



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

# Bachelorarbeit

Sascha Waltz

**Software-Bibliothek zur gerätebasierten und  
standortabhängigen Interaktion mit Public Displays**

*Fakultät Technik und Informatik  
Studiendepartment Informatik*

*Faculty of Engineering and Computer Science  
Department of Computer Science*

Sascha Waltz

**Software-Bibliothek zur gerätebasierten und  
standortabhängigen Interaktion mit Public Displays**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Wirtschaftsinformatik  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Kai von Luck  
Zweitgutachter: Prof. Dr. Philipp Jenke

Eingereicht am: 16. Februar 2016

**Sascha Waltz**

**Thema der Arbeit**

Software-Bibliothek zur gerätebasierten und standortabhängigen Interaktion mit Public Displays

**Stichworte**

Public-Displays, Interaktion, gerätebasiert, standortabhängig, Bibliothek

**Kurzzusammenfassung**

Für die Interaktion mit Public Displays sind die Authentifizierung und der Standort des Users wichtige Faktoren. Je nach Anwendung und Art der Interaktion variieren Wichtigkeit, Genauigkeit und Umfang der benötigten Informationen. In dieser Arbeit werden Konzepte und Implementationsmöglichkeiten zur Konzeption und Erstellung einer Softwarebibliothek mit Funktionen zur Feststellung der Identität und des Standortes des Users diskutiert. Es werden verschiedene Methoden und Technologien vorgestellt, untersucht und teilweise implementiert und getestet, sowie über die Evaluation der Implementation auf der Medienfassade der HAW-Hamburg am Campus Berliner Tor berichtet.

**Sascha Waltz**

**Title of the paper**

Software-Library for device-based and location-based interaction with public displays

**Keywords**

public-display, interaction, device-based, location-based, library

**Abstract**

To interact with public displays authentication and the location of the user are important factors. Depending on the application and type of interaction vary importance, accuracy and amount of information needed. In this work, concepts and implementation possibilities for designing and creating a software library with functions for establishing the identity and the location of the user will be discussed. There are different methods and technologies presented, examined and partially implemented and tested, as well as on the evaluation of the implementation on the media facade of the HAW Hamburg on campus reports Berliner Tor.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Zielsetzung . . . . .	2
1.3	Gliederung . . . . .	2
<b>2</b>	<b>Analyse</b>	<b>4</b>
2.1	Public Displays . . . . .	4
2.1.1	Digital Signage . . . . .	4
2.1.2	Einsatz Public Displays . . . . .	5
2.1.3	Bespielungszustände einer Medienfassade . . . . .	5
2.2	Grundlagen der Interaktion . . . . .	6
2.2.1	Definition einer Interaktion . . . . .	6
2.2.2	Ablauf . . . . .	6
2.3	Interaktion mit Public Displays . . . . .	7
2.3.1	Arten der Interaktion . . . . .	7
2.3.2	Phasen einer Interaktion . . . . .	9
2.3.3	Interaktion mit öffentlich Wahrnehmbarem . . . . .	9
2.3.4	Bedingungen für eine Interaktion . . . . .	10
2.4	Geräte basierte Interaktion . . . . .	11
2.4.1	Mögliche Arten von Geräten zur Interaktion . . . . .	11
2.4.2	Nutzbare Technologien mobiler Endgeräte . . . . .	11
2.4.3	Was passiert im Frontend? . . . . .	12
2.4.4	Was passiert im Backend? . . . . .	12
2.5	HAW-Hamburg - Medienfassade . . . . .	13
2.5.1	Mögliche Interaktionen . . . . .	13
2.5.2	Use-Cases auf dem HAW-Hamburg Campus Berliner Tor . . . . .	13
2.5.2.1	Posts . . . . .	13
2.5.2.2	Spiele . . . . .	14
2.5.2.3	Informationen für Studierende . . . . .	14
2.6	User-Authentifikation . . . . .	14
2.6.1	Grundlagen . . . . .	15
2.6.2	Authentifizierung mit Hochschulaccounts . . . . .	15
2.6.3	Was bedeutet Zurückverfolgbarkeit . . . . .	15
2.7	Standortbestimmung . . . . .	15
2.7.1	Grundlagen der Standortbestimmung . . . . .	15
2.7.2	GPS . . . . .	17

2.7.3	Mobilfunk . . . . .	18
2.7.4	WLAN . . . . .	21
2.7.5	Weitere Möglichkeiten . . . . .	23
2.7.6	Verfügbarkeit und Qualität . . . . .	24
2.7.7	Bewertung und Kombination . . . . .	24
2.7.7.1	Indoor . . . . .	24
2.7.7.2	Urban . . . . .	25
2.7.7.3	Rural . . . . .	25
2.8	Verwandte Arbeiten . . . . .	25
2.8.1	Diplomarbeit - Christian Epperlein . . . . .	25
2.8.2	Masterarbeit - Sven Boris Bornemann . . . . .	26
2.8.3	Großer Beleg - Florian Schneider . . . . .	26
2.8.4	Forschungsprojekt Medienfassade - Stephan Bergemann , Robin Schlegel . . . . .	26
2.9	Zusammenfassung . . . . .	26
2.9.1	Anforderungen . . . . .	27
<b>3</b>	<b>Entwurf der Bibliothek</b> . . . . .	<b>28</b>
3.1	Einleitung . . . . .	28
3.1.1	Charakteristiken von Bibliotheken . . . . .	28
3.1.1.1	Latenzen . . . . .	28
3.1.1.2	Programmiermethodik . . . . .	29
3.2	Grundlagen . . . . .	29
3.2.1	JavaScript . . . . .	29
3.2.2	Android Library . . . . .	29
3.2.3	Ortungsdienste . . . . .	31
3.3	Architektur-Übersicht . . . . .	31
3.3.1	Einleitung . . . . .	31
3.3.2	JavaScript . . . . .	31
3.3.3	User-Authentifizierung . . . . .	33
3.3.4	Designentscheidungen im Bezug auf Medienfassaden . . . . .	33
3.4	Erweiterbarkeit . . . . .	34
<b>4</b>	<b>Evaluation</b> . . . . .	<b>35</b>
4.1	Einleitung . . . . .	35
4.2	Evaluierung der Implementierung . . . . .	35
4.2.1	JavaScript . . . . .	35
4.2.2	Web Application . . . . .	36
4.2.3	Android WebApps . . . . .	37
4.2.4	Messwerte und Genauigkeit . . . . .	37
4.2.5	Messfehler . . . . .	40
4.2.6	Erleichterung der Programmierung . . . . .	40
4.2.7	Einbindung in Anwendungen . . . . .	41
4.3	Fazit . . . . .	41

<b>5</b>	<b>Schluss</b>	<b>42</b>
5.1	Zusammenfassung . . . . .	42
5.2	Ausblick . . . . .	43

# Tabellenverzeichnis

2.1	SPS Position/Time Service Accuracy Standards . . . . .	17
2.2	Verfügbarkeit und Qualität in Abhängigkeit von der Umgebung . . . . .	24
2.3	Abfragefunktionen der Bibliothek . . . . .	27
2.4	Abfrageparameter der Bibliothek . . . . .	27
4.1	Messungen Smartphone Standortabfrage High Accuracy HAW Hamburg Medienfassade . . . . .	38
4.2	Messungen Smartphone Standortabfrage Low Accuracy HAW-Hamburg Medienfassade . . . . .	38
4.3	Messungen Tablet ohne Mobilfunk Standortabfrage High Accuracy HAW-Hamburg Medienfassade . . . . .	39
4.4	Messungen Tablet ohne Mobilfunk Standortabfrage Low Accuracy HAW-Hamburg Medienfassade . . . . .	40

# Abbildungsverzeichnis

2.1	Interaktion Smartphone . . . . .	8
2.2	Positionsbestimmung mit GPS . . . . .	18
2.3	Timing advance mit omnidirektionalen und sectorisierten Antennen . . . . .	19
2.4	Uplink Time Difference of Arrival (U-TDoA) Configuration . . . . .	20
2.5	Hyperbolic D-TDoA . . . . .	21
2.6	A-GPS infrastructure . . . . .	22
2.7	WLAN fingerprinting environment . . . . .	23
4.1	Ausgabe Positionsdaten Web Applikation . . . . .	36
4.2	Ausgabe IP-Adresse und verarbeitete Abfragen . . . . .	37



# 1 Einleitung

## 1.1 Motivation

Location Based Services werden durch den technologischen Fortschritt immer weiter verbreitet. An diversen Orten erinnert das Smartphone an Veranstaltungen oder Sehenswürdigkeiten in der Umgebung, Medienfassaden oder Public Displays zeigen Inhalte die auf den User in ihrer Umgebung zugeschnitten sind oder diesen zur Interaktion auffordern. Möglich ist dies alles zu aller erst durch das Smartphone, welches ein Großteil der Menschen heutzutage bei sich trägt. Gepaart mit den Informationen, die das Smartphone mit seinen Apps und Diensten sammelt und bereitstellt, ist der Standort für Location Based Services der wichtigste Faktor. Über diverse technische Möglichkeiten lässt sich inzwischen der Standort immer genauer bestimmen, so dass sich auch für die Interaktion mit Dingen in der Umgebung immer neue Möglichkeiten eröffnen. Um den Usern in der Umgebung passende Informationen oder Angebote anzeigen zu können, benötigen auch die Endgeräte Informationen über Ihre Umgebung, sie müssen Kontextsensitiv sein.

Diese Allgegenwärtigkeit von Computern, die dennoch im alltäglichen Leben verschwindet und nicht hervorgehoben wird, wurde schon 1988 von Mark Weiser als Ubiquitous computing bezeichnet. (vgl. [Weiser \(1999\)](#)) Gerade diese Allgegenwärtigkeit macht eine zuverlässige und genaue Standortbestimmung immer wichtiger, da die so genannten Smart Objects zur Interaktion wissen müssen, ob und welcher User sich in ihrer Umgebung aufhält. Smart Objects sind meist über das Internet vernetzt und gehören zum Internet der Dinge. (vgl. [Uckelmann u. a. \(2011\)](#)) Smart Objects interagieren mit Objekten in ihrer Umgebung, was deren Standortbestimmung voraussetzt, um festzustellen, ob und welche Objekte sich in ihrer Umgebung befinden.

Diese Context und Location awareness ist ein komplexes und vielschichtiges Gebiet, da es viele Aspekte zu beachten gibt. Wie ist Umgebung definiert, wie sind die Voraussetzungen für eine Interaktion und welche Informationen sollen dem User zur Verfügung gestellt werden? Diese Problemstellungen finden sich in der Umgebung der Medienfassade auf dem Campus der Hochschule für Angewandte Wissenschaften Hamburg (HAW-Hamburg) wieder. Hier soll den Studenten und Besuchern kontextsensitiver Inhalt dargestellt werden, welcher in einigen

Fällen auch zur Interaktion auffordert. In dieser Umgebung soll diese Arbeit durchgeführt und die Implementierung getestet werden.

### 1.2 Zielsetzung

In dieser Arbeit soll eine Software-Bibliothek konzipiert und entwickelt werden, die Entwickler dabei unterstützt, die Position eines Endgerätes zu bestimmen, um kontextsensitiven Inhalt darzustellen und so die Interaktion mit Dingen in der Umgebung zu ermöglichen. Da die Entwicklung in Bezug auf die Medienfassade der HAW-Hamburg erfolgen soll, muss zunächst analysiert werden, wie Public Displays arbeiten und wie eine Interaktion mit ihnen ablaufen kann. Des Weiteren muss aufgezeigt werden, was Geräte basierte Interaktion bedeutet und wie diese Hardwareseitig funktioniert. Im Umfeld der HAW-Hamburg gehört auch die User-Authentifizierung zu der Interaktion dazu, um sicherstellen zu können, dass eventuelle rechtswidrige Inhalte oder Interaktionen einer realen Person zugeordnet werden können, um solche Art der Interaktion zurück verfolgbar zu machen.

Ein Großteil der Arbeit wird darin bestehen, die Möglichkeiten der Standortbestimmung zu analysieren und zu bestimmen, welche Möglichkeiten sich für die Umgebung der Medienfassade an der HAW-Hamburg am besten eignen. Außerdem muss festgelegt werden, welche Parameter eingehalten werden müssen, um eine Interaktion zuzulassen und welche Informationen dem Entwickler zur Verfügung gestellt werden sollen. Um dem Entwickler die Arbeit zu erleichtern müssen bestimmte Funktionen gegeben sein und einige Daten verarbeitet werden, bevor sie zurückgegeben werden.

Am Ende dieser Arbeit soll ein Prototyp einer Bibliothek entwickelt worden sein, der die ermittelten Anforderungen erfüllt. Zu Testzwecken soll eine einfache Web-Application implementiert werden, welche die entsprechenden Funktionen testet und die Ergebnisse darstellt.

### 1.3 Gliederung

**Kapitel 2** befasst sich mit der Analyse der in der Zielsetzung definierten Komponenten. Zunächst werden Public Displays und die Interaktion mit öffentlich Wahrnehmbarem genauer betrachtet. Im Folgenden wird genauer auf die Geräte basierte Interaktion eingegangen und analysiert, welche Möglichkeiten nach dem heutigen Stand der Technik bestehen und wie eine solche Interaktion funktioniert.

Im darauf folgenden **Kapitel 3** wird erläutert, welche Charakteristiken Bibliotheken mit sich bringen, welche Technologien genutzt werden sollen und welche Funktionen implementiert

werden sollen, um die vorausgesetzten Eigenschaften umzusetzen. Die Entscheidungen zu Architektur und Designentscheidungen werden hier ebenfalls erläutert.

Im Anschluss werden in **Kapitel 4** die Implementierung und die Ergebnisse evaluiert. Hier werden Messwerte und Messfehler aufgeführt und die Erleichterung der Programmierung für Entwickler beleuchtet.

Das abschließende **Kapitel 5** fasst die Ergebnisse dieser Arbeit noch einmal zusammen und gibt einen Einblick in weiterführende Arbeiten oder Arbeiten, die von den Ergebnissen dieser Arbeit profitieren können.

## 2 Analyse

In diesem Kapitel sollen die Anforderungen für eine Interaktion eines Users mit einem Public Display analysiert werden. Hierzu wird zunächst vorgestellt, was Public Displays sind, welche Interaktionsmöglichkeiten es gibt, wie diese grundlegend ablaufen und welche Voraussetzungen dafür geschaffen sein müssen. Im Anschluss werden verwandte Arbeiten und bereits geschaffene Voraussetzungen vorgestellt und darauf eingegangen, welchen Einfluss diese mit sich bringen. Zum Abschluss des Kapitels wird darauf eingegangen, welche Interaktionen mit der Medienfassade der HAW-Hamburg am sinnvollsten sind und wie diese realisiert werden können.

