

# Bachelorarbeit Abteilung Informatik

Titel der Bachelorarbeit  
**„Mobile Quiz“**

Frühlingssemester 2012

20. Februar 2012 - 15. Juni 2012

Autoren: Khalid Abdul und Patrik Naef  
Betreuer: Prof. Dr. Peter Heinzmann  
Experte: Dr. Th. Siegenthaler, CSI Consulting AG  
Gegenleser: Prof. Stefan F. Keller

Arbeitsumfang: 360 Arbeitsstunden bzw. 12 ECTS pro Student



## **Abstract**

Während Vorlesungen finden häufig kurze mündliche Umfragen und Lernkontrollen statt, die direkt an die Teilnehmenden gerichtet sind. In solchen Situationen ist der Rücklauf der Antwortenden eher bescheiden, da die Teilnehmenden ihre Antwort öffentlich vor ihren Kommilitonen kundtun müssen und die Auswertung selbst nicht transparent ist. Bei Audience-Response-Systeme können die Teilnehmenden mit Hilfe spezieller Eingabegeräte antworten. Für Studierende ist dies attraktiver, da sie ihre Meinung anonym abgeben können.

Im Rahmen der vorliegenden Bachelorarbeit soll ein webbasierter Prototyp realisiert werden, in dem die speziellen Eingabegeräte durch Laptops, Tablets oder Smartphones ersetzt werden.

Dozierende können Lernkontrollen und Umfragen erstellen und haben Möglichkeit, unterschiedliche Durchführungen zu planen. So können die Resultate einer bestimmten Zielgruppe evaluiert werden. Dozierende können bei der Erfassung von Lernkontrollen angeben, ob Antworten richtig oder falsch sind. Bei Umfragen geht es darum, Tendenzen einer Gruppe zu ermitteln.

Normalerweise werden Umfragen von den Teilnehmenden autonom und in eigener Geschwindigkeit ausgefüllt. Zudem bietet Mobile Quiz die Möglichkeit, eine Umfrage im Präsentationsmodus zu starten. Hierbei werden die Teilnehmenden durch die Umfrage geführt und die Resultate zur jeweiligen Frage auf dem Präsentationsmedium fortlaufend aktualisiert.

Lernkontrollen werden von Teilnehmenden autonom ausgefüllt. Dozierende haben bei Lernkontrollen eine vielfältige Auswertungsmöglichkeit, die eine detaillierte Analyse der Durchführung zulässt.

## Aufgabenstellung

Studiengang: Informatik (I)

Semester: FS 2012 (20.02.2012 - 16.09.2012)

---

Institut: ITA: Internet-Technologien und -Anwendungen

Gruppe: Khalid Abdul, Patrik Naef

---

Verantwortlicher: Prof. Dr. Peter Heinzmann

Gegenleser: Prof. Stefan Keller

Experte: Dr. Th. Siegenthaler, CSI AG, Zürich

Industriepartner: HSR, Abteilung Informatik

## Ausgangslage

Mit sogenannten Audience-Response-Systemen bieten Referenten ihren Zuhörern die Möglichkeit, mit Hilfe spezieller Eingabegeräte an Umfragen teilzunehmen. Auch in Vorlesungen und Kursen finden häufig kleine Umfragen oder Lernkontrollen statt. An der HSR wurde ein System mit speziellen Eingabegeräten (Clickern) entwickelt. An der ETH Zürich werden bereits systematisch Prüfungen auf dem Computer abgenommen. Mittlerweile gibt es eine Vielzahl von Mobile Apps, welche Smartphones mit Audience-Response-System-Funktionalität ausstatten. Besonders interessant ist das Plugin Quiz & Poll für Google Docs.

## Ziel

Im Rahmen dieser Bachelorarbeit soll ein System realisiert werden, mit welchem Lernkontrollen und Umfragen erstellt und durchgeführt werden können. Die Teilnehmenden sollen per Uniform Resource Locator oder QR-Code eingeladen werden. Studierende können die Fragen per Laptop, Tablet oder Smartphone beantworten. Die Fragen sollen am Präsentationsmedium und/oder auf den Endgeräten der Teilnehmenden angezeigt werden. Für die Anzeige der Resultate soll es zwei Möglichkeiten geben: unmittelbar bei der Beantwortung der Fragen oder am Ende der Durchführung. Ferner sollen statistische Auswertungen möglich sein.

Das System soll via Internet für die Allgemeinheit verfügbar sein.

Rapperswil, den.....

.....  
Verantwortlicher Professor



## Erklärung<sup>1</sup>

Die vorliegende Arbeit basiert auf Ideen, Arbeitsleistungen, Hilfestellungen und Beiträgen gemäss folgender Aufstellung:

Gegenstand, Leistung	Person	Funktion
Frontend, JavaScript, jQuery	Khalid Abdul	Autor der Arbeit
Backend, PHPUnit, Doctrine	Patrik Naef	Autor der Arbeit
Idee, Aufgabenstellung, allgemeines Pflichtenheft, Betreuung während der Arbeit	Prof. Dr. Peter Heinzmann	Verantwortlicher Professor

Ich erkläre hiermit

- dass ich die vorliegende Arbeit gemäss obiger Zusammenstellung selber und ohne weitere fremde Hilfe durchgeführt habe,
- dass ich sämtliche verwendeten Quellen erwähnt und gemäss gängigen wissenschaftlichen Zitierregeln korrekt angegeben habe.
- dass ich keine durch Copyright geschützten Materialien (z.B. Bilder) in dieser Arbeit in unerlaubter Weise genutzt habe.

Rapperswil, den.....  
Student

Rapperswil, den.....  
Student

---

<sup>1</sup>Diese Erklärung basiert auf der Muster-Erklärung in den Richtlinien der HSR zur Durchführung von Projekt-, Studien-, Diplom- oder Bachelorarbeiten vom 16. Februar 2009.

---

## Vereinbarung<sup>2</sup>

### 1. Gegenstand der Vereinbarung

Mit dieser Vereinbarung werden die Rechte über die Verwendung und die Weiterentwicklung der Ergebnisse der Bachelorarbeit „Mobile Quiz“ von Khalid Abdul und Patrik Naef unter der Betreuung von Prof. Dr. P. Heinzmann (für die Arbeit verantwortlicher Professor) geregelt.

### 2. Urheberrecht

Die Urheberrechte stehen der Studentin / dem Student zu.

### 3. Verwendung

Die Ergebnisse der Arbeit dürfen sowohl von allen an der Arbeit beteiligten Parteien, d.h. von den Studenten, welche die Arbeit verfasst haben, vom verantwortlichen Professor sowie vom Industriepartner verwendet und weiter entwickelt werden. Die Namensnennung der beteiligten Parteien ist bei der Weiterverwendung erwünscht, aber nicht Pflicht.

Rapperswil, den.....  
.....  
Student

Rapperswil, den.....  
.....  
Student

Rapperswil, den.....  
.....  
Verantwortlicher Professor

---

<sup>2</sup>Diese Vereinbarung basiert auf den Muster-Vereinbarungen in den Richtlinien der HSR zur Durchführung von Projekt-, Studien-, Diplom- oder Bachelorarbeiten vom 16. Februar 2009.

# Mobile Quiz Kurzzusammenfassung (Broschürentext)

**Ausgangslage:** Während Vorlesungen finden häufig kurze mündliche Umfragen und Lernkontrollen statt, die direkt an die Teilnehmenden gerichtet sind. In solchen Situationen ist der Rücklauf der Antwortenden eher bescheiden, da die Teilnehmenden ihre Antwort öffentlich vor ihren Kommilitonen kundtun müssen und die Auswertung selbst nicht transparent ist. Bei Audience-Response-Systeme können die Teilnehmenden mit Hilfe spezieller Eingabegeräte antworten. Für Studierende ist dies attraktiver, da sie ihre Meinung anonym abgeben können.

Im Rahmen der vorliegenden Bachelorarbeit soll ein webbasierter Prototyp realisiert werden, in dem die speziellen Eingabegeräte durch Laptops, Tablets oder Smartphones ersetzt werden.

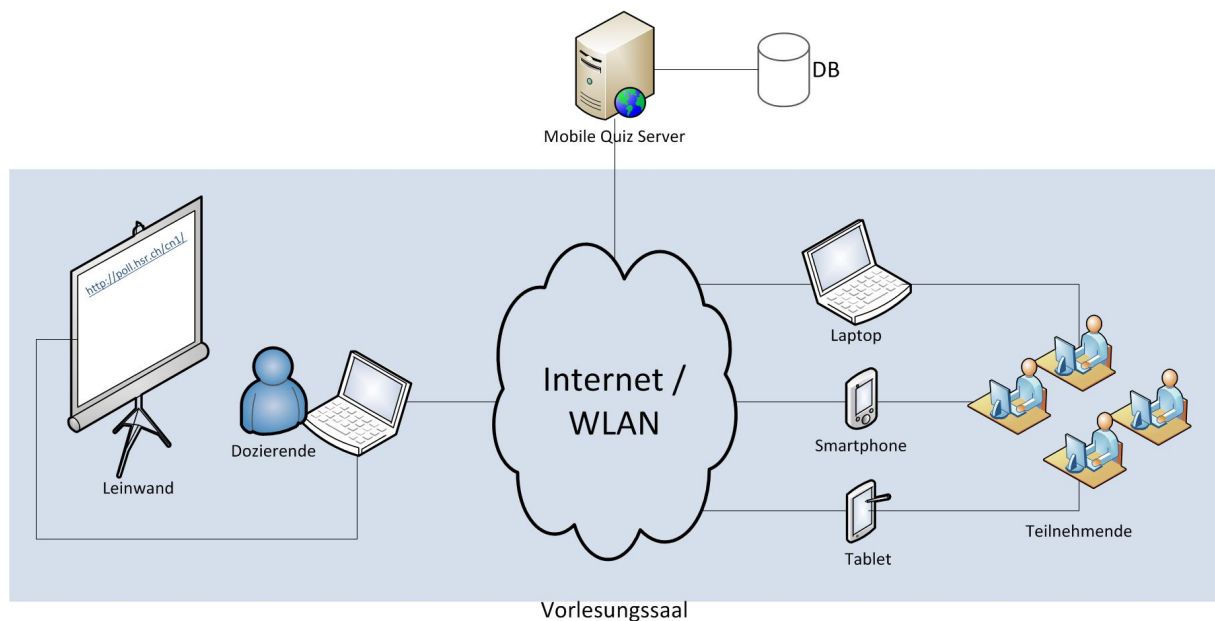


Abbildung 0.1.: Anwendung der Applikation

**Vorgehen/Technologie:** Zu Beginn der Arbeit wurde der Funktionsumfang der Mitbewerber evaluiert und ausgewertet, um die funktionalen Anforderungen an die Anwendung zu ermitteln. Weiter wurden durch Interviews mit potentiellen Nutzern zusätzliche Bedürfnisse geklärt. In der Entwicklungsphase stand der Kunde im Zentrum der Entwicklung.

In regelmässigen Abständen wurden Bedienbarkeitstests vollzogen. Die daraus erhaltenen Rückmeldungen flossen in die Weiterentwicklung der Benutzeroberfläche ein. Clientseitig wurde auf die Technologien HTML5, CSS3 und JavaScript gesetzt, während auf der Serverseite die Technologien PHP 5.3 und MySQL 5.5 zum Einsatz kommen.

**Ergebnis:** Mit dem entwickelten Prototypen können Umfragen und Lernkontrollen erstellt, durchgeführt und ausgewertet werden. Teilnehmende werden per Uniform Resource Locator bzw. QR-Code zur Teilnahme eingeladen, um Lernkontrollen autonom auszufüllen. Bei Umfragen werden die Teilnehmenden im Präsentationsmodus durch die Fragen geleitet. Die Fragen und Resultate werden fortlaufend auf dem Präsentationsmedium aktualisiert.

Zur Erstellung der Umfragen und Lernkontrollen steht eine intuitive Benutzeroberfläche zur Verfügung. Der Ersteller kann für die Teilnahme aus den drei Modi authentisiert, pseudonym und anonym auswählen. Im Bereich Lernkontrollen stehen vielfältige Auswertungsmöglichkeiten zur Verfügung. Sie lassen eine detaillierte Analyse der Durchführungen zu, wobei der Fokus auf einzelne Teilnehmende gelegt werden kann.

Von Prof. Dr. Peter Heinzmann wird die Anwendung in Zukunft im Unterricht eingesetzt. Im Laufe der Zeit werden eventuell weitere Dozierende der HSR die Anwendung einsetzen. Längerfristig könnte die Anwendung auch in anderen Organisationen, wie Berufs- oder Mittelschulen zum Einsatz kommen.

# Management Summary

**Ausgangslage:** Innerhalb von Vorlesungen finden häufig kleine Umfragen oder Lernkontrollen statt, die direkt an die Teilnehmenden gerichtet sind. In solchen Situationen ist der Rücklauf der Antwortenden eher bescheiden, da die Teilnehmenden ihre Antwort offen vor ihren Kommilitonen kundtun müssen. Audience-Response-Systeme sind eine Möglichkeit für Dozierende, Fragen direkt an die Teilnehmenden zu richten und diese zu einer aktiven Teilnahme zu bewegen, ohne dass sie sich vor ihren Kommilitonen exponieren. Dabei kommen spezielle Eingabegeräte zum Einsatz, um die Fragen direkt zu beantworten. Dozierende können anhand solcher Umfragen oder Lernkontrollen ermitteln, wie das Publikum die behandelten Lerninhalte aufnimmt und Teilnehmende können ihren Wissensstand überprüfen.

