m) ausgestattet. Das Endgerät, soll einen entsprechenden Empfänger enthalten, um die Daten auszuwerten, die eine aktive Landmarke sendet. Zum Aufbau einer Spontanverbindung kann z.B. Bluetooth oder Wireless LAN verwendet werden.

Das mobile Gerät des Benutzers sucht ständig nach einer aktiven Landmarke, das heißt, es wird versucht, eine Verbindung zwischen dem Gerät und der nächstliegenden aktiven Landmarke aufzubauen.

Abbildung ?? zeigt den Informationsfluß einer aktiven Verbindung. Zunächst findet das mobile Gerät eine aktive Landmarke (1). Dann wird mit den Koordinaten, die bei der aktiven Landmarke gespeichert sind, mit Hilfe eines zentralen Servers eine Detailkarte erzeugt bzw. eine der aktiven Landmarke entsprechende Informationsseite erstellt (2). Diese Karte bzw. Seite wird in einem weiteren Schritt zurück an die aktive Landmarke gegeben (3) und zuletzt von dort über die Luftschnittstelle dem Endgerät des Benutzers geschickt (4) [8]. Der Benutzer erhält nun eine Detailkarte passend zu der aktiven Landmarke auf seinem Gerät. Damit wird das Erreichen des Sendebereiches einer aktiven Landmarke signalisiert. Der Benutzer findet für seine Wegwahl eine Bestätigung und erhält gleichzeitig eine Hilfe für den weiteren Weg.

Um eine sinnvolle Navigation zu ermöglichen, ist eine große Anzahl von aktiven Landmarken notwendig. Daher ist die Umsetzung dieses Konzepts aus heutiger Sicht noch problematisch, da es zu hohen Errichtungskosten führen würde. Unter Umständen können sich auch

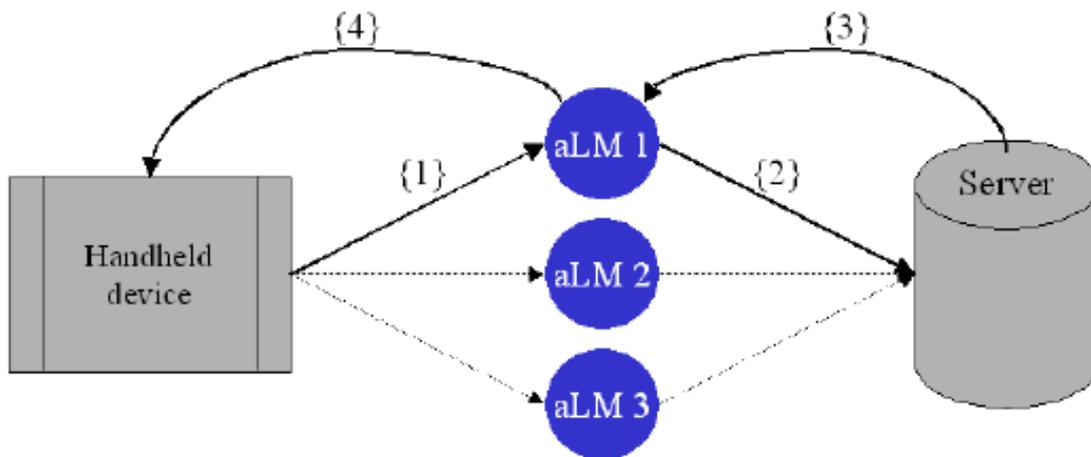


Abbildung 2.4.: Informationsfluss einer aktiven Verbindung; Quelle: [8]

die Sendebereiche überschneiden. Das Konzept wird in einem weiteren Projekt „UCPnavi“ weiterentwickelt.

### 2.4.2. Bum Bag Navigator

Der Bum Bag Navigator ist ein Projekt des Informatik-Departments an der Universität Saarland. Es ist ein Navigationssystem, das zwei PDAs nutzt, um die Informationen zu übermitteln [5]. Die Ressourcen sind auf den beiden PDAs verteilt, so dass die PDA-Geräte die aufkommenden Aufgaben gemeinsam lösen können. Die Kommunikation zwischen den beiden Geräten passiert über WLAN. Das ganze System basiert auf einer Client-Server-Architektur. Ein PDA stellt den Server dar und der andere den Client. Ein PDA-Gerät, das den Client-Teil spielt, kann als eigenständiges Navigationssystem agieren. Es stellt die Komponente für den Benutzer-Input dar.

Der andere PDA wird dagegen als ein Server für die VGA-Karte benutzt, um den clip-on-Display zu betreiben, das auf einer Brille befestigt ist. Weil das System mit zwei Bildschirmen ausgestattet ist, kann es ohne den Einsatz von Händen benutzt werden.

Getragen wird die Konstruktion in einer Gürteltasche, daher kommt auch der Name „Bum Bag Navigator“. Im oberen Abteil der Gürteltasche wird der Client-PDA angesiedelt, so dass man durch eine Öffnung auf seinen Bildschirm schauen kann. Damit hat man eine Sicht auf die Anwendungsanzeige. Im hinteren Teil der Tasche liegt der Server-PDA und weitere Hardware zum Betreiben vom HMD (head mounted display).

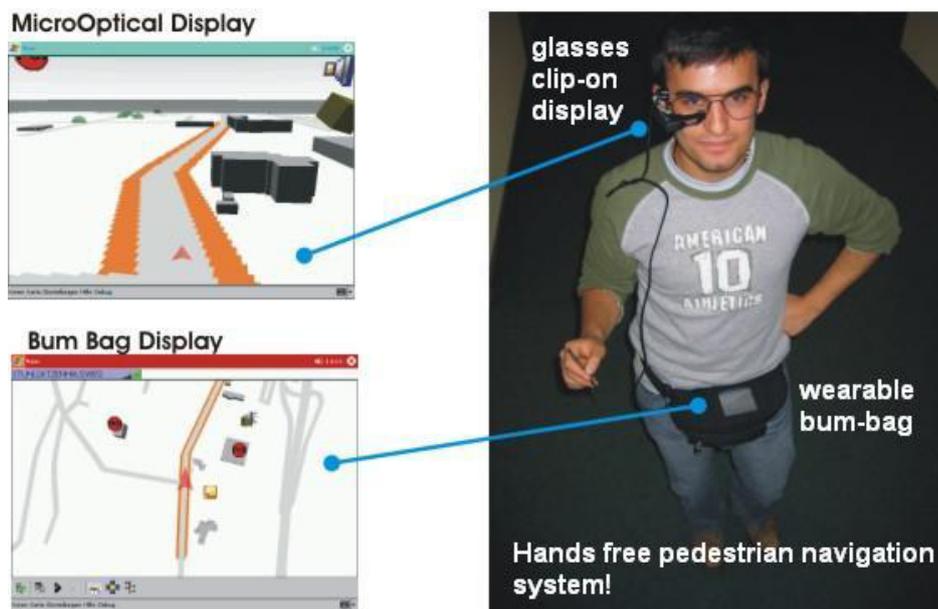


Abbildung 2.5.: Bum Bag Navigator; Quelle: [5]

Der Bum Bag Navigator bietet fünf verschiedene Sicht-Modi an, die eine bequeme Navigation unterstützen sollen [6].

- **Extra-Sich (x-view):** in diesem Modus erweitert das HM-Display das Bild des „statischen“ Displays vom Client-PDA vertikal oder horizontal (siehe Abbildung 2.6).

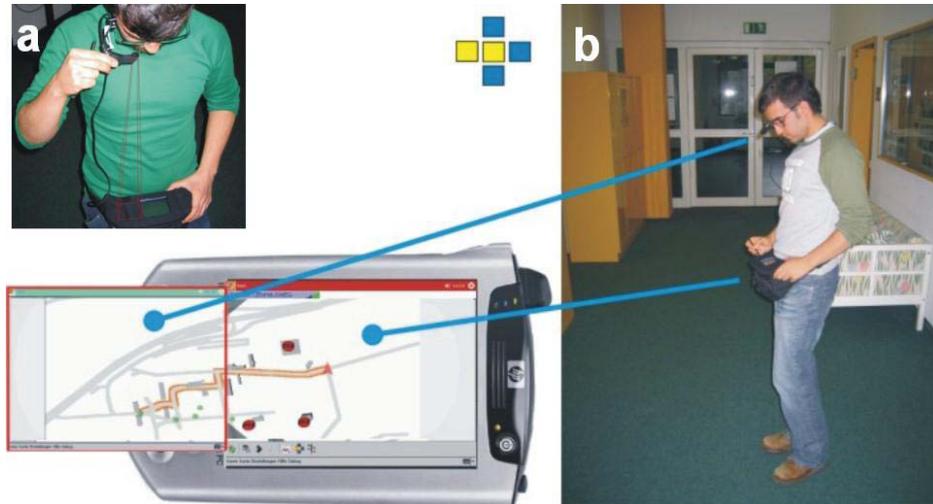


Abbildung 2.6.: Extra View; Quelle: [4]

- **Abstrakte Sicht (a-view):** liefert eine abstrakte Weg-Beschreibung (siehe Abbildung 2.7). Dabei werden die Informationen von der Umgebung dynamisch gefiltert. Diese Sicht wird auf dem clip-on-Display präsentiert.

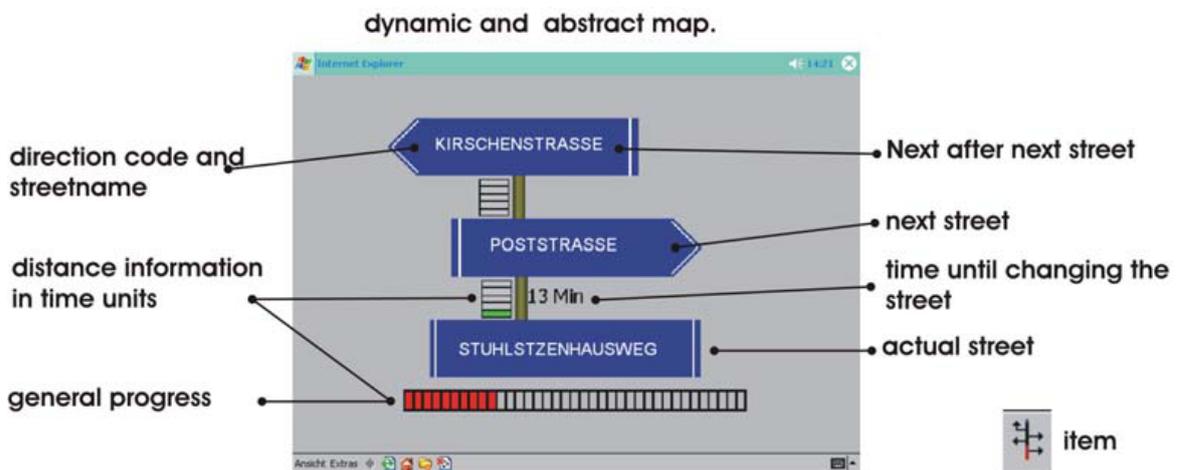


Abbildung 2.7.: Abstract View; Quelle: [4]

- **Übersicht-Modus (p-view):** zeigt die ganze Route von der aktuellen Position bis zum Ziel in einem Durchlauf. Eine virtuelle Kamera „fliegt“ über die ganze Strecke und zeigt sie aus der Vogel-Perspektive an.

- **Fotos** : Es wird die Möglichkeit angeboten, per Hand annotierte Fotos von Kreuzungen abzubilden, um die Richtungsabzweigung zu verdeutlichen.
- **Neigung-Kontrolle**: merkt die Änderungen der Neigung vom Client-PDA und bildet es entsprechend auf dem clip-on-Display. Man kann den Client-PDA als ein Lenkrad nutzen, um durch ein virtuelles Bild zu laufen.

### 2.4.3. Navigon 7100

Navigon von [3] bietet eine fotorealistische 3D-Ansicht der Umgebungskarte. Die Software ist ein Produkt von der Navigon AG und ist auf PDAs und Handys lauffähig. Alle Straßen sind in der 3D-Ansicht weiß dargestellt, dementsprechend wird die Route kontrastreich gelb angezeigt. Der aktuelle Standort wird durch ein tellerförmiges 3D-Objekt mit gelbem Pfeil angezeigt. Der Pfeil weist auf die Fahrtrichtung [25].



Abbildung 2.8.: Das Navigon 7100 in der Handy-Version (3D)

Die Software setzt eine mindest Taktfrequenz von 500MHz voraus und benötigt eine Arbeitsspeicher von 128MB-RAM. Für mobile Geräte, beispielsweise ein Smartphone wird zusätzlich ein Betriebssystem Windows vorausgesetzt.

Die Stadtumgebung wird in Grautönen und die Landschaft, beispielsweise Felder, Wiesen und Wälder, wird in Grüntönen dargestellt. Es werden auch wahlweise POIs angezeigt (z.B. Tankstellen). Eine Spuranzeige präsentiert vorausschauend die Straßenspur, die augenblicklich zu befahren ist, um damit den nächsten Routenhinweis nicht zu verpassen. Die Straße, auf der sich der Benutzer befindet, wird durch ihren Namen gekennzeichnet. Auf das Ziel wird mittels erwarteter Ankunftszeit bzw noch ausstehender Kilometer hingewiesen.

Die Software Mobile Navigator von Navigon kann von der Navigon-Homepage kostenlos auf Java-Handy heruntergeladen werden.

### 2.4.4. City Maps

City Maps ist ein Produkt von TeleAtlas [2]. Es ist eine Erweiterung von digitalen Karten dieses Herstellers. Dabei geht es um die 2D- und 3D-Darstellung von Stadtplänen. Die wichtigsten Städte Europas und Nordamerikas werden in den multidimensionalen Karten abgebildet.

Die 2D-Stadtpläne zeigen detailliert Gebäudegrundrisse, Stadtviertel einschließlich Geh- und Wasserwege sowie das Schienennetz.

Die 3D-Darstellung präsentiert die städtischen Objekte im drei-dimensionalen Format. Der Benutzer kann sich für eine der beiden Darstellungsmöglichkeiten entscheiden, die ihm entsprechende Orientierungshilfe bieten soll.



Abbildung 2.9.: Darstellung der Karte durch City Maps

Der wichtige Aspekt für die Fußgänger-Navigation sind die 3D-Landmarken (Abbildung 2.10). Die 3D-Landmarken umfassen Gebäude, Statuen oder andere Bauwerke, die einen hohen Bekanntheitsgrad sowohl bei der einheimischen Bevölkerung als auch bei Touristen aufweisen. Alle diese Landmarken sind realistisch dargestellt und können vergrößert und gedreht werden, so dass Nutzer ein besseres Bild der direkten Umgebung erhalten. Alle 3D-Landmarken haben Namen, anhand deren sie erkannt werden können und verfügen über einen exakt geo-kodierten Mittelpunkt.



Abbildung 2.10.: 3D-Darstellung von Landmarken am Beispiel

## 2.5. Zusammenfassung

Die natürliche menschliche Wegfindung setzt voraus, dass bestimmte Orientierungspunkte - Landmarken - in die Routenbeschreibung einbezogen werden. Als Landmarken („Merkzeichen“) werden jegliche Orientierungspunkte im Raum bezeichnet [9]. Eine andere Definition sagt, dass Landmarken „in Begriff und Wahrnehmung klar unterscheidbare Orte sind“ [22]. Auf Basis dieser Definition kann man sich bereits einige Navigationsinstruktionen vorstellen, wie z.B. „am Supermarkt nach links“ oder „vor der Bar nach rechts“.

Im Kontext der Wegsuche soll eine Landmarke folgende Eigenschaften aufweisen:

- eine spezielle visuelle Charakteristik (auffällige Abweichung zu Nachbarschaft)
- eine einzigartige Funktion oder Bedeutung oder
- eine zentrale oder hervorstechende Lage.

Dabei steigt die Bedeutung als Landmarke mit der Zunahme der zutreffenden Eigenschaften. Angemessene Landmarken sind demnach z.B. Bars/Kneipen, Restaurants (incl. Fast-Food-Ketten), Hotels und Banken.

Die moderne Entwicklung von Navigationssystemen für Fußgänger soll die Kenntnisse über Landmarken nutzen. Landmarken sollen in die Navigationsanweisungen eingebaut werden, um Navigationssysteme benutzerfreundlich und bequem zu gestalten.