### 2.1 Public Displays

Heutzutage sind Public Displays, Mediale Fassaden, auch Medienfassaden genannt, aus dem Stadtbild nicht mehr weg zu denken. Als Public Displays werden dabei sämtliche Anzeigegeräte wie Monitore, LED-Displays oder Ähnliches bezeichnet, die in öffentlich zugänglichen Räumen aufgestellt sind. Anders als Displays in Räumen, die nur ausgewählten Personen zugänglich sind, wie z.B. Büros, hier spricht man in der Regel von Semi-Public Displays. Als "Medienfassade" werden dabei alle Fassaden bezeichnet, die mit Hilfe von computergesteuerten Leuchtmitteln oder Displays wandelbar sind. [Sauter \(2004\)](#)

#### 2.1.1 Digital Signage

Im Kontext von Public Displays spricht man häufig auch von Digital Signage. Dieser Begriff wird technisch den verteilten Systemen zugeordnet, was die Vernetzung der Public Displays beinhaltet, um aktuelle und wechselnde Informationen darzustellen. Diese werden von Personen zusammengestellt oder auch automatisch von Systemen im Hintergrund zusammengetragen und aufgearbeitet (Backend).

### 2.1.2 Einsatz Public Displays

Public Displays kommen heutzutage an diversen Stellen innerhalb unseres alltäglichen Lebens zum Einsatz. Meistens werden sie gar nicht bewusst wahrgenommen (pervasive displays). Dazu gehören z.B. Werbeanzeigen in Schaufenstern oder Kaufhäusern, Verkehrs- und Informationsanzeigen im Straßenverkehr sowie Anzeigen in Bahnen oder Bahnhöfen, die Reisende mit aktuellen Nachrichten und Informationen versorgen. Zusammengefasst bieten Public Displays eine freie Zugänglichkeit zu kontextabhängigen Informationen an (meist) öffentlichen Orten.

### 2.1.3 Bespielungszustände einer Medienfassade

Ein Großteil der Medienfassaden lässt keine Interaktion mit dem Benutzer oder der Umgebung zu und dient nur der Darstellung von Informationen, aber eine Medienfassade kann, nach Joachim Sauter, drei verschiedene Bespielungszustände einnehmen **Sauter (2004)**:

**Autoaktiv** Auf einer dynamischen Fassade werden Bewegtbilder abgespielt. Eine direkte Interaktion ist nicht möglich, nur das eingespielte Bildmaterial kann vor der Einspielung verändert werden.

**Reaktiv** Eine reaktive Medienfassade reagiert auf ihr Umfeld. Ein weit verbreitetes Anwendungsgebiet hierfür sind Informations-Fassaden und -Displays, beispielsweise Wetteranzeigen oder Abfahrtmonitore an Bus- und Bahnstationen.

**Interaktiv** Handelt es sich um eine interaktive Medienfassade, hat der Mensch die Möglichkeit mit der Fassade in einen wechselseitigen Dialog zu treten. Die Interaktion kann dabei über vorhandene Interfaces in der Umgebung der Medienfassade erfolgen oder auch über Schnittstellen zu mobilen Endgeräten wie beispielsweise Smartphones.

Zusätzlich kann die Fassade partizipativ gestaltet sein, was soviel bedeutet, dass einer interessierten Öffentlichkeit die Möglichkeit gegeben wird, die Medienfassade mit eigenen Inhalten zu bespielen. Die Beiträge der Benutzer werden hierbei in die Einspielungen eingearbeitet, dadurch erreicht man, dass sich die Benutzer nicht nur mit der Fassade beschäftigen, sondern sich auch mit ihr identifizieren. Die Fassade dient somit als Plattform für interessierte User, die durch dieses gleiche Interesse zu einer Community werden.

Die folgenden Abschnitte behandeln überwiegend den interaktiven Fassadentyp. Es werden zum einen die möglichen Eingabegeräte vorgestellt, als auch die Voraussetzungen analysiert, die für eine sichere und nachvollziehbare Interaktion notwendig sind.

## 2.2 Grundlagen der Interaktion

Im Folgenden werden die Grundlagen und der grundlegende Ablauf einer Interaktion beschrieben und anhand eines Szenarios für eine Medienfassade verdeutlicht.

### 2.2.1 Definition einer Interaktion

Im Bereich der Informatik ist eine Interaktion meistens eine Human-Computer-Interaction (HCI). Die Interaktion mit einer Medienfassade gehört ebenfalls zur HCI. Diese Art der Interaktion unterliegt etwas anderen Regeln als die alltägliche Interaktion zwischen Mensch und Computer, da der Maßstab größer und die Bedienung, wie in Abschnitt 2.4 beschrieben, nicht über die herkömmlichen Eingabegeräte wie Maus und Tastatur geschieht. Außerdem ist die Interaktion hier auch an eine räumliche Komponente gebunden, welche in dieser Arbeit behandelt werden soll.

### 2.2.2 Ablauf

In diesem Abschnitt wird ein beispielhafter Ablauf anhand eines Szenarios einer Interaktion mit einer Medienfassade dargestellt, um vereinfacht zu verdeutlichen, wie eine erfolgreiche Interaktion ablaufen kann. Eine detailliertere Darstellung folgt im nächsten Abschnitt.

**Display** Stellt Daten auf der Medienfassade dar und bietet die Möglichkeit zur Interaktion und wartet auf entsprechende Reaktion eines oder mehrerer Benutzer

**User** reagiert auf die Interaktions-Aufforderung, öffnet die entsprechende Webseite

**Mobilgerät** Authentifizierung und Anmeldung des Users zur Interaktion

**Display und Backend** bei erfolgreicher Authentifizierung Rückmeldung an das Smartphone, öffnen der entsprechenden Anzeige, Sperren der Nutzung durch andere User

**Display** Optische Meldung an den User (Anleitung etc.)

**User** Eingabe von Befehlen an Smartphone

**Mobilgerät** Senden der Eingaben an das Backend

**Backend** Verarbeiten der Eingaben

**Display** Anzeigen der Informationen oder Statusänderungen für den User

**Backend** Rückmeldung an das Mobilgerät

**User** Aufnahme der optischen Informationen und der Rückmeldungen des Smartphones

... Wiederholung der Eingabe, Verarbeitung und Ausgabe

**User** Beenden der Verbindung durch Abmeldung, schließen der Webseite, Verlassen des Interaktionsbereiches oder Timeout

**Display und Backend** Beenden der Verbindung, Freigabe der Anmeldung für den nächsten User

**Mobilgerät** Beenden der Interaktion innerhalb der Webseite

Bei diesem Ablauf interagiert der User durch sein mobiles Endgerät. Die Daten der Interaktion werden an das Backend geschickt, dort verarbeitet und auf dem Public Display dargestellt. Die Anmeldung startet die Interaktion und die Abmeldung, durch einen der Teilnehmer, beendet sie wieder. Im folgenden Abschnitt wird speziell auf die Interaktion mit Public Displays eingegangen.

## 2.3 Interaktion mit Public Displays

In diversen Szenarien ist eine Interaktion mit dem Public Display notwendig oder möglich. Dabei unterscheidet man im Allgemeinen zwischen der direkten und der indirekten Interaktion. Direkte Interaktion wird man in Bezug auf Public Displays selten antreffen, da der Nutzer hier auf eine Schnittstelle in Form eines Eingabegerätes angewiesen ist. Dies kann ein Smartphone oder auch eine Kamera sein, die Bewegungen des Nutzers aufnimmt und an das Backend weiterleitet. Somit spricht man hier von indirekter Interaktion. Betrachten wir die Interaktion mit einem Smartphone als Interface: In Abbildung 2.1 zu erkennen, verläuft die Kommunikation der einzelnen beteiligten Komponenten hier bidirektional, der Austausch von Daten und Informationen zwischen Nutzer und Smartphone sowie zwischen Display (bzw. Backend) und dem Smartphone passiert direkt, während der Nutzer nur vom Display direkte Informationen erhält, dem Display selbst aber nur über das Smartphone Daten schicken kann.

### 2.3.1 Arten der Interaktion

Es gibt diverse Möglichkeiten eine Interaktion mit Etwas durchzuführen. Dabei unterscheidet man in der Regel zwischen Körper basierter- und Geräte basierter Interaktion und einigen Möglichkeiten, die sich nicht in eins dieser beiden Schemata einordnen lassen.

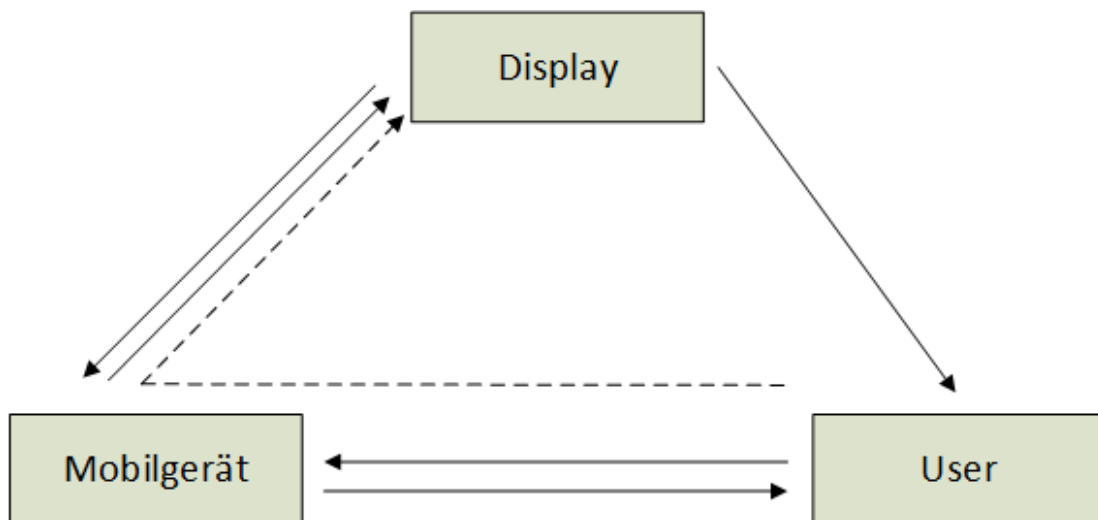


Abbildung 2.1: Interaktion Smartphone

- Körper basierte Interaktion:
  - Körperbewegungen
  - Gesten
  - Anwesenheit / Nähe
  - rufen / sprechen
- Geräte basierte Interaktion
  - Smartphone (z.B. Kamera, RFID, Bluetooth, etc.)
  - Tastatur, Maus
  - andere Geräte (z.B. RFID-tag, Kamera, Media-Player, Laptop)
- Schatten
- Multi-Touch
- Dinge werfen
- Stimmungserkennung

In dieser Arbeit wird die Geräte basierte Interaktion behandelt, da bei dieser eine Standortbestimmung durchgeführt werden kann.



### 2.3.2 Phasen einer Interaktion

Die Interaktion mit einem Public Display sollte sich in folgende Phasen aufteilen:

**Aufforderung zur Interaktion** Auf dem Display wird die Möglichkeit zur Interaktion dargestellt, damit potentielle User wissen, dass sie jetzt mit der Fassade interagieren können.

**Öffnen des User-Interfaces** Der User öffnet auf seinem mobilen Endgerät die Bedienoberfläche, meist eine Webseite oder eine App.

**Authentifizierung und Standortbestimmung** Der User meldet sich mit gegebenenfalls mit seinen Userdaten in der Bedienoberfläche an. Um die Berechtigung zur Interaktion zu vervollständigen, wird die Position des Users bestimmt um festzustellen, ob er sich in dem Bereich vor dem Display befindet, an dem eine Interaktion erlaubt ist. Im Falle der HAW-Hamburg reicht zur Authentifizierung die Abfrage der IP-Adresse.

**Sperrung und Interaktion** Die eigentliche Interaktion des Users, welche im Backend verarbeitet und auf dem Display ausgegeben wird. Damit nur eine begrenzte Anzahl an Usern gleichzeitig mit dem Display interagieren kann, muss für alle anderen User die Nutzung gesperrt werden.

**Abmeldung** Der User beendet seine Interaktion indem er sich selbst oder das Backend ihn abmeldet. Hierdurch wird die Sperrung für andere User aufgehoben.

### 2.3.3 Interaktion mit öffentlich Wahrnehmbarem

Bei der Interaktion mit öffentlich Wahrnehmbarem gibt es viele Faktoren zu beachten. Zunächst die Arten der Interaktion:

- Welche Geräte und Hilfsmittel sollen genutzt werden?
- Welche Datenschnittstellen will man nutzen?
- Kann der User nur Informationen abrufen oder auch selbst Informationen anzeigen?
- Ist eine Registrierung des Nutzers erforderlich? Wenn ja, wie?
- Wie und wie lange wird einem User die Möglichkeit gegeben mit dem Display zu interagieren?

Die Interaktion besteht in der kleinsten Zusammensetzung aus 3 Komponenten: Dem Display, dem Smartphone und dem User. Diese Zusammensetzung wächst mit jedem User und Smartphone, das gleichzeitig mit dem Display interagieren kann. Grundlegende Voraussetzung zur Interaktion ist die Verbindung zwischen Smartphone und Display. Im Normalfall übernimmt eine entsprechende App auf dem Smartphone den Aufbau und die Verwaltung der Verbindung und stellt ebenfalls Elemente zur Interaktion bereit (Buttons, Eingabefelder, etc.). Die Schnittstelle zur Verbindung (WLAN, Bluetooth, NFC, etc.) wird vom Display vorgegeben und kann gegebenenfalls vom User aus den Vorgaben ausgewählt werden. Gleichzeitig kann durch diese Vorgabe sichergestellt werden, dass der User sich in der unmittelbaren Umgebung des Displays aufhält. Zusätzlich lassen sich zur Bestimmung des Standortes die Daten des Smartphones, wie z.B. GPS, nutzen. Eine weitere Möglichkeit für die Anmeldung und die gleichzeitige Sicherstellung der Position des Users, ist das Abfotografieren eines QR-Codes innerhalb der App, welche dann anhand der kodierten Daten die Verbindung aufbauen kann. Ein Aspekt, der ebenfalls eine nicht zu vernachlässigende Rolle spielt, ist die Anzahl der gleichzeitigen User. Handelt es sich um eine einfache Interaktion, also eine Interaktion zwischen einem User und dem Display, muss allen anderen möglichen Usern der Zugriff verweigert werden. Bei der Interaktion mittels Smartphone wäre dies einfach zu realisieren, da nur das angemeldete Smartphone den Zugriff erhält. Bei einer Interaktion mittels Bewegungssensoren oder Geräuschen muss immer die relevante Information des interagierenden Users von allen unwichtigen Umgebungseinflüssen unterschieden werden. Bei einer mehrfachen Interaktion haben mehrere User zur selben Zeit die Möglichkeit mit dem Display zu interagieren. Die technische Umsetzung unterscheidet sich nicht grundlegend von der Interaktion mit einem User, allerdings müssen hier die Informationen mehrerer User verarbeitet und aufeinander abgestimmt werden. Das heißt, ein User kann unter Umständen nicht auf alle Informationen auf dem Display Einfluss nehmen. Auch hier müssen sich alle User im Aktionsradius des Displays befinden und von allen unwichtigen Einflüssen unterschieden werden. Abgesehen von den technischen Fragen müssen auch die rechtlichen Fragen geklärt sein, was das Anzeigen von Inhalten sowie den Zugriff von Nutzern angeht.