Im Rahmen dieser Bachelorarbeit wurde ein stabiler webbasierter Prototyp realisiert, in dem die systemspezifischen Eingabegeräte durch Laptops, Tablets oder Smartphones ersetzt werden. Mindestens eines dieser Geräte tragen Studierende im Normalfall auf sich. Die realisierte Anwendung spricht sowohl die Dozierenden, als auch die Studierenden an.

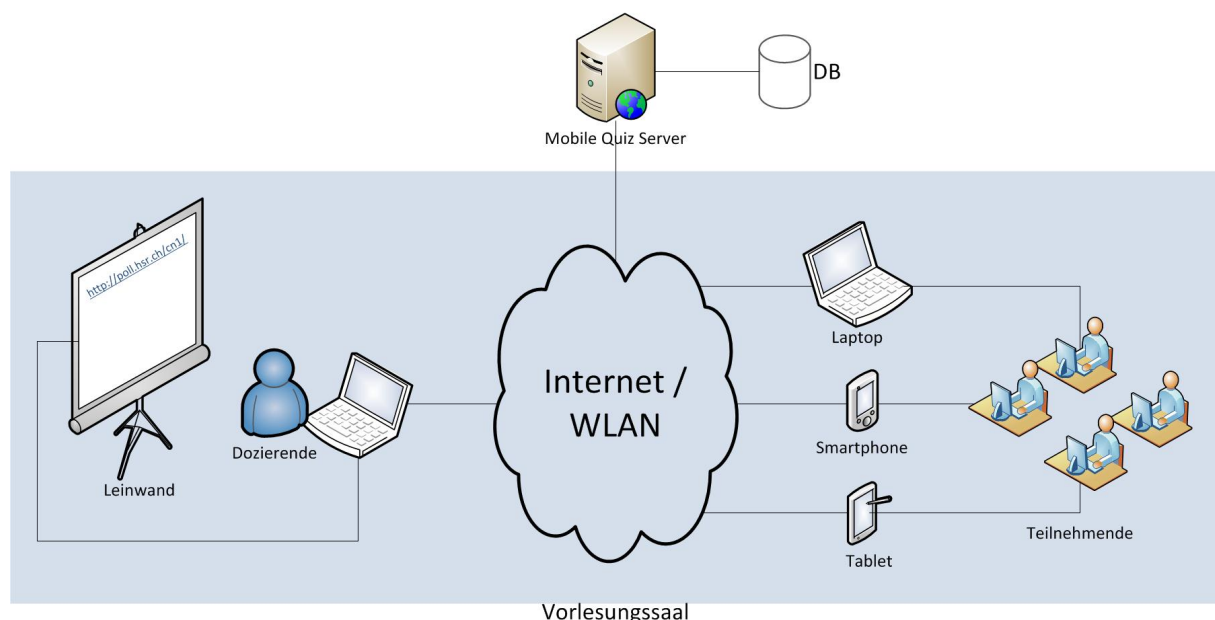


Abbildung 0.2.: Anwendung der Applikation

**Vorgehen/Technologie:** Zu Beginn der Arbeit wurden die Funktionen der Mitbewerber evaluiert und ausgewertet, um die funktionalen Anforderungen an die Anwendung zu

ermitteln. Das Augenmerk lag nicht nur auf den Funktionen der Mitbewerber, sondern auch auf der Art und Weise ihrer Realisierung. Die eruierten Funktionen ergänzten die bestehenden Anforderungen, die bisher von keinem Mitbewerber vollumfänglich abgedeckt wurden. Zudem wurden Interviews mit der potentiellen Zielgruppe geführt, um neue Bedürfnisse zu erörtern.

Da die nichtfunktionalen Anforderungen eine plattformübergreifende Anwendung umfassen, wurde für die Realisierung PHP 5.3.8 ausgewählt. Zur Persistierung der Daten wird MySQL 5.5.16 kombiniert mit dem Framework Doctrine 2 eingesetzt. Zur Darstellung wurden die Auszeichnungssprachen HTML5 und CSS3 verwendet. Um die Präsentationsschicht von der Datenhaltung und Implementierung zu trennen, wird die Smarty Template Engine verwendet. Somit wird gewährleistet, dass die Präsentationsschicht alternativen Darstellungsanforderungen weichen kann. Zur Sicherung der Qualität wird die Kernarchitektur mittels Test Driven Development implementiert, welche jederzeit eine Überprüfung der Implementationsqualität ermöglicht.

**Ergebnis:** Der funktionsfähige und stabile Prototyp umfasst die wichtigsten Funktionen der Anforderungsspezifikation. Die Anwendung ermöglicht die Erstellung von Umfragen und Lernkontrollen. Mittels portablen Geräten, wie Laptops, Tablets und Smartphones, wird daran teilgenommen. Normalerweise werden Umfragen von den Teilnehmenden autonom und in eigener Geschwindigkeit ausgefüllt. Eine Umfrage kann ebenfalls im Präsentationsmodus gestartet werden. Hierbei werden die Teilnehmenden durch die Umfrage geführt und die Resultate zur jeweiligen Frage auf dem Präsentationsmedium fortlaufend aktualisiert. Lernkontrollen werden von Teilnehmenden autonom ausgefüllt. Dozierende haben bei Lernkontrollen vielfältige Auswertungsmöglichkeiten. Sie lässt eine detaillierte Analyse der Durchführung zu, wobei der Fokus auf einzelne Teilnehmende gelegt werden kann. Die Teilnahme an einer Umfrage oder Lernkontrolle erfolgt in einem der folgenden Modi:

- Authentisiert: In der Auswertung ist der Vor- und Nachname des Teilnehmenden ersichtlich.
- Pseudonym: In der Auswertung ist das Pseudonym des Teilnehmenden ersichtlich.
- Anonym: In der Auswertung wird der Alias des Teilnehmenden angezeigt.

**Ausblick:** Der momentane Entwicklungsstand der Anwendung erlaubt es den Dozierenden, diese in ihren Vorlesungen einzusetzen. In Zukunft wären kleinere Anpassungen des Designs denkbar, da während der Bachelorarbeit die Funktionalität der Anwendung im Vordergrund stand und das Design bewusst schlicht gehalten wurde. Dies ist jedoch dank HTML5 und CSS3 einfach zu bewerkstelligen. Ausserdem ist es möglich, dass die Performance der Anwendung optimiert werden muss. Jedoch ist dies ohne Langzeittests in einem realen Umfeld lediglich eine Annahme. Zusätzlich wird empfohlen einen neuen Modus einzubauen, in der sich Teilnehmende vor Beginn der Teilnahme nicht anmelden bzw. registrieren müssen. Dies würde den Dozierenden, die nicht an Langzeitauswertungen interessiert sind, einiges an Zeit während der Vorlesung einsparen. Eine multilinguale Benutzeroberfläche wäre ebenfalls eine Erweiterung, die zukünftig vorgenommen werden könnte. Die notwendigen Vorarbeiten dafür wurden allesamt bereits geleistet.

# Inhaltsverzeichnis

<b>I. Einführung</b>	<b>14</b>
<b>1. Einführung</b>	<b>15</b>
1.1. Ausgangslage . . . . .	15
1.2. Kontextdiagramm . . . . .	16
<b>2. Grundlagen</b>	<b>17</b>
2.1. Lernkontrollen . . . . .	17
2.2. Umfragen . . . . .	18
2.3. Teilnahme . . . . .	18
2.4. Punktesystem . . . . .	19
2.5. Authentisierung . . . . .	19
2.6. Datensicherheit . . . . .	20
2.7. Bedrohungen und Verletzlichkeiten . . . . .	21
<b>II. Requirements Engineering</b>	<b>23</b>
<b>3. Analyse ähnlicher Anwendungen</b>	<b>24</b>
<b>4. Interviews</b>	<b>25</b>
<b>5. Funktionsliste</b>	<b>27</b>
<b>6. Produktbeschreibung</b>	<b>29</b>
6.1. Dozierende . . . . .	29
6.2. Teilnehmende . . . . .	34
<b>III. Software Engineering</b>	<b>38</b>
<b>7. User Interface</b>	<b>39</b>
7.1. Übersicht . . . . .	39
<b>8. Architektonische Darstellung</b>	<b>41</b>
8.1. Deployment Diagramm . . . . .	41
8.2. Domainmodell . . . . .	42
8.3. Klassendiagramm . . . . .	43
8.4. Entity-Relationship-Diagramm . . . . .	46
8.5. Logische Architektur . . . . .	48
8.6. Sequenzdiagramme . . . . .	56

<b>9. Verwendete Technologien</b>	<b>57</b>
9.1. Entwicklungsumgebung . . . . .	57
9.2. Frameworks . . . . .	57
 <b>IV. Qualitätssicherung</b>	 <b>59</b>
<b>10. Testing</b>	<b>60</b>
10.1. Systemtests . . . . .	60
10.2. Informelles zum Code . . . . .	61
10.3. Usability Tests . . . . .	62
<b>11. Reviews</b>	<b>68</b>
11.1. Code-Review . . . . .	68
11.2. Dokument-Review . . . . .	68
 <b>V. Schlussfolgerung</b>	 <b>69</b>
<b>12. Schlussfolgerung</b>	<b>70</b>
12.1. Zusammenfassung . . . . .	70
12.2. Ausblick . . . . .	71
 <b>VI. Verzeichnisse</b>	 <b>73</b>
12.3. Quellenverzeichnis . . . . .	74
<b>13. Glossar</b>	<b>77</b>
<b>14. Bilderverzeichnis</b>	<b>79</b>
<b>15. Tabellenverzeichnis</b>	<b>81</b>
 <b>VII. Anhang</b>	 <b>82</b>
<b>A. Vision</b>	<b>83</b>
A.1. Einführung . . . . .	83
A.2. Positionierung . . . . .	83
A.3. Allgemeine Schlüsselziele und Probleme der Dozierenden und Teilnehmenden	85
A.4. Produktübersicht . . . . .	85
<b>B. Analyse der Mitbewerber</b>	<b>88</b>
B.1. Analyse . . . . .	88
B.2. Detailanalyse . . . . .	97
B.3. Auswertung . . . . .	98
<b>C. Anforderungsspezifikation</b>	<b>100</b>
C.1. Einführung . . . . .	100
C.2. Allgemeine Beschreibung . . . . .	101



C.3. Funktionale Anforderungen . . . . .	105
C.4. Nichtfunktionale Anforderungen . . . . .	106
C.5. Qualitätsmerkmale . . . . .	108
C.6. Andere Anforderungen . . . . .	108
<b>D. Erfahrungsberichte</b>	<b>110</b>
D.1. Khalid Abdul . . . . .	110
D.2. Patrik Naef . . . . .	111
<b>E. Projektmanagement</b>	<b>112</b>
E.1. Management Abläufe . . . . .	112
E.2. Risikomanagement . . . . .	115
E.3. Zeitauswertung . . . . .	115
<b>F. Werkzeuge und Tools</b>	<b>120</b>
F.1. Werkzeuge . . . . .	120
F.2. Tools . . . . .	122
<b>G. Einrichtung Produktivumgebung</b>	<b>124</b>

# **Teil I.**

## **Einführung**

# 1. Einführung

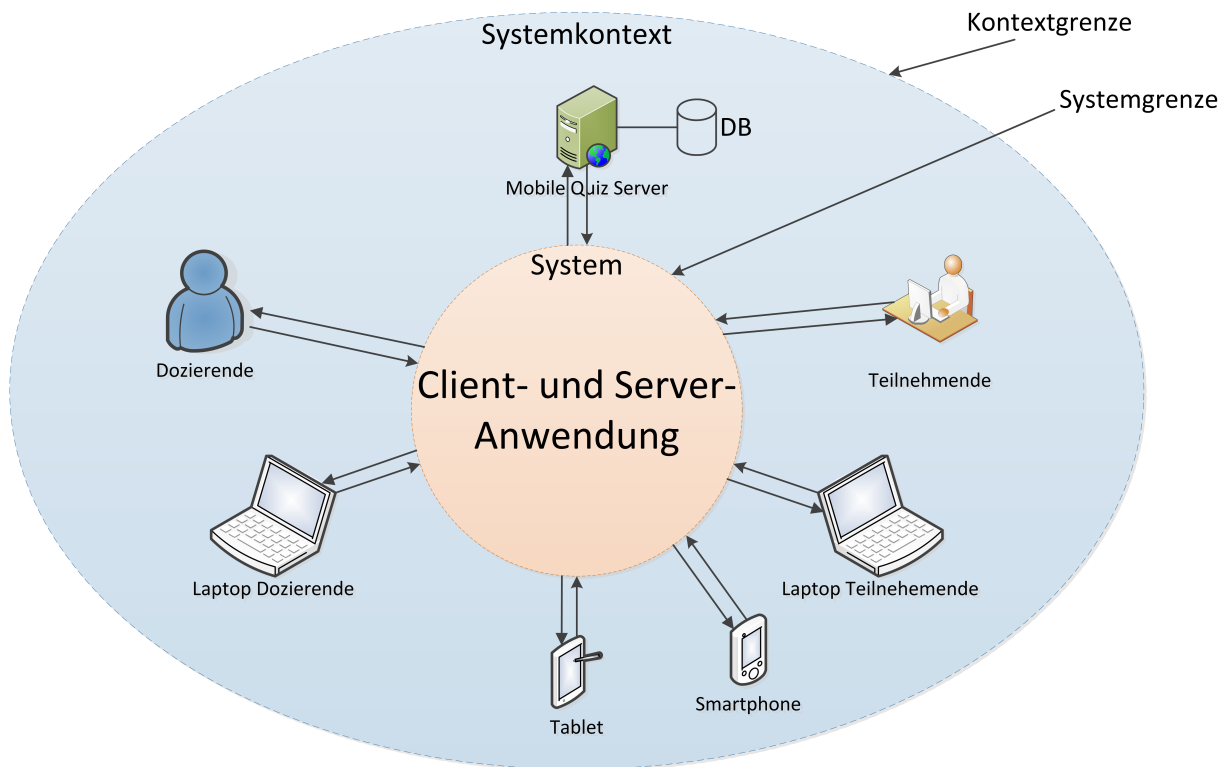
## 1.1. Ausgangslage

Dozierende an der Hochschule für Technik in Rapperswil haben eine webbasierte Anwendung erhalten, die sie beim Lehrauftrag unterstützt. Anhand von Lernkontrollen und Umfragen kann der Lernfortschritt der einzelnen Studierenden ausgewertet werden. So sind Dozierende in der Lage, die behandelten Stoffe gegebenenfalls zu repetieren oder profundierter zu behandeln. Studierende der HSR erhalten eine Anwendung, die sie durch das Semester begleitet. Mit Hilfe von Mobile Quiz ist es ihnen möglich, ihren Lernfortschritt selbst zu überprüfen. Mobile Quiz ist damit für Dozierende, wie auch für Studierende eine Bereicherung. Dozierende können Fragen direkt an Teilnehmende richten und diese zu einer aktiven Teilnahme bewegen, ohne dass sie sich vor ihren Kommilitonen exponieren.

