

Harald Urwalek

# Potentiale von Smartwatches für Audience-Response-Systeme

**Masterarbeit**

Technische Universität Graz

Institut für Informationssysteme und Computer Medien  
Vorstand: Univ.-Prof. Dipl.-Ing. Dr.techn. Kappe Frank

Betreuer: Univ.-Doz. Dipl.-Ing. Dr.techn. Ebner Martin

Graz, Jänner 2016



# Eidesstattliche Erklärung<sup>1</sup>

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz am, \_\_\_\_\_  
Datum

\_\_\_\_\_  
Unterschrift

# Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, \_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

---

<sup>1</sup> Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008  
Genehmigung des Senates am 1.12.2008



# Kurzfassung

Die Vernetzung des alltäglichen Lebens mit der digitalen Welt mittels der Einbettung von Computern in Gebrauchsgegenstände erweist sich als stetig voranschreitender Prozess, der durch den Begriff Ubiquitous Computing geprägt ist. Der Besitz eines Smartphone gilt mittlerweile als der Regelfall, eine flächendeckende Verbreitung von Smartwatches könnte in den nächsten Jahren folgen. Im Bildungssektor finden die mobilen, ubiquitären Technologien vermehrt Einzug. Dies ermöglicht den Unterricht interaktiver zu gestalten und einen nahtlosen Übergang zwischen der realen und der digitalen Welt zu schaffen. Die persönliche Kommunikation im Rahmen einer Vorlesung kann durch Audience-Response-Systeme ergänzt werden. Diese bieten den Vortragenden und dem Publikum neben der Durchführung von Umfragen bei Massenvorlesungen einen weiteren quantitativen Kommunikationskanal - den Backchannel. Dabei verwenden die Teilnehmer und Teilnehmerinnen ihre privaten Smartphones oder andere mobile Geräte nach dem Bring-your-own-device-Ansatz um anonym und permanent Rückmeldungen zu geben. Um dieses Feedback den Vortragenden leichter zugänglich zu machen, werden mittlerweile tragbare Geräte, wie die Smartwatch, eingesetzt. Ziel dieser Diplomarbeit ist die Integration der Smartwatch in ein bestehendes Audience-Response-System. Das vorgestellte System liefert ein ständiges Echtzeit-Feedback über den Backchannel. Dadurch müssen die Vortragenden in der Lage sein die Informationen schnell wahrzunehmen und zu interpretieren. Um ein reaktionsschnelles Verhalten der Vortragenden Person zu gewährleisten, wird bei diesem Projekt auf Vibrationssignale und auf visuelle Signale zurückgegriffen.

Schlagwörter: Smartwatch, Wearable Computing, Seamless Learning, Audience-Response-System, Bring your own device



# Abstract

The interaction between everyday life and the digital world due to embedded computers is an emerging and continuous growing process, which is coined by the notion Ubiquitous Computing. Owning a smartphone is the norm nowadays. A comprehensive pervasion of the smartwatch could follow in the next years. More and more mobile, ubiquitous technology is used in the educational sector. This makes lectures interactive and realizes a seamless switch between the real and the digital world. The personal communication in huge classrooms can be complemented by Audience Response Systems. They provide for speakers and their auditorium beside performing surveys in large lectures another quantitative communication channel - the backchannel. The audience can use their private smartphones or other mobile devices - a policy which is called Bring your own device - for voting anonymously and permanently. The usage of wearable devices like the smartwatch can ease the access to the feedback for the lecturer. This thesis focuses on the integration of the smartwatch in an existing Audience Response System. The applied system provides a continuous live feedback over the backchannel. Therefore the lecturer needs to notice and interpret the information in time. Vibration and visual signals are used to ensure a quick-response behaviour of the lecturer.

keywords: Smartwatch, Wearable Computing, Seamless Learning, Audience Response System, Bring your own device





# Danksagung

Zu allererst möchte ich mich bei meinem Betreuer Herrn Univ.-Doz. DI Dr. Martin Ebner bedanken. Er hat mir dieses Thema vorgeschlagen und mein Interesse an ihm geweckt. Neben wertvollen fachlichen Ratschlägen stellte er mir ebenfalls die technischen Geräte zur Verfügung.

Ein weiterer Dank gilt Frau DI Karin Pichler und Herrn DI Christian Haintz von der Firma Carrot & Company GmbH. Bei Fragen zu ihrem System standen sie stets zur Verfügung und brachten bei diversen Terminen ihr Fachwissen ein.

Abschließend möchte ich mich noch bei meiner Familie für ihre Unterstützung und ihr Durchhaltevermögen bedanken.



# Inhaltsverzeichnis

1	Einleitung .....	1
1.1	Aufbau der Arbeit .....	2
2	Definitionen .....	3
2.1	Ubiquitous Computing.....	3
2.2	Wearable Computing.....	4
2.2.1	Context Awareness .....	5
2.3	Seamless Learning.....	6
2.3.1	Mobilität, mobile Geräte und mobiles Lernen.....	6
2.3.2	Entstehung des SL.....	9
2.3.3	Formales und Informelles Lernen .....	10
2.3.4	kognitive Seamless Learning Framework .....	11
2.3.5	10 Dimensionen des Mobile Assisted Seamless Learning .....	12
2.4	Bring your own device .....	15
2.5	Audience-Response-Systeme.....	16
2.5.1	Einleitung .....	16
2.5.2	Frontchannel und Backchannel.....	17
2.5.3	Feedbackmethoden.....	18
2.5.4	Herausforderungen beim Erhalt des Feedback .....	19
3	Die Smartwatch .....	21
3.1	Geschichte.....	21
3.2	Definition.....	24
3.3	Wahl der Smartwatch .....	25
3.3.1	Entscheidung nach Umgebung.....	26
3.3.2	Entscheidung nach Modell .....	27
4	Stand der Technik .....	29
5	Projektumgebung.....	31
5.1	Backchannel.....	31
5.1.1	Erstellung eines Vortrages .....	32

5.1.2	Teilnehmen und Bewerten eines Vortrages.....	33
5.2	Sony SmartWatch 3.....	34
5.3	Android Wear OS .....	35
5.3.1	Verbindung mit dem Smartphone.....	35
5.3.2	User Interface .....	36
5.3.3	Design Grundsätze.....	38
6	Implementierung.....	41
6.1	Entwicklungsumgebung.....	41
6.2	Design.....	44
6.3	Smartphone App.....	47
6.4	Smartwatch App .....	53
6.5	Use Cases .....	54
6.6	Fehlerhandling .....	58
7	Evaluation.....	61
7.1	Testeinsatz.....	61
7.1.1	Konfiguration des Prototypen.....	62
7.2	Rückmeldungen.....	62
7.2.1	Vortragender .....	62
7.2.2	Teilnehmer .....	63
7.2.3	Entwickler des Backchannel .....	63
7.3	Analyse.....	64
7.3.1	Persönliche Beobachtungen .....	64
7.3.2	Auswertung der Daten.....	66
7.3.3	Potential der Smartwatch.....	68
8	Zusammenfassung .....	71
9	Ausblick .....	73
	Anhang.....	75
	Abkürzungsverzeichnis.....	77
	Literaturverzeichnis .....	79

# Abbildungsverzeichnis

Abbildung 2.1: Framework zur Analyse des mobilen Lernens [Frohberg, Göth & Schwabe, 2009].....	7
Abbildung 2.2: Die Matrix der Lernumgebungen [Chen et al., 2010] .....	10
Abbildung 2.3: kognitive Seamless Learning Framework [Seow et al., 2008]	11
Abbildung 2.4: Visualisierung der 10 Dimensionen von MSL [Wong, 2012]	13
Abbildung 2.5: Learning Hub [Wong, 2014] .....	14
Abbildung 3.1: Die Uhr von Dick Tracy zur Sprachübertragung.....	21
Abbildung 3.2: Seiko RC-1000 [Marshall, 2015]; Timex Datalink [Ellis, 2015]	21
Abbildung 3.3: IBM Linux Watch; Microsoft SPOT watch [Youssef et al., 2005] .....	22
Abbildung 3.4 IBM Watchpad.....	25
Abbildung 4.1: Erstellung einer Veranstaltung bei Tweedback <sup>11</sup> .....	29
Abbildung 4.2: Tweedback Chatwall Eintrag auf der Smartwatch [Cap, Delfs & Vetterick, 2015] .....	30
Abbildung 5.1: Startschirm des Backchannels .....	31
Abbildung 5.2: Schirm für die Vortragenden .....	32
Abbildung 5.3: Schirm für die Zuhörerschaft.....	33
Abbildung 5.4: Sony Smartwatch 3 .....	34
Abbildung 5.5: Beispiel für Suggest, aufgenommen im Emulator .....	36
Abbildung 5.6: Beispiel für Demand, aufgenommen im Emulator .....	37
Abbildung 5.7: Anwendung starten, aufgenommen im Emulator .....	37
Abbildung 6.1: Kommunikation über MessageApi [Hahn, 2015] .....	43
Abbildung 6.2: Icon der Anwendung .....	46
Abbildung 6.3: Startschirm Smartphone App .....	47
Abbildung 6.4: Gebundenes Service [Becker & Pant, 2015].....	48
Abbildung 6.5: Verbindungsstatus Smartphone App .....	55
Abbildung 6.6: Verbindungsstatus Smartwatch App.....	55
Abbildung 6.7: Anzeige einer Rückmeldung auf dem Smartphone .....	56
Abbildung 6.8: Anzeige einer Rückmeldung auf der Smartwatch.....	56
Abbildung 6.9: Einstellen der Lecture ID .....	57
Abbildung 6.10: Auswahl eines Vibrationsmusters.....	57
Abbildung 6.11: Fehlermeldung Verbindung zum Internet.....	58
Abbildung 6.12: Fehlermeldung Google Play Services; Fehlermeldung Verbindung zur Smartwatch.....	59
Abbildung 6.13: Fehlermeldung Smartphone Anwendung nicht gestartet...	59
Abbildung 7.1: Verlauf der Werte über die Zeit.....	65

Abbildung 7.2: Voteverhalten über die Zeit.....	66
Abbildung 7.3: Dimensionen über die Zeit .....	67
Abbildung 7.4: Anwendung im Praxiseinsatz .....	68

## **Auflistungsverzeichnis**

Auflistung 6.1: Prüfen ob Dimension aktualisiert werden kann.....	49
Auflistung 6.2: Instanzieren des GoogleApiClient.....	50
Auflistung 6.3: Abfragen der Knoten mit der NodeApi.....	51
Auflistung 6.4: Senden einer Nachricht mit der MessageApi .....	51
Auflistung 6.5: Eintrag des PhoneListenerService in der Manifestdatei.....	53
Auflistung 6.6: Auszug PhoneListenerService - Erhalt eines MessageEvents	54

# 1 Einleitung

Audience-Response-Systeme - in weiterer Folge ARS abgekürzt - sind unterstützende Tools, die unter anderem im Bereich des mobilen Lernens eingesetzt werden. Durch diese Systeme kann der Informationsaustausch zwischen den Vortragenden und dem Publikum erhöht werden. Vor allem bei Massenvorlesungen, wo eine persönliche Interaktion aufgrund der Teilnehmergröße nur schwer möglich ist, sind zusätzliche Kommunikationskanäle notwendig. Die Kommunikation gilt als zentraler Punkt des mobilen Lernens und kann durch ihre Förderung zu einem tieferen Verständnis des vorgetragenen Themas führen. Mobile Technologie ermöglicht in einer mobilen Lernumgebung neue Kommunikationswege [Frohberg, Göth & Schwabe, 2009]. Einerseits können die Vortragenden durch den Einsatz von ARS die Teilnehmerinnen und Teilnehmer aktivieren und deren Wissensstand bzw. deren Verständnis überprüfen [Judson & Sawada, 2002; Frohberg, Göth & Schwabe, 2009], andererseits erlauben sie dem Auditorium durch die Übermittlung von Fragen bzw. Anmerkungen an die Referenten sowie durch die Bewertung des Vortrages, wie die Geschwindigkeit, selbstständig tätig zu werden. Die Studentinnen und Studenten haben die Möglichkeiten aktiv in den Lehrunterricht einzugreifen und dadurch die Lehrinhalte mitzugestalten. Eine mögliche Passivität beim Frontalunterricht kann aufgebrochen werden und ein interaktives Lernszenario wird geschaffen.

Durch die Verwendung von mobilen, internetfähigen Geräten ergeben sich neue Anwendungsszenarien in der Lehre. Heute liegt der Fokus auf der Frage nach der Integration und Orchestrierung der mobilen Technologie als Mittel zur Lehrunterstützung [Specht, Ebner & Löcker, 2013]. Bezogen auf ARS können Smartphones, Tablets etc. optimal in webbasierte Vertreter ihrer Art integriert werden. Die Zuhörerinnen und Zuhörer verwenden ihre private, personalisierten Geräte entsprechend dem Ansatz von Bring your own device - in weiterer Folge BYOD abgekürzt [siehe Kapitel 2.4]. Damit wird in einer formalen Lernumgebung ein nahtloser Übergang zwischen der realen Welt und den digitalen Informationen während eines Vortrages geschaffen.

Der Backchannel, als spezielle Form eines ARS, wird eingesetzt für Echtzeitrückmeldungen durch die Studentinnen und Studenten, die nicht von den Sprecherinnen und Sprechern initiiert werden müssen. Für

## Einleitung

Vortragende scheint daher ein entscheidender Aspekt die Filterung aller Rückmeldungen nach Relevanz sowie eine Reaktion innerhalb eines angemessenen Zeitraumes. Bei fehlender Reaktion wäre der gewünschte Kommunikationskreislauf gestoppt. Rückmeldungen von ARS werden vorwiegend am primären Vortragsgerät angezeigt, wodurch die Vortragende Person in ihrer Mobilität eingeschränkt ist.

Diese Arbeit beschäftigt sich mit der Anbindung der Smartwatch [siehe Kapitel 3] an ein bereits existierendes ARS - dem Backchannel [siehe Kapitel 5.1]. Dabei sollen die Potentiale der Smartwatch als ein direkt am Körper anliegendes Gerät untersucht und genutzt werden. Mit der Smartwatch sollen die Vortragenden durch leicht wahrnehmbare Vibrationssignale und visuelle Signale auf eine Rückmeldung aufmerksam gemacht werden und somit eine schnellere Reaktion auf das ständig aktualisierte Echtzeit-Feedback des Auditoriums fördern. Das Ziel ist es die Referentin und den Referent noch stärker in das ARS zu integrieren.

Aufgrund des relativ kleinen Displays dient die Anwendung auf der Smartwatch rein der Visualisierung von Rückmeldungen. Andere Funktionalitäten wie der Verbindungsaufbau zum Backchannel, die Einstellungen etc. erfolgen auf der Anwendung am Smartphone, welches mit der Smartwatch gekoppelt ist. Die entscheidenden Faktoren beim Feedback sind die rasche Wahrnehmbarkeit und die intuitive Interpretierbarkeit. Des Weiteren liegt der Fokus auf eine flüssige Fortsetzung des Vortrages und minimale Ablenkung durch die Smartwatch. Permanente Rückmeldungen sind zu vermeiden, ein Mechanismus zur Filterung ist daher unabdingbar.

### **1.1 Aufbau der Arbeit**

Die Arbeit gliedert sich in mehrere Kapitel, um eine schrittweise Umsetzung des Projektes zu gewährleisten. Das Kapitel 2 beschäftigt sich mit den Definitionen und ARS. Im Kapitel 3 wird auf die Geschichte der Smartwatch eingegangen und die Auswahl des passenden Modelles erläutert. Folgend wird im Kapitel 4 der Stand der Technik bei Smartwatches in Verbindung mit ARS dargelegt. Die festgelegte Projektumgebung wird im Kapitel 5 beleuchtet. Im Kapitel 6 folgen die technischen Details zur Implementierung der Anwendung. Das Kapitel 7 widmet sich dem Testeinsatz und der Bewertung der Ergebnisse. Mit einer Zusammenfassung in Kapitel 8 und einem Ausblick in Kapitel 9 wird die Arbeit abgeschlossen.



## 2 Definitionen

### 2.1 Ubiquitous Computing

Ubiquitous Computing - in weiterer Folge UC abgekürzt - beschreibt die Allgegenwärtigkeit von Computern im Alltag. Dazu werden sie in Gebrauchsgegenstände eingebettet. In der Literatur wird UC oft mit Pervasive Computing (die durchdringende Vernetzung mit Computern) gleichgesetzt [Satyanarayanan, 2001; Saha & Mukherjee, 2003]. Den Begriff des UC prägte Weiser [1991] im Jahr 1991 mit seinem Bericht "The Computer for the 21st Century". In seiner Vision werden Computer in einer sehr hohen Anzahl nahtlos in die Welt bzw. in die Alltagsgegenstände integriert.

*"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."* [Weiser, 1991 - Seite 94]

Die beiden entscheidenden Kernpunkte sind die Ubiquität und die Unsichtbarkeit der Technologie. Unter Unsichtbarkeit versteht Weiser [1991] das Verschwinden der Technologie – sie wird unbewusst, automatisch verwendet. Dabei führt er als Beispiel den elektrischen Motor an. Des Weiteren sieht der Autor UC als Gegenteil von Virtual Reality. Während bei Virtual Reality der Mensch in die Welt des Computers eindringt, versucht UC den Computer mit der realen Welt zu verknüpfen – der Begriff "embodied virtuality" soll den Gegensatz betonen.

Satyanarayanan [2001] nimmt Mobile Computing - in weiterer Folge MC abgekürzt - als Voraussetzung für UC, mit der Begründung, dass Mobilität als wesentlicher Bestandteil des Lebens durch die Technologie unterstützt werden muss, um ihre Allgegenwärtigkeit zu gewährleisten. Satyanarayanan [2001] führt vier ergänzende Bedingungen für MC an, die für die Realisierung von UC notwendig sind:

- Smart spaces: Die Einbettung von Computern in die Gebäudestruktur.
- Invisibility: Das Verschwinden der Technologie aus dem Bewusstsein – ihre Verwendung erfolgt unterbewusst.
- Localized scalability: Die Interaktion über große Distanzen soll auf ein Minimum reduziert werden.

## Definitionen

- Masking Uneven Conditioning: Die Maskierung der Unterschiede der verschiedenen technologischen Umgebungen um unerwartetes Verhalten zu vermeiden.

Saha & Mukherjee [2003] sehen als weiteren Aspekt die Fähigkeit der Systeme bzw. der Geräte den Kontext erkennen zu können und diesen entsprechend zu nutzen - für eine gesteigerte Qualität der Interaktion zwischen Mensch und Maschine. Nur eine intelligente Umgebung ermöglicht proaktives Handeln der Computer.

## 2.2 Wearable Computing

Wearable Computing versteht sich als Wissenschaft für tragbare Computersysteme, für die sich der Begriff Wearable eingebürgert hat. Ein Wearable wird von Starner [2014] wie folgt definiert:

*"A wearable Computer is any body-worn computer that is designed to provide useful services while the user is performing other tasks."* [Starner, 2014 - Seite 10]

Der Einsatzbereich von Wearables erweist sich als sehr vielfältig. Entweder sind sie für einen limitierten Einsatz konzipiert (beispielsweise Fitnessbänder) oder sie erfüllen multifunktionale Aufgaben wie das Smartphone oder die Smartwatch. Rhodes [1997] legte typische Charakteristiken von Wearables fest:

### Tragbar im Betrieb

Ein Wearable kann verwendet werden während die Trägerin und der Träger sich in Bewegung befindet. Dies erweist sich als grundlegenden Unterschied gegenüber Desktop-PCs oder Laptops.

### Freihändige Verwendung

Die Interaktion erfolgt freihändig mittels Spracheingabe, Körpergesten und Sprachausgabe oder durch minimalem Einsatz der Hände über vordefinierte Wähldialoge, Joysticks etc.

### Sensoren

Im Gerät sind Sensoren zur Erfassung der realen Umgebung integriert. Dies umfasst die kabellose Kommunikation, GPS, Kameras oder Mikrofone - beim heutigen Stand der Technik noch um einiges mehr.

### Proaktiv

Ein Wearable arbeitet voraushandelnd, indem es Informationen aufbereitet bzw. der Benutzerin und dem Benutzer zur Verfügung stellt, auch wenn es nicht aktiv genutzt wird.

### Immer eingeschalten

Grundsätzlich sind die Geräte ständig in Betrieb. Dabei führen sie selbstständig Hintergrundoperationen durch und nehmen die Umgebung über die Sensoren wahr.

## **2.2.1 Context Awareness**

Die Einbeziehung des Kontexts spielt eine entscheidende Rolle beim Wearable Computing. Die Definition für den Kontext nach Dey & Abowd [2000] lautet:

*"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves."* [Dey & Abowd, 2000 - Seiten 3-4]

Als Kontext versteht sich folglich jede Information, mit welcher die Umgebung beschrieben werden kann - beispielsweise der Ort, die Zeit, die Beleuchtung, Geräusche. Da Wearables während der Bewegung verwendet werden, ändert sich der Kontext kontinuierlich. Ihr Verhalten soll sich dem ändernden Kontext anpassen.

*"The changing context can be used to adapt the user interface to an application, providing relevant services and information to the user."* [Dey et al., 1999 - Seite 21]

*"A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task."* [Dey & Abowd, 2000 - Seiten 6-7]

Um die Umgebung stärker wahrnehmen zu können, werden Sensoren in Wearables verbaut. Damit kann das Verständnis des Computers für den Kontext gesteigert werden und sich dadurch die Qualität in der Mensch-Maschine-Interaktion erhöhen [Dey et al., 1999]. Anwendungsprogramme die Informationen aus der realen Welt heranziehen gelten als context-aware.

## 2.3 Seamless Learning

In der Literatur findet sich der Begriff des Seamless Learning - in weiterer Folge SL abgekürzt - wieder, welcher wörtlich übersetzt nahtloses Lernen bedeutet. Durch SL sollen Brüche zwischen formalen und informellen Lernsettings, zwischen personalisierter und sozial eingebetteter Lernunterstützung, zwischen verschiedenen Lernzeiten und Lernorten, zwischen physikalischer Umgebung und digitalen Informationen, zwischen verschiedenen Geräten sowie zwischen verschiedenen Lernaufgaben und Lernaktivitäten überbrückt werden [Specht, Ebner & Löcker, 2013].

### 2.3.1 Mobilität, mobile Geräte und mobiles Lernen

Nach Sharples, Taylor & Vavoula [2007] formt jede Technologieära zu einem gewissen Ausmaß den Bildungssektor, da der technologische Einfluss in Konvergenz zu Theorien und Praktiken im Bildungsbereich steht. Die Autoren So, Kim & Looi [2008] führen als einflussreiche Kriterien die Portabilität und die Vielseitigkeit der mobilen Geräte an. Ihrer Ansicht nach haben die mobilen Geräte das Potential eine pädagogische Verlagerung der Lernumgebung von den Lehrenden auf den Lernenden zu unterstützen. Der Lernende kann selbstständig agieren und seinen Lernweg bestimmen, der nicht einen vorgegebenen Pfad folgen muss. Looi et al. [2010] sehen dank der mittlerweile allgegenwärtigen Verfügbarkeit von mobilen Geräte eine Veränderung der Bedeutung von Ort und Zeit beim Lernen - Lernen ist weder an den Klassenraum noch an einen Lehrplan gebunden. Es können nahtlosen Übergänge zwischen den einzelnen Lernumgebungen bzw. Kontexten geschaffen werden. Dabei sehen Sharples, Taylor & Vavoula [2007] den Kontext nicht nur als Dinge die uns umgeben, von denen Informationen gewonnen werden können, sondern meinen, dass dieser ebenso durch Lernen geschaffen wird. Beispielweise wird der Kontext durch das Zusammentreffen von Personen mit dem gleichen Interesse geformt - ob persönlich (reale Welt) oder in sozialen Netzwerken (digitale Welt).

Der Ansatz von MC und UC nimmt merklich Einfluss auf das Lernen mit digitalen Technologien. Während traditionelles E-Learning vor einem Desktop-PCs stattfindet, finden sich in neuern computerunterstützten Lernumgebungen mehr mobile und ubiquitäre Elemente wieder. Eine frühere Definitionen des mobilen Lernens nach Quinn [2000] bezeichnet mobiles Lernen als E-Learning mit mobilen Geräten wie Palmtops. Dabei war der entscheidende Faktor die Mobilität der Technologie. Spätere

Definitionen des mobilen Lernens sehen die Mobilität des Lernenden als zentrales Kriterium [Specht, Ebner & Löcker, 2013].