### 2.3.4 Bedingungen für eine Interaktion

Folgende Bedingungen müssen für eine Interaktion vollständig oder teilweise erfüllt sein:

- Geografisch im Interaktionsbereich des Displays
- Smartphone oder mobiles Endgerät
- Anmeldung (A-Kennung) im WLAN der HAW-Hamburg

- Entsprechende App
- Ggfs. Bluetooth oder NFC

Der Benutzer muss sich geografisch im Interaktionsbereich der Medienfassade befinden, um mit ihr interagieren zu können. Möglichkeiten hierzu werden in den folgenden Abschnitten behandelt. Die Standortbestimmung muss, je nach Anforderung der Anwendung, auch während der Interaktion weiterhin durchgeführt werden, um sicherzustellen, dass der User sich nicht aus dem Interaktionsbereich heraus bewegt. Hierzu kann der Entwickler die Abfrage nach der Position erneut ausführen, wenn sie benötigt wird. Eine weitere Möglichkeit ist die Implementierung einer kontinuierlichen Positionsbestimmung und Rückgabe an die Anwendung. Da es sich um gerätebasierte Interaktion handelt, ist ein Gerät, z.B. Smartphone oder Tablet, nötig, um eine Interaktion durchzuführen. Die Anmeldung im WLAN der HAW-Hamburg ist ebenfalls zwingend nötig, da so die Identität des Users eindeutig festgestellt werden kann, um einen Missbrauch zu unterbinden. Eine entsprechende App und Bluetooth- bzw. NFC-Verbindungen sind hier optionale Bedingungen und hängen von der Umsetzung der Programme ab, die auf der Medienfassade dargestellt werden. In dieser Arbeit werden die Interaktionen mit Web-basierten Darstellungen behandelt, somit entfallen diese beiden Bedingungen.

## 2.4 Geräte basierte Interaktion

Eine gerätebasierte Interaktion setzt ein oder mehrere Eingabegeräte voraus, die der User nutzt, um mit der Medienfassade zu interagieren

### 2.4.1 Mögliche Arten von Geräten zur Interaktion

Die gerätebasierte Interaktion lässt sich über verschiedene Geräte realisieren. Die am wenigsten aufwendige Variante ist es, Geräte zu benutzen, die der User mitbringt, sprich Handy oder Smartphone, eventuell ein Tablet, also generell mobile Endgeräte.

### 2.4.2 Nutzbare Technologien mobiler Endgeräte

- GPS
- WLAN
- GSM
- Bluetooth

- NFC

### 2.4.3 Was passiert im Frontend?

Als Frontend fungieren hier zwei Geräte: Zum einen die Medienfassade zur Darstellung von Inhalten, zum anderen das Smartphone sowohl zur Darstellung von Inhalten als auch zur Interaktion mit der Medienfassade. Die Medienfassade dient hierbei nur zur reinen Darstellung der Inhalte, die das Backend liefert, die Interaktion findet über das Smartphone des Users (oder andere Eingabegeräte wie Kameras etc.) statt. Das Smartphone als Frontend kann mehrere Funktionen übernehmen. Die Darstellung von Inhalten, evtl. Zusatzinformationen zu den Infos auf der Fassade oder auch Infos "zum Mitnehmen", die den User über weitere Details informieren, während er den Interaktionsraum des Displays schon verlässt, ist sehr gut auf einem Smartphone realisierbar. Des Weiteren dient das Smartphone als Interaktions-Frontend für den User. Hier lassen sich Eingabefelder oder Buttons darstellen, mit deren Hilfe der User mit der Medienfassade interagieren kann.

### 2.4.4 Was passiert im Backend?

Das Backend ist in diesem Fall der Rechner, der die Medienfassade steuert sowie die entsprechenden Anwendungen im Hintergrund, somit auch die Anwendungen, mit denen der User interagieren kann. Das Backend umfasst dabei alle wichtigen Funktionen, die zur Darstellung auf und auch zur Interaktion mit der Medienfassade zuständig sind.

Hierzu zählen u.a.

- die Darstellung der entsprechenden Anwendung
- die User-Authentifizierung
- die Autorisierung der interagierenden User
- die Verwaltung der Verbindungen
- Verarbeitung der Interaktionen der User

Das Backend ist für die Verarbeitung und Aufbereitung aller Daten zuständig, die auf der Medienfassade angezeigt werden sollen. Dabei geht es zum einen um die korrekte Darstellung der Inhalte, zum anderen aber auch um die Überprüfung und Protokollierung der Inhalte, die von Usern dargestellt werden. Um gewährleisten zu können, dass die eingespielten Inhalte auch realen Personen zugeordnet werden können, muss eine Verknüpfung zu einem Benutzerkonto

hergestellt werden, darum müssen User sich im WLAN der HAW-Hamburg anmelden. Diese Anmeldung läuft über die Kennung, die von der Hochschule bei der Zulassung vergeben wird und somit ist sie eindeutig einer Person zuzuordnen.

### **2.5 HAW-Hamburg - Medienfassade**

Seit Mitte 2015 steht auf dem Campus der HAW-Hamburg am Berliner Tor, direkt vor dem Living Place Hamburg, eine große Medienfassade. Der Innenhof der HAW-Hamburg wurde im Zuge des Urban Interactions Projektes neu geplant, was auch den Aufbau der Medienfassade beinhaltet. Dieses Public Display wird einerseits zur Anzeige von Informationen genutzt, zum anderen werden aber auch Medien dargestellt werden, die den Menschen auf dem Campus die Möglichkeit bieten, mit der Medienfassade zu interagieren. Hier ist es in den meisten Fällen notwendig, dass der User sich in einem bestimmten Bereich vor der Medienfassade befindet, darum wird die Implementierung in dieser Umgebung getestet.

#### **2.5.1 Mögliche Interaktionen**

Die Möglichkeiten zur Interaktion mit der Medienfassade der HAW-Hamburg auf dem Campus Berliner Tor beschränken sich zur Zeit auf Geräte basierte und Körper basierte Interaktionen. Die Geräte basierte Interaktion wird in dieser Arbeit behandelt, die Körper basierte Interaktion basierend auf der Erkennung von Gesten und Körperbewegung mittels Kameras oder Ähnlichem soll nicht Teil dieser Arbeit sein.

#### **2.5.2 Use-Cases auf dem HAW-Hamburg Campus Berliner Tor**

Es gibt diverse Möglichkeiten für die Verwendung der Medienfassade. Die folgenden Beispiele sollen die verschiedenen Arten aufzeigen und darstellen für welche Art der Nutzung welche Art der Interaktion in Frage kommt und welche Voraussetzungen, beispielsweise in Bezug auf die Position des Users, erfüllt sein müssen.

##### **2.5.2.1 Posts**

Eine Möglichkeit der Interaktion ist beispielsweise die Darstellung von Tweets über den Microblogging Dienst Twitter<sup>1</sup>. Durch die Kennzeichnung eines Tweets mit einem entsprechenden Hashtag wird der Tweet auf der Medienfassade angezeigt. Dies wäre eine indirekte Interaktion,

---

<sup>1</sup><https://twitter.com> Stand 31.10.2015

da der User nicht direkt sondern nur über das absetzen eines Tweets mit der Medienfassade interagieren kann. Folglich muss sich der User für diese Interaktion auch nicht im Interaktionsbereich der Medienfassade befinden. Das Absetzen des Tweets kann sowohl von einem Smartphone als auch von jedem beliebigen PC erfolgen. Eine Voraussetzung ist jedoch, dass der User eindeutig identifizierbar und der Tweet so zurückzuverfolgen ist. Diese Voraussetzung muss erfüllt sein, um die Darstellung illegaler Inhalte zu unterbinden.

### 2.5.2.2 Spiele

Spiele an denen User direkt teilnehmen können erfordern eine direkte Interaktion mit der Medienfassade. Somit ist es notwendig sicherzustellen, dass der User sich im Interaktionsbereich der Medienfassade befindet. Außerdem müssen bestimmte Bedingungen erfüllt sein, damit die Interaktion durchgeführt werden kann. Diese werden in 2.3.4 behandelt.

### 2.5.2.3 Informationen für Studierende

Diese Art der Darstellung erfordert grundlegend keine Interaktion mit einem User, da Informationen nur angezeigt werden. Allerdings ist es möglich und auch sinnvoll personalisierte Inhalte anzuzeigen. Hierzu ist es erforderlich einen User zu identifizieren, der sich gerade im Interaktionsbereich aufhält oder sich auf die Medienfassade zu bewegt. Eine denkbare Anwendung für eine Interaktion in diesem Bereich wäre die Anzeige von bestimmten oder eigenen Inhalten. Beispielsweise kann ein User eine bestimmte Information aufrufen oder auch eine eigene Information oder News anzeigen lassen, wie z.B. Anzeigen für Angebote oder Gesuche oder Flyer für Veranstaltungen.

## 2.6 User-Authentifikation

“Hat ein Subjekt einen Zugriff bzw. eine Aktion durchgeführt, so ist es vielfach notwendig, dass auch noch im Nachhinein die Urheberschaft des Zugriffs bzw. der Aktion eindeutig dem entsprechenden Subjekt zuordenbar ist. Man spricht hier von der Verbindlichkeit oder Zuordenbarkeit des Systems. [...] Die Authentizität eines Subjekts bzw. Objekts wird durch Maßnahmen zur Authentifikation (engl. authentication) überprüft. Dazu muss nachgewiesen werden, dass eine behauptete Identität eines Objekts oder Subjekts mit dessen charakterisierenden Eigenschaften übereinstimmt.“ [Eckert \(2013\)](#)

### **2.6.1 Grundlagen**

Die eindeutige Zuordnung von Aktionen ist gerade in öffentlichen Bereichen von großer Wichtigkeit, da es dem User ermöglicht wird, Inhalte auf der Medienfassade darzustellen. Die dargestellten Inhalte können in vielen Fällen nicht vor der Darstellung kontrolliert werden und somit muss es möglich sein, zurück zu verfolgen, wer wann welchen Inhalt veröffentlicht hat.

### **2.6.2 Authentifizierung mit Hochschulaccounts**

Der Zugriff auf die Medienfassade ist nur aus dem WLAN-Netz der HAW-Hamburg gestattet. Um sich mit diesem Netz zu verbinden, muss der User sich mit seiner, von der Hochschule bei der Immatrikulation zugewiesenen, Kennung und seinem persönlichen Passwort anmelden. Durch diese Anmeldung bestätigt der User seine Identität und ist somit anhand der ihm zugewiesenen IP-Adresse und seinem Benutzer-Account zurück verfolgbar.

### **2.6.3 Was bedeutet Zurückverfolgbarkeit**

Zurückverfolgbarkeit bedeutet in diesem Fall, es lässt sich von der IP-Adresse auf den Benutzer-Account schließen. An den Benutzer-Account wiederum ist eine reale Person gebunden. Bei der Immatrikulation an einer Hochschule müssen Name und Adresse angegeben werden, daraufhin wird dem User ein Account erstellt, den er nicht selbst anlegen oder ändern kann. Somit sind Account und User eindeutig miteinander verbunden.

## **2.7 Standortbestimmung**

Eine Interaktion mit einer Medienfassade gehört zu den Location-Based Services. Der User muss sich für eine Interaktion in einem bestimmten Bereich vor der Medienfassade aufhalten, um eine weitere Bedingung zu erfüllen. Hierzu muss seine Position bestimmt werden. Bei einer Geräte basierten Interaktion mit Hilfe eines Smartphones stehen hierzu verschiedene Möglichkeiten zur Verfügung. Der folgende Abschnitt soll einen Überblick über diese geben.

### **2.7.1 Grundlagen der Standortbestimmung**

Für eine Standortbestimmung muss man zunächst zwischen physischen und virtuellen Standorten unterscheiden. Physische Standorte beschreiben einen Ort in der realen Welt, virtuelle Standorte dagegen beschreiben einen Ort in einer virtuellen Welt, beispielsweise im Internet. Da in dieser Arbeit nur physische Standorte benötigt werden, werden die virtuellen Standorte

im Folgenden nicht weiter behandelt.

Die physischen Standorte lassen sich in 3 Kategorien aufteilen ((vgl. [Kupper, 2005](#), S.18ff)):

**Beschreibende Standorte** Beschreibende Standorte sind immer an geographische Orte oder Objekte gebunden, wie z.B. Gebiete, Berge, Seen oder auch von Menschen geschaffene Objekte wie Gebäude, Straßen, Städte oder Länder. Diese Standorte werden durch Namen, Faktoren oder Nummern beschrieben. Diese Art von Standortbeschreibung hilft Menschen im täglichen Leben sich zu orientieren, ist aber keine genaue Positionsbestimmung, mit welcher der Standort eines Users im Falle einer Interaktion bestimmt werden kann.

**Netzwerk Standorte** Netzwerk Standorte beziehen zur Bestimmung einer Position Kommunikationsnetzwerke, wie das WLAN, Internet oder Mobilfunknetze, ein. Diese Netzwerke sind durch ihre verschiedenen Zugangspunkte in kleinere Subnetzwerke aufgeteilt. Ist ein User mit einem dieser Netzwerke verbunden, lässt sich seine Position anhand diverser Faktoren bestimmen. Hierzu zählen unter anderem die IP-Adresse des Internetproviders, Zellen-ID des Mobilfunknetzes oder der WLAN-Fingerprint, dieser setzt sich aus den empfangenen WLAN-Netzen, deren SSID und der Empfangsstärke zusammen.

**Räumliche Standorte** Räumliche Standorte beschreiben einen Punkt in einem Koordinatensystem, meistens wird für diese Art Standort auch der Begriff "Position" verwendet. Die Darstellung erfolgt zwei- oder dreidimensional. Einen Standort mit Koordinaten zu beschreiben ist für den Umgang im täglichen Leben nicht sehr praktikabel, aber zur genauen Bestimmung einer Position ist es unerlässlich. Die Position kann in verschiedenen Koordinatensystemen erfolgen, dem Kartesischen Koordinatensystem oder dem Ellipsoidischen Koordinatensystem.

**Kartesischen Koordinatensystem** Im kartesischen Koordinatensystem werden die X-, Y- und Z-Koordinaten in Metern angegeben, der Nullpunkt ist hier der Erdmittelpunkt. Die Z-Achse stellt dabei die Rotationsachse der Erde dar, X- und Y-Achse sind erdfixiert und rotieren mit der Erde mit.

**Ellipsoidischen Koordinatensystem** Das ellipsoidische Koordinatensystem ist besser zur Positionsangabe auf der Erde geeignet, da die Erde ein Ellipsoid ist. In diesem Koordinatensystem werden Längen- und Breitengrade sowie die Höhe zur Angabe der Position genutzt.