In der Vergangenheit hatten Dozierende keine Möglichkeit, oder nur mit erhöhtem Aufwand, während des Semesters die Lernfortschritte der Studierenden zu überprüfen. Beispielsweise war es nur mit schriftlichen Prüfungen möglich, die Lernkurve des Moduls zu erfassen. Studierende waren auf sich alleine gestellt, was die Überprüfung ihrer erworbenen Kenntnisse anging.

Diese beiden Schwachpunkte wurden durch die Implementierung von Mobile Quiz massiv vereinfacht.

## 1.2. Kontextdiagramm



**Abbildung 1.1.:** Das Kontextdiagramm der Anwendung

Das Kontextdiagramm beschreibt die Rollen, Dozierende und Studierende, die im System vorhanden sind. Weiter wird beschrieben, durch welche Geräte mit dem System interagiert werden kann. Das System wird auf einem Webserver betrieben und hat eine Verbindung zu einer Datenbank.

## 2. Grundlagen

Grundsätzlich gibt es zwei Arten der Anwendung von Mobile Quiz. Lernkontrollen (Quiz) dienen dazu Fachwissen abzufragen, während mit Umfragen (Poll) Tendenzen ermittelt werden.

### 2.1. Lernkontrollen

Eine ist die Lernkontrolle, bei der die Teilnehmenden Fragen nacheinander beantworten können und das Resultat, je nach Einstellung, direkt nach der Beantwortung der einzelnen Frage oder am Ende der Durchführung sehen. Bei einer Lernkontrolle handelt es sich um einen Multiple-Choice-Test. Ein Multiple-Choice-Test erlaubt es den Teilnehmenden lediglich zwischen zwei Zuständen zu wählen, nämlich richtig und falsch. Dies bedeutet, dass Teilnehmende keinerlei Möglichkeit haben die Frage nicht zu beantworten. Normalerweise werden Multiple-Choice-Fragen mit einem Kästchen dargestellt, welche die zwei Status repräsentieren.

Im Gegensatz zu regulären Multiple-Choice-Tests, bietet die Anwendung die Möglichkeit, eine Frage nicht zu beantworten. Dies aus dem Grund, weil es in regulären Semesterprüfungen an der Hochschule Rapperswil ebenfalls möglich ist, sich der Stimme zu enthalten und um die sogenannte Testwisseness (“Testfähigkeit”) der Teilnehmenden zu verhindern. Unter Testwisseness versteht man das Prinzip, bei schlecht konzipierten Fragen die richtige Antwort zu erraten. Dabei gibt es verschiedene Tipps (Clues)<sup>[1][2]</sup>, an die sich Teilnehmende halten können:

1. Die korrekte Antwort ist womöglich die längste Antwort, da der Ersteller die richtige Antwort am präzisesten zu beschreiben versucht.
2. Eine Antwort, die Rechtschreibfehler enthält, kann nicht richtig sein. Beim Kontrollieren der Fragen und Antworten durch den Ersteller werden meist nur die korrekten Antworten durchgeschaut. So werden Fehler in den falschen Antworten mit grosser Wahrscheinlichkeit nicht erkannt.
3. Die Antwort, in der ein Schlüsselwort der Frage wiederholt wird, wird vermutlich korrekt sein.
4. Antworten, die Wörter wie “alle”, “jeden” und “immer” enthalten, können mit grosser Wahrscheinlichkeit ausgeschlossen werden.
5. Wenn mehrere Antworten das Gleiche bedeuten oder aussagen, werden sie vermutlich falsch sein.
6. Falls kein anderer Tipp zur möglichen Lösung führt und raten die letzte Möglichkeit ist, wird die Antwort C richtig sein.

Das Einführen einer neutralen Antwort bietet dem Ersteller und den Teilnehmenden einen wesentlichen Vorteil. Die Teilnehmenden werden nicht gezwungen eine Antwort als richtig oder falsch anzukreuzen. So wird auch das Durcharbeiten der oben genannten Tipps verhindert und die Teilnehmenden können bei Zweifeln auf die neutrale Antwort zurückgreifen.

## 2.2. Umfragen

Die zweite Art ist die Umfrage. Im Gegensatz zu einer Lernkontrolle können die Teilnehmenden nur eine der Antworten (Single-Choice) wählen. Bei einer Umfrage geht es nicht darum, Fachwissen abzufragen und auf ihre Korrektheit zu prüfen, sondern es geht darum zu sehen, wie die Teilnehmenden zu einem Thema stehen. Dies kann beispielsweise dann der Fall sein, wenn Dozierende wissen wollen, wie viele der Benutzer bereits ein Smartphone besitzen, oder wie der Unterricht in der nächsten Lektion gestaltet werden soll.

Die Anwendung bietet den Dozierenden die Möglichkeit, Umfragen zu gestalten, die entweder während den Lektionen oder auch in der Freizeit von den Teilnehmenden ausgefüllt werden können. Im Normalfall können die Dozierenden, ähnlich wie bei der Lernkontrolle, die Umfrage starten und die Teilnehmenden diese ausfüllen lassen. Falls die Dozierenden die Umfrage während der Lektion starten, wird diese von ihnen geleitet. Dies bedeutet, dass eine Frage erst dann freigegeben wird, wenn der Dozierende dies erlaubt. Dabei kann am Projektionsmedium fortlaufend verfolgt werden, wie die Teilnehmenden votierten. Diese Methode erlaubt es den Dozierenden eine Umfrage zu Beginn der Lektion zu starten und am Ende abzuschliessen. So müssen nicht mehrere Umfragen erstellt und an die Teilnehmenden verschickt werden.

## 2.3. Teilnahme

Während den Vorlesungen der letzten Jahre hat sich gezeigt, dass einige Dozenten im Unterricht gerne und viele Fragen stellen, andere hingegen weniger. In den Interviews, die in der Analyse-Phase mit den Dozierenden der Hochschule für Technik Rapperswil geführt wurden, wurden die Hintergründe für die verschiedenen Auffassungen erläutert. Aus den meisten Interviews ist abzuleiten, dass seitens der Teilnehmenden eine relativ geringe Resonanz bezüglich solcher Umfragen oder Lernkontrollen vorherrscht. Über die Gründe lässt sich lediglich spekulieren. Einerseits machen sie nicht aktiv mit, weil die gestellte Frage zu einfach ist, andererseits nehmen sie nicht teil, weil die Frage zu schwierig erscheint und sich nicht vor ihren Kommilitonen mit einer falschen Antwort blossstellen wollen. Aus diesem Grund bietet die Anwendung drei Modi zur Absolvierung von Lernkontrollen/Umfragen:

- Authentisiert: In der Auswertung ist der Vor- und Nachname der Teilnehmenden ersichtlich.
- Pseudonym: In der Auswertung ist das Pseudonym der Teilnehmenden ersichtlich.
- Anonym: In der Auswertung ist der Alias der Teilnehmenden ersichtlich.

Diese Einstellung wird den Teilnehmern auf der Startseite der Lernkontrolle bzw. der Umfrage angezeigt. Die Teilnehmenden werden vor Beginn jeder Teilnahme aufgefordert

sich anzumelden. Auch bei Umfragen bzw. Lernkontrollen die anonym ausgefüllt werden. Dies aus dem Grund, dass Dozierende über ein gesamtes Semester die Lernkurve eines anonymen Teilnehmenden verfolgen wollen. Da die Anwendung verschiedene Eingabegeräte zur Teilnahme zulässt, ist dies mit herkömmlichen Mitteln, wie dem Speichern eines generierten Tokens in einem Cookie, nicht umsetzbar.

## 2.4. Punktesystem

An der HSR ist es üblich, dass bei Multiple-Choice-Fragen in einer Prüfung die Möglichkeit besteht, sich der Stimme zu enthalten. So ist es für die Prüfungsteilnehmenden möglich, das Risiko eines Minuspunktes zu verhindern. In der Anwendung wird dasselbe Prinzip verfolgt. Stimmt der Tipp mit der Antwort nicht überein, kassiert der Teilnehmende einen Minuspunkt. Damit nicht ein Richtig oder Falsch abgegeben werden muss, wird an Stelle von konventionellen Check-boxen und Radiobuttons ein Schalter angezeigt, der drei Status darstellt: falsch, neutral und richtig. Die Punkte berechnen sich so, dass bei einer Multiple-Choice-Frage mit fünf Antworten, in der drei richtig und zwei falsch sind, minimal -5 Punkte und maximal 5 Punkte erzielt werden können. Wenn der Schalter auf neutral ist, enthält sich der Student der Stimme und erhält weder einen Plus- noch Minuspunkt. Die Abbildung 2.2 erläutert das Prinzip.



Abbildung 2.1.: Tristate Toggle Button

	Richtige Antwort #1	+1
	Richtige Antwort #2	+1
	Richtige Antwort #3	+1
	Falsche Antwort #1	+1
	Falsche Antwort #2	+1
5 Punkte		
	Richtige Antwort #1	+1
	Richtige Antwort #2	+1
	Richtige Antwort #3	-1
	Falsche Antwort #1	+1
	Falsche Antwort #2	+1
3 Punkte		
	Richtige Antwort #1	+1
	Richtige Antwort #2	+1
	Richtige Antwort #3	+/- 0
	Falsche Antwort #1	+1
	Falsche Antwort #2	+1
4 Punkte		

Abbildung 2.2.: Beispiel der Punktevergabe

## 2.5. Authentisierung

Die Authentisierung beschreibt das Nachweisen der Identität eines Benutzers.

In der Anwendung ist es möglich, sich als Benutzer direkt auf der Webplattform zu registrieren. Dies bedeutet, dass in der Datenbank Passwörter und weitere Benutzerinformationen gespeichert werden. Wie die Sicherheit der Passwörter gewährleistet wird, ist im Kapitel 2.6. “Datensicherheit” näher beschrieben. Zudem ermöglicht die Anwendung den Benutzern, sich über die momentan gängigen sozialen Netzwerke Facebook[3], Twitter[4] und Google[5] zu registrieren und anzumelden. Dies bietet den wesentlichen Vorteil, dass die Überprüfung der Nutzerdaten durch die Drittanbieter erfolgt. Von diesen erhält die Anwendung lediglich Auskunft darüber, ob die Nutzerinformationen korrekt sind oder nicht. Dadurch, dass die Anwendung die Interaktionen und die erstellten Lernkontrollen/Umfragen in der Datenbank dem Benutzer zuordnet, muss sich dieser beim

ersten Anmelden registrieren. Dabei wird der Benutzer mit den Attributen ID der Drittanbieter, Benutzername, Vorname, Nachname und E-Mail Adresse erfasst. Ein Teil dieser Informationen werden bei der Registrierung direkt dem Benutzerprofil der Drittanbieter entnommen. Dies ist von Plattform zu Plattform unterschiedlich. Bei Facebook können alle oben genannten Daten im Profil ermittelt werden. Bei Twitter fehlt die E-Mail Adresse. Dies ist aus Gründen der Design-Entscheidung der Twitter-API nicht möglich[6]. Bei Google fehlt lediglich der Benutzername.