Sharples, Taylor & Vavoula [2007] bezeichnen mobiles Lernen als Lernen mit portierbarer Technologie sowie Lernen, das durch die Mobilität der Personen und des Wissens geprägt ist. Die Mobilität beim mobilen Lernen kann auf unterschiedliche Aspekte bezogen werden: Mobilität des Standorts, Mobilität der Technologie, Mobilität des Lernthemas, Mobilität der Lernumgebung (bezogen auf die Gesellschaft) und Mobilität des Zeitpunktes des Lernens [Sharples et al., 2009]. Aufgrund dieser Mobilität ändert sich auch jeweils der aktuelle Kontext des Lernens. Beispielsweise sieht sich eine Person am Vormittag mit einem formalen Kontext im Klassenraum konfrontiert, am Nachmittag befindet sie sich in einem informellen Kontext zuhause.

Sharples, Taylor & Vavoula [2007] erstellten ein Framework zur Analyse des mobilen Lernens. Jedes Projekt des mobilen Lernens kann anhand der identifizierten Faktoren in einem strukturierten Weg beschrieben werden [Frohberg, Göth & Schwabe, 2009].

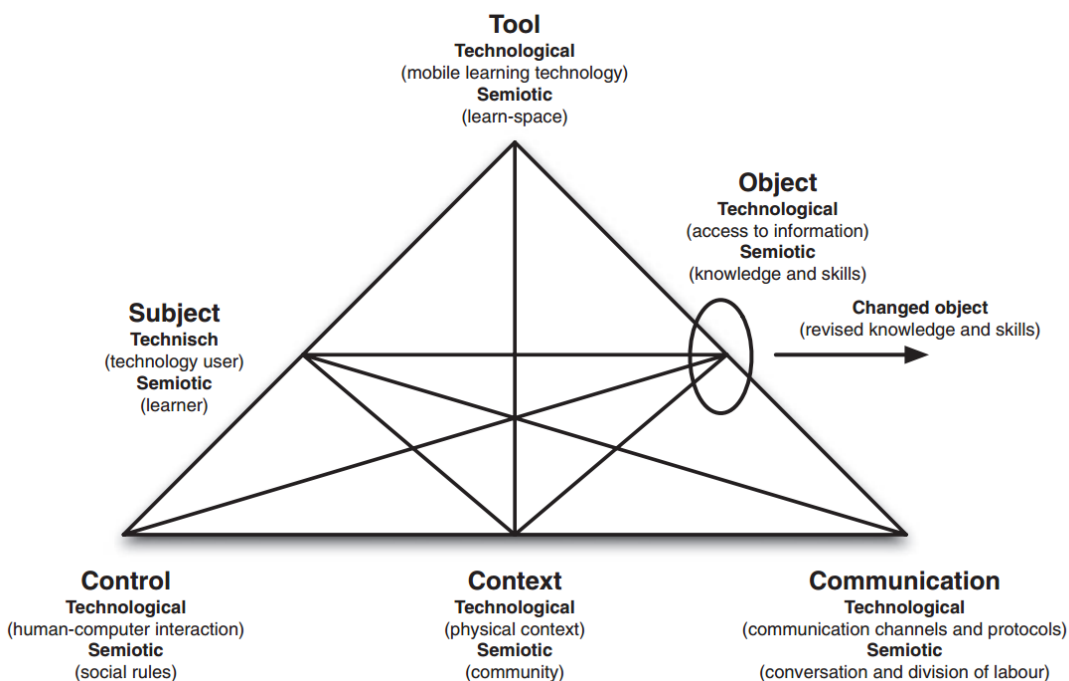


Abbildung 2.1: Framework zur Analyse des mobilen Lernens [Frohberg, Göth & Schwabe, 2009]

## Definitionen

Die Dimensionen des Framework sind der Kontext (wo und wann?), die Werkzeuge (womit?), Kontrolle (wie?), Kommunikation (mit wem?), Subjekt (wer?) und Lernziel (was?) [siehe Abbildung 2.1] - Übersetzung von Specht, Ebner & Löcker [2013]. Frohberg, Göth & Schwabe [2009] klassifizierten anhand dieses Frameworks mobile Lernprojekte. Für den Faktor Kontext legten die Autoren vier Optionen fest - den unabhängigen, den formalen, den physischen und den sozialen Kontext. Dabei nannten sie ARS als typisches Beispiel für den formalen Kontext.

So, Kim & Looi [2008] sehen die Mobilität aufgrund ihrer kontinuierlichen Eigenschaft als einen ausschlagenden Faktor für mobile Lernszenarien. Sie ermöglichen eine Verlagerung der Kommunikation von direkt zwischen zwei Personen auf  $n$  zu  $n$  Personen, der individuellen zur gemeinschaftlichen Interaktion und von einem zentralisierten zu einem dezentralisierten System.

Die kontextübergreifende Sicht des Technology Enhanced Learning in Kombination mit kabellosen Technologien und UC wird als Ubiquitous Learning bezeichnet. Die Einbeziehung des Kontext gilt bei Ubiquitous Learning als Kernkonzept [Milrad et al., 2013]. Chan et al. [2006] nennen sechs Eigenschaften von mobilen, ubiquitären Geräten, die ihre Verwendung im Ausbildungssektor interessant machen: die Portabilität, die Unterstützung sozialer Interaktivität, die individuelle Anpassbarkeit, die Kontextsensibilität, die Verbindbarkeit untereinander und die Verknüpfung der realen und der digitalen Welt.

Chen et al. [2010] führten Charakteristiken von SL an und nennen Wege wie mobile Technologien als Unterstützung für SL eingesetzt werden können. Unter anderem ermöglichen mobile Geräte ein zeit- und ortsunabhängiges Lernen und sind zum Datenaustausch, Kommunikation oder Zusammenarbeit in Gruppen einsetzbar. Specht, Ebner & Löcker [2013] schlagen in dieselbe Kerbe und meinen, dass sich eine ubiquitäre Lernunterstützung aus dem mobilen Lernen und der Verbindung zu allgegenwärtigen Technologien entwickelt.

### 2.3.2 Entstehung des SL

Wong [2014] berichtet von zwei unterschiedlichen und voneinander unabhängigen Entstehungsgeschichten des SL. Auf der einen Seite entwickelte sich SL aufgrund von Reformen im Hochschulwesen, wobei Kuh [1996] angeführt wird, der den Begriff wie folgt definiert:

*"The word seamless suggests that what was once believed to be separate, distinct parts (e.g., in-class and out-of-class, academic and non-academic, curricular and co-curricular, or on-campus and off-campus experiences) are now of one piece, bound together so as to appear whole or continuous. In seamless learning environments, students are encouraged to take advantage of learning resources that exist both inside and outside the classroom. ... Students are asked to use their life experiences to make meaning of material introduced in classes..." [Kuh, 1996 - Seite 136]*

Auf der anderen Seite entwickelte sich SL laut Wong [2014] aus dem Feld des Mobile Ubiquitous Learning. Verwiesen wird auf Chan et al. [2006], die eine Neuausrichtung des SL im Kontext des Technology Enhanced Learning sehen. Laut Specht, Ebner & Löcker [2013] erkennen Chan et al. [2006] eine neue Verbindung zwischen SL und dem Technology Enhanced Learning.

*"We see that ready-to-hand access creates the potential for a new phase in the evolution of technology-enhanced learning (TEL), characterized by "seamless learning spaces" and marked by continuity of the learning experience across different scenarios (or environments), and emerging from the availability of one device or more per student ("one-to-one")." [Chan et al., 2006 - Seite 5]*

*"Seamless learning implies that a student can learn whenever they are curious in a variety of scenarios and that they can switch from one scenario to another easily and quickly using the personal device as a mediator." [Chan et al., 2006 - Seite 6]*

Wong [2014] identifiziert als kleinsten gemeinsamen Nenner dieser zwei Entstehungsgeschichten des SL die dauerhafte, individuelle Lernerfahrung des Lernenden über multiple Lernumgebungen - vor allem die Verbindung der formalen und der informellen Lernumgebung.

Wang & Looi [2011] führten den Begriff des Mobile Assisted Seamless Learning [siehe Kapitel 2.3.5] für die zweite Entstehungsgeschichte ein. Als ein zentraler Punkt der zweiten Entstehungsgeschichte zeigt sich die ständige Verfügbarkeit von mobilen Geräten für jeden Lernenden. In der Literatur



## Definitionen

wird dabei oft die Notation 1:1 bzw. one-to-one verwendet, was die permanente Ausstattung (24 Stunden an 7 Tagen) eines Lernenden mit de facto mindestens einem mobilen Gerät bedeutet [Wang & Looi, 2011]. Dabei fungiert das mobile Gerät als Mediator, damit die Lernende und der Lernende in einer SL-Umgebung rasch und einfach zwischen den Lernszenarien wechseln kann.

### 2.3.3 Formales und Informelles Lernen

So, Kim & Looi [2008] beschreiben den Unterschied zwischen formalen und informellen Lernen anhand des Auslösers. Formales Lernen wird zu einer bestimmten Uhrzeit entsprechend eines festgelegten Lehrplans initiiert, wohingegen informelles Lernen aufgrund des Eigeninteresses bzw. der Intention des Lernenden stattfindet. Abbildung 2.2 zeigt die vier Lernumgebungen nach Chen et al. [2010] (adaptiert von So, Kim & Looi [2008]).

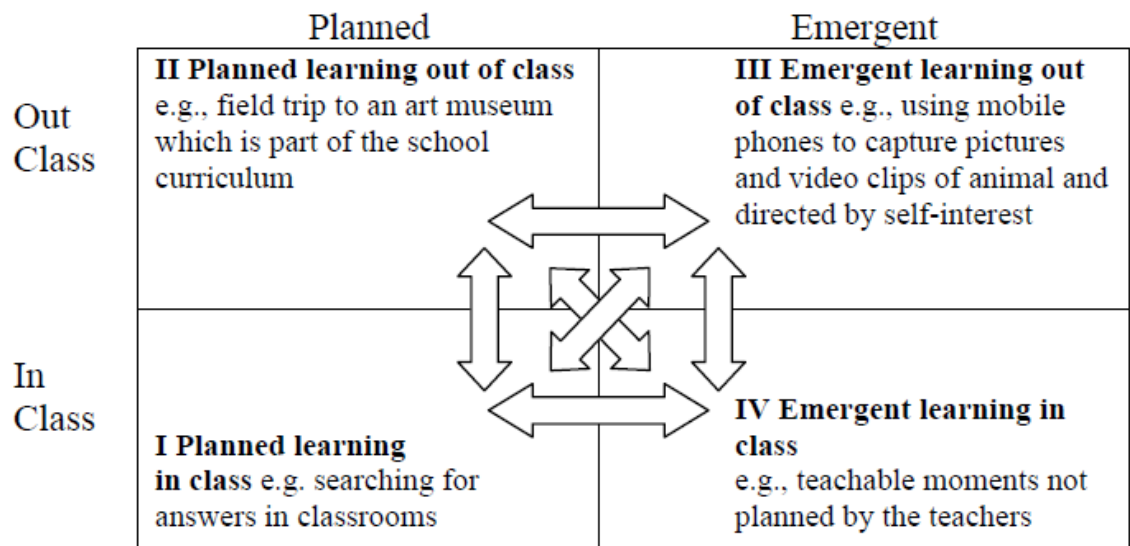


Abbildung 2.2: Die Matrix der Lernumgebungen [Chen et al., 2010]

Dabei werden die Lernumgebungen anhand von zwei Dimensionen klassifiziert - innerhalb bzw. außerhalb der Klasse und geplant bzw. nicht geplant. Der Typ I entspricht der traditionellen Lernumgebung innerhalb des Klassenraumes nach dem Lehrplan. Während Typ II einer geplanten Lernsituation außerhalb des Klassenraumes entspricht, treten in Typ III und Typ IV ungeplante Lernprozesse auf.



Durch SL soll die Grenze zwischen formalen und informellen Lernen verblassen. Chan et al. [2006] sehen in SL das Potential die formale Lernzeit auf die informelle Lernzeit zu erweitern. Aufgrund mobiler Technologien können nahtlose Lernszenarien geschaffen werden, die den Übergang zwischen formalen und informellen Lernen überbrücken [So, Kim & Looi, 2008]. Die Mobilität der Geräte ermöglicht ihre Mitnahme in jene Umgebung (Kontext), in der das Lernen den meisten Sinn ergibt [Frohberg, Göth & Schwabe, 2009].

### 2.3.4 kognitive Seamless Learning Framework

Lernen bzw. kognitive Prozesse bei SL findet durch drei unterschiedliche Prinzipien, jeweils innerhalb eines Kontexts, statt [Seow et al., 2008; Looi et al., 2010]:

- individuelles Lernen in der privaten Lernumgebung
- gemeinschaftliches Lernen in der öffentlichen Lernumgebung
- durch kognitive Artefakte, die unabhängig von Zeit und Raum mit den kognitiven Werkzeugen erstellt werden

Seow et al. [2008] erstellten ein Framework [siehe Abbildung 2.3], welches die verteilten kognitiven Prozesse beim SL visualisiert.

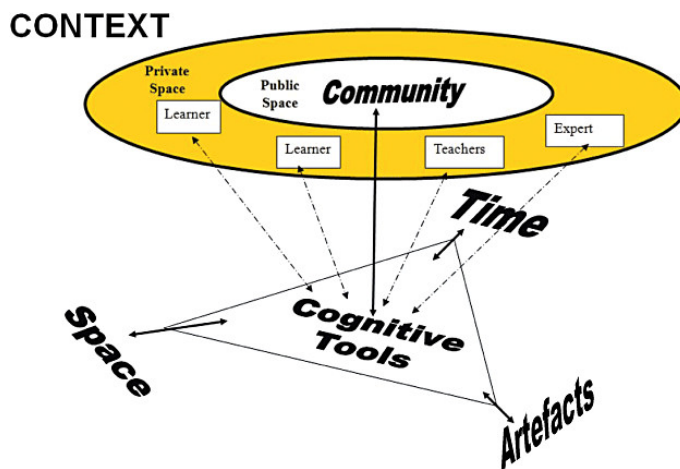


Abbildung 2.3: kognitive Seamless Learning Framework [Seow et al., 2008]

Das Framework besteht aus sechs Komponenten, die zum Lernen in einer SL-Umgebung beitragen: Raum, Zeit, Kontext, die Gemeinschaft, kognitive Werkzeuge und kognitive Artefakte. Als Beispiele für kognitive Werkzeuge in ihren Studien nannten Seow et al. [2008] mobile Geräte zur

Datensammlung und ein Online-Portal zum Hochladen der Daten. Die kognitiven Artefakte waren die von den Studentinnen und Studenten erstellten Dokumente, mit denen die anderen Teilnehmerinnen und Teilnehmer wiederum ihr Wissen erweitern konnten.

### 2.3.5 10 Dimensionen des Mobile Assisted Seamless Learning

Wong & Looi [2011] führten eine Literatursuche mit dem Begriff „Seamless Learning“ in Verbindung mit „Mobile Learning oder Ubiquitous Learning oder Handheld“ und mit dem Begriff „Seamless Mobile Learning“ durch und identifizierten im Rahmen ihrer Analyse zehn signifikante Charakteristiken des Mobile Seamless Learning – in weiterer Folge MSL abgekürzt - die MSL aus Sicht des Lernenden beschreiben:

- MSL 1: die Dimension des formalen und informellen Lernens
- MSL 2: die Aspekte des persönlichen und des gemeinsamen Lernens
- MSL 3: die zeitliche Unabhängigkeit im Lernen
- MSL 4: das standortübergreifende bzw. ortunabhängige Lernen
- MSL 5: die Dimension der allgegenwärtigen Verfügbarkeit von Lernressourcen
- MSL 6: die Präsenz sowohl der realen als auch der digitalen Welt im Lernprozess
- MSL 7: ein kombinierter Einsatz von verschiedenen Gerätetypen
- MSL 8: eine nahtlose und schnelle Umschaltung zwischen verschiedenen Lernaufgaben
- MSL 9: die Wissenssynthese: Syntheseprozesse zwischen schon vorhandenem und neuem Wissen
- MSL 10: pädagogische Modelle zur Unterstützung des nahtlosen mobilen Lernens durch die Lehrpersonen

Die Übersetzung ins Deutsche wurde von Specht, Ebner & Löcker [2013] übernommen und leicht modifiziert.

Wong [2012] griff diese zehn Dimensionen auf und führte eine Hierarchie für sie ein, die er anschließend visualisierte [siehe Abbildung 2.4]. Dabei wurden MSL3 (zeitliche Unabhängigkeit) und MSL4 (örtliche Unabhängigkeit) als die höchsten und universellsten Dimensionen eingestuft (im Diagramm scheint der Text bei MSL3 und MSL4 vertauscht). Innerhalb dieses zweidimensionalen Raumes befindet sich eine Vereinigung von drei Sub-Räumen - MSL1 (formale und informelle Lernen), MSL2 (persönliche und

gemeinsame Lernen) und MSL6 (Lernen durch die reale und die digitale Welt). MSL5 (allgegenwärtiger Zugriff auf Lernressourcen) und MSL10 (pädagogische Modelle) sind laut Wong [2012] im Diagramm als zwei externe Input anzusehen, welche der Initiierung und der Steigerung der Lernaufgabe dienen. Das pädagogische Modell (MSL10) ist für den Lernenden in einer formalen Lernumgebung relevant, während die allgegenwärtigen Lernressourcen (MSL5) auf die digitale Lernumgebung Einfluss haben. Innerhalb der Lernumgebung greift der Lernende je nach Bedarf auf unterschiedliche Geräte zurück (MSL7). Das Zusammenspiel der genannten Dimensionen erlaubt es den Lernenden zwischen den multiplen Lernaufgaben nahtlos zu wechseln (MSL8) und soll zur Wissenssynthese (MSL9) führen.

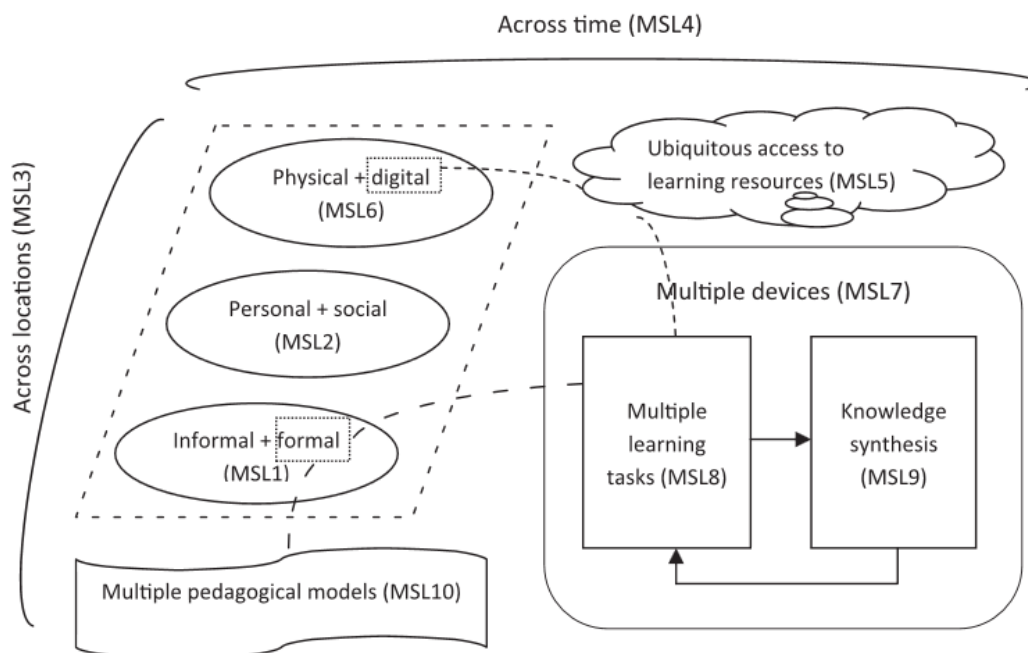


Abbildung 2.4: Visualisierung der 10 Dimensionen von MSL [Wong, 2012]

Milrad et al. [2013] sehen zwei essentielle Funktionen von SL – Seamless Adaptivity und Seamless Connectivity. Unter Seamless Adaptivity verstehen die Autoren die Anpassung der Technologie an die Lernende und den Lernenden, ohne dass es ihr und ihm bewusst ist. Beispielsweise durch das an die lernende Person und des Wissenstandes angepasste Anbieten von Lerninhalten und Services. Die Seamless Connectivity ermöglicht eine

## Definitionen

kontinuierliche Lernerfahrung durch das Aufrechterhalten des Lernens auch bei der Verwendung mehrerer mobiler Geräte und über eine Lernhistorie – es kann dort angefangen werden, wo aufgehört wurde und die bisherigen Lernaktivitäten werden für den Lernenden nachvollziehbar.

Der Learning Hub [siehe Abbildung 2.5] stellt einen zentralen, individuellen und persönlich gestalteten Mittelpunkt des Lernens dar, der nach Wong [2014] die Lernaktivitäten durch ein Angebot unterstützt und den Lernverlauf transparent macht. Laut Wong [2012] soll im Sinne von MSL der Learning Hub nicht mit fixer Hardware assoziiert werden. Anstelle existieren für jeden Lernende Konten zu einer Cloud-basierten Lernplattform, damit der Zugriff auf Lernressourcen von multiplen Geräten (MSL7) möglich ist.

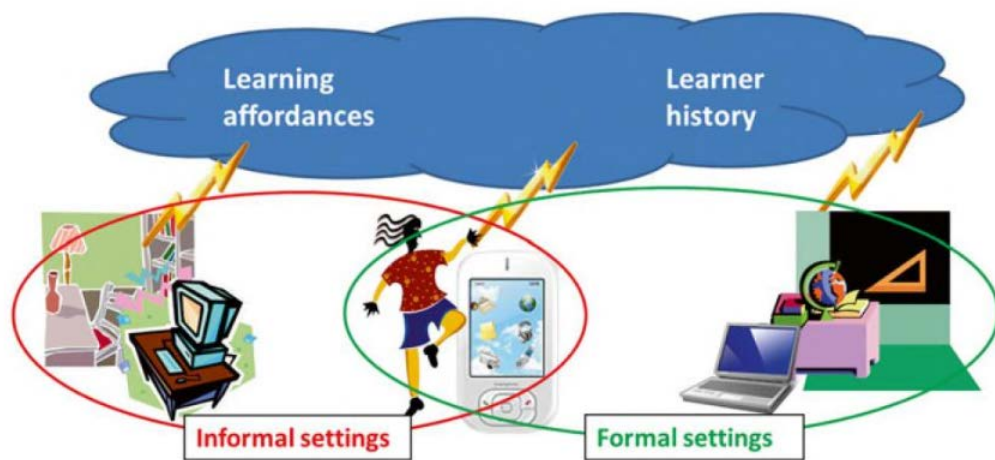


Abbildung 2.5: Learning Hub [Wong, 2014]

Als zentrale Charakteristik einer SL-Umgebung erweist sich die Möglichkeit des durchgängigen Lernens in unterschiedlichem Kontext – zeitunabhängig und standortübergreifend. Dabei werden mobile, ubiquitäre Technologien zur Lernunterstützung eingesetzt. Eine Möglichkeit die notwendige Technik zur Verfügung zu stellen, findet sich in der BYOD-Politik [siehe Kapitel 2.4] wieder. Diese stellt eine Form des 1:1 Computing dar [NSW Department of Education and Communities, 2013].

Beispielsweise führt Song [2014] eine Studie mit BYOD im Bereich der „Science Inquiry“ in einer SL-Umgebung durch. Laut seinen Schlussfolgerungen hatte BYOD in diesem Setting einen positiven Einfluss auf die Wissensentwicklung.

