



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

# Masterarbeit

Ivan Demin

**Entwicklung einer Plattform für Second Screen Experimente**

*Fakultät Technik und Informatik  
Studiendepartment Informatik*

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

Ivan Demin

## **Entwicklung einer Plattform für Second Screen Experimente**

Masterarbeit eingereicht im Rahmen der Masterprüfung

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

Betreuender Prüfer: Prof. Dr. Kai von Luck  
Zweitgutachter: Prof. Dr. Bettina Buth

Eingereicht am: 30. November 2015

**Ivan Demin**

**Thema der Arbeit**

Entwicklung einer Plattform für Second Screen Experimente

**Stichworte**

Second Screen, Untersuchungsplattform, Text Mining, Text Summarization, Feature Extraction

**Kurzzusammenfassung**

Second Screen ist seit Jahren ein fester Bestandteil der Medienlandschaft. Diese Arbeit befasst sich mit der Entwicklung einer Untersuchungsplattform für Second Screen Experimente. Es präsentiert einen Weg, auf Basis von Feature Extraction aus Texten passende Informationsabfragen zu kuratierten Archiven zu generieren. Um den Benutzer nicht zu überfordern und ihm kompakte Informationsblöcke zu präsentieren, werden Texte anhand von Textzusammenfassungsalgorithmen komprimiert. Die Validierung der Untersuchungsplattform basiert auf der Frage, inwieweit Second Screen die Informationsaufnahme der Nutzer beeinflusst. Dazu wurde ein Experiment mit Probanden im Living Place Hamburg durchgeführt.

**Ivan Demin**

**Title of the paper**

Development of an Platform for Second Screen Experiments

**Keywords**

second screen, investigation platform, text mining, automatic text summarization, feature extraction

**Abstract**

Second Screen has been an integral part of the media landscape. This work is concerned with the development of an investigation platform for Second Screen experiments. It presents a way to generate pertinent information queries to curated archives based on feature extraction from texts. In order not to overwhelm the user and to present him compact blocks of information, texts are compressed using text summarization algorithms. The validation of the investigation platform is based on the question, in how far second screen affects the information receiving of users. For this purpose, an experiment was conducted with volunteers at the Living Place Hamburg.

# Inhaltsverzeichnis

<b>1. Einführung</b>	<b>1</b>
1.1. Zielsetzung . . . . .	2
1.2. Gliederung . . . . .	3
<b>2. Grundlagen</b>	<b>4</b>
2.1. Chromecast . . . . .	4
2.2. Dexter . . . . .	5
2.2.1. Arbeitsweise von Dexter . . . . .	5
2.2.2. Architektur von Dexter . . . . .	6
2.3. Sumy . . . . .	7
2.3.1. Extraktionsalgorithmus nach Luhn . . . . .	7
2.3.2. Edmundsons heuristisches Verfahren . . . . .	8
2.3.3. Enhanced Latent Semantic Analysis - ELSA . . . . .	10
2.3.4. LexRank . . . . .	11
2.3.5. TextRank . . . . .	12
2.4. Struktur und Ablauf von wissenschaftlichen Untersuchungen . . . . .	13
2.4.1. Versuchsvorbereitung . . . . .	13
2.4.2. Versuchsdurchführung . . . . .	14
2.4.3. Auswertung der Daten . . . . .	15
<b>3. Analyse</b>	<b>16</b>
3.1. Psychische Ebene . . . . .	16
3.1.1. Auswirkungen von Arbeitsunterbrechungen . . . . .	16
3.1.2. Arbeitsorganisation . . . . .	18
3.2. Technische Ebene . . . . .	19
3.3. Darstellende Ebene . . . . .	21
3.3.1. Social TV . . . . .	22
3.3.2. Fernbedienung . . . . .	22
3.4. Kontextuelle Ebene . . . . .	24
3.4.1. Erlernung der Wikifizierung . . . . .	24
3.4.2. Befüllung des Second Screens basierend auf der semantischen Verlinkung von Untertiteln . . . . .	26
3.5. Personalisierungs-Ebene . . . . .	29
3.6. Anforderungen an SecondCast . . . . .	31
3.6.1. Rollen . . . . .	31

3.6.2.	Funktionale Anforderungen . . . . .	32
3.6.3.	Technische Anforderungen . . . . .	33
3.7.	Fazit . . . . .	33
<b>4.</b>	<b>Design</b> . . . . .	<b>35</b>
4.1.	Kontextsicht . . . . .	35
4.2.	Bausteinsicht . . . . .	36
4.2.1.	SecondCast-Client . . . . .	37
4.2.2.	SecondCast-Multimedia-Library . . . . .	38
4.2.3.	SecondCast-Core . . . . .	39
4.2.4.	SecondCast-Summarizer . . . . .	40
4.2.5.	SecondCast-Context . . . . .	40
4.3.	Kommunikation der Architekturelemente . . . . .	41
4.4.	TI-Architektur . . . . .	42
4.5.	Konzeption der grafischen Benutzeroberfläche . . . . .	43
4.5.1.	User Story „Dokumentation mit Zusatzinformationen aufrufen“ . . . . .	44
4.5.2.	User Story „Zusatzinformationen verwalten“ . . . . .	45
4.6.	Fazit . . . . .	45
<b>5.</b>	<b>Realisierung</b> . . . . .	<b>47</b>
5.1.	SecondCast-Client . . . . .	47
5.1.1.	User Story „Dokumentation abspielen“ . . . . .	48
5.1.2.	Tracking von Nutzeraktionen . . . . .	49
5.2.	SecondCast-Server . . . . .	50
5.2.1.	SecondCast-Core . . . . .	51
5.2.2.	SecondCast-Context . . . . .	52
5.2.3.	SecondCast-Summarizer . . . . .	54
5.3.	Fazit . . . . .	56
<b>6.</b>	<b>Evaluierung</b> . . . . .	<b>58</b>
6.1.	Szenario . . . . .	58
6.1.1.	Versuchsdurchführung . . . . .	58
6.1.2.	Ethische Rahmenbedingungen . . . . .	60
6.1.3.	Technische Rahmenbedingungen . . . . .	60
6.2.	Ergebnisse der Untersuchung . . . . .	63
6.2.1.	Teilnehmerdaten . . . . .	63
6.2.2.	Wissenstest . . . . .	65
6.2.3.	Usability-Untersuchung . . . . .	66
6.3.	Auswertung der Untersuchungsergebnisse . . . . .	68
6.3.1.	Wissenstest . . . . .	68
6.3.2.	Fachliche und technische Fragen . . . . .	69
6.3.3.	Fehler und Verbesserung . . . . .	70
6.4.	Fazit . . . . .	72

<b>7. Schlussbetrachtung</b>	<b>73</b>
7.1. Zusammenfassung . . . . .	73
7.2. Ausblick . . . . .	74
<b>A. Anhang</b>	<b>84</b>
A.1. SecondCast-Summarizer . . . . .	84
A.2. Untersuchungsbögen . . . . .	85

# Tabellenverzeichnis

3.1.	Performance vom Algorithmus für die Linkerkennung in Prozent. <b>Milne und Witten (2008)</b> . . . . .	26
3.2.	Zusammenfassung der semantischen Verlinkungs-Ergebnisse aus der Abschlussuntersuchung. <b>Odijk u. a. (2013)</b> . . . . .	28
5.1.	Gesamtergebnis der Performance von Dexter anhand der untersuchten Dokumentationsbeschreibungen. . . . .	54
5.2.	Gesamteinschätzung der Verständlichkeit von den zusammengefassten Texten auf Basis der angewendeten Verfahren aus der Sumy Bibliothek. . . . .	56
6.1.	Prozentuale Zustimmung der Versuchsgruppen zu den Aussagen bezüglich der Second-Screen-Nutzung. . . . .	65
6.2.	Ergebnisse von den fachlichen Fragen aus der Usability-Untersuchung. . . . .	67
6.3.	Ergebnisse von den technischen Fragen aus der Usability-Untersuchung. . . . .	67
6.4.	Ausgewertete Ergebnisse des Wissenstests unter Einbeziehung beider Versuchsgruppen. . . . .	69
6.5.	Gesamtergebnis von den fachlichen Fragen aus der Usability-Untersuchung. . . . .	70
6.6.	Gesamtergebnis von den technischen Fragen aus der Usability-Untersuchung. . . . .	70
6.7.	Verbesserungsvorschläge der Probanden mit Einteilung in Kategorien. . . . .	71

# Abbildungsverzeichnis

2.1.	Lebenszyklus einer Chromecast Sender Applikation. . . . .	5
2.2.	Architektur von Dexter. <b>Ceccarelli und Lucchese (2015)</b> . . . . .	7
3.1.	Der Unterbrechungs und Fortführungs-Prozess, mit der Einbezugnahme einer Hauptaufgabe durch eine unterbrechende Aufgabe. <b>Trafton u. a. (2003)</b> . . . . .	17
3.2.	Architekturübersicht der <i>Yahoo! Connected TV</i> Plattform. <b>Cortez u. a. (2012)</b> . . . . .	20
3.3.	DNS-DS/Zeroconf übertragene Verbindungsinformationen. <b>Cortez u. a. (2012)</b> . . . . .	20
3.4.	Cloud-basierter Webdienst für Verbindungsinformationen. <b>Cortez u. a. (2012)</b> . . . . .	21
3.5.	Manuelle Eingabe der IP-Adresse vom Nutzer. <b>Cortez u. a. (2012)</b> . . . . .	21
3.6.	Playful Probing Paket mit Kreativitätskarten, Fragebögen, Videokamera und Spielkarten. <b>Bernhaupt u. a. (2012)</b> . . . . .	23
3.7.	Ausschnitt aus dem gebildeten Kontextgraphen. <b>Odiijk u. a. (2013)</b> . . . . .	28
4.1.	Diagramm zur Kontextabgrenzung. . . . .	36
4.2.	Gesamtansicht von der Second Screen Untersuchungsplattform SecondCast als Komponentendiagramm. . . . .	37
4.3.	Komponentendiagramm der möglichen beteiligten Komponenten von SecondCast- Client. . . . .	38
4.4.	Klassendiagramm des Datenbankschemas von SecondCast-Multimedia-Library. . . . .	39
4.5.	Komponentendiagramm von SecondCast-Core. . . . .	40
4.6.	Komponentendiagramm von SecondCast-Summarizer. . . . .	40
4.7.	Komponentendiagramm von SecondCast-Context. . . . .	41
4.8.	Kommunikation der Architekturelemente für die Anzeige einer Dokumentation mit Bereitstellung der Zusatzinformationen auf dem Second Screen. . . . .	42
4.9.	TI-Architektur von SecondCast als Verteilungsdiagramm . . . . .	43
4.10.	Storyboard für den Hauptanwendungsfall der Anzeige von Dokumentationen auf dem First Screen, sowie der Bereitstellung von Zusatzinformationen auf dem Second Screen . . . . .	44
4.11.	Zusatzinformationsansicht von SecondCast-Core. . . . .	45
5.1.	Umgesetzte User Story für die Anzeige von Dokumentationen auf dem First Screen, sowie der Bereitstellung von Zusatzinformationen auf dem Second Screen. . . . .	49
5.2.	Ausschnitt aus dem Google Analytics Dashboard für die App-Übersicht. . . . .	50
5.3.	Ansicht der umgesetzten Entität Information für das Anlegen und Verwalten von Datensätzen. . . . .	51

5.4.	Visuelle Darstellung der REST-Schnittstelle von SecondCast-Core. . . . .	52
5.5.	Dexter-Analyser für die Evaluierung der Messergebnisse anhand der Bildung von Werten: <i>Recall</i> , <i>Precision</i> und <i>F1 score</i> . . . . .	53
6.1.	Gesamtansicht des Living Place Hamburg - Smart Home Forschungsumgebung.	61
6.2.	Gesamtansicht des Versuchsaufbaus im Wohnbereich vom Living Place Hamburg mit vier eingezeichneten Teilnehmerplätzen, einem Platz für den Prüfer und First Screen. . . . .	62
6.3.	Wohnzimmer des Living Place Hamburg bestehend aus First Screen, sowie Sitzgelegenheiten wie Sofa und Sesseln. . . . .	62
6.4.	Demografische Informationen der Teilnehmer aus dem Experiment. . . . .	63
6.5.	Medienkonsum der Teilnehmer aus der Untersuchung. . . . .	64
6.6.	Resultate vom Wissenstest der beiden Versuchsgruppen, als gruppiertes Balkendiagramm mit Anzahl der richtig beantworteten Fragen. . . . .	65

# Listings

A.1. Python Code vom SecondCast-Summarizer. . . . .	84
---	----

# 1. Einführung

Second Screen hat sich in den vergangenen Jahren zu einem festen Bestandteil der Medienlandschaft entwickelt. Es handelt sich dabei in erster Linie um die parallele Nutzung eines internetfähigen Mediengerätes wie Smartphone, Tablet oder Laptops zum angezeigten Mediengeschehen auf dem Fernsehgerät. In diesem Fall stellen die Mediengeräte den Second Screen und das Fernsehgerät den First Screen dar. Außerdem können neben Informationen auch Nutzerinteraktionen in Form des Meinungsaustausches über die Sozialen Medien wie Twitter oder Facebook erfolgen. Nach einer repräsentativen Onlinestudie in der Bundesrepublik aus dem Jahr 2014 haben 39 Prozent der Befragten mindestens wöchentlich Fernsehen und Internet parallel benutzt (vgl. [Busemann und Tippelt, 2014](#)). Im Vergleich dazu sind die Nutzungszahlen im amerikanischen Umfeld seit Jahren noch höher (vgl. [Nielsen, 2014](#)).

Diese Thematik ist nicht neu. Hierzu gab es bereits in den neunziger Jahren Überlegungen zu der Verwendung von persönlichen digitalen Assistenten (PDA) neben dem Fernsehprogramm (vgl. [Robertson u. a., 1996](#)). Auf Basis von zusätzlichen Funktionalitäten wie der Einblendung von Karten auf den PDAs, sollte das Fernseherlebnis erweitert werden. Hierbei sollten die Zusatzinformationen auf dem PDA mit dem Fernsehprogramm thematisch Verknüpft werden. Durch den Siegeszug der Smartphones und Tablets rückte das Anwendungsfeld weiter in den Vordergrund.

Second Screen eröffnet neue Möglichkeiten im Bereich Digital Storytelling (siehe z.B. [Woletz, 2008](#)) mit denen die Art, wie Geschichten erzählt werden, verändert wird. So geschehen in dem niederländischen Film „App“, in dem die Filmemacher passend zu dem Film eine Anwendung für Smartphones anbieten (vgl. [Ebner, 2014](#)). Die mobile Applikation (App) muss zu Anfang des Films aktiviert werden. Sie bietet passend zu der auf dem Fernsehen angezeigten Handlung relevante Inhalte an. Im Bereich Gamification (siehe z.B. [Deterding u. a., 2011](#)) nutzt die Videospielekonsole Wii-U mit ihrem Wii-U-Gamepad das Second Screen Prinzip aus, um Spielern Zusatzinformationen in Form von z.B. Karten zum Spielgeschehen zu präsentieren. Des Weiteren bietet Second Screen neue Möglichkeiten im Werbebereich. Hierfür nutzt beispielsweise das

Medienunternehmen ProSiebenSat.1 Media SE ihre App 7TV<sup>1</sup> um neben zusätzlichen Inhalten zu Fernsehformaten auch Werbung synchronisiert auf dem Mediengerät parallel anzeigen zu können. Dies waren nur einige Beispiele aus Bereichen, wo schon heute das Second Screen Prinzip eingesetzt wird.

### 1.1. Zielsetzung

Trotz des Erfolgs von Second Screen gibt es immer noch Problemstellungen, die das Angebot schmälern. Dazu gehört, dass Inhalte noch zum Großteil von menschlicher Hand rausgesucht und eingepflegt werden müssen. Durch die Mehrbelastung der Redakteure kommt es zu erhöhten Kosten. Wodurch für viele Medien ein Second Screen Angebot unwirtschaftlich erscheint. Einen Weg, Informationen passend zum Geschehen aus kuratierten Archiven zu generieren, soll diese Arbeit aufzeigen.

Außerdem sollen für den Forschungsbereich Second Screen Problemstellungen bei der Entwicklung von mobilen Applikationen aufgezeigt werden. Dazu sollen Erkenntnisse aus Nutzersicht und technischer Sicht für die Konzeption, Entwicklung und Inbetriebnahme von Second Screen Anwendungen gesammelt werden. Der Forschungsbereich Second Screen kann in die folgenden drei Unterbereiche aufgeteilt werden: Technische Synchronisation zwischen zwei Mediengeräten, Ermöglichung der beiläufigen Bedienbarkeit eines Second Screens und semantische Verbindung eines Textkorpuses zum First Screen. Zurzeit existiert keine Gesamtplattform die alle drei Bereiche umfasst. Daher soll das zu entwickelnde System diese Bereiche adressieren.

Der Einfluss von Second Screen auf die Nutzer ist noch nicht eindeutig erforscht. Zwar gibt es einige Studien, warum Menschen soziale Medien wie Twitter als Second Screen nutzen und wie ihr Konsumverhalten vom Format bestimmt wird (vgl. [Renken u. a., 2014](#); [Fessler, 2014](#)), jedoch nicht wie z.B. ihre Informationsaufnahme durch den Einsatz von Second Screen Anwendungen beeinflusst wird. Diese Forschungslücke soll durch die Arbeit gemindert werden. Zudem soll eine Plattform für die Forscher im Second Screen Bereich geschaffen werden. In dieser Arbeit soll die Plattform SecondCast für Second Screen Experimente konzipiert und entwickelt werden. Das zu entwickelnde System soll bezüglich folgender Hypothese verifiziert werden:

*„Es ist möglich mit Hilfe von Second Screen die Informationsaufnahme mit dem First Screen zu intensivieren.“*

---

<sup>1</sup>7TV - [www.7tv.de](http://www.7tv.de); letzter Zugriff: 29.10.2015

Besonders für den öffentlich-rechtlichen Rundfunk, bestehend aus ARD, NDR und weiteren Fernsehsendern, ist diese Hypothese interessant. Demnach haben sie durch die Rechtsprechung des Bundesverfassungsgerichtes (BVerfG) zur Rundfunkfreiheit nach Artikel 5 Abs. 1 S. 2 GG einen Bildungsauftrag. Hierfür müssen für die Allgemeinheit geeignete Bildungsangebote erarbeitet und bereitgestellt werden. Die durch die empirische Untersuchung gewonnenen Erkenntnisse stellen dafür eine Anregung für den öffentlich-rechtlichen Rundfunk dar, das Second Screen Angebot zu erweitern. Ferner soll die Hypothese eine Grundlage für weitere Experimente im Second Screen Bereich, anhand der zu entwickelnden Plattform, schaffen. Durch eine transparente Herangehensweise bei der Versuchsvorbereitung und Durchführung des Experimentes soll die Nachvollziehbarkeit der Untersuchung durch Dritte ermöglicht werden. Die Überprüfbarkeit wird zu einer Nachstellbarkeit des experimentellen Aufbaus führen. Das führt dazu, dass der Versuch kopierbar und auf weitere Formate wie z.B. Filme übertragbar ist. Dies wiederum bringt nachfolgenden Erkenntnisgewinn in anderen Bereichen, wie beispielsweise Digital Storytelling oder Gamification.

### 1.2. Gliederung

Die Arbeit gliedert sich in sieben Kapitel auf. Im Kapitel 2 werden die Grundlagen für das spätere Verständnis dieser Forschungsarbeit gelegt. Das Kapitel 3 befasst sich mit der Analyse von Problemen bei der Entwicklung von Second Screen Anwendungen. Anhand von möglichen Lösungsansätzen werden hier Anforderungen an das SecondCast System formuliert. Im Kapitel 4 wird das Design von der zu entwickelnden Untersuchungsplattform dargestellt. Die erfolgreiche Implementierung wird in Kapitel 5 vorgestellt. Die Evaluation aus der technischen, konzeptionellen und kontextsensitiven Sicht wird in Kapitel 6 beschrieben. Zum Abschluss wird in Kapitel 7 eine Zusammenfassung der Ergebnisse, sowie einen Ausblick auf weitere Fragestellungen gegeben.

## 2. Grundlagen

Dieses Kapitel schafft die Grundlagen für das Verständnis bei der Entwicklung von SecondCast. Am Anfang wird im Abschnitt 2.1 auf die Funktionsweise des Streaming-Media-Adapters Chromecast eingegangen, mit dem es möglich ist Videos auf einem Fernseher zu streamen. Der nachfolgende Abschnitt 2.2 beschreibt den Aufbau des Open-Source-Frameworks Dexter, das für Extraktion von Schlagwörtern aus Texten eingesetzt wird. Danach werden im Abschnitt 2.3 die implementierten Textzusammenfassungsverfahren der Python-Bibliothek Sumy vorgestellt. Zum Ende des Kapitels wird im Abschnitt 2.4 auf die Struktur und den Ablauf von wissenschaftlichen Untersuchungen näher eingegangen.

### 2.1. Chromecast

Chromecast<sup>2</sup> stellt einen von Google entwickelten Streaming-Media-Adapter dar. Mit dem Gerät ist es z.B. möglich einen Videostream auf einem Fernseher anzeigen zu lassen. Des Weiteren erlaubt es die Steuerung vom Fernseher wie beispielsweise die Veränderung der Lautstärke oder das Pausieren von abgespielten Videos.

Um die angebotene Chromecast-API zu verwenden, muss verstanden werden wie eine Anwendung in dem Kontext funktioniert. Die Abbildung 2.1 beschreibt den Lebenszyklus einer Google Chromecast Sender Applikation von der Initialisierung der Anwendung bis hin zur Trennung vom Chromecast-Gerät. Es fängt mit dem Laden der Anwendung an. Als nächstes wird in L2 die API und die damit gebrauchten Listener: SessionJoinedListener und ReceiverListener initialisiert. Hierbei werden alle vermeintlichen Chromecast-Geräte in der näheren Umgebung gesucht. Sind Geräte in der Umgebung gefunden worden, wird ein Mutex in Form eines Wahrheitswertes gesetzt. Nun kann in L4 eine Session angefragt werden. Bei Erfolg einer Sitzungseröffnung können nun Nachrichten gesendet und empfangen werden. Zum Abschluss kann in L6 eine Verbindung zum Chromecast durch einen Aufruf beendet werden.

---

<sup>2</sup>Chromecast - <https://www.google.de/chrome/devices/chromecast>; letzter Zugriff: 29.10.2015

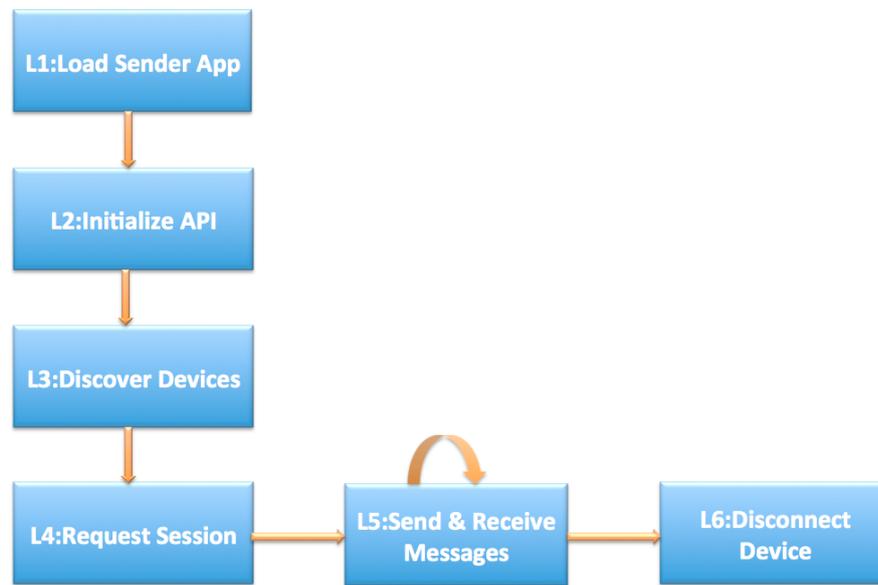


Abbildung 2.1.: Lebenszyklus einer Chromecast Sender Applikation.

## 2.2. Dexter

Das Open-Source-Framework Dexter<sup>3</sup> ist für die Entität-Verlinkung (vgl. [Ceccarelli und Lucchese, 2013](#)) konzipiert. Damit ist es möglich die Wikifizierung<sup>4</sup> eines Dokumentes unter Verwendung verschiedener Text Mining Algorithmen zu erlernen. In diesem Abschnitt wird auf die Arbeitsweise bei der Analyse eines Dokumentes, sowie beim Aufsetzen des Systems eingegangen.

### 2.2.1. Arbeitsweise von Dexter

Dexter arbeitet nach [Ceccarelli und Lucchese \(2013\)](#) in drei Schritten:

1. **Spotting:** Aus dem Dokument werden alle möglichen Kandidaten herausgesucht. Hierfür werden zur Identifizierung aller möglichen Linkkandidaten einzelne Wörter aus dem Dokument aufbereitet. Dazu erfolgt eine Berechnung, nachdem das Aufkommen der Entitäten im gebildeten Entität-Link-Graphen (siehe [Cudré-Mauroux u. a., 2009](#)) von Wikipedia gemessen wird. Falls eine Entität als Wikipedia-Artikel existiert, wird sie in die Ausgabeliste zur Weiterverarbeitung aufgenommen.

<sup>3</sup>Dexter - <https://github.com/dexter/dexter>; letzter Zugriff: 29.10.2015

<sup>4</sup>**Wikifizierung:** Wikifizierung ist ein mit der Entstehung von Wikipedia geschaffenes neues Wort. Es beschreibt die Verlinkung von Schlagwörtern auf andere Artikel zur Schaffung eines besseren Verständnisses der Gesamtthematik für den Nutzer.

2. **Disambiguation:** Die Mehrdeutigkeit einzelner Entitäten aus der Kandidatenliste wird aufgelöst. Hierfür werden die Erkenntnisse nach [Milne und Witten \(2008\)](#) angewandt. Dieses Verfahren beschreibt eine Möglichkeit mit Hilfe von maschinellen Lernens Zusatzinformationen anhand von Links zu Wikipedia-Artikeln bereitzustellen. Dazu erlernten die Autoren das Wikifizierungs-Verfahren<sup>5</sup> von Wikipedia. Hierfür übernahmen die Entwickler von Dexter die beiden Feature: *Commonness* und *Relatedness*. Die konkrete Berechnung der Werte wird im Abschnitt [3.1](#) erläutert.
3. **Ranking:** Die Liste der ausgesuchten Entitäten wird nach einem zugeordneten Wert sortiert. Unter anderem wird ein *Score* für die jeweiligen Kandidaten gebildet. Dieser besagt, dass im Kontext  $U$  ein Wikipedia-Artikel  $w$  die selbe Thematik wie andere Entitäten adressiert, wenn er die größte Gemeinsamkeit an Verlinkungen mit den als eindeutig erkannten Entitäten aufweist:

$$Score(c | U) = \frac{1}{\sum_u w_u} \sum_{u \in U} w_u * p^{MW}(u, c) \quad (2.1)$$

Desto größer der gemessene *Score*-Wert ist, desto höher seine Positionierung in der Ausgabeliste. Damit würde der Kandidat mit einer höheren Wahrscheinlichkeit ausgeliefert werden.

### 2.2.2. Architektur von Dexter

Dexter besteht aus vier Komponenten, die in aufeinanderfolgenden Arbeitsschritten aufgesetzt werden. Der Gesamtausschnitt von Dexter wird in der [Abbildung 2.2](#) veranschaulicht. Hierzu muss mit Hilfe der Komponente JSON-Wikipedia ein Wikipedia-Dump<sup>6</sup> vom XML-Format in JSON-Format umgewandelt werden. Ein Wikipedia-Dump stellt dabei einen Speicherauszug mit beinhalteten Wikipedia-Artikeln dar. Danach muss mit Hilfe von Dexter-Core der Speicherauszug in mehreren Teilschritten zu einem Entität-Link-Graphen umgewandelt werden. Unter anderem wird darunter mit Hilfe von Apache Lucene<sup>7</sup> aus dem Wikipedia-Dump ein Artikel-Index generiert. Danach muss die Komponente Dexter-Webapp konfiguriert werden. Das Teilsystem besteht dabei aus einer Web-App, die eine Swagger-Komponente<sup>8</sup> für die Visualisierung der angebotenen REST-API anbietet.

---

<sup>5</sup>Wikifizierung - <http://de.wikipedia.org/wiki/Wikipedia:Wikifizieren>; letzter Zugriff: 29.10.2015

<sup>6</sup>Wikipedia-Dumps - <https://dumps.wikimedia.org/dewiki/latest>; letzter Zugriff: 29.10.2015

<sup>7</sup>Lucene - <https://lucene.apache.org/core>; letzter Zugriff: 29.10.2015

<sup>8</sup>Swagger - <http://swagger.io/>; letzter Zugriff: 29.10.2015

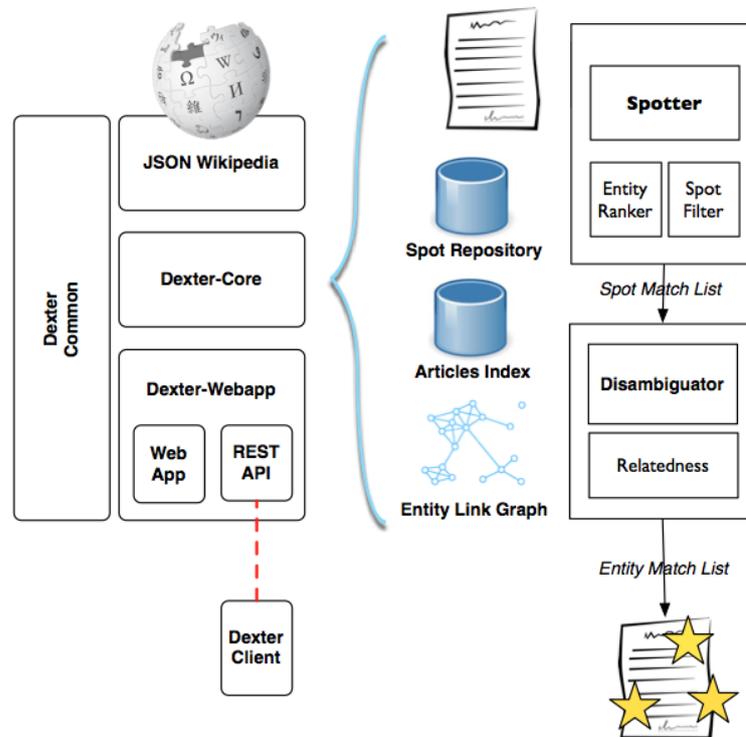


Abbildung 2.2.: Architektur von Dexter. Ceccarelli und Lucchese (2015)

## 2.3. Sumy

Sumy<sup>9</sup> ist eine Python-Bibliothek, welche Funktionalitäten für Textzusammenfassungen liefert. Diese implementiert fünf Textzusammenfassungsalgorithmen: Luhn, Edmundson, Enhanced Latent Semantic Analysis, LexRank und TextRank. Im weiteren Verlauf des Abschnitts wird auf die einzelnen implementierten Textzusammenfassungsalgorithmen eingegangen.