## 3. Analyse

Dieses Kapitel beschäftigt sich mit der Analyse der Anforderungen an ein Navigationssystem für Fußgänger. Das Navigationssystem soll eine natürliche Hilfe für Fußgänger darstellen. Es soll auf einem mobilen Gerät laufen und den Benutzer nicht überfordern. In der Anforderungsanalyse unterscheidet man zwischen funktionalen und nicht funktionalen Anforderungen. Dass die Ressourcen von mobilen Geräten eingeschränkt sind, wird in dieser Arbeit vernachlässigt. Der Grund dafür ist einerseits die ständige und schnelle Entwicklung der Technik, die es erlaubt zu behaupten, dass die Leistung der mobilen Geräte in der absehbaren Zeit soweit ist, dass man über ihre Ressourcen nicht mehr nachzudenken braucht. Andererseits ist der Schwerpunkt dieser Arbeit die einfache und bequeme Benutzung von Fußgänger- Navigationssystemen.

Die Ergonomie spielt in dieser Arbeit eine besondere Rolle. Damit ein System sich benutzerfreundlich und einfach bedienen lässt, soll es die Grundsätze der Ergonomie erfüllen, die im Kapitel 3.2 betrachtet wurden. Auf diesen Schwerpunkt orientieren sich entsprechend die funktionalen sowie nicht funktionalen Anforderungen.

Funktionale Anforderungen werden in dem Kapitel 3.3 beschrieben und präsentieren die Aufgaben des Systems und Dienste, die das System nach Außen bieten soll. Sie definieren den Funktionsumfang eines Systems.

Es gibt auch eine Reihe anderer nicht funktionaler Anforderungen, die man bei der Softwareentwicklung berücksichtigen soll. Das sind Qualitätsanforderungen wie Performanz, Sicherheit, Ergonomie etc., die ein gutes Softwaresystem erfüllen soll. Die Suche nach POIs soll schnell ablaufen, ebenso wie die Routenberechnung. Der Benutzer muss sicher sein, dass die Route, die ihm angezeigt wird, auch die richtige ist und ihn zu seinem Ziel führt. Je nach dem welche Ziele man bei der Entwicklung der Software verfolgt, stehen unterschiedliche nicht funktionale Anforderungen im Vordergrund. In dieser Arbeit geht es um die Benutzbarkeit und Handhabung, entsprechend werden in dem Kapitel 3.4 angeforderte Eigenschaften eines einfach benutzbares Navigationssystem für Fußgänger erläutert.

Bevor die funktionalen Anforderungen kommen, wird nun ein Szenario dargestellt, das das Aufgabengebiet der Anwendung abgrenzen soll. Es ist ein realistisches Szenario, das zeigt, wie es in der Realität aussehen könnte, wenn die Infrastruktur entsprechend vorbereitet wäre. Es dient als Basis für die Analyse und für das Design der System-Architektur.

## 3.1. Beispielszenario Frankfurter Flughafen

Das Szenario beschreibt, wie ein Benutzer das System für die Navigation nutzt. Mit dem Szenario „Frankfurter Flughafen“ bewegt man sich im Indoor-Bereich eines sehr komplizierten Geländes. Damit konzentrieren sich die Anforderungen auf den Aspekt des Innenbereichs.

Ziel der Anwendung ist es, dem Benutzer mit Hilfe seines mobilen Gerätes die Suche der POIs und Orientierung zu seinem Ziel zu erleichtern. Dadurch spart man nicht nur die Zeit, sondern eine Menge Umstände wie Erfragen der anderen nach dem Weg oder stressige Situationen, wenn man sich verirrt hat.

Die Positionierung, Tracking und LBS (Location-Based Services) sind nicht Gegenstand dieser Arbeit. Zum Untersuchungsfeld gehören Landmarken, ihre Verwendung in einem mobilen Navigationssystem für Fußgänger und darauf basierende Bewegungsanweisungen. Deswegen wird das Szenario aus der Sicht dieses Schwerpunktes betrachtet.

### Das Szenario

Ein Gast betritt das Gelände des Frankfurter Flughafens. Auf seinem Handy ist schon ein Navigationssystem mit den Daten des Flughafens vorinstalliert. Solange der Benutzer sich im lokalen Netz des Flughafens befindet, kann er Ziele innerhalb aller Terminals suchen und sich zu diesen Zielen navigieren lassen.

Ist das Ziel gewählt, wird die entsprechende Route von dem aktuellen Standort der Person zum Zielpunkt berechnet. Mathematisches Routing basiert auf Graphentheorie [33]. Mögliche Routen (Flughafen-Gebäudeplan) werden in Sektionen aufgeteilt, die von einer zu einer anderen Landmarke führen. Dabei sind Landmarken Verbindungsknoten in diesen Sektionen. Die üblichen Objekte, wie Treppen, Flure und Kreuzungen werden auch einbezogen.

Zuerst wird dem Benutzer die ganze oder wenn nicht möglich der größte Teil der Route präsentiert, damit er sich mit dieser vertraut machen kann. Dann kann die Sicht nach der Benutzer-Präferenz vergrößert werden. Außerdem kann sich der Benutzer einen Überblick der Gelände verschaffen, indem er die angezeigte Karte in jede der 4 Richtungen verschiebt. Nach dem Überblick kann er durch den Knopfdruck zu seiner aktuellen Position auf dem Bildschirm zurückkehren.

Die Karte und die Umgebung werden in der 2D-Perspektive oder so genannten *Vogelperspektive* dargestellt. Während der Wegfindung wird man von einem Knoten des Routengraphen zu dem anderen navigiert. Die Knoten sind - wie schon oben erwähnt wurde - Landmarken. Dabei sucht der Benutzer die Orientierungspunkte sowohl auf dem Bildschirm des

mobilen Gerätes als auch in der Realität. So müssen Landmarken von dem System so präsentiert werden, dass sie auch in der wirklichen Welt einfach wieder zu erkennen sind.

Um den Benutzer auf dem aktuellen Stand zu halten, sollte die Angabe der Entfernung zur nächsten Richtungsänderung angezeigt werden. Sollte der Weg des Benutzers aus unbekanntem Gründen von der Route abweichen, so sollte ihm dies auf der Anzeige und durch die Vibration des Gerätes signalisiert und dargestellt werden.

## 3.2. Ergonomie und ihre Anforderungen an die Fußgängerführung

Damit das System von Benutzern akzeptiert wird, soll es so gestaltet werden, dass es bequem und unkompliziert in der Nutzung ist. So ist es zu überlegen, auf welche Weise man am besten bestimmte Operationen durchführt, wie die Ablaufsteuerung am einfachsten gestaltet wird und welche grafische Komponente wie zum Beispiel Buttons, Textfelder man einsetzt. Außerdem soll die Anzeige angenehm und attraktiv für das Auge sein. Das heißt, Farben, Schrift, Bilder und die Weise, wie grafische Komponenten platziert werden, müssen gut durchdacht werden.

Die Grundsätze der ergonomischen Dialoggestaltung entsprechend der DIN EN ISO 9241 sind:

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Steuerbarkeit
- Erwartungskonformität
- Fehlertoleranz
- Individualisierbarkeit
- Lernförderlichkeit

Das Ziel der *Aufgabenangemessenheit* ist die effiziente Aufgabenbewältigung. Das System soll den Benutzern helfen ihre Ziele zu erreichen. Das Softwaresystem soll Nützlichkeit und Komfort gewährleisten, indem es sinnvolle und ausreichende Funktionalität und keine unnötige Belastung für Benutzer zu Verfügung stellt. Ein Navigationssystem soll daher z.B. keine Web-Shop-Funktionalität in sich tragen.

*Selbstbeschreibungsfähigkeit* bedeutet, dass es dem Benutzer jederzeit klar ist wo man sich im System befindet, was man da machen kann, wie man zu einer bestimmten Stelle im

System kommt, was die nächsten möglichen Handlungen sind. Jeder Dialogschritt ist also umgehend verständlich und am besten voraussehbar. Verständliche Anweisungen und Navigation sollen dem Fußgänger klären, wie er sein Ziel erreicht.

*Steuerbarkeit* heißt, dass Benutzer Kontrolle und Initiative besitzt. Nicht das System, sondern der Benutzer soll alles im Griff haben. Das Rückgehen sowie Abbruch sollen immer möglich sein. Bei Verwendung von mobilen Geräten hat man Einschränkungen durch die kleine Tastatur und Bildschirmgröße. Dadurch unterscheidet sich auch die Interaktion mit den mobilen Anwendungen von der normalen Systeminteraktion. Mobile Systeme sind Einhandanwendungen. Die Steuerung des Systems soll daher mit einer Hand ausführbar sein. Das heißt, dass die Ausführung einer Funktion durch drücken zweier oder mehrerer Tasten, wie es bei gewöhnlichen Systemen üblich ist (z.B. die Kombination Strg+S für das Speichern), ausgeschlossen ist. Die Steuerung des Systems soll sich dabei aber auf möglichst wenig Tasten beschränken.

Um *Erwartungskonformität* zu gewährleisten, soll das System Konsistenz aufweisen. Konsistentes Layout, Begriffe und Prinzipien sowie Systemstandards sollen erhalten bleiben. Wenn z.B. die Navigationsanweisungen als Text dargestellt werden, dürfen sie nicht plötzlich ohne einer Umstellung durch den Benutzer in Audioform ausgegeben werden. Außerdem soll die Rückmeldung über den Systemzustand und Operationen vorhanden sein.

Ein qualitatives Softwaresystem soll *fehlertolerant* sein. Das System soll mit den Fehlern, die die Benutzer verursachen (z.B. falsche Eingaben), umgehen können. Da nicht alle Fehler vermeidbar sind, helfen gute Fehlermeldungen Probleme zu lösen. Gute Fehlermeldungen sind sichtbar, verständlich, informativ und nicht beleidigend.

### 3.2.1. Multimodale Navigation

In meisten Situationen kann der Fußgänger nicht ständig das mobile Gerät in der Hand halten, da er z.B. Gepäck zu tragen hat oder Kinder an der Hand führt. Außerdem darf die Anwendung aus Sicherheitsgründen nicht die ganze Aufmerksamkeit der geführten Person rauben. Besonders im Außenbereich muss man dauernd auf den Straßenverkehr und auf seine Umgebung achten.

An dieser Stelle kommen die Ansätze multimodaler Interaktion gut an, die das System und seine Bedienung benutzerfreundlicher macht. Für ein Navigationssystem können im Prinzip verschiedene technische Möglichkeiten eingesetzt werden, um dem Benutzer die notwendige Informationen zu übermitteln. Die Autonavigation benutzt neben der graphischen und textlichen Beschreibung auch Audio-Ausgabe.

Im Szenario der Fußgänger-Navigation ist Sprach-Ausgabe auch denkbar. Die Anwendung soll dem Benutzer die Möglichkeit bieten, zwischen Sprach- und Textmodus auswählen zu

können oder beides anzuschalten. Um die Hände frei zu halten, können die mittlerweile sehr verbreiteten Freisprechsysteme für mobile Geräte benutzt werden. Durch solche Systeme wird der Einsatz von Audioausgabe auch in lauten Umgebungen ermöglicht.

Aber nicht nur Sprachkommandos sollen verwendet werden. Die Vibration des Gerätes soll dem Benutzer zusätzlich zu der graphischen und Audio- Ausgabe melden, wenn dieser von der Route abgewichen ist. Die Ein- bzw. Ausschaltung der Vibration soll auch konfigurierbar sein.

Es gibt alternative multimediale Techniken, die in der Fußgängerführung Verwendung finden können [22]. Sie werden in dieser Arbeit jedoch nicht in Betracht genommen.

### 3.3. Funktionale Anforderungen

In dem oberen Kapitel 3.1 wurde der mögliche Funktionsumfang eines Navigationssystems für Fußgänger dargestellt. Anhand dessen lassen sich die funktionalen Anforderungen an das System bestimmen. Es werden mögliche Benutzeraktionen ermitteln und mittels dieser die Funktionalität der Anwendung bestimmt. Im folgenden werden die ausgewählten Funktionalitäten der Anwendung beschrieben, die sich auch in Form der Geschäftsanwendungsfälle im Anhang A.1 widerspiegeln.

Um eine Navigation zu starten, muss dem System bekannt sein, was das Endziel der Navigation ist. Dafür soll das System eine Suche nach den POIs (points of interest) dem Benutzer anbieten. Um eine Suchanfrage ausführen zu können, soll als Voraussetzung die Wissensbasis vorhanden sein. Sie kann durch einen Dienstserver oder eine lokale Datensammlung zur Verfügung gestellt werden. Der Benutzer kann den gesuchten Begriff über die Tastatur eingeben oder das gewünschte Objekt aus einer Liste möglicher POIs auswählen. Sind die erforderlichen Daten nicht vorhanden, soll es dem Benutzer mitgeteilt werden. Sind die gewünschten POIs gefunden worden, können sie auf dem Bildschirm angezeigt werden.

Ist die Suche von POIs reibungslos abgelaufen, werden sämtliche Suchergebnisse angezeigt, die die angegebenen Suchkriterien erfüllen. Wenn kein POI gefunden wurde und kein Ergebnis vorhanden ist, ist dies dem Benutzer mitzuteilen. Wenn die Ergebnisse aber angezeigt wurden, kann der Benutzer ein konkretes Ziel als das Endziel auswählen.

Der Benutzer soll sich für ein Navigationsziel entscheiden und es aus der Ergebnisliste per Tastendruck auswählen. Damit wird dem System auch die Position des Ziel bekannt und die Route kann berechnet werden. Dafür soll auch die aktuelle Position der Person bekannt sein. Wenn es dem System nicht möglich ist, die Position des Navigierenden zu bestimmen, soll es dem Benutzer mitgeteilt werden.

Da sich das Szenario in einem Indoor-Bereich abspielt, werden entsprechend Indoor-spezifische Landmarken zur Verfügung gestellt. Die folgende Tabelle aus [25] zeigt die Indoor-spezifische Landmarken und ihre Charakteristik. Die Relevanz sagt, ob die Landmarke vorhanden ist und wie wichtig sie für die Navigation sein kann. In der Spalte „Klassifizierung“ ist der Nutzen der Landmarke für die Orientierung angegeben. Die letzte Spalte enthält mögliche Detailnamen und Beispiele.

Landmarke	Relevanz	Klassifizierung	Detailnamen (Zusätze)
Raum	ja	essenziell	Küche, Vorlesung, Bad, Anlage, Sekretariat, Arbeitsraum, Unterrichtsraum, Mensa, Büro
Flur	ja	essenziell	Nord, West, Süd, Ost, Mitte
Fahrstuhl	wenn vorhanden	essenziell	Nord, West, Süd, Ost, Mitte
Schilder	ja	essenziell	Wegbeschreibung
Raum-Beschreibungen	ja	sehr wichtig	Raumnummer, Name des Bediensteten
Tür	ja	sehr wichtig	Material, Aussehen
Maschinen	wenn vorhanden	wichtig	Anlagen, Ausstellungsstücke, Computer
Treppen	ja	essenziell	Nord, West, Süd, Ost, Mitte
Bilder	wenn wenig Ausstattung	weniger wichtig	Als Treffpunkte, Malername
Gebäudezugänge	ja	essenziell	Nord, West, Süd, Ost
Fussbodenbelag	wenn prägnant	nicht wichtig	Material
Kamera	wenn wenig Ausstattung	nicht wichtig	
Fenster	ja	weniger wichtig	Nord, West, Süd, Ost
Mobilar (statische)	ja	wichtig	Plastiken, Werkbänke, Tafel Schrank
Telefone	Wenn vorhanden	sehr wichtig	
Ersthelfer	ja	sehr wichtig	ERSTHELFER
Notausgänge	ja	sehr wichtig	Notausgang
Rolltreppen	wenn vorhanden	sehr wichtig	Rolltreppe
Brücken zw. Gebäuden	wenn vorhanden	wichtig	[Geb. A  Geb. B]

Nachdem das Ziel ausgewählt wurden, kann die Route berechnet werden. Hat hat System sie fertig gestellt, wird die Route auf dem Bildschirm angezeigt. Die Route wird zusammen

mit der entsprechenden Karte der Umgebung dargestellt, damit der Benutzer sich in der Umgebung orientieren kann.