## 2.4 Bring your own device

*"Employees or students bring personally owned mobile devices (laptops, netbooks, tablets, smartphones, etc.) to their workplace or educational institution and use those devices to access corporate, institutional and other information, applications and services."* [Future Classroom Lab, 2015 - Seite 1]

Der starke Anstieg der Verwendung von mobilen, internetfähigen Geräten wie Smartphones [Ebner, Nagler & Schön, 2012] macht BYOD erst umsetzbar. Dabei kann dieser Ansatz in unterschiedlichen Bereichen eingesetzt werden, wie in Unternehmen, an Schulen oder an Universitäten. Ein Risiko birgt jedoch die Sicherheit, die vor allem im Unternehmensbereich als essentiell anzusehen ist [Niehaves et al., 2012]. Im Ausbildungssektor sind diese Risiken aufgrund der geringeren Sensibilität der Daten oder des öffentlichen Zugriffs vernachlässigbarer.

In einem Review zur Literatur wurden drei Gründe für BYOD im Ausbildungsbereich identifiziert [NSW Department of Education and Communities, 2013]:

- Der finanzieller Druck, da die Kosten der Anschaffung zeitgemäßer Technik meist zu hoch sind
- Der Druck von Schülerinnen und Schülern bzw. Studentinnen und Studenten und Angestellten ihre ohnehin vorhandenen privaten Geräte verwenden zu können
- Die mittlerweile hohe Verbreitung von mobilen Geräten in der Bevölkerung

Disterer & Kleiner [2013] sehen als größte Chance für BYOD den Komfort für die Benutzerinnen und die Benutzer nur ein einziges Gerät für jede Tätigkeit, an jedem Ort und zu jeder Uhrzeit zu verwenden. Die Vorteile der Kostenersparnis für die Institutionen, die "nur" mehr die Infrastruktur anbieten müssen, scheinen eine zu erwartende Folge von BYOD zu sein. Des Weiteren entfällt die Zeit für das Kennenlernen einer anderen Umgebung für die Lernenden, weil sie die personalisierten und gewohnten Geräte einsetzen [MacGibbon, 2012]. Laut dem Review des NSW Department of Education and Communities [2013] sollen bei einer Zentrierung der Lernumgebung auf den Lehrenden bei BYOD Standardgeräte verwendet werden, damit jeder Lernende die gleiche Umgebung verwendet. Bei einer Zentrierung auf den Lernenden ist weniger Standardisierung und Kontrolle notwendig.

## 2.5 Audience-Response-Systeme

In der Literatur verwenden die einzelnen Autorinnen und Autoren unterschiedliche Bezeichnung für ARS, zum Beispiel: clickers [Caldwell, 2007], classroom response systems [Fies & Marshall, 2006], electronic response systems [Judson & Sawada, 2002] oder electronic voting systems [Simpson & Oliver, 2007]. ARS bestehen aus Geräten für die Teilnehmerinnen und Teilnehmer zur Benutzereingabe, einer zentralen Stelle zur Datensammlung und Software für die Kommunikation und der visuellen Darstellung der Resultate in Echtzeit [Fies & Marshall, 2006; Haintz, Pichler & Ebner, 2014]. Caldwell [2007] beschreibt ein ARS als ein Managementtool, mit dem sich Lernende bei Massenvorlesungen beteiligen können.

### 2.5.1 Einleitung

Die ersten Einsätze von ARS fanden in den 1950er Jahren beim Militär statt [Judson & Sawada, 2002]. Judson & Sawada [2002] datieren ihre Verwendung bei Massenvorlesungen auf die 1960er Jahre. Technologisch haben sich ARS in den letzten Jahren weiterentwickelt. Laut Caldwell [2007] läuft der Datentransfer bei modernen Systemen kabellos. Dank mobiler Geräte und des Internets greifen Systeme der Gegenwart dem Stand der Technik entsprechend auf webbasierte Lösungen zurück [Pichler, 2013; Haintz, Pichler & Ebner, 2014]. Sie sind somit mit dem BYOD-Ansatz [siehe Kapitel 2.4] kompatibel, weil jedes internetfähige Gerät mit den Systemen verbunden werden kann. In Bezug auf ARS versteht sich unter BYOD, dass jede Teilnehmerin und jeder Teilnehmer sein privates mobiles Gerät (zum Beispiel Smartphone, Notebook, Tablet) mit zum Vortrag bringt und für Umfragen, Multiple-Choice-Fragen, Bewertungen etc. einsetzt.

Gegenüber traditionellen ARS, wo alle Geräte von der Veranstalterin und dem Veranstalter bereitgestellt werden, entstehen durch webbasierte ARS in Kombination mit BYOD erhebliche Vorteile. In einer Studie über mobile-based Polling werden die enorme Kostenersparnis und der verminderte Wartungsaufwand von Software und Hardware zusammengefasst [Noel, Stover & McNutt, 2015]. Bei den traditionellen Systemen musste vor einem Vortrag die Hardware - die sogenannten Clickers - an die Teilnehmerinnen und Teilnehmer ausgegeben und am Ende wieder eingesammelt werden, die Anzahl der Geräten ist begrenzt. Für die Anschaffung und Wartung der Geräte ist mit hohen Kosten zu rechnen. Ein herkömmlicher Clicker kann aufgrund seines Aufbaus mit einer fixen Anzahl an Tasten und deren



Bedeutung nur für gewisse Funktionen, wie beispielsweise Multiple-Choice-Fragen, eingesetzt werden. Internetfähige Geräte bieten weitermehr Flexibilität bezüglich des User Interfaces und der Art des Feedbacks.

Kay & LeSage [2009] evaluierten im Rahmen einer Literaturrecherche die Vorteile bei der Verwendung von ARS. Zusammengefasst haben sie festgestellt, dass der Einsatz von ARS die Anwesenheit bei Vorlesungen steigert, unter der Voraussetzung einer Verknüpfung mit der Benotung. Des Weiteren kann die Aufmerksamkeit des Publikums durch eine intervallartige Verwendung von ARS erhöht werden, weil sie dadurch immer wieder aktiv am Lernprozess teilnehmen. Ein anderer Aspekt für eine erhöhte Bereitschaft zur Mitarbeit führt auf den Schutz der Anonymität zurück. Eine Frage anonym zu beantworten oder zu stellen, nimmt die Angst vor Beurteilungen durch die Kollegen. Studentinnen und Studenten berichten von einem höheren Interesse an Konzepten, die mittels ARS vermittelt werden. Als Lernvorteile führen Kay & LeSage [2009] die erhöhte Interaktion bei Vorlesungen, die gesteigerte Qualität bei Diskussionen, das bessere Einstellen des Lehrenden auf das Auditorium dank des Feedbacks, eine gesteigerte Lernperformance sowie eine erhöhte Qualität des Lernens per se an.

Bei Massenvorlesungen können Multiple-Choice-Frage durch den Einsatz von ARS im Gegensatz zum traditionellen „Handheben“ in angemessener Zeit durchgeführt und ausgewertet werden [Kay & LeSage, 2009]. Ein ARS kann das Speichern der Rückmeldungen für spätere Vergleiche ermöglichen. Dies erweist sich vor allem bei frei formulierten Fragen und Kommentaren des Publikums als sinnvoll, die Vortragenden können in der nächsten Vorlesung komplexere Themen nochmals erläutern. Ein weiterer identifizierter Punkt sieht ARS als unterstützende Maßnahme eines effektiven "Formative Assessment". Ein "Formative Assessment" erlaubt ein notenloses Feststellen des Verständnisses der vermittelten Konzepte durch die Befragung der Teilnehmerinnen und Teilnehmer [Kay & LeSage, 2009].

### **2.5.2 Frontchannel und Backchannel**

Yardi [2006] definiert den Backchannel als zweiten Kommunikationskanal, der den Frontchannel ergänzt und unterstützt. Der Frontchannel beschreibt die persönliche Kommunikation zwischen dem Publikum und der vortragenden Person. Im Gegensatz dazu findet der Informationsaustausch beim Backchannel im Hintergrund statt. Sind bei der Verwendung eines ARS

## Definitionen

die Vortragenden unmittelbar involviert, kann das System dem Frontchannel zugeordnet werden. Zum Beispiel unterbricht die Referentin und der Referent bewusst den Vortrag, startet eine Umfrage und präsentiert anschließend die Resultate. Beim Backchannel haben die Teilnehmerinnen und Teilnehmer ohne Einbeziehung der Sprecherin und des Sprechers die Möglichkeit ständig zu interagieren.

Der Backchannel selbst lässt sich in qualitative und quantitative System untergliedern [Haintz, 2013]. Bei qualitativen System übermitteln die Teilnehmerinnen und Teilnehmer frei formulierte Fragen oder Kommentare an die Vortragenden. Die Anzahl der Rückmeldungen sollte geringer sein, da sie aufgrund ihrer Individualität von der Vortragenden und dem Vortragenden selbstständig interpretiert werden muss. Quantitative Systeme sind hingegen darauf ausgelegt eine hohe Masse von Rückmeldungen automatisiert zu verarbeiten. Die Daten werden vom System gesammelt und können anschließend präsentiert werden. Dabei handelt es sich beispielsweise um Umfragen mit fixen Antwortmöglichkeiten oder der Bewertung der Vortragenden Person bzw. des Vortrages selbst, wie die Geschwindigkeit.

### 2.5.3 Feedbackmethoden

Durch das Feedback tritt die Vortragende und der Vortragende in Interaktion mit dem Publikum. Je nach ARS sind unterschiedliche Arten des Feedbacks möglich. Dabei spielt der Zeitpunkt eine entscheidende Rolle - wird das Feedback am Ende einer Umfrage einmalig eingeblendet oder sollen die Daten permanent bzw. in Echtzeit aktualisiert werden. Nach Vetterick, Garbe & Cap [2013] existieren folgende Methoden: Peer Instruction, Chatwall und Speech Parameter. Im Sinne der Definition von Frontchannel und Backchannel zählt die Peer Instruction zum Frontchannel. Das Feedback bei der Chatwall und beim Speech Parameter läuft in Echtzeit im Hintergrund ab und kann dem Backchannel zugerechnet werden.

#### **Peer Instruction**

Bei Peer Instruction handelt es sich um Umfragen im Sinne von Multiple-Choice-Fragen. Dabei wird den Teilnehmerinnen und Teilnehmern eine Frage gestellt und sie haben die Möglichkeit eine oder mehrere Antwortmöglichkeiten auszuwählen. Die gesammelten Daten der Umfrage werden im Anschluss ausgewertet und dem Publikum präsentiert.



### **Chatwall**

Chatwall ist der umgekehrte Weg zur Peer Instruction. Die Studentinnen und Studenten können Frage und Kommentare posten. Diese sind in Form eines Chats für alle Teilnehmerinnen und Teilnehmer ersichtlich. Für bestehende Fragen können Stimmen abgegeben werden, um eine Priorität festzulegen.

### **Speech Parameter**

Das Prinzip von Speech Parameter beruht nicht auf dem Stellen und Beantworten von Fragen. Vor einem Vortrag werden gewisse Parameter wie die Geschwindigkeit oder die Verständlichkeit festgelegt. Die Teilnehmerin und der Teilnehmer kann kontinuierlich eine Bewertung durch die Parameter abgeben. Dadurch können die Vortragenden in Echtzeit auf das Publikum reagieren. Eine Situation, in der das Publikum nicht mehr folgen kann, wird frühzeitig vermieden.

#### **2.5.4 Herausforderungen beim Erhalt des Feedback**

Am Ende einer Umfrage - Peer Instruction - werden die Daten grafisch für die Teilnehmerinnen und Teilnehmer dargestellt. Dies kann beispielsweise in Form eines Balkendiagramms, eines Kreisdiagramms oder eines Blasendiagramms erfolgen [Pichler, 2013]. Die Referentin und der Referent muss nicht speziell auf Rückmeldungen achten, da der Vortrag bewusst unterbrochen wird. Nach der Beendigung der Abstimmung wird die Grafik eingeblendet.

ARS bestehen neben der Beantwortung von Multiple-Choice-Fragen noch aus anderen Komponenten. Ein Feedback könnte mittels Fragen oder Kommentaren aus dem Publikum - Chatwall - oder der direkten Bewertung des Vortrages - Speech Parameter - durchgeführt werden. Diese Kommunikation läuft über den Backchannel, wodurch die Vortragende Person nicht unmittelbar eingebunden ist. Dem Auditorium stehen die aktuellen Daten zumeist zur Verfügung, ohne dass der Vortrag unterbrochen wird. Damit es nicht zu Verzögerungen bei der Reaktion der Referentinnen und Referenten kommt, müssen diese durch Kontrollblicke selbstständig auf den Erhalt von dringenden Rückmeldungen prüfen. Die Präsentationssoftware soll jedoch weiterhin im Vordergrund bleiben. Es stellt sich die Frage nach einem passenden User Interface für die Vortragenden.

## Definitionen

Durch die Nutzung eines zweiten Gerät, welches für die Anzeige des Feedbacks verwendet wird, könnte das Problem gelöst werden. Die Konzentration auf zwei Bildschirme scheint jedoch schwierig und kann zu einer hohen visuellen Belastung führen. Die Filtertheorie der Aufmerksamkeit nach Broadbent sagt, dass zwei Eingangsreize desselben Typus schwerer verarbeitet werden können, als zwei unterschiedliche Eingangsreize [Müller & Krummennacher, 2012]. Alternativ kann die Anzeige des Feedbacks in die Präsentationssoftware integriert werden. Jerweils ergibt sich die Notwendigkeit, dass speziell auf das Vortragsgerät geachtet werden muss und sich dieses ständig im Blickfeld der Vortragenden Personen befindet [Vetterick et al., 2014]. Die Referentin und der Referent sind folglich in ihrer Mobilität eingeschränkt. Des Weiteren sind Vortragende dazu gezwungen selbstständig an die Möglichkeit einer neuen Rückmeldung zu denken und dies bewusst zu überprüfen. Dadurch kann es zu einer verspäteten Reaktion auf das Feedback kommen [Cap, Delfs & Vetterick, 2015].

Im Rahmen dieser Arbeit wird die Smartwatch als "verlängerter Arm" des ARS eingesetzt, um Vortragenden die Rückmeldungen auf das Handgelenk zu bringen und dadurch die geschilderte Problematik bei einem Echtzeit-Feedback über den Backchannel zu lösen. Eine verspätete Reaktion auf eine Rückmeldung und die Gefahr der visuellen Überladung könnten durch Vibrationssignale eingedämmt werden. Dank der Smartwatch wird die Mobilität der Vortragenden im Sinne der Position im Raum als auch beim Zugriff auf die digitalen Informationen, die durch das ARS produziert werden, erweitert.

### 3 Die Smartwatch

Die Vision eine Uhr zur Kommunikation einzusetzen liegt mindestens siebenzig Jahre zurück. Bereits 1946 nutzte Dicky Tracy in dem gleichnamigen Comicstrip seine Uhr zum Telefonieren [siehe Abbildung 3.1]. In späteren TV-Serien wie Raumschiff Enterprise oder Knight Rider konnten die Darsteller die Geräte am Handgelenk für die Kommunikation nutzen.



Abbildung 3.1: Die Uhr von Dick Tracy zur Sprachübertragung<sup>2</sup>

#### 3.1 Geschichte

Die Smartwatch des Jahres 2015 findet ihren Ursprung in der Digitaluhr, welche erstmals in den 1970er Jahren hergestellt wurde. Das erste Modell war die Pulsar P1 aus dem Jahr 1972 von der Hamilton Watch Company [Charlton, 2013]. Die Anzeige wurde zu Beginn mittels LEDs realisiert, spätere Modelle verwendeten ein LC-Display.



Abbildung 3.2: Seiko RC-1000 [Marshall, 2015]; Timex Datalink [Ellis, 2015]

<sup>2</sup> <http://www.internationalhero.co.uk/d/diktracy.htm> (besucht am 17.12.2015)

## Die Smartwatch

Mit der Seiko RC-1000 [siehe Abbildung 3.2] aus dem Jahr 1984 konnten Daten per Kabel synchronisiert werden. Dabei war sie mit verschiedenen PCs kompatibel [Marshall, 2015]. Eine kabellose Verbindung zur Übermittlung von Daten ermöglichte die Timex Datalink [siehe Abbildung 3.2], die 1994 veröffentlicht wurde. Dabei empfing sie durch einen Lichtsensor optische Informationen in Form von Strichcodes vom Bildschirm des PCs [Bieber, Haescher & Vahl, 2013; Ellis, 2015].

Im Jahr 2000 wurde die IBM Linux Watch [siehe Abbildung 3.3] vorgestellt. Auf ihr liefen dank Linux und X11 grafische Anwendungen wie zur Darstellung von Ziffernblättern oder zur Anzeige von Bildern [Narayanaswami & Raghunath, 2000]. 2004 entwickelte Microsoft die SPOT watch [siehe Abbildung 3.3]. Sie konnte Informationen wie Wetter, Verkehr etc., welche über FM Radiowellen übertragen wurden, empfangen und visualisieren [Youssef et al., 2005].



Abbildung 3.3: IBM Linux Watch<sup>3</sup>; Microsoft SPOT watch [Youssef et al., 2005]

Die Verbreitung von Smartphones stieg in den letzten Jahren stark an. Eine Studie aus den USA zeigte, dass 2013 bereits 56% der Erwachsenen ein Smartphone besitzen. Verglichen damit lag dieser Wert 2012 bei 46% und 2011 bei 35% [Smith, 2013]. Zukünftige Generationen von Smartwatches machen sich diesen Trend zu Nutze, indem sie sich mit dem ohnehin vorhandenen Smartphone über Bluetooth verbinden und somit Benachrichtigungen vom Smartphone anzeigen können.

---

<sup>3</sup> [http://researcher.watson.ibm.com/researcher/view\\_group\\_subpage.php?id=6102](http://researcher.watson.ibm.com/researcher/view_group_subpage.php?id=6102)  
(besucht am 28.12.2015)

Als eine dieser modernen Smartwatches erweist sich die Pebble, die sowohl mit diversen Android-Geräten als auch mit diversen iPhones kompatibel ist. Finanziert wurde das Projekt über die Crowdfunding-Plattform kickstarter.com im Jahr 2012. Dabei erwies sich die Pebble als eine der erfolgreichsten Crowdfunding-Kampagnen der Geschichte<sup>4</sup>. Damit lenkte sie die Aufmerksamkeit der Öffentlichkeit noch intensiver auf die Smartwatch. Neben der Pebble kamen im Jahr 2013 einige weitere Modelle auf den Markt - unter anderem die Qualcomm Toq, die Sony Smartwatch 2 oder die Samsung Galaxy Gear. Dabei fehlte es an einem herstellerübergreifenden Betriebssystem, die Kompatibilität mit dem Smartphone variierte je Hersteller bzw. Gerät und die Portabilität von Anwendungen war eingeschränkt.

Unter diesen Voraussetzungen erschien im Juni 2014 das Betriebssystem Android Wear von Google [Pelegriin, 2014]. Wie beim Smartphone sollen diverse Hersteller für dieses OS gewonnen werden und somit für Einheitlichkeit bei der Software der Smartwatch sorgen. Bis Ende des Jahres 2014 erschienen sechs Smartwatches mit Android Wear bzw. wurden angekündigt - die LG G Watch, die Samsung Gear Live, die LG G Watch R, die Sony SmartWatch 3, die Moto 360 und die Asus ZenWatch.

Samsung brachte 2014 mit der Gear S eine Smartwatch auf den Markt, die über einen Steckplatz für eine SIM-Karte verfügt. Damit kann sie eigenständig eine Verbindung zum Mobilfunknetz (3G) aufbauen und zum Telefonieren verwendet werden, was das Smartphone überflüssig macht. Als Betriebssystem kommt Tizen OS von Samsung zum Einsatz [Stein, 2014]. Als bisher letzter Meilenstein gilt der Einstieg von Apple in den Smartwatch-Markt, der Verkauf der Apple Watch mit watchOS startete im April 2015<sup>5</sup>. Weitere Modelle der Pebble, von Samsung oder Geräten mit Android Wear folgten. Die Smartwatch scheint definitiv das Potential zu haben sich als neues Wearable zu etablieren.

---

<sup>4</sup> <https://www.kickstarter.com/projects/597507018/pebble-e-paper-watch-for-iphone-and-android> (besucht am 25.12.2015)

<sup>5</sup> <http://www.mactechnews.de/news/article/Apple-Watch-ab-jetzt-vorbestellbar-161147.html> (besucht am 29.12.2015)

### 3.2 Definition

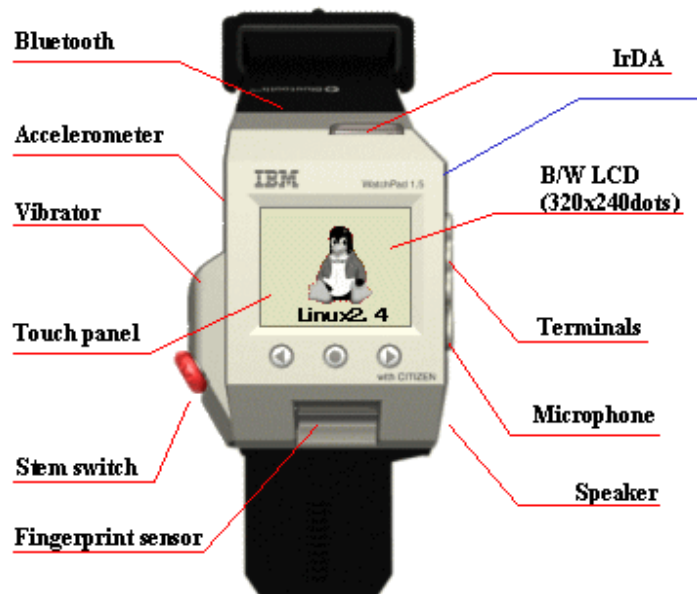
Bei der Smartwatch ereigneten sich in den letzten Jahren aufgrund der Anbindung an das Smartphone und des technologischen Fortschritts erhebliche Entwicklungssprünge. Dementsprechend mussten ältere Definitionen angepasst werden, neuere Beschreibungen der Smartwatch lauten:

*"Smart watches are connected to the internet and provide sensor functionality that allows an enhanced human-computer-interaction."* [Bieber, Haescher & Vahl, 2013 - Seite 1]

*"a wrist-worn device with computational power, that can connect to other devices via short range wireless connectivity; provides alert notifications; collects personal data through a range of sensors and stores them; and has an integrated clock"* [Cecchinato, Cox & Bird, 2015 - Seite 2134]

Anhand der Charakteristiken von Wearables nach Rhodes und der Bedeutung von Context Awareness werden gewisse Anforderungen an eine Smartwatch gestellt. Um diese zu erfüllen, scheint die Notwendigkeit eines Betriebssystems unabdingbar. Einerseits wird durch ein Betriebssystem mit der Verwaltung der Anwendungen ein ständiger Betrieb ermöglicht. Andererseits dient es der Steuerung der Hardware, wie die Sensoren.

Um Context Awareness zu gewährleisten sind eine Vielzahl von Sensoren in einer Smartwatch verbaut, unter anderem: Umgebungslichtsensor, Beschleunigungssensor, Kompass, Gyroskop, Pulsmesser oder GPS. Kontextbezogen zu agieren und die Fähigkeit der kabellosen Verbindung mit anderen Geräten bzw. dem Internet über Bluetooth oder Wifi zur Informationsübertragung erlauben es der Smartwatch proaktiv zu handeln. Um eine freihändige Bedienung zu ermöglichen sind unter anderem Mikrofone zur Spracheingabe eingebaut. Ebenso minimieren der Touchscreen und grafische Anwendungen wie Wähldialoge den notwendigen Einsatz der Hände. Das IBM Watchpad [siehe Abbildung 3.4] aus dem Jahr 2001 konnte diese Anforderungen bereits erfüllen und folglich als erste Smartwatch angesehen werden [Cecchinato, Cox & Bird, 2015].

Abbildung 3.4 IBM Watchpad<sup>6</sup>

Im aktuellen Wortgebrauch versteht sich eine Smartwatch als Gerät, welches mit dem Smartphone verbunden ist. Der Großteil der derzeit am Markt befindlichen Smartwatches erweitert die Funktionalität bzw. den Bildschirm eines Smartphones. Sie können dementsprechend als verlängerter Arm des Smartphones angesehen werden. Dabei dienen sie grundsätzlich zur Anzeige von Benachrichtigungen vom Smartphone, aber auch zur Steuerung von Anwendungen des Smartphones sowie zur Ausführung von eigenständigen Programmen. Die Samsung Gear S Reihe gilt hingegen als komplett autarke Smartwatch, mit der dank eigenen Slots für eine SIM-Karte telefoniert werden kann.