### 2.3.1. Extraktionsalgorithmus nach Luhn

Das Paper von Luhn (1958) beschäftigt sich mit der automatischen Erstellung von Abstracts durch Satzextraktion. Hierfür setzte Luhn auf das Maß der Häufigkeit von Wörtern und ihre Position in Sätzen. Der Grundgedanke des Ansatzes basierte auf der Annahme, dass ein Autor bestimmte Wörter, welche mit dem Thema verbunden sind im Text immer wieder wiederholt. Außerdem sagt die Position von relevanten Wörtern innerhalb eines Satzes etwas über die Relevanz der Aussage aus. Auf Basis der beiden Werte Position und Häufigkeit wird dann die

<sup>9</sup>Sumy - <https://github.com/miso-belica/sumy>; letzter Zugriff: 29.10.2015

Relevanz eines Satzes bestimmt.

Es wird ein Set an Wörtern mit ihrer Häufigkeit gebildet. Danach werden wenig relevante Wörter wie Stoppwörter aus der Liste entfernt. Die Wörter, welche in der Liste übrig bleiben sind die signifikanten Wörter einer Wortgruppe. Im Anschluss daran wird der Signifikanzwert eines Satzes berechnet, wonach die Position und Beziehung von relevanten Wörtern mit berücksichtigt wird. Eine Analyse von Dokumenten ergab, dass der Abstand von zwei signifikanten Wörter zu der selben Wortgruppe gehören, wenn sie maximal vier oder fünf Wörter auseinander liegen. Der Signifikanzfaktor wird demnach wie folgt berechnet:

$$\text{Signifikanzfaktor} = \frac{(\text{Anzahl signifikanter Wörter der Gruppe})^2}{\text{Anzahl der Wörter in der Gruppe}} \quad (2.2)$$

Hierzu wird für jeden Satz aus dem Dokument wird Signifikanzfaktor gebildet. Es gelangen nur die Sätze in das Abstract, welche den höchsten Signifikanzfaktor aufweisen. Der Vorteil von diesem Verfahren ist, dass solange sich der Autor an die aufgestellten Vorgaben bei seinem Schreibstil hält, konsistente und zuverlässige Zusammenfassungen erstellt werden können. Der Nachteil von dem Verfahren sind diese Stilvorgaben, weicht der Autor davon ab kann das Verfahren nicht mehr richtig greifen und die Qualität der generierten Zusammenfassung des Textes leidet daran.

### 2.3.2. Edmundsons heuristisches Verfahren

Das Paper von **Edmundson (1969)** beschreibt ein heuristisches Verfahren um Texte zusammenzufassen. Es baut auf dem Extraktionsalgorithmus nach Luhn auf. Als Trainingskorpus wurden 200 wissenschaftliche Dokumente aus den Feldern: Physik, Informationswissenschaft und humanistische Studien verwendet. Für den Testkorpus wurden 200 wissenschaftliche Dokumente aus der Chemie genutzt. Der aufgestellte Algorithmus besteht aus einer Kombination von vier Basismethoden: Anhaltspunkt-Methode  $C$ , Schlüssel-Methode  $K$ , Titel-Methode  $T$  und Lokations-Methode  $L$ . Des Weiteren wird jeder Methodik eine Gewichtung in Form eines Parameters  $a_1$  bis  $a_4$  zugeordnet. Für jeden Satz wird damit ein Gewicht auf Basis folgender linearen Funktion berechnet:

$$a_1C + a_2K + a_3T + a_4L \quad (2.3)$$

Im nachfolgenden Verlauf des Textes wird auf die einzelnen Basismethoden näher eingegangen: *Anhaltspunkt-Methode*: Basiert auf der Hypothese, dass die Relevanz eines Satzes von pragmatischen Wörtern wie zum Beispiel: „signifikant“ und „unmöglich“ abhängig ist. Hierfür

## 2. Grundlagen

---

wird ein Anhaltspunkt-Wörterbuch bestehend aus: Bonus-Wörtern, welche positiv relevant, Stigma-Wörtern die negativ relevant und Nullwörtern die irrelevant sind, angewandt. Jedem Satz aus dem Textkörper wird so eine Gewichtung zugeordnet.

*Schlüssel-Methode:* Das Prinzip der Gewichtung eines Satzes stützt sich auf dem entwickelten Ansatz nach Luhn. Hierfür werden auf Basis eines statistischen Verfahrens ein Schlüssel-Glossar gebildet. Es werden nur die Wörter im Textkörper betrachtet, die nicht im Anhaltspunkt-Wörterbuch stehen. Die Gesamtgewichtung der einzelnen Sätze wird durch Aufsummierung des errechneten Gewichtes von Schlüssel-Wörtern berechnet.

*Titel-Methode:* Die Methode basiert auf der Annahme, dass Wörter, welche im Titel oder Überschriften genannt werden, positiv relevant sind. Es wird ein Titel-Glossar bestehend aus den Wörtern von Titel, Untertitel und Überschriften gebildet. Erkannte Nullwörter werden entfernt. Das Gesamtgewicht eines Satzes basiert somit auf der Aufsummierung von den errechneten Titel-Gewichten.

*Lokations-Methode:* Die Lokations-Methode greift auf den Grundaufbau und Überschriften zu. Sie basiert auf zwei Hypothesen:

1. Sätze die nach einer Überschrift kommen sind positiv relevant.
2. Sätze die das Thema adressieren, tendieren eher dazu, am Anfang oder Ende des Dokumentes bzw. Absatzes zu stehen.

Für die Berechnung der Gewichtung eines Satzes verwendet die Methodik ein vorgespeichertes Überschriften-Wörterbuch. Dieses beinhaltet die rausgesuchten Wörter der Überschriften des Dokumentes. Zudem werden die Ordnungsposition von Sätzen im Text, ob sie unter anderem sich im ersten oder letzten Absatz des Textes befinden und ob sie am Anfang oder Ende des Absatzes stehen, mit in die Berechnung einbezogen. Damit wird Gesamtgewichtung eines Satzes aus dem gebildeten Überschrift-Gewicht und Ordnungs-Gewicht berechnet.

Die Problematik an dem Verfahren ist, dass ein breites Spektrum an Annahmen getroffen werden. So geht z.B. die Titel-Methode davon aus, dass der Titel etwas mit dem Text zu tun hat. Der Titel wird aber oft als Teaser für den Leser verwendet. So werden Begrifflichkeiten im Titel verwendet um die Aufmerksamkeit der Leser zu bekommen, aber nicht im eigentlichen Text genutzt. Abgesehen davon greift die Lokations-Methode nur dann, wenn der Stil des Autors von den getroffenen Annahmen nicht abweicht.

### 2.3.3. Enhanced Latent Semantic Analysis - ELSA

Im Paper von [Steinberger und Ježek \(2004\)](#) wird eine Erweiterung des Verfahrens Latent Semantic Analysis (LSA) vorgestellt. LSA ist dabei insbesondere für die Suche nach großen Datenmengen im Internet interessant. Es wurde zuerst 1990 von [Deerwester u. a. \(1990\)](#) erwähnt. Dieses Verfahren versucht den Kontext eines Dokumentes zu erfassen. Um die Sinnhaftigkeit eines Textes zu begreifen wird eine Termfrequenz-Matrix, die in verschiedenen Dimensionen aufgesplittet wird, verwendet. Es wird hierbei zwischen Hauptdimensionen, welche für die Bedeutung relevante Wörter beinhalten und weniger bedeutsame Dimensionen für die geringer relevanten Wörter, unterschieden. Bestünde ein Dokument aus den vier Wörtern: Mensch, Person, Dickhäuter und Elefant, würde sich die Anzahl an Dimensionen auf vier belaufen, aber bei der generalisierten Termfrequenz-Matrix auf zwei Dimensionen verringert werden.

Es hat nach Meinung der Autoren zwei Nachteile bei der Berechnung von relevanten Sätzen aus Texten. Der erste Nachteil bezieht sich auf die Dimensionen der Termfrequenz-Matrix, die gleich der Anzahl an geforderten Sätzen der Zusammenfassung entsprechen muss. Desto größer die Anzahl von aufgestellten Dimensionen ist, desto mehr irrelevanten Themen gelangen in die Zusammenfassung. Dieses Konzept ist dann von Vorteil, wenn das genaue Auftreten von Themen eines Dokumentes und damit die Anzahl an Dimensionen genau bestimmt werden kann. Der zweite Nachteil bezieht sich darauf, dass Sätze, welche einen hohen Indexwert aufweisen, aber nicht den höchsten Wert haben, nicht in die Zusammenfassung gelangen, obwohl ihr Inhalt für das Verständnis des Textes bedeutsam ist.

Zuerst wird eine Eingangsmatrix  $A$  nach  $m \times n$  erstellt, wobei  $n$  die Sätze und  $m$  die einzelnen Wörter repräsentieren. Im zweiten Schritt wird die Singulärwertzerlegung angewandt nach der aus  $A$  zwei orthogonalen Matrizen  $U$  und  $V^T$ , sowie eine Diagonalmatrix  $\Sigma$  abgeleitet wird.

$$A = U \Sigma V^T \quad (2.4)$$

Nach der Singulärwertzerlegung erfolgt die Satzauswahl für die Zusammenfassung. Hierfür wird die unten aufgeführte Formel verwendet. Demnach wird für einen Satz  $k$  dessen Länge des Vektors  $s_k$  berechnet. Die Anzahl an Dimensionen der Termfrequenz-Matrix wird von  $n$  repräsentiert. In die Zusammenfassung gelangen nur Sätze mit den höchsten Werten im Vektor  $s$ .

$$s_k = \sqrt{\sum_{i=1}^n v_{k,i}^2 * \sigma_i^2} \quad (2.5)$$

Die Problematik an dem Algorithmus ist, dass zwar nur Sätze mit einem hohen berechneten Wert in die Zusammenfassung gelangen, aber die Themenvielfalt nicht mehr an das eigentliche Verfahren LSA reicht. Damit würde die berechnete Zusammenfassung nach dem erweiterten LSA konkreter ein Konzept beschreiben, aber weniger Themen aus dem eigentlichen Text beinhalten.

### 2.3.4. LexRank

Das Paper von [Erkan und Radev \(2004\)](#) bietet einen Graph-basierten Ansatz, mit dessen Texte zusammengefasst werden können, an. Es wird die inverse Dokumenthäufigkeit nach [Spark Jones \(1972\)](#) angewendet:

$$idf_i = \log\left(\frac{N}{n_i}\right) \quad (2.6)$$

Diese Formel misst die allgemeine Bedeutung eines Terms für die Gesamtmenge des betrachteten Dokumentes. Der Term  $N$  steht für die totale Anzahl an Dokumenten in einer Gruppe und  $n_i$  ist die Nummer von Dokumenten in dem der Wert  $i$  auftritt. Wörter wie Artikel, die häufig in jedem Dokument auftreten, würden einen niedrigen idf-Wert bekommen, während seltene Wörter wie z.B. medizinische Begriffe dagegen einen hohen idf-Wert erhalten.

So wird um eine Ähnlichkeit zwischen zwei Sätzen zu bestimmen ein *Bag of Word*-Modell verwendet um einen  $N$ -dimensionalen Vektor zu bilden. Zuvor werden die Sätze in einer Tabelle mit dem zugeordneten Identifikator abgelegt. Die Variable  $N$  steht für Anzahl an möglichen Wörtern einer Sprache. Für jedes Wort, was in einem Satz auftritt, wird ein Wert gebildet und die inverse Dokumenthäufigkeit  $idf_i$  angewandt. Die Ähnlichkeit zwischen zwei Sätzen wird dann anhand des Kosinus der beiden entsprechenden Vektoren gebildet:

$$idf - modified - cosine(x, y) = \frac{\sum_{w \in x, y} tf_{w,x} tf_{w,y} (idf_w)^2}{\sqrt{\sum_{x_i \in x} (tf_{x_i,x} idf_{x_i})^2} \times \sqrt{\sum_{y_i \in y} (tf_{y_i,y} idf_{y_i})^2}} \quad (2.7)$$

Der Parameter  $tf_{w,s}$  steht für das Auftreten eines Wortes  $w$  in einem Satz  $s$ . Die Variablen  $x$  und  $y$  stehen für die beiden untersuchten Sätze. So würde nach der Formel zwischen jedem Satz in dem untersuchten Text ein Wert gebildet. Es wird danach ein Graph aufgespannt, wo

die Sätze Knoten und die gebildeten Werte Kantengewichte zwischen zwei Knoten darstellen. Danach werden alle Kanten mit einem niedrigen Kantengewicht entfernt. Desto höher die minimale Wertigkeit ist, desto weniger Kanten zwischen den Knoten bleiben übrig.

Nachteil dieses Verfahrens ist, dass Überschriften oder Satzpositionen im Text nicht mit in die Berechnung der Zusammenfassung einfließen können. So kann ein Autor das Thema X als Überschrift adressieren, aber die Kurzfassung über eine Thematik Y berichten. Dies erscheint als Vorteil, wenn die Überschrift wirklich nichts mit dem Text zu tun hat. Aber falls die Thematik X aus dem Titel zwar in dem Text vorkommt, doch ein Thema Y überproportional auftritt, so würde der zweite Inhalt für die Zusammenfassung verwendet.

### 2.3.5. TextRank

Einen weiteren Graph-basierten Ansatz für die Zusammenfassung von Texten bieten [Michalcea und Tarau \(2004\)](#). Hierbei stellen Knoten Sätze und die Kanten zwischen ihnen ihre Beziehungen dar. Die Sätze im Text werden durch die Bildung eines trainierten Modells getrennt.

Zuerst werden die Sätze in ihre einzelnen Wörter zerlegt. Als nächstes wird eine dünnbesetzte Matrix von Wörtern mit einem Zähler erstellt. Jedes Wort aus der dünnbesetzten Matrix wird dann durch das tf-idf-Maß normalisiert. Durch die relative Häufigkeit eines Wortes wird eine Verzerrung der Ergebnisse in langen Dokumenten verhindert, dazu wird die Anzahl an Vorkommen von einem Term durch die maximale Häufigkeit eines Terms geteilt. Danach erfolgt die Durchführung einer Ähnlichkeitsanalyse zwischen den Sätzen. Die Gesamtgewichtung eines Satzes erfolgt dann auf Basis des modifizierten PageRank-Algorithmuses (siehe [Page u. a. \(1998\)](#)):

$$WS(V_i) = (1 - d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \quad (2.8)$$

Die Variablen:  $V_i$  und  $V_j$  stehen dabei für zwei Knoten und  $w_{ij}$  ist die Gewichtung, welche die Kante zwischen den beiden befindet hinzu addiert wird. Mit Hilfe eines Dämpfungsfaktors  $d$  wird angegeben mit wie viel Gewichtung der berechnete Wert in das Graphen-Model hinzugefügt werden soll. Die Bestimmung der bedeutsamen Sätze im Text erfolgt nach dem PageRank-Algorithmus. Desto höher die Wertung eines Knotens, desto bedeutsamer ist der repräsentierte Satz.

Wie bei dem Graph-basierten Ansatz LexRank werden Textgegebenheiten wie Titel und Positionierung der Sätze nicht mit betrachtet. Ein Text, welcher im Titel eine Thematik X adressiert kann gegen eine Thematik Y, wegen eines Übervorkommens verlieren und nicht in der Zusammenfassung auftauchen.

### 2.4. Struktur und Ablauf von wissenschaftlichen Untersuchungen

Das Buch von **Eid u. a. (2013)** befasst sich mit Statistik und Forschungsmethoden. Hierbei wird auf die Struktur und den Ablauf von wissenschaftlichen Untersuchungen eingegangen. Dazu werden die gesammelten Erkenntnisse aus dem Buch für eine Untersuchung in drei Schritte eingeteilt: Versuchsvorbereitung, Versuchsdurchführung und Auswertung der Daten. Die vorgestellten Punkte werden im weiteren Verlauf des Abschnitts näher betrachtet.

#### 2.4.1. Versuchsvorbereitung

Eine wissenschaftliche Untersuchung fängt mit der Aufstellung einer Behauptung an, die beispielsweise wie folgt lauten könnte: „Wer intelligent ist, der ist auch kreativ“. Doch nicht alle Aussagen können untersucht werden. Hierbei wird zwischen prüfbar und nicht-prüfbar Aussagen unterschieden. Demnach gibt es Aussagen wie: „In der Hölle ist es heißer als 220 Grad“, die nicht nachprüfbar sind, da die Zustände in der Hölle nicht erfahrbar sind. Aber Erfahbarkeit ist in der empirischen Wissenschaft eine der notwendigen Grundvoraussetzungen. Eine Aussage muss falsifizierbar sein, damit sie untersucht werden kann. Sie muss prinzipiell falsch sein können, damit sie empirisch prüfbar ist. Bei einer wissenschaftlichen Untersuchung wird eine wissenschaftliche Hypothese gebildet, die empirisch testbar und damit falsifizierbar sein muss. Außerdem muss diese wissenschaftliche Hypothese begründet sein.

Um eine Hypothese empirisch zu prüfen müssen ihre Bestandteile konkretisiert werden. Hierzu gibt es in der Theorie keine allgemeingültigen Aussagen darüber, wie eine solche Konkretisierung erfolgen soll. Als nächstes müssen die konkretisierten Bestandteile einer Aussage in Zahlen umgewandelt werden. Hat sich ein Forscher für Größen entschieden, kann er Datenerhebung und Datenauswertungsmethoden anwenden. Hierfür stehen ihm verschiedene Möglichkeiten zur Verfügung. Darunter kann er beispielsweise eine Verhaltensanalyse oder einen Fragebogen passend zu den konkretisierten Bestandteilen der Aussage erstellen. Für die zweite Option kann dem Buch „Die Fragebogen-Methode“ von **Mummendey und Grau (2008)** eine Anleitung zur Entwicklung eines Fragebogens entnommen werden. Außerdem

muss sich je nach wissenschaftlicher Hypothese ein Versuchsaufbau überlegt werden. Je nach dem gesetzten Schwerpunkt müssen danach für die Stichprobe der Untersuchung Personen rekrutiert werden (vgl. [Eid u. a., 2013](#)).

### 2.4.2. Versuchsdurchführung

Nachdem die Entscheidung für eine bestimmte Form der empirischen Überprüfung der Hypothese gefallen ist, muss eine Auseinandersetzung mit den Problemen bei der Versuchsdurchführung erfolgen. Unter anderem werden drei verschiedene potenzielle Probleme vorgestellt:

1. **Mangelnde Validität:** Dieses Problem besteht, wenn beispielsweise teilnehmende Probanden falsche Aussagen tätigen. So könnten z.B. zwei Teilnehmer bei der Frage, wie oft sie einen Diebstahl in diesem Jahr begangen haben, mit „noch nie“ bzw. „drei Mal“ antworten, was der Unwahrheit entsprechen kann. Damit würde bei einer Fragestellung, die auf das normabweichende Verhalten von Menschen abzielt, nicht die Normabweichung erfasst werden, sondern die Ehrlichkeit von Menschen.
2. **Systematisch fehlende Werte:** Dieses Problem kann auftreten, wenn Personen aus der Stichprobe sich weigern gestellte Fragen zu beantworten. Dann würden Werte in den erhobenen Datensätzen fehlen. Das führt dazu, dass die Aussagekraft der erworbenen Daten darunter leidet.
3. **Ethische Unbedenklichkeit:** Ein Aspekt bei der Planung einer Untersuchung stellt die ethische Unbedenklichkeit dar. Hierbei steht dem Erkenntnisinteresse der Wissenschaftler die ethische Verantwortlichkeit gegenüber den Versuchspersonen gegenüber. Eine Untersuchung ist ethisch bedenklich, wenn sie die Menschenwürde verletzt oder mit potenziellen Gefahren für Leib, Leben und Wohlergehen verbunden ist. Die Deutsche Gesellschaft für Psychologie (DGPs) und der Berufsverband Deutscher Psychologinnen und Psychologen (BDP) haben dafür ethische Richtlinien erstellt (siehe [Deutsche Gesellschaft für Psychologie e.V., 2004](#)).

Des Weiteren müssen vor Beginn des Experiments die Versuchspersonen über potenzielle Risiken des Experimentes soweit sie vorhanden sind, aufgeklärt werden. Außerdem muss eine Belehrung darüber erfolgen, dass die Teilnahme an der Untersuchung freiwillig ist und jederzeit ohne Angabe von Gründen abgebrochen werden kann. Damit die erworbenen Datensätze ausgewertet werden können, muss eine Einwilligungserklärung seitens der Versuchsperson eingeholt werden. Spätestens nach dem Abschluss der Untersuchung muss eine ausführliche

und lückenlose Aufklärung der Versuchspersonen über den Zweck der Untersuchung erfolgen (vgl. Eid u. a., 2013).

### 2.4.3. Auswertung der Daten

Nachdem die Erhebung abgeschlossen wurde erfolgt die Auswertung der Daten. Dazu müssen die Datensätze zuerst aufbereitet werden. Hierbei kann eine Transformation der Informationen in Zahlen erfolgen. Nachdem die Daten in ein Zahlenformat umgewandelt worden sind, stehen fünf verschiedene Auswertungsmethoden zur Verfügung: Datenmatrix, Deskriptivstatistik, Streuung, Kovariation und zentrale Tendenz. Diese werden im weiteren Verlauf des Abschnitts näher beschrieben.

1. **Datenmatrix:** Die Datenmatrix kann für die Darstellung der Daten verwendet werden. Hierbei wird eine Matrix mit einer Anordnung von Zahlen in Tabellenform angelegt. Die Tabelle wird durch die Anzahl der Zeilen und Spalten definiert. In der Form können Spalten beispielsweise einzelne Fragen des Fragebogens darstellen. In einer Zelle der Matrix kann dann die numerisch kodierte Antwort stehen.
2. **Deskriptive Statistik:** Mit Hilfe der deskriptiven Statistik können Daten beschrieben werden. Hierzu erfolgt die Auswertung der Zahlen aus der Datenmatrix anhand der Kombination von bestimmten Regeln. Um zu bestimmen, wie oft eine bestimmte Verhaltensweise vorkommt, kann die Summe der Spalten eines Fragebogens ermittelt werden, in denen die Antworten der Probanden stehen. Die Summe der jeweiligen Spalte steht dann dafür, wie häufig das entsprechende Verhalten in der Gruppe insgesamt vorkam.
3. **Kovariation:** Das ist die Verbindung zwischen zwei oder mehreren Fragen. Damit kann bestimmt werden, wie viele Personen mehrere Verhaltensweisen aufweisen. Kovariation gehört der deskriptiven Datenanalyse an, weil sie die Verhältnisse in der Untersuchungstichprobe beschreibt.
4. **Streuung:** Damit kann beispielsweise bestimmt werden, wie stark sich das Verhalten einer Person von dem normalen Verhalten unterscheidet. Analysen dieser Form intendieren die Beschreibung der Verteilung von Verhaltensweisen in der untersuchten Gruppe.
5. **Zentrale Tendenz:** Um die Häufigkeit der jeweiligen Verhaltensweise zu bestimmen, kann die Summe einer Spalte durch die Anzahl der Probanden dividiert werden. Der gebildete Wert stellt die Gesamtwertung einer Gruppe dar.

## 3. Analyse

Dieses Kapitel dient der zielgerichteten Vorbereitung der Konzeption von SecondCast. Hierfür soll auf die Problembereiche bei der Entwicklung von Second Screen Anwendungen anhand von möglichen Lösungsansätzen eingegangen werden. Mit Hilfe der gewonnenen Erkenntnisse sollen Anforderungen an das zu entwickelnde System abgeleitet werden.

Dabei kann die Problemstellung in fünf Teilbereiche eingeteilt werden. Der erste Teilbereich geht im Abschnitt 3.1 auf die bedingte Multitasking-Fähigkeit der Menschen ein. Als nächster Problembereich wird im Abschnitt 3.2 die Kommunikation der beteiligten Mediengeräte beleuchtet. Im darauffolgenden Abschnitt 3.3 wird der dritte Teilbereich beim Aufbau der Erzählstruktur für die Nutzergruppen und die Bereitstellung von Zusatzinformationen für den Nutzer vorgestellt. Der nächste Teilbereich im Abschnitt 3.4 stellt die kontextuelle Ebene von Second Screen dar, nachdem die Informationsbereitschaft betrachtet wird. Der letzte Problembereich behandelt im Abschnitt 3.5 auf die Fragestellung, wie Informationen für die Nutzer personalisiert werden können. Der vorletzte Abschnitt 3.6 geht auf die Benutzerrollen, sowie auf die abgeleiteten funktionalen und technischen Anforderungen an die zu entwickelnde Untersuchungsplattform ein. Im letzten Abschnitt 3.7 wird ein Fazit gezogen.

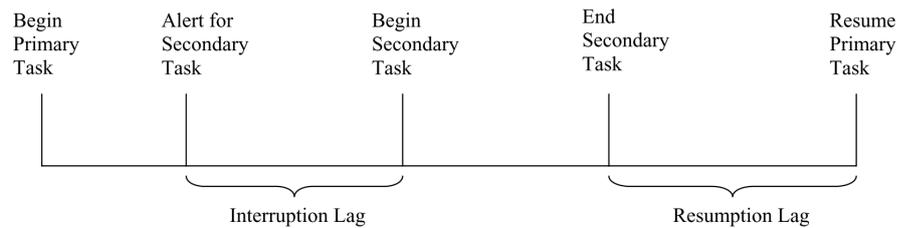
### 3.1. Psychische Ebene

Der folgende Abschnitt geht auf die bedingte Multitasking-Fähigkeit von Menschen ein. Dazu werden die Ergebnisse aus zwei ausgesuchten Studien vorgestellt, die auf Auswirkungen von Arbeitsunterbrechungen und Arbeitsstrukturierung im beruflichen Umfeld untersuchen.

#### 3.1.1. Auswirkungen von Arbeitsunterbrechungen

Die Studie von **Trafton u. a. (2003)** untersucht die Auswirkungen von Unterbrechungen auf ausgeführte Aufgaben. Dabei wurde die Vermutung aufgestellt, dass die zeitverzögerte Ausführung einer Aufgabe nach einer Unterbrechung durch eine Unterbrechungsverzögerung (*interruption lag*) erleichtert wird. Die Abbildung 3.1 beschreibt in Form einer Zeitlinie die

Unterbrechung einer Hauptaufgabe gefolgt von ihrer Wiederaufnahme. In diesem Fall könnte die Hauptaufgabe das Verfassen einer E-Mail und die zweite Aufgabe ein Handyanruf sein. Die Zeit zwischen dem Klingeln und dem Herangehen an das Handy ist die Unterbrechungsverzögerung. Während die Zeitspanne zwischen dem Verlassen der zweiten Aufgabe und das Beginnen der Hauptaufgabe die Wiederaufnahmeverzögerung (*resumption lag*) darstellt.



**Abbildung 3.1.:** Der Unterbrechungs und Fortführungs-Prozess, mit der Einbezugnahme einer Hauptaufgabe durch eine unterbrechende Aufgabe. [Trafton u. a. \(2003\)](#)

### Untersuchung

Um zu testen, ob die Unterbrechungsverzögerung als mentale Vorbereitung eine Auswirkung auf die Fortführung der Arbeit hat, wurde ein Experiment durchgeführt. Die Versuchsgruppe bestand aus 17 Personen (14 Männern und 3 Frauen) und wurde in zwei Teile aufgeteilt. Alle Teilnehmer des Experimentes mussten Aufgaben mit einem Computerprogramm erledigen. Die Kontrollgruppe bestand aus neun Personen, sie mussten von ihrer Hauptaufgabe ohne Vorwarnung auf eine zweite Aufgabe umschalten. Ferner hatte die zweite Gruppe ein acht Sekunden langes visuelles Zeichen angezeigt bekommen, was ihnen vermittelt hat, dass es gleich zu einer Unterbrechung ihrer Hauptaufgabe durch eine andere Aufgabe kommt. Hierbei wurde vermutet, dass Probanden, die keine Unterbrechungsverzögerung nach der Unterbrechung ihrer Hauptaufgabe haben, langsamer anfangen wurden ihre Arbeit fortzuführen und damit länger für die Bearbeitung der Aufgabe brauchen.

### Ergebnisse

Die Untersuchung hat ergeben, dass die Probanden mit Unterbrechungsverzögerung ihre Hauptaufgabe schneller fortsetzen konnten als die Teilnehmer der Kontrollgruppe. Es ist davon auszugehen, dass das Gehirn eine gewisse Verzögerungszeit braucht, um von einer Aufgabe zu der nächsten umzuschalten. Je nach der Komplexität der konkurrierenden Aufgabenstellungen

braucht das Gehirn eine kürzere oder längere Unterbrechungsverzögerung. Damit muss einer Person eine ausreichend lange Zeit für den Wechsel zwischen zwei Aufgaben gegeben werden.