In dem Kapitel 2 wird gesagt, dass Landmarken eine wichtige Orientierungshilfe darstellen. Ein benutzerorientiertes Navigationssystem für Fußgänger soll daher Landmarken bei der Berechnung der Route berücksichtigen. Die berechnete Route wird in die Sektionen aufgeteilt und Landmarken werden als Verbindungsknoten in diese Sektionen eingesetzt. Somit werden Landmarken in die Navigation eingebunden. Das Nutzen von Landmarken ist aber nur dann möglich, wenn es eine Sammlung möglicher Landmarken gibt. Wenn keine Informationen über Landmarken zur Verfügung stehen, wird die Route ohne sie berechnet und angezeigt.

Sind aber Landmarken bei der Routenberechnung berücksichtigt worden, werden sie auf dem Bildschirm zusammen mit der Route dargestellt. Landmarken können z.B. als kleine Icons dem Benutzer präsentiert werden.

Zusammen mit der Route werde auch Orientierungshilfen wie Maßstab, Entfernung zum Ziel, Abzweigungshinweise sowie Nordausrichtung präsentiert. Diese Hilfsmittel erleichtern die Orientierung in einer fremden Umgebung.

Das Anzeigen der Route mit den Navigationshilfen und Landmarken reicht für die gute Navigation aber nicht aus. Zur eindeutigen Beschreibung der Route werden textliche Anweisungen eingesetzt. Die graphische Anzeige wird damit durch textliche Beschreibungen erweitert.

Oft möchte der Benutzer mehr von der Umgebung sehen, in der er sich befindet. Oder er möchte die Einzelheiten näher betrachten. Da die Größe des Bildschirms eines Handys relativ klein ist, sind diese Aktionen nur durch das Zoomen der Umgebungskarte möglich. Das System soll also das Verkleinern oder das Vergrößern der Karte und anderer angezeigten Elemente wie Landmarken mittels Tasten ermöglichen.

Um eine bessere Übersicht über die Umgebung zu erhalten, kann Benutzer die Karte mittels vier Navigationstasten entsprechend nach links, rechts, oben oder unten verschieben. Mit der Karte verschieben sich auch die Landmarken. Sie können unter Umständen von dem Bild verschwinden bzw. neue Landmarken können auf der Anzeige erscheinen.

Hat der Benutzer das Ziel erreicht oder wird das System von ihm nicht mehr gebraucht, kann er die Anwendung beenden. Die Laufende Anwendung wird durch den Tastendruck beendet. Dabei sollen alle Verbindungen zu den verschiedenen Diensten getrennt werden.

### **Fehlerbehandlung**

Es ist nicht auszuschließen, dass während des Laufens der Anwendung Fehler auftreten können. Die Verbindung zu dem LBS-Server oder zum Positionsmodul kann unterbrochen

werden. Oder die Anwendung selbst kann fehlerträchtig sein. Solche Situationen sollen berücksichtigt werden und geeignete Kompensationsmaßnahmen sollen getroffen werden. Der Benutzer soll über die Fehler informiert werden.

<b>Nr.:12</b> Fehler ausgeben	
Zusammenfassung	Der Benutzer soll über aufgetretene Fehler informiert werden.
Vorbedingung	Die Anwendung muss laufen.
Beschreibung des Ablaufs	Falls während der Ausführung des Systems ein Fehler auftritt, wird dieser auf dem Bildschirm angezeigt.
Nachbedingungen	<b>N1:</b> Eine Fehlermeldung ist angezeigt. <b>N2:</b> Das System befindet sich in einem konsistenten Zustand

### 3.4. Nicht funktionale Anforderungen

Außer Anforderungen, die die Funktionalität der Software beschreiben, gibt es weitere, nicht funktionale Anforderungen. Nicht funktionale Anforderungen können genauso kritisch für den Erfolg eines Systems sein wie die funktionalen Anforderungen. Diese Anforderungen stellen die Qualitätsmerkmale des Softwaresystems dar. Sie beschreiben wie ein System die funktionalen Anforderungen erfüllen soll. Demzufolge sind sie „Anforderungen an die Umstände, unter denen die geforderte Funktionalität zu erbringen ist“ [13].

Die nicht funktionalen Anforderungen einschließlich Bedienbarkeit sind stark architekturabhängig. Das heißt, dass sie beim Architekturentwurf berücksichtigt werden sollen. Dabei sollen diese Anforderungen, je nach den verfolgten Zielen, in „Muss“- und „Kann“- Anforderungen geteilt werden, da es nicht möglich ist, alle Qualitätsmerkmale gleichzeitig zu erfüllen (Zeit-Kosten-Qualität-Faktor). Nach dem Qualitätsmodell aus ISO/IEC 9126 (DIN 66272) gehen folgende nicht funktionale Anforderungen heraus, die für das in dieser Arbeit beschriebene Softwaresystem relevant sind.

#### Funktionalität

Um das Qualitätsmerkmal *Funktionalität* zu erfüllen, soll zunächst der Funktionsumfang des Navigationssystems für Fußgänger geklärt werden. Die zu implementierenden Funktionen sind im Kapitel 3.3 beschrieben worden.

Die in den Anwendungsfällen dargestellten Funktionen des Systems sollen angemessen sein und damit die Anforderung *Angemessenheit* erfüllen. Das heißt, dass alle für die Fußgängerführung benötigten Funktionen existieren und sich für die Bewältigung bestimmter

Aufgaben eignen. Wenn z.B. Landmarken als Orientierungshilfe genutzt werden, sollen sie von der Position der Benutzer sichtbar und erkennbar sein. Es wird z.B. eine überschaubare Anzahl von gesuchten POIs (points of interest) präsentiert, oder das Zoomen von der Karte kann nicht unendlich weit gehen.

*Richtigkeit* des Systems besteht in der korrekten Angabe und Wirklichkeitstreue von Maßstab und Darstellung der Karte (Positionierung der Räume, Türen, POIs oder Straßen, Plätze) samt Orientierungshilfsmittel (Distanz zum Ziel, Positionierung von Landmarken). Ebenfalls sollen die Navigationsanweisungen den richtigen Weg verfolgen. Wenn die Route nach links geht, soll die Anweisung den Benutzer nach links führen und nicht nach rechts oder geradeaus. Die Suche nach den POIs soll auch richtige Ergebnisse liefern, also wenn man ein Restaurant sucht, dürfen Duty Free Shops nicht in der Ergebnisliste auftreten.

*Interoperabilität* zeichnet sich durch das Zusammenwirken des Systems mit den anderen Systemen aus. So kommuniziert das Navigationssystem mit einem LBS-Server um die Informationen über POIs für den Benutzer zu erhalten.

Die Hauptaufgabe der *Sicherheit* ist die Verhinderung des unberechtigten Zugriffs auf das System (auf den Datenserver und Dienste). Außerdem ist die sichere Kommunikation und Datenübertragung zu gewährleisten, damit sie während der Übertragung nicht bössartig geändert werden können. Privacy soll auch gewährt werden. Viele Personen geben ungern ihre privaten Daten an und wollen sie schützen.

## Zuverlässigkeit

Als Zuverlässigkeit wird die Fähigkeit eines Systems bezeichnet, während einer festgelegten Zeitdauer ohne Ausfälle ausführbar zu sein [18]. Die Zuverlässigkeit bewertet nach der DIN 66272 die Robustheit, die Verfügbarkeit und die Korrektheit der Software [41]. Es soll für das Aufrechterhalten der Anwendung gesorgt werden. Eine Anwendung ist z.B. *robust*, wenn sie trotz Fehler in einem konsistenten Zustand bleibt. Als erstes soll versucht werden Fehler zu vermeiden. Da es nicht möglich ist, alle Fehler zu beseitigen, sind die Fehlerzustände zu minimieren und, wenn ein Fehler unvermeidbar ist, zu kompensieren. Während der Laufzeit soll das System auch nach einem Fehlerauftritt möglichst weiterlaufen. Die fehlerhaften Benutzereingaben sollen abgefangen werden. Fehler, die das Nutzen des Systems unmöglich machen, z.B. wenn die Route nicht berechnet und angezeigt werden konnte, sollen ausgeschlossen werden. Falls solche Fehler trotzdem auftreten, soll im Sinne der *Robustheit* eine alternative Funktionalität angeboten werden, die es dem Benutzer erlaubt das System weiter zu nutzen. Bei einer fehlgeschlagenen POI-Suche könnte dem Benutzer z.B. eine Liste mit allen möglichen Diensten angezeigt werden.

Wenn Fehler aufgetreten sind, für die es keinen konsistenten Zustand gibt (z.B. die Verbindung zu dem LBS ist ausgefallen), sollen sie dem Benutzer gemeldet werden, z.B. durch die Anzeige auf dem Bildschirm.

*Die Verfügbarkeit* wird als Wahrscheinlichkeit, dass ein System zu einem gegebenen Zeitpunkt funktionsfähig ist, bezeichnet. In der Praxis wird sie als Verhältnis der Laufzeit der Anwendung zu der Summe dieser Laufzeit und der Ausfall-Zeit betrachtet [18].

*Die Korrektheit* bedeutet Übereinstimmung zwischen der funktionalen Spezifikation (Anwendungsfälle) und Programmfunktionalität. Also wenn z.B. der Benutzer die Karte auf dem Bildschirm verkleinern will, soll sie auch kleiner und nicht größer dargestellt werden [18].

*Wiederherstellbarkeit* gehört zu den Eigenschaften einer zuverlässigen Software. Es soll möglich sein, die benötigten Daten auch nach dem Ausfall der Anwendung wiederherzustellen und nach dem Neustart des Systems weiterzubenutzen.

## **Benutzbarkeit**

Die Kriterien für die Benutzbarkeit sind Verständlichkeit, Erlernbarkeit und Bedienbarkeit [28].

Um bessere *Verständlichkeit* zu gewährleisten, soll das System die Sprache des Benutzers „sprechen“. Dazu gehört nicht nur die Möglichkeit, die Muttersprache des Benutzers auszuwählen, sondern auch die Abbildung der Realität und Anwendung von Metaphern. Die Verwendung von Landmarken bei der Routebeschreibung bringt das System näher an die Realität. Die Software soll auch dem Benutzer ausreichende Informationen geben (z.B. Statusleiste, Hinweise). Das Navigationssystem soll z.B. die Distanz zum Ziel anzeigen. Die Abweichung von der Route soll entsprechend signalisiert werden und das System soll den Fußgänger darüber informieren.

Die Erklärungen und Beschreibungen der Komponenten sollen anhand von Tooltips erfolgen. Die Anwendung soll ohne weitere Hilfsmittel einfach *erlernbar* und verständlich sein.

*Bedienbarkeit* spiegelt den Aufwand zur Bedienung und Ablaufsteuerung. Die Ablaufsteuerung soll unkompliziert und für den Benutzer gut verständlich sein. Den nächsten Schritt bei der Bedienung vom System soll man voraussehen können. Bei der Belegung der Tasten mit Funktionen soll Redundanz bzw. Doppelbelegung vermieden werden.

## **Effizienz**

In der Effizienz spiegelt sich das Verhältnis zwischen dem Leistungsniveau und dem Umfang der eingesetzten Mittel [27]. Die wichtigen Begriffe sind dabei *Zeitverhalten* und *Verhaltensverhalten*.

Im Zeitverhalten bilden sich Antwort- und Verarbeitungszeiten ab [27]. Da die normale menschliche Reaktionszeit unter zwei Sekunden liegt, sollen die Antwortzeiten des Systems unter diesem Wert liegen. Dauert jedoch die Operation länger als zehn Sekunden, wird der normale Arbeitsfluss unterbrochen und die Aufmerksamkeit der Benutzer wandert ab. In diesem Fall ist ein Feedback vom System an den Benutzer notwendig.

Das Verbrauchsverhalten schließt die zur Erfüllung der Funktionen benötigten Betriebsmittel ein [27]. Das mobile Navigationssystem soll die ganze Infrastruktur berücksichtigen. Die Besonderheit der mobilen Geräte ist die Eingeschränktheit der Hardware-Ressourcen, nämlich CPU, Speicher und Anzeige. Man soll auf die Qualität der angezeigten Karte und Symbole achten, um die Anwendung besonders benutzerfreundlich zu gestalten, was bei den kleinen Bildschirmen schwierig ist. Die CPU soll für die bessere Effizienz der Anwendung ausgelastet werden. Dabei ist aber auf den Verbrauch des Akkus zu achten.

### **Änderbarkeit**

Um den Aufwand zur Durchführung von Änderungen zu minimieren, soll die Architektur der Anwendung gut durchdacht sein. Sie soll mittels Design Pattern [11] entworfen werden und aus in sich geschlossenen Modulen bestehen, die mit den anderen Modulen über ihre Schnittstellen interagieren. Eine gute Architektur erhöht die *Modifizierbarkeit* der Software. Das heißt, dass der Aufwand zur Verbesserung, Fehlerbeseitigung oder Anpassung an Umgebungsänderungen geringer wird. Da die Module über Schnittstellen kommunizieren, haben Änderungen eines Moduls keine Auswirkung auf die anderen Module des Systems. Dadurch steigt die *Stabilität* der Software und Seiteneffekte werden reduziert. Die Anwendung von Design Pattern fördert auch die *Analysierbarkeit* und senkt damit den Aufwand zur Bestimmung von Mängeln oder Ursachen von Fehlern.

Die *Prüfbarkeit* ist nicht nur von der Architektur der Anwendung, sondern auch von der Soft- bzw. Hardware abhängig, die für das Testen des Systems notwendig sind.

### **Übertragbarkeit**

Unter Übertragbarkeit versteht man die „Eignung der Software, von einer Umgebung in eine andere übertragen zu werden“ [27]. Die Anwendung soll also ohne technische oder software-spezifische Anpassung auf ein anderes mobiles Gerät übertragen werden können. Verschiedene mobile Geräte nutzen Betriebssysteme unterschiedlicher Hersteller. Die Anwendung soll dies berücksichtigen und an die unterschiedlichen Umgebungen angepasst werden. Dabei spielt die Programmiersprache, die für die Entwicklung des Systems eingesetzt wurde eine wichtige Rolle. Sie soll möglichst plattformunabhängig sein. Der Aufwand zur Installation der Anwendung soll diskutiert werden. Denn es kann passieren, dass nach dem Absturz der Software eine neue Installation des Systems notwendig ist.

## 3.5. Zusammenfassung

In diesem Kapitel wurden Anforderungen an ein Navigationssystem für Fußgänger erläutert. Dabei ist es zwischen den funktionalen und nicht funktionalen Anforderungen zu unterscheiden. Das betrachtete Beispiel-Szenario beschreibt die mögliche Anwendung des Navigationssystems für Fußgänger und dient als Basis für die daraus folgenden funktionalen Anforderungen.

Als Schwerpunkt dieser Arbeit wird die einfache Benutzung des Systems betrachtet. Wird bei dem Entwurf der Anwendung den Prinzipien der Ergonomie gefolgt, gewinnt sie mehr Chancen, von den Benutzern akzeptiert zu werden.

Die funktionalen Anforderungen beschreiben die Funktionalität und Aufgaben des Systems. Sie sind in den Anwendungsfällen zusammengefasst und beschrieben. Die wichtigsten Funktionen der Anwendung, die in Rahmen dieser Arbeit betrachtet wurden, sind:

- POI suchen
- Suchergebnisse anzeigen
- Ziel wählen
- Route darstellen
- Landmarken in die Routenberechnung einbeziehen
- Landmarken anzeigen
- Orientierungshilfen darstellen
- Navigationsanweisungen zu der Route darstellen
- Karte zoomen
- Karte verschieben
- Anwendung beenden

Die nicht funktionalen Anforderungen spiegeln die Qualitätsmerkmale, die das System aufweisen soll. Entsprechend den Anforderungen muss die Architektur des Systems entworfen werden. Eine detaillierte Beschreibung des Architekturentwurfs wird im folgenden Kapitel 4 dargestellt.

## 4. Design

In diesem Kapitel wird die Umsetzung der im Kapitel 3 erarbeiteten funktionalen und nicht funktionalen Anforderungen umgesetzt. Hier und im darauf folgenden Kapitel 4.5 wird ein Konzept zur Entwicklung eines einfach benutzbaren Navigationssystems für Fußgänger ausgearbeitet.

Das Lösungskonzept enthält die statische Struktur und die dynamischen Abläufe der Anwendung. Die statische Struktur spiegelt sich in der Architektur des Systems und in den Anwendungsfällen, die mittels Use-Case-Diagrammen dargestellt werden.