### 3.3 Wahl der Smartwatch

Für die Projektumsetzung soll zunächst eine passende Smartwatch ausgewählt werden. Die Kaufentscheidung für ein gewisses Modell fiel im Januar 2015, weshalb für die Auswahl der damalige Stand von Smartwatches herangezogen wurde. Hier werden in erster Instanz die unterschiedlichen Umgebungen gegenübergestellt. Anschließend wurden in zweiter Instanz

<sup>6</sup> [http://researcher.watson.ibm.com/researcher/view\\_group\\_subpage.php?id=6104](http://researcher.watson.ibm.com/researcher/view_group_subpage.php?id=6104)  
(besucht am 29.12.2015)



die konkreten Modelle verglichen und eine Entscheidung getroffen. Hierbei spielten neben den technischen Möglichkeiten der Smartwatch auch anwendungsspezifische Kriterien eine entscheidende Rolle.

### 3.3.1 Entscheidung nach Umgebung

Für die Entscheidung nach einer Umgebung wurden diverse Kriterien aufgestellt. Diese waren das Betriebssystem, die Kompatibilität zu Smartphones, der Support für die Entwickler sowie die Möglichkeiten der Gestaltung des User Interfaces und der Interaktion mit den Anwendern.

Das installierte Betriebssystem der Smartwatch soll speziell auf Wearables ausgerichtet sein. Bei einem solchen OS kann davon ausgegangen werden, dass das Potential für Weiterentwicklungen in Zukunft am größten ist. Damit schieden alle Uhren mit einem proprietären OS aus. Die Apple Watch mit watchOS war im Jänner 2015 noch nicht auf den Markt. Damit kamen mit Tizen OS, Pebble OS und Android Wear OS nur drei Betriebssysteme in die engere Auswahl.

In Punkto Kompatibilität sollte die Smartwatch mit möglichst vielen Smartphones gekoppelt werden können. Mit Tizen OS von Samsung waren nur die Gear 2 und die autonome Gear S auf dem Markt. Die Tatsache, dass sie nur mit Smartphones von Samsung kompatibel sind, lies dieses Betriebssystem ausscheiden. Bei Android Wear ist die Kompatibilität der Smartwatches zu Smartphones auf Geräte mit mindestens Android OS 4.3 eingeschränkt - mittlerweile auch iOS 8.2 und höher<sup>7</sup>. Die Kompatibilität der Pebble ist für Android-Geräte mit OS 4.1 oder höher sowie mit iPhones mit iOS 7 oder höher optimiert<sup>8</sup>.

Bezüglich des Supports bei der Entwicklung von Anwendungen war dieser sowohl für Android Wear<sup>9</sup> als auch für Pebble OS<sup>10</sup> gegeben. Dabei standen sowohl eine Entwicklungsumgebung und zugehörige Tools als auch Beispielanwendungen zur Verfügung.

---

<sup>7</sup> <https://www.android.com/wear/check> (besucht am 29.12.2015)

<sup>8</sup> <https://help.getpebble.com/customer/de/portal/articles/1722510-kompatibilit%C3%A4t> (besucht am 29.12.2015)

<sup>9</sup> <http://developer.android.com/wear/index.html> (besucht am 29.12.2015)

<sup>10</sup> <https://developer.getpebble.com> (besucht am 29.12.2015)



Als letztes Kriterium wurden das User Interface und die Interaktionsmöglichkeiten herangezogen. Die zu implementierende Anwendung benötigt unbedingt Vibrationssignale, was sowohl mit Android-Wear-Uhren als auch mit der Pebble realisiert werden kann. Als ein Vorteil bei den Android-Wear-Uhren stellte sich das Display heraus. Eine leuchtintensivere Anzeige, höhere Bildauflösung und die Farbdarstellung können das visuelle Signal deutlich verstärken und machen das User Interface variabel gestaltbar. Dank des ePaper-Display weist die Pebble zwar eine viel längere Laufzeit von fünf bis zu sieben Tagen gegenüber den Android-Wear-Uhren (ein bis zwei Tagen) auf, was für die Anwendung aber nicht entscheidend war. Die Interaktion mit der Anwenderin und dem Anwender wird bei Android-Wear-Uhren dank des Touchscreens sehr erleichtert.

Somit fiel schlussendlich die Entscheidung zugunsten einer Smartwatch mit Android Wear. In einem nächsten Schritt wurden die verfügbaren Modelle gegenübergestellt.

### 3.3.2 Entscheidung nach Modell

Nach der Entscheidung für eine Android-Wear-Uhr wurden die im Jänner 2015 verfügbaren Modelle verglichen. Für die Erstellung eines Vergleiches dienten die Produktbeschreibungen der Hersteller und diverse Testberichte. Folgenden Smartwatches kamen in Frage - für eine detaillierte Aufstellung siehe [Anhang](#):

- LG G Watch
- Samsung Gear Live
- LG G Watch R
- Sony SmartWatch 3
- Moto 360 (1. Gen.)
- Asus ZenWatch

Für das schlussendliche Modell wurden nachfolgende Kriterien festgelegt:

Displayform: Zur Auswahl standen entweder eine runde oder eine eckige Form. Da in der Anwendung angedacht war, den Bildschirm zu teilen, schien ein eckiges Display besser geeignet zu sein.

Akku: Von den Herstellern wurden unterschiedliche Angaben zur Akkulaufzeit gegeben. Dabei schwankten die Angaben für den normalen

## Die Smartwatch

Betrieb zwischen ein bis zwei Tagen. Da der tatsächliche Energiebedarf der Anwendung noch nicht abzusehen war, sollte die Akkukapazität möglichst groß gewählt werden.

Reichweite: WLAN Verbindungen wurden zum Zeitpunkt des Vergleichs von Android Wear noch nicht unterstützt. Deshalb war unklar welche Smartwatch dieses Feature in Zukunft unterstützen werden. Teilweise gaben Hersteller aber vorab bekannt, ob ein solches Modul verbaut wurde. Dies würde ein mögliches Problem bezüglich der Reichweite zwischen Smartphone und Smartwatch im Vergleich zur Bluetooth-Verbindung unerheblich werden lassen.

Sonstiges: Die verbauten Sensoren spielten nicht unmittelbar für die Anwendung eine Rolle. Für zukünftige Projekte könnten sie aber von Nutzen sein. Deshalb waren andere verbaute Module wie Schrittzähler, GPS etc. für den Vergleich heranzuziehen.

Schließlich fiel die Entscheidung aufgrund der oben angeführten Kriterien zugunsten der Sony SmartWatch 3. Zu diesem Zeitpunkt schien sie die beste Wahl, da sie über ein rechteckiges Display, den größten Akku und als einziges Modell über ein GPS Modul verfügte. Ebenso war ein WLAN-Modul integriert, das in Zukunft vermeintlich unterstützt wird und sie lag preislich im Mittelfeld. Lediglich ein Abstrich beim Pulsmesser musste in Kauf genommen werden.

## 4 Stand der Technik

Im Zuge der Recherche für die Umsetzung der Anwendung konnte ein ARS identifiziert werden, für welches bereits ein Versuch mit der Smartwatch durchgeführt wurde. Dabei handelt es sich um das System Tweedback<sup>11</sup>, welches ein Live-Feedbacksystem der Universität Rostock ist.

Tweedback besteht aus drei Komponenten: Quiz, Chatwall und einem Panik-Button. Von diesen Bausteinen wählt man bei der Erstellung des Vortrages je nach Einsatzbereich aus [siehe Abbildung 4.1]. Auf der rechten Seite des Bildschirms können nachträglich Komponenten eingeschalten bzw. wieder ausgeschalten werden.

### Wähle Deine Features

Es sind keine Features aktiv.  
Gewünschte Features können hier ausgewählt werden. Mit dem "Lass mich loslegen" Button werden die ausgewählten Features aktiviert.  
Später können in der Navigationsleiste Features aktiviert oder deaktiviert werden.



#### Panik-Button

Erlaubt es den Studenten, während der Vorlesung dem Dozenten Rückmeldung zu geben, dass etwas nicht stimmt.



#### Quiz

Ermöglicht es dem Dozenten eine Multiple-Choice-Frage an das Auditorium zu richten. Anschließend können die Studenten sich für eine der Antwortmöglichkeiten entscheiden und abstimmen.



#### Chatwall

Gestattet es den Studenten während der Veranstaltung Fragen an den Dozenten zu richten. Diese Fragen erreichen den Dozenten erst, wenn genügend Kommilitonen die Frage als wichtig erachten.

Lass mich loslegen!

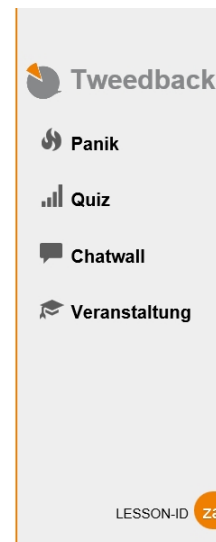


Abbildung 4.1: Erstellung einer Veranstaltung bei Tweedback<sup>11</sup>

Mit der Verwendung der Smartwatch wurde versucht die Probleme beim User Interface des Vortragenden zu lösen [siehe Kapitel 2.5.4]. Umgesetzt wurde die Anwendung unter Android Wear. Dabei kommuniziert Tweedback mit der Smartwatch über das Smartphone oder das Tablet, das der Vortragende bei sich trägt. Im Falle einer Rückmeldung wird eine Benachrichtigung erzeugt – Android Wear stellt sie als Card dar. Diese Card enthält nur die Information von welcher Komponente des Tweedbacks die Rückmeldung kommt. Für weitere Details kann sich die Benutzerin und der

<sup>11</sup> <http://www.tweedback.de> (besucht am 16.12.2015)

Benutzer den gesamten Inhalt der Card anzeigen lassen. In Abbildung 4.2 ist der Erhalt einer Frage aus der Chatwall Komponente auf der Smartwatch dargestellt. Die Autoren haben den Prototypen auf der LG G Watch, der LG G Watch R und der Moto 360 getestet [Cap, Delfs & Vetterick, 2015].



Abbildung 4.2: Tweedback Chatwall Eintrag auf der Smartwatch [Cap, Delfs & Vetterick, 2015]

Nachfolgende potentielle Vorteile der Smartwatch bei ARS wurden von den Autoren identifiziert [Cap, Delfs & Vetterick, 2015]:

- Die Vortragenden tragen das Gerät ständig bei sich, somit benötigt es keine aktive Aufmerksamkeit.
- Fühlbarer Vibrationskanal bei Benachrichtigungen.
- Mit einer einfachen Handbewegung können sich die Vortragenden direkt am aktuellen Standort die Rückmeldungen ansehen.
- Die Antwort auf Fragen von der Chatwall kann durch Spracherkennung in Text konvertiert und hochgeladen werden.
- Die Smartwatch bietet eine einfache visuelle Oberfläche für die Vortragenden.

Aus der Beschreibung des Projektes geht nicht hervor, ob für die unterschiedlichen Komponenten von Tweedback auch verschiedene Vibrationssignale eingesetzt werden/einstellbar sind. Ebenso ist unklar, ob eine beantwortete Fragen von der Chatwall oder eine erfolgte Reaktion auf den Panik-Button mittels der Smartwatch als erledigt markiert werden kann. Bei der Integration der Smartwatch in den Backchannel sollen für die Parameter des Vortrages [siehe Kapitel 5.1.2] jeweils eigene Vibrationssignale gesetzt werden können und erhaltene Rückmeldung durch die vortragende Person mittels der Smartwatch bestätigbar sein.

## 5 Projektumgebung

In diesem Kapitel wird auf die Projektumgebung eingegangen. Zunächst wird der Backchannel [siehe Kapitel 5.1] und dessen Funktion vorgestellt. Mit der Unterstützung von Abbildungen wird dessen Verwendung für die Vortragenden und das Auditorium erklärt. Danach folgt eine Auflistung der technischen Details der Sony SmartWatch 3 [siehe Kapitel 5.2]. Abgeschlossen wird das Kapitel mit Android Wear [siehe Kapitel 5.3].

### 5.1 Backchannel

Für die Umsetzung des Projektes bediente man sich als ARS dem Backchannel. Dieser wurde zunächst im Rahmen einer Masterarbeit an der TU Graz realisiert [Haintz, 2013] und in Folge von der Firma Carrot & Company GmbH<sup>12</sup> weiterentwickelt.

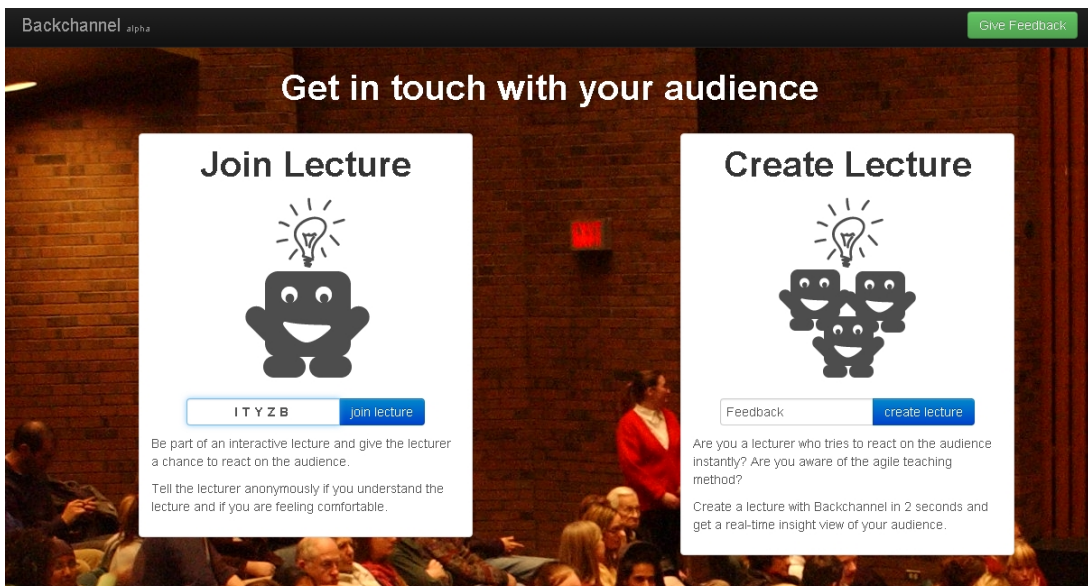


Abbildung 5.1: Startschirm des Backchannels<sup>13</sup>

Bei diesem System handelt es sich um einen quantitativen Backchannel [siehe Kapitel 2.5.2], da die Bewertung im Hintergrund abläuft und fixe Dimensionen angegeben sind. Im Sinne der Einteilung der Methoden für ein

<sup>12</sup> <http://www.cnc.io> (besucht am 08.12.2015)

<sup>13</sup> <http://www.backchannel.cnc.io> (besucht am 08.12.2015)

Feedback [siehe Kapitel 2.5.3] kann das System dem Speech Parameter zugeordnet werden. Der Backchannel ist ein webbasiertes ARS und daher mit der BYOD-Politik kompatibel. Das Publikum hat permanent die Möglichkeit den Vortrag zu bewerten oder seine abgegebenen Werte zu aktualisieren. Dabei steht der Stand der eingetroffenen Bewertungen in Echtzeit zur Verfügung, wodurch ein Vortragender sofort auf das Feedback reagieren kann. Die Benutzerin und der Benutzer kann das System ohne der Überwindung von Barrieren, wie eine Registrierung oder eine Anmeldung, verwenden. Auf dem Startschirm [siehe Abbildung 5.1] findet man sofort die zwei Optionen in welcher Rolle man den Backchannel nutzen möchte. Auf der rechten Seite kann ein Vortrag erstellt und auf der linken Seite an einem Vortrag teilgenommen werden.

### 5.1.1 Erstellung eines Vortrages

Die Vortragende und der Vortragende wählt einen Titel für den Vortrag und erstellt diesen über den zugehörigen Button. Das System navigiert auf eine andere Seite, die in zwei Spalten unterteilt ist [siehe Abbildung 5.2].

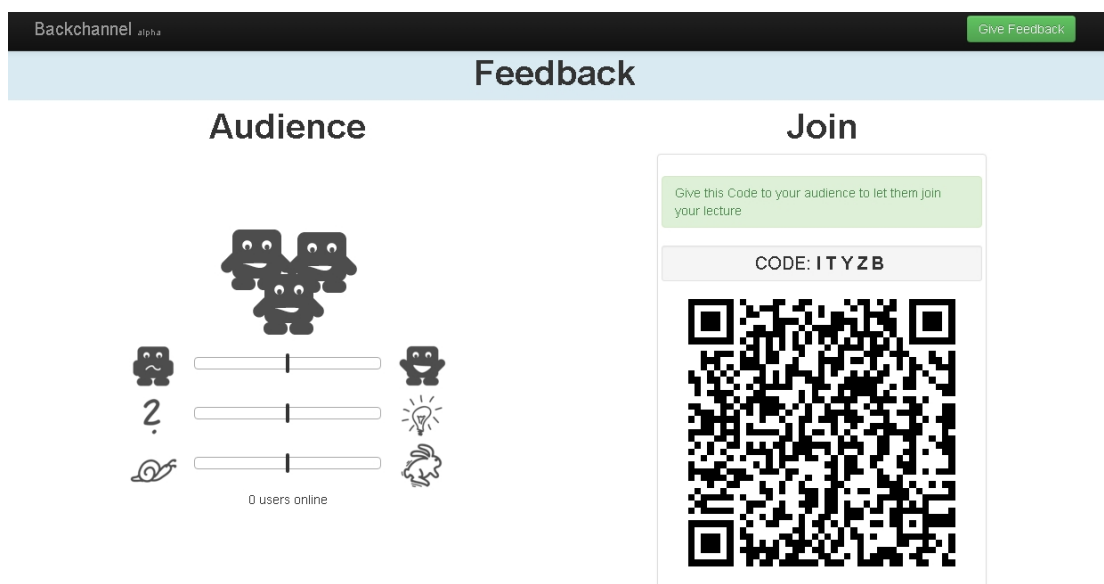


Abbildung 5.2: Schirm für die Vortragenden<sup>14</sup>

Im linken Teil sieht die vortragende Person den kalkulierten Gesamtwert. Im rechten Teil wird ein vom System erstellter und eindeutiger Code zum

<sup>14</sup> <http://www.backchannel.cnc.io> (besucht am 08.12.2015)

Vortrag - in weiterer Folge Lecture ID genannt - angezeigt. Mit der Lecture ID können die Benutzerin und der Benutzer dem Vortrag beitreten.

### 5.1.2 Teilnehmen und Bewerten eines Vortrages

Das Publikum verwendet die bei der Erstellung erhaltene Lecture ID um dem Vortrag beizutreten. Der Bewertungsschirm [siehe Abbildung 5.3] teilt sich ebenfalls in zwei Bereiche auf. Im linken Bereich hat die Teilnehmerin und der Teilnehmer seine individuellen Schieberegler um den Vortrag zu bewerten. Im rechten Bereich wird wie für die Referenten ein kalkulierter Wert, der sich aufgrund aller abgegebenen Bewertungen ergibt, grafisch dargestellt. Für die Berechnung der Gesamtwerte ist ein Aging-Mechanismus eingebaut - neuere Bewertungen sind höher gewichtet [Haintz, 2013].

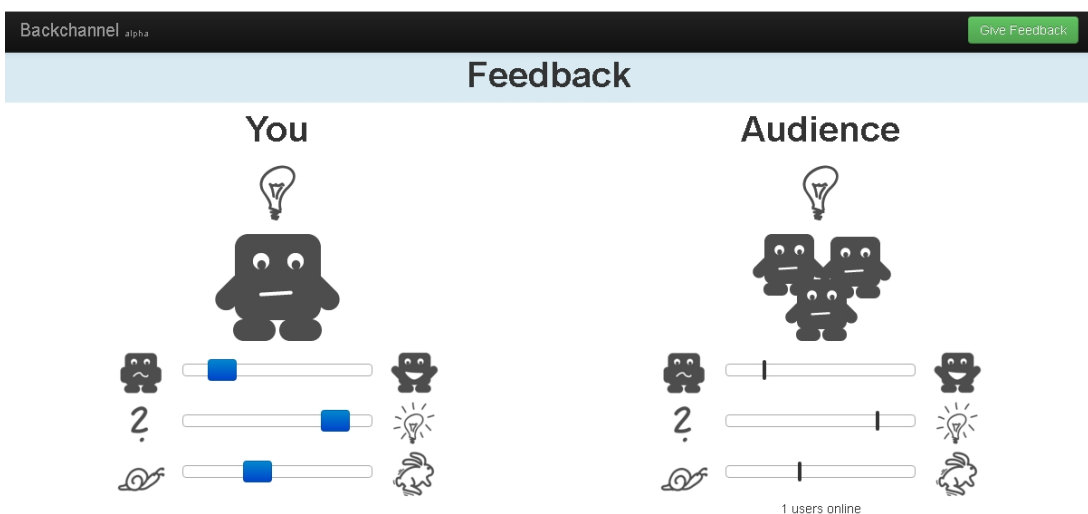


Abbildung 5.3: Schirm für die Zuhörerschaft<sup>15</sup>

Für die Bewertung stehen drei Dimensionen zur Verfügung:

- Zufriedenheit
- Verständnis
- Geschwindigkeit des Vortrages

Der Wert eines einzelnen Votes hat eine Bandbreite von -100 bis +100. Bewegt man den Schieberegler ganz nach links ist der Wert -100, in der Mitte 0 und ganz rechts +100. Oberhalb der Schieberegler befindet sich ein Avatar, der sich entsprechend der Werte der Dimensionen verändert [Haintz, 2013].

<sup>15</sup> <http://www.backchannel.cnc.io> (besucht am 08.12.2015)



## 5.2 Sony SmartWatch 3

Die Sony SmartWatch wurde bereits in der dritten Generation ausgeliefert. Im Gegensatz zu ihren beiden Vorgängern läuft sie mit Android Wear OS. Abbildung 5.4 zeigt die Smartwatch in der Ansicht von vorne und von hinten. Die technischen Kerndaten sind wie folgt [siehe [Anhang](#) für weitere Details]:

- Prozessor: 1.2 GHz, Quad-core ARM Cortex A7
- Display: eckig, transflektives TFT LCD, 1,6 Zoll, 320x320 Pixel
- Arbeitsspeicher: 512 MB
- interner Speicher: 4 GB eMMC
- Anschlüsse: Bluetooth 4.0, NFC, Micro USB, WLAN
- Sensoren: Umgebungslichtsensoren, Beschleunigungssensor, Kompass, Gyroskop, GPS
- Mikrofon
- Vibrationssignal



Abbildung 5.4: Sony Smartwatch 3<sup>16</sup>

Die Eingabe erfolgt entweder durch Sprachbefehle, durch Berührungen des Touchscreens oder per Gesten durch Handgelenksbewegungen. Die Sony Smartwatch 3 verfügt auf der rechten Seite über eine Ein-/Aus-/Aufwach-Taste.

---

<sup>16</sup> <http://www.sonymobile.com/de/products/smartwear/smartwatch-3-swr50/> (besucht am 27.12.2015)



## 5.3 Android Wear OS

Die erste Android-Wear-Version wurde im Juni 2014 als Version 4.4W (API 20) veröffentlicht. Sie gilt als leichtgewichtiger Version von Android, die speziell für den Bedarf einer Smartwatch und anderen Wearables modifiziert wurde. Dabei steht unter anderem der Strombedarf - die Kapazität des Akkus einer Smartwatch ist beschränkt und die Geräte halten im Normalbetrieb zwischen ein und zwei Tagen - und das User Interface im Fokus. Die aktuelle Version von Android Wear ist 5.1.1 (API 22).

Der Entwicklerin und dem Entwickler bietet Android Wear einige Möglichkeiten, wie Nachrichten oder Daten mit dem Smartphone auszutauschen<sup>17</sup>. Ein anderes Feature ist die Voice Activation<sup>18</sup>, mit der Anwendungen per Spracheingabe gestartet werden. Dank der großen Menge von verbauten Sensoren können Anwendung context-aware [siehe Kapitel 2.2.1] entwickelt werden<sup>19</sup>. Beispielsweise ist dank eingebautem GPS Sensor der Smartwatch der aktuelle Standort bewusst, welche Anwendungen nutzen können<sup>20</sup>.