Für die Bestimmung des Standortes bei einer gerätebasierten Interaktion lassen sich diverse Technologien nutzen. Im Falle eines Smartphones gibt es hier Technologien, die über unterschiedlich große Distanzen funktionieren. Die einzelnen technischen Funktionsweisen sollen



nicht Teil dieser Arbeit sein und werden nur soweit behandelt, wie sie für diese Arbeit relevant sind.

### 2.7.2 GPS

Das Global Positioning System (GPS) ist seit Jahren ein Standard in der satellitengestützten Positionsbestimmung. Das GPS bietet für den zivilen Bereich den Standard Positioning Service (SPS), welcher für eine Genauigkeit von 100m horizontal und 156m vertikal entwickelt wurde. Das Precise Positioning System ist für eine höhere Genauigkeit entwickelt worden, ist allerdings nur für militärische Zwecke nutzbar.

Durch die Abschaltung einer künstlichen Abweichung durch das Department of Defense der USA im Jahre 2000 wurde eine höhere Genauigkeit im zivilen Bereich erzielt. Im GPS SPS Performance Standard (GPS 2001) sind die Werte wie in [Tabelle 2.1](#) dargestellt festgehalten.

Accuracy standard	95% Vertical error	95% Horizontal error
Global average	$\leq 15m$	$\leq 9m$
Worst site	$\leq 37m$	$\leq 17m$

Tabelle 2.1: SPS Position/Time Service Accuracy Standards (vgl. [DoD, 2008](#), S.34)

Die Bestimmung der Position läuft in drei Schritten ab (vgl. [Kupper, 2005](#), S.171ff)

**Identifikation der Satelliten** Der GPS Empfänger, in diesem Fall ein Smartphone, identifiziert die Satelliten, von denen es ein Signal empfängt. Meistens sind 5-10 Satelliten sichtbar, allerdings hängt diese Anzahl auch von der Umgebung oder dem Wetter ab.

**Entfernungsmessung** Der zweite Schritt ist dann die Entfernungsmessung von den einzelnen Satelliten zum Empfänger.

**Positionsberechnung** Die Positionsberechnung erfolgt im dritten Schritt anhand der ermittelten Entfernungen zu den einzelnen Satelliten. Anhand dieses Radius wird der Schnittpunkt der Umkreise errechnet. An deren Schnittpunkt befindet sich dann der Empfänger. Siehe hierzu [Abbildung 2.2](#)

Die Positionsbestimmung per GPS-Signal ist stark davon abhängig, wie viele Satelliten frei empfangen werden können. In Urbanen Gebieten kann die Genauigkeit stark sinken, wenn z.B. zwischen Hochhäusern der Empfang der Satellitensignale gestört ist.

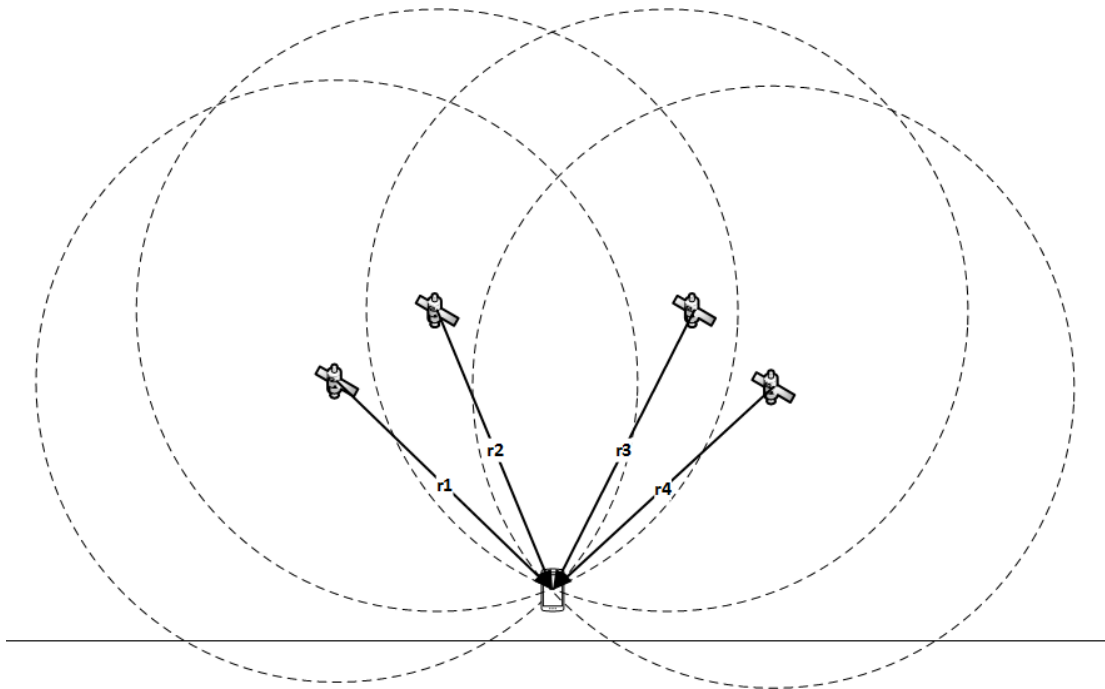


Abbildung 2.2: Positionsbestimmung mit GPS (vgl. Kupper, 2005, S.172)

### 2.7.3 Mobilfunk

Zur Standortbestimmung über das Mobilfunknetz wurden verschiedene Methoden entwickelt. Diese unterscheiden sich in ihrer Umsetzung durch Genauigkeit und Aufwand (vgl. Kupper, 2005, S.186).

Da die technischen Details dieser Methoden sehr umfangreich sind, soll in dieser Arbeit nur ein kurzer Überblick gegeben werden, durch welche Verfahren und mit welchen technischen Möglichkeiten welche Genauigkeit der Standortermittlung erreicht werden kann.

**Cell-Id in combination with timing advance** Dieses Verfahren setzt sich aus der Cell ID und Timing Advance zusammen. Die Cell ID gibt dabei die ID der Funkzelle an, in der sich das zu ortende Gerät befindet. Die Genauigkeit ist hier, je nach Dichte der Funkzellen, unterschiedlich, aber immer sehr ungenau. Allerdings kommt dieses Verfahren ohne Änderungen auf Netz- oder Endgeräteseite aus. Nimmt man noch Timing Advance hinzu, also die Laufzeit des Signals zwischen der Basisstation und dem Endgerät, kommt man auf eine genauere Ortung. Hierfür ist eine entsprechende netzseitige Ausstattung notwendig. Wie in [Abbildung 2.3](#) zu erkennen ist, kommt es bei der Genauigkeit auch darauf an,

ob es sich um omnidirektionale oder um sektorisierte Antennen handelt. Strahlt die Antenne ihre Signale in alle Richtungen, kann nur ein Umkreis um die Antenne als Standort angegeben werden. Strahlt die Antenne nur in eine bestimmte Richtung, kann man den Standort auf den Kreisausschnitt in dieser Richtung eingrenzen.

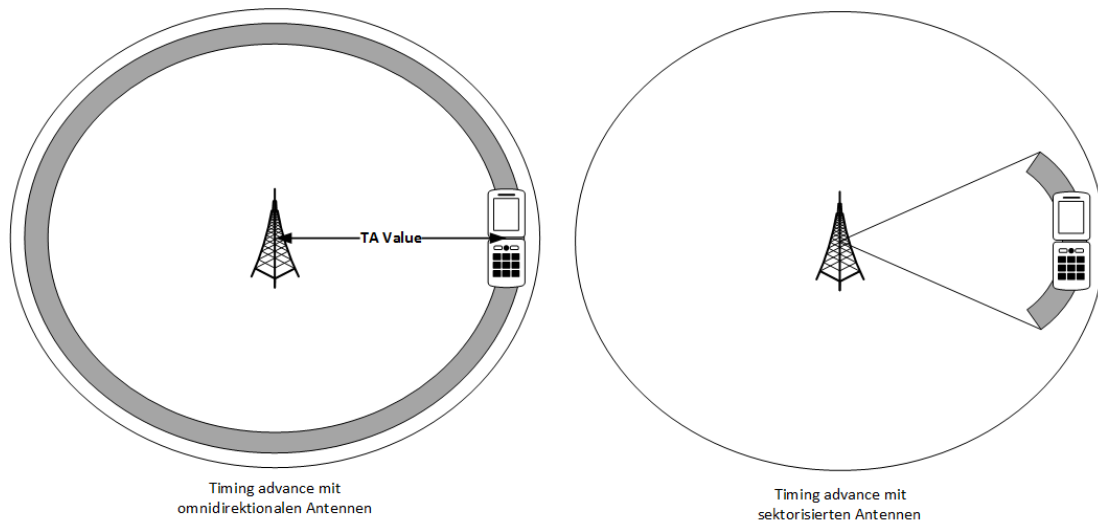


Abbildung 2.3: Timing advance mit omnidirektionalen und sektorisierten Antennen (vgl. [Kupper, 2005, S.193](#))

**Uplink Time Difference of Arrival (U-TDoA)** Für dieses Verfahren werden netzseitig so genannte Location Measurement Units (LMU) benötigt. Anhand der Laufzeiten der Signale vom Endgerät zu den Messstationen kann dann durch Lateration eine Position errechnet werden. Für eine Bestimmung der Position sind mindestens drei Messstationen nötig, man spricht hier auch von Trilateration. Wie eine solche Umgebung aussehen kann, zeigt [Abbildung 2.4](#)

**Enhanced Observed Time Difference (E-OTD)** Dieses Verfahren benötigt sowohl netzseitig als auch endgeräteseitig Ausstattung, die dieses Verfahren unterstützt. Die Basisstationen schicken dann eine Burst-Nachricht an das Endgerät, welches von jeder Nachricht die genaue Empfangszeit aufnimmt. Anhand der unterschiedlichen Laufzeiten kann dann die Position durch Lateration ermittelt werden. Eine LMU nimmt auf dem gleichen Weg Referenzmessungen vor, um zeitliche Verschiebungen aufzuzeichnen, mit Hilfe derer die

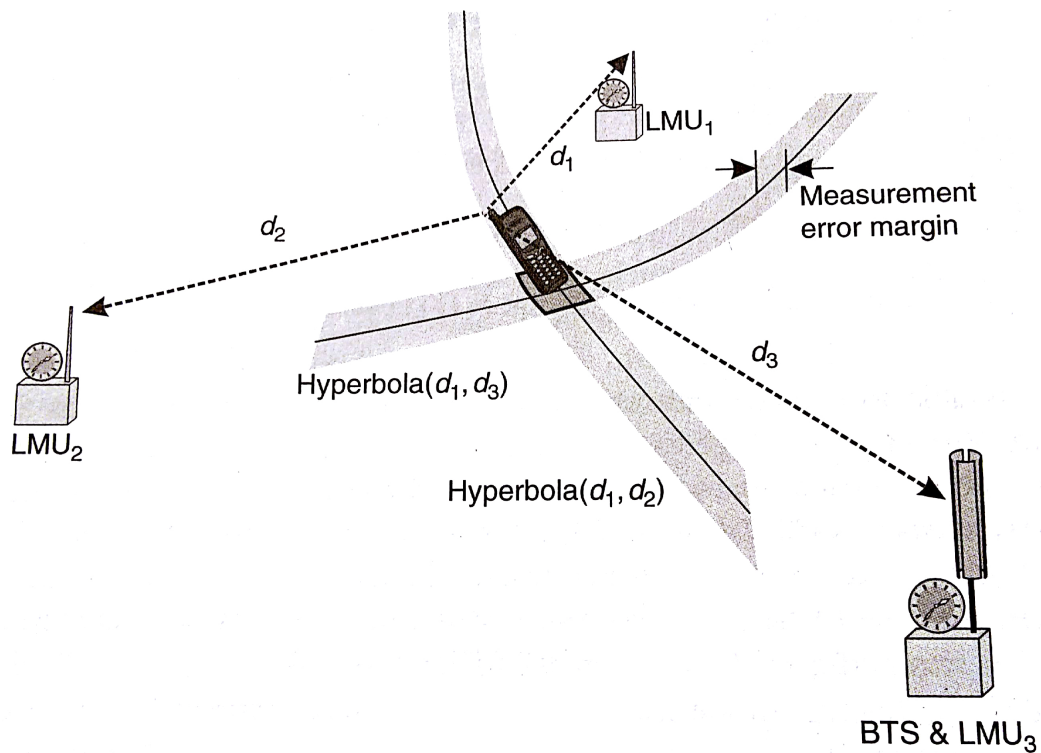


Abbildung 2.4: Uplink Time Difference of Arrival (U-TDoA) Configuration (Kupper, 2005, S.208)

Messungen synchronisiert werden können. Wie die hyperbolische Lateration abläuft zeigt [Abbildung 2.5](#)

**Assisted GPS (A-GPS)** Das Assisted GPS Verfahren bedient sich, wie der Name schon sagt, der Unterstützung von GPS. Hierbei wird das Endgerät über die Basisstationen mit zusätzlichen Daten versorgt, die es ermöglichen, schneller eine Verbindung zu GPS-Satelliten aufzubauen und die Position mit Hilfe einer Referenz-Station im Mobilfunknetz, welche Korrektur-Daten übermittelt, genauer zu berechnen. [Abbildung 2.6](#) zeigt die Signale, die bei A-GPS zum Einsatz kommen.

Es gibt diverse Wege der Positionsbestimmung über das Mobilfunksignal, welche aber entsprechende Ausstattung und Kompatibilität voraussetzen.