Im Unterkapitel 4.2 wird die gesamte Anwendungsarchitektur spezifiziert, die als Basis zur späteren Realisierung dient [26]. Zuvor wird im folgenden Unterkapitel das Gebäudemodell, das bei dem Entwurf benutzt wurde, präsentiert. Im Kapitel 4.4 wird ein Framework für die Entwicklung mobiler Anwendungen und seine Architektur beschrieben, das als Gerüst für die Realisierung des Prototyps diente.

### 4.1. Gebäudemodell

Das Beispielszenario spielt sich in einem Flughafen ab. Also muss ein Gebäudemodell vom Flughafen zur Verfügung stehen, damit die absolute Position von Personen und POIs innerhalb des Flughafenkomplexes bestimmt werden kann.

Für die Verwaltung von Gebäudeinformationen kommt in der Regel ein Gebäudeinformationssystem (GebIS) zum Einsatz. Es ist ein System zur Erfassung, Dokumentation und Herausgabe von Gebäudeinformationen. Das System umfasst ein Datenmodell, das sich in Geometrie- und Sachdaten unterteilen lässt. Bei den Geometriedaten handelt es sich zu meist um Vektor- oder Rasterdaten, mit denen die Gebäudebeschreibung in Drahtmodellen, Begrenzungsflächen oder Volumenmodellen vorgenommen wird [21]. Als eine optionale Erweiterung des Gebäudemodells können die fachlichen Daten dienen. Hiermit lassen sich beispielsweise Raum- oder Telefonnummern innerhalb eines Gebäudes hinterlegen.

In dieser Arbeit wird zur Gebäudeverwaltung ein standardisiertes semantisches Datenformat für Stadt- und Gebäudemodelle benutzt, nämlich CityGML.

### 4.1.1. CityGML

CityGML ist ein einfaches semantisches Informationsmodell, das zur Repräsentation von städtischen Objekten in 3D dient. Es ist als ein offenes Datenmodell und XML-basiertes Format entworfen worden, um virtuelle städtische 3D-Modelle abzulagern und auszutauschen. CityGML beruht auf GML3, der Geographic Markup Language, und wurde durch eine Vielzahl von neuen Konstrukten zur Spezifikation virtueller Stadtmodelle erweitert. Es basiert unter anderem auf einer Reihe von Normen, die durch ISO 191xx-Familie, dem Open Geospatial Konsortium, W3C Konsortium, dem Web 3D Konsortium und OASIS festgelegt sind [15].

CityGML definiert Klassen und Beziehungen für die meisten wichtigen topographischen Objekte der Stadt und Umgebung. Dabei werden ihre geometrischen, topologischen, semantischen und visuellen Eigenschaften berücksichtigt. Der Begriff „City“ ist allgemein definiert und bedeutet in diesem Fall nicht nur Bauwerke, sondern auch Vegetation, Gewässer, Bodenebenenheiten, „Ausstattung“ der Stadt (z.B. Ampel, Sitzbank) und vieles mehr. Im Schema sind generalisierte Hierarchien zwischen konkreten Klassen, Aggregationen, Beziehungen zwischen Objekten und räumliche Eigenschaften enthalten.

CityGML kann sowohl für große Bereiche als auch für kleine Gebiete wie einzelne Gebäude angewendet werden, da es verschiedene Abstraktionsstufen der Darstellung anbietet. Insgesamt gibt es fünf aufeinander folgende Levels of Detail (LOD), bei denen Objekte mehr Details mit der Steigerung des Levels bekommen. Die Levels unterstützen effiziente Visualisierung und Datenanalyse. In CityGML-Schema kann ein und dasselbe Objekt in verschiedenen Levels dargestellt werden. Weiter können diese Darstellungen von einem Objekt miteinander kombiniert werden.

Der Level LOD0 ist der „größte“ Level. Das ist ein zweieinhalb-dimensionales digitales Gelände-Modell (Digital Terrain Model), das mit einem Luftbild einer Gegend oder einer Karte dekoriert werden kann. LOD1 präsentiert ein einfaches Modell, das rechteckige Blöcke von Gebäuden mit flachen Dächern darstellt. Im Gegenteil von LOD1 unterscheiden sich im LOD2 Oberflächen von Gebäuden und Dächern. Außerdem kann auch die Vegetation einbezogen werden. Mehr Details zeigt Level 3. Hier werden Bauwerke und ihre Textur, Dächer- und Wändekonstruktionen, Balkons ausführlich beschrieben. Zusätzlich enthält der LOD3 genaue Beschreibung der Vegetation und Transportobjekte. Der LOD4 vervollständigt den LOD3 durch die Aufnahme der Objekte der Innenausstattung. Der LOD4 umfasst Objekte wie Räume, Türen, Fenster, Treppen und Möbel. Im Anhang [A.2](#) wird der Gebäudeplan (ein Teil) im CityGML-Format dargestellt.

Für diese Arbeit ist der LOD4 von Relevanz, da die Fußgänger-Navigation einen hohen Grad an Genauigkeit erfordert. Der Level 4 kann diese Anforderung mit seinen Objekten vollstän-

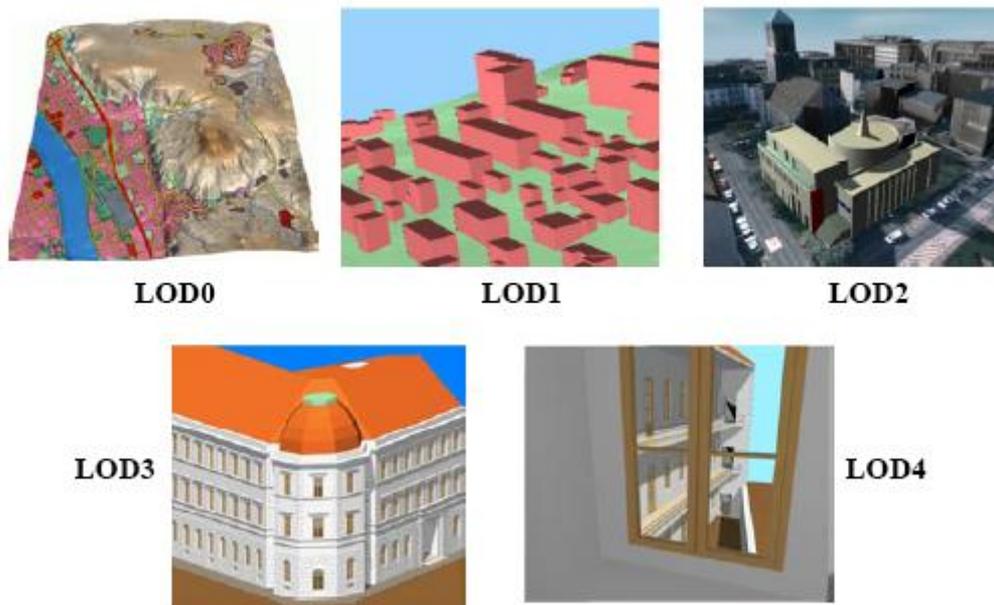


Abbildung 4.1.: Die fünf Levels of Detail, definiert von CityGML; Quelle: [15]

dig erfüllen. Also muss die Datei, die Karte oder Gebäudeplan beschreibt, den LOD4 implementieren, um fußgängergerecht zu sein.

## 4.2. Entwurf der Anwendungsarchitektur

Der Übergang von der Analyse zur Realisierung wird durch eine Anwendungsarchitektur unterstützt, die das Softwaresystem verständlich macht. Eine Anwendungsarchitektur „bezeichnet die Modelle und die konkreten Komponenten eines Software-Systems in ihrem statischen und dynamischen Zusammenspiel. Sie kann selbst als explizites Modell dargestellt werden. Eine Software-Architektur beschreibt ein konkretes System in seinem Anwendungskontext“ [40]. Die Architektur stellt also die Gesamtstruktur der Anwendung auf hoher Abstraktionsebene [41] dar.

Eine saubere, gut durchdachte Anwendungsarchitektur hilft dabei,

- eine sinnvolle Arbeitsleistung und Übersicht,
- langfristige Flexibilität in der Systementwicklung und
- einen höheren Wiederverwendungsgrad zu erzielen [24].

In der Softwareentwicklung wird zwischen *technischer* und *fachlicher* Architektur unterschieden [24].

Die technische Architektur beschreibt, aus welchen technischen Komponenten und Schichten die Anwendung besteht. Sie geht nicht auf die konkreten fachlichen Aufgaben des Systems ein, sondern repräsentiert eine einfache Sicht auf allgemeine Eigenschaften der Anwendung. Bei einem mobilen System geht es in den meisten Fällen um eine Client-Server-Architektur (siehe Kapitel 4.3). Wobei eine Aufteilung in einen Client und Server nicht notwendig ist. Die Informationen über die POIs und die Umgebungskarte können gleich bei der Installation des Systems im Speicher abgelegt werden. Das Prototyp-System, das im Rahmen dieser Arbeit entwickelt und im Kapitel 4.5 beschrieben wird, stellt mit sich einen Fat-Client dar. Alle nötigen Informationen werden lokal auf dem mobilen Client gehalten. Der Server wird ausschließlich zum Aktualisieren der Karten und Informationen über POIDs benutzt. Außerdem wird eine Infrastruktur zur Positionsbestimmung gebraucht. Es kann GPS, ein IMAPS-Modul oder ein anderes System sein. Auf die Positionsbestimmung wird nicht weiter eingegangen, da dieses Thema nicht Gegenstand dieser Arbeit ist.

Die fachliche Architektur beschreibt die grundlegende Struktur und Zerlegung des Systems nach dem fachlichen Aspekt und aus der Sicht des Anwendungsgebietes. Die fachliche Architektur basiert somit auf den im Kapitel 3.3 beschriebenen funktionalen Anforderungen und Anwendungsfällen.

Von einer Vision von Geschäftsprozessen, kommt man durch die Analyse von diesen zu einer fachlichen Architektur. Ein mobiles Navigationssystem ist ein interaktives System. Das System soll die Suche nach Objekten, die einen Benutzer interessieren, ermöglichen. Eine direkte Eingabe des Endziels soll auch erlaubt sein. Somit steht fest, dass die Anwendung über eine graphische Oberfläche verfügen und eine Präsentationskomponente enthalten soll, die für die Darstellung verantwortlich ist.

Für die Suche nach POIs braucht man eine Sammlung und eine Beschreibung dieser Objekte. Diese Informationen kann ein Location-Based-Service (LBS) zur Verfügung stellen. Dieser Dienst wird durch eine separate Komponente präsentiert.

Ist das Ziel bekannt, soll anhand der aktuellen Position der Person und des Zielobjekts die Route berechnet werden. Zum Ermitteln der aktuellen Position wird ein separater Modul gebraucht. Dieser Modul soll die Positionsdaten vom Sender empfangen, diese eventuell verarbeiten und an die anderen Komponenten weiterleiten. Eine extra Komponente soll für die Berechnung von Routen sorgen. Sie nutzt Daten, die sie vom LBS und dem Positionsmodul bekommt. Außerdem braucht diese Komponente Informationen über Landmarken und Entscheidungspunkte, die als Knoten im Routengraph dienen. Diese Informationen werden vom Kartenmodell geliefert, das ihre Daten von einer Persistenzkomponente erhält.

Die ermittelte Route wird dann zusammen mit dem Umgebungsplan und Landmarken auf dem Bildschirm angezeigt. Für die Vermittlung der Daten zwischen der Präsentationskomponente und den oben beschriebenen Modulen soll eine weitere Komponente (Anwendungslogik) sorgen.

Die Umgebungskarten und evtl. die Sammlung von POIs wird im Speicher des mobilen Geräts abgelegt. Eine Komponente, die für das Laden dieser Daten und ihrer Weiterleitung verantwortlich ist, stellt damit die Persistenzkomponente dar.

Die Abbildung 4.2 stellt eine grobe Gesamtarchitektur des Navigationssystems für Fußgänger dar. Es ist ein Schichten-Modell und vermittelt die oben beschriebene Analyse der Geschäftsprozesse.

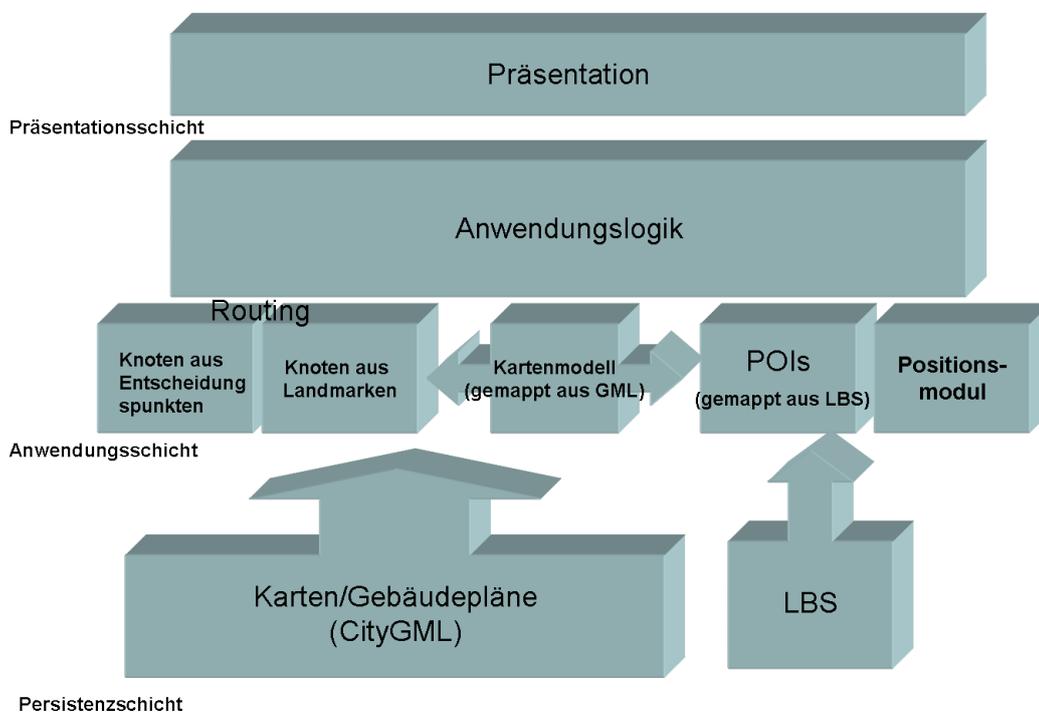


Abbildung 4.2.: Drei-Schichten-Architektur

Die dargestellte Anwendungsarchitektur basiert auf dem Drei-Schichten-Modell. Die Vorteile eines Schichtenmodells liegen in der Trennung der Schichten bzw. Module und im verständlichen Strukturkonzept. In der Drei-Schichten-Architektur werden fachliche und technische Systemteile von einander getrennt [41]. Es wird zwischen Präsentation (Benutzeroberfläche), Fachdomäne und technischer Infrastruktur unterschieden. Jede Schicht nutzt die Dienste,

die die unten liegende Schicht anbietet. Die Drei-Schichten-Architektur besitzt folgende Vorteile, die besonders für interaktive Anwendungen relevant sind:

- Präsentation ist einfach zu bauen.
- Präsentation ist von der Anwendungslogik entkoppelt.
- Fachschicht ist klar abgetrennt.
- Die Entwicklung, Wartung bzw. Pflege sind einfacher.
- Komplexität ist gesenkt, da jede Schicht eine klare Aufgabe hat.