#### **3.1.2. Arbeitsorganisation**

Die Studie von **González und Mark (2004)** befasst sich damit wie Angestellte ihre Aufgaben organisieren, um sie zu bewerkstelligen. Ein Hauptaugenmerk wurde auf Arbeitsunterbrechungen zwischen einer Hauptaufgabe durch eine zweite Aufgabe gestellt. Dabei wurde im Gegensatz zu anderen Studien, welche sich nur in einem konkreten vorgegebenen Setting befinden, die Strukturierung der Arbeiten den Probanden selbst überlassen.

#### **Untersuchung**

Für die Untersuchung wurde aus einem amerikanischen Investmentunternehmen ein 25-köpfiges Team bestehend aus Datenbankadministratoren, Finanzanalysten, Softwareentwicklern und Managern gebildet. Aus dem Team wurden 14 Probanden ausgewählt. Die Erhebungsmethoden der Studie waren Verhaltensanalysen und die Verwendung von langen Interviews. Dabei haben die Forscher alle Tätigkeiten der Probanden mit Namen und Zeit, die sie für die Tests gebraucht haben, notiert. Hat beispielsweise ein Forscher bemerkt, wie ein Proband telefoniert, so notierte es sich die Tätigkeit „Telefonieren“ und die Dauer des Gesprächs. Die Untersuchung erfolgte über sieben Monate. Es wurden insgesamt 477 Arbeitsstunden in der Studie ausgewertet.

#### **Ergebnisse**

Auf Basis der gesammelten Daten für die durchgeführten Tätigkeiten der Probanden, wurde die Durchschnittszeit für die jeweilige Aufgabe pro Tag und einmalige Durchführung gebildet. Die Auswertung der Datensätze hat ergeben, dass die Abarbeitung der Aufgaben sehr fragmentiert war. Es wurde festgestellt, dass die Durchschnittszeit eines Probanden für den Wechsel zwischen zwei Aufgaben sich bei drei Minuten liegt. Des Weiteren brauchten die Teilnehmer während der Arbeit am PC im Durchschnitt zwei Minuten um zwischen zwei Programmen zu wechseln.

Die vorgestellten Arbeiten erfolgten im beruflichen Kontext. Im Gegensatz dazu findet Second Screen überwiegend im Freizeitbereich statt. Dementsprechend können die Ergebnisse nur analog übernommen werden. Dennoch zeigen die Arbeiten interessante Resultate im Bezug auf die Grenzen von Multitasking, die bei der Entwicklung von Second Screen Anwendungen

mit einbezogen werden sollten. Konkret müssen Überlegungen im Bezug, welches Format bei der später angestrebten Untersuchung verwendet werden sollte, getätigt werden. Da die Aufmerksamkeitspanne der Menschen begrenzt ist muss ihnen eine ausreichend lange Umschaltzeit zwischen zwei Aufgaben gegeben werden. Dazu eignen sich kurze Beiträge, wie Nachrichtenblöcke, im Gegensatz zu längeren Formaten, wie Dokumentationen und Filme nicht. Weitere Studien, welche sich mit der Multitasking-Fähigkeit von Menschen beschäftigen werden in den folgenden Arbeiten besprochen: [Benbunan-Fich u. a. \(2011\)](#), [Lehmann u. a. \(2013\)](#) und [Lottridge u. a. \(2015\)](#).

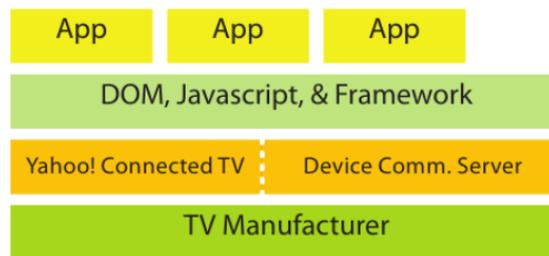
## 3.2. Technische Ebene

Ein entscheidender Faktor bei der Entwicklung von Second Screen Anwendungen ist die Kommunikation zwischen den beteiligten Mediengeräten. Dieser Abschnitt stellt einen Ansatz für ein Protokoll vorgestellt. Hierbei wird in [Cortez u. a. \(2012\)](#) auf Gerätekommunikation zwischen zwei Multimediageräten von der Authentifizierung bis zur eigentlichen Datenübertragung eingegangen. Dabei geht es um die Realisierung einer Architektur und eines Protokolls für eine Second Screen API. Das Bestreben war es eine anspruchsvolle Plattform zwischen TV und fortschrittlichen Smartphones, Tablets und Notebooks zu realisieren.

Die Nutzerinteraktion erfolgt durch eine Fernbedienung und Tastatur. Außerdem ermöglicht die Plattform die bidirektionale Versendung von Nachrichten über den Kommunikationskanal eines Netzwerks mit Hilfe eines Open-Source-Protokolls zu versenden. Die Einrichtung der Plattform erfolgte nach der W3C Spezifikation für Heimnetzwerk-Szenarien (vgl. [Clarke u. a., 2013](#)). Bei der Realisierung wurde die Softwareplattform *Yahoo! Connected TV* verwendet. Es wurden an das Kommunikationsprotokoll folgende vier technische Anforderungen gestellt:

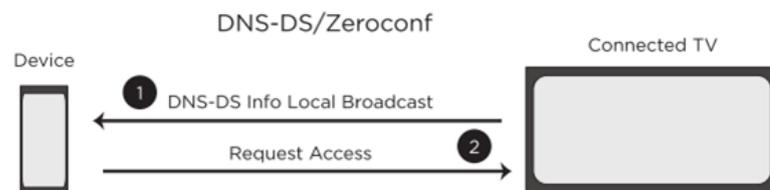
1. Sicherheit
2. Echtzeit Ereignisse
3. Einfachheit
4. Flexibilität

Die Abbildung [3.2](#) veranschaulicht die Architekturübersicht der Softwareplattform. Durch die Verwendung eines Cross-Compilers wird JavaScript-Code in native Funktionalitäten, wie z.B. unter Android, umgewandelt. Diese ermöglichen den Gebrauch von Webapplikationen (Web-Apps) auf dem Gerät.



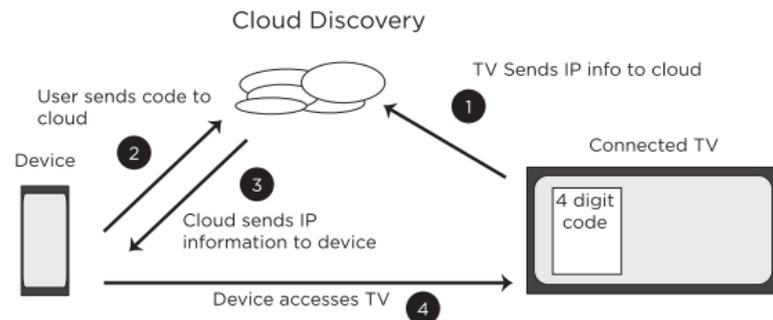
**Abbildung 3.2.:** Architekturübersicht der *Yahoo! Connected TV* Plattform. Cortez u. a. (2012)

Außerdem werden verschiedene Ansätze für die Authentifizierung zwischen zwei Medien-geräten vorgestellt. Der erste Ansatz von der DNS-SD Discovery wird von [Abbildung 3.3](#) verbildlicht. Dieses System ermöglicht es Services anhand von Multicast-Paketen zu annon-cieren. Hierbei wird ein Kommunikationsservice auf dem Fernseher ausgeführt. Dieses gibt seinen einen eindeutigen Namen, IP-Adresse und Port im Netzwerk bekannt. Die auf den Smartphones oder Tablets befindenen Clients, hören auf diese Pakete und können daraufhin verbraucherfreundliche grafische Benutzeroberflächen anzeigen. Bei diesem Ansatz müssen die Benutzer keine zusätzlichen Schritte für einen Verbindungsprozess eingehen.



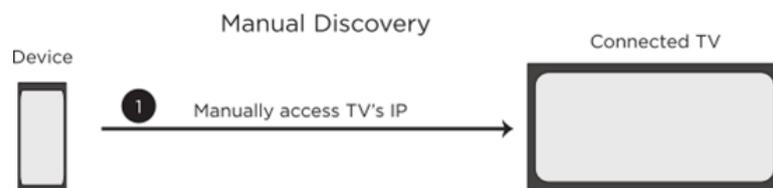
**Abbildung 3.3.:** DNS-DS/Zeroconf übertragene Verbindungsinformationen. Cortez u. a. (2012)

Dagegen stellt die [Abbildung 3.4](#) ein Verfahren dar mit dem die IP-Adressen und Ports in einer Cloud verwaltet werden. Dazu versendet ein Smart TV seine IP-Adresse und den dazugehörigen Port an die Cloud. Daraufhin wird ein Code bestehend aus vier Buchstaben auf dem Fernseher ausgegeben. Der Benutzer muss händisch den Buchstaben-Code in der Anwendung angeben. Danach wird der Code an die Cloud gesendet, um die korrekte IP-Adresse des Smart TVs festzustellen, mit dem sich der Client dann verbinden kann.



**Abbildung 3.4.:** Cloud-basierter Webdienst für Verbindungsinformationen. Cortez u. a. (2012)

Die dritte Variante stellt einen manuellen Zugang dar (siehe Abbildung 3.5). Dazu werden IP-Adresse und Port auf dem Fernseher angezeigt. Danach muss der Benutzer um sich verbinden zu können die angegebenen Informationen zur Authentifikation händisch in ein vorgesehenes Eingabefeld der Anwendung eingeben.



**Abbildung 3.5.:** Manuelle Eingabe der IP-Adresse vom Nutzer. Cortez u. a. (2012)

Neben *Yahoo! Connected TV* existieren weitere Softwareplattformen wie: *Apple TV*, *Chromecast* und *Google TV*, die zur Umsetzung von Second Screen Anwendungen verwendet werden könnten. Konkret muss bei der Entscheidung, welches Gerät als Kommunikationsplattform für die Realisierung einer Second Screen Anwendung verwendet werden sollte, anhand der Erfüllung von in der Arbeit vorgestellten technischen Anforderungen entschieden werden. Alternative Kommunikationsstrukturen können den Papern von: Cesar u. a. (2008), Chen u. a. (2010) und Murray und Goldenberg (2012) entnommen werden.

### 3.3. Darstellende Ebene

Der nachfolgende Abschnitt geht darauf ein, was Nutzer von einer Second Screen Anwendung erwarten würden. Hierfür wird in diesem Abschnitt auf die Ergebnisse aus zwei Designstudien eingegangen. Auf Basis der gewonnenen Erfahrungen sollen Anforderungen im Verlauf der Arbeit abgeleitet werden.

#### 3.3.1. Social TV

Die Designstudie von [Hess u. a. \(2011\)](#) setzte sich mit der Entwicklung eines Konzeptes für Social TV auseinander. Der Begriff Social TV steht dabei für die Verbindung zwischen Fernsehen und Social Media. Sie wurde im Rahmen der Arbeitsgruppe Social Media an der Uni Siegen durchgeführt. An der Untersuchung haben 27 Probanden aus 16 Haushalten teilgenommen. Hierbei mussten die Teilnehmer in einem Zeitraum von mehreren Wochen ein Medientagebuch über ihren Medienkonsum führen. Dazu dokumentierten sie jegliche Mediennutzung mit Datum, Dauer, Inhalt und Motivation des Medienkonsums in dem mitgegebenen Tagebuch. Bei der Studie war die zentrale Frage, welche multimediale Kommunikationsarten dienlich sind und was für Anwendungsfälle, Konzepte und Unterstützungswerkzeuge den Benutzern helfen können. Die Auswertung der erarbeiteten Konzepte erfolgte mit Hilfe zweier Workshops an denen die Probanden teilgenommen haben. Sie ergaben, dass Second Screen Anwendungen im Bereich Social TV folgendes unterstützen sollten:

1. Einbindung von sozialen Netzwerken
2. Lieferung von Zusatzinhalten
3. Personalisierung von Informationen
4. Empfehlungen über weitere interessante Inhalte geben

#### 3.3.2. Fernbedienung

Die Designstudie von [Bernhaupt u. a. \(2012\)](#) beschäftigte sich mit möglichen Empfehlungen für die Gestaltung von Smartphone-Anwendungen, die eine Kontrolle von IPTV-Systemen ermöglichen. Hierfür wurde eine Untersuchung mit 30 Probanden aus 15 Haushalten im November 2011 durchgeführt. Als Erhebungsmethode diente Playful Probing (vgl. [Bernhaupt u. a., 2007](#)), was einen spielerischen Ansatz zur Anforderungserhebung beschreibt. Die Fragestellung an die Teilnehmer der Studie bezog sich darauf, wie sie ihr Smartphone im Heimnetzwerk verwenden und welche Problemstellungen, die IPTV betreffen, mit Unterstützung des mobilen Gerätes gelöst werden können. Hierfür mussten die Probanden gestellte Aufgaben, wie beispielsweise das Nutzungsszenario beim Heraussuchen von Inhalten zum Konsum auf dem IPTV, lösen. Die Probanden eines Haushaltes bekamen ein Paket an Dokumentationsmöglichkeiten zugewiesen. Für Dokumentation der Resonanzen standen ihnen unter anderem Fragebögen, Kreativitätskarten, eine Videokamera, sowie ein extra entwickeltes Brettspiel, in dem das Thema der Studie adressiert wurde, zur Verfügung (siehe Abbildung 3.6).



**Abbildung 3.6.:** Playful Probing Paket mit Kreativitätskarten, Fragebögen, Videokamera und Spielkarten. [Bernhaupt u. a. \(2012\)](#)

Die Resultate der Untersuchung wurden gebündelt und ausgewertet. Es resultierten folgende Empfehlungen, was eine Anwendung für das Smartphone zur Kontrolle von IPTV im Heimnetzwerk erfüllen müsste:

1. Direkte Kontrolle aller Mediengeräte im Wohnzimmer für die optimale IPTV Erfahrung
2. Unterstützung von benutzerorientierten Szenarien
3. Personalisierung der Fernbedienung
4. Kontrolle der Mediengeräte im Heimnetzwerk
5. Verbesserung der allgemeinen Benutzerfreundlichkeit durch Sicherheitsmaßnahmen
6. Unterstützung von Berührungen und Sprache als Interaktionstechniken

Die beiden vorgestellten Designstudien bieten einen Einblick auf die Erwartungshaltung von Anwendern. Es zeigt wie unterschiedlich die Anforderungen im Bezug auf das jeweilige Aufgabenfeld einer Second Screen Anwendung sein können. Dennoch lassen sich die beiden vorgestellten Konzepte miteinander verbinden, um möglichst breite Masse an Nutzern zu erreichen. Alternative Studien die sich mit der Erwartungshaltung beschäftigen, was eine Second Screen Anwendung mit Bezug zu einer TV-Plattform leisten sollte, werden in: [Pirker u. a. \(2010\)](#), [Basapur u. a. \(2011\)](#) und [Lochrie und Coulton \(2012\)](#) besprochen.

### 3.4. Kontextuelle Ebene

Nicht immer werden zu einem Video Zusatzinformationen angeboten. Meistens gibt es einen Beschreibungstext zu dem Video. Der nachfolgende Abschnitt geht auf die Problemstellung ein, wie aus einem Beschreibungstext zum Video weitere Zusatzinformationen anhand der Analyse des Kontextes generiert werden können.

#### 3.4.1. Erlernung der Wikifizierung

Das Paper von [Milne und Witten \(2008\)](#) beschreibt ein Verfahren aus algorithmisch-anwendungsorientierter Perspektive zur automatischen Kreuzreferenzierung von Dokumenten mit Wikipedia-Artikeln. Dies ist eine Möglichkeit mithilfe maschinellen Lernens Zusatzinformationen anhand von Links zu Wikipedia-Artikeln bereitzustellen. Hierbei haben schon einige Arbeiten den Ansatz für die Generierung von strukturierten Anfragen an Informationssysteme implementiert (siehe [Jadidinejad und Mahmoudi, 2009](#); [Ceccarelli u. a., 2013](#)).

Damit die Kreuzreferenzierung ermöglicht werden kann, wollten die Autoren das Wikifizierungs-Verfahren von Wikipedia erlernen. Dieses Verfahren wird dazu genutzt, um möglichen Verständnisschwierigkeiten der Nutzer vorzubeugen. Demnach werden die Wikipedia-Artikel miteinander durch eine Verlinkung verbunden, die semantisch in Verbindung zueinander stehen. Hierfür wurden aus zwei Millionen Artikeln 700 Artikel rausgesucht, die mindestens eine Verlinkung zu 50 Artikeln aufwiesen. Die gesammelten Artikel wurden als Trainingsdaten und Testdaten verwendet. Als Klassifikatoren wurden mehrere Klassifikationsverfahren, darunter der Entscheidungsbaum C4.5, die lineare Support Vektor Maschine und der auf Häufigkeit basierte Naiver Bayes verwendet.

Das Verfahren teilt sich in Link-Begriffsklärungsphase und Link-Ermittlungsphase auf. In der ersten Phase wird die Mehrdeutigkeit der vermuteten Linkkandidaten aufgelöst. Hierbei können mögliche Linkkandidaten eine doppeldeutige Bedeutung haben. So kann zum Beispiel das Wort Note als Banknote oder Musiknote verstanden werden. Zu jedem der beiden Bedeutungen wird ein Wikipedia-Artikel angeboten. Abschließend werden in der zweiten Phase Links zu den gefundenen Schlagworten ermittelt.

#### Link-Begriffsklärungsphase

Um die Sinnhaftigkeit einzelner Begriffe festzustellen, werden die beiden Features *Commonness* und *Relatedness* verwendet. Hierbei beschreibt *Commonness* die Häufigkeit, wie oft auf

den Wikipedia-Artikel von Wikipedia im Vergleich zu den anderen möglichen Bedeutungen der Wörter verwiesen wird. Dagegen beschreibt *Relatedness* die Verbundenheit der Linkkandidaten mit dem Wikipedia-Artikel. Dazu werden die eingehenden und ausgehenden Links der beiden Artikel untersucht. Je höher der gemessene Wert ist, desto höher ist die semantische Gemeinsamkeit des Linkkandidaten zu dem untersuchten Artikel:

$$Relatedness(a, b) = \frac{\log(\max(|A|, |B|)) - \log(|A \cap B|)}{\log(|W|) - \log(\min(|A|, |B|))} \quad (3.1)$$

Hier sind  $a$  und  $b$  zwei ausgesuchte Artikel, wobei  $A$  und  $B$  die Sets aller Artikel sind, die auf die beiden Artikel verweisen.  $W$  ist das Set aller Artikel in Wikipedia. Die Verbundenheit der beiden Texte basiert auf dem durchschnittlichen Kontext der in Verbindung stehenden Artikel-Sets. Je höher der gemessene Wert ist, desto größer ist die semantische Gemeinsamkeit des Linkkandidaten mit dem untersuchten Wikipedia-Artikel.

#### **Link-Ermittlungsphase**

Die Link-Ermittlungsphase dient dazu herauszufinden, was für Terme verlinkt werden sollten. Hierfür wurde ein fünfteiliges Feature-Set bestehend aus Allgemeingültigkeit, Linkswahrscheinlichkeit, Lokation, Konfidenz der Begriffsklärung und Verbundenheit definiert. Hierfür steht die Allgemeingültigkeit des Linkkandidaten zu anderen Artikeln. Dabei definiert es den minimalen Abstand vom Thema des untersuchten Wikipedia-Artikels zu dem Kandidaten als aufgespannte Baumstruktur. Das nachfolgende Feature beschreibt Linkswahrscheinlichkeit für die Gesamtwahrscheinlichkeit eines Links. Dazu haben Linkkandidaten, die aus mehreren Wörtern bestehen eine höhere Linkswahrscheinlichkeit zugesagt. Damit würde die Wortkombination „angewandte Informatik“ besser bewertet, als nur das Wort Informatik. Als nächstes verdeutlicht das Feature der Lokation die Position des Linkkandidaten im Artikel. Demnach haben Verlinkungen in der Einleitung und am Ende des Textes einen höheren Wert für die Verständlichkeit des Wortlauts (vgl. [David, C., L. Giroux, S. Bertrand-Gastaldy, 1995](#)). Die Konfidenz der Begriffsklärung beschreibt den kombinierten Durchschnitt und maximalen Wert dafür, dass der Link in Beziehung zum Kontext steht. Zum Abschluss geht die Verbundenheit auf eine Beziehung der Linkkandidaten zu allen ausgewählten Linkkandidaten ein. Demnach werden die eingehenden und ausgehenden Links untersucht. Je größer die Ähnlichkeit der Verlinkungen des Linkkandidaten mit dem untersuchten Artikel ist, desto höher ist seine Verbundenheit zu dem Artikel.

### Untersuchung

Bei der Abschlussuntersuchung wurde Klassifikator C4.5 verwendet, da er bessere Zwischenergebnissen als Naive Bayes und SVN lieferte. Die endgültige Untersuchung des trainierten Klassifikatoren erfolgte anhand von 50 ausgewählten Nachrichtenartikeln. Dazu mussten 88 Probanden die Korrektheit der durch die Verwendung des trainierten Klassifikators generierten Links bestimmen. Die Tabelle 3.1 veranschaulicht die Resultate der durchgeführten Untersuchung. Die generierten Verlinkungen waren zu 76,4 Prozent als korrekt und zu 23,6 Prozent nicht richtig klassifiziert worden.

**Tabelle 3.1.:** Performance vom Algorithmus für die Linkerkennung in Prozent. [Milne und Witten \(2008\)](#)

correct	76.4
incorrect (wrong destination)	0.9
incorrect (irrelevant and/or unhelpful)	19.8
incorrect (unknown reason)	2.9

#### 3.4.2. Befüllung des Second Screens basierend auf der semantischen Verlinkung von Untertiteln

Das nachfolgende Paper beschreibt die Generierung von Wikipedia-Links aus Untertiteln von Talkshows als Zusatzinformationen für den Second Screen nach [Odijk u. a. \(2013\)](#). Dabei werden die Datensätze als Kontextgraph (siehe [Ferré und Rudolph, 2009](#)) modelliert. Es basiert auf den Erkenntnissen aus [Meij u. a. \(2012\)](#) nachdem die Autoren zu Twitter-Post Semantik Zusatzinformationen anhand der Kreuzreferenzierung von Schlagwörter erzeugt haben. Dazu übernahmen die Verfasser zu größten Teilen das gebrauchte Feature-Set. Darüber hinaus verwenden die Autoren aufgrund der besseren Resultate gegenüber Support Vektor Maschinen das Klassifikationsverfahren der Entscheidungsbäume *Random Forest* (siehe [Breiman, 2001](#)).

Der entwickelte Algorithmus funktioniert in drei Schritten. Zuerst werden zum Kontext passende Linkkandidaten gefunden. Danach wird ihnen ein Wert zugewiesen. Zum Abschluss wird die Reihenfolge im Kontextgraphen gegebenenfalls angepasst. Hierfür werden die Untertitel in einzelne Teile  $t_i$  aufgeteilt. Jedem gefundenen Linkkandidaten  $L$  wird ein Fixpunkt  $a$  zu einem bestimmten Wikipedia-Artikel  $w$  zugeordnet.

Im zweiten Schritt wird jedem der Linkkandidaten ein Wert zugeordnet. Zu diesem Zweck wird die Häufigkeit, wie oft auf einen bestimmten Wikipedia-Artikel verwiesen wird, gemessen:

$$\text{Commonness}(a, w) = \frac{|L_{a,w}|}{\sum_{w' \in W} |L_{a,w'}|} \quad (3.2)$$

Jedem gefundenen Linkkandidaten  $L$  wird ein Fixpunkt  $a$  zu einem bestimmten Wikipedia-Artikel  $w$  zugeordnet. Der Grundgedanke davon ist, dass Linkkandidaten mit Fixpunkten, die zum selben Wikipedia-Artikel verweisen, passendere Repräsentationen sind als Linkkandidaten, die oft auf andere Ziele verweisen.

#### **Umsortierung**

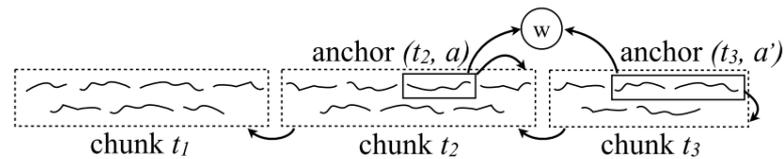
Zur Umsortierung der Linkkandidaten in dem zu modellierenden Kontextgraphen wird ein 26-teiliges Feature-Set verwendet. Dieses wird in vier Hauptteile unterteilt, die aus: Fixpunkt-, Ziel-, Kombination aus Fixpunkt- und Ziel- sowie Kontext-Feature bestehen.

Hierbei beziehen sich die Fixpunkte-Features auf den extrahierten Linkkandidaten in einem Textteil. Dazu wird unter anderem die Anzahl der Wikipedia-Artikel mit dem gleichen Titel wie der Fixpunktname mit betrachtet. Des Weiteren wird die Eintrittswahrscheinlichkeit, dass der Name vom Fixpunkt bei Wikipedia verwendet wird, einbezogen. Dagegen beziehen sich die Ziel-Features auf den eigentlichen gefundenen Wikipedia-Artikel. Hierfür werden die Anzahl der Verweise von Wikipedia selbst, sowie die Gesamtanzahl der Verlinkungen auf den Wikipedia-Artikel in der Berechnung verwendet. Außerdem wird Besucherzahl der Seite in den zeitlichen Perioden: Woche, Monat und Jahr mit betrachtet.

Anschließend wird im zweiten Hauptteil die Kombination aus Fixpunkt und Zielpunkt-Feature betrachtet. Hierfür wird untersucht, ob der Titel vom gefundenen Wikipedia-Artikel im Fixpunkt enthalten ist, sowie die Umkehrung davon. Der dritte Hauptteil betrachtet den Kontext, indem der Kontextgraph mit in die Bewertung hinzugezogen wird. Hierzu wird die Anzahl der Kanten, sowie die Zentralität im Kontextgraphen zu dem Wikipedia-Artikel ermittelt. Zum Abschluss wird die Metrik PageRank, die vergeben wird um die Bedeutung einer Seite im Internet zu bestimmen (siehe [Page u. a., 1998](#)), hinzugezogen.

#### **Kontextmodellierung**

Die Datensätze wurden von den Verfassern als Kontextgraph modelliert. Dazu veranschaulicht die Abbildung 3.7 einen Ausschnitt aus dem Kontextgraphen. Dieser besteht aus drei extrahierten Textteilen  $(t_1, t_2, t_3)$  sowie zwei verschiedenen Fixpunkten  $((t_2, a)$  und  $(t_3, a')$ ), die zu einem bestimmten Wikipedia-Artikel  $w$  verweisen.



**Abbildung 3.7.:** Ausschnitt aus dem gebildeten Kontextgraphen. [Odijk u. a. \(2013\)](#)

Das Verfahren zur Erstellung vom Kontextgraphen erfolgt in drei Schritten. Im ersten Schritt werden für jeden Textteil (*chunks*) mögliche Links generiert. Danach wird anhand der platzierten Grenze entschieden, ob ein Link hinzugefügt wird oder nicht. Im abschließenden Schritt wird geprüft, ob sich die Anzahl der Verlinkungen und damit das Kantengewicht zu einem bestimmten Wikipedia-Artikel ( $w$ ) als Knoten ändert. Damit wird die Position des Knotens im Kontext-Graphen geändert.

### Untersuchung

Die Abschlussuntersuchung erfolgte anhand eines Test-Sets. Dieser bestand aus Untertiteln mit rund 36 tausend Wörtern aus sechs Talkshow-Episoden. Dazu wurden für die extrahierten Textteile aus dem Test-Set mit Hilfe des trainierten Klassifikators Links generiert. Im Anschluss darauf wurde ein Goldstandard<sup>10</sup> etabliert, nachdem ein Annotationsspezialist manuell Links aus dem Test-Set generiert hat. Die Resultate der eingesetzten Verfahren wurden anschließend mit den Links aus dem Goldstandard verglichen. Eine Auflistung der erreichten Werte kann aus der Tabelle 3.2 entnommen werden.

**Tabelle 3.2.:** Zusammenfassung der semantischen Verlinkungs-Ergebnisse aus der Abschlussuntersuchung. [Odijk u. a. \(2013\)](#)

	Average classification time per chunk (in ms)	R-Prec	MAP
1. Baseline retrieval model	54	0.5753	0.6235
2. Learning to rerank approach	99	0.7177	0.7884
<i>Learning to rerank (L2R) + one context graph feature</i>			
3. L2R+DEGREE	104	0.7375	0.8252
4. L2R+DEGREECENTRALITY	108	0.7454	0.8219
5. L2R+PAGERANK	119	0.7380	0.8187
<i>Learning to rerank (L2R) + three context graph features</i>			
6. L2R+DEGREE+PAGERANK +DEGREECENTRALITY	120	0.7341	0.8204

<sup>10</sup>Goldstandard: Ist ein Ausdruck aus der wissenschaftlichen und medizinischen Umgangssprache. Er beschreibt ein Verfahren, welches unter konkurrierenden Verfahren in allgemeiner Auffassung als das Beste gilt.

Der Basisansatz hat mit einer durchschnittlichen Genauigkeit von 62 Prozent richtige Ergebnisse geliefert. Durch das Hinzuziehen der aufgestellten Merkmale konnte die durchschnittliche Genauigkeit um rund 17 Prozent gesteigert werden. Durch die Hinzuziehung der Kontextmerkmale konnte eine durchschnittliche Verbesserung von 21 Prozent erbracht werden, wogegen die Kombination aller Kontextmerkmale keine weitere Verbesserung der Genauigkeit gebracht hatte.