### 5.3.1 Verbindung mit dem Smartphone

Grundsätzlich erfolgt die Kommunikation zwischen Smartphone und Smartwatch über Bluetooth. Beim Einrichten der Smartwatch wird diese mit einem Smartphone gekoppelt. Dabei muss am Smartphone die Android Wear Companion-App installiert sein. Diese App bietet wichtige Steuerungsoptionen bezüglich der Smartwatch, wie beispielsweise das Watchface oder das Ignorieren von Benachrichtigungen. Seit Android Wear Version 5.1 können die zwei Geräte auch über das WLAN miteinander kommunizieren, wobei sie sich nicht im gleichen Netzwerk befinden müssen [Brodersen, 2015]. Die sogenannte Cloudsynchronisierung kann ebenfalls über die Companion App aktiviert werden.

---

<sup>17</sup> <http://developer.android.com/training/building-wearables.html> (besucht am 08.12.2015)

<sup>18</sup> <http://developer.android.com/training/wearables/apps/voice.html> (besucht am 18.12.2015)

<sup>19</sup> <http://developer.android.com/design/wear/context.html> (besucht am 18.12.2015)

<sup>20</sup> <http://developer.android.com/training/articles/wear-location-detection.html> (besucht am 18.12.2015)

### 5.3.2 User Interface

Die Smartwatch als neuer Formfaktor benötigt aufgrund anderer Stärken, Schwächen und Anwendungsfälle ein entsprechendes User Interface. Der Entwickler Guide für Android Wear beschreibt eine genaue Vision, um der Benutzerin und dem Benutzer ein optimales Erlebnis zu garantieren. Die Vision:

*"Android Wear devices provide just the right information at just the right time, allowing users to be more connected to both the virtual world and the real world."*<sup>21</sup>

Die zwei Kernfunktionen des User Interface werden als Suggest und Demand<sup>22</sup> bezeichnet und differenzieren die zwei möglichen Auslöser der Mensch-Maschine-Interaktion. Entweder startet sie die Smartwatch per Vorschlag (Suggest) oder die Benutzerin und der Benutzer per Anfrage (demand).

#### Suggest: The Context Stream

Der Context Stream [siehe Abbildung 5.5] besteht aus einer vertikalen Liste von sogenannten Cards. Die oberste Card sehen die Benutzerinnen und Benutzer zu einem Teil direkt am Homescreen. Durch vertikales Wischen kann zwischen den Cards navigiert werden. Besteht eine Benachrichtigung aus mehreren Cards, werden durch horizontales Wischen die zusätzlichen Seiten angezeigt. Cards können Action Buttons beinhalten, durch Tippen dieser wird beispielsweise eine Anwendung im Vollbildmodus geöffnet<sup>23</sup>.



Abbildung 5.5: Beispiel für Suggest, aufgenommen im Emulator

<sup>21</sup> <http://developer.android.com/design/wear/creative-vision.html> (besucht am 18.12.2015)

<sup>22</sup> <http://developer.android.com/design/wear/index.html> (besucht am 11.12.2015)

<sup>23</sup> <http://developer.android.com/design/wear/patterns.html> (besucht am 18.12.2015)

### Demand: The Cue Card

Die Cue Card [siehe Abbildung 5.6] wird mit dem Sprachbefehl "OK Google" oder händisch geöffnet. Durch Wischen von oben bzw. nach unten wird durch die Cue Card navigiert. Sie enthält eine Liste von vordefinierten Sprachbefehlen, wobei die Funktionen ebenso per Hand auslösbar sind.

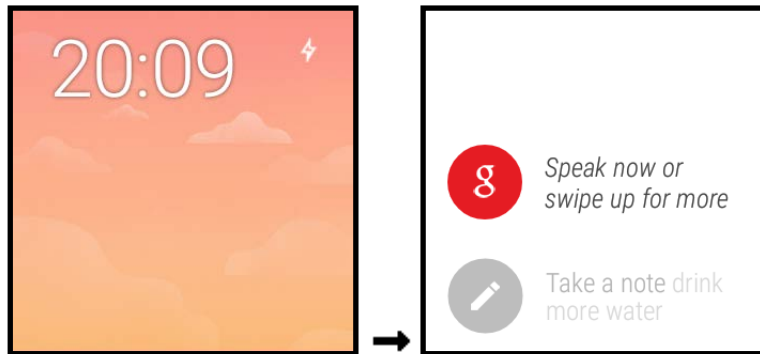


Abbildung 5.6: Beispiel für Demand, aufgenommen im Emulator

Anwendungen können auf zwei Wege gestartet werden. Entweder gelangt man vom Startbildschirm aus über horizontales Wischen zur Liste der installierten Anwendungen [siehe Abbildung 5.7] oder über die Cue Card durch das Sprachkommando "Starte" und dem in der Manifestdatei<sup>24</sup> festgelegten Text.

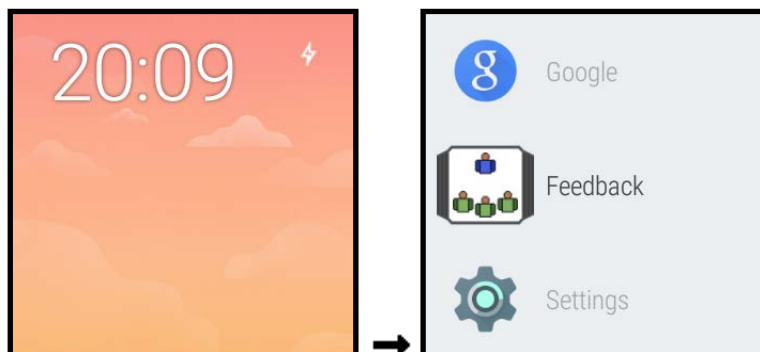


Abbildung 5.7: Anwendung starten, aufgenommen im Emulator

<sup>24</sup> Für jede Android Anwendung muss die Datei AndroidManifest.xml existieren. In ihr sind unter anderem die minimale Android-Version, ihre Komponenten oder die benötigten Berechtigungen festgelegt.

<http://developer.android.com/guide/topics/manifest/manifest-intro.html> (besucht am 18.12.2015)

### 5.3.3 Design Grundsätze

Für die Entwicklung von Anwendungen für Android Wear wurden Design-Grundsätze <sup>25</sup> als unterstützende Maßnahme herausgegeben. Folgend werden sie sinngemäß wiedergegeben:

"Focus on not stopping the user and all else will follow"

Der Benutzerin und dem Benutzer soll um die Anwendung verwenden zu können nicht seine derzeitige Tätigkeit unterbrechen müssen.

"Design for big gestures"

Wenn mit der Anwendung interagiert wird, darf keine präzise Bedienung (zum Beispiel Tippen, Wischen) verlangt sein. Aufgrund des kleinen Bildschirms führt beispielsweise das Auswählen einen Listenelementes aus einer langen Liste zu einem sehr kleinen Bereich, welcher ohne Konzentration kaum getroffen werden kann.

"Think about stream cards first"

Anwendungen sollen context-aware [siehe Kapitel 2.2.1] entwickelt sein. Nach dem Suggest-Prinzip sollen aufgrund erhaltener Daten durch die Sensoren oder eines Ereignisses Cards mit entsprechendem Inhalt eingeblendet werden.

"Do one thing, really fast"

Die Informationen von einer Card bzw. Anwendung sollen minimal und präzise sein.

"Design for the corner of the eye"

Für die Benutzerin und dem Benutzer muss es möglich sein mit nur einem Blick die Informationen der Anwendung aufzunehmen und interpretieren zu können.

"Don't be a constant shoulder tapper"

Die Smartwatch soll nicht permanent die Trägerin und den Träger durch Vibrationssignale auf sich aufmerksam machen.

---

<sup>25</sup> <http://developer.android.com/design/wear/principles.html> (besucht am 11.12.2015)

Diese Grundsätze spiegeln zum Teil auch die Grundgedanken für eine erfolgreiche Umsetzung des Projektes wieder: Das Lesen eines Feedbacks vom Backchannel soll nicht dazu führen, dass der Vortrag unterbrochen werden muss. Ebenso wenig sollen permanent Rückmeldungen gegeben werden. Sie sollen nur minimale Informationen enthalten, damit sie mit einem Blick interpretierbar sind. Die Bedienung der Anwendung muss während eines Vortrages möglich sein.



## 6 Implementierung

Für die Implementierung wurde eine passende Entwicklungsumgebung [siehe Kapitel 6.1] ausgewählt. Für die Kommunikation zwischen der Smartwatch und dem Smartphone bietet die Google-Play-Services-Bibliothek die perfekte Schnittstelle. Für die WebSocket-Verbindung zwischen dem Smartphone und dem Backchannel wird der Socket.IO Java Client von Gottox eingesetzt. Anschließend wird auf das grundlegende Design der Anwendung eingegangen [siehe Kapitel 6.2]. Danach folgen Details zur Implementierung für das Smartphone [siehe Kapitel 6.3] und der Smartwatch [siehe Kapitel 6.4], zu den Use Cases [siehe Kapitel 6.5] und zum Fehlerhandling [siehe Kapitel 6.6].

### 6.1 Entwicklungsumgebung

Als Entwicklungsumgebung wurde Android Studio<sup>26</sup> gewählt - die offizielle Android IDE. Sie basiert auf IntelliJ IDEA und wurde speziell für die Implementierung von Applikationen von Geräten, die mit dem Android OS laufen, entwickelt. Damit ist Android Studio die empfohlene Umgebung zur Durchführung eines Android-Projektes. Diese IDE ermöglicht es ein Grundgerüst von Projekten für Smartphones, Smartwatches und anderen Wearables zu erstellen. Mit Hilfe von Gradle werden externe Bibliotheken eingebunden. Durch Emulatoren können die entwickelten Anwendungen unter den unterschiedlichsten Umgebungen/Bedingungen (zum Beispiel OS Version, Gerätetyp, Bildschirmgröße etc.) getestet werden. Ebenso kann die entwickelte Anwendung direkt auf physikalische Geräte installiert werden. In der Online-Dokumentation zur Anwendungsentwicklung für Android findet sich ein eigenes Kapitel über die Erstellung von Apps für Wearables<sup>27</sup>. Für die Umsetzung des Projektes wurde folgende Umgebung eingesetzt:

- Android 5.1.1 (API 22) SDK
- Google Play Services<sup>28</sup>
- Wearable UI support library
- Socket.IO Java Client von Gottox<sup>29</sup>

---

<sup>26</sup> <http://developer.android.com/sdk/index.html> (besucht am 08.12.2015)

<sup>27</sup> <http://developer.android.com/training/wearables/apps/index.html> (besucht am 08.12.2015)

<sup>28</sup> <https://developers.google.com/android/guides/overview> (besucht am 08.12.2015)

## Implementierung

Zu Beginn der Implementierung war die Cloudsynchronisierung noch nicht verfügbar - Android-Wear-Version 4.4W (API 20) - und wurde daher nicht berücksichtigt. Für die Verwendung der Anwendung sollte dieses Feature ausgeschaltet sein - die Kommunikation zwischen dem Smartphone und der Smartwatch läuft ausschließlich über Bluetooth.

### GooglePlayServices

Um eine Verbindung mit den Google-Play-Diensten aufzubauen, verwendet man die Google-Play-Services-Bibliothek, konkret den `GoogleApiClient`<sup>30</sup>. Der `GoogleApiClient` ermöglicht die Integration von Google Play Services in die Anwendung. Dabei können ihm unterschiedliche APIs hinzugefügt werden. Für Android Wear wird als Kommunikationskanal zwischen der Anwendung auf dem Smartphone und der Anwendung auf der Smartwatch die `Wearable API`<sup>31</sup> eingesetzt. Sie ist unter folgenden Bedingungen<sup>32</sup> nicht verfügbar:

- Android Version 4.2 (API 17) oder früher
- Android Wear Companion-App nicht installiert
- Android Wear Gerät nicht verbunden

Die `Wearable API` besteht aus unterschiedlichen Schnittstellen, wobei für die Umsetzung auf die `NodeApi` und die `MessageApi` zurückgegriffen wurde. Die `MessageApi` übermittelt die Daten des Senders an den `Data Layer`, die Übertragung zwischen den Geräten erfolgt über die Bluetooth-Verbindung bzw. über das WLAN. Auf der Empfängerseite überwacht das `WearableListenerService`<sup>33</sup> den `Data Layer`. Der Lebenszyklus dieses Services wird von Android Wear selbstständig verwaltet, eingehende Events werden ihm übergeben. Abbildung 6.1 stellt den Nachrichtenfluss bei Verwendung der `MessageApi` grafisch dar. Das Schema zeigt den Transport der Nachricht

---

<sup>29</sup> <https://github.com/Gotttox/socket.io-java-client> (besucht am 08.12.2015)

<sup>30</sup>

<https://developers.google.com/android/reference/com/google/android/gms/common/api/GoogleApiClient> (besucht am 12.12.2015)

<sup>31</sup>

<https://developers.google.com/android/reference/com/google/android/gms/wearable/Wearable> (besucht am 08.12.2015)

<sup>32</sup> [https://developers.google.com/android/guides/api-client#access\\_the\\_wearable\\_api](https://developers.google.com/android/guides/api-client#access_the_wearable_api) (besucht am 08.12.2015)

<sup>33</sup>

<https://developers.google.com/android/reference/com/google/android/gms/wearable/WearableListenerService> (besucht am 18.12.2015)



vom Smartphone in Richtung Smartwatch. Im Rahmen von Kapitel 6.3 und 6.4 finden sich die entsprechenden Codeauflistungen.

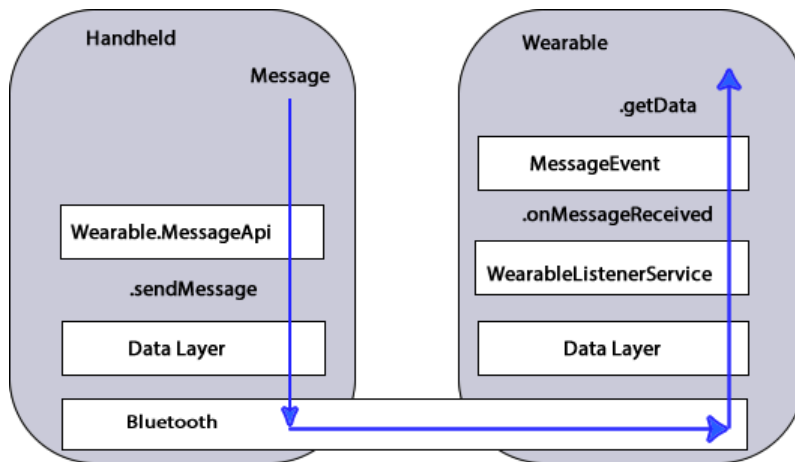


Abbildung 6.1: Kommunikation über MessageApi [Hahn, 2015]

### Socket.IO Java Client

Die Kommunikation zwischen der Smartphone-Anwendung und dem Backchannel basiert auf Socket.IO<sup>34</sup>. Socket.IO ist eine Bibliothek, welche intern das WebSocket-Protokoll verwendet. WebSockets ermöglichen eine bidirektionale Kommunikation zwischen einem Server und seinen registrierten Clients. Dabei muss von Client-Seite nur eine initiale Verbindung geöffnet werden. Der Server kann sie anschließend so lange diese aktiv ist, ohne erneute vorangegangene Anfrage des Clients, verwenden. Im Rahmen des Projektes wurde auf eine bestehende Bibliothek eines Java Clients für Socket.IO zurückgegriffen. Dabei transformiert dieser Client-Anfragen und Antworten Socket.IO kompatibel um. Intern verwendet er die Java-WebSocket-Implementierung<sup>35</sup>, welche für die Verwendung mit der neuesten Version ausgetauscht wurde. Der Socket.IO Client von Gottox wurde gewählt, weil er eine Implementierung von Socket.IO in der Version 0.9 ist und der Backchannel diese Version verwendet. Die Kompatibilität von Socket.IO Version 1.0 mit früheren Versionen ist nicht gegeben, weshalb sie migriert werden sollten<sup>36</sup>.

<sup>34</sup> <http://socket.io> (besucht am 08.12.2015)

<sup>35</sup> <http://java-websocket.org> (besucht am 08.12.2015)

<sup>36</sup> <http://socket.io/docs/migrating-from-0-9> (besucht am 08.12.2015)

## Implementierung

Um den WebSocket Client zu verwenden muss lediglich eine Instanz der SocketIO Klasse erstellt werden, wobei hier die URL mitgegeben wird und optional eine Implementierung eines Callback-Handlers. Der Callback-Handler wird intern vom WebSocket Client sowohl bei Änderungen des Verbindungsstatus als auch beim Erhalt von Nachrichten (Event, Inhalt) aufgerufen. Die weitere Verarbeitung obliegt den Entwicklern.

## 6.2 Design

### Allgemein

Die Sony Smartwatch 3 verfügt nicht über einen direkten Zugang zum Internet. Deshalb sind eigenständige Anwendungen für das Smartphone und die Smartwatch notwendig. Auch für den Verbindungsaufbau zum Backchannel und das Setzen der Einstellungen schien eine Anwendung auf dem Smartphone ohnehin sinnvoll.

Beim grundlegenden Design wurden die geschilderte Problematik bei der Anzeige des Feedback aus Kapitel 2.5.4 sowie die Design-Grundsätze für Android Wear aus Kapitel 5.3.3 berücksichtigt. Für die Lehrenden sind zwei Punkte wichtig. Einerseits sollen sie ein eingetroffenes Feedback als solches erkennen können und andererseits muss ein schnelles Verständnis der Bedeutung der Rückmeldungen gewährleistet sein. Dazu dienen sowohl Vibrationssignale als auch visuelle Signale. Ebenso darf der Vortragende nicht permanent ein Feedback erhalten. Deswegen existieren folgende Einstellungsmöglichkeiten:

#### Farbe je Dimension

Eine Dimension soll mit einer bestimmten Farbe assoziiert werden

#### Vibrationsmuster je Dimension

Eine Dimension soll mit einem bestimmten Vibrationsmuster assoziiert werden

#### Vibration Repeat

Das Vibrationsmuster wiederholt sich solange, bis entweder neue Nachrichten eingehen oder eine Bestätigung durch die Benutzerin und den Benutzer erfolgt.

### Schwellwert je Dimension

Die Vortragenden filtern die Dimensionen nach persönlicher Wichtigkeit. Nur bei Unterschreiten des Schwellwertes wird informiert. Bei Überschreitung des Schwellwertes wird die Dimension aufgrund fehlender Aktualität wieder von der Anzeige der Smartwatch entfernt.

### Snooze Time

Das Unterschreiten des Schwellwertes einer Dimension soll nicht andauernd gemeldet werden. Die Snooze Time gibt folglich das minimale Benachrichtigungsintervall an. Für jede Dimension existiert ein eigener Mechanismus, ob sie aufgrund der Snooze Time zu ignorieren ist. Des Weiteren bleibt den Vortragenden dank der Snooze Time ausreichend Zeit die Dimension zu bestätigen, da sie während dieses Zeitraumes nicht von der Anzeige entfernt werden darf.

### Confirmed Snooze Time

Wie Snooze Time, außer dass sie erst durch das explizite Bestätigen des Feedbacks ausgelöst wird - eine bestätigte Dimension wird für einen gewählten Zeitraum ignoriert. Die vortragende Person benötigt nach der Bestätigung des Empfanges ausreichend Zeit zur Reaktion. Dementsprechend hoch sollte sie gewählt werden, damit auch das Publikum in weitere Folge seine Bewertungen anpassen kann.

Die Anwendung wurde so konzipiert, dass die Benutzerin und der Benutzer sie nicht selbständig starten muss - entsprechend des Suggest-Ansatzes [siehe Kapitel 5.3.2]. Auf die Verwendung von Cards wurde bewusst verzichtet. Während eines Vortrages sollen nur die Rückmeldungen des Backchannels relevant sein. Ein Durchsuchen des Context Streams, der auch Benachrichtigungen von anderen Anwendungen enthalten könnte, wäre nicht zielbringend. Vor jedem Vortrag mittels der Companion-App die zu blockierenden Benachrichtigungen einzustellen, erschien als zu umständlich. Aus diesem Grund wird die Anwendung auf der Smartwatch für die Anzeige des Feedbacks vom Backchannel im Vollbildmodus gestartet. Bei gestarteter Anwendung muss der Bildschirm aktiv bleiben, da bei Versäumnis eines Vibrationssignales so zumindest die visuelle Rückmeldung verbleibt. Eine Dimension, die ihren Schwellwert unterschritten hat, wird am Display in der gewählten Farbe dargestellt, bei mehr als einer Dimension teilt sich der Bildschirm.

### Smartphone App

Die Anwendung auf dem Smartphone dient als Verbindungstück zwischen dem Backchannel und der Smartwatch. Die empfangenen Nachrichten vom Backchannel werden interpretiert und je nach Einstellungen entschieden, ob die Dimensionen auf der Smartwatch anzuzeigen bzw. zu verstecken sind oder die Anwendung auf der Smartwatch zu schließen ist. Des Weiteren sind die Einstellungen für die Anwendung mit ihr festlegbar. Für die einzelnen Aufgaben wurde die Anwendung in Komponenten zerlegt. Für das User Interface sind nur zwei Activities notwendig - eine für die Anzeige der Werte und eine für die Einstellungen. Statusmeldungen werden über Dialoge der Benutzerin und dem Benutzer mitgeteilt. Im Hintergrund arbeiten das ARSWatchService und das WearListenerService eng mit dem BackchannelClient und dem WearClient zusammen.

### Smartwatch App

Die Anwendung auf der Smartwatch hat primär die Aufgabe der Vortragenden und dem Vortragenden per Vibrationssignal auf die Unterschreitung eines festgelegten Schwellwertes aufmerksam zu machen und die aktuellen Werte vom Backchannel zu visualisieren. Das User Interface existiert hierzu lediglich aus einer einzigen Activity. Im Hintergrund horcht das PhoneListenerService auf eingehende Benachrichtigungen und startet bzw. schließt die Anwendung oder aktualisiert die Anzeige.

### Icon

Für die Anwendung wurde ein einfaches, aber sprechendes Icon entworfen - siehe Abbildung 6.2.

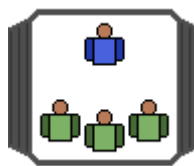


Abbildung 6.2: Icon der Anwendung

## 6.3 Smartphone App

In diesem Kapitel werden die wichtigsten Komponenten der Smartphone Anwendung und deren Funktionalität erläutert.

### MainActivity

Die MainActivity ist der Eintrittspunkt der Anwendung und lädt bei ihrer Erstellung das definierte Layout [siehe Abbildung 6.3]. Vom Startschirm aus kommt man über das Optionsmenü zu den Einstellungen. Im oberen Bereich wird der Verbindungsstatus, die Lecture ID und die Anzahl der Teilnehmerinnen und Teilnehmer eingeblendet. Darunter folgen die aktuellen Werte der drei Dimensionen Happiness, Comprehension und Speed, welche in Form von Balken grafisch aufbereitet sind. Unterhalb eines jeden Balkens befindet sich ein "T", das den eingestellten Schwellwert (threshold) signalisiert. Im letzten Bereich findet man einen Connect-Button und einen Disconnect-Button.

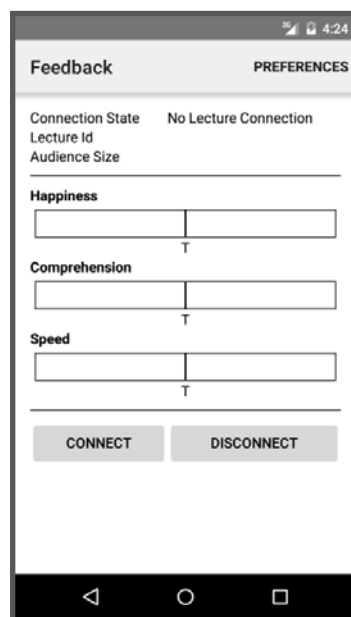


Abbildung 6.3: Startschirm Smartphone App

Beim Starten der Anwendung wird eine ServiceConnection erstellt, über die sich die MainActivity mit dem ARSWatchService verbindet. Die MainActivity ist die einzige Komponente, die dieses Services bindet, wodurch mit ihrer Lebensdauer auch jene des Services endet. Über einen

## Implementierung

sogenannten Binder kommuniziert die MainActivity mit dem ARSWatchService. Dieser wird als Schnittstelle zum Service eingesetzt, um die Verbindung zum Backchannel aufzubauen bzw. zu trennen. Ein der MainActivity zugehöriger Handler<sup>37</sup> fungiert für das Service als Callback-Mechanismus. Eine schematische Darstellung von Becker & Pant [2015] zeigt die Abbildung 6.4.