Die einzelnen Schichten sind Module. Jede für sich soll eine konkrete Aufgabe des Geschäftsprozesses der Anwendung erfüllen. Zusammengefasst werden die Module im Folgenden präsentiert:

- **Präsentationsschicht:** Auf dieser Schicht wird die Präsentation des Navigationssystems realisiert. Auf dem kleinen Bildschirm des mobilen Gerätes wird die Karte samt Weganweisungen angezeigt, die man drehen, verkleinern und vergrößern kann.
- **Anwendungsschicht:** Die Anwendungsschicht stellt den fachlichen Teil der Anwendung dar. Die fachlichen Anforderungen werden ausschließlich in der Anwendungsschicht realisiert. Die Anwendungsschicht muss alle fachlichen Funktionalitäten vollständig umfassen. Erst dann, wenn die Fachlichkeit komplett in ihr realisiert ist, sind die Schichten entkoppelt und die Präsentation sowie die Persistenzschicht können ohne große Auswirkungen auf die Anwendungsschicht ausgetauscht werden. Die Anwendungsschicht besteht aus mehreren Teilen, die getrennte Module sind.
  - **Positionsmodul:** Das Modul dient zur Ermittlung von Positionsdaten des Fußgängers und könnte durch das Imaps-Modul realisiert werden. Die Komponente bekommt Daten von einem Sender und konvertiert diese in Koordinatenpaare. Das berechnete Koordinatenpaar kann von der Präsentationsschicht genutzt werden, um die aktuelle Position der Person anzuzeigen.
  - **Kartenmodell:** Die Karte oder der Gebäudeplan werden als eine Datei in einem standardisierten Format abgelegt. In dieser Arbeit wird CityGML, das im Kapitel [4.5.3](#) beschrieben wird, als solcher Format für die Beschreibung der Karte benutzt. Nach dem Parsen dieser Datei entsteht ein Kartenmodell, das in der Anwendung von den anderen Modulen (z.B. Präsentationsschicht) weiter benutzt werden kann.
  - **POIs:** Neben dem Kartenmodell wird ein Modell für die von dem LBS-Server gelieferten POIs erstellt. Das Modell wird bei der Suche von POIs eingesetzt.

- Routing: Diese Komponente stellt die Anwendung zur Bereitstellung von Route, Anweisungen und Landmarken.
- **Persistenzschicht:** Die Kartendaten, Pläne und POIs können zusätzlich zu den auf dem mobilen Gerät gespeicherten Daten auf den Servern abgelegt werden. Bei Bedarf (z.B. für eine Aktualisierung der Karte) können Daten von diesen heruntergeladen werden.

Somit ergibt sich, dass jede Schicht ihre Funktionalität von den anderen Modulen abkapselt. Die Kommunikation zwischen den Schichten findet über die Schnittstellen statt. Diese werden mittels Interfaces als technische Komponenten realisiert. Die technische Realisierung wird auch durch die Verwendung von Design-Mustern unterstützt und im Kapitel 4.5 näher beschrieben.

### 4.3. Entwurf der Client-Server-Architektur

Hardware-technisch liegen alle Module bis auf einen Teil der Persistenzschicht auf dem mobilen Gerät. Da aber die Anwendung zum Nachladen von Daten einen anderen Hardware-Teil benutzt, ist sie nach Definition eine Client-Server-Anwendung.

In der Fachliteratur [34] werden fünf unterschiedliche Ansätze der nützlichen Kompetenzverteilung für Client-Server-Systeme beschrieben. Die Abbildung 4.3 stellt diese dar.

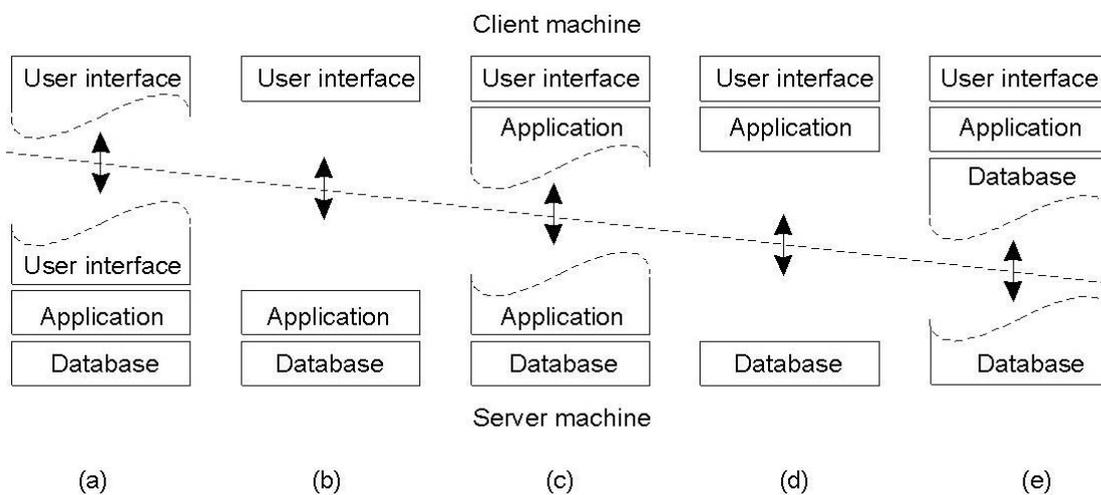


Abbildung 4.3.: Eine Kompetenzverteilung nach Tanenbaum [34]

Zur Realisierung der Anwendung sollte die Fat-Client-Variante eingesetzt werden, so wie es in der Abbildung 4.3 unter (e) dargestellt ist. Der Fat-Client erhält nicht nur die gesamte Applikation, sondern hat zusätzlich Dateneinträge gespeichert. Auf diese kann die Anwendung im Verlauf zugreifen. Auf dem Server liegen Daten, die bei Bedarf geladen werden können, aber zum Teil oder möglicherweise ganz auf dem Client gespeichert sind. So können z.B. Karten vom Server nachgeladen oder auf eine neuere Version aktualisiert werden.

## 4.4. Framework für die Entwicklung mobiler Anwendungen

In der ersten Phase des Designs und der Realisierung wird ein Prototyp entwickelt, der auf einer Software laufen soll, die ein mobiles Gerät simuliert. Die Verwendung eines Emulators befreit die Realisierung von lästigen Übertragungen der entwickelten Software auf das mobile Gerät (z.B. zum Testen). Diese Übertragung ist umständlich und kostet viel Zeit. In dieser Arbeit wird ein Emulator namens Android eingesetzt.

Android ist eine Software-Sammlung, die ein Betriebssystem, eine Middleware und einige fertige Anwendungen für mobile Geräte enthält. Anwendungen die in Java geschrieben sind und Android SDK benutzen, können auf dieser Plattform laufen [1]. Die Abbildung 4.4 zeigt die wichtigen Komponenten vom Android-System.

- **Applications:** Android ist mit einer Reihe von Basis-Anwendungen ausgestattet. Dazu gehören ein Email-Client, SMS-Anwendung, Kalender, Maps, Internet-Browser, Adressbuch und andere. Alle Anwendungen sind in Java realisiert.
- **Application Framework:** Zur Entwicklung neuer Anwendungen werden Framework-APIs angeboten, auf die man einen vollen Zugriff hat. Die Basis-Applikationen nutzen die gleichen APIs. Die neuen Applikationen können die bereits vorhandenen Komponenten einfach wiederbenutzen. Unter dem Framework liegt eine Menge von Diensten und Systemen, die folgende Komponenten beinhalten:
  - *Views* sind ein Satz von erweiterbaren graphischen Komponenten, die benutzt werden können, um eine Applikation zu bauen. Views bestehen aus Listen, Grids, Text-Boxes, Buttons und einem eingebauten Web-Browser.
  - *Content Providers* erlauben Anwendungen auf die Daten anderer Anwendungen zuzugreifen (z.B. vom Adressbuch) oder ihre eigenen Daten zu verteilen.
  - *Resource Manager* unterstützt den Zugriff auf verschiedene Ressourcen anderer Herkunft, wie z.B. Graphiken und Layout-Dateien.

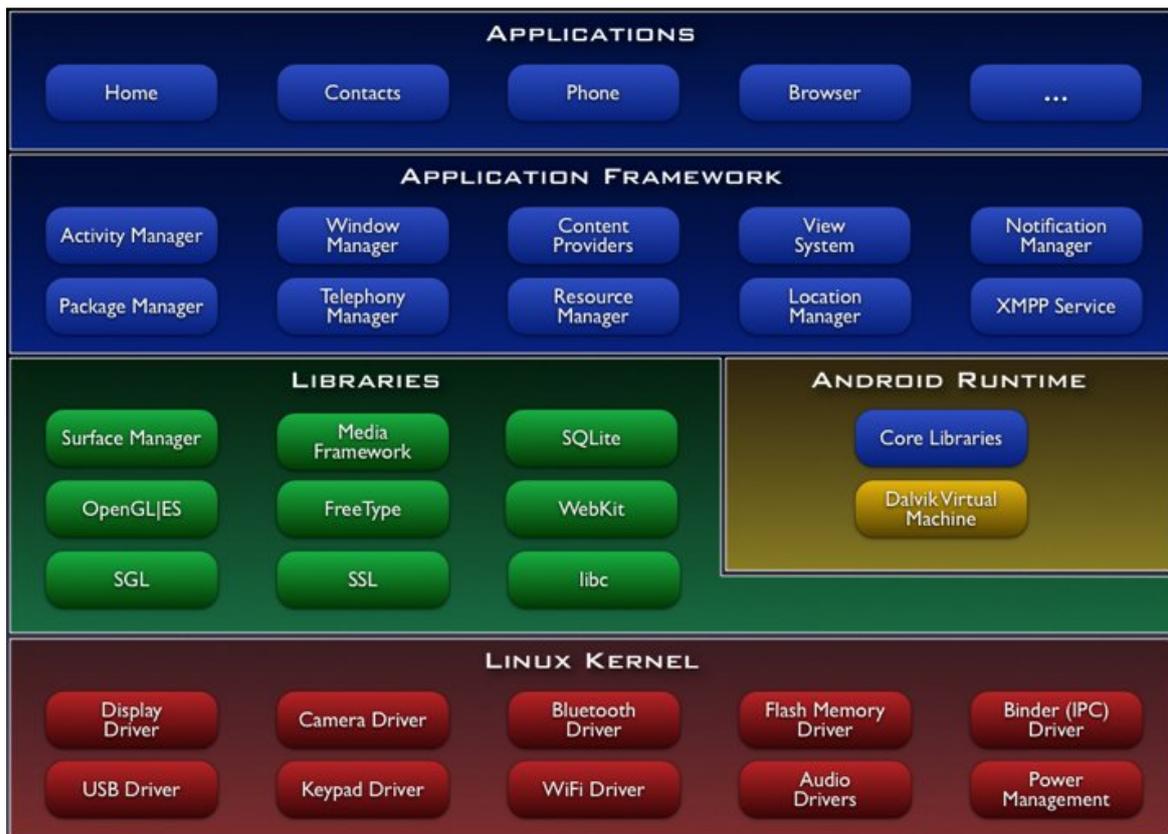


Abbildung 4.4.: Android-Architektur; Quelle: [1]

- *Notification Manager* ist für die Ausgabe von Warnungen im Status-Bar verantwortlich.
- *Activity Manager* verwaltet den Lebenszyklus von Anwendungen und stellt die wichtigen Navigationsfunktionen bereit.

Viele Komponenten im Framework überschneiden sich in ihrer Funktionalität mit den Komponenten der Java Standard Edition. So bietet Android eigene Bibliotheken für XML-Verarbeitung (`android.sax`; `android.xml.dom`; `android.xml.parsers`), I/O-Dienste (`com.google.common.io`; `com.google.common.io.android`) und unterschiedliche Utility-Klassen (`android.util`). Das macht die Anwendungen, die in Standard Java geschrieben sind, mit dem Android nicht kompatibel.

- **Libraries (Bibliotheken):** Android-System enthält C/C++ Bibliotheken (z.B. Medien-Bibliothek, 3D-Bibliotheken), die von seinen Komponenten benutzt werden. Durch das Application-Framework liegen diese Bibliotheken offen für die Entwicklung.

- **Android Runtime:** stellt die Bibliotheken dar, die die Funktionalität der Java Kernbibliotheken anbieten. Jede Android-Anwendung läuft in ihrem eigenen Prozess mit der eigenen Instanz von *Dalvik virtual machine (VM)*. Dalvik ist so realisiert, dass das Gerät mehrere VMs effizient ausführen kann. Die Dalvik VM bearbeitet Dateien im *Dalvik Executable (.dex)* Format, der für den minimalen Speicherverbrauch optimiert ist. Die VM ist Register-basiert und kann Klassen ausführen, die durch den Java-Compiler gebaut worden sind und in den .dex-Format transformiert wurden. Die Dalvik VM ist auf den Linux-Kernel angewiesen, um die low-level Funktionalitäten wie Threading oder Speicherverwaltung ausführen zu können.
- **Linux Kernel:** Android beruht auf der Linux Version 2.6, um die Kerndienste wie Sicherheit, Prozess- und Speicherverwaltung, Netzwerkdienste und andere anbieten zu können. Außerdem dient Linux Kernel als eine Schicht zwischen Hardware und den anderen Softwarekomponenten.

Zum Android-System gehört auch ein Emulator mobiler Geräte. Der Emulator imitiert alle Funktionen und das Verhalten von einem Handy oder PDA, außer dass er keine Anrufe empfangen und tätigen kann. Der Android-Emulator enthält verschiedene Navigations- und Kontrolltasten, die man mit Hilfe der Maus oder Tastatur „drücken“ kann und damit z.B. Events für die neuen Anwendungen generieren. Außerdem bietet der Emulator Debug-Möglichkeiten. Mit Hilfe einer Konsole kann man die Ausgaben loggen, Anwendungsinterrupts, Latenzeffekte und Fehler auf dem Datenkanal simulieren.

## 4.5. Realisierung des Software-Prototyps für die Evaluation

In diesem Kapitel wird die Software-Realisierung des im Kapitel 5 beschriebenen Tests erläutert. Für die Realisierung sind geeignete Hard- sowie Software notwendig. Zu diesem Zweck wird im ersten Abschnitt 4.5.1 des Kapitels die eingesetzte Hardware beschrieben, auf der die Anwendung laufen soll. Des Weiteren wird die Software und die Programmiersprache zur Erstellung des Codes festgelegt. Nachdem die Hard- und Software bestimmt sind, kann mit der Realisierung der im Kapitel 3 beschriebenen Anforderungen begonnen werden.

Die Realisierung des Tests soll einen Prototyp hervorbringen, mit dem es möglich ist, eine Simulation der Indoor-Navigation durchzuführen. Dabei soll bewiesen werden, dass die Landmarken ein unverzichtbares Hilfsmittel zur Wiedererkennung der Umgebung und Orientierung sind. Deswegen wird zur Erfüllung der Funktionalität insbesondere auf die in den Kapiteln 3.4 und 3.2 gestellten nicht funktionalen Anforderungen Bezug genommen.



Abbildung 4.5.: Graphische Darstellung vom Android-Emulator

### 4.5.1. Hardware

Zur Erstellung eines Navigationssystems für Fußgänger wird verschiedene Hardware benötigt. Zum einen braucht man ein mobiles Gerät, auf dem die Anwendung laufen soll; zum anderen die Hardware zum Erstellen des Codes. Außerdem wird eine Infrastruktur aufgebaut, die Positionsdaten ermittelt und weiterleitet.

Die Erstellung des Anwendungscode passiert auf einem handelsüblichen Laptop. Zur Positionsermittlung können ein IMAPS-Modul eingesetzt werden [14], welches Positionsdaten selbstständig errechnen kann. Da die Positionsermittlung nicht zum Schwerpunkt dieser Arbeit gehört, wird auf sie nicht weiter Bezug genommen.

Unter mobilen Geräten versteht man ein PDA oder ein Handy. Die Applikation soll auf einem Handy laufen. Durch die Verwendung eines mobilen Telefons entstehen Einschränkungen für die im Kapitel 3 ermittelten Anforderungen. Der Grund dafür sind die technischen Eigenschaften von mobilen Geräten. Man soll bei der Realisierung der Anwendung diese Einschränkungen berücksichtigen, um ein benutzergerechtes System zu entwickeln.

Zu den eingeschränkten Ressourcen gehören der kleine Bildschirm, die Tastatur, der Speicher, der Prozessor, der Akkumulator und die Netzwerkschnittstelle. Der Bezug wird in dieser

Arbeit allerdings nicht auf Speicher, Akkumulator, Prozessorleistung und Netzwerkschnittstelle genommen. Diese Komponenten eines mobilen Gerätes entwickeln sich ständig weiter und werden immer leistungsvoller. Ihre Einschränkungen der Realisierung werden immer geringer. Schließlich ist es nur eine Frage der Miniaturisierung (man könnte auch einen Leptop im Rucksack auch als mobiles Gerät betrachten). Außerdem werden sie bei der Erläuterung von Darstellung und Interaktion berücksichtigt.

Neben den anderen Ressourcen verbessert auch die Anzeige von mobilen Geräten ihre Qualität, jedoch nicht ihre Größe. Es bleibt eine Herausforderung, die Präsentation der Anwendung auf dem kleinen Bildschirm so zu gestalten, dass Benutzer es akzeptieren.