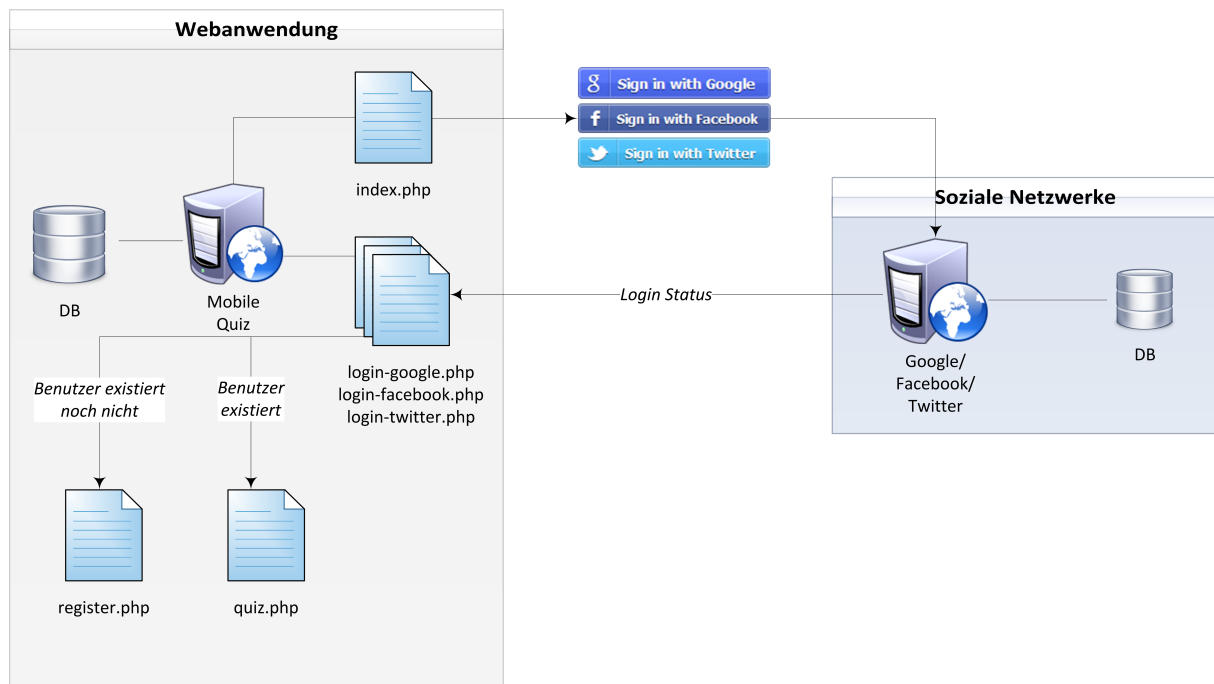


Abbildung 2.3.: Login via Google, Facebook oder Twitter

## 2.6. Datensicherheit

Die Datensicherheit beschreibt den Schutz durch die Anwendung, vor Missbrauch persönlicher Daten. Erlangt ein Unberechtigter Zugriff auf die Daten, könnten die Kennwörter der registrierten Benutzer ausgelesen werden. Mobile Quiz hält sich an die gängigen Standards für Datensicherheit. In der Datenbank werden die Passwörter verschlüsselt abgespeichert. Dies geschieht bei der Registrierung. Zuerst wird der Benutzername mit SHA1 gehashed. Der resultierende String wird daraufhin mit base64 codiert und ein Teil davon wird als Salt weiterverwendet. Schliesslich wird das Kennwort mit dem generierten Salt zusammengeführt, mit Blowfish[7] verschlüsselt und in der Datenbank abgelegt.



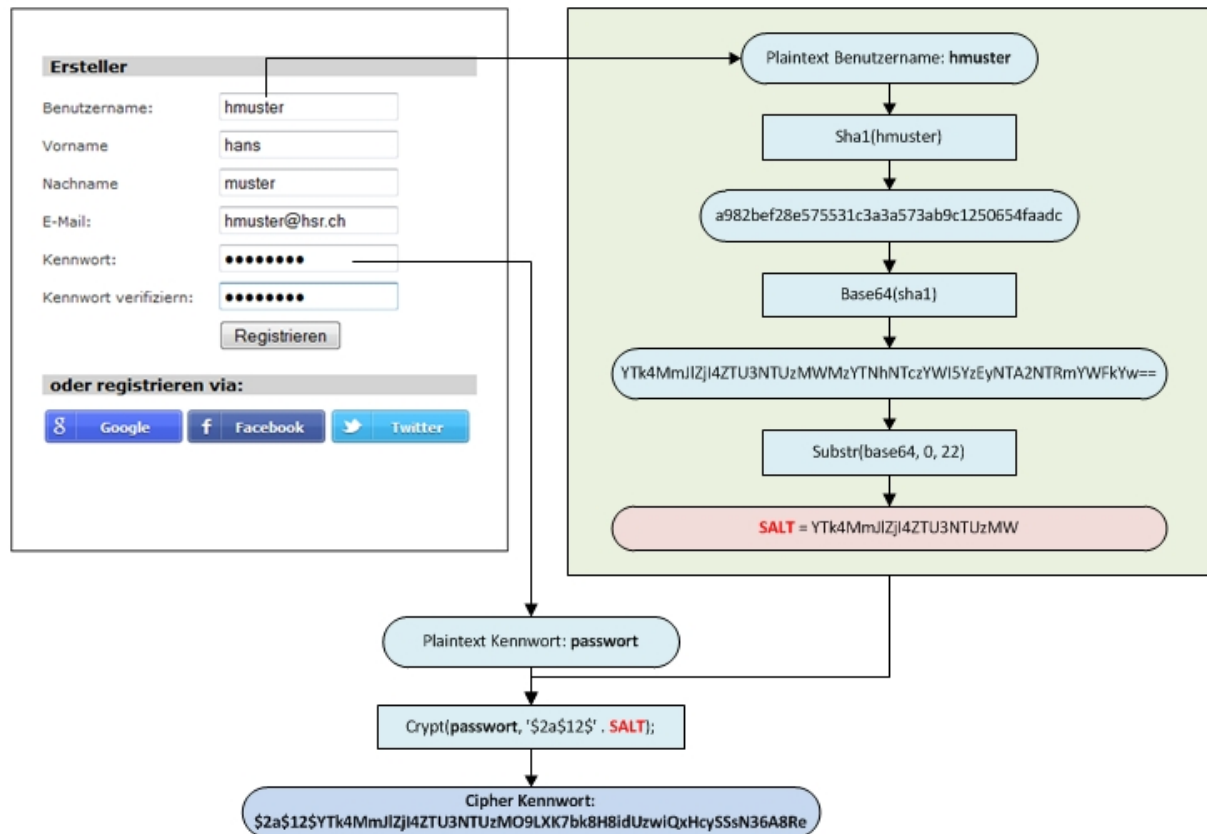


Abbildung 2.4.: Kennwortverschlüsselung bei Registrierung

Da die Anwendung zusätzlich zur regulären Registrierung die Möglichkeit bietet, sich über eines der gängigen sozialen Netzwerke anzumelden, wird bei dieser Art der Registrierung kein Kennwort gespeichert.

## 2.7. Bedrohungen und Verletzlichkeiten

### 2.7.1. Session Hijacking

Unter Session Hijacking wird der Missbrauch einer Session verstanden, die von einem Nutzer gegenüber der Webanwendung eröffnet wurde. Diese Attacke dient dazu, Daten eines Benutzers zu stehlen oder zu manipulieren. Hierbei gibt sich der Angreifer, als eine andere Person aus. Dabei gibt es verschiedene Möglichkeiten die Session-ID des Opfers zu erhalten. Eine davon ist das Abhören. Der Angreifer erfährt durch das Abhören der Verbindung die Session-ID, welche er dann selber benutzt um sich als Opfer auszugeben und damit Informationen zu erhalten oder diese zu manipulieren. Eine weitere Methode ist die Cross-Site-Scripting-Attacke. Unter Cross-Site-Scripting wird verstanden, dass der Angreifer über Sicherheitslücken im Code, einen JavaScript Code einschleust, welcher beim Opfer ausgeführt wird. So könnte zum Beispiel die Session-ID ausgelesen und dem Angreifer übermittelt werden. Zu den weiteren Methoden gehört auch die Man-In-The-Middle-Attacke, bei welcher der Angreifer sich in die originale Verbindung zwischen Server und Client stellt und als “Proxy” fungiert, der die Aufrufe vom Client zum Server und umgekehrt über seinen Rechner laufen lässt.

### 2.7.2. Session Fixation

Bei Session Fixation handelt es sich im Gegensatz zu Session Hijacking um einen Angriff, bei dem die Session vom Angreifer erstellt und dem Opfer untergeschoben wird. Wenn das Opfer sich mit dieser Session-ID angemeldet, erlangt der Angreifer ebenfalls Zugriff auf die Webseite und damit auf die Daten des Opfers. Zum einen gibt es die Möglichkeit, per URL-Manipulation die ID dem Opfer mitzugeben. Dabei wird an eine URL per GET-Parameter die Session-ID angehängt. Beispiel: `www.sample.com/login.php?session=kh2u12fds`. Um dies nicht so offensichtlich zu machen besteht die Möglichkeit, die URL mit `http://goo.gl` oder anderen Short-Link-Anbietern zu verschleiern. Andererseits ist es möglich dem Opfer via Cross-Site-Scripting eine Session-ID unterzuschieben. Dabei schleust der Angreifer ein JavaScript Code ein, welcher vom Opfer ausgeführt wird.

### 2.7.3. Schutz vor Attacken

Um sich vor solchen Attacken zu schützen, gibt es einige Methoden. Die einfachste, ist die Verschlüsselung der Kommunikation zwischen Server und Client. Falls dies nicht gewährleistet werden kann, können Clientseitig folgende Punkte berücksichtigt werden:

- Sich nach dem Besuch der Webseite immer abmelden. Vor allem bei der Verwendung öffentlicher Computer.
- Auf die Möglichkeit "Anmeldung merken" verzichten, da sonst ein Cookie auf dem Rechner abgelegt wird.

Serverseitig sind folgende Aspekte wichtig:

- Bei der Anmeldung eines Benutzers sollten keine veralteten Daten verwendet werden, die Informationen des Benutzers enthalten.
- Einem nicht angemeldeten Benutzer sollte keine Session zugewiesen werden.
- Sessions sollten mit einem Timeout versehen werden.
- Sessions sollten einen Fingerprint oder eine andere Kennung aufweisen die den Benutzer zusätzlich identifiziert. Falls die Session von einem anderen Computer aufgerufen wird, sollte eine neue Session eröffnet werden.
- Die Anwendung sollte eine Logout-Möglichkeit bieten.

In Mobile Quiz werden die beiden Attacken Session Fixation und Session Hijacking mit den oben genannten Punkten abgewehrt. Um exakter zu sein, wird einer Session ein Fingerprint hinzugefügt, der auf den Benutzer schliessen lässt. Der Fingerprint besteht aus einem vordefinierten Wort, einem zufälligen Salt Wert, dem User Agenten des Benutzers und einem Teil der IP-Adresse. Dieser String wird mit einem Hashing-Algorithmus verschleiert und schliesslich in der Session untergebracht. Weiter wird die Session-ID regelmässig erneuert, was Session Hijacking und Fixation erschwert.

# **Teil II.**

## **Requirements Engineering**

### 3. Analyse ähnlicher Anwendungen

Im Internet existiert ein breites Angebot an Anwendungen, welche das Erstellen von Lernkontrollen und Umfragen ermöglichen. Die Anwendungen unterscheiden sich im Umfang der Funktionen. Das Spektrum geht vom kleinen Angebot an Funktionen zum vermeintlichen Alleskönner, der wiederum kompliziert und schwerfällig ist. Mit anderen Worten, keiner der bisher bestehenden Anwendungen erfüllt die Anforderungen der Dozierenden der HSR.

Um die Anforderungsspezifikation erstellen zu können, wurde eine Reihe von Mitbewerbern analysiert. Unter anderem folgende:

- Quiz & Poll
- Rational Survey
- FluidSurveys
- SurveyGizmo
- ClassMarker
- QuizStar
- Moodle

Der Fokus der Evaluation lag auf der Funktionalität, die durch die einzelnen Mitbewerber angeboten werden. Resultat der Analyse ist eine Liste an Funktionen, die im Kapitel 5. Funktionsliste beschrieben ist.

## 4. Interviews

Um die Bedürfnisse der Dozierenden an der HSR ermitteln zu können, wurden Interviews mit mehreren Dozenten durchgeführt. Bei den interviewten Dozenten handelte es sich um:

- Prof. Stefan F. Keller
- Prof. Dr. Andreas Steffen
- Prof. Dr. Markus Stolze
- Prof. Dr. Peter Heinzmann
- Hans Noser

Zu Beginn der Interviews wurden die Personen über die Bachelorarbeit und ihre Ziele orientiert. Das Interview beleuchtete mehrere Schwerpunkte.

Einerseits war die Frage wie die Dozierenden im Moment Fragen an die Studierenden stellen. Diese Frage wurde von allen Dozierenden ähnlich beantwortet. Die Dozierenden stellen eine Frage in den Raum, wobei jeder die Möglichkeit hat, diese zu beantworten. Allerdings haben alle Dozierenden angemerkt, dass die Fragen jeweils nicht auf reges Interesse stossen. Da die Studierenden nur schwer zu motivieren sind, werden immer seltener Fragen gestellt.

Des Weiteren war die Frage, ob die Dozierenden bereits ein ähnliches Produkt kennen. Wurde die Frage bejaht, war die Folgefrage welche Eigenschaften sie daran mochten und welche nicht. Die Frage wurde von fast allen Dozierenden bejaht. Bei einer der Lösungen handelte es sich um hardwarebasierte Lösungen, die früher an der ETH eingesetzt wurde. Dabei waren die Tische mit Knöpfen versehen, die eine Teilnahme an einer Umfrage ermöglichten. Dieses System erlaubte eine anonyme Teilnahme, da die Sitzordnung der Studenten nicht vorgeschrieben war. Zudem konnte das System rasch in Betrieb genommen werden, weil die Infrastruktur bereits vorhanden war. Die Auswertung der Umfragen wurde in Echtzeit auf dem Präsentationsmedium angezeigt. Die Teilnahme an diesen Umfragen hatte jedoch nach Aussage von Prof. Dr. Andreas Steffen wenig, bis gar keinen Einfluss auf den Lernerfolg.

Im Rahmen der Interviews wurde auch die Anwendung Moodle erwähnt. Diese Plattform wird bereits im Master-Studiengang der HSR verwendet. Moodle wird hauptsächlich zur Ablage von Dokumenten verwendet. Die Teilnahme der Studierenden im Masterstudium ist bescheiden, da die Anwendung komplex ist.