Die beiden vorgestellten Ansätze zeigen zwei Verfahren aus der algorithmischen-anwendungsorientierten Perspektive. Sie ermöglichen die Konstruktion von Systemen, die praktisch relevante Lernaufgaben, wie die Generierung von Links als Zusatzinformationen für einen adressierten Aspekt eines Textes, lösen können. Besonders das zweite Verfahren kann dazu genutzt werden Schlagworterkennung in Texten zu betreiben. Die gefundenen Wörter können für die Abfragegenerierung an Informationsdienste verwendet werden. Weitere vergleichbare Ansätze finden sich in [Mihalcea und Csomai \(2007\)](#), [Ratinov u. a. \(2011\)](#) und [Wang u. a. \(2012\)](#).

### 3.5. Personalisierungs-Ebene

Die Interessen von Menschen sind unterschiedlich. So können Informationen verschiedene Stellenwerte, je nachdem wem sie präsentiert werden, einnehmen. In diesem Abschnitt wird drauf eingegangen, wie anhand von Online-Ressourcen Empfehlungen, generiert werden können. Dazu bietet die Arbeit von [Bostandjiev u. a. \(2012\)](#) ein Empfehlungssystem *TasteWeight*, welches Vorhersagen auf Basis von Social-Media und Web-Ressourcen wie Wikipedia, Facebook und Twitter erzeugt. Das System implementiert hybride-Techniken zur Empfehlungsfindung und bietet eine grafische Oberfläche. Um die Komplexität zu verringern, wurde sich auf die Generierung von Musik-Empfehlungen konzentriert. Der Ansatz erfolgt in drei Schritten:

1. *Profil Initialisierung*: Es werden Benutzerprofile auf Basis der abgefragten Nutzerpräferenzen durch die Facebook Graph API<sup>11</sup> generiert. Diese ermöglicht den Zugriff auf die hinterlegten Facebook-Datensätze, falls die nötigen Rechte durch den Besitzer des Profils gegeben wurden. Dazu werden die Musik-Vorlieben abgefragt und jeweils mit einem Wert von 0,5 initialisiert.
2. *Modellierung der semantischen Ähnlichkeit*: Für jeden der drei verwendeten Kontext Ressourcen wird ein spezifisches Model erstellt. Dabei werden je nach dem verwendete

---

<sup>11</sup>Facebook Graph API - <https://developers.facebook.com/docs/graph-api>; letzter Zugriff: 29.10.2015

ten Model verschiedene Strategien verwendet. Diese werden im weiteren Verlauf des Abschnitts vorgestellt.

3. *Generierung von Empfehlungen*: Nachdem die relevanten Einträge erkannt wurden, werden Empfehlungen generiert. Hierzu wird die Summe der gewichteten Einträge, die zu einem Vorschlag verlinkt sind, gebildet.

#### Spezifische Modelle

*Wikipedia Model*: Die Facebook-Musikprofile werden zu Wikipedia-Artikeln zugeordnet. Dies erfolgt durch eine Suche mit Hilfe der GoogleSearch API. Dazu werden die ausgegebenen englischen Wikipedia-Artikel mit der besten Position unter den Google Suchergebnissen zu dem untersuchten Profil Element des aktiven Nutzers zugeordnet. Danach wird eine Anfrage an DBpedia<sup>12</sup>, um die zu dem Element passenden Artikel zu dem Profil des aktiven Nutzers herauszufinden, gesendet. DBpedia ist eine Open-Source-Software, um strukturierte Informationen aus Wikipedia zu extrahieren und Webanwendungen bereitzustellen. Daraufaufgehend ist es möglich Empfehlungen einem bestimmten Kontext zuzuordnen. Das passiert anhand der untereinander verlinkten Artikel. So kann zum Beispiel der Band Pink Floyd eine semantische Verlinkung zu der Kategorie „English Rock Music Groups“ zugeordnet werden. Die Band wie „The Beatles“, die derselben Kategorie zugeordnet ist, kann dann als Empfehlung verwendet werden.

*Facebook Model*: Für die Generierung von Vorhersagen werden Facebook-Freunde mit ähnlichem Geschmack verwendet. Hierfür wird allen Freunden eines Nutzers ein Rang anhand der Ähnlichkeit des Geschmacks zugeordnet. Dazu wird die folgende Formel verwendet:

$$W_{friends_i} = \frac{TWCI_{user,friend_i}}{\sqrt{TWI_{user}^2 * TWI_{friend_i}^2}} \quad (3.3)$$

Demnach ist  $TWCI_{user,friend_i}$  die Gesamtgewichtung eines Nutzers zu einem Freund  $i$  und  $TWI_x$  die Gesamtgewichtung aller gemochten Elemente einer Person.

*Twitter Model*: Anhand von rausgesuchten Twitter-Nutzern, die Erfahrungen mit der gehörten Musik des aktiven Nutzers haben, sollen Experten-Empfehlungen generiert werden. Dazu wird jedes gemochte Element eines aktiven Nutzers in einen Twitter-Hashtag umgewandelt (beispielsweise wird aus Michel Jackson der Hashtag #micheljackson). Danach wird mit Hilfe

---

<sup>12</sup>DBpedia - <http://de.dbpedia.org/sparql>; letzter Zugriff: 29.10.2015

eines Portals nach dem Hashtag gesucht. Für jeden gefundenen Experten wird eine Gewichtung nach der folgenden Formel gebildet:

$$W_{exp_i,item_j} = \frac{|Exp_{item_j}| - Rank_{exp_i,item_j}}{|Exp_{item_j}|} \quad (3.4)$$

Hierbei steht  $Rank_{exp_i,item_j}$  für Rang eines Experten und  $|Exp_{item_j}|$  ist die Anzahl der Experten zu dem jeweiligen Element. Beispielsweise würde ein Experte mit einem Rang von 20 aus 100 Experten für ein spezifisches Element eine Wertung von 0,8 bekommen. Die Gesamtgewichtung eines Twitter-Experten wird dann durch die folgende Linearkombination gebildet:

$$W_{exp_i} = \sum_{Linked(proj_j,exp_i)} (W_{proj_j} * S_{exp_i,proj_j}) \quad (3.5)$$

In diesem Abschnitt wurde ein Ansatz zur Entwicklung eines Empfehlungssystems auf Basis von Social-Media und Web-Ressourcen wie Wikipedia, Facebook und Twitter vorgestellt. Es wurden Musikempfehlungen generiert. Dennoch ist es denkbar einen angepassten Ansatz für die Generierung von Empfehlungen für bestimmte Informationsgebiete auf Basis der abgelegten Interessen zu bilden. Weitere Arbeiten, welche sich mit der Generierung von Empfehlungen beschäftigen, können den folgenden Arbeiten entnommen werden: [Shi u. a. \(2010\)](#), [Gantner u. a. \(2010\)](#) und [Felfernig und Ninaus \(2012\)](#).

## 3.6. Anforderungen an SecondCast

Auf Basis der gewonnen Eindrücke aus den vorgestellten Arbeiten sollen Anforderungen an das zu entwickelnde System mit weiteren Funktionalitäten abgeleitet werden. Ziel ist es den Systemkontext durch Auflistung der folgenden Anwendungsfälle abzuleiten und eine Fokussierung auf das Wesentliche im System zu ermöglichen. In diesem Zusammenhang wird auf die beteiligten Benutzergruppen und die Funktionalitäten für Smartphone-Applikation und Serveranwendung eingegangen. Die aufgelisteten Anforderungen können sich im Laufe des Projektes ändern, somit sind sie als Leitwert zu verstehen.

### 3.6.1. Rollen

In diesem Abschnitt wird auf die Benutzergruppen eingegangen, die das System später bedienen sollen. Es wird in diesem Fall zwischen den beiden Benutzergruppen Prüfer und Probanden unterschieden. In weiteren Verlauf werden die beiden Benutzergruppen näher veranschaulicht.

#### **Prüfer**

Der Prüfer stellt den Hauptanwender dar, der eine oder mehrere Hypothesen anhand der Plattform untersuchen möchte. Dazu muss ihm die Möglichkeit der Verwaltung von Datensätzen für die Prüfungsvorbereitung sowie Aufzeichnung von Aktionen gegeben werden.

#### **Probanden**

Ein Proband stellt den Benutzer dar, der eine Prüfung absolvieren muss. Dazu hat der Proband nur eine Applikation auf dem Smartphone oder Tablet zur Verfügung. Seine Bedürfnisse müssen direkt in die Funktionalitäten und die Gestaltung der Ansichten des SmartCast-Clients einfließen.

### **3.6.2. Funktionale Anforderungen**

#### **Anforderungen an die Smartphone Applikation:**

1. *Abspielen von Videos*: Die Applikation soll dem Probanden ermöglichen das ausgesuchte Video auf dem Fernseher abzuspielen.
2. *Fernbedienung*: Die Applikation muss dem Probanden die Möglichkeit bieten das abgespielte Video zu abzuspielen, stoppen, pausieren und die Lautstärke zu verändern.
3. *Austausch von Informationen mit Personen*: Die Applikation muss dem Probanden die Möglichkeit bieten, sich mit anderen Personen über das Geschehen auszutauschen.
4. *Bereitstellung von Zusatzinformation*: Die Anwendung muss Zusatzinformationen zu dem Geschehen für den Probanden in einer lesbaren Form liefern können.
5. *Tracking von Nutzeraktionen*: Die Anwendung muss dem Prüfer die Möglichkeit bieten alle Aktionen des Probanden zu verfolgen um sie später auszuwerten.

#### **Anforderungen an die Serveranwendung:**

1. *Zusatzinformationsbeschaffung*: Die Plattform muss dem Prüfer die Möglichkeit bieten, Informationen von Informationsdiensten zu erfragen.
2. *Informationsverwaltung*: Die Plattform muss dem Prüfer die Möglichkeit bieten, vorhandene Informationen zu verwalten (anlegen, bearbeiten und löschen).
3. *Textanalyse*: Die Plattform muss dem Prüfer die Möglichkeit bieten, den Kontext eines Textes zu erfassen.

### 3. Analyse

---

4. *Textzusammenfassung*: Die Plattform muss dem Prüfer die Möglichkeit bieten, Texte zusammenzufassen.
5. *Personalisierung von Informationen*: Die Plattform muss dem Prüfer die Möglichkeit bieten, Zusatzinformationen an die persönlichen Präferenzen der Probanden anzupassen.

#### 3.6.3. Technische Anforderungen

1. *Wiederverwendbarkeit*: Die zu entwickelnden Komponenten von SecondCast sollen so konzipiert werden, dass es möglich wäre sie in anderen Projekten zu verwenden.
2. *Erweiterbarkeit*: Das Gesamtsystem ist so zu konzipieren, dass es möglich ist weitere Komponenten zu integrieren.
3. *Portierbarkeit*: Es soll gewährleistet werden, dass die einzelnen Systemteile plattformunabhängig verwendet werden könnten.
4. *Wartbarkeit*: Änderungen an dem Softwaresystem sollen mit einem angemessenem Aufwand umsetzbar sein.
5. *Austauschbarkeit*: Es soll möglich sein die einzelnen Komponenten von SecondCast mit anderen in der Funktionalität gleichen oder höherwertigen Systemen zu ersetzen.

### 3.7. Fazit

Das Ziel der Analyse war es, Problemstellungen bei der Entwicklung von Second Screen Anwendungen zu analysieren und mögliche Lösungsansätze zu präsentieren. Anhand der gewonnenen Eindrücke sollten fachliche und technische Anforderungen an SecondCast spezifiziert werden. Dazu wurden mögliche Problemstellungen von Second Screen in fünf Teilbereiche eingeteilt.

Auf der psychologischen Ebene konnte anhand von mehreren durchgeführten Studien die bedingte Multitasking-Fähigkeit der Menschen aufgezeigt werden. Darauf folgend wurde ein technischer Ansatz zur Kommunikation von zwei beteiligten Mediengeräten vorgestellt. Dieser hat verdeutlicht, welche technischen Anforderungen an ein Kommunikationsprotokoll für Heimnetzwerk-Szenarien gestellt werden sollten. Als nächstes wurden die Erwartungen von Nutzern an Second Screen Anwendungen anhand von zwei Designstudien analysiert. Es ist aufgefallen, dass die Nutzer neben personalisierten Inhalten zum Geschehen eine Fernbedienung-Funktion gewünscht haben, welche es ihnen erlaubt Mediengeräte zu steuern.

### 3. Analyse

---

Auf der kontextuellen Ebene wurden Ansätze zur Schlagworterkennung aus Texten für die Generierung von Anfragen vorgestellt. Besonders der Ansatz zum Erlernen des Algorithmus der Wikifizierung ist interessant. Zum Schluss wurden Ansätze zur Personalisierung von Informationen vorgestellt. Diese stellen einen Beitrag zur Wertsteigerung der ausgelieferten Inhalte dar. Der Aspekt von Personalisierung der Informationen für den jeweiligen Benutzer soll nur architektonisch vorgesehen, aber nicht realisiert werden, weil es sonst den Rahmen der Arbeit sprengen würde.

Aus den Eindrücken wurden technische und fachliche Anforderungen an SecondCast definiert. Die funktionalen Anforderungen wurden in Erwartungen an die Smartphone Applikation und Serveranwendung eingeteilt. Für beide Seiten muss ein GUI-Konzept entwickelt werden, damit die Nutzer die Anwendungen möglichst einfach grafisch bedienen können. Außerdem muss Textanalyse in der Serveranwendung, um den Kontext eines Textes zu erfassen, ermöglicht werden. Es muss ein strukturiertes Archiv als Informationsdienst gefunden werden, welcher eine große Themenvielfalt abdeckt. Des Weiteren müssen Texte, um den Nutzer nicht zu überfordern, gekürzt werden.

Um die in der Zielsetzung aufgestellte Hypothese (siehe Abschnitt 1.1) zu testen muss eine Untersuchungsplattform realisiert werden. Dazu stellen die Resultate aus der durchgeführten Analyse eine gute Grundlage für die weitere Konzeption und anschließende Realisierung von SecondCast dar. Das Design der Anwendung soll in dem Masse erfolgen, dass die angestrebte Untersuchung ermöglicht wird.

## 4. Design

Die Konzeption der Architektur von SecondCast erfolgt nach dem Leitspruch des amerikanischen Systemingenieurs Rehtin:

*„Wenn du es nicht in fünf Minuten erklären kannst, hast du es entweder nicht verstanden oder es funktioniert nicht.“* Rehtin und Maier (2000)

Demnach wird der klassische und systematische Ansatz zur Beherrschung von Komplexität „teile und herrsche“ (*divide et impera*) angewendet und das angestrebte System in kleinere Teile zerteilt. Jedes Systemteil übernimmt dabei eine bestimmte fachliche oder technische Funktion.

Dieses Kapitel dient dabei als zielgerichtete Vorbereitung für die Realisierung von der Untersuchungsplattform SecondCast. Im Abschnitt 4.1 wird eine konzeptionelle Übersicht auf das zu entwickelnde System geliefert. Der nachfolgende Abschnitt 4.2 zeigt den Aufbau des Systems, als Gesamtansicht in Form eines Komponentendiagramms auf. Danach wird auf die einzelnen Systemkomponenten eingegangen. Um die Interaktion der Komponenten zu verdeutlichen wird die Kommunikation der Systemteile im Abschnitt 4.3 veranschaulicht. Darauf folgend wird die technische Infrastruktur im Abschnitt 4.4 vorgestellt. Der vorletzte Abschnitt 4.5 dreht sich um die Konzeption der grafischen Benutzeroberfläche. Zum Ende des Kapitels wird ein Fazit im Abschnitt 4.6 gezogen.

### 4.1. Kontextsicht

Auf Basis der durchgeführten Anforderungsanalyse wurde eine Kontextabgrenzung bezüglich des Umfelds vom System, sowie dessen Zusammenhang mit seinem Umfeld durchgeführt. Dazu stellt Abbildung 4.1 eine Kontextabgrenzung von fachlicher Sicht des Systems aus der Vogelperspektive dar. Sie verschafft dabei ein grobes Gesamtbild von der Kommunikation beteiligter Systeme. Des Weiteren dient sie als Diskussionsgrundlage, um mit Experten über das System zu diskutieren und damit mögliche Schwachstellen frühzeitig möglichst vor der Entwicklung zu erkennen.

Das Kontextdiagramm besteht aus fünf interagierenden Systemen, einem Probanden und Prüfer. Ein Prüfer, welcher eine Untersuchung starten möchte, kann für eine in der Mediathek befindenden Video eine Reihe von für das Experiment notwendigen Zusatzinformationen angeben.

Der Proband startet die Applikation auf seinem Smartphone und bekommt eine Liste von Videos angezeigt. Damit die Videos angezeigt werden können muss die Liste von Medien mit Informationen von einer Mediathek gezogen werden. Das vom Benutzer aufgerufene Video wird dann an den Media-Streaming-Adapter zum Fernseher gesendet. Die Zusatzinformationen zu der Dokumentation werden vom SecondCast-Server angefragt. Falls keine Informationen zu dem Medieninhalt vorhanden sind erfolgt eine Abfrage an einen Informations-Service.

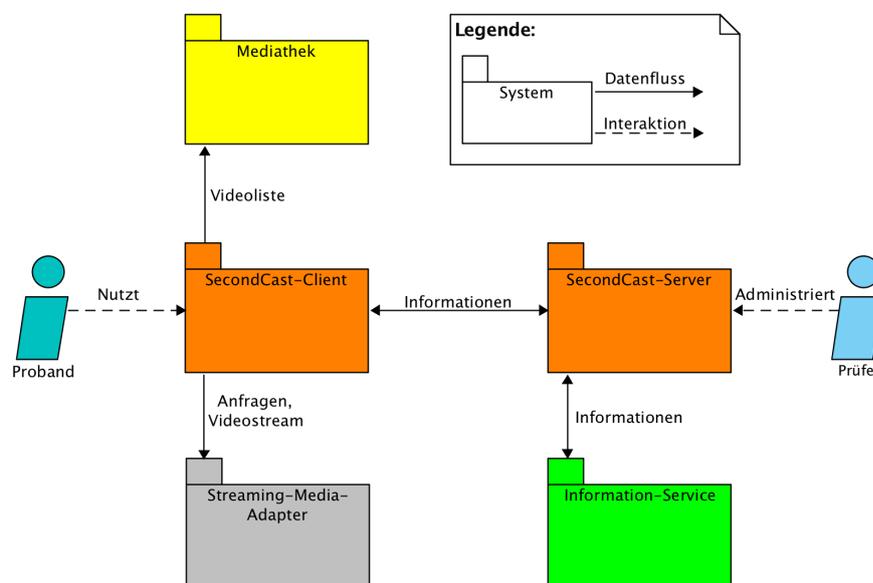
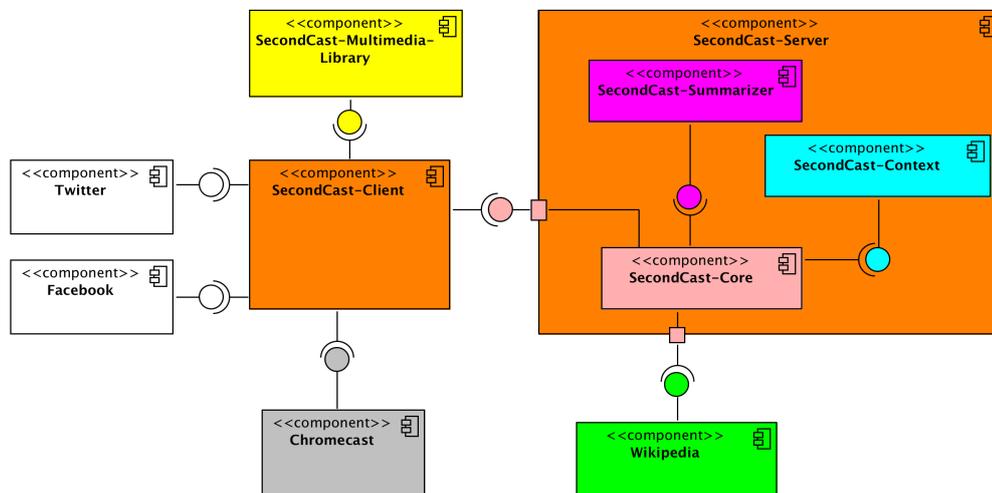


Abbildung 4.1.: Diagramm zur Kontextabgrenzung.

## 4.2. Bausteinsicht

Die geforderten Aufgaben des Systems werden auf Komponenten abgebildet. Sie sollen die Systemstruktur und Zusammenhänge zwischen den Bausteinen des Systems explizit machen. Die angestrebte Gesamtansicht vom SecondCast-System ist in Abbildung 4.2 dargestellt. Dabei besteht die Untersuchungsplattform aus den zwei Hauptkomponenten SecondCast-Client und SecondCast-Server (Farbe Orange). Clientseitig wird neben Social-Media-Diensten (Farbe Weiß) auf eine Mediathek von Dokumentationen (Farbe Gelb) zugegriffen und für das Ab-

spielen der Videos auf dem First Screen der Streaming-Media-Adapter Chromecast (Farbe Grau) verwendet. Serverseitig erfolgt die Kommunikation über den SecondCast-Core (Farbe Pink). Dieser verwendet eine Kontext-Komponente (Farbe Türkis) zur Extraktion von Schlagwörtern aus Texten, die für Informationsabfragen von Informationsdiensten (Farbe Grün) wie Wikipedia verwendet werden. Als gepflegtes Archiv bietet Wikipedia über eine REST-Schnittstelle den Zugriff auf rund 1,7 Millionen deutschsprachige Artikel (vgl. [Wikimedia, 2015](#)). Der SecondCast-Summerizer (Farbe Lila) soll die erhaltenen Texte für den Nutzer zusammenfassen. Des Weiteren ist SecondCast-Core für die Aufbereitung und Persistierung von Datensätzen verantwortlich. In den nachfolgenden Unterabschnitten wird konkreter auf die einzelnen Komponenten eingegangen. Die getroffene Farbwahl für die Systemteile zieht sich dabei durch den weiteren Verlauf der Arbeit.



**Abbildung 4.2.:** Gesamtansicht von der Second Screen Untersuchungsplattform SecondCast als Komponentendiagramm.

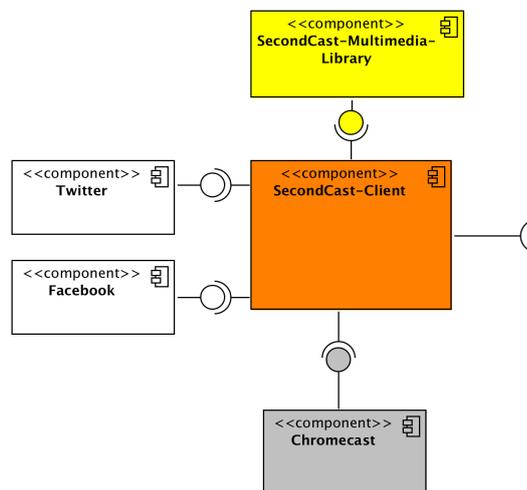
#### 4.2.1. SecondCast-Client

Die Rolle des Second Screens soll von SecondCast-Client übernommen werden. Hierfür beschreibt die Abbildung 4.3 die konzeptionelle Sicht auf SecondCast-Client. Die Anwendung ist dabei zu Informationsaufbereitung, Austausch mit anderen Nutzern und als Fernbedienung gedacht. Sie wird aufgrund eines globalen Marktanteils von rund 80 Prozent im ersten Quartal 2015 (vgl. [IDC Corporate USA, 2015](#)), als Android Applikation konzipiert. Zudem erlaubt Android durch die geringen Einstiegshürden bei der Verwendung von Android-Geräten eine einfachere Implementierung.

Die angebotenen Dokumentationsbeschreibung und Dokumentationsnamen sollen während des Verarbeitungsprozesses an SecondCast-Core gesendet werden. Die Kommunikation zwischen Client und Server erfolgt vorrangig über das REST-Protokoll.

Um die vom Nutzer ausgewählten Dokumentationen auf dem First Screen anzuzeigen zu können soll der Streaming-Media-Adapter Chromecast (Farbe Grau) verwendet werden. Der Adapter bietet eine Schnittstelle an, mit dem Videos abgespielt, gestoppt und die Lautstärke verändert werden kann. Die Arbeitsweise von Chromecast wird im Abschnitt 2.1 veranschaulicht.

Zum Informationsaustausch mit anderen Nutzern und für eine mögliche Personalisierung der Zusatzinformationen stehen die beiden Sozialen Netzwerke (Farbe Weiß) Twitter und Facebook zur Verfügung. Die Netzwerkschnittstellen sollen dazu an SecondCast gekoppelt werden.

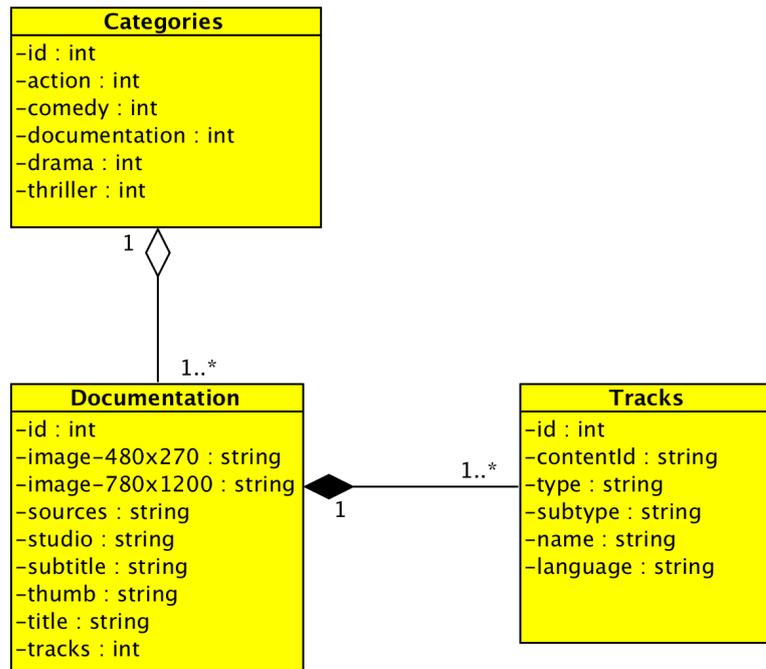


**Abbildung 4.3.:** Komponentendiagramm der möglichen beteiligten Komponenten von SecondCast-Client.

#### 4.2.2. SecondCast-Multimedia-Library

Die SecondCast-Multimedia-Library soll alle nötigen Informationen für das Abspielen und die Repräsentation von Videos auf dem SecondCast-Client liefern. Die Abbildung 4.4 verdeutlicht das Datenbankschema der persistierten Datensätze als Klassendiagramm. Insgesamt werden drei Datenbanktabellen: Kategorien, Dokumentationen und Spuren angezeigt. Eine Kategorie kann verschiedene Genres von Action bis Thriller abdecken. Diese stellen pro Kategorie eine Datenbanktabelle mit einem Verweis auf die ID des jeweiligen Videos dar. Im Falle der

Dokumentationen sollen Informationen als Verweise von auf die abgelegten Vorschaubilder, Beschreibungstext, Stream-URL, Title und Spurverweis, welche als Untertitel verwendet werden, fungieren. Einem Video können dabei verschiedene Spuren zugeordnet werden. Diese bietet neben einem Verweis auf die abgelegte Web-Video-Text Tracks-Datei einen Namen und eine Sprachangabe.



**Abbildung 4.4.:** Klassendiagramm des Datenbankschemas von SecondCast-Multimedia-Library.

#### 4.2.3. SecondCast-Core

Um eine starke Kohäsion nach dem Single-Responsibility-Prinzip zu gewährleisten wurde SecondCast-Server in einzelne Aufgabenbereiche aufgeteilt. Hierfür dient SecondCast-Core (siehe Abbildung 4.5) als Kernkomponente für Aufbereitung, Kommunikation und Persistierung der Datensätze.

Für die Annahme von Anfragen soll eine REST-API angeboten werden. Neben dem Empfang von Erhebungen eines SecondCast-Clients sollen, falls keine Datensätze in der Datenbank zu der jeweiligen Dokumentation enthalten sind, Zusatzinformationen aus der beigefügten Dokumentationsbeschreibung erzeugt werden. Hierfür sendet der SecondCast-Core den Text die REST-API von SecondCast-Context, als Antwort wird eine Entitätenliste erwartet. Anhand

der Ergebnisse mögen Anfragen an Informationsdienste generiert werden. Die erhaltenen Informationen werden aufbereitet und in der Datenbank abgelegt. Danach sollen die Datensätze gebündelt an SecondCast-Client versendet werden.

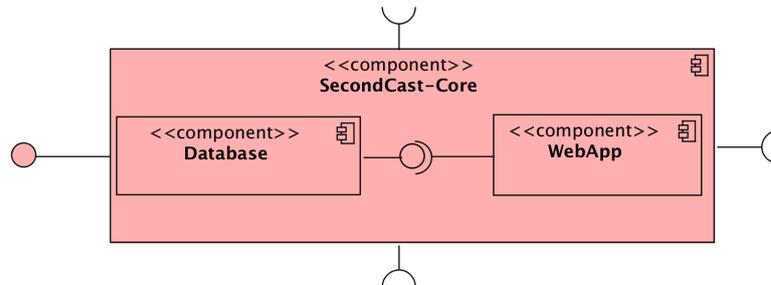


Abbildung 4.5.: Komponentendiagramm von SecondCast-Core.

### 4.2.4. SecondCast-Summarizer

Da, wie im Abschnitt 3.5 gezeigt, die Menschen nur bedingt Multitasking fähig sind und nur eine begrenzte Aufmerksamkeitspanne haben, soll die Informationsaufnahme erleichtert werden. Dazu müssten die gesammelten Texte zusammengefasst werden. Für diese Aufgabe soll der SecondCast-Summarizer, wie in Abbildung 4.6 dargestellt, dienen. Die Kommunikation mit der Komponente soll anhand einer REST-Schnittstelle erfolgen. Hierfür soll der Text, mit der Angabe ihn in zehn Sätzen zusammenzufassen, zu dem SecondCast-Summarizer gesendet werden.