Die Interaktion mit dem System passiert über die Tastatur. Die Tastatur ist ziemlich klein und die Anzahl der Tasten ist im Vergleich zu einer gewöhnlichen PC-Tastatur viel geringer. Sie ist auf ca. 20 Tasten beschränkt. Deshalb sind die Tasten mehrfach belegt, um die Buchstaben-Zeichen komplett umzusetzen. Dadurch wird die Texteingabe erschwert.

#### **4.5.2. Software**

Zu der Software, die zur Realisierung der Applikation notwendig ist, gehört eine Entwicklungsumgebung, mit deren Hilfe der Programmcode erstellt wird und eine Softwareumgebung, auf der die Anwendung laufen soll. Bei der zweiten Software ist es wichtig, dass die Forderung nach der Lauffähigkeit der Anwendung auf unterschiedlichen mobilen Geräten verschiedener Hersteller erfüllt wird.

Gemäß der Anforderung Übertragbarkeit aus dem Kapitel 3.4 wird Java EE als Entwicklungstechnologie gewählt. Sie garantiert die Plattformunabhängigkeit. Eine in Java implementierte Software ist auf jedem bekannten Betriebssystem lauffähig, wo eine Java Virtual Machine (JVM) ausgeführt wird. Java verfügt über eine sehr große Klassenbibliothek, die es erlaubt, Probleme unterschiedlicher Art zu lösen.

Es wird nicht die mobile Version J2ME eingesetzt. Der Java EE-Hersteller will diese Technologie einstellen und in der Zukunft die Standard Edition zur Entwicklung mobiler Applikationen einsetzen [31]. Mit der Standard Edition bekommt man die Möglichkeit, eine Software auf eine gewöhnliche Art zu erstellen, was die J2ME nicht bietet.

#### **4.5.3. Realisierung der Karte**

## XML-Parsing

Die XML-Datei, die das Gebäudemodell darstellt, wird geparkt, um daraus Objekte zu erzeugen, die in der Anwendung benutzt werden, z.B. um diese Objekte dann auf dem Bildschirm zu präsentieren. Der CityGML hat ein kompliziertes Schema und wurde für die 3D-Darstellung entworfen. Der zu entwickelnde Prototyp soll aber die Karte aus der Vogelperspektive, also in 2D anzeigen. Daher wird ein Parser gebaut, der Dateien im CityGML-Format parst, die resultierende Objekte jedoch den Prototyp-Anforderungen gerecht sind. Der Parser basiert auf dem SAX-Parser.

Der Prozess des Parsens besteht aus zwei Teilen. Im ersten Teil wird der Parser der durch die `GmlDomBuilder.java`-Klasse präsentiert wird, quasi konfiguriert, indem er die abstrakte CityGML-Beschreibungsschema einliest. Diese braucht der SAX-Parser, der als grundlegender Parser dient. Weiter überprüft die `GmlModelCheck.java`-Instanz, ob die Datei, die einen Gebäudeplan beschreibt, dem CityGML-Schema entspricht. Ist die XML-Datei korrekt, startet der Parser. Die Handler `GmlDocumentHandler.java`, `GmlContentHandler.java` und `GmlDataHandler.java` gehen durch den XML-Code und erzeugen Objekte, die in dieser ersten Phase in einem DOM-Dokument abgelegt werden. Die Objekte - Bestandteile eines Innen- oder Außenraums - werden im unteren Kapitel genauer in Betracht genommen.

In der zweiten Phase wird das resultierende DOM-Dokument betrachtet. Es wird durch seine Kinder (*children*) durchgegangen. Sie werden aussortiert und in einer Liste abgelagert. Somit erhält man eine Liste mit allen nötigen Raumobjekten, die dann weiter bearbeitet und z.B. auf dem Bildschirm in 2D-Form angezeigt werden können.

## Karten-Mapping und Darstellung

Die in der XML-Datei festgelegten Objekte wie Räume, Türen, Fenster und Landmarken und ihre Attribute werden bei dem Parsen in Instanzen der Java-Klassen umgewandelt. Diese Klassen sind Erweiterungen von einer der zwei abstrakten Klassen `AbstractRoomElement.java` und `AbstractRoomFeature.java`. Diese abstrakten Klassen implementieren ihrerseits das `ModelElement.java`-Interface. Die vier für die Realisierung des Prototyps notwendigen Klassen sind `Room.java`, `Door.java`, `Landmark.java`, `Window.java`.

Alle diese Objekte besitzen unter Anderem ein Attribut des Typen `GMLShape.java`. Dieses Attribut enthält Koordinaten für diese Raumobjekte. Besonders Räume haben eine komplizierte Geometrie, da sie eher selten eine rechteckige Form haben. Räume werden somit als Paths, die mittels `path.moveTo(x, y)` und `path.lineTo(x, y)` erstellt werden, oder als Rechtecke gezeichnet.

Türen und Fenster haben zwar eine einfachere Geometrie als Räume, werden aber aus dem Grund der Wiederverwendung auf die gleiche Weise gezeichnet, also als Path-Objekte.

Landmarken sollen im Gegenteil zu den Raumobjekten als Icons dargestellt werden. Daher sind für die Darstellung von Landmarken nur zwei Koordinaten notwendig. Die XML-Datei soll aber eine URL für das entsprechende Bild enthalten, mittels deren das Bild geladen wird. Auf dem Bildschirm werden dann Landmarken als Bitmap-Objekte gezeichnet.

Die Kartenelemente werden in der `onDraw(Canvas canvas)` - Methode des Hauptpanels gezeichnet. Der Parameter ist eine Instanz von `Canvas`. `Canvas` ist eine „Leinwand“, auf der verschiedene 2D-Objekte auf dem Bildschirm gezeichnet werden können. `Canvas` enthält `draw`-Methoden zum Zeichnen von Bitmaps, Linien (`drawLine(...)`), Kreisen (`drawCircle(...)`), Rechtecken (`drawRect(...)`), Paths (`drawPath(...)`), etc. Jede `draw`-Methode braucht außer den 2D-Objekten selbst oder deren Koordinaten auch ein `Paint`-Objekt als Parameter. Das `Paint`-Objekt enthält Informationen darüber, wie die geometrischen Objekte (Rechtecke, Polygone), Texte und Bilder gezeichnet werden sollen, welche Farbe, Form und Stil sie haben.

## 4.6. Zusammenfassung

In diesem Kapitel wurde ein grober Entwurf eines Navigationssystems für Fußgänger präsentiert. Folgende Themengebiete wurden betrachtet:

- Gebäudemodell: Hier wurde ein standardisiertes Format - CityGML - für die Beschreibung von Karten und Gebäudeplänen vorgestellt. CityGML ist XML-basiert und erlaubt Beschreibung topologischer Objekte auf unterschiedlichen Detail-Levels. Insgesamt gibt es fünf verschiedene Genauigkeitsstufen. Der vierte Level LOD4 umfasst auch die Landmarken und kann somit für die Fußgängernavigation benutzt werden.
- Anwendungsarchitektur: soll als Basis für spätere Realisierung dienen. Sie ist eine Schichten-Architektur und besteht aus mehreren geschlossenen Modulen, die über ihre Schnittstellen miteinander kommunizieren.
- Client-Server-Architektur: Die gesamte Applikation soll auf dem Client laufen. Die LBS- und Karten-Server werden zusätzlich auf den Servern abgelegt.
- Framework für die Entwicklung mobiler Anwendungen und Emulation mobiler Geräte: dieser Teil beschreibt einen Android-Framework, der Bibliotheken anbietet, welche Design und Entwicklung unterschiedlicher Anwendungen für mobile Geräte in Java unterstützen.

- Realisierung des Prototyps: Zur Unterstützung der Durchführung des im Kapitel 5 beschriebenen Tests wurde ein Software-Prototyp in Java entwickelt. Der Prototyp kann die Umgebung maßstabsgetreu in 2D darstellen. Es wird auch die Route von dem Start bis zu dem Zielpunkt angezeigt. Die Anwendung berücksichtigt Landmarken und stellt diese als Icons auf dem Bildschirm dar. Bei der Änderung der Richtung, also beim Abbiegen nach links oder rechts dreht sich die Darstellung der Karte entsprechend, so dass die Karte immer aus der Benutzer-Perspektive angezeigt wird.

Die Karte, die in einem XML-basierten Format abgelegt ist, wird beim Starten der Anwendung aus dem Speicher gelesen und geparkt. Die Informationen wie Position und Art von Landmarken, Raumtyp, etc., die beim Parsen der Karten-Datei herausgefiltert werden, werden für die Darstellung und für die Navigationsanweisungen benutzt.

Die Positionsbestimmung und Routenberechnung wurden in den Prototyp nicht einbezogen. Der Grund dafür ist die Spezifikation des Untersuchungsschwerpunktes, der sich mit der Ergonomie eines Navigationssystems für Fußgänger und mit der Rolle der Landmarken beschäftigt.

## 5. Evaluation

In oberen Kapiteln [2](#) und [3](#) wird mehrmals darauf hingewiesen, dass Landmarken „die Eckpfeiler der Navigation“ [\[35\]](#) sind und die natürlichere Form der Orientierung für Menschen darstellen. Daraus folgt, dass ein Navigationssystem für Fußgänger nicht auf Landmarken verzichten darf, wenn es einfach in der Handhabung sein soll und die Akzeptanz der Benutzer gewinnen will.

Um diese Aussagen zu bestätigen wurde ein Evaluationstest durchgeführt. Dieser Test sollte zur Untersuchung folgender Fragen dienen:

- inwieweit Landmarken die Orientierung und Navigation mittels eines Navigationssystems einfacher machen;
- welche Landmarken sind am besten geeignet;
- wie werden Landmarken am besten auf der Anzeige präsentiert.

### 5.1. Test-Szenario: Informationstage an der HAW Hamburg

Jedes Jahr werden an der Hochschule für angewandte Wissenschaften (HAW) in Hamburg Informationstage veranstaltet. Das Ziel dabei ist es, dass die Abiturienten und andere Interessenten die Fachbereiche der Hochschule kennenlernen und sich mit den anderen Informationen rund um das Studium an der HAW vertraut machen. Dabei werden in den Fachbereichen verschiedene Vorträge und Präsentationen in unterschiedlichen Gebäuden und Räumen durchgeführt, die die Interessenten besuchen können.

Das Test-Szenario spielt sich im Gebäude des Informatik-Departments ab und soll den Verlauf eines Informationstages simulieren. Die Teilnehmer der Info-Tage haben eine Reihe von Veranstaltungen ausgesucht, die sie besuchen möchten. Sie kennen das vierzehnstöckige Gebäude nicht und brauchen eine Navigationshilfe, um sich im Gebäude zurecht zu finden, und bei allen Veranstaltungen dabei sein zu können, ohne die Räume lange zu suchen.



Abbildung 5.1.: künstliche Landmarken

### 5.1.1. Durchführung des Tests

Für den Test wurde eine Umgebung im elften Stock des Informatik-Departments der HAW Hamburg aufgebaut, die ein komplexes Gebäude simulieren sollte. Dafür wurden in den Fluren künstliche Landmarken aufgestellt (Abbildung 5.1), die unterschiedliche Räume darstellten und als Orientierungshilfe dienen sollten. Die Testperson sollte mit Hilfe der Navigationsanweisungen vom Software-System ein vorbestimmtes Ziel erreichen, indem sie von einer Landmarke zu der anderen geht. Dabei nutzte das System Landmarken, die in die Anweisungen und in die Präsentation eingebaut sind.

Zum Durchführen des Tests wurde außer aufgestellten Landmarken ein System zum Verfolgen der Blicke der Testpersonen aufgebaut, das unten beschrieben wird.

Das mobile Gerät wird durch den im Kapitel 4.4 beschriebenen Emulator simuliert. Die Anwendung, die ein Prototyp eines Navigationssystems ist, läuft auf dem Android-Emulator. Der Prototyp lädt den Gebäudeplan für den elften Stock, der im XML-Format gespeichert ist,

und präsentiert diesen in 2D auf dem Emulator-Bildschirm. Die Landmarken werden dabei als Icons dargestellt (Abbildung 5.2). Bei jedem weiteren Schritt erscheint eine Navigationsanweisung, die den entsprechenden Abschnitt der Route beschreibt.



Abbildung 5.2.: Darstellung der Karte mit Landmarken

Die Emulator-Software läuft auf dem Laptop, der auf einem Gestell platziert ist. Vor dem Laptop steht der Eyetracker (Abbildung 5.3), der die Blicke der Testpersonen verfolgen soll.



Abbildung 5.3.: Tobii Eye Tracker

Aus technischen Gründen, da die Evaluierungssoftware vom Eyetracker nur die auf einem Bildschirm dargestellten Sachen auswerten kann, wird die Realität durch eine Webcam auf

dem Laptop-Bildschirm projiziert. Dadurch kann man die Interaktion „Software-Benutzer-Realität“ beobachten, da die Darstellung der Applikation und die Abbildung der Realität sich auf einem Bildschirm befinden und der Eyetracker die Blicke auf den Bildschirm verfolgen kann. Somit soll beobachtet werden, inwieweit Landmarken der Fußgänger-Navigation behilflich sind und wie Benutzer auf die Anweisungen mit eingeschlossenen Landmarken reagieren. Der Gestell, auf dem sich die ganze Konstruktion Webcam-Laptop-Eyetracker befindet, wird auf einen Rolltisch aufgestellt, damit man das aufgebaute System bewegen kann (siehe Abbildung 5.4). Der zusätzliche Gestell wird benutzt, damit der Eyetracker und der Bildschirm des Laptops entsprechend hoch positioniert sind. Die Augen der Testperson können dabei erfasst werden, ohne dass die Person sich beugen oder setzen muss.



Abbildung 5.4.: Infrastruktur-Aufbau für den Evaluierungstest

## 5.2. Auswertung

Bei der Durchführung des Tests haben drei Personen teilgenommen - zwei Bachelor-Studenten und eine Master-Studentin der HAW Hamburg. Bei allen Personen hat man das gleiche Verhalten beobachtet, das unten beschrieben wird.

Nachdem die Testperson die Navigationsanweisung gelesen hat, hat sie zuerst die entsprechende Landmarke auf dem Bildschirm des mobilen Gerätes gesucht (Abbildung 5.5).



Abbildung 5.5.: Suchen der Landmarke auf dem Bildschirm des mobilen Gerätes (der rote Punkt zeigt die Fokussierung der Augen, die rote Linien zeigen Augenbewegungen)

Danach ändert die Testperson die Blickrichtung und sucht die Landmarke in der realen Umgebung (Abbildungen 5.6; 5.7).

Als nächstes schaut die Testperson noch einmal auf die Anwendungsanzeige, um sich zu vergewissern, dass die Landmarke, die sie gefunden hat, der Landmarke auf dem Bildschirm entspricht. Dieses Schauen zur Anwendung und dann zur Landmarke und zurück passiert mehrmals (Abbildung 5.8).