In den Interviews wurde auch Quiz & Poll mehrfach erwähnt. Diese Anwendung wurde von den Dozierenden ebenfalls bereits ausprobiert. Die Anwendung wurde gelobt, da sie einfach aufgebaut ist. Die Fragen können online, anhand eines Spreadsheets erfasst

werden. Den Dozierenden fehlte die Möglichkeit, an einer Lernkontrolle mehr als zehn Fragen stellen zu können. Werden zu einer Lernkontrolle mehr als zehn Fragen erfasst, kann nicht kontrolliert werden, welche zehn Fragen von den Teilnehmenden beantwortet werden müssen. Von allen Dozenten wurde der nötige Google-Account bemängelt. Es kann von Studierenden nicht verlangt werden, ausschliesslich für eine Umfrage einen Account bei Google zu erstellen.

Der gemeinsame Wunsch der Dozierenden ist es, eine einfache und intuitive Anwendung zu erhalten. Eine Anwendung die im Unterricht schnell eingesetzt werden kann, damit nicht zu viel Zeit verloren geht. Einer der wichtigen Punkte ist die Forderung, dass die Anwendung plattformübergreifend verwendbar ist. Eine Teilnahme mit Laptop und Tablet ist gleichermassen wichtig, wie der Einsatz eines Smartphones. Dies bedingt eine Anwendung, die auf mobilen Geräten komfortabel zu bedienen ist.

## 5. Funktionsliste





## 6. Produktbeschreibung

Eine Anforderung an die Anwendung ist, dass diese einfach und intuitiv bedienbar ist und die gängigen Browser unterstützt. Konkret bedeutet dies, dass der Internet Explorer 9, Mozilla Firefox 12+ und Google Chrome 18+ unterstützt werden. Auf den mobilen Geräten werden die Standardbrowser unterstützt. Der Umfang der Funktionen umfasst das Nötigste in einfachster Form. Eine gelegentliche Bedienung soll gleichermassen einfach von der Hand gehen, wie eine alltägliche Anwendung. Es gibt die Rollen der Dozierenden und der Teilnehmenden. Dabei wird die Anwendung nicht nur auf die HSR zugeschnitten, sondern öffentlich im Internet zur Verfügung gestellt. Die Rolle des Dozierenden umfasst die Möglichkeit, Lernkontrollen und Umfragen zu erstellen, durchzuführen und die dazugehörigen Auswertungen zu evaluieren. Die Rolle der Teilnehmenden besteht ausschliesslich in der Teilnahme von Lernkontrollen/Umfragen und deren Einsicht. Sie erhalten per Mail einen Link zur Lernkontrolle/Umfrage und können dann daran teilnehmen. Im Unterricht erreichen die Teilnehmenden die Lernkontrolle/Umfrage direkt über einen QR-Code.

### 6.1. Dozierende

Um Zutritt zur Anwendung zu erhalten, ist ein Login notwendig. Bei der Registrierung werden Vor- und Nachname, E-Mail und Benutzername erfasst. Ein Login über soziale Netzwerke (Google, Facebook und Twitter) ist ebenfalls möglich.

#### 6.1.1. Erfassen

Der Prozess der Erfassung beginnt mit der Erstellung einer Lernkontrolle oder Umfrage. Dazu werden Name und Kürzel benötigt. Logo und Beschreibung sind optional. Daraufhin können Fragen und dazugehörige Antworten erfasst werden. Standardmässig werden vier Felder für die Erfassung der Antworten bei der Erstellung einer Frage angezeigt. Die Anzahl der Antworten ist auf sechs beschränkt. Soll eine geringere Anzahl an Antworten, als bereits im Formular enthalten sind, erfasst werden, kann dieses Feld leer gelassen werden. Bei der Erfassung einer Frage wird zugleich die maximale Zeit für die Beantwortung berechnet. Die Zeit zur Beantwortung setzt sich anhand der Anzahl enthaltener Zeichen zusammen und wird mit 300ms multipliziert. Diese Zeitangabe kann bei der Publikation einer Lernkontrolle, bzw. Umfrage deaktiviert werden. Die Erfassung der Lernkontrolle unterscheidet sich nur in einem Punkt von der Erstellung der Umfrage. Bei Umfragen kann eine Antwort nicht richtig oder falsch sein. Ansonsten ist der Ablauf identisch. Um die Erfassung zu erleichtern, können Fragen und Antworten importiert werden. Demnach ist es möglich, Fragen und Antworten in einem Tabellen-Kalkulationsprogramm aufzuarbeiten und diese mittels einer CSV-Datei (comma separated value), in dem die Werte durch ein Semikolon (;) getrennt sind, zu importieren. Die bereits erfassten Fragen und Antworten können in gleicher Form exportiert werden.

Fragen und Antworten von Lernkontrollen und Umfragen können nachbearbeitet werden.

Diese Funktion steht solange zur Verfügung, bis die erste Durchführung stattgefunden hat. Sobald eine Durchführung läuft, ist das Editieren und Importieren von Fragen und Antworten nicht mehr möglich. Diese Funktion bleibt selbst nach Ablauf einer Durchführung verwehrt. Dies aus dem Grund, da sonst die Auswertungen zu bestehenden Durchführungen an Aussagekraft verlieren. Soll eine Lernkontrolle oder Umfrage bearbeitet werden, muss diese dupliziert werden. Die duplizierte Lernkontrolle oder Umfrage kann bearbeitet und ergänzt werden. Die Möglichkeit des Exports besteht zu jeder Zeit.

### 6.1.2. Vorschau

Bevor eine Durchführung gestartet wird, kann die Lernkontrolle, wie auch die Umfrage in der Vorschau überprüft werden. So können weiterhin Änderungen an einer Lernkontrolle oder Umfrage vorgenommen werden. Das Editieren ist nach Start einer Durchführung nicht mehr möglich. Werte die in der Vorschau eingegeben werden, fließen nicht in die Auswertung mit ein.

### 6.1.3. Publikation

Eine Publikation wird für eine fixe Dauer definiert. Dafür wird ein Start- und Enddatum festgelegt. Das Startdatum darf sich in der Zukunft befinden. Zum Start- und Enddatum wird zusätzlich eine Zeit bestimmt. Um die Eingabe der Daten zu erleichtern, wird ein Kalender bereitgestellt. Standardmässig wird eine Dauer von sieben Tagen vom Erstellungsdatum vorgeschlagen.

Bei der Publikation kann optional ein Zeitlimit festgelegt werden. Einerseits pro Frage oder pro Quiz. Bei der Erstellung von Fragen wird eine Zeit berechnet. Diese ergibt sich aus der Anzahl Zeichen der Fragen und Antworten. Die Zeit für die Lernkontrolle wird vom Dozierenden festgelegt. Es kann nur eine der Einstellungen gewählt werden. Zeit pro Frage: Nach Ablauf der Zeit werden die Teilnehmenden zur nächste Frage weitergeleitet. Zeit pro Quiz: Die Durchführung wird beendet.

Ein weiterer Punkt bei der Publikation sind die Teilnahmemodi. Bei den Teilnahmemodi kann zwischen authentisiert, pseudonym und anonym gewählt werden. Diese drei unterschiedlichen Modi sind für die Auswertung von Bedeutung. Bei der authentisierten Variante werden die Teilnehmenden in der Auswertung namentlich erwähnt - es sind Rückschlüsse auf die einzelnen Personen möglich. Im pseudonymen Modus erscheint in der Auswertung lediglich das Pseudonym. Die anonyme Variante zeigt in der Auswertung den Alias an.

Um Lernkontrollen und Umfragen interessanter zu gestalten, können die Fragen und Antworten in eine zufällige Reihenfolge gebracht werden. Diese zwei Optionen sind unabhängig voneinander verfügbar.

Um den Lernerfolg zu optimieren, stehen Optionen für die Einblendung der Lösungen zur Verfügung. Entweder wird die Lösung bei einer Lernkontrolle nach jeder einzelnen Frage eingeblendet. Bei der Umfrage steht diese Option nicht zur Verfügung. Oder die gesamte Lösung mit der Korrektur kann nach Beendigung der Lernkontrolle angezeigt werden. Bei Umfragen handelt es sich bei dieser Option um die Einsicht der aktuellen

Abstimmungsergebnisse.

Die Publikation kann mit einem Kommentar versehen werden, um die Unterscheidung von parallelen Durchführungen zu ermöglichen. Jede Durchführung wird mit Hilfe eines zufälligen Tokens geschützt. Dozierende versenden Einladungen für eine Lernkontrolle oder Umfrage per E-Mail. Auf diesem Weg nehmen nur Teilnehmende einer ausgewählten Zielgruppe an einer Durchführung teil.

Sobald mindestens eine Durchführung läuft, wird diese zuoberst auf der Editierseite aufgeführt. Somit haben Erstellende eine Übersicht über die aktuell laufenden Durchführungen.

### 6.1.4. Navigation

Die Suche nach Lernkontrollen oder Umfragen beinhaltet die Felder Name, Beschreibung und Kürzel. Die Suche zeigt alle Einträge an, die den Suchtext enthalten.

Die Navigation ist unterteilt in Lernkontrollen und Umfragen. Innerhalb der Kategorie sind die einzelnen Einträge nach Ablaufdatum der letzten Durchführungen sortiert. Einträge ohne Durchführungen werden darunter aufgeführt. Das Setzen eines Filters ist durch Eingabe eines Kürzels, Namens oder einer Beschreibung möglich. Sobald ein Filter gesetzt wird, erscheint ein Knopf um den Filter zu entfernen. Jeder Eintrag wird mit einem Symbol, Namen (die Beschreibung erscheint, wenn mit dem Mauszeiger über den Namen gefahren wird), Anzahl der Fragen und Anzahl der Durchführungen beschrieben. Das Symbol zum jeweiligen Eintrag beschreibt folgendes:



**Abbildung 6.1.:** Symbole für den Status der Einträge.

- **Weisser Kreis mit schwarzem Rand** - Es hat bis jetzt noch keine Durchführung stattgefunden.
- **Grüner Kreis** - Es findet zur Zeit eine Durchführung statt.
- **Roter Kreis** - Die Durchführung ist bereits abgeschlossen.

Zudem stehen zu jedem Eintrag folgende Funktionen zur Verfügung:

#### Lernkontrollen

##### Quiz

Bei dieser Funktion ist die Lernkontrolle an sich einsehbar, um sie den Studierenden zur Verfügung zu stellen, bzw. um die Lernkontrolle in schriftlicher Form zu verwenden.

##### Lösung

Diese Funktion ermöglicht die Einsicht der Lösungen einer Lernkontrolle, um sie

Studierenden zur Verfügung zu stellen, bzw. um die Lösungen in schriftlicher Form zu verwenden.

### editieren

Diese Funktion ermöglicht die Bearbeitung einer Lernkontrolle. Es können zusätzliche Fragen und Antworten erfasst, oder gelöscht werden. Die Fragen und Antworten können importiert und exportiert werden. Hier werden die Durchführungen geplant.

### kopieren

Gesamte Lernkontrolle wird dupliziert, inklusive der Fragen und Antworten und ohne Berücksichtigung der Durchführungen.

### auswerten

Damit wird die Auswertung zur Lernkontrolle geöffnet. Die Seite wird mit allen vorhandenen Durchführungen angezeigt. Die Durchführungen werden durch Rechtecke repräsentiert, die von einem gestrichelten Rahmen umgeben sind. Es werden alle Durchführungen aufgeführt:



**Abbildung 6.2.:** Symbole für den Status der Durchführungen

- Durchführung die erst noch beginnt: Das Rechteck ist hellgrau mit einem dunkelgrauen gestrichelten Rahmen. Sie ist nicht aufrufbar.
- Durchführung die läuft, jedoch noch keine Teilnehmenden hat: Das Rechteck ist hellgrau mit einem dunkelgrauen gestrichelten Rahmen. Sie ist nicht aufrufbar.
- Durchführung die läuft und bereits Teilnehmende gefunden hat: Das Rechteck ist hellgrün mit einem dunkelgrünen gestrichelten Rahmen. Sie kann aufgerufen werden.
- Durchführung die abgeschlossen ist, jedoch keine Teilnehmenden hatte: Das Rechteck ist weiss mit einem dunkelgrauen gestrichelten Rahmen. Sie kann nicht aufgerufen werden.
- Durchführung die abgeschlossen ist und Teilnehmende fand: Das Rechteck ist hellblau mit einem dunkelblauen Rahmen. Sie kann aufgerufen werden.

Die Auswertung einer Lernkontrolle umfasst eine Grobübersicht, Detailansicht und Rangliste. Die Grobübersicht enthält die besten fünf Teilnehmenden. Der Inhalt der ersten Spalte ist abhängig vom Modus, in dem die Lernkontrolle ausgeführt wurde.

- Authentisiert - Name, Vorname
- Pseudonym - Pseudonym
- Anonym - Alias

Die restlichen Spalten sind bei allen Modi identisch. Die erreichte Punktzahl, benötigte Zeit für die Lernkontrolle und wie viele Punkte pro Zeiteinheit gemacht wurden.

Dazu wird die am Besten und am Schlechtesten beantwortete Frage angezeigt.

Zudem werden die Anzahl Teilnahmen angezeigt, gruppiert in geschlossene und offene Sitzungen. In die Auswertung fließen nur die Teilnehmenden, die die Lernkontrolle abgeschlossen haben. Diese Zahlen werden zusätzlich in einem Kuchendiagramm repräsentiert.

Des Weiteren wird eine kleine Zeitauswertung angeboten. Diese Auswertung enthält folgende Punkte:

- Kürzeste Sitzung
- Längste Sitzung
- Zentralwert der Punkte aller Teilnehmenden
- Durchschnittliche Dauer der Sitzungen

Ausserdem enthält die Grobübersicht eine Auswertung der Teilnahmen pro Tag. Diese Werte werden in einem Liniendiagramm dargestellt.