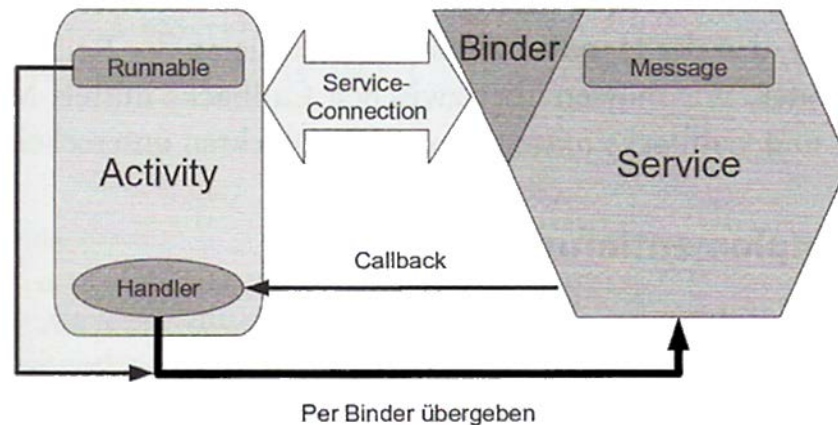


Abbildung 6.4: Gebundenes Service [Becker & Pant, 2015]

### ARSWatchService

Das ARSWatchService dient als Schnittstelle zwischen dem Backchannel und der Smartwatch. Dabei hält das Service je eine Instanz des BackchannelClient und des WearClient.

Alle Nachrichten des Backchannels - vom BackchannelClient an das Service weitergeleitet - aktualisieren mittels des Handler die MainActivity. Die Nachrichten an die Smartwatch - vom Service an den WearClient weitergegeben - werden hingegen gefiltert. Dabei ist entscheidend ob mindestens eine Dimension aufgrund der eingestellten Snooze Time und Confirmed Snooze Time aktualisiert werden darf. Dimensionen werden ignoriert (IGNORE), wenn sie aufgrund der beiden Zeitspannen noch nicht

---

<sup>37</sup> Mittels eines Handler werden Nachrichten bzw. ausführbare Objekte in die MessageQueue des zugehörigen Threads gestellt. Die Einträge werden nach dem FIFO-Verfahren abgearbeitet.

<http://developer.android.com/reference/android/os/Handler.html> (besucht am 23.12.2015)

verändert werden dürfen. Für die anderen wird der eingestellte Schwellwert schlagend. Befindet sich der tatsächliche Wert unterhalb des Schwellwertes soll die Dimension auf der Smartwatch angezeigt werden (SHOW), anderenfalls soll sie ausgeblendet werden (HIDE) [siehe Auflistung 6.1]. Des Weiteren übergibt das ARSWatchService dem WearClient die Statusmeldungen.

```

...
1  if (happinessUpdateable && happinessConfirmedUpdateable) {
2      if (Double.compare(bUpMsg.getThresholdHappiness(), valueHappiness) >= 0) {
3          bUpMsg.setHappinessStatus(SHOW);
4          happinessUpdateable = false;
5          resetUpdateable(DIMENSION_HAPPINESS, false, snoozeTime);
6      } else {
7          bUpMsg.setHappinessStatus(HIDE);
8      }
9  }
...

```

Auflistung 6.1: Prüfen ob Dimension aktualisiert werden kann

### BackchannelClient

Der BackchannelClient baut eine Verbindung zum Backchannel über Socket.IO auf, um von ihm Nachrichten zu empfangen und sie über den Binder an das ARSWatchService zu übermitteln. In der Anwendung läuft der Verbindungsaufbau zum Backchannel in zwei Schritten ab.

Im ersten Schritt wird ein POST-Request an folgende URL abgesetzt, wobei `<lecture_id>` entsprechend zu ersetzen ist:

*[http://backchannel.cnc.io:80/api/v1/lecture/<lecture\\_id>/join](http://backchannel.cnc.io:80/api/v1/lecture/<lecture_id>/join)*

Die Antwort beinhaltet unter anderem den eindeutigen "auditoraccesshash", der für die WebSocket-Verbindung benötigt wird. Dank des "auditoraccesshash" weiß der Backchannel vom welcher Teilnehmerin und welchem Teilnehmer eine Bewertung abgegeben wurde.

Im zweiten Schritt kommt der Socket.IO Java Client zum Einsatz, um die WebSocket-Verbindung herzustellen. Der WebSocket-Server des Backchannels ist unter folgender URL erreichbar:

*<ws://backchannel.cnc.io/socket.io/auditor>*

## Implementierung

Das zu übermittelnde Event für die Teilnahme an einem Vortrag lautet "auditor\_connect". Der Inhalt der Nachricht an den WebSocket-Server enthält den zuvor erhaltene "auditoraccesshash" sowie die "lectureid". Jedes Vote einer Teilnehmerin und eines Teilnehmers beinhaltet diese zwei Parameter für eine eindeutige Zuordnung. In der Datenbank des Backchannels werden zusätzlich der Timestamp, der ausgewählte Wert und die Bezeichnung der Dimension gespeichert [Haintz, 2013].

Nach der Registrierung des BackchannelClient beim Backchannel empfängt er alle Nachrichten. Nachrichten des Events "changed\_stats" enthalten sowohl die Anzahl der Teilnehmerinnen und Teilnehmer als auch die Werte der drei Dimensionen. Folgende Form weist eine beispielhafte Nachricht auf:

```
{"num_users": "1", "speed": -55.671, "comprehension": 2.983, "happiness": 77.254}
```

### WearClient

Der WearClient hat die Aufgabe erhaltene Nachrichten vom Backchannel sowie Meldungen über den Verbindungstatus an die Smartwatch zu übermitteln. Für den Verbindungsaufbau zur Smartwatch wird der GoogleApiClient instanziiert und die Wearable API hinzugefügt [siehe Auflistung 6.2].

```
1 public WearClient(final ARSWatchService context, final
2 ARSWatchService.ARSServiceBinder arsWatchService) {
3     this.arsWatchService = arsWatchService;
4     googleApiClient = new GoogleApiClient.Builder(context)
5         .addApi(Wearable.API)
6         .addConnectionCallbacks(this)
7         .addOnConnectionFailedListener(this)
8         .build();
9
10    appClosed = false;
11 }
```

Auflistung 6.2: Instanzieren des GoogleApiClient



Mit Hilfe der `NodeApi`<sup>38</sup> werden die verbundenen Knoten abgefragt [siehe Auflistung 6.3]. Die `MessageApi`<sup>39</sup> dient schließlich der Übermittlung von Nachrichten an die Knoten, dabei wird die zuvor erhaltene eindeutige `nodeId` verwendet [siehe Auflistung 6.4].

```

1  private Collection<String> getNodes() {
2      HashSet<String> results = new HashSet<String>();
3      NodeApi.GetConnectedNodesResult nodes =
4          Wearable.NodeApi.getConnectedNodes(googleApiClient)
5              .await(ArsWatchConstants.NODES_TIMEOUT, TimeUnit.SECONDS);
6
7      for (Node node : nodes.getNodes()) {
8          results.add(node.getId());
9      }
10
11     return results;
12 }

```

Auflistung 6.3: Abfragen der Knoten mit der `NodeApi`

```

1  private void send(final String nodeId, final String path, final String message) {
2      Log.d(TAG, "send:" +
3          " nodeId: " + nodeId +
4          " path: " + path +
5          " message: " + message);
6
7      Wearable.MessageApi
8          .sendMessage(googleApiClient, nodeId, path, message.getBytes())
9          .setResultCallback(new ResultCallback<MessageApi.SendMessageResult>() {
10             @Override
11             public void onResult(final MessageApi.SendMessageResult
12                 sendMessageResult) {
13                 Log.d(TAG, "send: sendMessageResult.getStatus().isSuccess(): "+
14                     sendMessageResult.getStatus().isSuccess());
15             }
16         });
17 }

```

Auflistung 6.4: Senden einer Nachricht mit der `MessageApi`

<sup>38</sup>

<https://developers.google.com/android/reference/com/google/android/gms/wearable/NodeApi> (besucht am 12.12.2015)

<sup>39</sup>

<https://developers.google.com/android/reference/com/google/android/gms/wearable/MessageApi> (besucht am 12.12.2015)

### **WearListenerService**

Das `WearListenerService` dient dem Empfang von Nachrichten die von der Anwendung auf der Smartwatch gesendet werden. In Kapitel [6.4](#) unter `PhoneListenerService` findet sich eine vertiefende Erläuterung. Bei den Nachrichten kann es sich um eine Statusabfrage oder um eine Bestätigung eines Feedbacks handeln. Diese werden an das `ARSWatchService` weitergegeben. Eine Bestätigungsmeldung enthält die Dimensionen, die für eine festgelegte Zeitspanne nicht aktualisiert werden dürfen. Die Dauer hängt von der eingestellten `Confirmed Snooze Time` ab.

## 6.4 Smartwatch App

In diesem Kapitel werden die wichtigsten Komponenten der Smartwatch Anwendung und deren Funktionalität erläutert.

### MainActivity

Die MainActivity ist der Eintrittspunkt der Anwendung und lädt bei ihrer Erstellung das definierte Layout. Für die Benutzerin und den Benutzer existiert nur eine einzige Interaktionsmöglichkeit. Dabei handelt es sich um die Bestätigung des Empfangs einer Rückmeldung vom Backchannel. Mittels Tippen auf das Display werden die angezeigten Dimensionen als bestätigt markiert. Mit dem GoogleApiClient wird eine Nachricht über die Bestätigung an die Smartphone Anwendung übermittelt, bei welcher das WearListenerService [siehe Kapitel 6.3] auf diese Nachrichten horcht.

### PhoneListenerService

Das PhoneListenerService soll über eingehende Events benachrichtigt werden. Aus diesem Grund leitet dieses Service vom WearableListenerService<sup>40</sup> ab und erbt seine Funktionalität. Der Eintrag des PhoneListenerService in die Manifestdatei der Android-Anwendung wird in Auflistung 6.5 dargestellt. Das Betriebssystem bindet das PhoneListenerService und je nach Typ des Events wird die entsprechende Methode aufgerufen. Auflistung 6.6 zeigt den Erhalt eines MessageEvents. Aufgrund der erhaltenen Informationen wird die Anwendung gestartet bzw. die Anzeige aktualisiert oder die Anwendung geschlossen, wenn sich keine Dimension unter ihren definierten Schwellwert befindet.

```

1 <service android:name=".PhoneListenerService">
2   <intent-filter>
3     <action android:name="com.google.android.gms.wearable.BIND_LISTENER"/>
4   </intent-filter>
5 </service>

```

Auflistung 6.5: Eintrag des PhoneListenerService in der Manifestdatei

<sup>40</sup>

<https://developers.google.com/android/reference/com/google/android/gms/wearable/WearableListenerService> (besucht am 18.12.2015)

## Implementierung

```
1  @Override
2  public void onMessageReceived(final MessageEvent messageEvent) {
3      String message = new String(messageEvent.getData());
4      String path = messageEvent.getPath();
5
6      Log.d(TAG, "onMessageReceived:" +
7          " requestId: " + messageEvent.getRequestId() +
8          " sourceNodeId: " + messageEvent.getSourceNodeId() +
9          " path: " + path +
10         " message: " + message);
11
12     if (path.equals(ArsWatchConstants.PATH_STATUS)) {
13         handleStatus(message);
14     } else if (path.equals(ArsWatchConstants.PATH_CONTROL_CONNECTION)) {
15         handleControlConnection(message);
16     } else if (path.equals(ArsWatchConstants.PATH_UPDATE)) {
17         handleUpdate(message);
18     }
19 }
```

Auflistung 6.6: Auszug PhoneListenerService - Erhalt eines MessageEvents

## 6.5 Use Cases

Dieses Kapitel schilderte die wichtigsten Use Cases der Anwendung. Zur Darstellung werden Screenshots von dem Smartphone und von der Smartwatch verwendet, welche mit dem Emulator von Android Studio aufgenommen wurden. Die Anwendungsfälle sind:

- Verbindung herstellen bzw. trennen
- Erhalt des Feedbacks
- Setzen der Einstellungen

### Verbindung herstellen bzw. trennen

Das Herstellen bzw. Trennen einer Verbindung zum Backchannel erfolgt über den Connect-Button bzw. Disconnect-Button. Ein Dialog zur Anzeige des jeweiligen Verbindungsstatus wird eingeblendet [siehe Abbildung 6.5].

#### Smartphone App

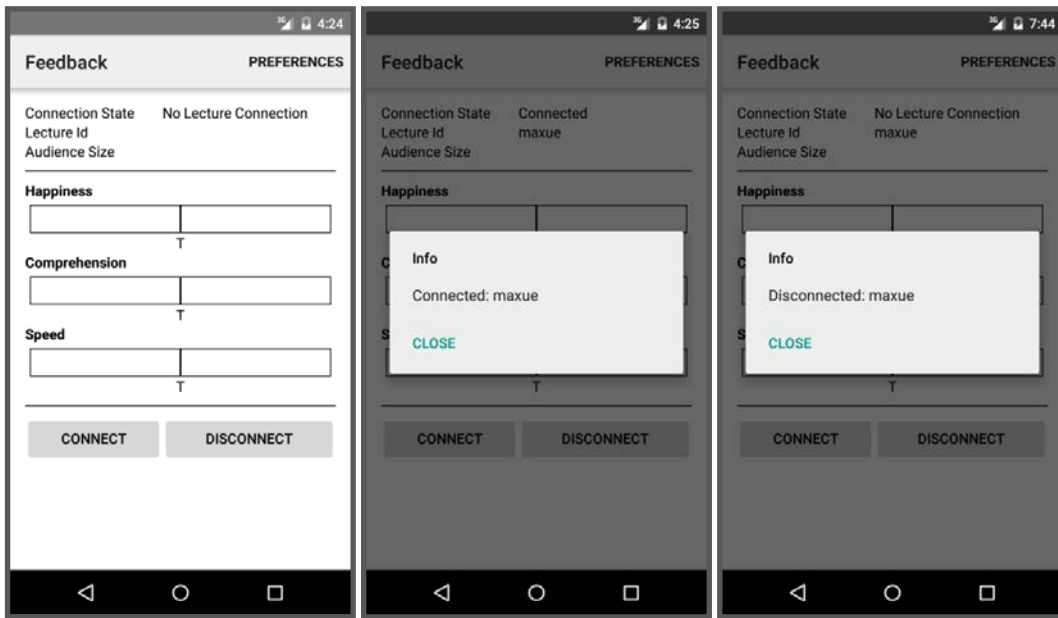


Abbildung 6.5: Verbindungsstatus Smartphone App

#### Smartwatch App



Abbildung 6.6: Verbindungsstatus Smartwatch App

## Implementierung

### Erhalt des Feedbacks

Die empfangenen Nachrichten des Backchannels werden in der Anwendung am Smartphone [siehe Abbildung 6.7] und auf der Smartwatch [siehe Abbildung 6.8] dargestellt.

### Smartphone App

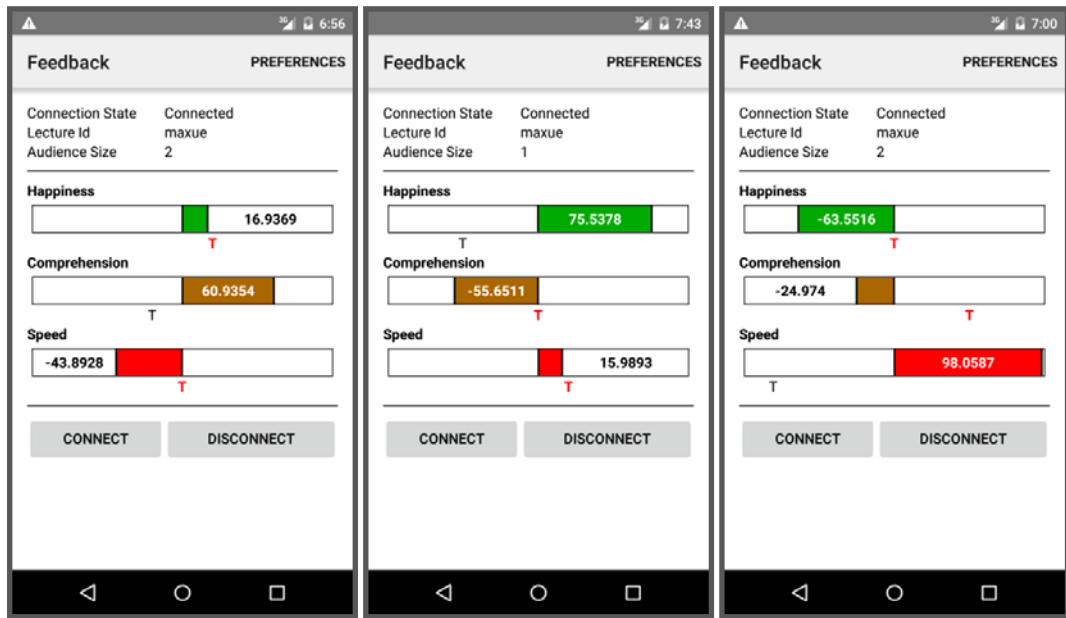


Abbildung 6.7: Anzeige einer Rückmeldung auf dem Smartphone

### Smartwatch App

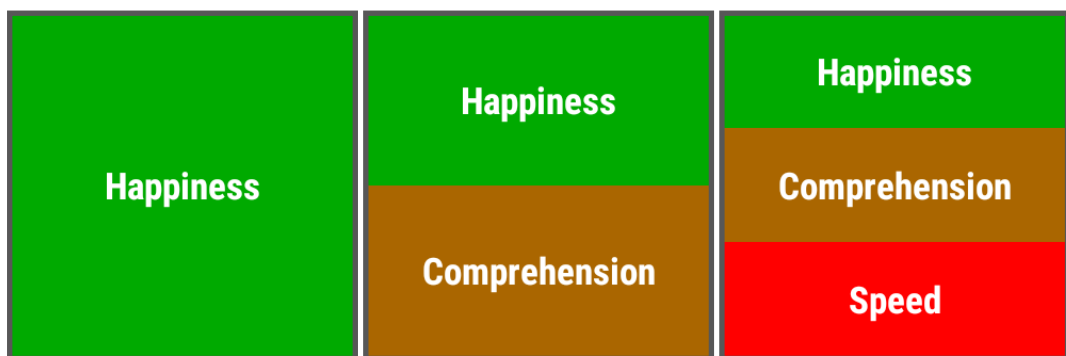


Abbildung 6.8: Anzeige einer Rückmeldung auf der Smartwatch

### Setzen der Einstellungen

Unter den Einstellungen sind die Lecture ID, das Ignore Wear Error Flag sowie die in Kapitel 6.2 angesprochenen Parameter festlegbar.

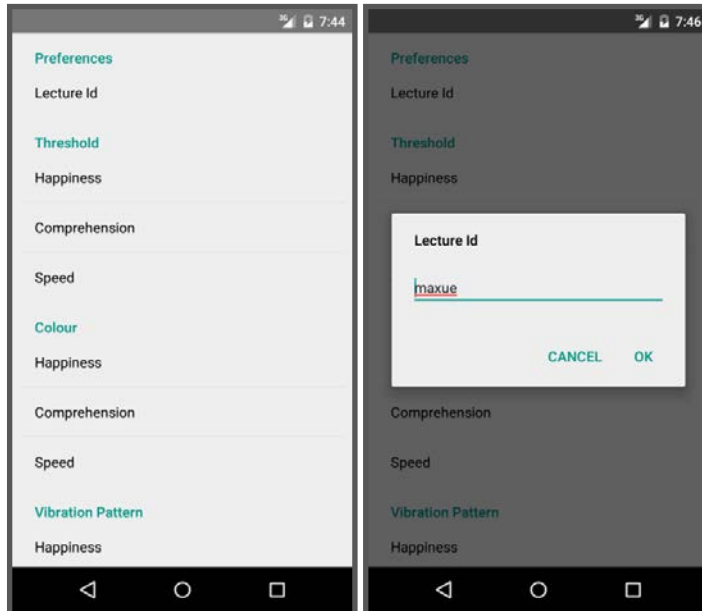


Abbildung 6.9: Einstellen der Lecture ID

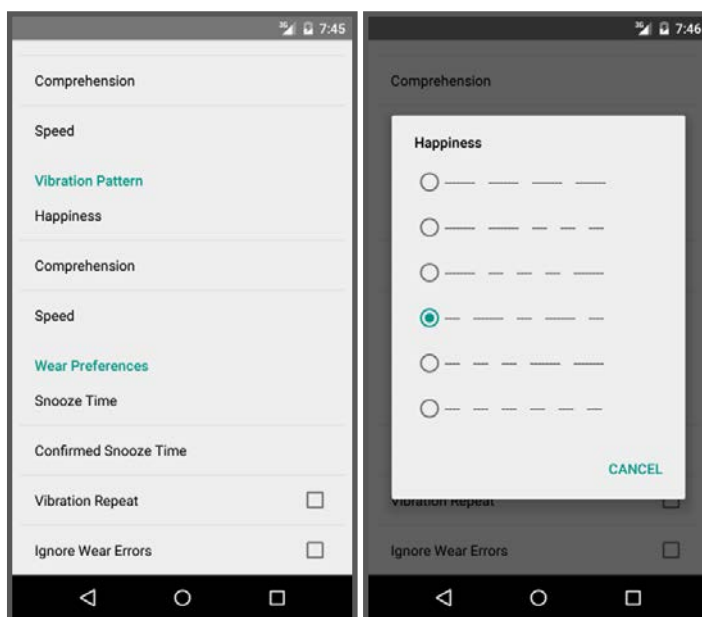


Abbildung 6.10: Auswahl eines Vibrationsmusters

## 6.6 Fehlerhandling

Die Anwendung soll auf potentielle Fehler mit einer passenden Meldung reagieren. Ein typisches Szenario wäre der Verlust der Verbindung mit dem Internet. Die Vortragende und der Vortragende soll darüber informiert sein, dass keine weiteren Rückmeldungen vom Backchannel empfangen werden können. Abbildung 6.11 zeigt einen Screenshot von dem Smartphone und einen Screenshot von der Smartwatch.

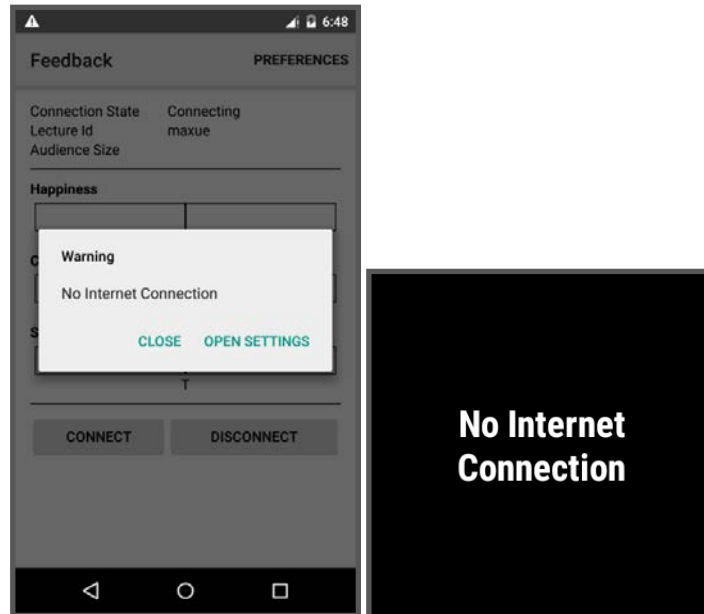


Abbildung 6.11: Fehlermeldung Verbindung zum Internet

Eine andere mögliche Fehlerquelle birgt der Verbindungsaufbau mit der Wearable API über den GoogleApiClient. Dies könnte auftreten, wenn die Companion-App nicht installiert ist oder sich die installierte Google Play Service Version nicht mehr auf dem aktuellen Stand befindet [siehe linke Fehlermeldung aus Abbildung 6.12]. Ebenso könnte keine Smartwatch verbunden sein bzw. sie sich außer Reichweite befinden. Liefert die NodeApi eine leere Liste von verbundenen Knoten zurück, wird die rechte Fehlermeldung von Abbildung 6.12 eingeblendet. Für den Fall, dass die Smartphone-Anwendung ohne Smartwatch eingesetzt wird, existiert die Einstellung "IgnoreWear Errors". Durch Setzen dieser würden die genannten Fehler ignoriert werden.