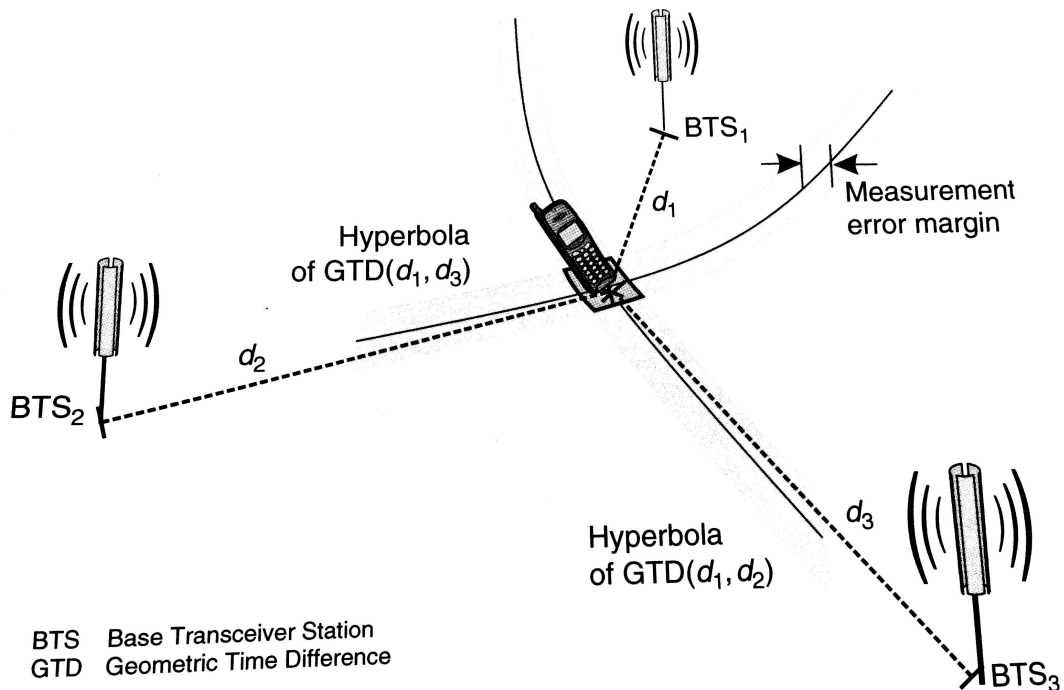


Abbildung 2.5: Hyperbolic D-TDoA (Kupper, 2005, S.195)

#### 2.7.4 WLAN

Die Ortung per WLAN-Signal eignet sich überwiegend für Innenräume, kann aber, soweit das Signal reicht, auch im Außenbereich verwendet werden. Voraussetzung für die Standortbestimmung ist, neben den Signalen von mehreren WLAN-Netzen, eine Datenbank, in welcher die Standorte der empfangenen WLAN-Netze verzeichnet ist. Inzwischen gibt es diverse Betreiber solcher Datenbanken. Will man aber innerhalb eines Gebäudes oder in einer Umgebung mit wenigen bis keinen WLAN-Netzen eine Ortung durchführen, muss man selbst zunächst eine Datenbank mit Referenzwerten erstellen. Hierzu muss die Umgebung in der die Ortung stattfinden soll, abgelaufen werden, um entsprechende Daten zu sammeln. Auf diese Art und Weise werden beispielsweise auf großen Messen Location Based Services angeboten, wie die Navigation zu einem bestimmten Stand oder zu den Ausgängen. Gleichzeitig lassen sich so Abläufe oder Unregelmäßigkeiten analysieren.

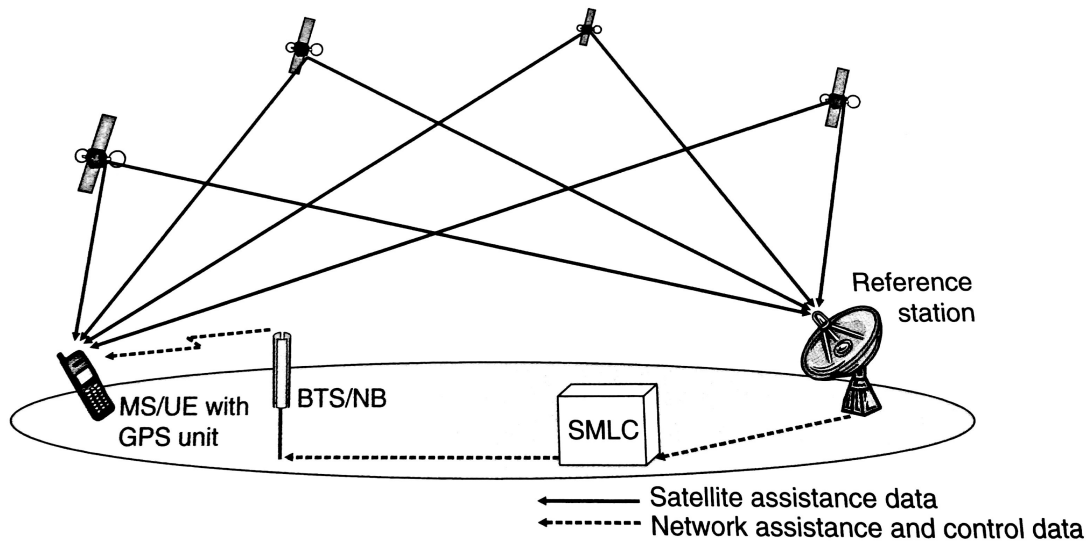


Abbildung 2.6: A-GPS infrastructure (Kupper, 2005, S.226)

Da die Ortung per WLAN nicht von einwandfreiem Satellitenempfang abhängig ist, ist sie in urbanen Umgebungen oft präziser einsetzbar als GPS, welchem grade die Häuserschluchten Probleme bereiten. Oftmals haben öffentliche- und Firmengebäude weitreichende WLAN-Netze, welche zur Ortung genutzt werden können. Zudem bieten auch Internet-Provider selbst vermehrt öffentliche WLAN-Hotspots an.

Die Ortung selber kann auch in diesem Bereich über verschiedene Verfahren funktionieren: (vgl. Kupper, 2005, S.235)

**Proximity sensing** Die Position wird anhand des Access Point bestimmt, welcher mit der besten Signalqualität empfangen wird. Als Standort des Gerätes wird dann der Standort dieses Access Point angenommen. Durch diese Annahme ist Proximity sensing die einfachste, aber auch ungenaueste Methode.

**Lateration** Lateration arbeitet ähnlich wie die Positionsbestimmung bei GPS. Die Distanz zu verschiedenen Access Points wird anhand des Verlustes eines Signals berechnet. Mit dieser Entfernung als Radius werden Umkreise berechnet, in denen sich das zu ortende Gerät befindet. Durch die Berechnung der Schnittpunkte dieser Umkreise erhält man die Position des Gerätes. Um diese Methode zuverlässig ausführen zu können, müssen die Positionen aller Access Points genau aufgezeichnet und eingehalten werden.

**Fingerprinting** Das Fingerprinting arbeitet mit der "received signal strength"(RSS). Um anhand dieses Signals eine Position zu bestimmen, ist die schon angesprochene Vorbereitung in Form einer Datenbank zu treffen. In dieser Datenbank werden beim abschreiten und sammeln der Daten die empfangenen WLAN-Netzze und ihre RSS, also der WLAN-Fingerprint, an einer Position verzeichnet. Zur Bestimmung des Standortes wird der WLAN-Fingerprint des Standortes mit denen in der Datenbank verglichen und der am besten passende wird als Standort des Gerätes angenommen.

Wurde für das Fingerprinting eine sehr detaillierte Karte erstellt, ist diese Methode die Genaueste. Allerdings ist hierfür eine aufwendige Vorarbeit notwendig, um eine Karte des Gebäudes oder der Umgebung einzumessen. **Abbildung 2.7** zeigt beispielhaft eine für WLAN-Fingerprinting eingemessene Umgebung. **Kupper (2005)**

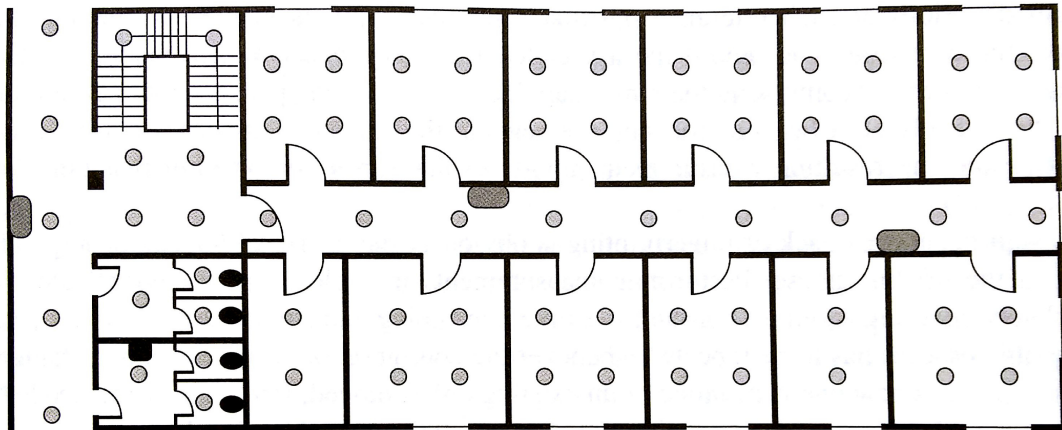


Abbildung 2.7: WLAN fingerprinting environment (**Kupper, 2005**, S.237)

### 2.7.5 Weitere Möglichkeiten

Weitere Möglichkeiten, um sicherzustellen, dass sich ein User im Interaktionsbereich einer Medienfassade befindet, sind unter Anderem der Industrie-Standard Near-Field-Communication (NFC), oder sogenannte iBeacons, die auf einer Bluetooth-Low-Energy Technologie basieren. NFC lässt sich hier weitestgehend ausschließen, da es nur eine Reichweite von ca. 10cm hat. Radio Frequency Identification (RFID) bringt zwar direkt die Möglichkeit mit sich, den User auch zu identifizieren, allerdings lässt sich, auf Grund der geringen Reichweite, nicht feststellen, wann der User den Interaktionsbereich vor der Medienfassade verlässt. (vgl. **Want, 2011**)

Bluetooth-Low-Energy Beacons gehört zu den Technologien der Wireless Personal Area Networks und hat eine Reichweite von ca. 10m. Somit würde es sich ebenfalls anbieten, um sicherzustellen, dass ein User sich im Interaktionsbereich aufhält. (vgl. Corbellini u. a., 2015) Diese Technologie soll aber nicht Teil dieser, kann aber in einer nachfolgenden Arbeit behandelt werden.

### 2.7.6 Verfügbarkeit und Qualität

Die Verfügbarkeit und Qualität der Standortdienste ist überwiegend von der Umgebung abhängig. Tabelle 2.2 zeigt die Bewertungen der einzelnen Standortbestimmungen in ruralen, urbanen und Indoor-Umgebungen.

Qualität	Rural (ländlich)	Urban (städtisch)	Indoor
GPS	sehr gut	mittelmäßig bis gut	schlecht bis gar nicht
WLAN	nicht möglich	gut bis sehr gut	sehr gut
Mobilfunk	sehr schlecht	gut	gut

Tabelle 2.2: Verfügbarkeit und Qualität in Abhängigkeit von der Umgebung (vgl. Epperlein, 2010, S.28)

Je nach Umgebung ist es somit sinnvoll, die einzelnen Methoden zu kombinieren, um ein brauchbares Ergebnis zu erzielen. Da Medienfassaden zumeist in urbanen Umgebungen eingesetzt werden, lassen sich alle angesprochenen Techniken nutzen. Die Bewertungen der einzelnen Umgebungen und Kombinationen folgt im nächsten Abschnitt.

### 2.7.7 Bewertung und Kombination

Jede der angesprochenen Ortungstechnologien hat bestimmte Voraussetzungen, Stärken und Schwächen. Im folgenden Abschnitt soll erläutert werden, in welchen Umgebungen sich welche Verfahren am besten eignen.

#### 2.7.7.1 Indoor

Die Ortung innerhalb eines Gebäudes wird durch Wände, Decken oder auch Möbel und andere elektrische Geräte erschwert. GPS ist überwiegend nicht nutzbar, da das Endgerät keine Verbindung zu den Satelliten aufbauen kann. Mobilfunkortung ist möglich, allerdings reicht die Genauigkeit nicht aus, um eine verlässliche Ortung durchzuführen. Die geeignetste Methode zur Ortung innerhalb eines Gebäudes ist die Ortung per WLAN. Wie in Abschnitt 2.7.4 erläutert



und in [Abbildung 2.7](#) dargestellt, ist dafür eine vorherige Einmessung notwendig, um eine hohe Genauigkeit zu erreichen. Eine weitere Möglichkeit wäre hier wiederum der Einsatz von Bluetooth Low Energy Modulen, auf die später im Ausblick noch einmal eingegangen wird.

### 2.7.7.2 Urban

In städtischen Gebieten mit meist vielen hohen Gebäuden ist die GPS Ortung möglich, meist aber durch unzureichende Sichtverbindung zu den Satelliten schwierig oder zu ungenau.

Die Ortung durch Mobilfunk ist in diesen Gebieten aufgrund von sehr vielen Mobilfunkantennen und Mobilfunkzellen sehr gut durchführbar und ermöglicht auch eine sehr hohe Zuverlässigkeit und Genauigkeit.

Auch die WLAN-Ortung lässt sich in urbanen Gebieten sehr gut durchführen, da es sehr viele Access Points gibt, die sich oft überlappen. Firmen wie Google oder Skyhook Wireless<sup>2</sup> haben in den letzten Jahren umfangreiche Datenbanken aufgebaut, mit deren Hilfe eine sehr genaue Position errechnet werden kann.

### 2.7.7.3 Rural

In ländlichen Gebieten ist die Ortung durch WLAN so gut wie unmöglich, da es nur vereinzelte Access Points gibt, die meistens nicht in den Datenbanken auftauchen.

Ähnlich verhält es sich bei der Ortung über das Mobilfunknetz. Die Mobilfunkantennen sind weitläufiger aufgestellt und die Mobilfunkzellen entsprechend groß, so dass eine Ortung sehr ungenau wird, wie in [2.7.3](#) beschrieben.

Die ländlichen Gebiete eignen sich hingegen hervorragend für die Ortung per GPS. Es gibt selten hohe Gebäude, die eine Verbindung zu den Satelliten behindern, so dass eine hohe Genauigkeit erreicht werden kann.

## 2.8 Verwandte Arbeiten

Der folgende Abschnitt befasst sich mit einigen Arbeiten, die Grundlagen und Voraussetzungen für die Interaktion mit Public Displays gelegt haben oder auf denen man aufbauen kann.

### 2.8.1 Diplomarbeit - Christian Epperlein

Qualitätsbewusste Bereitstellung von Standortinformationen durch flexible und intelligente Kombination verschiedener Ortungstechnologien - Technische Universität Dresden

---

<sup>2</sup>[Skyhook Wireless \(2015\)](#) - Stand: 31.10.2015

Die Diplomarbeit von Christian Epperlein beschäftigt sich mit der Entwicklung einer Middleware, welche Standortdaten für Location Based Services bereitstellen soll. In seiner Arbeit geht Epperlein auf die gängigen Lokalisierungsverfahren ein, die auch für diese Arbeit essentiell sind. Des Weiteren geht er auf Standortinformationen und Mobile Endgeräte ein.

### **2.8.2 Masterarbeit - Sven Boris Bornemann**

Entwicklung eines kontextsensitiven Berechtigungssystems für Smart Homes - HAW-Hamburg  
In dieser Masterarbeit geht es um ein Berechtigungssystem in einer kontextsensitiven Smart-Environment Umgebung. Dieses System ermöglicht die Vergabe von Berechtigungen an Benutzer anhand von statischen Regeln oder kontextabhängig.

### **2.8.3 Großer Beleg - Florian Schneider**

Mobile Interaktion mit Public Displays im öffentlichen Verkehr - Technische Universität Dresden

In dieser Arbeit geht der Autor Florian Schneider auf die Interaktion mit Public Displays im öffentlichen Verkehr ein. Auch hier geht es überwiegend um die gerätebasierte Interaktion.

### **2.8.4 Forschungsprojekt Medienfassade - Stephan Bergemann , Robin Schlegel**

Medienfassade - Möglichkeiten von Visualisierung und Interaktion

Dieses Forschungsprojekt der Hochschule für Technik und Wirtschaft Berlin von 2012 beschäftigt sich mit Public Displays und deren Integration in die alltägliche Umgebung. Die Autoren gehen dabei auf die verschiedenen Technologien der Medienfassaden, der Möglichkeit von Visualisierung auf einer Fassade und der Interaktion mit Medienfassaden ein.