Die Detailansicht zeigt die gesamte Lernkontrolle auf. Dazu werden alle Fragen und ihre dazugehörigen Antworten aufgelistet. Bei den einzelnen Antworten ist sichtbar, ob diese Antwort richtig oder falsch ist.

Der Frage entsprechend wird die Durchschnittspunktzahl der Teilnehmenden angegeben. Zu jeder Antwort wird angegeben, wie viele Teilnehmende diese Antwort richtig, falsch oder neutral beantwortet haben. Diese Werte werden in einem kleinen Balkendiagramm dargestellt. Der Balken der korrekt beantworteten Fragen ist grün, der Falschen rot und der Neutralen hellgrau.

Die Rangliste listet alle Teilnehmenden absteigend sortiert nach der erreichten Punktzahl auf. Der Rang wird in der ersten Spalte angezeigt. Die zweite Spalte ist abhängig vom Modus in der die Lernkontrolle ausgeführt wurde.

- Authentisiert - Name, Vorname
- Pseudonym - Pseudonym
- Anonym - Alias

Die restlichen Spalten sind bei allen Modi identisch. Die erreichte Punktzahl, benötigte Zeit für die Lernkontrolle und wie viele Punkte pro Zeiteinheit gemacht wurden.

In der zweiten Spalte wird der Name, das Pseudonym oder der Alias als Link dargestellt, welcher zur Detailansicht des Teilnehmenden führt. In der Detailansicht des Teilnehmenden wird die gesamte Lernkontrolle mit allen Fragen und ihren dazugehörigen Antworten angezeigt. Bei den Antworten wird auf der linken Seite angegeben, ob diese korrekt sind oder nicht. Auf der rechten Seite werden die Angaben des Teilnehmenden angezeigt. Dabei wird angegeben was für ein Tipp, richtig, falsch oder neutral abgegeben wurde. War der Tipp richtig, wird die Linie grün eingefärbt. War der Tipp falsch, wird die Linie rot eingefärbt. Für einen neutralen Tipp bleibt die Linie weiss. Die visuelle Unterstützung kann ein- und ausgeschaltet werden.

### **löschen**

Lernkontrollen werden mit dieser Funktion archiviert. Nach Ausführung dieser Funktion sind die jeweiligen Lernkontrollen über die Benutzeroberfläche nicht mehr erreichbar. So wird sicher gestellt, dass aus Versehen gelöschte Daten wiederhergestellt werden können.

### **Umfrage**

#### **Umfrage**

In dieser Funktion ist die Umfrage einsehbar, um sie Studierenden zur Verfügung zu stellen, bzw. um die Lernkontrolle in schriftlicher Form zu verwenden.

#### **Jetzt starten!**

Hiermit wird der Präsentationsmodus gestartet, welcher die Umfrage im Vollbild auf dem Präsentationsmedium anzeigt. Die Startseite zeigt Link und QR Code zur Umfrage an. Die Dozierenden führen die Teilnehmenden durch die Umfrage. Auf dem Präsentationsmedium erscheinen jeweils die aktuellen Resultate zur Frage. Die Werte werden alle fünf Sekunden aktualisiert. Die Teilnehmenden sehen auf ihrem Gerät jeweils nur die aktuelle Frage. Nach Abgabe der Stimme überprüft das Gerät alle fünf Sekunden, ob eine neue Frage ansteht. Ist dies der Fall, springt die Anwendung zur nächsten Frage. Der Präsentationsmodus kann jederzeit verlassen werden, dabei wird die Durchführung geschlossen und eine Wiederaufnahme ist ausgeschlossen.

### **editieren**

Innerhalb dieser Funktion wird eine Umfrage bearbeitet. Es können zusätzliche Fragen und Antworten erfasst, oder gelöscht werden. Im Zusammenhang mit den Fragen und Antworten steht die Import- und Exportfunktion zur Verfügung. Hier werden auch die Durchführungen geplant.

### **kopieren**

Gesamte Umfrage wird dupliziert, inklusive der Fragen und Antworten, ohne Berücksichtigung der Durchführungen.

### **auswerten**

Das Resultat der Umfrage wird angezeigt. Dabei wird mit kleinen Balken die Anzahl der Stimmen der jeweiligen Antwort dargestellt. Die Auswertung der Antworten einzelner Teilnehmer ist möglich.

### **löschen**

Umfragen werden mit dieser Funktion archiviert. Nach Ausführen dieser Funktion sind die jeweiligen Umfragen über die Benutzeroberfläche nicht mehr erreichbar. Durch die Archivierung wird sicher gestellt, dass aus Versehen gelöschte Daten wiederhergestellt werden können.

## **6.2. Teilnehmende**

### **6.2.1. Teilnahme**

Die Teilnehmenden folgen dem ihnen zugestellten Link, oder bei Umfragen dem QR-Code. Die Teilnahme ist mit dem Gerät ihrer Wahl möglich. Für jede Art der Durchführung,

authentisiert, pseudonym oder anonym, ist ein Anmelden nötig. Die verschiedenen Arten der Durchführung unterscheiden sich für die Teilnehmenden lediglich seitens des Anmeldeformulars. Im Einstiegsfenster einer Durchführung erscheinen Name und die Beschreibung einer Lernkontrolle oder Umfrage, gefolgt von einem Logo. Dazu wird angegeben, in welchem Modi die Durchführung stattfindet:

- Für die authentifizierte Durchführung ist Benutzername und Kennwort erforderlich. Bei der Registrierung werden Vor- und Nachname, E-Mail, Benutzername und Kennwort erfasst. Ein Login über soziale Netzwerke (Google, Facebook und Twitter) ist ebenfalls möglich.
- Für die pseudonyme Durchführung ist Pseudonym und Kennwort erforderlich. Bei der Registrierung werden Pseudonym, E-Mail und Kennwort erfasst.
- Für die anonyme Durchführung ist Alias und Kennwort erforderlich. Bei der Registrierung wird nur das Alias und Kennwort erfasst.

Für die Teilnahme an unterschiedlichen Arten einer Durchführung ist jeweils ein erneuter Login nötig. Nur so kann die Wahrung der Identität gewährleistet werden. Eine Teilnahme an einer Lernkontrolle oder Umfrage ist nur innerhalb der vom Dozierenden definierten Zeitspanne möglich. Sonst erscheint dem Teilnehmenden eine Meldung, dass die Durchführung bereits vorüber ist.

Die Ergebnisse anonymer Teilnehmer an Lernkontrollen und Umfragen werden im Verlauf einer gewissen Zeitspanne ebenfalls verfolgt um die Lernerfolge festzuhalten. In diesem Fall ist es wichtig, dass ein Alias mit einem Kennwort versehen werden kann. Ansonsten ist ein Missbrauch nicht ausgeschlossen. Trotz der Registrierung kann kein Mapping auf die reale Person dahinter gemacht werden, da lediglich das Alias und Kennwort erhoben werden.

### 6.2.2. Einsicht

Die Navigation unterteilt Lernkontrollen und Umfragen. Innerhalb der Kategorie sind die einzelnen Einträge nach Ablaufdatum der Durchführungen geordnet. Die Einträge werden erst angezeigt, wenn die Durchführung, an der teilgenommen wurde, abgelaufen ist. Das Setzen eines Filters nach Name oder Beschreibung ist möglich. Sobald ein Filter gesetzt wird, erscheint ein Knopf, um den Filter zu entfernen. Jeder Eintrag wird mit einem Namen und der Beschreibung der Lernkontrolle oder Umfrage aufgelistet. Zudem stehen zu jedem Eintrag folgende Funktionen zur Verfügung:

#### **Lernkontrollen**

##### **Quiz**

Bei dieser Funktion ist die Lernkontrolle an sich einsehbar.

##### **Lösung**

Diese Funktion ermöglicht die Einsicht der Lösung einer Lernkontrolle.

### Umfrage

#### Resultate

Diese Funktion ermöglicht das Einsehen der Umfrageresultate.

### 6.2.3. Use Cases

#### Use Case UC01: Lernkontrolle/Umfrage erstellen

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle oder Umfrage. Dort erstellen sie eine neue Lernkontrolle/Umfrage mit Namen, Kürzel, Beschreibung und Logo.

#### Use Case UC02: Lernkontrolle/Umfrage löschen

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle oder Umfrage. In der Liste wählen sie eine Lernkontrolle/Umfrage aus und führen die Lösch-Operation aus.

#### Use Case UC03: Lernkontrolle/Umfrage bearbeiten

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle oder Umfrage. In der Liste wählen sie die Lernkontrolle/Umfrage aus die sie editieren wollen und klicken auf editieren. Der Link führt sie zu einer neuen Seite, auf der sie die Lernkontrolle/Umfrage editieren können. Nachdem die Änderungen getätigt wurden, führen sie die Speichern-Operation aus.

#### Use Case UC04: Benutzerkonto erstellen

Dozierende oder Teilnehmende kommen auf die Webseite von Mobile Quiz und tragen unter dem Punkt registrieren ihre Personalien ein. Die Registrierung schliessen sie mit der Registrier-Operation ab.

#### Use Case UC05: Benutzerkonto löschen

Dozierende oder Teilnehmende melden sich am System als Ersteller oder Teilnehmer an. In der Anwendung navigieren sie zu ihrer Profilseite und geben ihr Kennwort ein, um das Benutzerkonto endgültig zu löschen. Dies bestätigen sie mit der Lösch-Operation.

#### Use Case UC06: An Lernkontrolle/Umfrage teilnehmen

Teilnehmende erhalten per E-Mail einen Link mit der Aufforderung an einer Lernkontrolle/Umfrage teilzunehmen. Diesen öffnen sie auf ihrem mobilen Gerät und melden sich am System an. Danach füllen sie die Lernkontrolle/Umfrage aus.

#### Use Case UC07: Lernkontrolle/Umfrage exportieren

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle/Umfrage. In der Liste wählen sie die Lernkontrolle/Umfrage aus, die sie exportieren wollen und klicken auf editieren. Dort angelangt, führen sie die Export-Operation aus und speichern die erstellte CSV-Datei auf ihrem Computer ab.



### **Use Case UC08: Lernkontrolle/Umfrage importieren**

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle/Umfrage. Sie erstellen eine neue Lernkontrolle/Umfrage und führen die Importierfunktion aus. Im Explorer/Finder wählen sie die zu importierende CSV-Datei aus.

### **Use Case UC09: Auswertung einsehen**

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle/Umfrage. In der Liste wählen sie die Lernkontrolle/Umfrage aus, die sie auswerten wollen und klicken auf auswerten. Diese leitet sie auf die Auswerteseite der Lernkontrolle/Umfrage.

### **Use Case UC10: Umfrage leiten**

Dozierende melden sich am System als Ersteller an und navigieren zur Seite der Lernkontrolle/Umfrage. In der Liste wählen sie die Lernkontrolle/Umfrage aus, die sie im Präsentationsmodus starten wollen und klicken auf Jetzt starten!. Diese Operation startet den Präsentationsmodus und der Dozierende kann die Teilnehmenden durch die Umfrage führen.

# **Teil III.**

## **Software Engineering**

## 7. User Interface

### 7.1. Übersicht

Das User Interface ist in zwei Bereiche unterteilt. Der eine Bereich beinhaltet die Administration, welche für die Verwendung mit einem Computer optimiert ist. Der andere Bereich hingegen ist für die Verwendung mit einem mobilen Gerät optimiert.

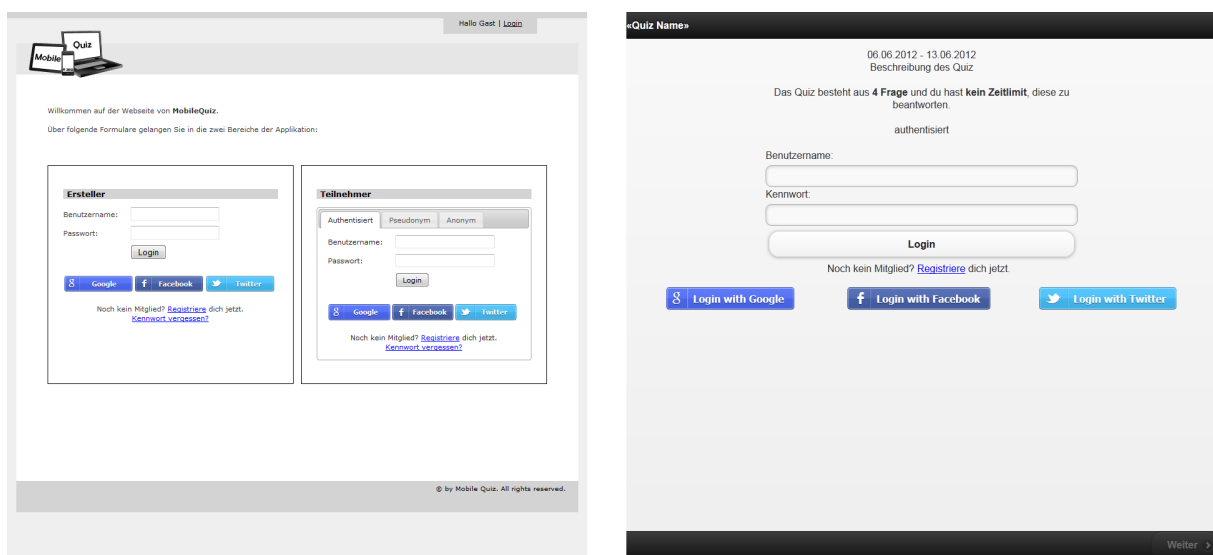


Abbildung 7.1.: Einstiegspunkte Administration (links), Teilnahme (rechts) der Webanwendung

#### 7.1.1. Administration

Die Administration ist ebenfalls in zwei Bereiche gegliedert. Es ist möglich, sich als Ersteller oder Teilnehmer im System anzumelden. Dies führt jeweils zu unterschiedlichen Ansichten.