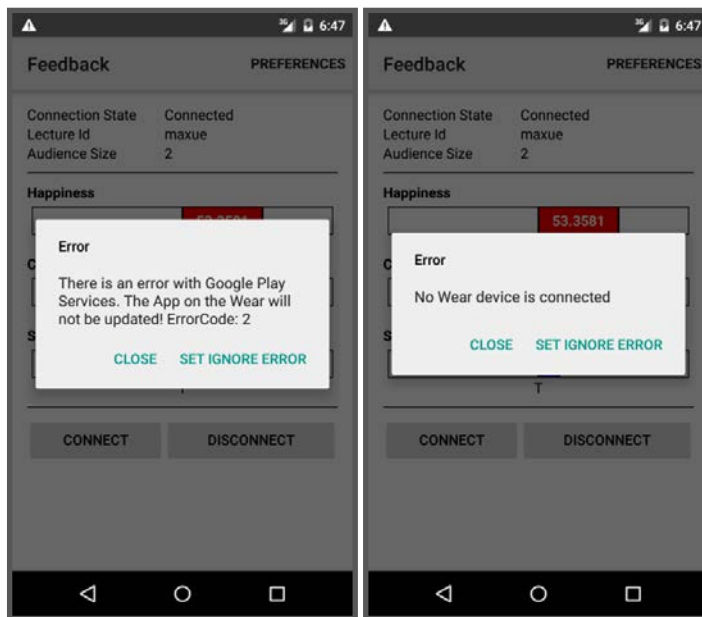


Abbildung 6.12: Fehlermeldung Google Play Services; Fehlermeldung Verbindung zur Smartwatch

Ein Fehler auf den die laufende Smartwatch-Anwendung reagieren soll, wäre das nicht erreichbar sein der Smartphone-Anwendung [siehe Abbildung 6.13]. Beispielsweise befindet sich die Vortragende und der Vortragende außerhalb der Reichweite zum Smartphone. Auch ein Programmabsturz der Smartphone-Anwendung könnte einen plausiblen Grund darstellen.



Abbildung 6.13: Fehlermeldung Smartphone Anwendung nicht gestartet

## Implementierung

## 7 Evaluation

Dieses Kapitel fasst die Evaluation der Integration der Smartwatch in den Backchannel zusammen. Dazu musste zunächst ein erster Testeinsatz durchgeführt werden [siehe Kapitel 7.1]. Nach der ersten Verwendung des Prototyps wurden diverse Rückmeldungen betroffener Personen eingeholt [siehe Kapitel 7.2]. Im Rahmen der Analyse [siehe Kapitel 7.3] werden die persönlichen Beobachtungen geschildert, die Testdaten ausgewertet und die Smartwatch bewertet.

### 7.1 Testeinsatz

Nach Abschluss der Implementierungsphase des Prototyps wurde eine geeignete Umgebung für einen Testeinsatz gesucht. Der erste Live-Test fand im Rahmen einer Veranstaltung der Welcome Days der TU Graz am 29. Oktober 2015 statt<sup>41</sup>. Die Örtlichkeit war ein Hörsaal der Karl-Franzens-Universität Graz, der ein Fassungsvermögen von knapp unter 500 Personen hat und vollständig belegt war. Da es sich um die Einführungstage für Studienanfängerinnen und Studienanfänger handelte, kann für den Großteil der Personen ein geschätztes Alter von rund 20 Jahren angenommen werden. Ein Bericht von Nielsen besagt, dass bei der Generation Y die größte Dichte an Smartphone-Besitzern vorliegt - ein Wert von über 85%<sup>42</sup>. Die privaten Smartphones, Tablets etc. konnten somit entsprechend des BYOD-Ansatz als Eingabegeräte verwendet werden. Der Teil der Veranstaltung in dem der Backchannel zum Einsatz kam, befasste sich thematisch mit E-Learning und die an der TU Graz für diesen Bereich eingesetzten Technologien<sup>43</sup>. Somit waren optimale Voraussetzungen für den ersten Einsatz des Prototyps gegeben.

Den Vortrag hielt Herr Dr. Martin Ebner, der die Leitung der Abteilung "Vernetztes Lernen" des Zentralen Informatikdienst der TU Graz innehat.

---

<sup>41</sup> <http://welcomedays.tugraz.at> (besucht am 05.11.2015)

<sup>42</sup> Als Generation Y, auch als Millennials bezeichnet, definiert Nielsen Personen, die zwischen 1977 und 1995 geboren wurden.

<http://www.nielsen.com/us/en/insights/news/2014/mobile-millennials-over-85-percent-of-generation-y-owns-smartphones.html> (besucht am 21.12.2015)

<sup>43</sup> <http://www.tugraz.at/oe/lehr-und-lerntechnologien/lehrtechnologien-und-services/> (besucht am 20.01.2016)

## Evaluation

Die Einsatzdauer der Smartwatch betrug rund eine halbe Stunde. Einleitend gab es für das Publikum eine kurze Erklärung zur Bedienung des Backchannels und der Bedeutung der drei Eingabedimensionen. Der Referent erstellte einen neuen Vortrag und gab die Lecture ID an die Zuhörerinnen und Zuhörer weiter [siehe Kapitel 5.1.1].

### 7.1.1 Konfiguration des Prototypen

Die Schwellwerte wurden wie folgt festgelegt:

Zufriedenheit	Wert ignorieren
Verständnis	Wert kleiner oder gleich 0.0
Geschwindigkeit des Vortrages	Wert kleiner oder gleich 0.0

Die Feedbackintervalle wurden wie folge eingestellt:

Snooze Time	30 Sekunden
Confirmed Snooze Time	1 Minute

## 7.2 Rückmeldungen

Eine Zeit nach der Veranstaltung wurde Herr Dr. Martin Ebner aus Sicht des Vortragenden und Herr DI Christian Haintz aus der Sicht eines Teilnehmers sowie aus der Sicht eines Entwicklers des Backchannels um ein Feedback gebeten. Die Fragen bezogen sich auf die Eindrücke zum Vortrag sowie zum generellen Einsatz der Smartwatch in Verbindung mit einem ARS. Von Interesse war vor allem die Wahrnehmbarkeit des Feedbacks für den Vortragenden. Der nächste Abschnitt gibt die per E-Mail erhaltenen Rückmeldungen wieder.

### 7.2.1 Vortragender

Fühlten Sie sich durch die Smartwatch von Ihrem Vortrag abgelenkt?

*Nein gar nicht, man trägt einfach eine Uhr.*

Konnten Sie sich gleichzeitig auf Ihre Vortragsunterlagen und auf die Smartwatch konzentrieren?

*Grundsätzlich ja, andererseits ist mir nicht immer die Vibration aufgefallen.*

Konnten Sie eine erhaltene Rückmeldung leicht wahrnehmen?

*Teilweise - manchmal war es gut spürbar, manchmal bemerkte ich es nicht. Liegt wohl ein wenig daran wie die Hand wohl liegt, bzw. wie konzentriert man gerade an etwas denkt.*

Waren die Feedbacks verständlich aufbereitet?

*Ja, aber es ist gewöhnungsbedürftig, d.h. man braucht ein wenig Übung darin.*

Haben Sie zu oft eine Rückmeldung erhalten?

*Kann ich so nicht beantworten, weil ich ja nicht alles gespürt habe.*

Sehen Sie generelle Vorteile bei am Körper anliegenden Geräten zum Erhalt eines Feedbacks?

*Ja, die Vibration ist deutlich besser spürbar, als z.B. beim Smartphone und sie ist nicht störend - sowohl für mich als auch für das Auditorium. Die Rückmeldung ist so "smarter" als mit anderen Devices.*

### **7.2.2 Teilnehmer**

Hatten Sie das Gefühl, dass der Vortragende durch die Smartwatch abgelenkt wurde?

*Nein.*

Konnte Ihrer Meinung nach der Vortragende durch das Feedback schneller auf das Publikum reagieren?

*Ja.*

### **7.2.3 Entwickler des Backchannel**

Welche Potentiale sehen Sie generell bei der Smartwatch als "verlängerter Arm" eines Audience Response Systems?

*Armbanduhren sind in unserer Gesellschaft allgegenwärtig und damit keine Irritation für die Gesellschaft. Eine Smartwatch kann daher als unauffälliges und nicht störendes Eingabe- aber vor allem als Ausgabegerät genutzt werden. Vor allem im ARS Kontext bietet das Möglichkeiten den Vortragenden subtil und in live auf Geschehnisse im Publikum aufmerksam zu machen die das ARS wahrnimmt.*

Empfinden Sie es als wichtig, dass die vortragende Person bei der Verwendung des Backchannels nicht an ein fix positioniertes Gerät gebunden ist?

*ARS Systeme sollten den Vortragenden unterstützen und so wenig wie möglich einschränken. Die Abhängigkeit von einem fix positionierten Gerät ist daher so gut wie geht zu vermeiden.*

Würden Sie die Smartwatch bei Vorträgen, die durch den Backchannel unterstützt werden, einsetzen?

*Ja.*

## **7.3 Analyse**

### **7.3.1 Persönliche Beobachtungen**

Zu Beginn des Vortrages schienen die Teilnehmerinnen und Teilnehmer viele Bewertungen abzugeben. Demensprechend schwankten die Werte der drei Dimensionen. Nach wenigen Minuten flachte die Beteiligung ab und die Werte pendelten sich ein. Zwischenzeitlich hatte man als Beobachter den Eindruck, als würden kaum noch Votes durchgeführt. Durch den eingebauten Aging-Mechanismus des Backchannels näherten sich die Werte langsam dem Wert 0 an. Erst gegen Ende des Vortrages traten wieder vermehrt Schwankungen auf. Die anfänglichen Votes flossen aufgrund ihres Alters - Aging-Mechanismus - nicht mehr in die Bewertung ein. Gleiches galt für Votes von getrennten Verbindungen zum Backchannel. Daher konnten einzelnen Bewertung einen größeren Einfluss nehmen. Unter anderem sank die Dimension für die Geschwindigkeit des Vortrages unter den festgelegten Schwellwert und die Anwendung auf der Smartwatch wurde aktiv. Den tatsächlichen Erhalt des Feedbacks untermauerte der Vortragende sinngemäß mit den Worten: "OK, ich bin anscheinend zu schnell." Er reagierte prompt und verlangsamte sein Tempo.

Die Kontrollblicke des Vortragenden auf die Smartwatch waren für mich nicht ersichtlich. Demensprechend hatte ich nie das Gefühl, dass er durch die Smartwatch von seiner Präsentation abgelenkt war. Als Schwellwerte unterschritten waren, empfand ich die Reaktion des Referenten als verzögerungslos. Die Rückmeldungen des Backchannels direkt auf der Smartwatch zu erhalten, erschien mir als großer Vorteil, da das

Präsentationsgerät kaum bedient wurde. Nachfolgende Tabelle gibt einen groben Überblick über die Themen des Vortrages in Verbindung mit der jeweiligen Uhrzeit:

Thema	Uhrzeit
Einleitung	13:25
TeachCenter	13:27
TeachCenter	13:28
TeachCenter	13:30
Tube	13:34
Tube	13:37
MyApps - TU Graz WebApps/iMoox	13:39
ABC E-Books	13:40
TU Graz Learning Lab	13:41
TU Graz Learning Lab	13:42
Fragen zum Vortrag	13:43
Skripten an der TU Graz	13:45

In Abbildung 7.1 sind die beobachteten Werte von den drei Dimensionen zu denselben Uhrzeiten ersichtlich - die Daten wurden händisch mitgeschrieben. Dabei fällt auf, dass beim Thema „TU Graz Learning Lab“ die Werte in den Negativbereich gefallen sind. Ebenso konnte bei der Auswertung der Daten [siehe Kapitel 7.3.2] eine erhöhte Anzahl an Votes in diesem Zeitraum festgestellt werden [siehe Abbildung 7.2].

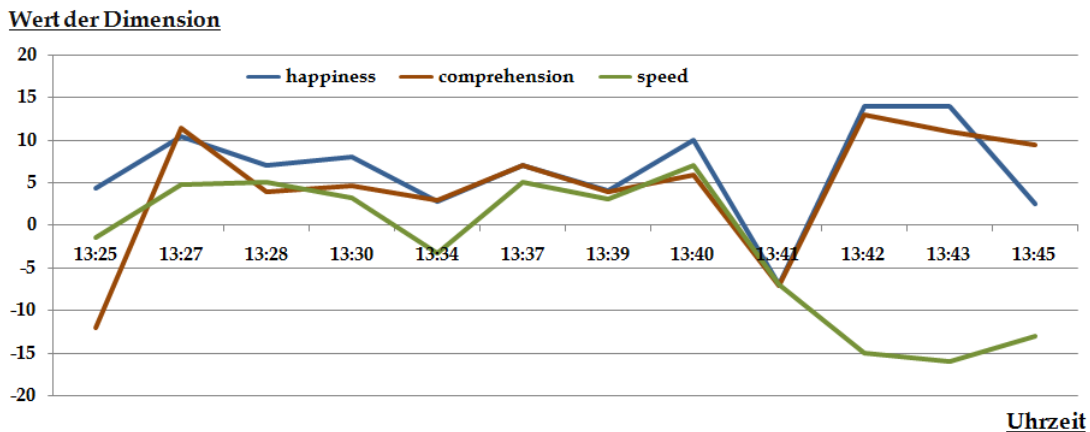


Abbildung 7.1: Verlauf der Werte über die Zeit

### 7.3.2 Auswertung der Daten

Die Firma Carrot & Company GmbH stellte mir im Anschluss die zum Vortrag gesammelten Daten (die Votes der Teilnehmerinnen und der Teilnehmer) zur Verfügung. Anhand dieser Daten wurden zwei Auswertungen durchgeführt.

#### Voteverhalten des Auditoriums

##### Anzahl User/Votes

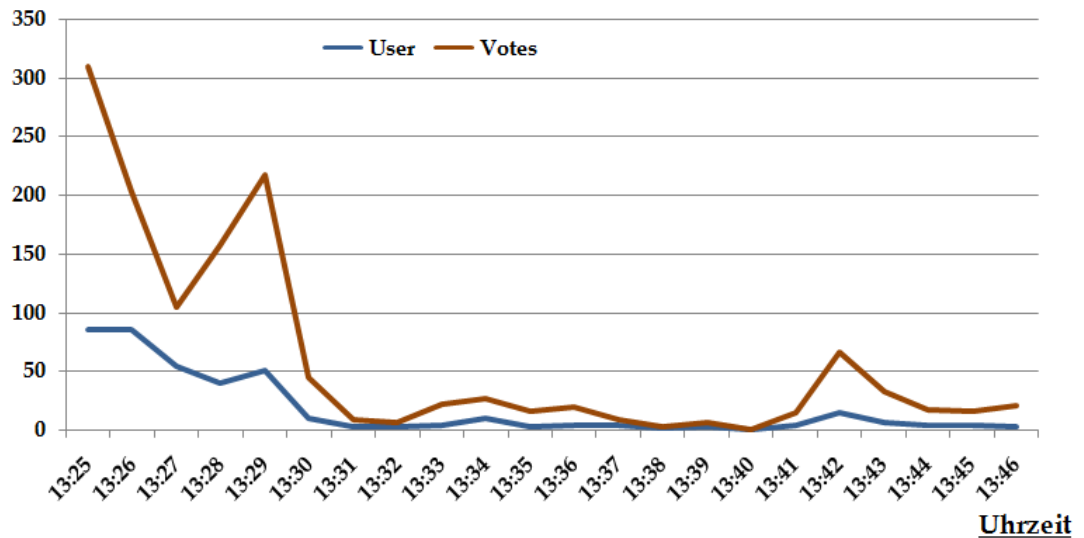


Abbildung 7.2: Voteverhalten über die Zeit

Mit der ersten Auswertung wurde das Voteverhalten des Auditoriums reproduziert. Die Abbildung 7.2 zeigt die Anzahl der Votes sowie die Anzahl der Benutzerinnen und Benutzer, die ein Vote abgaben, innerhalb einer bestimmten Zeitspanne<sup>44</sup>. Diese Auswertung untermauert die persönlichen Beobachtungen dahingehend, dass zu Beginn des Vortrages eine Vielzahl von Bewertungen abgegeben wurden und mit Dauer der Präsentation das Voteverhalten des Publikums rückläufig war. Daraus ist anzunehmen, dass die Teilnehmerinnen und Teilnehmer zuerst das System ausprobieren wollten, schlussendlich aber vorwiegend auf den Vortragenden fokussiert waren.

<sup>44</sup> Die Uhrzeit 13:25 beinhaltet die Votes der Zeitspanne 13:25:00 bis 13:25:59



## Werte der Dimensionen

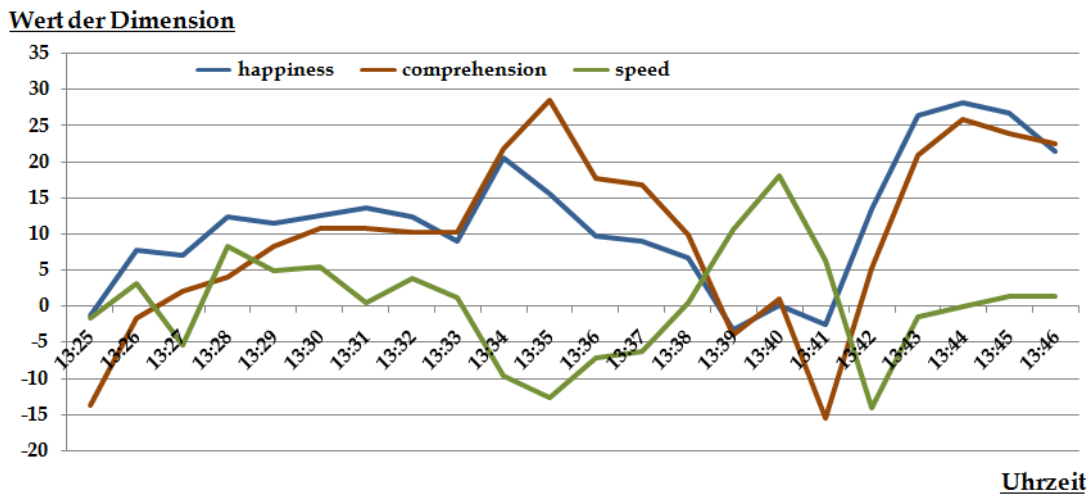


Abbildung 7.3: Dimensionen über die Zeit

In der zweiten Auswertung wurden die Gesamtwerte für die drei Dimensionen nachgerechnet, wobei diese Daten nicht mit den beobachteten Werten verglichen werden können. Herangezogen für die Grafik in Abbildung 7.3 wurden die Rohdaten, der Aging-Mechanismus und getrennte Verbindungen mussten unberücksichtigt bleiben. Stattdessen sind immer jene Votes eingeschlossen, die jünger als fünf Minuten sind<sup>45</sup>.

<sup>45</sup> Die Uhrzeit 13:25 beinhaltet die Votes der Zeitspanne 13:21:00 bis 13:25:59

### 7.3.3 Potential der Smartwatch

#### Allgemein

Die Smartwatch stellte sich als geeigneter Formfaktor für die Anbindung an ein ARS heraus. Viele Personen tragen ohnehin ständig eine Uhr am Handgelenk, eine Eingewöhnungsphase entfällt somit für die meisten Benutzerinnen und Benutzer. Die Smartwatch-Anwendung erfordert abgesehen von der Bestätigung einer Rückmeldung keine Benutzereingaben. Ebenso fällt es dem Publikum voraussichtlich nicht auf, ob es sich um eine traditionelle oder eine smarte Uhr handelt. Abbildung 7.4 zeigt die Anwendung auf einem Smartphone und auf der Sony SmartWatch 3.

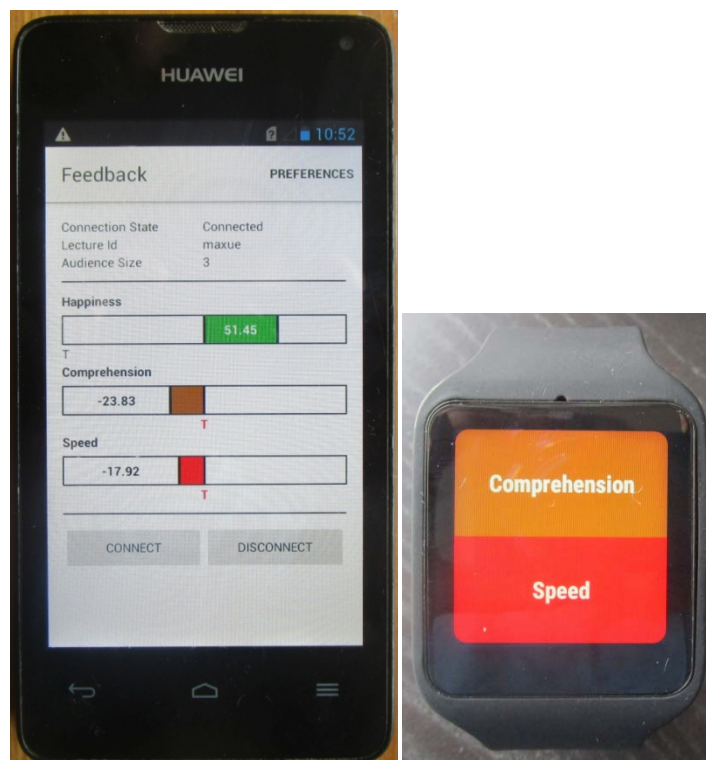


Abbildung 7.4: Anwendung im Praxiseinsatz

Ebner [2015] führte eine ähnliche Arbeit mit der Google Glass und ebenfalls dem Backchannel durch. Dabei wurden die Bewertungen des Publikums visuell mit der Brille dargestellt. Der Autor berichtet von einer Eingewöhnungsphase in der Nutzung - vor allem für Nichtbrillenträger - und dass sich ein Teil des Publikums beobachtet fühlte.

### **Signalwahrnehmung**

Das Vibrationssignal der Smartwatch kann prinzipiell gut wahrgenommen werden. Im Redefluss könnten einzelne Signale aber durchaus verpasst werden. Aus diesem Grund wurde die Anwendung so konzipiert, dass sich der Bildschirm bei angezeigter Anwendung nicht ausschaltet. Die Rückmeldungen bleiben damit mindestens bis zur eingestellten Snooze Time visuell wahrnehmbar. Darum sollte die Smartwatch nicht von der Kleidung verdeckt werden. Die Snooze Time sollte gering gewählt werden, damit ein versäumtes Feedback rasch wieder empfangen werden kann. Bestätigte Dimensionen dürfen den Referenten erst nach einer längeren Zeitspanne wieder übermittelt werden, dementsprechend hoch ist die Confirmed Snooze Time zu wählen. Bei Bedarf besteht die Option die permanente Wiederholung des Vibrationsmusters zu aktivieren.

Die visuelle Darstellung einer Rückmeldung auf der Smartwatch zeigte sich als verbesserungsfähig. Für die Vortragenden erweist sich das Einprägen der gewählten Farben und deren Bedeutung im Vorhinein als empfehlenswert. Der Schriftzug in der Dimension kann nur als unterstützende Maßnahme gesehen werden.

### **Weiterführende Tests**

Neben den Testvortrag wurden mit der Smartwatch noch spezielle Tests durchgeführt.

Eine Erkenntnis bezog sich auf die Entfernung zum Smartphone, wobei außerhalb eines Radius von etwa 10 Metern die Verbindung zwischen Smartphone und Smartwatch nicht mehr garantiert war. Die vortragende Person sollte das Smartphone bei sich tragen, um dieses Risiko auszuschalten. Ebenso galt es die Laufzeit zu untersuchen, wobei bei ständig beleuchtetem Display mit etwa 5 Stunden Einsatzzeit gerechnet werden kann. Eine höhere Laufzeit darf erwartet werden, da die Anwendung in der Praxis kaum permanent aktiv sein wird.