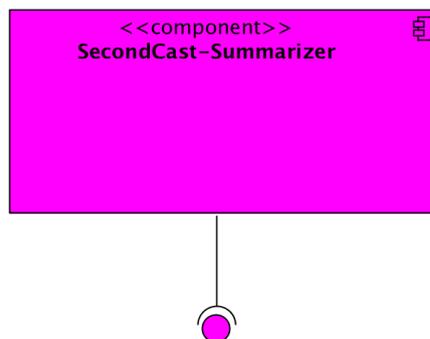


Abbildung 4.6.: Komponentendiagramm von SecondCast-Summarizer.

### 4.2.5. SecondCast-Context

Da für die ausgesuchten Dokumentationen nur eine Dokumentationsbeschreibung zur Verfügung steht, sollen weitere Zusatzinformationen anhand der Schlagwörter aus dem Text erzeugt

werden. Hierfür dient die in Abbildung 4.7 dargestellte Komponente SecondCast-Context. Sie besteht aus den zwei Hauptkomponenten Dexter-Core und Dexter-Webapp. Der Dexter-Core soll in mehreren Teilschritten ein Entität-Link-Graphen aus 1,7 Millionen Wikipedia-Artikeln aus dem Wikipedia-Dump von 07.06.2015 erstellen. Unter anderem muss mit Hilfe von Apache Lucene ein Artikel-Index generiert werden. Die Dexter-Webapp bietet eine REST-API mit der die Anfragen an Dexter-Core weitergeleitet werden. Die Funktionsweise und Aufbau von Dexter wird im Abschnitt 2.2 veranschaulicht.

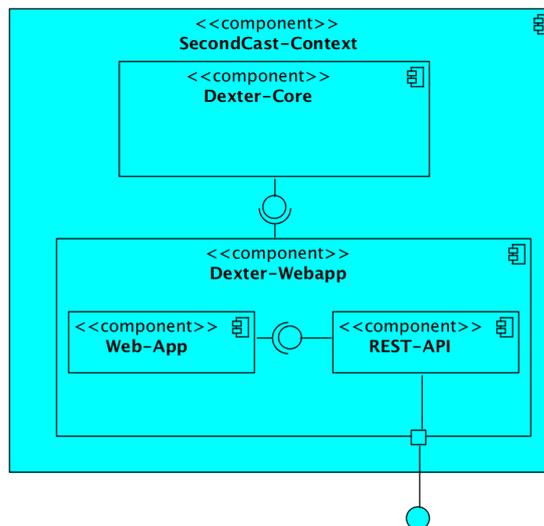


Abbildung 4.7.: Komponentendiagramm von SecondCast-Context.

### 4.3. Kommunikation der Architekturelemente

In diesem Abschnitt wird beschrieben wie die Architekturelemente sich bei dem Hauptanwendungsfall verhalten sollen. Die Initialisierung soll nach dem Starten von SecondCast-Client durch einen Tap auf das Icon vom Benutzer erfolgen. Die Abbildung 4.8 verdeutlicht den Kommunikationsprozess. Demnach wird von SecondCast-Client eine Videoliste mit Informationen wie im Abschnitt 4.2.2 angefragt. Die erhaltene Videostream-URL wird an Chromecast gesendet. Die empfangene Dokumentationsbeschreibung wird mit dem Namen der Dokumentation und einer Angabe zur Sprache an SecondCast-Core versendet. In diesem Fall sind noch keine Zusatzinformationen zu der Dokumentation abgelegt. Daher wird der Beschreibungstext an SecondCast-Context versendet. Dieser generiert eine Liste von Entitäten und sendet sie zurück. Aus den erhaltenen Schlagwörtern werden dann Anfragen an Wikipedia generiert. Nachdem der Informationsdienst alle Anfragen beantwortet hat, werden die Ergebnisse gebündelt an

SecondCast-Summarizer für die Zusammenfassung der Texte versendet. Die zusammengefassten Texte werden zurück an die Kernkomponente gesendet. Anschließend erfolgt die Versendung der aufbereiteten Ergebnisse an den SecondCast-Client. Die mobile Anwendung führt, eine Informationsaufbereitung für die Darstellung der Zusatzinformationen für den Benutzer auf dem Second Screen, durch.

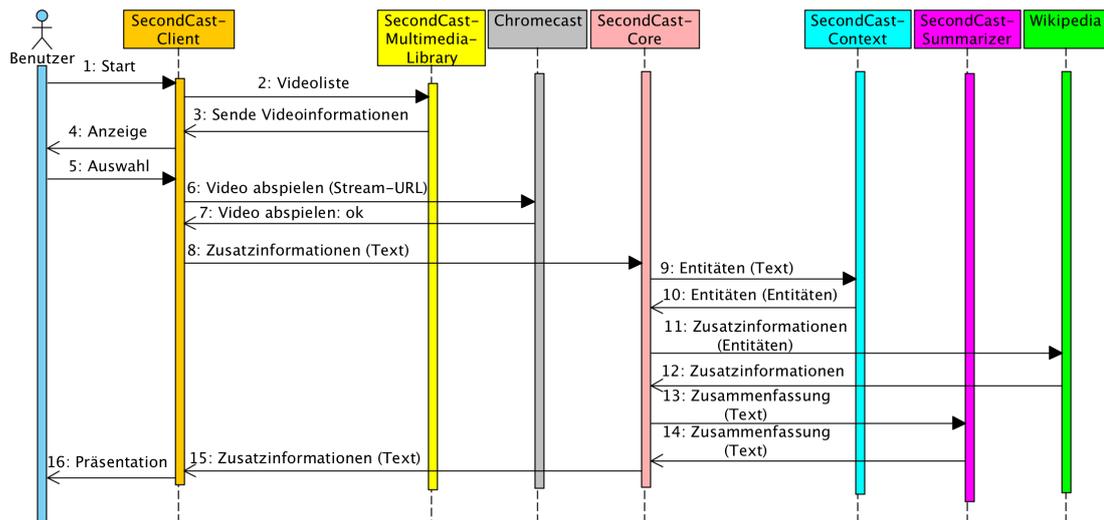


Abbildung 4.8.: Kommunikation der Architekturelemente für die Anzeige einer Dokumentation mit Bereitstellung der Zusatzinformationen auf dem Second Screen.

#### 4.4. TI-Architektur

Ohne Hardware und Infrastruktur kann Software nicht in Betrieb genommen werden. In diesem Abschnitt soll auf die Netzwerktopologie eingegangen werden. Die Architektur der technischen Infrastruktur von der Untersuchungsplattform SecondCast ist als Verteilungsdiagramm in der Abbildung 4.9 dargestellt. Sie besteht aus sieben Knoten: Multimediaserver, Smartphone, Chromecast, SecondCast-Server, SecondCast-Context, SecondCast-Core und Wikimedia. Die Kanäle, welche die Knoten verbinden, repräsentieren die physikalischen Übertragungswege, auf denen Informationen zwischen den Knoten ausgetauscht werden können.

Der Knoten Smartphone steht für die Menge von mobilen Geräten auf welchen der SecondCast-Client läuft. Dieser kommuniziert über HTTPS mit SecondCast-Server, Multimediaserver und Chromecast. Um möglichst eine breite Masse an Multimediageräte zu erreichen, soll die Applikation unter dem Betriebssystem Android lauffähig sein. Auf dem Knoten SecondCast-Server laufen die beiden Laufzeitumgebungen SecondCast-Core und SecondCast-Context,

welche über HTTPS miteinander kommunizieren. Als Informationsdienst wird Wikimedia verwendet.

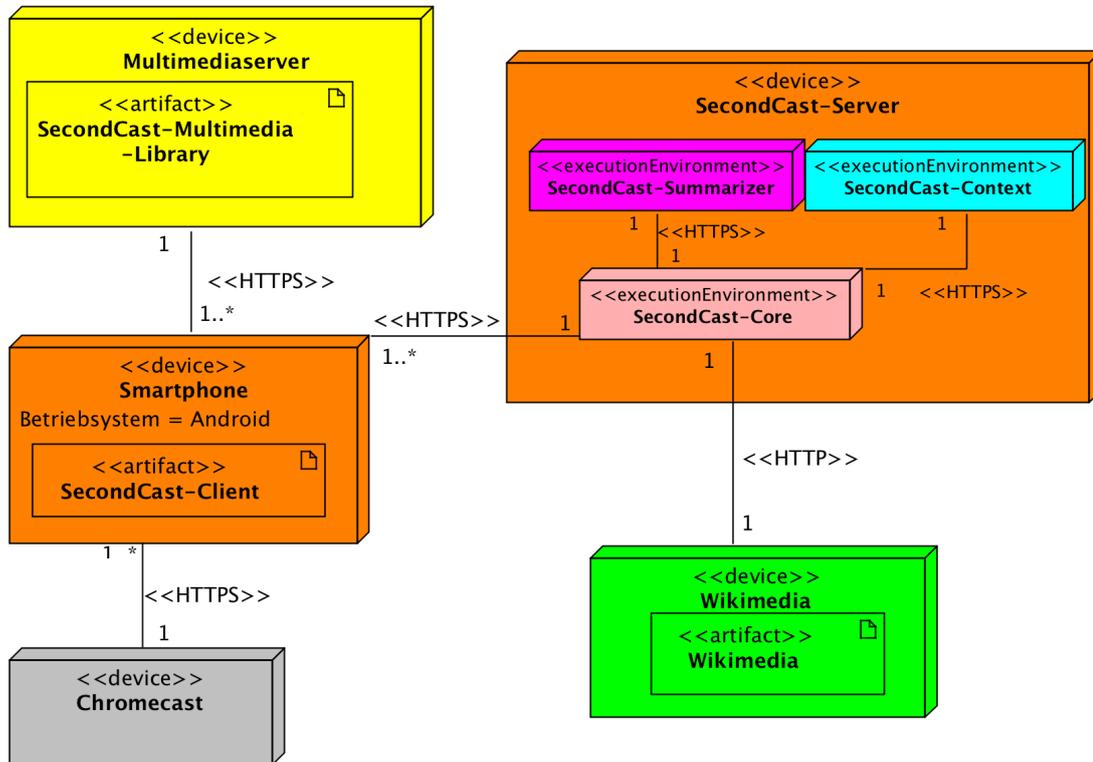


Abbildung 4.9.: TI-Architektur von SecondCast als Verteilungsdiagramm

## 4.5. Konzeption der grafischen Benutzeroberfläche

Für die Konzeption der grafischen Benutzeroberfläche werden Wireframes verwendet. Diese sind eine Art Prototyp einer Anwendung und stellen seine Funktionsweise dar. Sie reduzieren die Außenform eines Elementes auf das Wesentliche und abstrahieren die Bedienelemente wie Schaltflächen und Listeneinträge auf einfache Rechtecke (vgl. Franke und Ippen, 2012). Die einzelnen Wireframes beschreiben jeweils eine Ansicht eines Anwendungsfalls und bilden in der Gesamtheit die Details für eine User Story. Die User Stories stellen dabei Funktionalitäten dar, welche für einen Nutzer oder Käufer der Software oder Systems von Wert sind (vgl. COHN, 2010). Im Folgenden Abschnitt wird auf den Hauptanwendungsfall, der Anzeige von Dokumentationen mit dazugehörigen Repräsentation von Zusatzinformationen auf dem Second Screen für die im Verlauf folgende Untersuchung, eingegangen. Des Weiteren wird die bedeutsame Ansicht für die Verwaltung der Zusatzinformationen durch den Prüfer aufgezeigt.

#### 4.5.1. User Story „Dokumentation mit Zusatzinformationen aufrufen“

Die angestrebte Untersuchung soll anhand von aufgezeigten Dokumentationen mit der Verbindung von Informationen aus dem strukturierten Archiv Wikipedia erfolgen. Dazu stellt die Abbildung 4.10 den Hauptanwendungsfall von SecondCast dar. Die Pfeile zwischen den einzelnen Wireframes verdeutlichen die Klick-Ereignisse der Nutzer. Hierbei wird der Nutzer mit einer Auswahlliste begrüßt, wo er zwischen den Einträgen: Dokumentationen, Info und Tutorial wählen kann. In diesem Fall wählt der Benutzer die Dokumentationen aus. Durch das Anklicken des Listeneintrags Dokumentationen gelangt der Nutzer zu der Ansicht aller Beiträge zu diesem Typ. Danach kann sich der Nutzer mit Hilfe des Chromecast-Knopfes mit dem jeweiligen Streaming-Media-Adapter verbinden, auf dem er den Beitrag anschauen möchte. Anschließend wird das Video auf dem Fernseher angezeigt. Während der Beitrag abgespielt wird, klappen sich die Informationen zu der Dokumentation auf. Mit einer *Swipe-Right* Geste<sup>13</sup>, kann sich der Nutzer eine Fernbedienung anzeigen mit der er den Beitrag Pausieren, wieder abspielen und die Lautstärke ändern kann.



Abbildung 4.10.: Storyboard für den Hauptanwendungsfall der Anzeige von Dokumentationen auf dem First Screen, sowie der Bereitstellung von Zusatzinformationen auf dem Second Screen

<sup>13</sup>Swipe-Right Geste: Bezeichnet das schnelle Wischen mit dem Finger zur rechten Seite.

#### 4.5.2. User Story „Zusatzinformationen verwalten“

Die Verwaltung der gesammelten Informationen soll für den Prüfer so einfach wie möglich anhand einer grafischen Oberfläche erfolgen. Hierfür wird die konzipierte Ansicht für die Verwaltung von Zusatzinformationen in Abbildung 4.11 angezeigt. Die Einflussnahme auf die Datensätze soll über eine Tabelle erfolgen. Hierfür müssten die Informationen editiert, gelöscht oder neue Informationseinträge angelegt werden können. Über die Navigationsleiste sollen weitere Entitäten und die Benutzerinformationen verwaltet werden können.

ID	Title	Information	Action
1	Macht.Mensch.Sc...	[[{"title": "Ausbildung und Beruf", "information": "..."}, {"title": "Politische Positionen", "information": "..."}]]	<div>Edit</div> <div>delete</div>
2	Der Tod von Lam...	[[{"title": "Lampedusa", "information": "..."}]]	<div>Edit</div> <div>delete</div>

Abbildung 4.11.: Zusatzinformationsansicht von SecondCast-Core.

## 4.6. Fazit

Dieses Kapitel dient als konzeptionelle Grundlage und zielgerichtete Vorbereitung für die fachliche, technische und technologische Realisierung von SecondCast. Das zu entwickelnde System ist als Client-Server-Anwendung konzipiert und besteht damit aus zwei Hauptkomponenten.

Daneben gibt es noch drei weitere Komponenten, die Funktionalitäten der Informationsquellen für Videos und Zusatzinformationen, sowie der Schnittstelle für einen Fernseher darstellen. Auf die einzelnen Komponenten wurde in diesem Kapitel näher eingegangen. Als Informationsquelle für Videos wurde der Multimediaserver SecondCast-Multimedia-Library konzipiert. Um Videos auf einem Fernseher abzuspielen, wird Chromecast verwendet. Der Streaming-Media-Adapter bietet die nötigen Funktionalitäten, um neben der benötigten Abspielfunktionalität auch die Fernbedienungsfunktionalität zu erfüllen. Die Zusatzinformationen werden von dem Informationsdienst Wikipedia bezogen. Der Informations-Service bietet mit der angebotenen REST-Schnittstelle einen Zugriff auf rund 1,7 Millionen deutschsprachige Artikel (Stand August 2015).

Auf die Konzeption der grafischen Oberfläche von SecondCast-Client wurde anhand eines Storyboards eingegangen. Hierfür wurde auf die geforderte Funktionalität für das Anzeigen einer Dokumentation mit der Bereitstellung von Zusatzinformationen auf dem Second Screen vorgestellt. Des Weiteren wurde auf die Zusatzinformationsansicht in Form eines Mockups veranschaulicht. Abschließend kann festgestellt werden, dass die Brauchbarkeit der Ergebnisse aus Konzeption, sich im Realisierungs-Kapitel zeigen wird.

## 5. Realisierung

In diesem Kapitel werden die angestellten Überlegungen aus der Konzeption anhand der Umsetzung einzelner Systemteile überprüft. Hierfür wird im Abschnitt 5.1 die Realisierung der mobilen Anwendung anhand des erarbeiteten GUI-Konzeptes vorgestellt. Danach wird im Abschnitt 5.2 auf die einzelnen erforderlichen Komponenten für die Serveranwendung eingegangen. Zum Ende des Kapitels wird im Abschnitt 5.3 ein Fazit bezüglich der Schwierigkeiten bei der Umsetzung und der Bereitschaft des Systems für die angestrebte Fallstudie gezogen.

### 5.1. SecondCast-Client

Um eine mobile Applikation zu entwickeln gibt es drei gängige Möglichkeiten: Web-Apps, hybride Apps und native Apps (vgl. BACK u. a., 2009). Die Web-App stellt die Möglichkeit dar das Softwareprodukt auf Basis von HTML5, CSS3 und modernen JavaScript-Frameworks als eine sich an die Bildschirmgröße anpassende Anwendung zu entwickeln. Der Vorteil davon ist, dass die Web-App in allen modernen mobilen Webbrowsern funktioniert. Der Nachteil dieser Form des Anwendungsprogramms ist, dass nicht auf die Hardwarefunktionalitäten des Smartphones zugegriffen werden kann. Da aber die Schnittstelle zu Google-Chromecast nicht von einem mobilen Webbrowser unterstützt wird, ist diese Form der Entwicklung nicht möglich gewesen.

Die hybride Applikationsentwicklung erlaubt es dagegen eine programmierte Web-App in einen nativen Container zu packen und so auf Hardwarefunktionalitäten zuzugreifen. Zwar wird eine Anbindung an Chromecast als Cordova-Chromecast-Bibliothek<sup>14</sup> angeboten, doch die Schwierigkeiten bei dem Implementationsprozess haben gezeigt, dass sie ungeeignet ist. Stattdessen fiel die Wahl aufgrund der besseren Performance und Unterstützung auf einen nativen Entwicklungsprozess.

---

<sup>14</sup> Cordova Chromecast - <https://github.com/GetVideostream/cordova-chromecast>; letzter Zugriff: 29.10.2015

Die interaktive Second Screen Anwendung SecondCast-Client<sup>15</sup> konnte als Android-Applikation anhand des erarbeiteten GUI-Konzeptes umgesetzt werden. Als integrierte Entwicklungsumgebung wurde Android Studio verwendet. Um den Probanden einen Zugriff auf die Applikation zu erleichtern und Updates der Anwendung zu bewerkstelligen, wurde der SecondCast-Client in den Google Play Store geladen. Außerdem ist es damit möglich neue Versionen der Anwendung erstmal im alpha-Stadium zu Veröffentlichen, um es Testnutzern zur Verfügung zu stellen. Nachdem die Anwendung getestet wurde kann sie vom alpha-Stadium in den veröffentlicht-Status verschoben werden.

### 5.1.1. User Story „Dokumentation abspielen“

Die Abbildung 5.1 zeigt anhand von Ausschnitten aus der umgesetzten Applikation die User Story für die Anzeige von Dokumentationen auf dem First Screen und die Bereitstellung der Zusatzinformationen auf dem Second Screen. Beim Aufruf vom SecondCast-Client wird der Nutzer mit einem Auswahlmenü begrüßt. Neben einer Infoansicht und einem eingebauten Tutorial, kann der Nutzer auf die Mediathek der ausgewählten Dokumentationen zugreifen.

Der Punkt A1 verdeutlicht das Klick-Ereignis, nachdem der Nutzer zur Liste mit den von SecondCast-Multimedia-Library angebotenen Dokumentationen gelangt. Falls ein Chromecast in der Nähe ist, leuchtet im Titelbereich rechts ein Media-Streamer-Knopf auf.

Der Übergang A2 zeigt das Aufspringen eines Dialogfensters mit der Liste aller sich in der Nähe befindenden Media-Stream-Adapter an. Nachdem der Nutzer sich in diesem Fall mit dem Chromecast-ChromeDB verbunden hat, gelangt er durch die Auswahl einer Dokumentation in die Fernbedienungsansicht vom SecondCast-Client. In dieser Ansicht hat der Benutzer die Möglichkeit neben den Steuerungselementen wie der Lautstärkeregelung das ausgewählte Video auf dem verbundenen First Screen abzuspielen.

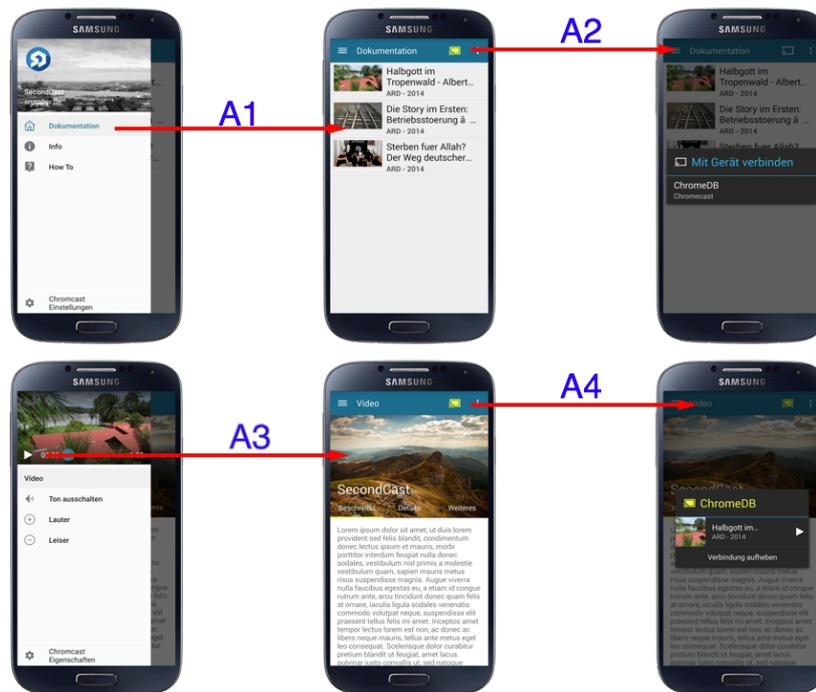
Nach der Betätigung des Abspielknopfes gelangt der Nutzer im Übergang A3 in die Zusatzinformationsansicht. Hier werden dem Benutzer die aufbereiteten Informationen von dem SecondCast-Server präsentiert. Die Fernbedienungsansicht kann dabei jederzeit mit der Durchführung einer Swipe-Right-Geste ausgeklappt und eingeklappt werden.

---

<sup>15</sup>SecondCast Android App - <https://play.google.com/store/apps/details?id=de.haw.ivandemin.cast>; letzter Zugriff: 29.10.2015

## 5. Realisierung

Nachdem der Nutzer sich die Dokumentation zu Ende angeschaut hat, kann er durch die Betätigung des Media-Stream-Adapter-Knopfes, wie in A4 dargestellt, die Verbindung mit dem Chromecast beenden. Nach der Verbindungstrennung verschwindet das Leuchten des Chromecast-Knopfes und der Benutzer bleibt auf der Zusatzinformationsansicht.



**Abbildung 5.1.:** Umgesetzte User Story für die Anzeige von Dokumentationen auf dem First Screen, sowie der Bereitstellung von Zusatzinformationen auf dem Second Screen.

### 5.1.2. Tracking von Nutzeraktionen

Eine der geforderten funktionalen Anforderungen an die Smartphone Applikation ist das Tracking von Nutzeraktionen. Hierbei muss dem Prüfer die Möglichkeit geboten werden alle Aktionen des Probanden zu verfolgen, um sie später auswerten zu können. Dies hat den Hintergrund, dass neben dem Verfolgen des Nutzungsverhaltens bei Untersuchungen auch die breite Maße an Android Geräten der potenziellen Probanden und Kunden unterstützt werden müssen. So kann die Anwendung auf einer breiten Masse von Smartphone-Modellen problemlos funktionieren, während sie aber auf einigen Gerätemodellen zum unerwünschten

## 5. Realisierung

Verhalten führt. Die Smartphones mit der jeweiligen mobilen Betriebssystemversion, die zu Problemen führen, gilt es zu finden und die Anwendung dahingehend auszubessern.

Für die Aufzeichnung der Nutzerinteraktionen und Erhebung der Datensätze über das Hardwareverhalten wird der Online-Dienst Google Analytics<sup>16</sup> verwendet. Die Abbildung 5.2 zeigt einen Ausschnitt aus der angebotenen Dashboard App-Übersicht von den erworbenen Datensätzen des SecondCast-Clients. Dieses stellt neben der durchschnittlichen Sitzungsdauer, die verwendeten Gerätemodelle, Bildschirmufrufe nach einem definierten Bildschirm, Gesamtaufrufe der App und aus welchem Land die Anfrage kommt dar. Die Anwendung ist so konstruiert, dass für die Bildschirme das Aufrufen der Dokumentations- und Zusatzinformationsansicht aufgezeichnet werden. Zudem erlauben die angebotenen Filter zum Beispiel die Angabe des mobilen Betriebssystems oder der Zeitspanne für die konkrete Betrachtung der Datensätze. Die erhaltenen Informationen lassen sich unter anderem in die Formate: CSV, Excel und PDF exportieren. So können die erworbenen Datensätze mit Tools wie Grafana<sup>17</sup> in Form von anderen grafischen Darstellung visuell aufbereitet werden.

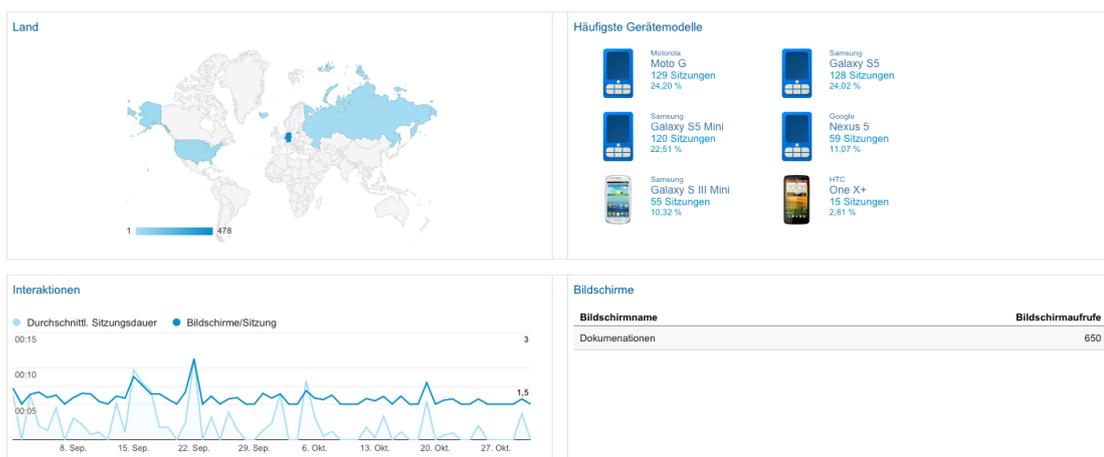


Abbildung 5.2.: Ausschnitt aus dem Google Analytics Dashboard für die App-Übersicht.

## 5.2. SecondCast-Server

Die Hauptkomponente SecondCast-Server besteht aus drei Unterkomponenten. Diese wurden in verschiedenen Teilschritten umgesetzt. Die Umsetzung der Teilkomponenten wird in diesem Abschnitt vorgestellt. Als erstes wird auf die Kernkomponente Secondcast-Core eingegangen,

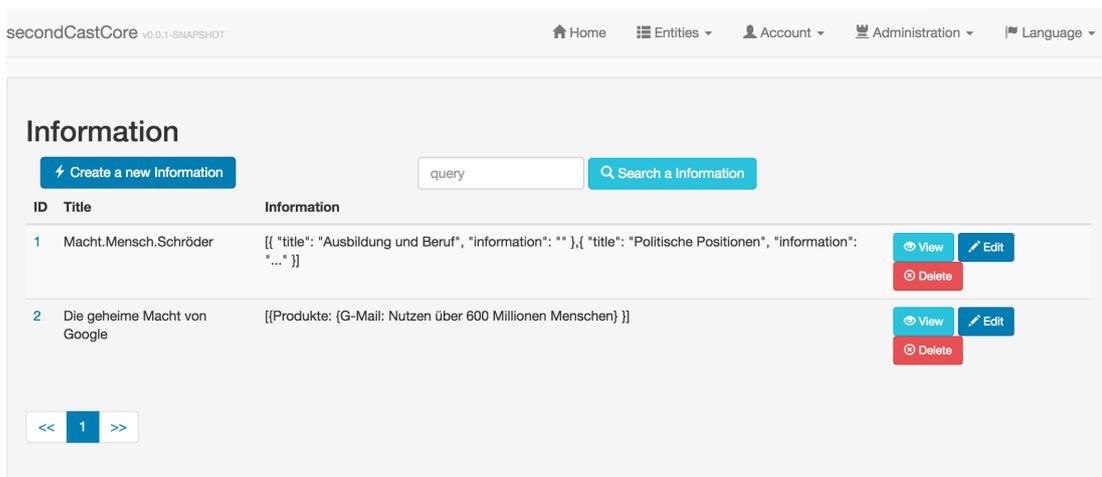
<sup>16</sup>Google Analytics - [http://google.com/intl/de\\_de/analytics](http://google.com/intl/de_de/analytics); letzter Zugriff: 29.10.2015

<sup>17</sup>Grafana - <http://grafana.org>; letzter Zugriff: 29.10.2015

darauffolgend wird die für Kontextermittlung verwendete Komponente SecondCast-Context vorgestellt. Zum Ende des Abschnitts wird die für Textzusammenfassung entwickelten Teilkomponente SecondCast-Summarizer betrachtet.