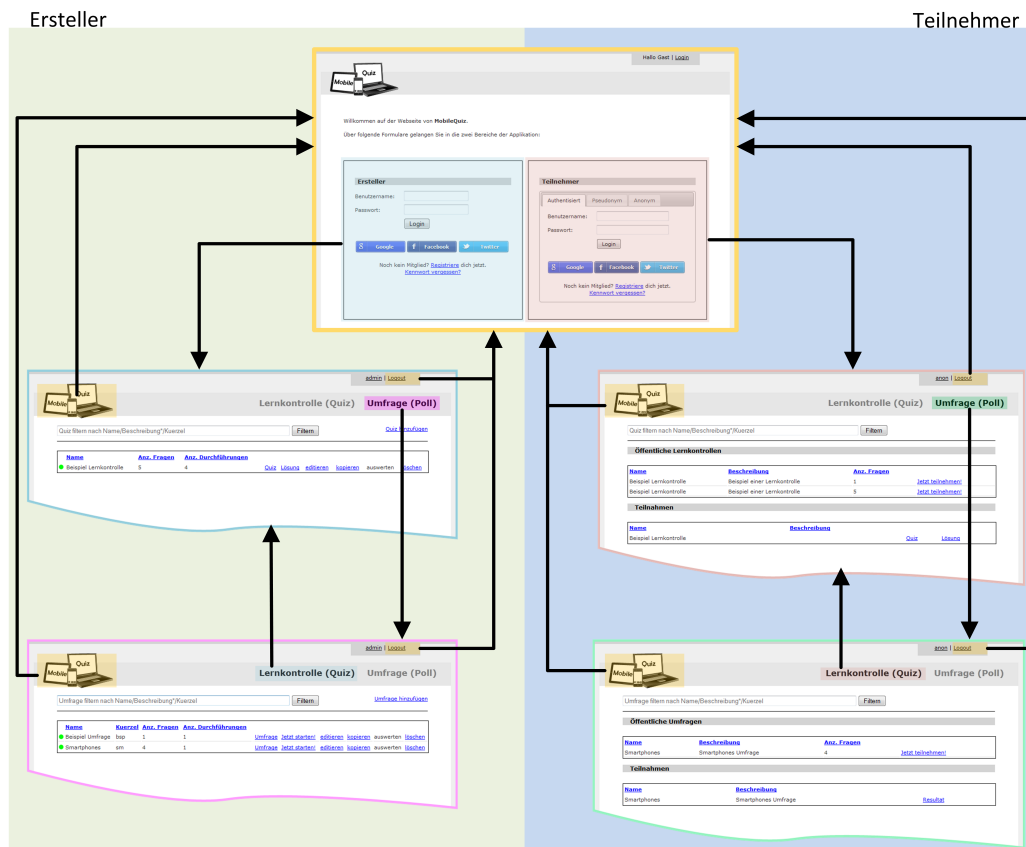


Abbildung 7.2.: Das Storyboard der User Interface

## Ersteller

Der Ersteller hat die Möglichkeit, Lernkontrollen/Umfragen zu erstellen, zu bearbeiten, auszuwerten, zu kopieren und zu löschen. Zudem kann die Lernkontrolle, wie auch dessen Lösung, eingesehen werden. Bei Umfragen ermöglicht das System den Erstellern die Resultate einzusehen.

## Teilnehmer

Je nach Teilnahmemodus (anonym, pseudonym oder authentisiert), werden auf der Seite des Teilnehmenden die Lernkontrollen/Umfragen angezeigt. Es werden alle Teilnahmen angezeigt, bei denen die Durchführung bereits abgeschlossen ist. Der Teilnehmende hat die Möglichkeit, die Lernkontrolle mitsamt ihren Lösungen einzusehen. Bei Umfragen können die Resultate betrachtet werden. Die Einträge können mit Hilfe der Suchmaske gefiltert werden.

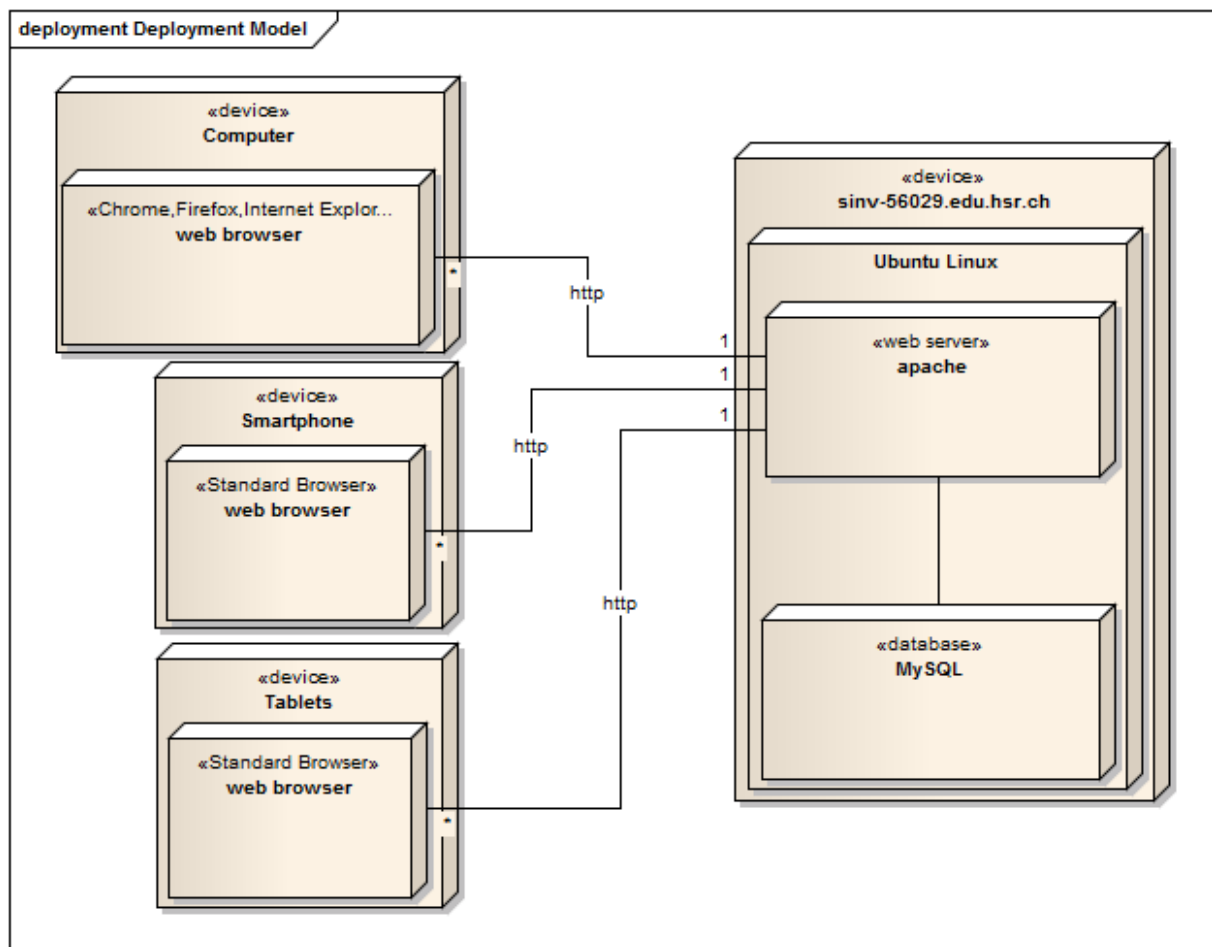
### 7.1.2. Teilnahme

Die Teilnahme-Seite ist der Einstiegspunkt für Studierende, die eine URL erhalten oder den QR-Code mit ihrem Smartphone abfotografiert haben. Das User Interface wurde mit jQuery Mobile realisiert und setzt sich aus den typischen Elementen von jQuery zusammen. Die Eingabemethode besteht bei der Lernkontrolle aus einem Toggle, der drei Status repräsentiert: falsch, richtig und neutral. Bei Umfragen wird hingegen auf den konventionellen Radiobutton gesetzt.

## 8. Architektonische Darstellung

### 8.1. Deployment Diagramm

Das Deployment Diagramm der Anwendung sieht wie folgt aus:



**Abbildung 8.1.:** Das Deployment Diagramm mit den unterschiedlichen Geräten

Die Webanwendung läuft auf einem Apache Webserver, der auf einem Ubuntu Server installiert ist. Dabei wird mit den verschiedenen Eingabegeräten via Hypertext Transfer Protokoll auf die Anwendung zugegriffen. Bei den Geräten werden jeweils die gängigen Standardbrowser unterstützt. Auf den Computern oder Laptops werden die drei meistbenutzten<sup>[27]</sup> Browser, namentlich Chrome 18+, Firefox 12+ und Internet Explorer 9, unterstützt.

## 8.2. Domainmodell

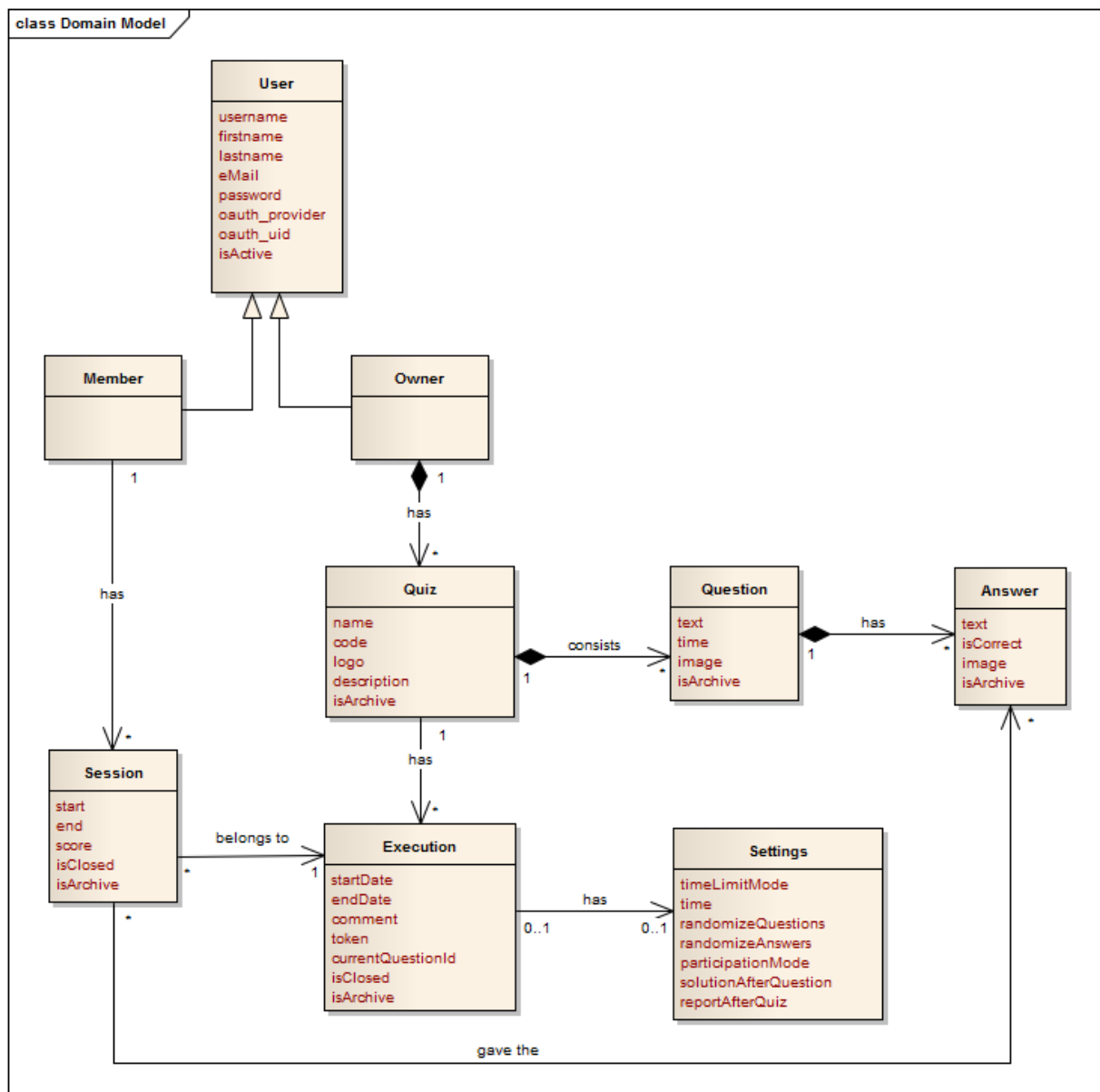


Abbildung 8.2.: Das Domainmodell der Problemdomäne

In diesem Abschnitt werden lediglich die Objekte erläutert, die nicht selbsterklärend sind.

### 8.2.1. User

Das Objekt User beschreibt den Benutzer mit seinen Attributen. Dabei unterscheidet sich ein Benutzer, der sich bei der Anwendung registriert, grundsätzlich nicht von einem, der sich über ein soziales Netzwerk registriert. Der Unterschied liegt in zwei Attributen, die gesetzt werden, wenn sich jemand über ein soziales Netzwerk registriert. Das Attribut `oauth_provider` beinhaltet den Namen des sozialen Netzwerkes. Im Attribut `oauth_uid` wird die ID des Benutzers vom sozialen Netzwerk festgehalten. Der Benutzer kann dabei

die Rolle Member (Teilnehmer) oder Owner (Ersteller) einnehmen. Dies wird anhand der Benutzerinformationen und dem Einstiegspunkt in die Anwendung geregelt.