Ein weiterer Testpunkt war der Empfang von Benachrichtigungen von anderen Anwendungen, was während des Vortrages problematisch sein könnte. Dieser kann durch die Companion-App unterdrückt werden. Dabei führen eingestellte Cards in den Context Stream ohnehin nicht dazu, dass die aktive Anwendung in den Hintergrund gestellt wird. Das Starten von

## Evaluation

anderen Anwendungen im Vollbildmodus kann und soll für einen Entwickler nicht unterdrückbar sein, was folglich nicht verhindert werden kann.

## 8 Zusammenfassung

Diese Arbeit untersucht die Potentiale der Smartwatch in Verbindung mit ARS. Einleitend wurden einige Begriffe erläutert. Danach folgte eine Schilderung der Geschichte der Smartwatch. Anhand von Informationen über den aktuellen Stand bei Smartwatches konnte ein geeignetes Modell für die Umsetzung eines Prototyps ausgewählt werden. Schlussendlich fiel die Entscheidung auf die Sony Smartwatch 3.

Als ARS stand ein quantitativer Backchannel zur Verfügung - der Backchannel von der Firma Carrot & Company GmbH. Dieser ist als Webanwendung realisiert, die Teilnehmerinnen und Teilnehmer verwenden mobile, internetfähige Geräte für ihre Eingabe. Dabei können sie einen Vortrag anhand von drei Dimensionen bewerten:

- Zufriedenheit
- Verständnis
- Geschwindigkeit des Vortrages

Nachdem das ARS und die Smartwatch feststanden, war ein nächster Schritt ein Verständnis für die Anwendungsentwicklung unter Android bzw. Android Wear zu schaffen. Dabei erwies sich die Dokumentation im Internet als sehr umfangreich und hilfreich bei den Designentscheidungen.

Im Rahmen der Implementierungsphase wurde ein passender Socket.IO Java Client für die Anbindung an den Backchannel gesucht und gefunden. Mit Einbindung der Google-Play-Service-Bibliothek konnte die Kommunikation zwischen Smartphone und Smartwatch realisiert werden. Schließlich wurde eine Anwendung für das Smartphone, die als Verbindungsstück zwischen dem Backchannel und der Smartwatch dient, und eine Anwendung für die Smartwatch zur Visualisierung des Feedbacks implementiert.

Mit dem entwickelten Prototypen wurde ein ersten Live-Test bei den Einführungstagen der TU Graz durchgeführt. Dabei stellte sich die Smartwatch als geeignetes Gerät zur Darstellung von Rückmeldungen des Backchannels heraus. Die Vortragenden sind durch die Smartwatch viel mobiler, da ein Feedback überall im Raum empfangen werden kann. Dank Vibrationssignale wird spürbar auf den Erhalt aufmerksam gemacht.

## Zusammenfassung

Während der Durchführung der Arbeit haben sich die Smartwatches weiterentwickelt. Die neu entstandenen Möglichkeiten und die Verbesserungspotentiale des Prototyps werden im abschließenden Kapitel erläutert.

## 9 Ausblick

Im Rahmen des Ausblicks werden Verbesserungspotentiale und weiterführende Gedanken, die während des Projektes aufgekommen sind, angesprochen.

Um den Vortragenden die Erkennung einer Rückmeldung noch deutlicher aufzubereiten, könnte auf Grafiken zurückgegriffen werden. Zu validieren gilt es, ob diese aufgrund des kleinen Displays der Smartwatch aussagekräftig darstellbar sind. Eine eigene Grafik je Dimension einzublenden erscheint aus aktuellem Standpunkt jedoch schwierig - die Größe könnte eine schnelle Interpretation stark einschränken. Eine andere Option wäre die Einblendung eines einzigen Avatars wie beim Backchannel [Haintz, 2013], um das intuitive Verständnis der Vortragenden Person einzubeziehen.

Des Weiteren kann die Schwere der jeweiligen Dimension von Interesse sein. Beispielsweise gilt es zu erkennen, ob der Vortrag für das Auditorium nur leicht oder viel zu schnell abgehalten wird. Denkbar wäre den gewählten Farbton anhand des Verhältnisses vom Schwellwert zum tatsächlichen Wert abzustufen und dabei immer alle Dimensionen einzublenden.

Ebenso könnten akustische Signale eingesetzt werden. Die Sony Smartwatch 3 verfügt wie viele erste Uhren mit Android Wear nicht über einen Lautsprecher, da die Sprachausgabe von Seiten des Betriebssystems noch nicht unterstützt wurde. An einer Nachrüstung wird Berichten zufolge gearbeitet<sup>46</sup>. Bei anderen Modellen wie die Apple Watch und die Samsung Gear S oder Gear S 2 existiert dieses Feature bereits. Ob akustische Signale während eines Vortrags aufgrund des Lärmpegels untergehen, müsste geklärt werden.

Die Anwendung soweit zu entwickeln, dass die Referentin und der Referent ohne Blick auf die Smartwatch die Rückmeldung interpretieren kann, wäre die optimale Lösung. Zu evaluieren wären bestimmte Vibrationssignale, die sich signifikant voneinander unterscheiden, und wie viel Übungsaufwand zur eindeutigen Erkennung notwendig wäre. Ebenso muss die Überlegung

---

<sup>46</sup> <http://www.giga.de/downloads/android-wear/news/android-wear-smartwatches-telefonieren-und-senden-von-whatsapp-nachrichten-bald-moeglich> (besucht am 21.12.2015)

## Ausblick

getroffen werden, welches Vibrationsmuster beim Unterschreiten von mehreren Dimensionen ausgelöst wird.

Im Allgemeinen verfügt die Smartwatch über die Technologie vielseitig eingesetzt zu werden. Auf ARS bezogen sind Steuerungsfunktionen, wie das Starten und Beenden von Umfragen, oder das Markieren von Freitextrückmeldungen als beantwortet, überlegenswert. Will man Vorträge im Allgemeinen unterstützen, wie das Blättern in Präsentationen, ist noch eine Vielzahl von Anwendungsszenarien vorstellbar. Auch in Punkto Bedienung existiert noch Potential. Denkbar sind Eingaben durch die durch Bewegung des Handgelenks oder über die Sprache.

Dank der Cloudsynchronisierung bei Android Wear brauchen sich die Geräte nicht mehr in Bluetooth-Reichweite befinden. Notwendige Adaptionen am Prototypen wären durchzuführen. Womöglich könnten in Zukunft mit der Smartwatch auch direkt Informationen aus dem Internet abgerufen werden, wodurch die komplette Anwendung auf die Smartwatch portierbar wäre. Bereits jetzt könnte man evaluieren, ob eine autarke Applikation mit der Samsung Gear S oder Gear S 2 dank eigenständiger SIM-Karte und dem Betriebssystem Tizen realisiert werden kann.



## Anhang

### Vergleich Android Wear Smartwatch

Modell	G Watch <sup>47</sup>	Gear Live <sup>48</sup>	G Watch R <sup>49</sup>
Hersteller	LG	Samsung	LG
Display	eckig LCD, IPS Farbdisplay 1,65 Zoll 280x280 Pixel	eckig Super AMOLED 1,63 Zoll 320x320 Pixel	rund P-OLED 1,3 Zoll 320x320 Pixel
Prozessor	Qualcomm Snapdragon 400 Quadcore 1.2GHz	1.2GHz	Qualcomm Snapdragon 400 Quadcore 1.2GHz
Akku	400 mAh	300 mAh	410 mAh
Festplatte	4 GB	4 GB	4 GB
RAM	512 MB	512 MB	512 MB
Bluetooth	Ja	Ja	Ja
WLAN	?	?	?
NFC	Nein	Nein	Nein
GPS	Nein	Nein	Nein
Sensoren	Beschleunigungsmesser Kompass Gyroskop	Beschleunigungsmesser Kompass Gyroskop Pulsmesser	Umgebungslichtsensor Beschleunigungssensor Kompass Gyroskop Pulsmesser Barometer
Mikrofon	Ja	Ja	Ja
Vibration	Ja	Ja	Ja

<sup>47</sup> <http://www.lg.com/de/wearables/lg-G-Watch> (besucht am 12.01.2015)

<sup>48</sup> <http://www.samsung.com/at/consumer/mobile-devices/wearables/gear/SM-R3820ZKAATO> (besucht am 12.01.2015)

<sup>49</sup> <http://www.lg.com/de/wearables/lg-G-Watch-R> (besucht am 12.01.2015)

## Vergleich Android Wear Smartwatch

Modell	SmartWatch 3 <sup>50</sup>	Moto 360 (1. Gen.) <sup>51</sup>	ZenWatch <sup>52</sup>
Hersteller	Sony	Motorola	Asus
Display	eckig Transflekatives TFT LCD 1,6 Zoll 320x320 Pixel	rund IPS LCD Backlit 1,56 Zoll 320x290 Pixel	eckig AMOLED 1,63 Zoll 320x320 Pixel
Prozessor	Quad ARM A7, 1.2 GHz	TI OMAP3630 Singlecore 1GHz	Qualcomm Snapdragon 400 Quadcore 1.2GHz
Akku	420 mAh	320 mAh	369 mAh
Festplatte	4 GB	4 GB	4 GB
RAM	512 MB	512 MB	512 MB
Bluetooth	Ja	Ja	Ja
WLAN	Ja	?	?
NFC	Ja	Nein	Nein
GPS	Ja	Nein	Nein
Sensoren	Umgebungslichtsensor Beschleunigungssensor Kompass Gyroskop	Umgebungslichtsensor Beschleunigungssensor Gyroskop Pulsmesser	Beschleunigungssensor Kompass Gyroskop Pulsmesser
Mikrofon	Ja	Ja	Ja
Vibration	Ja	Ja	Ja

<sup>50</sup> <http://www.sonymobile.com/at/products/smartwear/smartwatch-3-swr50> (besucht am 12.01.2015)

<sup>51</sup> <https://www.motorola.com/us/products/moto-360> (besucht am 12.01.2015)

<sup>52</sup> [http://www.asus.com/ZenWatch/ASUS\\_ZenWatch\\_WI500Q](http://www.asus.com/ZenWatch/ASUS_ZenWatch_WI500Q) (besucht am 12.01.2015)

## Abkürzungsverzeichnis

<b>ARS</b>	Audience Response System
<b>BYOD</b>	Bring your own device
<b>MC</b>	Mobile Computing
<b>MSL</b>	Mobile Seamless Learning
<b>SL</b>	Seamless Learning
<b>UC</b>	Ubiquitous Computing

## Abkürzungsverzeichnis

## Literaturverzeichnis

- [Becker & Pant, 2015] Becker, A. & Pant, B. (2015). Android 5 - Programmieren für Smartphones und Tablets, Auflage: 4, ISBN: 978-3864902604, Seiten 182-192.
- [Bieber, Haescher & Vahl, 2013] Bieber, G., Haescher, M. & Vahl, M. (2013). Sensor Requirements for Activity Recognition on Smart Watches. In *Proceedings of the 6th International Conference on Pervasive Technologies Related to Assistive Environments*, ISBN: 978-1-4503-1973-7, Artikel Nr.: 67.
- [Brodersen, 2015] Brodersen, B. (2015). WLAN-Sync für Smartwatches und bald auch Audio-Support fürs Telefonieren. URL: <http://www.areamobile.de/news/32515-android-wear-update-5-1-wlan-sync-fuer-smartwatches-und-bald-auch-audio-support-fuers-telefonieren> (besucht am 8.12.2015).
- [Caldwell, 2007] Caldwell, J. (2007). Clickers in the Large Classroom: Current Research and Best-Practice Tips. *CBE Life Sciences Education*, 6(1): 9-20.
- [Cap, Delfs & Vetterick, 2015] Cap, C., Delfs, C. & Vetterick, J. (2015). Tweedback goes Smart Watch - Why Classroom Response Systems Need Smart Watch User Interfaces. In *Workshop Proceedings of the 11th International Conference on Intelligent Environments*, ISBN: 978-1-61499-529-6, Seiten 273-280.
- [Cecchinato, Cox & Bird, 2015] Cecchinato, M., Cox, A. & Bird, J. (2015). Smartwatches: the Good, the Bad and the Ugly?. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, ISBN: 978-1-4503-3146-3, Seiten 2133-2138.
- [Chan et al., 2006] Chan, T.-W., Roschelle, J., Hsi, S., Kinshuk, Sharples, M. et al. (2006). One-to-one Technology Enhanced Learning: An Opportunity for Global Research Collaboration. *Research and Practice in Technology Enhanced Learning*, 1(1): 3-29.

## Literaturverzeichnis

- [Chen et al., 2010] Chen, W., Seow, P., So, H.-J., Toh, Y. & Looi, C.-K. (2010). Connecting learning spaces using mobile technology. *Education Technology*, 50(5): 45-50.
- [Charlton, 2013] Charlton, A. (2013). From IBM to Microsoft – A Brief History of The Smartwatch. URL: <http://www.ibtimes.co.uk/smartwatch-history-apple-iwatch-samsung-galaxy-gear-503752> (besucht am 29.12.2015).
- [Dey & Abowd, 2000] Dey, A. & Abowd, G. (2000). Towards a Better Understanding of Context and Context-Awareness. *CHI 2000 Workshop on the What, Who, Where, When, and How of Context-Awareness*.
- [Dey et al., 1999] Dey, A., Salber, D., Abowd, G. & Futakawa, M. (1999). The Conference Assistant: Combining Context-Awareness with Wearable Computing. In *Proceedings of the 3rd IEEE International Symposium on Wearable Computers*, ISBN: 0-7695-0428-0, Seiten 21-28.
- [Disterer & Kleiner, 2013] Disterer, G. & Kleiner C. (2013). BYOD Bring Your Own Device. *Procedia Technology*, 9:43-53.
- [Ebner, 2015] Ebner, M. (2015). Google Glass zur Verbesserung der Interaktion bei Massenvorlesungen. Masterarbeit ,Technische Universität Graz.
- [Ebner, Nagler & Schön, 2012] Ebner, M., Nagler, W. & Schön, M. (2012). Have They Changed? Five Years of Survey on Academic Net-Generation. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2012*, Seiten 343-353.
- [Ellis, 2015] Ellis, T. (2015). Apple Watch? Whatever. Reviewing the Timex Datalink, the world's first smartwatch. URL: <http://www.geekwire.com/2015/reviewing-worlds-first-smartwatch-timex-datalink> (besucht am 29.12.2015).
- [Fies & Marshall, 2006] Fies, C. & Marshall, J. (2006). Classroom Response Systems: A Review of the Literature. *Journal of Science Education and technology*, 15(1): 101-109.

- [Frohberg, Göth & Schwabe, 2009] Frohberg, D., Göth, C. & Schwabe, G. (2009). Mobile Learning projects – a critical analysis of the state of the art. *Journal of Computer Assisted Learning*, 25(4): 307-331.
- [Future Classroom Lab, 2015] Future Classroom Lab (2015). BYOD - Bring Your Own Device - A guide for school leaders. *Designing the Future Classroom, Issue 3, Oktober 2015*. URL: <http://fcl.eun.org/byod> (besucht am 04.01.2016).
- [Hahn, 2015] Hahn, M. (2015). Android Wear Docs. URL: <http://android-wear-docs.readthedocs.org/en/latest/index.html> (besucht am 08.12.2015).
- [Haintz, 2013] Haintz, C. (2013). Quantitative Digital Backchannel: Developing a Web-Based Audience Response System for Measuring Audience Perception in Large Lectures. Masterarbeit, Technische Universität Graz.
- [Haintz, Pichler & Ebner, 2014] Haintz, C., Pichler, K. & Ebner, M. (2014). Developing a Web-Based Question-Driven Audience Response System Supporting BYOD. *Journal of Universal Computer Science*, 20(1): 39-56.
- [Judson & Sawada, 2002] Judson, E. & Sawada, D. (2002). Learning from Past and Present: Electronic Response Systems in College Lecture Halls. *Journal of Computers in Mathematics and Science Teaching*, 21(2): 167-181.
- [Kay & LeSage, 2009] Kay, R. & LeSage, A. (2009). Examining the benefits and challenges of using audience response systems: A review of the literature. *Computers & Education*, 53(3): 819-827.
- [Kuh, 1996] Kuh, G. (1996). Guiding Principles for Creating Seamless Learning Environments for Undergraduates. *Journal of College Student Development*, 37(2), 135-148.
- [Looi et al., 2010] Looi, C.-K., Seow, P., Zhang, B., So, H.-J., Chen W. & Wong, L.-H. (2010). Leveraging mobile technology for sustainable seamless learning: A research agenda. *British Journal of Educational Technology*, 41(2): 154-169.

- [Marshall, 2015] Marshall, G. (2015). Before Apple Watch: the timely history of the smartwatch, URL: <http://www.techradar.com/news/wearables/before-iwatch-the-timely-history-of-the-smartwatch-1176685> (besucht am 29.12.2015).
- [MacGibbon, 2012] MacGibbon A. (2012). Smarter use of home devices. URL: <http://www.smh.com.au/technology/technology-news/smarter-use-of-home-devices-20120527-1zcry.html> (besucht am 10.01.2015).
- [Milrad et al., 2013] Milrad, M., Wong, L.-H., Sharples, M., Hwang, G.-J., Looi, C.-K. & Ogata, H. (2013). Seamless Learning: An International Perspective on Next Generation Technology Enhanced Learning. In *Handbook of Mobile Learning*, ISBN: 978-0-415-50369-3, Seiten 95-108.
- [Müller & Krummennacher, 2012] Müller, H. & Krummenacher, J. (2012). Funktionen und Modelle der selektiven Aufmerksamkeit. In *Kognitive Neurowissenschaften*, ISBN: 978-3642255267, Seiten 307-321.
- [Narayanaswami & Raghunath, 2000] Narayanaswami, C. & Raghunath, M.T. (2000). Application Design for a Smart Watch with a High Resolution Display. In *Proceedings of the 4th IEEE International Symposium on Wearable Computers*, ISBN:0-7695-0795-6, Seiten 7-14.
- [Niehaves et al., 2012] Niehaves, B., Köffer, S., Ortbach, K. & Katschewitz, S. (2012). Towards an IT consumerization theory: A theory and practice review. Working Papers, ERCIS - European Research Center for Information Systems, No. 13.
- [Noel, Stover & McNutt, 2015] Noel, D., Stover, S. & McNutt, M. (2015). Student perceptions of engagement using mobile-based polling as an audience response system: Implications for leadership studies. *Journal of Leadership Education*, 14(3): 53-70.
- [NSW Department of Education and Communities, 2013] NSW Department of Education and Communities (2013). Bring Your Own Device(BYOD) in Schools, 2013 Literature Review. URL: [https://www.det.nsw.edu.au/policies/technology/computers/mobile-device/BYOD\\_2013\\_Literature\\_Review.pdf](https://www.det.nsw.edu.au/policies/technology/computers/mobile-device/BYOD_2013_Literature_Review.pdf) (besucht am 04.01.2016).



[Pelegrin, 2014] Pelegrin, W. (2014). Android Wear: Google reveals new watch details at I/O. URL: <http://www.digitaltrends.com/mobile/android-wear-os-news-release-features> (besucht am 29.12.2015).

[Pichler, 2013] Pichler, K. (2013). Relevant Information and Information Visualizations for Lecturers in Web-Based Audience Response Systems. Masterarbeit, Technische Universität Graz.

[Quinn, 2000] Quinn, C. (2000). mLearning: Mobile, Wireless, In-Your-Pocket Learning. URL: <http://www.linezine.com/2.1/features/cqmmwiyp.htm> (besucht am 09.01.2016)

[Rhodes, 1997] Rhodes, B. (1997). The Wearable Remembrance Agent: A system for augmented memory, In *Proceedings of the 1st International Symposium on Wearable Computers*, ISBN: 0-8186-8192-6, Seiten 123-128.

[Saha & Mukherjee, 2003] Saha D. & Mukherjee A. (2003). Pervasive Computing: A Paradigm for the 21st Century, *Computer*, 36(3): 25-31.

[Satyanarayanan, 2001] Satyanarayanan, M. (2001). Pervasive Computing: Vision and Challenges, *Communication, IEEE*, 8(4): 10-17.

[Seow et al., 2008] Seow, P., Zahng, B., So, H.-J., Looi, C.-K. & Chen, W. (2008). Towards a framework for seamless learning environments. In *Proceedings of the 8th international conference on International conference for the learning sciences - Volume 2*, Seiten 327-334.

[Sharples, Taylor & Vavoula, 2007] Sharples, M., Taylor, J. & Vavoula, G. (2007). A Theory of Learning for the Mobile Age. In *The Sage Handbook of E-learning Research*, ISBN: 9781412919388, Seiten 221-247.

[Sharples et al., 2009] Sharples, M., Arnedillo-Sanchez, I., Milrad, M. & Vavoula, G. (2009). Mobile Learning - Small devices, Big issues. In *Technology-Enhanced Learning: Principles and Products*, ISBN: 978-1-4020-9826-0, Seiten 233-249.

[Simpson & Oliver, 2007] Simpson, V. & Oliver, M. (2007). Electronic voting systems for lectures then and now: A comparison of research and practice. *Australasian Journal of Educational Technology*, 23(2): 187-208.

[Smith, 2013] Smith, A. (2013). Smartphone Ownership 2013. URL: <http://www.pewinternet.org/2013/06/05/smartphone-ownership-2013> (besucht am 29.12.2015).

[So, Kim & Looi, 2008] So, H.-J., Kim, I. & Looi, C.-K. (2008). Seamless Mobile Learning: Possibilities and Challenges Arising from the Singapore Experience. *Educational Technology International*, 9(2): 97-121.

[Song, 2014] Song, Y. (2014). "Bring Your Own Device (BYOD)" for seamless science inquiry in a primary school. *Computers & Education*, 74: 50-60.

[Specht, Ebner & Löcker, 2013] Specht, M., Ebner, M. & Löcker, C. (2013). Mobiles und ubiquitäres Lernen - Technologien und didaktische Aspekte. In Lehrbuch für Lernen und Lehren mit Technologien (2013) by Marcus Specht, Martin Ebner, Clemens Löcker edited by Sandra Schön, Martin Ebner

[Starner, 2014] Starner, T. (2014). How Wearables Worked their Way into the Mainstream. *IEEE Pervasive Computing*, 13(4): 10-15.

[Stein, 2014] Stein, S. (2014). Samsung Gear S review: The smartwatch that's also a smartphone. URL: <http://www.cnet.com/products/samsung-gear-s> (besucht am 29.12.2015).

[Vetterick, Garbe & Cap, 2013] Vetterick, J., Garbe, M. & Cap, C. (2013). Tweedback: A Live Feedback System for Large Audiences. In *Proceedings of the 5th International Conference on Computer Supported Education*, ISBN: 978-989-8565-53-2, Seiten 194-198.

[Vetterick et al., 2014] Vetterick, J., Garbe, M., Dähn, A. & Cap, C. (2014). Classroom Response Systems in the Wild: Technical and Non-Technical Observations. *International Journal of Interactive Mobile Technologies*, 8(1): 21-25.

[Weiser, 1991] Weiser, M. (1991). The Computer for the 21st Century. *Scientific American*, 265(3): 94-104.

[Wong & Looi, 2011] Wong, L.-H. & Looi C.-K. (2011). What seems do we remove in mobile assisted seamless learning? A critical review of the literature. *Computers and Education*, 57(4): 2364-2381.

[Wong, 2012] Wong, L.-H. (2012). A learner-centric view of mobile seamless learning. *British Journal of Educational Technology*, 43(1): E19-E23.

[Wong, 2014] Wong, L.-H. (2014). A Brief History of Mobile Seamless Learning. In *Seamless Learning in the Age of Mobile Connectivity*, ISBN: 978-981-287-113-8, Seiten 3-40.

[Yardi, 2006] Yardi, S. (2006). The Role of the Backchannel in Collaborative Learning Environments. In *Proceedings of the 7th International Conference on Learning Sciences*, ISBN: 0-8058-6174-2, Seiten 852–858.

[Youssef et al., 2005] Youssef, A., Krumm, J., Cermak, G. & Horvitz E. (2005). Computing Location from Ambient FM Radio Signals. *IEEE Wireless Communications and Networking Conference*, 2(1): 824-829.