### 5.2.1. SecondCast-Core

Die Kernkomponente von SecondCast wurde als Server-Anwendung durch den Technologie-Stapel jHipster<sup>18</sup> umgesetzt. Dieses bietet eine Reihe von Frameworks wie Spring<sup>19</sup> und AngularJS<sup>20</sup> zur Umsetzung von Client-Server-Anwendungen. Unter anderem bietet die Anwendung eine Möglichkeit neue Entitäten zu erzeugen und zu verwalten. Die Abbildung 5.3 zeigt die nach dem konzipierten Komponentendiagramm 4.5 umgesetzte Ansicht von der Entität Information. Hierbei können neue Einträge mit den Feldern Titel und Information erzeugt werden. Für die Verwaltung der einzelnen Einträge stehen Funktionalitäten zur Bearbeitung, Ansicht und Entfernung von Datensätzen zur Verfügung. Des Weiteren wird die Paginierung der Einträge und Suche nach Informationen angeboten.



**Abbildung 5.3.:** Ansicht der umgesetzten Entität Information für das Anlegen und Verwalten von Datensätzen.

Um einen möglichst einfachen Einstieg für die Abfrage von Datensätzen zu erhalten wird eine visuelle Repräsentation der REST-Schnittstelle angeboten. Sie kann unter dem Unterpunkt API des Navigationspunktes „Administration“ aufgerufen werden. Die Abbildung 5.4 verdeut-

<sup>18</sup>jHipster - <https://jhipster.github.io>; letzter Zugriff: 29.10.2015

<sup>19</sup>Spring - <https://spring.io>; letzter Zugriff: 29.10.2015

<sup>20</sup>AngularJS - <https://angularjs.org>; letzter Zugriff: 29.10.2015

## 5. Realisierung

licht die möglichen Aufrufe der Informationsressourcen, dabei stehen dem Nutzer folgende Operationen zur Verfügung: DELETE, GET, POST und PUT.

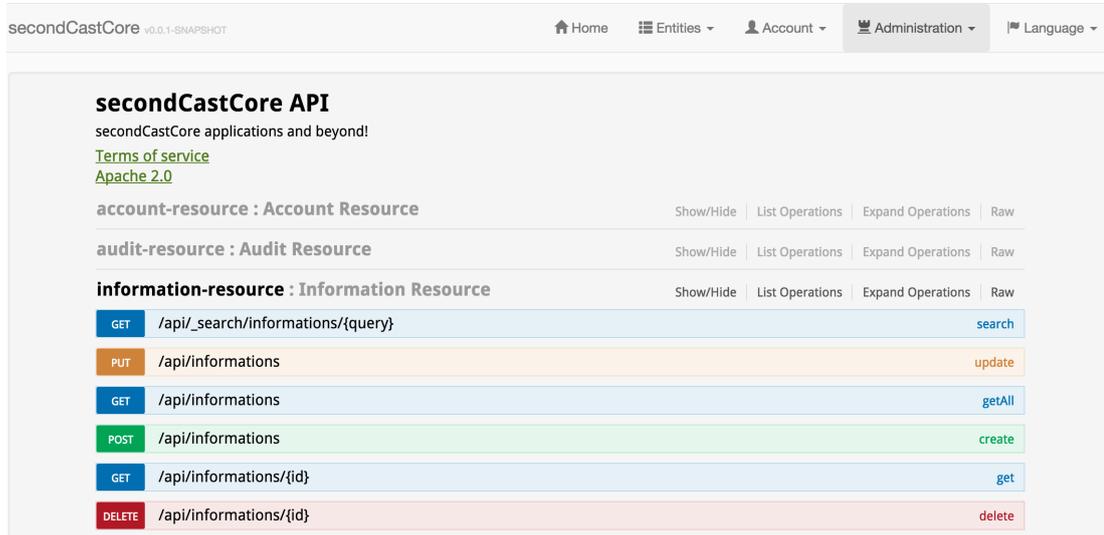


Abbildung 5.4.: Visuelle Darstellung der REST-Schnittstelle von SecondCast-Core.

Außerdem ist die Anwendung mehrsprachig aufgebaut, neben Deutsch werden Französisch und Englisch als Sprachen für die Ansichten angeboten.

### 5.2.2. SecondCast-Context

Für die Extraktion von Schlagwörtern aus Texten wurde die Kontextkomponente umgesetzt. Zur Realisierung wurde das Framework Dexter verwendet. In mehreren Teilschritten wurde aus den 1,7 Millionen Wikipedia-Artikeln des Wikipedia-Dumps vom 07.06.2015 ein Entität-Link-Graph erstellt. Dieser dient als Datenspeicher im Dexter-Core und bietet eine REST-Schnittstelle für den Zugriff auf die Datensätze an. Mit Hilfe der eingerichteten Dexter-Webapp ist es möglich Anfragen an Dexter-Core zu stellen.

### Evaluierung

Die Evaluierung der Komponente soll anhand des entwickelten Systems Dexter-Analyser erfolgen. Der Aufbau des Tools sowie die Untersuchung mit Bewertung wird in diesem Abschnitt veranschaulicht.

## Dexter-Analyser

Um die Qualität der aufgesetzten Kontextkomponente zu messen wurde das Evaluierungssystem Dexter-Analyser entwickelt. Die Abbildung 5.5 verdeutlicht den Aufbau der Evaluierungsansicht der Anwendung. Insgesamt ist die Ansicht in vier Abschnitte: F1, F2, F3 und F4 aufgeteilt.

Der obere Abschnitt F1 beinhaltet die einzelnen Regler für die Filterung der ausgegebenen Datensätze. Darunter können die minimale Linkwahrscheinlichkeit und Ergebnis angegeben werden. Im nachfolgenden Abschnitt F2 wird dem Nutzer die Möglichkeit geboten, einen Text für die Analyse einzugeben und an die REST-Schnittstelle des aufgesetzten Dexter-Systems zu versenden. Im nachfolgenden Teilbereich F3 kann der Anwender die Wörter eingeben, welche er für das Grundverständnis des Textes als bedeutsam erachtet. Anhand der eingegebenen Wörter und der Ergebnisse von Dexter werden die Werte für *Recall*, *Precision* und *F1 score* (siehe [W. Berry und Kogan, 2010](#)) gebildet. Die konkrete Ausgabe der erhobenen Datensätze von Dexter wird im Abschnitt F4 in einer Tabelle aufgegliedert. Neben der Möglichkeit die Ergebnisse im CSV-Format zu exportieren wird die Funktionalität der Sortierung nach den Ausgabeparametern angeboten.

The screenshot displays the Dexter-Analyser interface with four distinct sections:

- F1 (Filters):** Located at the top, it contains three sliders for 'Min-conf: 0.8', 'Min-linkprobability: 0.6', and 'Min-score: 0.6', along with a 'Number: 50' input field.
- F2 (Text Input):** A text area containing a paragraph about the 'Milliardendeal' between East and West Germany in 1983. It includes a 'Clear' button and a 'Send' button.
- F3 (Word Input and Metrics):** A list of words: 'Bayerischer\_Ministerpräsident', 'Staatsbankrott', and 'DDR'. Below the list, it shows calculated metrics: 'Recall: 0.750', 'Precision: 0.600', and 'F-measure: 0.667'.
- F4 (Result Table):** A table with columns 'Wikiname', 'Linkprobability', 'Commonness', and 'Score'. It lists five terms with their respective values.

Wikiname	Linkprobability	Commonness	Score
Alexander_Schalck-Golodkowski	1.000	1.000	1.000
Bayerischer_Ministerpräsident	1.000	1.000	1.000
Niederländische_Sprache	0.758	0.746	0.746
Staatsbankrott	1.000	1.000	1.000
Überschuldung	1.000	1.000	1.000

Abbildung 5.5.: Dexter-Analyser für die Evaluierung der Messergebnisse anhand der Bildung von Werten: *Recall*, *Precision* und *F1 score*.

### Untersuchung

Für die Messung der Qualität von Ausgabewerten wurde eine Untersuchung durchgeführt. Als Testdaten wurden die 40 ausgesuchten Dokumentationsbeschreibungen aus dem Projektbericht **Demin (2015)** verwendet. Alle implementierten Filter wurden auf 80 Prozent gestellt. Das Gesamtergebnis der Untersuchung kann aus der Tabelle 5.1 entnommen werden.

**Tabelle 5.1.:** Gesamtergebnis der Performance von Dexter anhand der untersuchten Dokumentationsbeschreibungen.

<i>Recall:</i>	49,37
<i>Precision:</i>	32,09
<i>F1 score:</i>	36,15

### Bewertung

Dexter hat Recall-Werte von knapp unter 50 Prozent erreicht. Dagegen liegt die Performance für Precision und F1 score unter 40 Prozent. Wenn die detaillierte Liste zur Bewertung von Teilergebnissen hinzugezogen wird, ist es auffällig, dass bei insgesamt zehn Dokumentationsbeschreibungen eine Nullrunde erreicht wird. Außerdem ist bei der Untersuchung aufgefallen, dass Dexter die Wörter nicht richtig umbricht. Damit werden oft Ortslandschaften gefunden und der ganze Text in Bezug zu diesen Entitäten genommen. Es stellte sich heraus, dass der Spotter nicht richtig arbeitet, da die Komponente Dexter-Core noch einen NER (Named Entity Recognition) für die englische Sprache verwendet. Damit werden den Entitäten falsche Klassen zugeordnet. Darüber hinaus sind Antwortzeiten bei größeren Texten von über 500 Wörtern noch zu lang.

#### 5.2.3. SecondCast-Summarizer

Längere Texte fordern mehr Aufmerksamkeit der Nutzer. Da die Menschen aber, wie im Abschnitt 3.1 beschrieben, nur bedingt multitaskingfähig sind, haben sie nur eine begrenzte Aufmerksamkeitspanne. Um die erhaltenen Informationen für die Second Screen Benutzer zu komprimieren, soll der SecondCast-Summarizer eingesetzt werden.

Der SecondCast-Summarizer konnte dabei als Webanwendung mit REST-API, bestehend aus den Python-Bibliotheken: Sumy<sup>21</sup> und Flask<sup>22</sup> umgesetzt werden. Die Bibliothek Sumy stellt

---

<sup>21</sup>Sumy - <https://github.com/miso-belica/sumy>; letzter Zugriff: 29.10.2015

<sup>22</sup>Flask - <http://flask.pocoo.org>; letzter Zugriff: 29.10.2015

die Funktionalitäten aus verschiedenen implementierten Textzusammenfassungsalgorithmen dar, mit denen Texte verkürzt werden können. Eine Vorstellung von Sumy mit den fünf zugehörigen angebotenen Algorithmen für das Zusammenfassen von Texten kann dem Abschnitt 2.3 entnommen werden. Durch das Web-Applications-Framework Flask wurde die REST-Schnittstelle entwickelt. Die Kommunikation von SecondCast-Summarizer und dem Webserver erfolgt dabei über eine WSGI-Schnittstelle (siehe Gardner, 2009). Diese stellt eine Schnittstellen-Spezifikation dar, die eine Kommunikationsform zwischen Webservern und Web-Applikations-Servern bestimmt, um die Portabilität von Webanwendungen auf unterschiedlichen Webservern zu fördern.

Dabei wurde der Webdienst so konfiguriert, dass er entweder einen übermittelten Text zusammenfasst oder einen Wortlaut aus der angegebenen Webseiten-URL extrahiert und verkürzt. Es ist für beide Fälle so eingestellt, dass die Texte in maximal zehn Sätzen zusammengefasst werden sollen. Dies entspricht einer durchschnittlichen Länge von 300 Wörtern pro zusammengefassten Text. Die Anwendung ist auf die deutsche Sprache konfiguriert, kann aber durch Erweiterung der Anwendung tschechische oder englische Sprache unterstützen.

### **Evaluierung**

Da die Bibliothek Sumy insgesamt fünf verschiedene Algorithmen zur Textzusammenfassung anbietet, muss eine Implementation gewählt werden. Dazu wurden fünf verschiedene Versionen von SecondCast-Summarizer mit jeweils einem ausgewählten Textzusammenfassungsalgorithmus implementiert. Die maximale Ausgabe von zusammengefassten Texten wurde auf zehn Sätze beschränkt. Danach wurden die selben zehn Anfragen durch ausgesuchte URLs mit Wikipedia-Artikeln, die mehr als 1000 Wörter aufweisen, für jede Implementation getätigt. Hierbei wurde die Verständlichkeit und durchschnittliche Textlänge der Ergebnisse geprüft. Das Bewertungsschema für die Einschätzung der Verständlichkeit geht von 1 für „sehr verständlich“ bis 5 für „unverständlich“. Dazu wird geschaut ob die einzelnen Sätze aus dem Ergebnis zueinander passen und in der Reihenfolge einen Sinn ergeben. Die Gesamteinschätzung auf Basis der angewendeten Verfahren aus der Sumy Bibliothek wird in der Tabelle 5.2 aufgeführt.

**Tabelle 5.2.:** Gesamteinschätzung der Verständlichkeit von den zusammengefassten Texten auf Basis der angewendeten Verfahren aus der Sumy Bibliothek.

Textzusammenfassungs- algorithmus	Verständlichkeit	Durchschnittliche Textlänge
Luhn	2	375
Edmundson	4	230
ELSA	1	278
LexRank	4	128
TextRank	4	286

### Bewertung

Für den späteren Einsatz im SecondCast System ist es bedeutsam, dass die generierten Zusatzinformationen für den Nutzer verständlich und von der Textlänge möglichst kurz sind. Da die Sumy Bibliothek insgesamt fünf Algorithmen anbietet musste auf Basis der vorgestellten Punkte ein Entschluss gefasst werden, was für ein Verfahren verwendet werden sollte. Die Evaluierung der implementierten Textzusammenfassungsverfahren haben eine Einschätzung gebracht. Demnach haben die von Sumy angebotenen Verfahren: Edmundson, LexRank und TextRank von der Verständlichkeit am schlechtesten abgeschnitten. Dies lag daran, dass keine vollständigen Sätze oder Quellenverweise in den Ergebnissen vorkamen. Auf Basis der Gesamteinschätzung wurde sich für die Verwendung von ELSA entschieden. Der Python Code von SecondCast-Summarizer kann dem Anhang [A.1](#) entnommen werden.

### 5.3. Fazit

Die Second Screen Untersuchungsplattform konnte zu weiten Teilen umgesetzt werden. Dabei wurde SecondCast-Client als Android-Applikation anhand des erarbeiteten GUI-Konzeptes umgesetzt. Außerdem konnten alle funktionalen Anforderungen bis auf „Austausch von Informationen mit Personen“ realisiert werden. Es ist möglich mit Hilfe von Chromecast Videos auf dem Fernseher abzuspielen. Die Verbindung zum Media-Stream-Adapter erlaubt auf den Abspielstatus Einfluss zu nehmen und die Lautstärke zu verändern. Außerdem konnte das Tracking von Nutzeraktionen mit Hilfe von Google Analytics ermöglicht werden. Darüber hinaus wurde die eigentliche Hauptfunktionalität die Bereitstellung von Zusatzinformationen für ein angezeigtes Video auf dem Second Screen, mit der Interaktion zwischen SecondCast-Client

und Server, realisiert.

Auf Seiten des SecondCast-Servers konnten alle funktionalen Anforderungen bis auf die Personalisierung von Informationen realisiert werden. Die Informationsverwaltung erfolgt in der nach dem GUI-Konzept entwickelten Ansicht auf dem SecondCast-Core. Mit Hilfe von SecondCast-Context können aus Beschreibungstexten relevante Schlagwörter durch das erlernte Wikifizierungs-Verfahren extrahiert werden. Dagegen werden die erhaltenen Schlagwörter als Anfragen nach Information von SecondCast-Core an Wikipedia zur Zusatzinformationsbeschaffung gesendet werden. Der SecondCast-Summarizer erlaubt es die von Wikipedia erhaltenen Informationen zusammenzufassen. Für die Zusammenfassung von Texten wird auf Basis der durchgeführten Einschätzung der von Sumy angebotene Algorithmus ELSA verwendet. Alle Umgesetzten Komponenten von SecondCast-Server sind als eigenständige Serveranwendungen mit REST-Schnittstellen umgesetzt, was zu einer losen Kopplung und zur Austauschbarkeit der Systemteile führt. Die Wartbarkeit der Projekte wird mit einer Versionsverwaltung durch GIT und der Dokumentation von REST-Schnittstellen mit Swagger ermöglicht.

Für die Umsetzung der Komponenten gibt es weitere Optimierungsmöglichkeiten. So könnte für die Erweiterung des SecondCast-Clients das Open-Source-Framework AndroidAnnotations<sup>23</sup> für die Entwicklung der Android Anwendung verwendet werden. Dieses bietet die Möglichkeit Annotationen im Java-Code zu verwenden, was die Anzahl an geschriebenen Zeilen Code reduziert. Durch die Reduktion des Codes könnte der weitere Entwicklungsprozess beschleunigt werden. Des Weiteren könnte bei der Gestaltung der GUI auf Material Design mit dem dazugehörigen Designprinzipien gesetzt werden. Da die Designsprache eine Vielzahl von Androidversionen nicht unterstützt, kann durch die Verwendung des Projekts MaterialDesign<sup>24</sup> diesem Umstand entgegengewirkt werden. Die Bibliothek ermöglicht die Unterstützung der Designelemente ab Android Version 2.2.

Trotz alledem konnten alle für die angestrebte Untersuchung nötigen funktionalen Anforderungen umgesetzt werden. Somit ist die technische Grundlage für die experimentelle Evaluation der Hypothese geschaffen worden. Im nächsten Schritt muss die Versuchsdurchführung genauer spezifiziert werden, um möglichen Risiken bei der Durchführung des Experimentes vorzubeugen.

---

<sup>23</sup> AndroidAnnotations - <http://androidannotations.org>; letzter Zugriff: 29.10.2015

<sup>24</sup> MaterialDesignLibrary - <https://github.com/navasmdc/MaterialDesignLibrary>; letzter Zugriff: 29.10.2015

## 6. Evaluierung

Wie in der Einleitung definiert, soll die entwickelte Untersuchungsplattform anhand der Überprüfung folgender Hypothese validiert werden:

*„Es ist möglich mit Hilfe von Second Screen die Informationsaufnahme mit dem First Screen zu intensivieren.“*

Diese beschäftigt sich mit der Fragestellung inwieweit Second Screen einen Einfluss auf die Informationsaufnahme des Menschen hat. Dazu soll in diesem Kapitel eine kontextsensitive Evaluierung anhand eines Experimentes mit Probanden erfolgen. Es werden die Erkenntnisse aus dem Abschnitt 2.4 angewandt. Hierzu werden zuerst die Rahmenbedingungen, Versuchsdurchführung und Versuchsaufbau im Abschnitt 6.1 beschrieben. Die Ergebnisse des durchgeführten Experimentes werden im darauf folgenden Abschnitt 6.2 präsentiert. Als nächstes erfolgt die Auswertung von den Resultaten im Abschnitt 6.3. Zum Abschluss des Kapitels wird ein Fazit im Abschnitt 6.4 gezogen.

### 6.1. Szenario

Das erstellte Szenario lehnt sich an den Hauptanwendungsfall von Second Screen mit der Bezugnahme auf die umgesetzten funktionalen Anforderungen an. Dafür sollen die Probanden eine private Nutzung im Freundeskreis oder alleine in einer Smart-Home Umgebung simulieren. Dazu werden im Abschnitt 6.1.1 die Rahmenbedingungen für das Experiment geklärt. Im nachfolgenden Abschnitt 6.1.2 wird auf die Versuchsdurchführung eingegangen. Der konkrete Versuchsaufbau mit Vorstellung der Untersuchungsumgebung wird im Abschnitt 6.1.3 vorgestellt.

#### 6.1.1. Versuchsdurchführung

Im Rahmen der Überprüfung der aufgestellten Hypothese wurde eine Untersuchung mit Probanden angestrebt. Dafür wurden insgesamt 32 technikaffine Personen akquiriert. Dies bedeutet, dass die Teilnehmer über einen sicheren Umgang mit Smartphones verfügten. Das

Geschlecht der Probanden war bei der Teilnahme an dem Experiment nicht im Vordergrund.

Für die Überprüfung der Hypothese wurde die Dokumentation „Die Story im Ersten: Die geheime Macht von Google“<sup>25</sup> des Senders ARD aus dem Jahre 2014 ausgewählt. Diese beleuchtet die Position von dem US-amerikanischen Internetunternehmen Google Inc. im Hinblick auf Europarecht, Konkurrenz und Nutzer.

Die Versuchsgruppe wurde in zwei gleich große Teilgruppen aufgegliedert. Der erste Teil stellte die Kontrollgruppe dar und schaute die Dokumentation ohne Second Screen. Die andere Hälfte der Probandengruppe schaute die Dokumentation mit Bezugnahme eines internetfähigen Gerätes als Second Screen. Beide Gruppen mussten einen Wissenstest, bestehend aus fünf Fragen (siehe A.2), zu der Dokumentation bearbeiten, wobei die Informationen zur Beantwortung der Fragen für beide Versuchsgruppen aus der Dokumentation entnommen werden konnten. Hierbei konnten die Fakten aus der Dokumentation bezogen werden. Für die Second Screen Gruppe wurden zusätzlich zu dem Mediengeschehen mit Hilfe von SecondCast-Client sieben Informationsblöcke zu der Dokumentation aus dem kuratierten Archiv Wikipedia angezeigt. Um die Aufmerksamkeitspanne der Nutzer nicht überstrapazieren, wurde mit kurzen Texten von maximal 300 Wörtern gearbeitet. In den Zusatzinformationen waren für den Wissenstest (bis auf die dritte Frage) alle erforderlichen Fakten versteckt. Neben dem angestrebten Wissenstest wurde eine Usability-Untersuchung durchgeführt, um die Bedienbarkeit und Funktionsfähigkeit des Produktes zu testen. Dazu wurde auch ein Fragebogen mit fachlichen und technischen Fragen entwickelt (siehe A.2). Ein Austausch der Probanden während des Versuchsdurchlaufs war nicht erwünscht. Die Kommunikation mit den nachfolgenden Teilnehmern wurde nicht erlaubt. Die Fragebogenentwicklung wurde von einer Soziologin begleitet.

Die Untersuchung wurde in dem Labor Living Place Hamburg (LPH) (Hamburg, 2009) aufgrund von technischen Möglichkeiten und des detailgetreuen Nachbaus eines multimedialen Wohnzimmers durchgeführt. Der Einfluss des LPH auf die Ergebnisse ist schwer einschätzbar, dennoch wurde die Laborsituation durch die Loft-Atmosphäre abgedeckt. Ein Versuchsdurchlauf hatte eine Dauer von 35 bis 40 Minuten. Dies entspricht mehr als die Hälfte der Dokumentationsdauer. Die Teilnehmerzahl wurde auf maximal vier Personen pro Versuchsdurchlauf beschränkt.

---

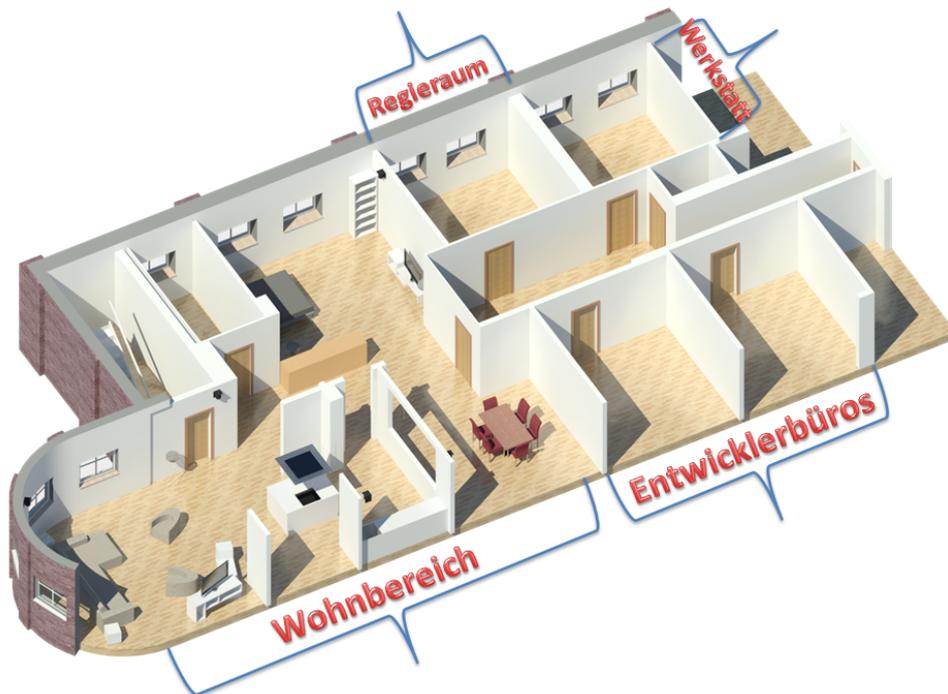
<sup>25</sup>Dokumentation - <http://www.ardmediathek.de/tv/Reportage-Dokumentation/Die-Story-im-Ersten-Die-geheime-Macht-v/Das-Erste/Video?documentId=25070276&bcastId=799280>; letzter Zugriff: 29.10.2015

### 6.1.2. Ethische Rahmenbedingungen

Bei der Versuchsplanung wurde auf die ethische Unbedenklichkeit der Untersuchung geachtet, da das Wohlergehen der Probanden im Vordergrund stand. Ethisch bedenklich sind Untersuchungen, welche die Menschenwürde verletzen oder potentielle Gefahren für das Wohlergehen darstellen (vgl. Eid u. a., 2013). Zur Einhaltung der ethischen Verantwortlichkeit wurden Standards nach der Deutschen Gesellschaft der Psychologie und dem Berufsverband Deutscher Psychologinnen und Psychologen (siehe Deutsche Gesellschaft für Psychologie e.V., 2004) eingehalten. Demnach wurde eine Einwilligungserklärung (siehe A.2) seitens der Versuchspersonen eingeholt. Die Teilnehmer wurden über potenzielle Risiken beim Konsumieren der Medieninhalte wie das Auslösen eines Epilepsie-Anfalls oder das Hervorrufen eines kindheitlichen Traumas vor Beginn des Experiments aufgeklärt. Des Weiteren war eine ausführliche und lückenlose Aufklärung der Probanden über den Zweck der Untersuchung spätestens nach dem Abschluss des Experiments Pflicht. Außerdem wurde ein Hinweis darauf gegeben, dass die Teilnahme an der Untersuchung freiwillig ist und jederzeit ohne Angabe von Gründen abgebrochen werden kann. Die Vertraulichkeit bei der Aufbereitung und Auswertung der Daten wurde zugesichert.

### 6.1.3. Technische Rahmenbedingungen

Die Versuche fanden in dem Labor Living Place Hamburg statt. Sie ist als eine Smart-Home-Forschungsumgebung konzipiert und besteht aus einem 140 m<sup>2</sup> Loft als Wohnbereich und einem separaten Arbeitsbereich. Die Abbildung 6.1 verdeutlicht den Aufbau des LPH aus der Vogelperspektive. Dieses besteht aus Schlafzimmer, Wohnzimmer, Esszimmer und Küche. Daneben bietet das Livingplace zwei Entwicklerbüros, einen Regieraum und eine kleine Werkstatt.



**Abbildung 6.1.:** Gesamtansicht des Living Place Hamburg - Smart Home Forschungs-  
umgebung.

Die Untersuchung fand im Wohnzimmer statt. Die Abbildung 6.2 zeigt den genauen Versuchsaufbau aus der Vogelperspektive. Diese teilt sich in vier Sitzgelegenheiten  $T1$  bis  $T4$  für die vier Teilnehmer, sowie einem Platz  $P$  für den Prüfer und einem feststehenden First Screen (siehe Abbildung 6.3). Aufgrund der technischen Gegebenheiten wurde die Dokumentation auf dem Fernseher über einen Medienserver abgespielt. Die Funktionalität von Chromecast als Videoübertragungsgerät wurde im Heimumfeld getestet und ist funktionsfähig. Des Weiteren wurden, um Verbindungsproblemen vorzubeugen, die angebotenen Informationen auf dem SecondCast-Client zwischengespeichert.



**Abbildung 6.2.:** Gesamtansicht des Versuchsaufbaus im Wohnbereich vom Living Place Hamburg mit vier eingezeichneten Teilnehmerplätzen, einem Platz für den Prüfer und First Screen.



**Abbildung 6.3.:** Wohnzimmer des Living Place Hamburg bestehend aus First Screen, sowie Sitzgelegenheiten wie Sofa und Sesseln.

## 6.2. Ergebnisse der Untersuchung

Die Resultate des Experimentes werden in diesem Abschnitt vorgestellt. Dazu erfolgt die Vorstellung von den Daten der Probanden, sowie dem Wissenstest und der Usability-Untersuchung. Zudem gilt, bis auf die Usability-Untersuchung, die Grundgesamtheit von  $N = 32$ , was die Anzahl der Befragten in dieser Studie repräsentiert.

### 6.2.1. Teilnehmerdaten

Durch die Erfassung der Teilnehmerdaten sollte die Altersstruktur mit Berücksichtigung des Mediennutzungsverhaltens von Internet und Fernsehen erfasst werden. Des Weiteren sollten Fragen bezüglich der Nutzung von Second Screen mit sozialem Austausch der Nutzer, sowie die Verwendung von angebotenen Second Screen-Apps, ein konkreteres Bild auf den Medienkonsum der Anwender, bieten.

Die Ergebnisse der Altersstruktur beider Versuchsgruppen können gebündelt, als deskriptive Statistik aus dem Säulendiagramm 6.4 entnommen werden. Demnach ist die demografische Struktur weitgehend homogen. Bis auf den Unterschied bei dem Geschlechtsverhältnis, dass bei Teilnehmern ohne Second Screen mit sechs eine weibliche Person mehr und ein männlicher Proband weniger teilgenommen hat als bei der anderen Versuchsgruppe, sind die es gleich.