Anhand der Bilder, die der Eyetracker lieferte und die als Nachweis für die Ergebnisse benutzt werden können, und durch das Verhalten der Personen bei der Durchführung des Tests und ihre Aussagen danach, lässt sich beweisen, dass Landmarken eine gute Orientierungshilfe sind. Außerdem berichteten die Testpersonen, dass sie bei der Führung keine weitere Orientierungshilfe gebraucht haben, als die durch die Anwendung beschriebenen Landmarken. Somit hat der Test bestätigt, dass die Fußgänger-Navigationssysteme, wenn sie eine einfache und bequeme Benutzung anstreben, Landmarken in ihre Navigationsanweisungen

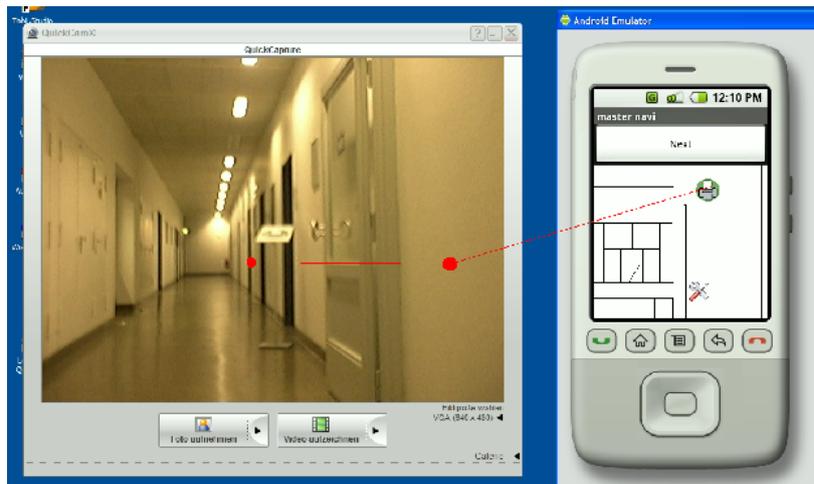


Abbildung 5.6.: Suche Landmarke in der realen Umgebung (a)

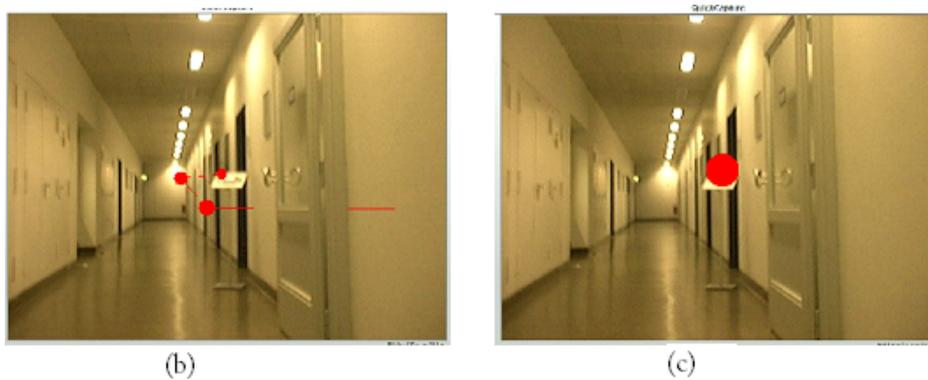


Abbildung 5.7.: Suche Landmarke in der realen Umgebung

sowie in die Darstellung der Umgebung einbeziehen sollen. Die Darstellung von Landmarken als Icons wurde als gut und verständlich empfunden. Jedoch sollten die Icons entweder allgemein verwendete Zeichen sein (z.B. Apotheken-Zeichen oder McDonald's-Logo) oder die präsentierten Landmarken sofort erkennen lassen.



Abbildung 5.8.: Interaktion zwischen dem Bildschirm und Realität

### 5.3. Zusammenfassung

In dem oben beschriebenen Test wurde eine Simulation der Fittage an der HAW Hamburg durchgeführt. Dabei mussten die Testpersonen den Anweisungen des Software-Prototyps folgen und sich mit Hilfe von diesen navigieren lassen. Als Orientierungshilfe wurden in den Fluren des elften Stockwerkes künstliche Landmarken aufgestellt, die auch in den Navigationsanweisungen der Anwendung vorgekommen sind.

Der Test wurde durchgeführt, um die Aussagen in den Kapiteln 2 und 3 zu prüfen, dass Landmarken eine wichtige Rolle bei der menschlichen Orientierung spielen und für die Fußgänger-Navigation unentbehrlich sind. Die Ergebnisse der Evaluation haben gezeigt, dass diese Aussagen richtig sind und Landmarken ein wichtiger Bestandteil eines einfach benutzbaren Navigationssystems für Fußgänger sein sollen.

## 6. Methodische Abstraktion

Der Ansatz von Navigation mittels Landmarken kann auch auf andere Szenarien nutzbringend übertragen werden. In konkret betrachteten Szenarien (siehe 3.1 und 5.1) geht es um ein Navigationssystem für Fußgänger im Indoor-Bereich. Das Anwendungsgebiet kann durchaus auf den Outdoor-Bereich erweitert werden. Outdoor-Bereiche haben eine komplexe Struktur, die einen Netz von Straßen, Plätzen, Wasserbereichen etc. darstellt. Sie bieten aber eine Vielfalt an Landmarken, die zum Navigieren eingesetzt werden können.

Ein weiteres Einsatzgebiet für die landmarkenbasierte Navigation ist das Rescue-Szenario. Navigationssysteme werden bereits für die Lokalisierung von Unfällen und Opfern benutzt. Die Landmarken-Navigation kann den Rettungsteams helfen, sich in den Umgebungen mit eingeschränkter Sicht besser zu orientieren und damit Rettungsaktionen schneller durchzuführen.

Weitere Potentiale in der Verwendung können solche Navigationssysteme bei dem Militär finden. Soldaten können sich mit Hilfe von Navigationssystemen lokalisieren, sammeln und Informationen austauschen/übermitteln. Landmarken helfen dabei wieder zur besseren Orientierung, aber sie liefern dabei auch Informationen über die Umgebung selbst.

Für die Navigation von sehbehinderten Menschen ist der Ansatz, Landmarken zu verwenden, auch von Bedeutung. Hier könnten statt visuellen Landmarken, Landmarken eingesetzt werden, die andere Sinnesorgane ansprechen. Unterschiedliche Gerüche oder Geräusche könnten z.B. als Landmarken dienen.

Die Auto-Navigation würde auch vom Einsatz von Landmarken profitieren. Dabei könnten Landmarken weniger in den Navigationsanweisungen auftreten, sondern viel mehr zusätzliche Informationen über die Umgebung liefern, wie z.B. die Position der nächsten Tankstelle bzw. einer Übernachtungsmöglichkeit oder eines Restaurants.

## 7. Zusammenfassung

In dieser Arbeit wurde untersucht, auf welche natürliche menschliche Orientierungshilfe sich ein Navigationssystem für Fußgänger stützen soll, damit es einfacher und bequemer in der Benutzung wird. Dabei wurde ein Test mit der Unterstützung eines Software-Prototyps durchgeführt, der die Behauptungen aus den Anfangskapiteln dieser Arbeit bestätigen sollte.

Fußgänger-Navigation kann sich nicht auf die bereits weit entwickelte Auto-Navigation stützen, da sie ganz anderen Voraussetzungen unterliegt wie z.B. Tageszeit, Ziel der navigierten Person (Wanderung, schnelles Ankommen etc.). Dabei ist es wichtig, das natürliche Verhalten der Menschen bei ihrer Navigation zu berücksichtigen.

Zahlreiche Studien aus unterschiedlichen Zeiträumen belegen, dass Menschen bei der Wegsuche Landmarken als Orientierungshilfe nutzen. Nach einer Definition werden als Landmarken jegliche Orientierungspunkte im Raum bezeichnet [9]. Man nutzt sie, um einerseits eigene so genannte räumliche Karte zu strukturieren und andererseits als Navigationshilfe bei der Wegsuche. Nicht jedes Objekt kann eine Landmarke sein. Die Eigenschaften, die ein Objekt zu einer Landmarke aufsteigen lassen, sind seine besondere visuelle Charakteristik (auffälliges Aussehen), seine einzigartige Bedeutung oder seine herausragende Lage. Bei der Wegsuche werden Landmarken an allen wichtigen Punkten eingesetzt: am Start, Ziel und an den Entscheidungspunkten.

Aus den Untersuchungen der menschlichen Orientierung lässt sich feststellen, dass man auch bei der Navigation, die durch die Software unterstützt wird, auf Landmarken nicht verzichten darf. Das muss auch in den Anforderungen an das Software-System festgehalten werden.

Als wichtige nicht-funktionale Anforderungen an ein einfach benutzbares Fußgänger-Navigationssystem haben sich Benutzbarkeit, Verständlichkeit, Zuverlässigkeit und Übertragbarkeit erwiesen. Eine besondere Beachtung erhielten Aspekte der Ergonomie. Dazu gehören unter anderem die Sprach-Ausgabe und Vibration des Gerätes als Meldung an den Benutzer. Sicherheit und Effizienz sind auch wichtige Aspekte, gehören aber nicht zu dem Schwerpunkt dieser Arbeit und wurden vernachlässigt.

Basierend auf den nicht-funktionalen und den in den Anwendungsfällen abgebildeten funktionalen Anforderungen an die Anwendung wurde ein Entwurf der System-Architektur erstellt. Die Architektur berücksichtigt Landmarken, die bei der Erstellung des Kartenmodells

aus einer XML-basierten Datei extrahiert und in die Routenberechnung einbezogen werden. Außerdem können Landmarken aus LBS-Informationen gewonnen werden.

Ein standardisiertes Format für die Kartenbeschreibung, das zusätzlich zu den Informationen von Gebäuden bzw. Räumen auch Informationen über mögliche Landmarken erhält, ist CityGML. CityGML ist zwar ein Modell für eine 3D-Darstellung von städtischen Objekten, hat sich aber in dieser Arbeit als ein geeignetes Modell für eine 2D-Beschreibung von Gebäuden und Indoor-Bereichen erwiesen.

Die Zielsetzung der Arbeit war nicht die Entwicklung eines Navigationssystems, sondern die Untersuchung, welchen Einfluss Landmarken auf Fußgänger-Navigation haben. Jedoch wurde zur Unterstützung des Evaluierungstests, der diesen Einfluss untersuchen sollte, ein Software-Prototyp gebaut. Die Anwendung stellte ein einfaches Navigationssystem dar, das einen Gebäudeplan und Landmarken angezeigt hat. Für jeden Orientierungsschritt lieferte die Applikation eine Navigationsanweisung und leitete somit die Testperson an das Ziel. Aus technischen Gründen sollte die ganze Anwendung auf einem Emulator des mobilen Gerätes laufen. Der Eyetracker, der die Bewegungen der Augen verfolgt hat, um die Reaktion der Testpersonen auf Landmarken zu betrachten, konnte nur die Blicke auswerten, die sich auf den Computerbildschirm richteten.

Die Ergebnisse des Evaluierungstests (siehe Kapitel 5) haben gezeigt, dass ein Software-Navigationssystem, das sich auf Landmarken stützt, eine natürlichere Orientierungshilfe darstellt als ein auf Entfernungen und Richtungen basiertes System. Außerdem lassen die Ergebnisse der Testuntersuchung erkennen, dass in einer homogenen Umgebung, wie z.B. ein Gebäude mit langen Fluren und unterschiedlichen Räumen, künstliche Landmarken sich als ein gutes Hilfsmittel für die Wegsuche erwiesen und diese wesentlich erleichtern. Der Test hat auch gezeigt, dass die Präsentation von Landmarken als Icons von den Benutzern akzeptiert wird.

Um die Informationen über Landmarken in ein System einzubauen, müssen diese erst gesammelt werden. Zurzeit existieren keine ausreichenden Datensammlungen von Landmarken für städtische Objekte. Es existieren zwar Methoden zur Identifizierung optimaler Landmarken [30], sie sind aber meistens nicht automatisiert (außer Data-Mining [9]) und dazu noch sehr ressourcenintensiv. Solche Methoden wurden in Rahmen von Forschungen und Experimenten gemacht und sind noch nicht industriereif. Dazu erschwert die Kontextabhängigkeit die Erfassung von Landmarken, da die Randbedingungen wie Alter, Tageszeit berücksichtigt werden müssen. Noch existieren kaum elektronische Sammlungen von Landmarken (besonders für größere Umgebungen). Möglichkeiten zur automatischen Erfassung existieren noch kaum. Eine Erweiterung des Themengebiets dieser Arbeit, die die Möglichkeit anbietet, Landmarken zu sammeln, stellt eine benutzergesteuerte Annotation von Landmarken dar. Benutzer von Navigationssystemen könnten ihnen aufgefallene Objekte mit der Handy-Kamera fotografieren und diese Objekte auf die Karte in ihrem System setzen. So hätten sie

die neuen Landmarken lokal ins System aufgenommen. Um neue Landmarken den anderen Benutzern zu übermitteln, könnte man die gewonnenen Informationen und Erfahrungen in einer Community im Internet austauschen. Die ersten Ansätze, die es ermöglichen, Routeninformationen aufzunehmen oder zu ändern, gibt es bereits (z.B. TomTom Go). Sie sind aber für die Auto-Navigation entwickelt worden und nicht an die Bedürfnisse von Fußgängern angepasst.

Es könnten auch andere Aspekte wie Audio-Ausgabe und technische Eigenschaften der mobilen Geräte wie Vibration extra in Betracht gezogen werden. Besonders Audio-Ausgabe wäre ein Vorteil für ein einfach benutzbares Navigationssystem. Sie hätte den Benutzer vom ständigen Schauen auf den Bildschirm befreit, und er könnte sich auf andere Aufgaben besser konzentrieren.

Das moderne Computing hat das Ziel, allgegenwärtig und gleichzeitig möglichst unbemerkbar für Benutzer zu sein (ubiquitous computing). Mobile Anwendungen, verteilte Systeme, Personalisierung und Anpassung an den Menschen treten dabei in den Vordergrund auf. Man möchte die Software-Bedienung näher an die natürlichen Bedürfnisse der Menschen bringen. Diese Arbeit trägt dazu bei, indem sie die Rolle von Landmarken für die menschliche Orientierung untersucht und wie Landmarken in einem mobilen Fußgänger-Navigationssystem verwendet werden, damit Benutzer das System ganz natürlich bedienen können.

# Literaturverzeichnis

- [1] Android homepage url - <http://code.google.com/android/documentation.html>.
- [2] Citymaps homepage url - <http://www.teleatlas.com>.
- [3] Navigon: Navigon 7100. - url <http://www.navigon.com/>.
- [4] I. Aslan. The bum bag navigator: A configurable mobile multi-purpose navigation system for pedestrians. Diplomarbeit, Universität Saarland, 2004.
- [5] I. Aslan and A. Krüger. The bum bag navigator (bbn): An advanced pedestrian navigation system. 2004.
- [6] I. Aslan, M. Schwalm, J. Baus, A. Krüger, and T. Schwartz. Acquisition of spatial knowledge in location aware mobile pedestrian navigation systems. In *Mobile HCI*, pages 105–108, 2006.
- [7] N. J. Bidwell, C. Lueg, and J. Axup. The territory is the map: designing navigational aids. In *CHINZ '05: Proceedings of the 6th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction*, pages 91–100, New York, NY, USA, 2005. ACM.
- [8] B. Brunner-Friedrich. Modellierung und Kommunikation von Active Landmarks für die Verwendung in Fußgängernavigationssystemen, 2003.
- [9] B. Elias. Extracting landmarks with data mining methods. Kartause Ittingen, Switzerland, 2003.
- [10] B. Elias and M. Sester. *Landmarks für Routenbeschreibungen*, 2003.
- [11] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Entwurfsmuster*. Addison-Wesley, 2004.
- [12] L. E. Gerharz and H. J. Müller. Usability of user adapted indoor walking descriptions. Universität Münster - Institut für Geoinformatik, 2006.
- [13] M. Glinz. *Nicht-funktionale Anforderungen*, 2006.

- 
- [14] S. Gregor. Entwicklung einer Hardwareplattform für die Ermittlung von Positionsdaten innerhalb von Gebäuden. Bachelorthesis, Hochschule für Angewandte Wissenschaften Hamburg, 2006.
- [15] G. Gröger, T. H. Kolbe, and A. Czerwinski. Candidate.opengis® citygml implementation specification (city geography markup language). Technical report, Open Geospatial Consortium, Inc., 2007.
- [16] E. Jonsson. *Der innere Kompass*. Walter, 2004.
- [17] K. Lynch. *Das Bild der Stadt*. Ullstein, 1965.
- [18] R. Marrone. *Software-Engineering*, 2008.
- [19] A. May, T. Ross, S. H. Bayer, and M. Tarkiainen. Pedestrian navigation aids: information requirements and design implications. *Personal und Ubiquitous Computing*, 7(6), Dezember 2003.
- [20] A. Millonig and K. Schechtner. Developing landmark-based pedestrian-navigation systems. *IEEE Transactions on Intelligent Transportation Systems*, 8(1), 2007.
- [21] J. Napitupulu. Indoor Map Server in einem Flughafenszenario. Projektbericht WS06/07, 2007.
- [22] J. Napitupulu. Multimediale Fluggastführung. Seminararbeit WS06/07, 2007.
- [23] U. Neisser. *Kognition und Wirklichkeit*. Klett-Cotta, 1979.
- [24] B. Oestereich. *Analyse und Design mit UML 2.1*. Oldenbourg Wissenschaftsverlag, 2006.
- [25] T. Pfaff. Entwicklung eines PDA-basierten Indoor-Navigationssystems. Bachelorthesis, Hochschule für Angewandte Wissenschaften Hamburg, 2007.
- [26] T. Posch, K. Birken, and M. Gerdorf. *Basiswissen Softwarearchitektur: Verstehen, entwerfen, bewerten und dokumentieren*. dpunkt Verlag, 2004.
- [27] C. J. Quix. Metadatenverwaltung zur qualitätsorientierten Informationslogistik in Data-Warehouse-Systemen. Doktorarbeit, Rheinisch-Westfälischen Technischen Hochschule Aachen, 2003.
- [28] K.-H. Rau. *Objektorientierte Systementwicklung: Vom Geschäftsprozess zum Java-programm*. Vieweg+Teubner Verlag, 2007.
- [29] S. Raubal and S. Winter. Enriching wayfinding instructions with local landmarks. *Geographic Information Science. Lecture Notes in Computer Science*, 2478:243–259, 2002.