## **2.9 Zusammenfassung**

Anhand der in 2.5 beschriebenen Use-Cases werden nun in diesem Abschnitt die Ergebnisse und ermittelten Anforderungen der Analyse erarbeitet werden.

Hierzu werden benötigte Komponenten und Anforderungen ermittelt, die eine erfolgreiche und ausreichende Standortbestimmung gewährleisten, um eine Interaktion mit einem Public Display zu ermöglichen.

### 2.9.1 Anforderungen

Die Bibliothek benötigt Methoden, die entsprechende Anfragen von Nutzern verarbeiten können und einheitliche Werte zurückliefern. Dabei gibt es einige Funktionen die unterstützt werden müssen und einige, welche die Usability verbessern, wenn sie implementiert sind.

Anfrageart	Beschreibung	Kriterium
einmalig	Einmalige Standortabfrage	Standard
periodisch	Periodische Standortabfrage in einem festgelegten Intervall	Verbesserung
Bereichsbedingt	Benachrichtigung bei Betreten oder Verlassen eines Bereiches	Verbesserung

Tabelle 2.3: Abfragefunktionen der Bibliothek

Anforderung	Beschreibung	Kriterium
Standortformat	Festlegung der Standortformate, Koordinaten, Symbolisch Orte, Bewegungsdaten	Standard
Datenqualität	Informationen über genutzte Ortungsdienste und die Qualität und Aktualität der Daten	Standard
Lage- und Bewegungssensoren	Abfrage der Lage- und Bewegungssensoren als zusätzliche Option	Verbesserung

Tabelle 2.4: Abfrageparameter der Bibliothek

Die Tabellen 2.3 und 2.4 geben an, welche Kriterien Standardfunktionen sind und welche zu den verbesserten Funktionen gehören sollen. (vgl. [Epperlein, 2010](#))

# 3 Entwurf der Bibliothek

## 3.1 Einleitung

In diesem Kapitel werden zunächst die Charakteristiken von Bibliotheken im Bezug auf Latenzen und Programmiermethodik behandelt. Darauf folgt die Entscheidung zur Umsetzung in einer oder mehrerer Programmiersprachen und welche Ortungsdienste genutzt werden sollen. Als nächstes wird die Architektur der Bibliothek beschrieben. Abschließend für dieses Kapitel folgt dann noch die Übersicht über die Implementierung und die Möglichkeiten zur Erweiterung.

### 3.1.1 Charakteristiken von Bibliotheken

A software library is an organized collection of code with associated tools supporting programming in general or in specific domains, usually united by a specified set of principles and conventions. Most libraries are intended for use by several people and in different environments." [Bloch u. a. \(2006\)](#)

Nach Josh Bloch's Definition bei einem Workshop zu Library-Centric Software Design (LCSD) aus dem Jahr 2006 ist eine Software Bibliothek eine organisierte Sammlung von Code mit den dazugehörigen Werkzeugen zur Unterstützung der Programmierung, welche von mehreren Personen in unterschiedlichen Umgebungen genutzt werden.

Hierdurch können Redundanzen vermieden und Ressourcen eingespart werden. Des Weiteren ändern sich Funktion und Rückgabewerte nicht, da alle Programmteile die selben Funktionen nutzen und diese nur an einer zentralen Stelle verändert oder verbessert werden müssen.

#### 3.1.1.1 Latenzen

Latenzen entstehen bei der Benutzung dieser Bibliothek, durch das Bestimmen der Position. Je genauer die Position ermittelt werden soll, desto größer wird die Latenz, da eine genauere Bestimmung der Position es erforderlich macht, mehr Satelliten oder weitere Quellen in die Berechnung einzubeziehen. vgl. [Abschnitt 2.7](#)

#### 3.1.1.2 Programmiermethodik

Bei der Entwicklung einer Bibliothek kommt es hauptsächlich auf zwei wichtige Eigenschaften an: Orthogonalität und Vollständigkeit.

In der Informatik bezeichnet Orthogonalität die freie Kombinierbarkeit formal unabhängiger, einander also nicht überschneidender Konzepte [...]. (Heitzer, 2012, S. 166)

Die Bibliothek sollte also möglichst von allen Diensten, Programmen, Entwicklern, etc. genutzt werden können, die sie benötigen. Sie sollte somit leicht zu benutzen und einzubinden sein und mit unterschiedlichen Programmiersprachen und -umgebungen genutzt werden können. Vollständigkeit bedeutet in diesem Zusammenhang, dass die Bibliothek möglichst zu allen eingehenden Anfragen Methoden zur Verfügung stellt, um dem User ein zufriedenstellendes Ergebnis zurück zu liefern. Sollte für eine Anfrage keine entsprechende Methode implementiert sein, muss die Bibliothek schnell und einfach um eine entsprechende Methode erweiterbar sein.

## 3.2 Grundlagen

In diesem Abschnitt geht es um die gewählten Programmiersprachen, -umgebungen und die Ortungsdienste, die hier zur Verfügung stehen. Bei der Auswahl wurde davon ausgegangen, dass es sich bei den Anwendungen, die diese Bibliothek nutzen werden, um Webangebote und Android Apps handelt. Die Nutzung der Angebote soll über ein mobiles Endgerät erfolgen, welches über entsprechende Ausstattung verfügt, um die Ortungsdienste nutzen zu können.

### 3.2.1 JavaScript

Bei JavaScript handelt es sich um eine Interpretersprache mit objektorientierten Möglichkeiten, die ursprünglich entwickelt wurde, um dynamisches HTML zu ermöglichen, also Inhalte in Webseiten dynamisch generieren, laden oder verändern zu können. JavaScript eignet sich somit sehr gut um bei webbasierten Anwendungen Positionsdaten und Ähnliches abzufragen.

### 3.2.2 Android Library

Google's Android Betriebssystem gehört zu den führenden Smartphone Betriebssystemen und bietet sich somit für die Nutzung von standortbezogenen Anwendungen an, da hier sämtliche Funktionen eines Smartphones oder mobilen Endgerätes genutzt werden können, um den Standort zu ermitteln.

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. Saha. (2008)

Android ist also nicht nur Betriebssystem, sondern bringt auch Schnittstellen und Schlüssel-funktionen mit, die von Entwicklern genutzt werden können. Meist sind diese Funktionen ebenfalls in Bibliotheken zusammen gefasst.

Um hier nicht zwei Bibliotheken entwickeln zu müssen, bietet Android die Möglichkeit, auch als Web Applikation (WebApp) entwickelte Anwendungen in Apps zu integrieren. Hierzu dient das WebView Framework, welches schon Bestandteil von Android ist. Somit lassen sich WebApps auf dem mobilen Endgerät entweder im Browser oder in einer eigenen App nutzen. <sup>1</sup>

#### **Best Practices for Web Apps**

Google hat eigene Richtlinien für die Entwicklung mit WebApps aufgestellt, da die WebApp selbst schon einige Voraussetzungen erfüllen sollte, um problemlos auf mobilen Endgeräten zu arbeiten und dargestellt werden zu können. <sup>2</sup>

**Serverseitige Weiterleitung auf eine mobile Version der Seite** Anhand der vom Web-browser zur Verfügung gestellten Informationen über den User-Agent lässt sich feststellen, ob der User ein mobiles Endgerät benutzt. Somit lässt sich der Zugriff auf eine für mobile Geräte optimierte Version der WebApp umleiten.

**Für mobile Geräte geeigneter DOCTYPE markup** Die meisten WebApps nutzen beispielsweise Sprachen wie XHTML, welche für die Darstellung auf mobilen Geräten optimiert sind. In der WebApp sollte der entsprechende Typ deklariert sein. Ebenso wie die entsprechende Zeichenkodierung, beispielsweise UTF-8.

**Viewport Metadaten zur Größenanpassung** Im Kopf der WebApp sollten Metadaten zur Verfügung gestellt werden, die es dem Browser ermöglichen, die Seite auf die Anzeigegröße des Displays zu skalieren.

**Mehrfache Dateianfragen vermeiden** Um die meist geringere Bandbreite oder ein beschränktes Datenvolumen nicht zu überlasten, sollten Dateien oder Webseitenaufrufe nicht öfter als notwendig erfolgen.

**Vertikales lineares Layout** Es sollte ein Layout genutzt werden, bei dem der User nur eine Scrollrichtung nutzen muss, entweder hoch und runter oder links und rechts.

---

<sup>1</sup>vgl. Google (2016d) - Stand 15.01.2016

<sup>2</sup>vgl. Google (2016a) - Stand 15.01.2016

In Bezug auf die Standortermittlung sollte man hier den Punkt der mehrfachen Dateianfragen erweitern und festlegen, dass auch der Standort nur so genau und so oft ermittelt werden soll, wie notwendig, um die Ressourcen des mobilen Gerätes zu schonen.

#### 3.2.3 Ortungsdienste

Die in [Abschnitt 2.7](#) beschriebenen Ortungstechniken werden heutzutage von den Herstellern der mobilen Endgeräte oder deren Betriebssystem oder Apps in der Regel direkt kombiniert eingesetzt. Im Falle des Android Betriebssystems ist dies die platform location API `android.location`, welche alle generellen Funktionalitäten zur Standortbestimmung mitbringt. Eine wesentlich umfangreichere Lösung stellt hier z.B. die Google Location Services API dar. Diese stellt, laut Google, Daten zu Standort, Bewegung und Genauigkeit zur Verfügung.<sup>3</sup>

Eine per Browser nutzbare Lösung gibt es von W3C, die Geolocation API. Diese lässt per JavaScript die Position des Gerätes auslesen. Die Positionsermittlung erfolgt dabei auf dem Endgerät selbst und wird wiederum mit Hilfe der angesprochenen Techniken, Geotargeting der IP-Adresse, WLAN-Fingerprinting, GSM- und GPS-Ortung, ermittelt und über den Browser auf dem Endgerät abgefragt. Somit ist es möglich, auch in webbasierten Applikationen die Position abzufragen, bzw. die abfragende Webseite per WebApps in eine Android App zu integrieren, ohne eine eigene Android Library zu entwickeln.<sup>4</sup>

## 3.3 Architektur-Übersicht

### 3.3.1 Einleitung

In diesem Abschnitt werden die Implementierung behandelt und die entsprechenden Methoden und ihre Rückgabewerte aufgeführt und diskutiert.

### 3.3.2 JavaScript

Die W3C Geolocation-API bietet einige optional zu definierende Parameter bei der Abfrage der Position:<sup>5</sup>

---

<sup>3</sup>vgl. [Google \(2016c\)](#) - Stand 15.01.2016

<sup>4</sup>vgl. [W3C \(2016\)](#) - Stand 15.01.2016

<sup>5</sup>vgl. [W3C \(2016\)](#) - position options interface, Stand 15.01.2016

**enableHighAccuracy** Diese Option kann auf true oder false gesetzt werden, um dem Gerät mitzuteilen, dass man die bestmögliche Position ermittelt haben möchte oder eben nicht. Ist die Option auf true gesetzt werden alle verfügbaren Möglichkeiten genutzt, um die Position zu bestimmen. Setzt man die Option auf false, wird eine ungenauere Position ermittelt, allerdings werden in diesem Fall die Ressourcen des Endgerätes geschont, indem z.B. Funktionen, die zu einem hohen Stromverbrauch führen (z.B. GPS), vermieden werden. In beiden Fällen entscheidet das Endgerät, welche Möglichkeiten genutzt werden, da einige Funktionen nicht verfügbar oder deaktiviert sein können.

**timeout** In diesem Parameter kann die Zeit in Millisekunden angegeben werden, die vom Aufruf der Funktion zur Positionsbestimmung bis zur erfolgreichen Antwort verstreichen darf. Wird in der angegebenen Zeit keine Position zurückgegeben, erfolgt eine Fehlermeldung. Die Zeit, die der User am Endgerät benötigt, um die entsprechende Abfrage nach der Berechtigung zur Positionsermittlung zu beantworten, wird hier nicht mit eingerechnet.

**maximumAge** Das maximale Alter einer Position gibt an, wie alt eventuell gecachte Positionsdaten sein dürfen, damit diese noch als aktuelle Position akzeptiert werden. Die Zeit wird auch hier in Millisekunden angegeben. Wird der Parameter auf 0 gesetzt, muss sofort eine aktuelle Position ermittelt werden. Ist keine gecachte Position, die nicht älter ist als die angegebene Zeit, verfügbar, muss ebenfalls eine neue Position ermittelt werden.

#### Methoden und Rückgabewerte

In diesem Abschnitt werden für diese Bibliothek sinnvolle Methoden und Rückgabewerte beschrieben, mit denen die Funktionalität der Bibliothek umgesetzt werden soll.

**long** [ `getCurrentPosition()` ] Liefert die aktuelle Position des Endgerätes. Hierzu wird eine aktuelle Position ermittelt und keine gecachten Daten verwendet. Es wird automatisch die bestmögliche Position ermittelt, die durch das Endgerät möglich ist.

**boolean isAtPosition(long latitude, long longitude, long accuracy)** Ermittelt, ob sich das Endgerät an der angegebenen Position befindet. Hierbei wird eine Genauigkeit von accuracy in Metern berücksichtigt.

**boolean isInRange(long latitude, long longitude, long range)** Ermittelt, ob sich das Endgerät in einem Umkreis um eine bestimmte Position befindet. Bei dieser Methode wird



nicht berücksichtigt, in welcher Richtung von der gegebenen Position sich das Endgerät befindet.

**boolean isInFrontOf(long latitude, long longitude, long range)** Ermittelt, ob sich das Endgerät in einem Umkreis um eine bestimmte Position und ob es sich in einer bestimmten Richtung zu dem Punkt befindet. Somit lässt sich feststellen, ob der User sich im Interaktionsraum vor der Medienfassade aufhält.

### 3.3.3 User-Authentifizierung

Dieser Abschnitt befasst sich mit der Authentifizierung des Users um festzustellen, ob er mit dem genutzten Endgerät berechtigt ist, mit dem Public Display zu interagieren. Wie im [Unterabschnitt 2.6.2](#) beschrieben, ist es zu Authentifizierung an der HAW-Hamburg und zur Interaktion mit der Medienfassade notwendig, in das WLAN Netzwerk der HAW-Hamburg eingeloggt zu sein.

Zur Kontrolle reicht es also aus die IP-Adresse des Endgerätes zu überprüfen. Um die Bibliothek alltagstauglicher zu machen, wurden 2 Methoden entworfen:

**String getCurrentIPv4()** Gibt die aktuelle IPv4-Adresse des Endgerätes als Array zurück, wobei die ersten 4 Zellen des Arrays die vier Teile der IP-Adresse enthalten.

**String getCurrentIPv6()** Gibt die aktuelle IPv6-Adresse des Endgerätes als Array zurück, wobei die ersten 4 Zellen des Arrays die vier Teile der IP-Adresse enthalten.