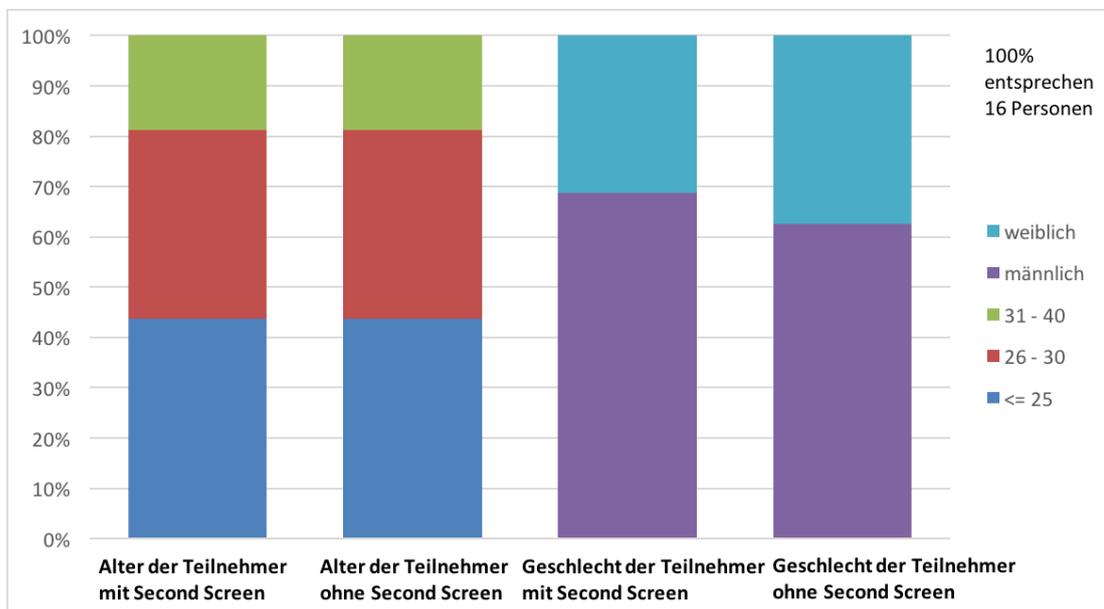


Abbildung 6.4.: Demografische Informationen der Teilnehmer aus dem Experiment.

## 6. Evaluierung

Die Teilnehmer mit Second Screen sind seit rund 14 Jahren im Internet aktiv während die zweite Gruppe seit rund 13 Jahr das Internet nutzt. Das Säulendiagramm 6.5 zeigt den konkreten Medienkonsum von Internet und Fernsehen der Teilnehmer, als deskriptive Statistik. Demnach verbringt die Mehrheit der Probanden fünf oder mehr Stunden pro Tag im Internet. Beim Fernsehkonsum der Nutzer zeichnen sich Unterschiede bei der Nutzungsdauer an. Demnach gibt es bei den Teilnehmern ohne Second Screen mit 37,5 Prozent einen erhöhten Anteil an Personen, die kein Fernsehen schauen.

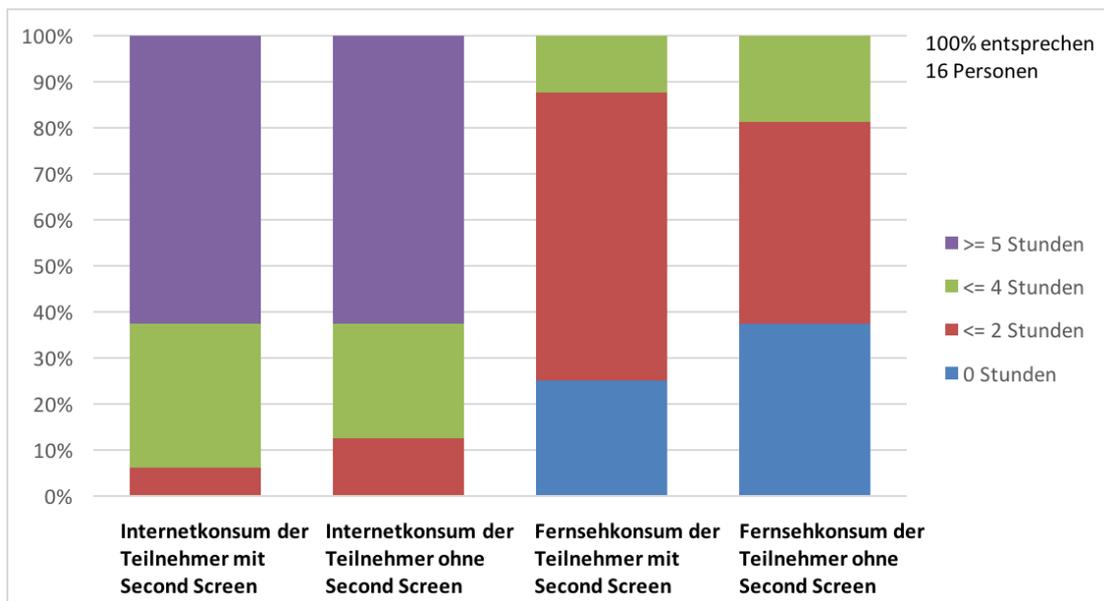


Abbildung 6.5.: Medienkonsum der Teilnehmer aus der Untersuchung.

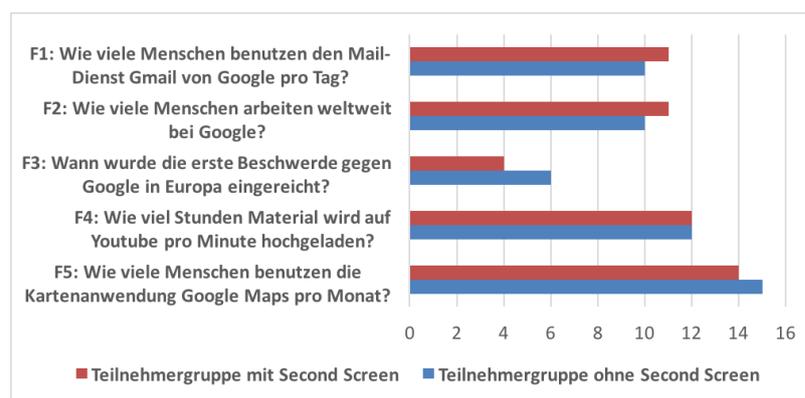
Die Mehrheit der Teilnehmer beider Versuchsgruppen gab an, sich gleichzeitig mit den gesehenen Medieninhalten auseinandergesetzt zu haben. Dies entspricht dem eigentlichen Second Screen Hauptanwendungsfall, nachdem das Informationsbedürfnis der Nutzer durch Zusatzinformationen zu dem Geschehen gestillt werden soll. Rund 40 Prozent der beiden Teilnehmergruppen, haben angegeben, sich mit einer Onlinecommunity über eine Sendung oder ein Medieninhalt ausgetauscht zu haben. Dagegen gaben rund 10 Prozent der Teilnehmer beider Nutzergruppen an, etwas über einen Medieninhalt bei Social Medien kommentiert zu haben. Des Weiteren haben nur Teilnehmern mit Second Screen angegeben zusätzliche Apps parallel zu einem Medieninhalt verwendet zu haben. Die konkreten prozentualen Ergebnisse können aus der Tabelle 6.1, die eine Datenmatrix darstellt, entnommen werden.

**Tabelle 6.1.:** Prozentuale Zustimmung der Versuchsgruppen zu den Aussagen bezüglich der Second-Screen-Nutzung.

Aussagen	Teilnehmer mit Second Screen	Teilnehmer ohne Second Screen
Ich schaue fern und informiere mich gleichzeitig im Internet über das Gesehene.	87,5 %	68,75 %
Ich sehe eine Fernsehsendung oder Video und unterhalte mich in einer Onlinecommunity über den Medieninhalt.	37,5 %	43,75 %
Ich sehe eine Fernsehsendung oder Video und kommentiere etwas zu diesem Medieninhalt in Social Medien wie Twitter oder Facebook.	12,5 %	6,25 %
Ich nutze zusätzliche Apps wie zum Beispiel ProSieben Connect oder 7TV parallel zum Mediengeschehen.	12,5 %	0,0 %

### 6.2.2. Wissenstest

Im Wissenstest wurden fünf Fragen zu dem Inhalt aus der gezeigten Dokumentation gestellt (siehe A.2). Dazu hatten die Teilnehmer die Möglichkeit eine der drei angebotenen Antworten zu der jeweiligen Fragestellung anzukreuzen. Abgefragt wurden Zahlen aus dem Google-Umfeld. Die Teilnehmergruppe mit Second Screen hatte (bis auf die dritte Fragestellung) aus dem Fragebogen alle Informationen in den sieben Informationsblöcken zur Verfügung gestellt bekommen. Die Gesamtansicht mit dem Vergleich der richtigen Antwortzahlen von den beiden Teilnehmergruppen wird im gestapelten Balkendiagramm 6.6 angezeigt.



**Abbildung 6.6.:** Resultate vom Wissenstest der beiden Versuchsgruppen, als gruppiertes Balkendiagramm mit Anzahl der richtig beantworteten Fragen.

### 6.2.3. Usability-Untersuchung

Mit dem Test soll die Usability des SecondCast-Clients, die Verfügbarkeit und Fehlertoleranz getestet werden. Nachdem die Teilnehmer der Second Screen Gruppe die Dokumentation angeschaut haben, wurde ihnen ein Fragebogen zur Bewertung der Usability präsentiert. Die Größe der Grundgesamtheit für die Erhebung der Resultate liegt bei  $N = 16$ . Das Bewertungsschema ging von 1 für „volle Zustimmung“ bzw. „sehr gut“ bis 5 für „volle Ablehnung“ bzw. „sehr schlecht“. Anhand des entwickelten Fragebogens (siehe [A.2](#)) sollen aufgelisteten Aspekte abgedeckt werden:

1. *Der fachliche Aspekt:* Die Fragen beziehen sich auf das Second-Screen-Erlebnis der Probanden während der Untersuchung. Hierbei soll festgestellt werden, inwieweit das Fernseherlebnis verbessert, gestört oder das Informationsbedürfnis zu den Gezeigten Inhalten unterstützt wurde.
2. *Der technische Aspekt:* Bezieht sich auf alle Fragen, welche sich an die Usability von SecondCast-Client richten. Damit sollen Performance und Design-Schwächen aufgedeckt werden.
3. *Fehler und Verbesserungen:* Es kann nicht garantiert werden, dass die Realisierung der Anwendung frei von Fehlern ist. Die Fragestellung, ob den Anwendern Probleme aufgefallen sind, soll Schwachstellen der Applikation aufdecken. Die möglichen Problemstellungen sollen in der nächsten Version von SecondCast-Client ausgebessert werden.

Wohingegen mögliche Verbesserungen der Anwendung an die unterschiedlichen Bedürfnisse und Auffassungen eines Second Screen-Erlebnisses der Probanden appelliert. Hierfür sollen die Anwender mögliche Verbesserungsvorschläge, wie Wünsche an die nächste Version, sei es aus der Informationssicht, Designsicht und Funktionssicht aufschreiben.

Insgesamt wurden sechs fachliche Fragen in dem Usability-Fragebogen gestellt. Die Messungen an den Versuchsobjekten wurden als Datenmatrix zusammengefasst. Sie können der [Tabelle 6.2](#) entnommen werden.

**Tabelle 6.2.:** Ergebnisse von den fachlichen Fragen aus der Usability-Untersuchung.

<b>Fachliche Fragen</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Wie ist der Gesamteindruck der App?	6,25 %	56,25 %	37,5 %	0,0 %	0,0 %
Die Nutzung der App hat das Fernseherlebnis verbessert.	0,0 %	37,5 %	43,75 %	18,75 %	0,0 %
Die Nutzung der App hat mich vom Fernsehen abgelenkt.	6,25 %	37,5 %	25,0 %	25,0 %	6,25 %
Die App hat meine Informationsbedürfnisse zum gezeigten unterstützt.	6,25 %	62,5 %	12,5 %	18,75 %	0,0 %
War es einfach Informationen aus der Anwendung zu ziehen.	56,25 %	25,0 %	6,25 %	18,75 %	12,5 %
Würden Sie die App auch privat nutzen wollen?	6,25 %	43,75 %	25,0 %	18,75 %	6,25 %

Die Ergebnisse der vier technischen Fragen der Usability-Untersuchung können in detaillierter Form aus Tabelle 6.3 entnommen werden.

**Tabelle 6.3.:** Ergebnisse von den technischen Fragen aus der Usability-Untersuchung.

<b>Technische Fragen</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
War die Bedienung der App intuitiv und einfach zu verstehen?	68,75 %	31,25 %	0,0 %	0,0 %	0,0 %
War die Menüstruktur für sie klar und verständlich?	56,25 %	43,75 %	0,0 %	0,0 %	0,0 %
Lief die App flüssig und problemlos?	62,5 %	25,0 %	18,75 %	0,0 %	0,0 %
Hat Sie das Design angesprochen?	18,75 %	31,25 %	25,0 %	25,0 %	0,0 %

Die Ergebnisse aus den Fragestellungen bezüglich aufgefallener Fehler und möglichen Verbesserungsvorschlägen werden aufbereitet dargestellt. Aussagen, die thematisch mehrfach aufgetreten sind, wurden in einem Punkt gruppiert.

#### **Welche Fehler in der Anwendung sind Ihnen aufgefallen?**

1. Beim Wechsel zwischen zwei Informationsblöcken durch das Wischen kam es zu einer Verzögerung.
2. Abstand zwischen der oberen Überschrift und Bildschirmrand zu groß.

3. Unterste Textzeile für den Informationsblock Youtube war abgehackt.

### **Welche Verbesserungsmöglichkeiten haben Sie für die App?**

1. Um sich besser konzentrieren zu können und nicht so viel Text lesen zu müssen, soll eine Option mit Stichpunktdarstellung oder Tabelle ermöglicht werden.
2. Funktionalität zur Umstellung der Schriftgröße für bessere Lesbarkeit.
3. Informationen zum Geschehen, sollen dem jeweiligen Stand der Sendung aktualisiert werden.
4. Ein Indexverzeichnis zu einem Video soll mit chronologischen Reihenfolge angeboten werden.
5. Fachbegrifflichkeiten sollen durch ein Klick im Browser suchbar sein.
6. Texte sollen von der App vorgelesen werden.
7. Personalisieren von Informationen zum Gezeigten Geschehen.
8. Anzeige von mehr Bildern in der App.
9. Die Informationen in Form einer ausklappbaren Liste mit übergeordneten Punkten darstellen.
10. Darstellung von Bildern in einem Bilderkarussell.

## **6.3. Auswertung der Untersuchungsergebnisse**

Die Untersuchungsergebnisse aus dem vorherigen Abschnitt werden in diesem Unterkapitel ausgewertet. Dazu wird zuerst auf den Wissenstest eingegangen. Danach erfolgt die Auswertung der fachlichen und technischen Fragen aus der Usability-Untersuchung unter Einbeziehung von entdeckten Fehlern und möglichen Verbesserungsvorschläge vom SecondCast-Client

### **6.3.1. Wissenstest**

Die Auswertung der Ergebnisse aus dem Wissenstest erfolgt anhand der Bildung eines Mittelwertes, als zentrale Tendenz für die Anzahl an der richtig beantworteten Fragen eines Probanden aus der jeweiligen Versuchsgruppe. Es ergab sich, dass die Teilnehmer aus der Kontrollgruppe durchschnittlich 70 Prozent richtig beantwortete Fragen hatten. Dagegen hat

die Teilnehmergruppe mit Second Screen einen Wert von 72 Prozent geschafft, was im Vergleich kein signifikanter Unterschied ist. Auch die Resultate ohne unter Einbeziehung der dritten Frage aus dem Wissenstest weisen mit einer Differenz von 3,12 Prozent keinen großen Abstand auf. Die ausgewerteten Resultate werden in der Tabelle 6.4 aufgeführt.

**Tabelle 6.4.:** Ausgewertete Ergebnisse des Wissenstests unter Einbeziehung beider Versuchsgruppen.

<b>Auswertungsart</b>	<b>Kontrollgruppe</b>	<b>Teilnehmergruppe mit Second Screen</b>
Durchschnittlicher Prozentsatz an richtig beantworteten Fragen.	70,0 %	72,5 %
Durchschnittlicher Prozentsatz an richtig beantworteten Fragen ohne Einbezugnahme der dritten Frage.	78,13 %	81,25 %

### 6.3.2. Fachliche und technische Fragen

Die Auswertung des Fragebogens erfolgt anhand der Bildung des Mittelwertes der erhobenen Resultate. Der Gesamteindruck vom SecondCast-Client ist gut. Die Nutzer finden knapp, dass durch die Anwendung das Fernseherlebnis nicht verbessert wurde. Außerdem finden die Anwender auch knapp, dass die App von dem Fernsehen nicht abgelenkt hat. Zudem tendieren die Nutzer dazu, dass das Informationsbedürfnis zu dem Gezeigten unterstützt wurde. Die Informationsaufnahme aus der Anwendung empfanden die Probanden als unkompliziert. Ferner hat die Auswertung gezeigt, dass die Probanden sich bei der privaten Nutzung nicht einig waren. Insgesamt haben sich jeweils acht Probanden für eine Nutzung oder gegen den privaten Gebrauch ausgesprochen.

Die Bedienbarkeit der Anwendung fanden die Probanden sehr gut. Die Menüstruktur war für die Nutzer klar und verständlich. Ansonsten lief die Applikation auf den meisten Smartphones problemlos. Des Weiteren kann nicht eindeutig evaluiert werden ob das Design die Anwender angesprochen hat. Es kann aber festgestellt werden, dass acht Probanden zufrieden mit der Darstellung waren. Die Ergebnisse der Untersuchung werden tabellarisch nochmals zusammengefasst:

**Tabelle 6.5.:** Gesamtergebnis von den fachlichen Fragen aus der Usability-Untersuchung.

Fachliche Frage	Gesamtergebnis
Wie ist der Gesamteindruck der App?	Gut
Die Nutzung der App hat das Fernseherlebnis verbessert.	Nein
Die Nutzung der App hat mich vom Fernsehen abgelenkt.	Nein
Die App hat meine Informationsbedürfnisse zum gezeigten unterstützt.	Ja
Es war einfach Informationen aus der Anwendung zu ziehen.	Ja
Würden Sie die App auch privat nutzen wollen?	Nicht eindeutig

**Tabelle 6.6.:** Gesamtergebnis von den technischen Fragen aus der Usability-Untersuchung.

Technische Frage	Gesamtergebnis
War die Bedienung der App intuitiv und einfach zu verstehen?	Ja
Lief die App flüssig und problemlos?	Ja
War die Menüstruktur für Sie klar und verständlich?	Ja
Hat Sie das Design angesprochen?	Nicht eindeutig

### 6.3.3. Fehler und Verbesserung

Bei der Usability-Untersuchung sind den Probanden von der Teilnehmergruppe mit Second Screen Fehler an SecondCast-Client aufgefallen. Außerdem wurden mögliche Verbesserungsvorschläge gemacht. Auf die beiden Punkte wird in diesem Abschnitt näher eingegangen.

#### Aufgefallene Fehler

Für die Definition eines Fehlers wird das Folgende Zitat verwendet:

*„Ein Fehler liegt vor, wenn ein Unterschied zwischen einem erwarteten und einem konkret vorliegenden Ergebnis oder Zustand entsteht.“* **Starke (2011)**

Es verdeutlicht die Bandbreite an Möglichkeiten was als Fehler anerkannt werden kann. Um eine Vereinfachung der Unterscheidung von Fehlern zu ermöglichen lassen sie sich in Kategorien fachliche und technische Fehler einteilen. Die von den Probanden entdeckten Probleme können in die Kategorie der technischen Fehler eingeordnet werden. Das sind die Probleme, welche die Infrastruktur betreffen oder reine Programmierfehler sind (vgl. **Starke, 2011**). In dem Fall liegt die Verzögerung beim Wechsel zwischen den Informationsblöcken an der Version von der Bibliothek `Android-ObservableScrollView`<sup>26</sup>. Diese Problem konnte nach dem Update

<sup>26</sup>Android-ObservableScrollView - <https://github.com/ksoichiro/Android-ObservableScrollView>; letzter Zugriff: 29.10.2015

gelöst werden. Die anderen beiden Fehler bezüglich der Informationsdarstellung lagen an Programmierfehlern der Designimplementierung, welche auch ausgebessert wurden.

### Mögliche Verbesserungsvorschläge

Die Verbesserungsvorschläge werden in drei Kategorien eingeteilt: Mögliche Features, Datenvisualisierung und Datenaggregation. Die möglichen Features stellen die Erweiterung von Funktionalitäten der App, um das Second Screen Erlebnis zu erweitern, dar. Während die Punkte der Datenvisualisierung das Design der Anwendungen betreffen. Dagegen stellen die Punkte aus der Datenaggregation Überlegungen dar, in welcher Form die Informationen für den Nutzer aufbereitet werden sollen.

Die Implementierung der einzelnen Punkte soll nach Prüfung auf Sinnhaftigkeit und Umsetzbarkeit in der nächsten Version erfolgen. Die Umsetzung der Verbesserungsvorschläge ist damit nicht mehr Gegenstand der Arbeit. Die in Kategorien eingeteilten Verbesserungsvorschläge werden in der Tabelle 6.7 zusammengefasst.

**Tabelle 6.7.:** Verbesserungsvorschläge der Probanden mit Einteilung in Kategorien.

Mögliche Features	Datenvisualisierung	Datenaggregation
Texte sollen von der App vorgelesen werden.	Die Informationen in Form einer ausklappbaren Liste mit übergeordneten Punkten darstellen.	Um sich besser konzentrieren zu können und nicht so viel Text lesen zu müssen, soll eine Option mit Stichpunktdarstellung oder Tabelle ermöglicht werden.
Informationen zum Geschehen, sollen dem jeweiligen Stand der Sendung aktualisiert werden.	Ein Indexverzeichnis zu einem Video soll mit chronologischen Reihenfolge angeboten werden.	Personalisieren von Informationen zum Gezeigten Geschehen.
Fachbegrifflichkeiten sollen durch ein Klick im Browser suchbar sein.	Darstellung von Bilder in einem Bilderkarussell.	Anzeige von mehr Bildern in der App.
Funktionalität zur Umstellung der Schriftgröße für bessere Lesbarkeit.		

## 6.4. Fazit

Die durchgeführte Untersuchung hat aufgezeigt, dass die Intensivierung der Informationsaufnahme durch Second Screen mit der Repräsentation von Zusatzinformationen als Textblöcke auf dem Second Screen nicht ausreichend ist. Dagegen konnte durch die Usability-Untersuchung viel nützliches Feedback in Form von möglichen Verbesserungsvorschlägen in Hinblick auf das Design und Funktionalität gegeben werden. Eine nochmalige Durchführung der Untersuchung mit verbesserter Repräsentation der Zusatzinformationen in Form von Stichpunkten und Bildern auf dem Second Screen, sowie mit einer größeren Versuchsgruppe sollte angestrebt werden.

Die geschaffene Untersuchungsplattform soll weitere Erkenntnisse im Second Screen Bereich schaffen. Die transparente Herangehensweise bei der Versuchsvorbereitung und Durchführung des Experimentes schafft eine Nachvollziehbarkeit der Untersuchung durch Dritte. Damit ist es möglich den Versuch zu kopieren, zu erweitern und auf weitere Formate zu übertragen, was zu nachfolgende Erkenntnisgewinne in anderen Bereichen wie beispielsweise UX Design und UI Entwicklung führen kann.

Außerdem soll die Untersuchungsplattform von Seiten beider Hauptkomponenten in weiteren Arbeiten erweitert werden. Auf visueller Ebene soll das Designkonzept mit Hilfe eines UX/UI-Experten auf Basis der Ergebnisse überarbeitet werden. Neue vorgeschlagene Funktionalitäten wie die Anpassung der Informationen zur Laufzeit und die Ermöglichung der Suche von Fachbegrifflichkeiten in einer Suchmaschine sollen mehr Interaktivität schaffen. Die Einbindung weiterer Informationsdienste neben Wikipedia soll die Abdeckung von Thematiken erweitern. Des Weiteren soll das Auffinden von Schlagwörtern mit Hilfe von SecondCast-Context durch das Trainieren des Klassifikators verbessert werden. All die genannten Schritte sollen zu einer Verbesserung des Second Screen-Nutzererlebnisses führen.

Darüber hinaus wirft die durchgeführte Untersuchung weitere Fragen auf. Es stellt sich die Frage, inwieweit die geforderte Funktionalität vom Design getrennt werden kann, um das Nutzererlebnis zu unterstützen. Außerdem eröffnet sich die Fragestellung, inwieweit es sinnvoll ist die Nutzer selbst entscheiden zu lassen, welche Zusatzinformationen auf dem Second Screen angezeigt werden sollten. Die letzte Fragestellung beschäftigt sich damit, was für Arten von Zusatzinformationen auf dem Second Screen eigentlich angezeigt werden sollten. Diese Fragen sollen in weiteren Arbeiten beantwortet werden.

## 7. Schlussbetrachtung

In diesem Kapitel werden die Ergebnisse der Arbeit zusammengefasst. Außerdem werden weitere interessante Fragestellungen für den Ausblick aufgeworfen.

### 7.1. Zusammenfassung

Diese Arbeit beschäftigte sich mit der Konzeption, Entwicklung und Inbetriebnahme der Experimentierumgebung SecondCast. Die kontextsensitive Evaluation des Systems erfolgte durch eine Fallstudie mit Probanden, welche in einer Smart Home Forschungsumgebung stattfand. Außerdem wurde ein Weg zur Generierung von Zusatzinformationen an dem Beispiel der Extraktion von Schlagwörtern aus Dokumentationsbeschreibung zur Abfragegenerierung an Informationsdienste aufgezeigt.

Zuerst wurde in den Grundlagen auf einzelne Tools, welche in SecondCast verwendet werden, eingegangen. Danach erfolgte die Analyse der Problemstellung Second Screen aus verschiedenen Blickwinkeln. Es wurde die bedingte Multitaskingfähigkeit der Menschen anhand von Studien veranschaulicht. Außerdem erfolgte ein Blick auf die technische Ebene, wonach das Zusammenspiel von Mediengeräten dargestellt wurde. Außerdem wurden Erwartungen der Nutzer an Second Screen Anwendungen auf Basis zweier Designstudien vorgestellt. Im Anschluss daran wurde auf die Problemstellung wie Zusatzinformationen zu einem gezeigten Inhalt generiert werden können anhand von verschiedenen Text Mining Ansätzen eingegangen. Des Weiteren wurden Ansätze zur Personalisierung von Informationen aufgezeigt. Auf Basis der gewonnenen Erkenntnisse wurden funktionale Anforderungen an das zu entwickelnde System abgeleitet. Diese stellten die Grundlage für das Design von SecondCast dar.

Auf Basis des systematischen Ansatzes der Beherrschung von Komplexität „teile und herrsche“ wurde das angestrebte System in kleinere Teile zerteilt. Zudem wurde eine Kontextabgrenzung bezüglich des Systemumfeldes durchgeführt. Aus der Kontextsicht wurde die Bausteinsicht mit den einzelnen Systemteilen abgeleitet. Die Experimentierumgebung besteht dabei aus zwei Hauptkomponenten SecondCast-Client und SecondCast-Server. Informationen zum streamen

von Videos werden aus SecondCast-Multimedia-Library bezogen. Um die Medieninhalte auf dem Fernseher abzuspielen wurde der Streaming-Media-Adapter Chromecast verwendet. Die Verwaltung von Zusatzinformationen erfolgte in SecondCast-Core. Falls keine Informationen vorhanden sind wurde die Beschreibung eines Videos zur Textanalyse an SecondCast-Context versendet. Die erworbenen Informationen aus dem kuratierten Archiv Wikipedia wurden für den vereinfachten Konsum des Nutzers durch SecondCast-Summarizer zusammengefasst. Darüber hinaus wurden grafische Konzepte für die mobile Anwendung und die Informationsverwaltungs-komponente erstellt.

Die Systemkomponenten konnten umgesetzt werden. Dazu wurde SecondCast-Client auf Basis des konzipierten grafischen Konzeptes als Android Anwendung umgesetzt. Die Kernkomponente SecondCast-Core von dem SecondCast-Server wurde als Server-Anwendung anhand des grafischen Konzeptes realisiert. Für die Schlagworterkennung aus Texten für die Generierung von Informationsabfragen an gepflegte Archive wurde SecondCast-Summarizer entwickelt. Hierfür wurde der Wikifizierungs-Algorithmus von Wikipedia mit Hilfe des Frameworks Dexter erlernt. Außerdem wurde die Textzusammenfassungskomponente SecondCast-Summarizer als Server-Anwendung durch Verwendung von den der Python-Bibliothek Sumy und Flusk realisiert. Alle geforderten REST-Schnittstellen wurden umgesetzt.

Die Evaluierung der Untersuchungsumgebung erfolgte durch die Validierung der aufgestellten Hypothese mit Hilfe einer Fallstudie. Diese fand in der Smart-Home-Umgebung Living Place Hamburg mit insgesamt 32 Probanden statt. Die Teilnehmer wurden in zwei Versuchsgruppen eingeteilt und schauten eine Dokumentation mit anschließendem Wissenstest. Die Auswertung ergab, dass die Intensivierung der Informationsaufnahme von den Second Screen Benutzern nicht erreicht wurde. Dagegen konnte aus der Usability-Untersuchung für die Benutzer von SecondCast weitere Erkenntnisse und mögliche Verbesserungsvorschläge gewonnen werden. Zudem lieferten die Resultate weitere Fragen für den Second Screen Bereich auf. Diese sollen in weiteren Arbeiten beantwortet werden.

### 7.2. Ausblick

Die Ergebnisse aus der Untersuchung wirft die Frage auf, inwieweit die Konstellation zwischen First Screen und Second Screen in der heutigen Zeit noch gegeben ist. Es wird vermutet, dass der First Screen das Gerät ist, wo die meisten Interaktionen stattfinden. In diesem Fall wäre das internetfähige Gerät wie Smartphone, Tablet oder Laptop der First Screen und der Fernseher

## 7. Schlussbetrachtung

---

damit ein Second Screen. Doch diese Aussage lässt sich in dieser Arbeit nicht mehr belegen. Weitere Untersuchungen in diesem Umfeld sind erwünscht.