- [30] R. Sefelin, M. Bechinie, R. Müller, V. Seibert-Giller, P. Messner, and M. Tscheligi. Landmarks: yes; but which?: five methods to select optimal landmarks for a landmark- and speech-based guiding system. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 287–290, New York, NY, USA, 2005. ACM.
- [31] S. Shankland. Sun starts bidding adieu to mobile-specific java. *CNet*, 2007.
- [32] M. Skubic, S. Blisard, C. Bailey, J. Adams, and P. Matsakis. Qualitative analysis of sketched route maps: translating a sketch into linguistic descriptions. *Systems, Man, and Cybernetics, Part B, IEEE Transactions on Intelligent Transportation Systems*, 34(2):1275–1282, 2004.
- [33] G. Steffen. Design einer Mitfahrer Börse auf Basis graphentheoretischer Verfahren. Diplomarbeit, Hochschule für Angewandte Wissenschaften Hamburg, 2004.
- [34] A. S. Tanenbaum and M. van Steen. *Verteilte Systeme*. Pearson Studium, 2003.
- [35] M. Tscheligi and R. Sefelin. Mobile navigation support for pedestrians: can it work and does it pay off? *Interactions*, 13(4):31–33, 2006.
- [36] B. Tversky and P. Lee. Pictorial and verbal tools for conveying routes. In *COSIT '99: Proceedings of the International Conference on Spatial Information Theory*, pages 51–64, Berlin, Germany, 1999. Springer-Verlag.
- [37] S. Werner, B. Krieg-Bruckner, H. A. Mallot, K. Schweizer, and C. Freksa. Spatial cognition: The role of landmark, route, and survey knowledge in human and robot navigation. In *GI Jahrestagung*, pages 41–50, 1997.
- [38] S. Winter and A. Klippel. Structural salience of landmarks for route directions. In *COSIT '05: Proceedings of the International Conference on Spatial Information Theory*, pages 347–362, Berlin, Germany, 2005. Springer-Verlag.
- [39] S. Winter, M. Raubal, and C. Nothegger. Focalizing measures of salience for wayfinding. *Map-Based Mobile Services - Theories, Methods and Design Implementations, Springer Geosciences*, 2004.
- [40] H. Zuellighoven, D. Bäumer, W. Bleek, and C. Lilienthal. *Das objektorientierte Konstruktionshandbuch*. Dpunkt Verlag, 1998.
- [41] O. Zukunft and J. Raasch. *Architekturen von Informationssystemen*, 2003.

# A. Anhang

## A.1. Anwendungsfälle

<b>Nr.:1</b> ein POI suchen	
Zusammenfassung	Ein POI (point of interest) wird gesucht
Vorbedingung	Eine Wissensbasis soll vorhanden sein, um eine Suchanfrage machen zu können.
Beschreibung des Ablaufs	Der Benutzer gibt das gesuchte Objekt über die Tastatur ein oder wählt es aus einer Liste möglicher POIs aus.
Ausnahmen	<b>E1:</b> Wenn die Verbindung zum Server nicht stattfinden konnte, soll dies dem Benutzer mitgeteilt werden.
Nachbedingungen	<b>N1:</b> Die Suchergebnisse müssen angezeigt werden können.

<b>Nr.:2</b> Suchergebnisse anzeigen	
Zusammenfassung	Sämtliche Suchergebnisse werden auf dem Bildschirm angezeigt.
Vorbedingung	Die Suche von POIs (Nr.1) ist ohne Probleme durchgelaufen.
Beschreibung des Ablaufs	Sobald die Suchergebnisse vorhanden sind, können sie angezeigt werden.
Ausnahmen	<b>E1:</b> Wenn die Verbindung zum Server nicht stattfinden konnte, soll dies dem Benutzer mitgeteilt werden. <b>E2:</b> Ist kein Ergebnis gefunden worden, ist dies dem Benutzer mitzuteilen.
Nachbedingungen	<b>N1:</b> Ein Ziel soll gewählt werden können.

<b>Nr.:3 Ziel wählen</b>	
Zusammenfassung	Ein Ziel wird aus der Ergebnisliste ausgewählt.
Vorbedingung	Die Suche von POIs ist abgeschlossen und die Ergebnisliste ist auf dem Bildschirm angezeigt.
Beschreibung des Ablaufs	Der Benutzer hat sich für ein Ziel entschieden und wählt dieses per Tastendruck aus.
Ausnahmen	<b>E1:</b> Wenn die Verbindung zum Server nicht stattfinden konnte, soll dies dem Benutzer mitgeteilt werden. <b>E2:</b> Falls die Verbindung zum Positionsbestimmungsmodul ausfällt, so ist dies dem Benutzer mitzuteilen.
Nachbedingungen	<b>N1:</b> Die Route zu dem Zielort soll berechnet werden können.

<b>Nr.:4 Route darstellen</b>	
Zusammenfassung	Die Route zum Ziel wird berechnet und auf der Anzeige dargestellt.
Vorbedingung	Ein Ziel muss ausgewählt worden sein.
Beschreibung des Ablaufs	Die Route wird durch das System berechnet und auf dem Bildschirm präsentiert.
Nachbedingungen	<b>N1:</b> Die Route wird angezeigt.

<b>Nr.:6 Landmarken anzeigen</b>	
Zusammenfassung	Landmarken werden auf der Anzeige dargestellt.
Vorbedingung	Eine Zusammenfassung möglicher Landmarken existiert und Landmarken sind in die Routenberechnung einbezogen worden.
Beschreibung des Ablaufs	Landmarken werden als Icons auf der dargestellten Route angezeigt.
Ausnahmen	<b>E1:</b> Falls Landmarken nicht bei der Routenberechnung berücksichtigt worden sind, werden sie nicht angezeigt.
Nachbedingungen	<b>N1:</b> Die Route samt Landmarken wird angezeigt.

<b>Nr.:7 Orientierungshilfen darstellen</b>	
Zusammenfassung	Orientierungshilfen wie Maßstab, Entfernung zum Ziel, Abzweigungshinweise sowie Nordausrichtung werden auf dem Bildschirm dargestellt.
Vorbedingung	Die Route wird dargestellt.
Beschreibung des Ablaufs	Zusammen mit der Route werden auf der Karte Orientierungshilfen angezeigt.

<b>Nr.:5</b> Landmarken in die Routenberechnung einbeziehen	
Zusammenfassung	Landmarken werden bei der Routenberechnung berücksichtigt.
Vorbedingung	Eine Zusammenfassung möglicher Landmarken existiert.
Beschreibung des Ablaufs	berechnete Route wird in die Sektionen aufgeteilt und Landmarken werden als Verbindungsknoten in diesen Sektionen eingesetzt.
Ausnahmen	<b>E1:</b> Gibt es keine Sammlung von Landmarken, wird die Route ohne diese berechnet.
Nachbedingungen	<b>N1:</b> Die Route samt Landmarken wird angezeigt.

<b>Nr.:8</b> Navigationsanweisungen zu der Route darstellen	
Zusammenfassung	Zur eindeutigen Beschreibung der Route und Navigationshilfe werden textliche Anweisungen angezeigt.
Vorbedingung	Die Karte mit der Route ist angezeigt.
Beschreibung des Ablaufs	Während der Navigation wird die grafische Anzeige durch die textliche Beschreibung der Route erweitert.
Nachbedingungen	<b>N1:</b> Textliche Beschreibung der Route wird angezeigt.

<b>Nr.:9</b> Karte zoomen	
Zusammenfassung	Die dargestellte Karte kann vergrößert oder verkleinert werden.
Vorbedingung	Die Karte ist auf dem Bildschirm dargestellt.
Beschreibung des Ablaufs	Die Karte und ihre angezeigten Elemente wie Landmarken können mittels Tasten vergrößert oder verkleinert werden.
Nachbedingungen	<b>N1:</b> Die Karte und ihre Elemente werden größer oder kleiner angezeigt.

<b>Nr.:10</b> Karte verschieben	
Zusammenfassung	Die Karte kann in vier verschiedene Richtungen verschoben werden.
Vorbedingung	Die Karte ist auf dem Bildschirm dargestellt.
Beschreibung des Ablaufs	Die Karte kann mittels vier Navigationstasten entsprechend nach links, rechts, unten oder oben verschoben werden.
Nachbedingungen	<b>N1:</b> Die Kartenansicht ist verschoben.

<b>Nr.:11</b> Anwendung beenden	
Zusammenfassung	Der Benutzer beendet die Anwendung.
Vorbedingung	Die Anwendung muss laufen und Benutzer soll das Beenden des Systems initiieren.
Beschreibung des Ablaufs	Der Benutzer beendet mittels Tastendruck die Anwendung. Alle Verbindungen zu den entsprechenden Diensten (LBS, Positionsmodul) werden getrennt.
Nachbedingungen	<b>N1:</b> Die Anwendung ist beendet.

## A.2. Gebäudemodell in CityGML

```
<?xml version="1.0" encoding="utf-8"?>
<CityModel xmlns="http://www.citygml.org/citygml/1/0/0"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
...

</gml:description>
<gml:name>IFC_Building_Variant</gml:name>
<gml:boundedBy>
<gml:Envelope srsName="EPSG:31467">
<gml:coord>
<gml:X>265.34323</gml:X>
<gml:Y>58.2458</gml:Y>
<gml:Z>0</gml:Z>
</gml:coord>
<gml:coord>
<gml:X>2817.15473</gml:X>
<gml:Y>5684.0866</gml:Y>
<gml:Z>0</gml:Z>
</gml:coord>
</gml:Envelope>
</gml:boundedBy>
<cityObjectMember>
<Building gml:id="GEB_TH_IFC_Building_Variant_GEB_0">
<gml:description>" "</gml:description>
<gml:name>Default Building</gml:name>
<consistsOfBuildingPart>
```

```
<BuildingPart gml:id="GEB_TH_IFC_Building_Variant_GEB_TEIL_1">
  <gml:description?></gml:description>
  <gml:name>HAW</gml:name>
  <yearOfConstruction>0001</yearOfConstruction>
  <roofType>1030</roofType>
  <measuredHeight uom="#m">0</measuredHeight>
  <storeysAboveGround>2</storeysAboveGround>
  <storeyHeightsAboveGround uom="#m">2.5 2.5</storeyHeightsAboveGround>
  <interiorRoom>
    <Room>
      <lod4MultiSurface>
        <gml:MultiSurface>
          <gml:surfaceMember>
            <TexturedSurface orientation="+">
              <gml:baseSurface>
                <gml:Polygon>
                  <gml:exterior>
                    <gml:LinearRing>
                      <gml:pos srsDimension="2">0 0</gml:pos>
                        <gml:pos srsDimension="2">270.98 63.884</gml:pos>
                        <gml:pos srsDimension="2">270.98 972.83</gml:pos>
                        <gml:pos srsDimension="2">1077.63 972.83</gml:pos>
                        <gml:pos srsDimension="2">1077.63 63.884</gml:pos>
                        <gml:pos srsDimension="2">270.98 63.884</gml:pos>
                    </gml:LinearRing>
                  </gml:exterior>
                </gml:Polygon>
              </gml:baseSurface>
            <appearance>
              <SimpleTexture>
                <textureMap>
                  ...
                </textureMap>
                <textureCoordinates> 704.07697 1455.1486 0</textureCoordinates>
              </SimpleTexture>
            </appearance>
          </TexturedSurface>
        </gml:surfaceMember>
      </gml:MultiSurface>
    </lod4MultiSurface>
  </interiorRoom>
</BuildingPart>
```

```
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
<gml:Polygon>
<gml:exterior>
<gml:LinearRing>
<gml:pos srsDimension="2">0 0</gml:pos>
  <gml:pos srsDimension="2">270.98 63.884</gml:pos>
  <gml:pos srsDimension="2">270.98 972.83</gml:pos>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
</gml:baseSurface>
<appearance>
</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
<gml:Polygon>
<gml:exterior>
<gml:LinearRing>
<gml:pos srsDimension="2">270.98 972.83</gml:pos>
  <gml:pos srsDimension="2">1077.63 972.83</gml:pos>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
```

```
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
<gml:Polygon>
<gml:exterior>
<gml:LinearRing>
<gml:pos srsDimension="2">1077.63 972.83</gml:pos>
<gml:pos srsDimension="2">1077.63 63.884</gml:pos>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
```

```
<TexturedSurface orientation="+">
  <gml:baseSurface>
    <gml:Polygon>
      <gml:exterior>
        <gml:LinearRing>
          <gml:pos srsDimension="2">1077.63 63.884</gml:pos>
            <gml:pos srsDimension="2">270.98 63.884</gml:pos>
          </gml:LinearRing>
        </gml:exterior>
      </gml:Polygon>
    </gml:baseSurface>
  <appearance>

  </appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
</Room>
</interiorRoom>
<interiorRoom>
  <Room>
    <lod4MultiSurface>
      <gml:MultiSurface>
        <gml:surfaceMember>
          <TexturedSurface orientation="+">
            <gml:baseSurface>
              <gml:Polygon>
                <gml:exterior>
                  <gml:LinearRing>
                    <gml:pos srsDimension="2">262.52042 1760.3944</gml:pos>
                    <gml:pos srsDimension="2">262.52042 3208.491</gml:pos>
                    <gml:pos srsDimension="2">880.71372 3208.491</gml:pos>
                    <gml:pos srsDimension="2">880.71372 1760.3944</gml:pos>
                    <gml:pos srsDimension="2">262.52042 1760.3944</gml:pos>
                  </gml:LinearRing>
                </gml:exterior>
              </gml:Polygon>
```

```
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
<gml:Polygon>
<gml:exterior>
<gml:LinearRing>
<gml:pos srsDimension="2">262.52042 1760.3944</gml:pos>
<gml:pos srsDimension="2">262.52042 3208.491</gml:pos>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
```

```
<gml:Polygon>
  <gml:exterior>
    <gml:LinearRing>
      <gml:pos srsDimension="2">262.52042 3208.491</gml:pos>
      <gml:pos srsDimension="2">880.71372 3208.491</gml:pos>
    </gml:LinearRing>
  </gml:exterior>
</gml:Polygon>
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
  <InteriorWallSurface>
    <lod4MultiSurface>
      <gml:MultiSurface>
        <gml:surfaceMember>
          <TexturedSurface orientation="+">
            <gml:baseSurface>
              <gml:Polygon>
                <gml:exterior>
                  <gml:LinearRing>
                    <gml:pos srsDimension="2">880.71372 3208.491</gml:pos>
                    <gml:pos srsDimension="2">880.71372 1760.3944</gml:pos>
                  </gml:LinearRing>
                </gml:exterior>
              </gml:Polygon>
            </gml:baseSurface>
          <appearance>

          </appearance>
        </TexturedSurface>
      </gml:surfaceMember>
    </gml:MultiSurface>
```

```
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
<boundedBy>
<InteriorWallSurface>
<lod4MultiSurface>
<gml:MultiSurface>
<gml:surfaceMember>
<TexturedSurface orientation="+">
<gml:baseSurface>
<gml:Polygon>
<gml:exterior>
<gml:LinearRing>
<gml:pos srsDimension="2">880.71372 1760.3944</gml:pos>
<gml:pos srsDimension="2">262.52042 1760.3944</gml:pos>

</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
</gml:baseSurface>
<appearance>

</appearance>
</TexturedSurface>
</gml:surfaceMember>
</gml:MultiSurface>
</lod4MultiSurface>
</InteriorWallSurface>
</boundedBy>
</Room>
</interiorRoom>
</BuildingPart>
</consistsOfBuildingPart>
</Building>
</cityObjectMember>
</CityModel>
```

# Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §24(5) ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 23. April 2008

Ort, Datum

Unterschrift