Die mögliche Erweiterung der User-Authentifizierung wird in einem späteren Kapitel noch ausführlicher behandelt. Für die Authentifizierung an der HAW-Hamburg reichen diese beiden Methoden aus.

### 3.3.4 Designentscheidungen im Bezug auf Medienfassaden

Die wichtigsten Eigenschaften im Bezug auf Medienfassaden sind der Umkreis und die Abfrage, ob sich der User bzw. das Endgerät zur Interaktion im Umkreis vor der Medienfassade befindet, da die meisten Interaktionen es erforderlich machen, dass der User sich im Interaktionsraum der Medienfassade aufhält.

Entscheidend ist hier, dass der User sich vor der Medienfassade befindet und die Ortung möglichst genau ausfällt, da der Interaktionsbereich vor dem Public Display meist nur ein paar Meter beträgt und eine zu große Abweichung ließe den User sich unter Umständen zu weit aus dem Interaktionsbereich herausbewegen.

### 3.4 Erweiterbarkeit

Die Erweiterbarkeit der Bibliothek ist ein wichtiger Punkt im Entwurf, da die Bibliothek so gestaltet werden soll, dass sie schnell und einfach um weitere Methoden oder Techniken erweitert werden kann. Die Erweiterung von Methoden bezieht sich unter Anderem darauf, neue Methoden für neue Umgebungen zu implementieren, wenn die Bibliothek z.B. in anderen Umgebungen als der Medienfassade der HAW-Hamburg genutzt werden soll.

Auch eine Nutzung innerhalb eines Gebäudes, z.B. des Living Place an der HAW-Hamburg würde eine Erweiterung der Bibliothek um einige neue Methoden erfordern.

Grade die Indoor-Ortung würde von der Nutzung weiterer Ortungsverfahren profitieren, da die Ortung innerhalb von Gebäuden, wie in [Abschnitt 2.7](#) beschrieben, sehr schnell ungenau werden kann. Auf die möglichen weiteren Ortungstechniken wird in einem späteren Abschnitt genauer eingegangen.

# 4 Evaluation

## 4.1 Einleitung

Die Implementierung der Bibliothek erfolgte nach den in [Kapitel 3](#) erarbeiteten Vorgaben. Hierzu wurden die beschriebenen Frameworks und APIs genutzt, die Ermittlung der IP-Adresse erfolgt über das WebRTC Framework von HTML5, welches beim W3C als offener Standard gilt.

Da die Bibliothek keinerlei grafische Darstellung mit sich bringt, wurde eine einfache Web Application entworfen, die nur dazu dient, die Ergebnisse der Abfragen darzustellen.

Die Entwicklung mit Android WebApps erfolgte in dem von Google bereitgestellten Android Studio in der Version 1.5.1 und auf einem Endgerät mit der Android Version 5.1.1.

Die Web Applications wurden auf einem Apache Webserver der Version 2.4.18 getestet. Als Webbrowser kamen hierfür Mozilla Firefox<sup>1</sup> in Version 44 und 45 und Google Chrome<sup>2</sup> in Version 48 zum Einsatz, da diese auch auf mobilen Android-Endgeräten weit verbreitet sind.

## 4.2 Evaluierung der Implementierung

### 4.2.1 JavaScript

Der erste Schritt der Implementierung war, die W3C Geolocation-API zu nutzen, um die Positionsdaten des Endgerätes zu ermitteln und dann entsprechend aufzuarbeiten. Die Abfragen werden standardmäßig mit folgenden, in [Unterabschnitt 3.3.2](#) erklärten, Parametern durchgeführt:

- enableHighAccuracy: true
- timeout: 5000
- maximumAge: 0

---

<sup>1</sup><https://www.mozilla.org/de/firefox/new/> - Stand: 05.02.2016

<sup>2</sup><https://www.google.com/chrome/> - Stand: 05.02.2016

Mit der Funktion `navigator.geolocation.getCurrentPosition(success, error, options)`; werden dann die Daten abgerufen, bei erfolgreichem Abruf wird die `success`-Funktion aufgerufen, bei einem Fehler die `error`-Funktion und der dritte Parameter enthält die zuvor erwähnten Parameter. Die `success`- und die `error`-Funktion sind sogenannte Callback-Funktionen, die bei erfolgreichem, bzw. fehlerhaftem Abrufen der Daten aufgerufen werden.

Die `success`-Funktion kann nur aufgerufen werden, wenn der User auf dem Endgerät die Berechtigung zur Ermittlung der Position erteilt hat, andernfalls wird die `error`-Funktion aufgerufen und der entsprechende Fehler für die nicht erhaltene Berechtigung ausgegeben.

Die erhaltenen Daten werden nun von den im vorherigen Abschnitt beschriebenen Funktionen verarbeitet und entsprechende Ergebnisse zurückgegeben. Die [Abbildung 4.1](#) zeigt eine einfache

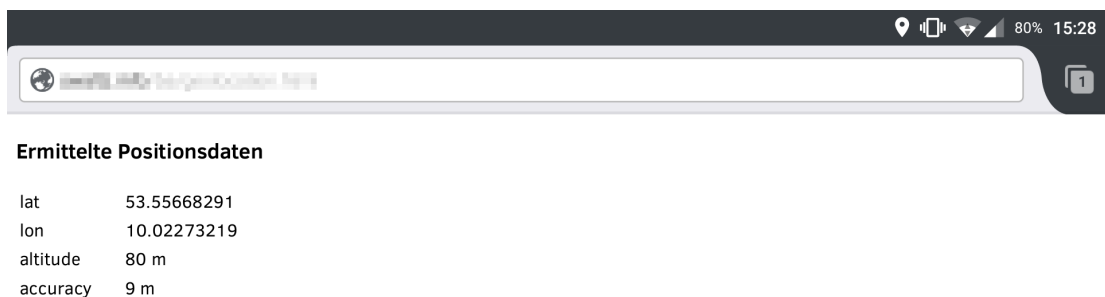


Abbildung 4.1: Ausgabe Positionsdaten Web Applikation

Ausgabe der ermittelten Werte auf einer Webseite ohne weitere Verarbeitung der Daten.

### 4.2.2 Web Application

Die einfachste Web Application ist eine einfache Webseite, die Daten anzeigt. Also wurde dies, wie in [Abbildung 4.1](#) gezeigt, umgesetzt. Diese Webseite nutzt die Bibliothek, um die Standortdaten des Endgerätes darzustellen, wie sie zurückgegeben werden.

Weitere Web Applications können nun intern mit diesen Daten weiter arbeiten und sie zu den benötigten Zwecken nutzen. Wie in den vorherigen Abschnitten beschrieben, soll die Bibliothek noch weitere Funktionen bereitstellen, um zu überprüfen, ob ein Endgerät sich an einem bestimmten Punkt oder in einer bestimmten IP-Range befindet. Wie in [Abbildung 4.2](#) zu sehen ist, werden die zuvor abgefragten Werte entsprechend verarbeitet und als boolean-Wert zurückgeliefert. Da an der HAW-Hamburg keine IPv6-Adressen vergeben werden, ist dieser Wert null.

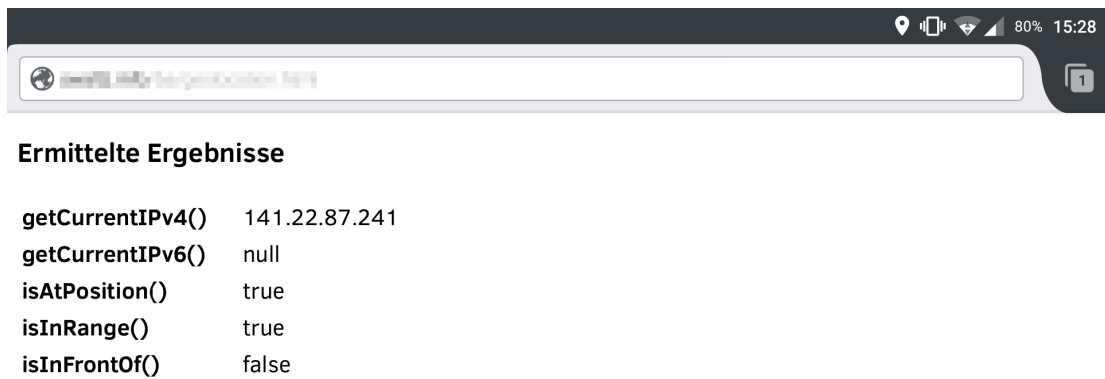


Abbildung 4.2: Ausgabe IP-Adresse und verarbeitete Abfragen

### 4.2.3 Android WebApps

Um eine Web Application in eine Android App zu verpacken, stellt Google, wie schon in [Unterabschnitt 3.2.2](#) erwähnt, das WebView Framework zur Verfügung. Hiermit lassen sich Web Applications einfach in eine Android App einbinden, ohne sie noch einmal neu entwickeln zu müssen. Das WebView Framework erlaubt ebenfalls die Nutzung von JavaScript in den Web Applications und somit kann die entwickelte Bibliothek ohne Weiteres genutzt werden.

### 4.2.4 Messwerte und Genauigkeit

Bei den Messungen vor der Medienfassade der HAW-Hamburg wurden zwei verschiedene Endgeräte genutzt, ein Smartphone mit Mobilfunk- und WLAN-Empfang und ein Tablet, welches kein Mobilfunkmodul beinhaltet und somit nur auf WLAN-Empfang zurückgreifen konnte. Die [Tabelle 4.1](#) zeigt die erhaltenen Messwerte des Smartphones mit dem Parameter `enableHighAccuracy: true`, [Tabelle 4.2](#) dagegen enthält die Messwerte mit dem Parameter `enableHighAccuracy: false`.

#### 4 Evaluation

Nr.	Latitude	Longitude	Altitude	Accuracy
0	53.55674458	10.02267755	99 m	18 m
1	53.55676024	10.02271596	85 m	16 m
2	53.55667883	10.02271575	82 m	18 m
3	53.5566798	10.02272883	80 m	22 m
4	53.5566822	10.02273071	80 m	15 m
5	53.55668239	10.02273146	80 m	15 m
6	53.55668279	10.02273172	80 m	11 m
7	53.55668291	10.02273219	80 m	9 m
8	53.55671237	10.02275453	72 m	15 m
9	53.5567078	10.02274963	71 m	13 m

Tabelle 4.1: Messungen Smartphone Standortabfrage High Accuracy HAW-Hamburg Medienfassade

Die deutlichsten Unterschiede sind hierbei offensichtlich: Stellt man *enableHighAccuracy: false* ein, wird die Höhe nur noch mit 0m ermittelt und die Genauigkeit sinkt auf ca 45m. Diese Werte sind auf keinen Fall ausreichend, um damit die Position zur Interaktion mit der Medienfassade zuverlässig zu bestimmen.

Ist der Parameter *enableHighAccuracy: true* eingestellt, wird für die Höhe ein Wert ermittelt und die Genauigkeit liegt zwischen bestenfalls 9m und im schlechtesten Fall bei 22m. Diese relativ schlechten Werte lassen sich anhand der Umgebung erklären: Die Medienfassade ist umgeben von diversen Hochhäusern und sowohl der WLAN- als auch der Mobilfunk-Empfang sind nicht optimal gegeben (vgl. [Abschnitt 2.7](#)), liegen aber für GPS noch im Durchschnitt, wie in [Tabelle 2.1](#) bereits erwähnt.

Nr.	Latitude	Longitude	Altitude	Accuracy
0	53.5569238	10.0223655	0 m	45.6080017 m
1	53.5569131	10.0223846	0 m	45.0999984 m
2	53.5569131	10.0223846	0 m	45.0999984 m

Tabelle 4.2: Messungen Smartphone Standortabfrage Low Accuracy HAW-Hamburg Medienfassade

#### 4 Evaluation

---

Die gemessenen Werte in **Tabelle 4.3** mit dem Tablet als Endgerät unterscheiden sich teilweise deutlich von den Messwerten des Smartphones. Trotz aktiviertem GPS wurde für die Höhe immer null zurückgeliefert. Die Genauigkeit zeigt hier einige Abweichungen, lässt man diese außen vor, liegt die Genauigkeit zwischen 13m und 29m bei gesetztem *enableHighAccuracy: true*.

Nr.	Latitude	Longitude	Altitude	Accuracy
0	53.5570611	10.022306	null m	52.5 m
1	53.5569258	10.0223519	null m	19.6690006 m
2	53.556917	10.023575	null m	16.4489994 m
3	53.5569202	10.0223504	null m	13.4440002 m
4	53.5569232	10.0223501	null m	13.1339998 m
5	53.557145	10.0223014	null m	48 m
6	53.5570644	10.02232	null m	49.5 m
7	53.5570544	10.0223196	null m	28.5779991 m
8	53.557258	10.0222447	null m	25.8579998 m
9	53.5572817	10.0222367	null m	19.5860004 m

Tabelle 4.3: Messungen Tablet ohne Mobilfunk Standortabfrage High Accuracy HAW-Hamburg Medienfassade

Hier macht es keinen so großen Unterschied, wenn *enableHighAccuracy: false* gesetzt wird, wie **Tabelle 4.4** zeigt. Sieht man wieder von den eindeutigen Ausreißern ab, liegt die Genauigkeit hier zwischen 11m und 28m.

Nr.	Latitude	Longitude	Altitude	Accuracy
0	53.5572589	10.0222457	null m	17.2369995 m
1	53.5572323	10.0222566	null m	16.2789993 m
2	53.5571934	10.0222725	null m	14.7600002 m
3	53.5571989	10.0222714	null m	12.6920004 m
4	53.5571891	10.0222734	null m	12.1110001 m
5	53.5571692	10.0222758	null m	11.0920000 m
6	53.5572471	10.0222502	null m	60 m
7	53.557072	10.022333	null m	45 m
8	53.5570643	10.0223364	null m	27.6040001 m

Tabelle 4.4: Messungen Tablet ohne Mobilfunk Standortabfrage Low Accuracy HAW-Hamburg Medienfassade

#### 4.2.5 Messfehler

Wie die Tabellen 4.1 bis 4.4 zeigen, tauchen vor allem bei Geräten mit eingeschränkten Möglichkeiten zur Standortbestimmung Messfehler bzw. Ausreißer auf, die deutlich von den sonstigen gemessenen Werten abweichen. Die ermittelten Koordinaten sind dennoch bis auf 3 Nachkommastellen gleich. Auch Geräte, die von der Hardwareausstattung her auf mehr Standortdienste zurückgreifen können, haben dennoch eine recht hohe Ungenauigkeit, die noch weiter wächst, wenn der Parameter *enableHighAccuracy: false* gesetzt wird oder das Endgerät diese Einstellung vorgibt. Auch die Ermittlung der Höhe funktioniert scheinbar nur, wenn alle Möglichkeiten mit hoher Genauigkeit genutzt werden können.

Die Ermittlung der IP-Adresse funktionierte in den Test-Messungen immer eindeutig und ohne Fehler.