Die zweite Überlegung, ist wie mehr Zusatzinformationen zu Videos generiert werden könnten. In wie weit gelieferte Metainformationen und Audio Mining Verfahren bei der Generierung von Informationen helfen könnten lässt sich noch nicht sagen.

Außerdem stellt sich die Frage, wie Individualisierung von Informationen durch die Verwendung von Benutzerprofilen einen Einfluss auf das Second Screen Erlebnis hat. Dazu müssten Wege gefunden werden wie Informationen an die Bedürfnisse der Benutzer und den derzeitigen Kontext in dem sie sich befinden angepasst werden könnten. All diesen Überlegungen müsste in weiteren Arbeiten nachgegangen werden. Hinzu kommt die Fragestellung, inwieweit der Fernseher als First Screen fungiert und nicht nur ein Hintergrundrauschen darstellt. Daraus ergibt sich die Überlegung, wie die Rolle des First Screens mit Hilfe eines Second Screen gestärkt werden könnte. All dies sind interessante Aspekte die Gegenstand weiterer Nachforschungen sein sollten.

## Literaturverzeichnis

- [BACK u. a. 2009] BACK, Andrea ; GRONAU, Norbert ; TOCHTERMANN, Klaus: *Web 2.0 in der Unternehmenspraxis: Grundlagen, Fallstudien und Trends zum Einsatz von Social Software*. Muenchen : Oldenbourg Verlag, 2009. – URL <https://books.google.com/books?hl=de&lr=&id=0FEY9aPRmpEC&pgis=1>. – ISBN 3486716859
- [Basapur u. a. 2011] BASAPUR, Santosh ; HARBOE, Gunnar ; MANDALIA, Hiren ; NOVAK, Ashley ; VUONG, Van ; METCALF, Crysta: Field trial of a dual device user experience for iTV. In: *Proceedings 9th Int. Interact. Conf. Interact. Telev. - EuroITV '11*. New York, New York, USA : ACM Press, jun 2011, S. 127. – URL <http://dl.acm.org/citation.cfm?id=2000119.2000145>. – ISBN 9781450306027
- [Benbunan-Fich u. a. 2011] BENBUNAN-FICH, Raquel ; ADLER, Rachel F. ; MAVLANOVA, Tamilla: Measuring multitasking behavior with activity-based metrics. In: *ACM Trans. Comput. Interact.* 18 (2011), jun, Nr. 2, S. 1–22. – URL <http://dl.acm.org/citation.cfm?id=1970378.1970381>. – ISSN 10730516
- [Bernhaupt u. a. 2012] BERNHAUPT, Regina ; BOUTONNET, Mael ; GATELLIER, Bastien ; GIMENEZ, Yannik ; POUCHEPANADIN, Christian ; SOUIBA, Latifa: A set of recommendations for the control of IPTV-systems via smart phones based on the understanding of users practices and needs. In: *Proc. 10th Eur. Conf. Interact. tv video - EuroITV '12*. New York, New York, USA : ACM Press, jul 2012, S. 143. – URL <http://dl.acm.org/citation.cfm?id=2325616.2325645>. – ISBN 9781450311076
- [Bernhaupt u. a. 2007] BERNHAUPT, Regina ; WEISS, Astrid ; OBRIST, Marianna ; TSCHELIGI, Manfred: Playful probing: making probing more fun. (2007). – URL [http://link.springer.com/chapter/10.1007/978-3-540-74796-3\\_60](http://link.springer.com/chapter/10.1007/978-3-540-74796-3_60)
- [Bostandjiev u. a. 2012] BOSTANDJIEV, Svetlin ; O'DONOVAN, John ; HÖLLERER, Tobias: TasteWeights : A Visual Interactive Hybrid Recommender System. In: *Proc. sixth ACM Conf. Recomm. Syst. - RecSys '12* (2012), S. 35. – URL <http://dl.acm.org/citation.cfm?doid=2365952.2365964>. ISBN 9781450312707

- [Breiman 2001] BREIMAN, Leo: Random Forests. In: *Mach. Learn.* (2001), S. 5–32
- [Busemann und Tippelt 2014] BUSEMANN, Katrin ; TIPPELT, Florian: Second Screen : Parallelnutzung von Fernsehen und Internet. In: *Media Perspekt.* (2014), S. 408–416
- [Ceccarelli u. a. 2013] CECCARELLI, Diego ; GORDEA, Sergiu ; LUCCHESI, Claudio ; NARDINI, Franco M. ; PEREGO, Raffale: When entities meet query recommender systems. In: *Proc. 28th Annu. ACM Symp. Appl. Comput. - SAC '13*. New York, New York, USA : ACM Press, mar 2013, S. 933. – URL <http://dl.acm.org/citation.cfm?id=2480362.2480540>. – ISBN 9781450316569
- [Ceccarelli und Lucchese 2013] CECCARELLI, Diego ; LUCCHESI, Claudio: Dexter: an open source framework for entity linking. In: *Proc. sixth ...* (2013), S. 2–4. – URL <http://dl.acm.org/citation.cfm?id=2513212>. ISBN 9781450324137
- [Ceccarelli und Lucchese 2015] CECCARELLI, Diego ; LUCCHESI, Claudio: Dexter, an Open Source Framework for Entity Linking, URL <http://dexter.isti.cnr.it/>, 2015
- [Cesar u. a. 2008] CESAR, Pablo ; BULTERMAN, DCA ; JANSEN, AJ: Usages of the secondary screen in an interactive television environment: Control, enrich, share, and transfer television content. In: *Chang. Telev. Environ.* (2008). – URL [http://link.springer.com/chapter/10.1007/978-3-540-69478-6\\_22](http://link.springer.com/chapter/10.1007/978-3-540-69478-6_22)
- [Chen u. a. 2010] CHEN, KC ; CHANG, JT ; PAN, JS ; HUANG, HC: The open source IPTV service development environment: IPTV service execution environment. In: *Proc. 4th ...* (2010). – URL <http://dl.acm.org/citation.cfm?id=2108677>. ISBN 9781605588933
- [Clarke u. a. 2013] CLARKE, Stevens ; LINDQUIST, Jan ; DUFOURD, Jean-Claude ; HAMMOND, Matt ; BERKOFF, Russell ; IGARASHI, Tatsuya: *Requirements for Home Networking Scenarios*. 2013. – URL <http://www.w3.org/TR/2011/NOTE-hnreq-20111201/>. – Zugriffsdatum: 2013-09-27
- [COHN 2010] COHN, Mike: *User Stories: für die agile Software-Entwicklung mit Scrum, XP u.a. mitp Professional: Amazon.de: Mike Cohn: Bücher*. Muenchen, 2010. – ISBN 978-3-826-65898-3
- [Cortez u. a. 2012] CORTEZ, James ; SHAMMA, David A. ; CAI, Linghan: Device communication. In: *Proc. 10th Eur. Conf. Interact. tv video - EuroiTV '12*. New York, New York, USA : ACM Press, jul 2012, S. 19. – URL <http://dl.acm.org/citation.cfm?id=2325616.2325622>. – ISBN 9781450311076

- [Cudré-Mauroux u. a. 2009] CUDRÉ-MAUROUX, Philippe ; HAGHANI, Parisa ; JOST, Michael ; ABERER, Karl ; DE MEER, Hermann: idMesh. In: *Proc. 18th Int. Conf. World wide web - WWW '09*. New York, New York, USA : ACM Press, apr 2009, S. 591. – URL <http://dl.acm.org/citation.cfm?id=1526709.1526789>. – ISBN 9781605584874
- [David, C., L. Giroux, S. Bertrand-Gastaldy 1995] DAVID, C., L. GIROUX, S. BERTRAND-GASTALDY, D. L.: INDEXING AS PROBLEM SOLVING: A COGNITIVE APPROACH TO CONSISTENCY. In: *Proc. ASIS Annu. Meet. Medford* (1995), S. 49–55. – URL <http://www.ualberta.ca/dept/slis/cais/david.htm>
- [Deerwester u. a. 1990] DEERWESTER, Scott ; DUMAIS, Susan T. ; FURNAS, George W. ; LANDAUER, Thomas K. ; HARSHMAN, Richard: Indexing by latent semantic analysis. In: *J. Am. Soc. Inf. Sci.* 41 (1990), S. 391–407. – URL [http://doi.wiley.com/10.1002/\(SICI\)1097-4571\(199009\)41:6<391::AID-ASI1>3.0.CO;2-9](http://doi.wiley.com/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9). – ISBN 9781450300322
- [Demin 2015] DEMIN, Ivan: Projektbericht / HAW-Hamburg. Hamburg, 2015. – Forschungsbericht. – 1–11 S. – URL [http://users.informatik.haw-hamburg.de/\\$\sim\\$subicomp/projekte/master2015-proj/demin.pdf](http://users.informatik.haw-hamburg.de/$\sim$subicomp/projekte/master2015-proj/demin.pdf)
- [Deterding u. a. 2011] DETERDING, Sebastian ; SICART, Miguel ; NACKE, Lennart ; O'HARA, Kenton ; DIXON, Dan: Gamification. using game-design elements in non-gaming contexts. In: *Proc. 2011 Annu. Conf. Ext. Abstr. Hum. factors Comput. Syst. - CHI EA '11*. New York, New York, USA : ACM Press, may 2011, S. 2425. – URL <http://dl.acm.org/citation.cfm?id=1979742.1979575>. – ISBN 9781450302685
- [Deutsche Gesellschaft für Psychologie e.V. 2004] DEUTSCHE GESELLSCHAFT FÜR PSYCHOLOGIE e.V.: Revision der auf die Forschung bezogenen ethischen Richtlinien. (2004), Nr. September, S. 1–5
- [Ebner 2014] EBNER, Ulrike: *Second-Screen-Film App auf ZDF: Das TV-Experiment bleibt ein bemuehter Versuch - Kultur - Stuttgarter Zeitung*. 2014. – URL <http://goo.gl/fj3NIY>. – Zugriffsdatum: 2015-09-29
- [Edmundson 1969] EDMUNDSON, H. P.: New Methods in Automatic Extracting. In: *J. ACM* 16 (1969), apr, Nr. 2, S. 264–285. – URL <http://dl.acm.org/citation.cfm?id=321510.321519>. – ISSN 00045411

- [Eid u. a. 2013] EID, Michael ; ; GOLLWITZER, Mario ; ; SCHMITT, Manfred: *Statistik und Forschungsmethoden - Lehrbuch. Mit Online-Materialien*. 1. Aufl. Beltz, 2013. – 7–18 S. – ISBN 978-3-621-27524-8
- [Erkan und Radev 2004] ERKAN, Günes ; RADEV, Dragomir R.: LexRank: Graph-based lexical centrality as salience in text summarization. In: *J. Artif. Intell. Res.* 22 (2004), S. 457–479. – ISBN 1076-9757
- [Felfernig und Ninaus 2012] FELFERNIG, A. ; NINAUS, G.: Group recommendation algorithms for requirements prioritization. (2012), jun, S. 59–62. – URL <http://dl.acm.org/citation.cfm?id=2666719.2666733>. ISBN 978-1-4673-1759-7
- [Ferré und Rudolph 2009] FERRÉ, Sébastien (Hrsg.) ; RUDOLPH, Sebastian (Hrsg.): *Lecture Notes in Computer Science*. Bd. 5548: *Formal Concept Analysis*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2009. – URL <http://www.springerlink.com/index/10.1007/978-3-642-01815-2>. – ISBN 978-3-642-01814-5
- [Fessler 2014] FESSLER, Simon: Twitter als interaktive Erweiterung des Fernsehens / Uni Mannheim. Mannheim, 2014. – Forschungsbericht. – 2–57 S
- [Franke und Ippen 2012] FRANKE, Florian ; IPPEN, Johannes: *Apps mit HTML5 und CSS3: für iPad, iPhone und Android Galileo Computing: Amazon.de: Florian Franke, Johannes Ippen: Bücher*. Bonn : Galileo Press GmbH, 2012. – 139 S. – ISBN 978-3-836-21848-1
- [Gantner u. a. 2010] GANTNER, Zeno ; RENDLE, Steffen ; SCHMIDT-THIEME, Lars: Factorization models for context-/time-aware movie recommendations. In: *Proc. Work. Context. Movie Recomm. - CAMRa '10*. New York, New York, USA : ACM Press, sep 2010, S. 14–19. – URL <http://dl.acm.org/citation.cfm?id=1869652.1869654>. – ISBN 9781450302586
- [Gardner 2009] GARDNER, James: *The Definitive Guide to Pylons*. Berkeley, CA : Apress, 2009. – 1–568 S. – URL <http://www.springerlink.com/index/10.1007/978-1-4302-0534-0>. – ISBN 978-1-59059-934-1
- [González und Mark 2004] GONZÁLEZ, Victor M. ; MARK, Gloria: Constant , Constant , Multi-tasking Crazy: Managing Multiple Working Spheres. 6 (2004), Nr. 1, S. 113–120. ISBN 1581137028

- [Hamburg 2009] HAMBURG, Living P.: A place for concepts of IT based modern living. (2009). – URL [http://livingplace.informatik.haw-hamburg.de/content/LivingPlaceHamburg\\_en.pdf](http://livingplace.informatik.haw-hamburg.de/content/LivingPlaceHamburg_en.pdf)
- [Hess u. a. 2011] HESS, Jan ; LEY, Benedikt ; OGOROWSKI, Corinna ; WAN, Lin ; WULF, Volker: Jumping between devices and services. In: *Proceedings 9th Int. Interact. Conf. Interact. Telev. - EuroITV '11*. New York, New York, USA : ACM Press, jun 2011, S. 11. – URL <http://dl.acm.org/citation.cfm?id=2000119.2000122>. – ISBN 9781450306027
- [IDC Corporate USA 2015] IDC CORPORATE USA: *IDC: Smartphone OS Market Share 2015, 2014, 2013, and 2012*. 2015. – URL <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>. – Zugriffsdatum: 2015-06-23
- [Jadidinejad und Mahmoudi 2009] JADIDINEJAD, AH ; MAHMOUDI, Fariborz: Query wikification: Mining structured queries from unstructured information needs using wikipedia-based semantic analysis. In: *Work. Notes CLEF 2009 ...* (2009). – URL [http://clef.isti.cnr.it/2009/working\\_notes/Jadidinejad-paperCLEF2009.pdf](http://clef.isti.cnr.it/2009/working_notes/Jadidinejad-paperCLEF2009.pdf)
- [Lehmann u. a. 2013] LEHMANN, Janette ; LALMAS, Mounia ; DUPRET, Georges ; BAEZA-YATES, Ricardo: Online multitasking and user engagement. In: *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manag. - CIKM '13*. New York, New York, USA : ACM Press, oct 2013, S. 519–528. – URL <http://dl.acm.org/citation.cfm?id=2505515.2505543>. – ISBN 9781450322638
- [Lochrie und Coulton 2012] LOCHRIE, Mark ; COULTON, Paul: Sharing the viewing experience through second screens. In: *Proc. 10th Eur. Conf. ...* (2012), S. 199–202. – URL <http://dl.acm.org/citation.cfm?id=2325655>. ISBN 9781450311076
- [Lottridge u. a. 2015] LOTTRIDGE, Danielle M. ; ROSAKRANSE, Christine ; OH, Catherine S. ; WESTWOOD, Sean J. ; BALDONI, Katherine A. ; MANN, Abrey S. ; NASS, Clifford I.: The Effects of Chronic Multitasking on Analytical Writing. In: *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. - CHI '15*. New York, New York, USA : ACM Press, apr 2015, S. 2967–2970. – URL <http://dl.acm.org/citation.cfm?id=2702123.2702367>. – ISBN 9781450331456
- [Luhn 1958] LUHN, H. P.: The Automatic Creation of Literature Abstracts. In: *IBM J. Res. Dev.* 2 (1958), Nr. 2, S. 159–165. – ISBN 0018-8646

- [Meij u. a. 2012] MEIJ, Edgar ; WEERKAMP, Wouter ; RIJKE, Maarten de: Adding semantics to microblog posts. In: *Proc. fifth ACM Int. Conf. Web search data Min. - WSDM '12*. New York, New York, USA : ACM Press, feb 2012, S. 563. – URL <http://dl.acm.org/citation.cfm?id=2124295.2124364>. – ISBN 9781450307475
- [Mihalcea und Csomai 2007] MIHALCEA, Rada ; CSOMAI, Andras: Wikify! In: *Proc. Sixt. ACM Conf. Conf. Inf. Knowl. Manag. - CIKM '07*. New York, New York, USA : ACM Press, nov 2007, S. 233. – URL <http://dl.acm.org/citation.cfm?id=1321440.1321475>. – ISBN 9781595938039
- [Mihalcea und Tarau 2004] MIHALCEA, Rada ; TARAU, Paul: TextRank: Bringing order into texts. In: *Proc. EMNLP 4 (2004)*, Nr. 4, S. 404–411. – URL <http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf>. – ISBN 9781605581934
- [Milne und Witten 2008] MILNE, David ; WITTEN, Ian H.: Learning to link with wikipedia. In: *Proceeding 17th ACM Conf. Inf. Knowl. Min. - CIKM '08*. New York, New York, USA : ACM Press, oct 2008, S. 509. – URL <http://dl.acm.org/citation.cfm?id=1458082.1458150>. – ISBN 9781595939913
- [Mummendey und Grau 2008] MUMMENDEY, Hans D. ; GRAU, Ina: *Die Fragebogen-Methode*. 5. überarb. Göttingen : Hogrefe Verlag, 2008. – 1–224 S. – ISBN 978-3-840-91948-0
- [Murray und Goldenberg 2012] MURRAY, J ; GOLDENBERG, Sergio: Story-map: iPad companion for long form TV narratives. In: *... Interact. tv ... (2012)*, S. 223–226. – URL <http://dl.acm.org/citation.cfm?id=2325659>. ISBN 9781450311076
- [Nielsen 2014] NIELSEN: THE DIGITAL CONSUMER MEET TODAY'S DIGITAL. (2014), Nr. February, S. 14
- [Odiijk u. a. 2013] ODIJK, Daan ; MEIJ, Edgar ; RIJKE, Maarten de: Feeding the second screen: semantic linking based on subtitles. (2013), may, S. 9–16. – URL <http://dl.acm.org/citation.cfm?id=2491748.2491751>
- [Page u. a. 1998] PAGE, Larry ; BRIN, Sergey ; MOTWANI, Rajeev ; WINOGRAD, Terry: 1 Introduction and Motivation 2 A Ranking for Every Page on the Web. (1998), S. 1–17
- [Pirker u. a. 2010] PIRKER, Michael ; BERNHAUPT, Regina ; MIRLACHER, Thomas: Investigating usability and user experience as possible entry barriers for touch interaction in the living room. In: *Proc. 8th Int. Interact. Conf. Interact. TVVideo - EuroITV'10*. New York, New York,

- USA : ACM Press, jun 2010, S. 145. – URL <http://dl.acm.org/citation.cfm?id=1809777.1809808>. – ISBN 9781605588315
- [Ratinov u. a. 2011] RATINOV, Lev ; ROTH, Dan ; DOWNEY, Doug ; ANDERSON, Mike: Local and global algorithms for disambiguation to Wikipedia. (2011), jun, S. 1375–1384. – URL <http://dl.acm.org/citation.cfm?id=2002472.2002642>. ISBN 978-1-932432-87-9
- [Rechtin und Maier 2000] RECHTIN, Eberhardt ; MAIER, Mark W.: *The Art of Systems Architecting, Second Edition*. CRC Press, 2000. – 344 S. – ISBN 1420058525
- [Renken u. a. 2014] RENKEN, Alica ; SAWO, Chris ; STALPH, Florian: Twitter als Ausprägung der Second Screen Nutzung Inhaltsverzeichnis / Uni Passau. Passau, 2014. – Forschungsbericht
- [Robertson u. a. 1996] ROBERTSON, Scott ; WHARTON, Cathleen ; ASHWORTH, Catherine ; FRANZKE, Marita: Dual device user interface design. In: *Proc. SIGCHI Conf. Hum. factors Comput. Syst. common Gr. - CHI '96*. New York, New York, USA : ACM Press, apr 1996, S. 79–86. – URL [http://dl.acm.org/ft\\_gateway.cfm?id=238408&type=html](http://dl.acm.org/ft_gateway.cfm?id=238408&type=html). – ISBN 0897917774
- [Shi u. a. 2010] SHI, Yue ; LARSON, Martha ; HANJALIC, Alan: Mining mood-specific movie similarity with matrix factorization for context-aware recommendation. In: *Proc. Work. Context. Movie Recomm. - CAMRa '10*. New York, New York, USA : ACM Press, sep 2010, S. 34–40. – URL <http://dl.acm.org/citation.cfm?id=1869652.1869658>. – ISBN 9781450302586
- [Spark Jones 1972] SPARK JONES, Karen: A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. In: *J. Doc.* 28 (1972), jan, Nr. 1, S. 11–21. – URL <http://www.emeraldinsight.com/doi/abs/10.1108/eb026526>. – ISSN 0022-0418
- [Starke 2011] STARKE, Gernot: *Effektive Software-Architekturen: Ein praktischer Leitfaden*. 2011. Carl Hanser Verlag GmbH Co KG (M), 2011. – ISBN 978-3-446-42728-0
- [Steinberger und Ježek 2004] STEINBERGER, J ; JEŽEK, Karel: Using Latent Semantic Analysis in Text Summarization. In: *Proc. ISIM 2004* (2004), S. 93–100
- [Trafton u. a. 2003] TRAFTON, J.Gregory ; ALTMANN, Erik M. ; BROCK, Derek P. ; MINTZ, Farilee E.: Preparing to resume an interrupted task: effects of prospective goal encoding and retrospective rehearsal. In: *Int. J. Hum. Comput. Stud.* 58 (2003), may, Nr. 5,

- S. 583–603. – URL <http://linkinghub.elsevier.com/retrieve/pii/S1071581903000235>. – ISSN 10715819
- [W. Berry und Kogan 2010] W. BERRY, Michael ; KOGAN, Jacob: *Text Mining: Applications and Theory*. New York, New York, USA : John Wiley Sons, 2010. – 136 S. – ISBN 047068965X
- [Wang u. a. 2012] WANG, Zhichun ; LI, Juanzi ; WANG, Zhigang ; TANG, Jie: Cross-lingual knowledge linking across wiki knowledge bases. In: *Proc. 21st Int. Conf. World Wide Web - WWW '12*. New York, New York, USA : ACM Press, apr 2012, S. 459. – URL <http://dl.acm.org/citation.cfm?id=2187836.2187899>. – ISBN 9781450312295
- [Wikimedia 2015] WIKIMEDIA: *Wikipedia*. 2015. – URL <https://de.wikipedia.org/wiki/Wikipedia:Hauptseite>. – Zugriffsdatum: 2015-06-22
- [Woletz 2008] WOLETZ, JD: *Digital Storytelling from Artificial Intelligence to YouTube*. 2008. – 587–601 S. – ISBN ISBN 9781599048635

# A. Anhang

## A.1. SecondCast-Summarizer

```
1 from __future__ import absolute_import
2 from __future__ import division, print_function, unicode_literals
3
4 from sumy.parsers.html import HtmlParser
5 from sumy.parsers.plaintext import PlaintextParser
6 from sumy.nlp.tokenizers import Tokenizer
7 from sumy.summarizers.lsa import LsaSummarizer as Summarizer
8 from sumy.nlp.stemmers import Stemmer
9 from sumy.utils import get_stop_words
10
11 from flask import Flask, request
12 app = Flask(__name__)
13
14 import sys
15
16 reload(sys)
17 sys.setdefaultencoding('utf8')
18
19 LANGUAGE = "german"
20 SENTENCES_COUNT = 10
21
22 @app.route("/url")
23 def summerizeFromURL():
24     url = request.args.get('url')
25     parser = HtmlParser.from_url(url, Tokenizer(LANGUAGE))
26
```

```
27 stemmer = Stemmer(LANGUAGE)
28 summarizer = Summarizer(stemmer)
29 summarizer.stop_words = get_stop_words(LANGUAGE)
30
31 result = ""
32 for sentence in summarizer(parser.document, SENTENCES_COUNT):
33 result+=str(sentence)
34
35 return result
36
37 @app.route("/text")
38 def summerizeFromPlaintext():
39 text = request.args.get('text')
40 parser = PlaintextParser.from_string(text, Tokenizer(LANGUAGE))
41 summarizer = LexRankSummarizer()
42
43 summary = summarizer(parser.document, SENTENCES_COUNT)
44
45 result = ""
46 for sentence in summary:
47 result+=str(sentence)
48
49 return result
50
51 if __name__ == "__main__":
52 app.run()
```

**Listing A.1:** Python Code vom SecondCast-Summarizer.

## A.2. Untersuchungsbögen

## Einverständniserklärung

# Einverständniserklärung

---

**Thema der Untersuchung:**

Untersuchung der Informationsaufnahme mit der Bezugnahme eines Second Screens.

**Durchführender Prüfer:**

Ivan Demin

**Einverständniserklärung:**

Mit Ihrer Unterschrift bestätigen Sie die o. g. Informationen zur Teilnahme an der Untersuchung verstanden zu haben, mit diesem Vorgehen einverstanden zu sein und als Proband an dieser Untersuchung teilzunehmen.

	JA	NEIN
Ich stimme zu, an o.g. Aufgaben teilzunehmen.		
Ich stimme Video, Audio- und Fotoaufnahmen zu.		
Ich stimme zu, dass meine anonymisierten Aussagen direkt zitiert werden dürfen.		
Ich stimme zu, dass die Videostandbilder und Fotos (Gesicht unkenntlich) für die Präsentation und/oder andere Veröffentlichungen verwendet werden dürfen.		

\_\_\_\_\_  
Name Teilnehmer/in

\_\_\_\_\_  
Unterschrift Teilnehmer/in

\_\_\_\_\_  
Datum

\_\_\_\_\_  
Mailadresse (optional)

## Teilnehmer/innen Daten

### Teilnehmer/innen Daten

---

Alter:  ≤ 25  26 - 30  31 - 40  41 - 50  ≥ 51

Geschlecht:  weiblich  männlich

Seit wann sind Sie im Internet aktiv?

Wie viele Stunden am Tag verbringen Sie durchschnittlich im Internet?

0 Stunden  ≤ 2 Stunden  ≤ 4 Stunden  ≥ 5 Stunden

Wie viele Stunden am Tag schauen Sie durchschnittlich Fernsehen?

0 Stunden  ≤ 2 Stunden  ≤ 4 Stunden  ≥ 5 Stunden

Welche der Aussagen trifft auf Sie zu?

Aussage	Ja	Nein
Ich schaue fern und informiere mich gleichzeitig im Internet über das Gesehene.		
Ich sehe eine Fernsehsendung oder Video und unterhalte mich in einer Onlinecommunity über den Medieninhalt.		
Ich sehe eine Fernsehsendung und kommentiere etwas zu dieser Sendung in Social Medien wie Twitter oder Facebook.		
Ich nutze zusätzliche Apps wie zum Beispiel ProSieben Connect oder 7TV parallel zur zum Mediengeschehen.		

## Wissenstest

### Wissenstest

---

Es werden fünf Aussagen aus der Dokumentation mit jeweils drei Antwortmöglichkeiten angegeben. Bitte kreuzen Sie die Antwortmöglichkeit an, die Ihrer Meinung nach am besten passt.

- 1) Wie viele Menschen benutzen den Mail-Dienst Gmail von Google pro Tag?
  - a. 500 Millionen
  - b. 600 Millionen
  - c. 700 Millionen
  
- 2) Wie viele Menschen arbeiten weltweit bei Google?
  - a. 50.000
  - b. 60.000
  - c. 70.000
  
- 3) Wann wurde die erste Beschwerde gegen Google in Europa eingereicht?
  - a. 2009
  - b. 2010
  - c. 2011
  
- 4) Wie viel Stunden Material wird auf Youtube pro Minute hochgeladen?
  - a. 120
  - b. 110
  - c. 100
  
- 5) Wie viele Menschen benutzen die Kartenanwendung Google Maps pro Monat?
  - a. 0.8 Mrd.
  - b. 1 Mrd.
  - c. 1.3 Mrd.

## Usability-Untersuchung

# Usability-Untersuchung

---

Haben Sie bereits Erfahrung im Umgang mit dem Betriebssystem Android?

ja     nein

Wenn ja, welche: .....

Bitte zu jeder Frage das rechts stehende Kästchen ankreuzen. Es wird nach einem Benotungssystem vorgegangen, bei dem die **1** für „volle Zustimmung“ bzw. „sehr gut“ und die **5** für „volle Ablehnung“ bzw. „sehr schlecht“ steht.

**Fachliche Fragen:**

- |  | <b>1</b>                 | <b>2</b>                 | <b>3</b>                 | <b>4</b>                 | <b>5</b>                 |
|--|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Wie ist Ihr Gesamteindruck von der App?                              | <input type="checkbox"/> |
| Die Nutzung der App hat das Fernseherlebnis verbessert.              | <input type="checkbox"/> |
| Die Nutzung der App hat mich vom Fernsehen abgelenkt.                | <input type="checkbox"/> |
| Die App hat meine Informationsbedürfnisse zum gezeigten unterstützt. | <input type="checkbox"/> |
| Es war einfach Informationen aus der Anwendung zu ziehen.            | <input type="checkbox"/> |
| Würden Sie die App auch privat nutzen wollen?                        | <input type="checkbox"/> |

**Technische Fragen:**

- |  | <b>1</b>                 | <b>2</b>                 | <b>3</b>                 | <b>4</b>                 | <b>5</b>                 |
|--|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| War die Bedienung der App intuitiv und einfach zu verstehen? | <input type="checkbox"/> |
| War die Menüstruktur für Sie klar und verständlich?          | <input type="checkbox"/> |
| Lief die App flüssig und problemlos?                         | <input type="checkbox"/> |
| Hat Sie das Design angesprochen?                             | <input type="checkbox"/> |

Welche Fehler in der Anwendung sind Ihnen aufgefallen?


Welche Verbesserungsmöglichkeiten haben Sie für die App?


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

Hamburg, 30. November 2015 Ivan Demin