#### 4.2.6 Erleichterung der Programmierung

Die im Zuge dieser Arbeit erstellte Bibliothek dient zur Erleichterung der Programmierung, indem sie dem Programmierer die Arbeit abnimmt, selbst Funktionen implementieren zu müssen, die prüfen, ob das Endgerät des Users vor der Medienfassade am richtigen Ort ist und ob der User berechtigt ist, mit der Fassade zu interagieren. Werden die entsprechenden Funktionen in der Bibliothek zur Verfügung gestellt, muss der Programmierer sich keine Gedanken darum machen, welche Koordinaten zur Überprüfung genutzt werden müssen, um festzustellen, ob sich das Endgerät im Interaktionsbereich der Fassade oder ob es sich im



richtigen IP-Bereich befindet.

Generell sorgen Bibliotheken dafür, dass weniger redundanter Code entsteht, da die Methoden an einer zentralen Stelle liegen und nicht jedes mal neu implementiert werden müssen. Dies bringt ebenfalls den Vorteil mit sich, dass Anpassungen und Korrekturen nur an dieser einen Stelle in der Bibliothek vorgenommen werden müssen und für alle Stellen in einem Programm gelten, an denen diese Methoden zum Einsatz kommen.

### 4.2.7 Einbindung in Anwendungen

Die Einbindung in Web-Applications geschieht, wie bei JavaScript Dateien üblich, in der jeweiligen Datei der Web-Application, in der die Funktionen genutzt werden sollen. Hierzu wird üblicherweise der Befehl `<script type="text/javascript" src="ordnename/dateiname.js"></script>` genutzt. Nach dem erfolgreichen einbinden der externen JavaScript Datei lassen sich sämtliche Funktionen in der Web-Application nutzen.

Die Nutzung von JavaScript in Android WebApps über WebView gestaltet sich ähnlich, ist allerdings von Haus aus deaktiviert und muss zuerst aktiviert werden. Anschließend lässt sich die Bibliothek ähnlich wie in einer Web-Application einbinden und nutzen.<sup>3</sup>

## 4.3 Fazit

Die Implementierung der Bibliothek erfolgte zunächst als Prototyp, um die Funktionen in der Umgebung der Medienfassade der HAW-Hamburg zu testen, Fehler auszumerzen und die Genauigkeit der Positionsbestimmung festzustellen. Die Messungen haben gezeigt, dass die Genauigkeiten der Standortbestimmung bei Nutzung der höchst möglichen Genauigkeit im Rahmen der Durchschnittswerte für eine GPS Ortung liegen. Fallen einige der Ortungstechniken weg, verringert sich die Genauigkeit geringfügig, allerdings schleichen sich öfters Ausreißer in den Messungen ein, welche im besten Fall nur zu einer erneuten Ortung führen, im schlechtesten Fall aber nicht für eine zuverlässige Ortung genutzt werden können. Auf Grund dieser Ungenauigkeiten lässt sich nicht eindeutig feststellen, ob sich ein User mit seinem Endgerät vor der Medienfassade und somit in ihrem Interaktionsbereich befindet. Die einfache Methode der Authentifizierung über die IP-Adresse an der HAW-Hamburg funktionierte zuverlässig und ohne Probleme.

---

<sup>3</sup>vgl. [Google \(2016b\)](#) UsingJavaScript - Stand 02.02.2016

## 5 Schluss

Im folgenden Abschnitt werden die vorangegangenen Kapitel dieser Arbeit noch einmal zusammengefasst und ein Ausblick auf weitere Entwicklungsmöglichkeiten und interessante Einsatzmöglichkeiten gegeben.

### 5.1 Zusammenfassung

In dieser Arbeit wurde eine Softwarebibliothek zur Unterstützung der Entwickler von Location Based Services entworfen. Die Entwicklerunterstützung basiert auf Funktionen zur Standortbestimmung und der Authentifizierung, mit welchen überprüft werden kann, ob ein User sich mit seinem Endgerät im Interaktionsbereich eines Objektes aufhält und dazu berechtigt ist, mit diesem zu interagieren. Als Umfeld wurde der Campus der HAW-Hamburg gewählt, da dort eine Medienfassade installiert wurde. In dieser Umgebung wurde später auch die Implementierung getestet.

Hierzu wurde zunächst analysiert (s. [Kapitel 2](#)), wie Public Displays arbeiten, was zu einer Geräte basierten Interaktion dazugehört, welche Technologien genutzt werden können, um den Standort eines Gerätes möglichst genau zu ermitteln, und wie eine Interaktion mit öffentlich Wahrnehmbarem ablaufen kann. Neben der Standortbestimmung wurde auch die Möglichkeit der User Authentifizierung analysiert. Da diese immer auf ein entsprechendes System im Hintergrund angewiesen ist, welches Daten von Usern bereitstellt, muss die Authentifizierung an die jeweilige Umgebung angepasst sein, was in diesem Fall bedeutete, dass User, die mit der Medienfassade interagieren wollen, im Netzwerk der Hochschule angemeldet sein müssen. Anhand dieser Analyse wurden letztendlich die Anforderungen an die Bibliothek aufgestellt.

Anhand dieser Anforderungen wurde die Bibliothek in [Kapitel 3](#) entworfen. Den größten Teil machen hierbei die Funktionen rund um die Standortbestimmung aus. Dem Entwickler sollte Arbeit abgenommen werden, indem die Funktionen die ermittelten Standortdaten verarbeiten und nur noch einen boolean-Wert zurückgeben. Hierzu mussten möglichst alle Anfragen abgedeckt werden, die in diesem Kontext auftauchen könnten. Des Weiteren wurde

auf Charakteristiken von Bibliotheken eingegangen, welche Programmierumgebung genutzt werden und wie die Architektur der Bibliothek aussehen sollte.

**Kapitel 4** beschäftigte sich anschließend mit der Evaluation der Implementierung. Hierzu wurde zunächst auf die Programmierumgebung eingegangen. Zu Testzwecken wurden mit dem Prototypen einige Messungen vor der Medienfassade der HAW-Hamburg vorgenommen und eine Auswertung in Bezug auf Genauigkeiten bei unterschiedlichen Voraussetzungen und Messfehlern angefertigt. Zuletzt wurde darauf eingegangen, wie die Bibliothek die Programmierung erleichtert und wie die Einbindung in Anwendungen vonstatten geht.

## 5.2 Ausblick

Zukünftigen Projekten an der HAW-Hamburg, insbesondere im Bereich der Medienfassade und anderen Standort basierten Diensten, können die Ergebnisse dieser Arbeit weiter genutzt werden. Die Anwendungen für die Medienfassade befinden sich größtenteils noch in der Entwicklung und werden ständig weiter- oder neu entwickelt. Die entwickelte Bibliothek bietet hier eine Grundlage an Funktionen, welche schnell und einfach erweitert werden können, wenn neue Anforderungen auftauchen. So lassen sich beispielsweise Funktionen einbauen, die Bewegungen registrieren oder Lageänderungen des Endgerätes anhand der Sensoren in einem Smartphone verarbeiten.

Auch eine Kombination oder Erweiterung der Bibliothek um weitere Ortungsmöglichkeiten ist denkbar. Da der Campus der HAW-Hamburg von vielen Hochhäusern umgeben ist, ist die Ortung dort recht ungenau. Dieses Problem könnte man eventuell beseitigen, indem man Technologien wie Bluetooth (vgl. [Decuir \(2014\)](#)) oder iBeacons, welche oft zur Indoor-Ortung verwendet werden (vgl. [Martin u. a. \(2014\)](#)), nutzt und die Bibliothek dahingehend erweitert. Die Erweiterung mit diesen Technologien könnte die Genauigkeit beträchtlich erhöhen und so auch Positionsbestimmungen ermöglichen, die dazu geeignet sind z.B. Spiele zu realisieren.

Die Medienfassade oder die Interaktion mit Public Displays muss nicht die einzige Nutzungsmöglichkeit dieser Bibliothek bleiben. Location Based Services finden sich heutzutage in vielen Bereichen des Alltages wieder, sie werden nur selten direkt wahrgenommen (Stichwort "Ubiquitous Computing" [Weiser \(1999\)](#)). In Bereichen wie Keyless Entry, also dem schlüssellosen Zugang, kann die Standort abhängige Interaktion und vor allem die Authentifizierung eine wichtige Rolle spielen. Die Entwicklung geht im Automobilbereich schon in die Richtung, dass

man Schlüssel, oder bei Carsharing Anbietern das Smartphone mit entsprechender App, nur noch bei sich tragen muss, um das Fahrzeug zu öffnen und zu starten, da das Fahrzeug erkennt, ob und welcher Nutzer sich in seiner Umgebung befindet. Ähnliche Ansätze sind auch schon im Smart Home Bereich zu finden, bei denen sich Haustüren per Smartphone öffnen lassen, vorausgesetzt der User befindet sich mit dem Gerät in der Umgebung des Gebäudes.

Die Verwendung der Bibliothek trägt zu einem einheitlichen Design bei der Umsetzung von Projekten bei und gewährleistet, dass alle Entwickler mit den selben Werten und Ergebnissen arbeiten. Durch die Entlastung der Entwickler, die sich nicht um die Implementierung der Standortbestimmung kümmern müssen, können sich diese mehr auf ihre Kerntätigkeiten konzentrieren. Auch die Erweiterung oder Anpassung der Bibliothek an einer zentralen Stelle hat die selben Auswirkungen auf alle Implementierungen im gleichen Maße, so dass die erhaltenen Werte und Ergebnisse weiterhin einheitlich bleiben.

## Literaturverzeichnis

- [Bloch u. a. 2006] BLOCH, Josh ; JÄARVI, Jaakko ; MUSSEY, David ; SCHUPP, Sibylle ; SIEK, Jeremy: LCSD: Library-centric Software Design. In: *Companion to the 21st ACM SIGPLAN Symposium on Object-oriented Programming Systems, Languages, and Applications*. New York, NY, USA : ACM, 2006 (OOPSLA '06), S. 618–618. – URL <http://doi.acm.org/10.1145/1176617.1176634>. – ISBN 1-59593-491-X
- [Corbellini u. a. 2015] CORBELLINI, Giorgio ; SCHMID, Stefan ; MANGOLD, Stefan: Two-Way Communication Protocol Using Bluetooth Low Energy Advertisement Frames. In: *Proceedings of the 1st International Workshop on Experiences with the Design and Implementation of Smart Objects*. New York, NY, USA : ACM, 2015 (SmartObjects '15), S. 19–24. – URL <http://doi.acm.org/10.1145/2797044.2797049>. – ISBN 978-1-4503-3535-5
- [Decuir 2014] DECUIR, J.: Introducing Bluetooth Smart: Part 1: A look at both classic and new technologies. In: *Consumer Electronics Magazine, IEEE* 3 (2014), Jan, Nr. 1, S. 12–18. – ISSN 2162-2248
- [DoD 2008] DoD: Global Positioning System Standard Positioning Service Performance Standard / Department of Defense, United States of America. Department of Defense, September 2008. – Forschungsbericht. – URL <http://www.gps.gov>. 4th Edition
- [Eckert 2013] ECKERT, Claudia: *IT-Sicherheit Konzepte - Verfahren - Protokolle*. Rosenheimer Straße 143, D-81671 München : Oldenbourg Wissenschaftsverlag GmbH, 2013. – 8., aktualisierte und korrigierte Auflage
- [Epperlein 2010] EPPERLEIN, Christian: *Qualitätsbewusste Bereitstellung von Standortinformationen durch flexible und intelligente Kombination verschiedener Ortungstechnologien*. Germany, Technische Universität Dresden, Diplomarbeit, June 2010
- [Google 2016a] GOOGLE, Inc.: *Best Practices for Web Apps | Android Developers*. 2016. – URL <http://developer.android.com/guide/webapps/best-practices.html>

- [Google 2016b] GOOGLE, Inc.: *Building Web Apps in WebView | Android Developers*. 2016. – URL <http://developer.android.com/guide/webapps/webview.html>
- [Google 2016c] GOOGLE, Inc.: *Location Strategies | Android Developers*. 2016. – URL <http://developer.android.com/guide/topics/location/strategies.html>
- [Google 2016d] GOOGLE, Inc.: *Web Apps | Android Developers*. 2016. – URL <http://developer.android.com/guide/webapps/index.html>
- [Heitzer 2012] HEITZER, Johanna: Zusammenfassung. In: *Orthogonalität und Approximation*. Vieweg+Teubner Verlag, 2012, S. 161–168. – URL [http://dx.doi.org/10.1007/978-3-8348-8629-3\\_7](http://dx.doi.org/10.1007/978-3-8348-8629-3_7). – ISBN 978-3-8348-1758-7
- [Kupper 2005] KUPPER, Axel: *Location-based Services: Fundamentals and Operation*. John Wiley & Sons, 2005. – ISBN 0470092319
- [Martin u. a. 2014] MARTIN, Paul ; HO, Bo-Jhang ; GRUPEN, Nicholas ; MUNOZ, Samuel ; SRIVASTAVA, Mani: An iBeacon Primer for Indoor Localization: Demo Abstract. In: *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*. New York, NY, USA : ACM, 2014 (BuildSys '14), S. 190–191. – URL <http://doi.acm.org/10.1145/2674061.2675028>. – ISBN 978-1-4503-3144-9
- [Saha. 2008] SAHA., Amit K.: A Developer's First Look At Android. In: *LINUX FOR YOU 1* (2008), January, S. 48–50
- [Sauter 2004] SAUTER, Joachim: Das vierte Format: Die Fassade als mediale Haut der Architektur. In: *Publizierung auf netzspannung.org: http://netzspannung.org/positions/digital-transformations* (2004). – URL <http://netzspannung.org/cat/servlet/CatServlet?cmd=netzkollektor&subCommand=showEntry&entryId=257044&lang=de>
- [Skyhook Wireless 2015] SKYHOOK WIRELESS, Inc.: *Skyhook | Global Location and Context Software Products*. 2015. – URL <http://www.skyhookwireless.com/>
- [Uckelmann u. a. 2011] UCKELMANN, Dieter ; HARRISON, Mark ; MICHAELLES, Florian: *Architecting the Internet of Things*. Kap. An Architectural Approach Towards the Future Internet of Things, S. 1–24. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011. – URL

[http://dx.doi.org/10.1007/978-3-642-19157-2\\_1](http://dx.doi.org/10.1007/978-3-642-19157-2_1). – ISBN 978-3-642-19157-2

[W3C 2016] W3C: *Geolocation API Specification*. 2016. – URL <https://www.w3.org/TR/geolocation-API/>

[Want 2011] WANT, R.: Near field communication. In: *Pervasive Computing, IEEE* 10 (2011), July, Nr. 3, S. 4–7. – ISSN 1536-1268

[Weiser 1999] WEISER, Mark: The Computer for the 21st Century. In: *SIGMOBILE Mob. Comput. Commun. Rev.* 3 (1999), Juli, Nr. 3, S. 3–11. – URL <http://doi.acm.org/10.1145/329124.329126>. – ISSN 1559-1662

*Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.*

Hamburg, 16. Februar 2016

---

Sascha Waltz