- Anonym: Der Benutzer enthält nur den Benutzernamen und das Kennwort.
- Pseudonym: Enthält Benutzername, E-Mail und Kennwort.
- Authentisiert: Enthält alle Felder.

### 8.2.2. Execution

Ein Execution-Objekt beschreibt eine Durchführung. Diese beinhaltet ein Start-, sowie ein Enddatum. Die `currentQuestionId` ist für Umfragen, die über „Jetzt starten!“ gestartet werden und um die momentan freigegebene Frage anzuzeigen. Die Teilnehmenden greifen jeweils im Fünf-Sekunden-Takt auf diese Information zu und zeigen die momentan aktive Frage an.

### 8.2.3. Settings

Das Objekt Settings beschreibt die einzelnen Einstellungen, die eine Durchführung hat. Unter den Einstellungsmöglichkeiten stehen folgende zur Auswahl:

**timeLimitMode:** Kein Zeitlimit, Zeitlimit pro Frage, Zeitlimit auf die Lernkontrolle/Umfrage.

**time:** Wird nur verwendet, wenn der `timeLimitMode` auf die gesamte Lernkontrolle/Umfrage gesetzt ist.

**randomizeQuestions:** Fragen werden in zufälliger Reihenfolge angezeigt.

**randomizeAnswers:** Antworten werden in zufälliger Reihenfolge angezeigt.

**participationMode:** Regelt die Teilnahmemodi (Anonym, Pseudonym, Authentisiert).

**solutionAfterQuestion:** Möglichkeit nach jeder Frage, die Lösung anzuzeigen oder nicht.

**reportAfterQuiz:** Nach der Lernkontrolle wird eine Zusammenfassung angezeigt.

## 8.3. Klassendiagramm

Das Klassendiagramm zeigt die Model-Klassen zusammen mit den Controllern und wie sie zueinander stehen.





lichkeit, Daten aus der Datenbank zu laden. Der ServiceLocator fungiert als eine Art Schnittstelle, die angefragt wird um Services, in diesem Fall Repositories, zu erhalten. So kann zu Testing-Zwecken beispielsweise ein Fake-Repository mitgegeben werden, um die Tests mit der Datenhaltung zu simulieren.

### Aufbau

Mit Hilfe des ServiceLocators werden die Instanzen der Repositories, der Datenbankverbindung und des EntityManager verwaltet.

Benötigt eine Klasse einen EntityManager, eine Datenbankverbindung oder ein Repository, fragt diese beim ServiceLocator nach einer Instanz. Der ServiceLocator prüft ob bereits eine Instanz vorhanden ist. Wenn noch keine Instanz besteht, wird das Objekt erzeugt und der anfragenden Klasse übergeben. Ist bereits eine Instanz vorhanden, so wird diese der anfragenden Klasse übergeben.

Mit dieser Methode wird ein Faken der Objekte möglich. So können Testing-Klassen ein Mock-Objekt für den EntityManager, die Datenbankverbindung oder die Repositories im ServiceLocator platzieren. Wenn danach die einzelnen Unit Tests auf den ServiceLocator zugreifen, erhalten diese das Mock-Objekt für die Tests.

### Faking und Mocking

Listing 8.1: Bevor die Unit Tests durchlaufen, setzt die Testing-Klasse ein Mock-Objekt der zu testenden Klasse im ServiceLocator ab.

```
1  /**
2   * @static
3   * @param Doctrine\Orm\EntityRepository $repository
4   */
5  public static function setQuizRepository($repository) {
6      self::$quizRepository = $repository;
7  }
```

Listing 8.2: Die Testing-Klasse in PHPUnit erzeugt ein Mock-Objekt einer Klasse. Mit setMethods() wird definiert, welche Methoden auf dem Mock-Objekt aufgerufen werden können. Anschliessend wird das Mock-Objekt im ServiceLocator gesetzt.

```
1 $this->quizRepositoryMock = $this->getMockBuilder('Repositories\
   QuizRepository')
2     ->disableOriginalConstructor()
3     ->setMethods(array('exist', 'findById', '
   findByIdWithoutUser', 'findByName'))
4     ->getMock();
5
6     ServiceLocator::setQuizRepository($this->quizRepositoryMock);
```

Listing 8.3: Benötigt ein Unit Test ein Quiz-Objekt, holt er dieses über den ServiceLocator. Da vorhin ein Mock-Objekt der Quiz-Klasse gesetzt wurde, wird dieses Mock-Objekt zurückgegeben. Damit können Tests durchgeführt werden, ohne die reale Datenbank zu verwenden.

```
1  /**
```

```
2      * @static
3      * @return Repositories\QuizRepository
4      */
5      public static function getQuizRepository() {
6          if (self::$quizRepository === null) {
7              self::$quizRepository = self::getEm()->getRepository('Model\
              Quiz');
8          }
9
10         return self::$quizRepository;
11     }
```

Listing 8.4: Damit die Unit Tests durchgeführt werden können, muss zuerst das Mock-Objekt gesetzt werden. Dazu wird angegeben, welcher Methodenaufruf welchen Wert zurückliefert. Die Mock-Objekte werden von PHPUnit verwaltet.

```
1      $quizRepositoryMock->expects($this->any())
2          ->method('findById')
3          ->will($this->returnValue(Quiz));
```

### Problematik

Bei der Verwendung eines ServiceLocator ist nicht bekannt, welche Abhängigkeiten eine Klasse hat. Es können kreuz und quer Services aufgerufen werden. Dies erschwert ein Debuggen.

Dieser Punkt ist für Mobile Quiz nicht von Bedeutung, da nur die Datenbankangelegenheiten mit Hilfe des ServiceLocators verwaltet werden.

## 8.4. Entity-Relationship-Diagramm

Das Entity-Relationship-Diagramm zeigt die Datenhaltung der Anwendung. Mit Doctrine können Objekte in der Datenbank abgebildet und wieder ausgelesen werden.

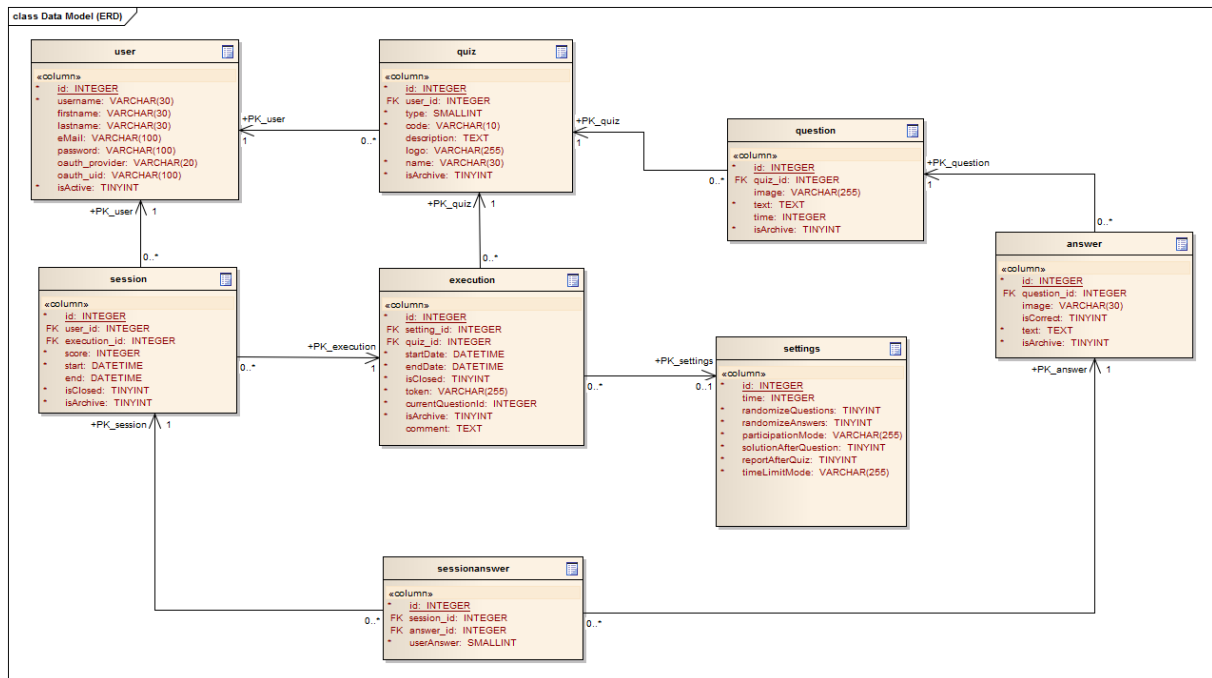


Abbildung 8.4.: Das Entity-Relationship-Diagramm der Anwendung

Im Entity Relationship Diagramm ist ersichtlich, dass der Benutzer in einer Tabelle abgebildet wird. Dies daher, da ein Erstellender ebenfalls ein Teilnehmender sein kann. Einerseits kann der Typ des Benutzers anhand der vorhandenen Attributen ermittelt werden, andererseits stehen über das User Interface zwei Einstiegspunkte zur Verfügung. Somit wird eine saubere Trennung von Erstellenden und Teilnehmenden ermöglicht.

## 8.5. Logische Architektur

### 8.5.1. Layer-Diagramm

Das Layer-Diagramm der Anwendung sieht wie folgt aus:

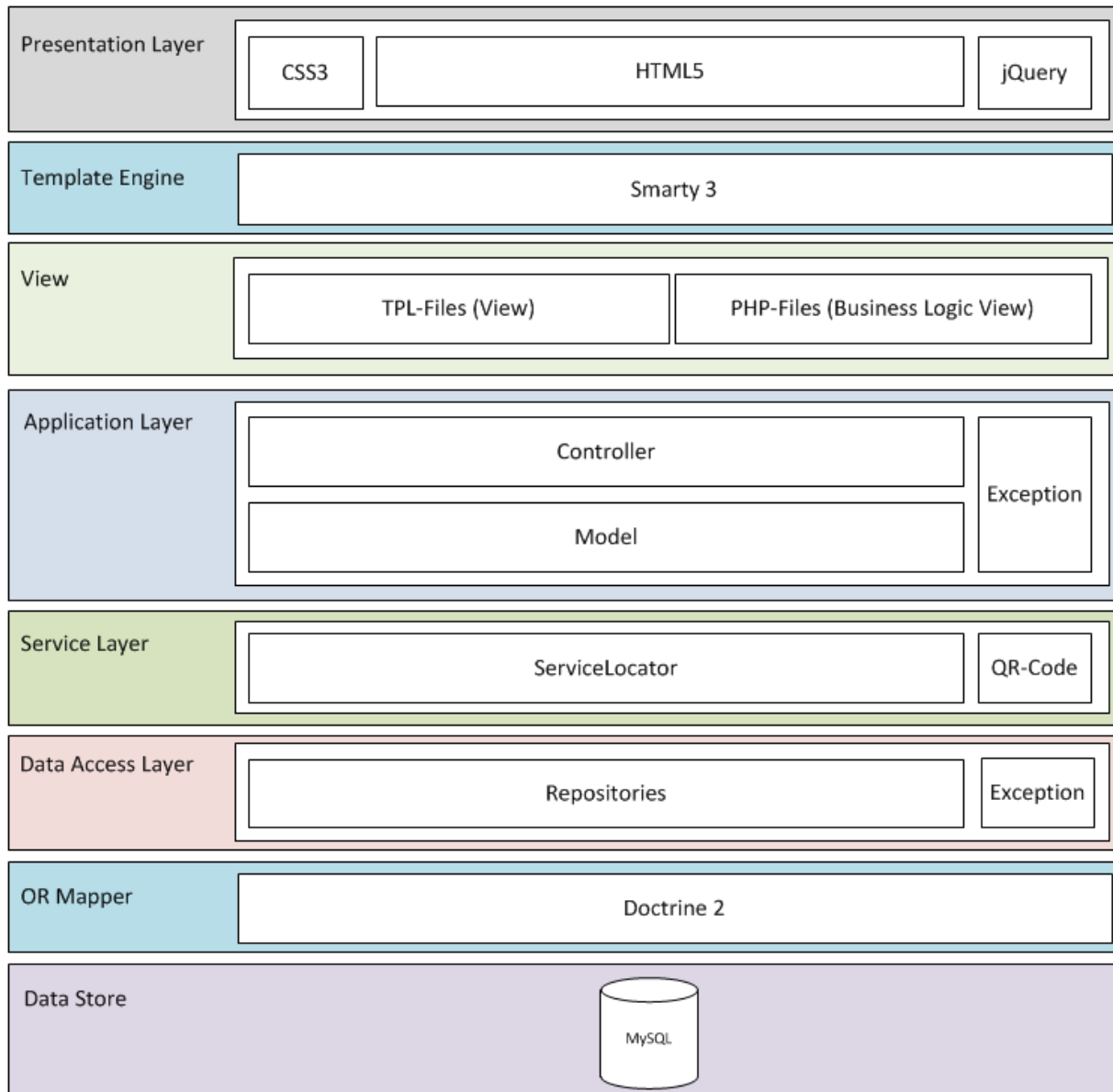


Abbildung 8.5.: Das Layer-Diagramm der Anwendung

## 8.5.2. Ordnerstruktur

Die Ordnerstruktur des Projekts sieht folgendermassen aus:

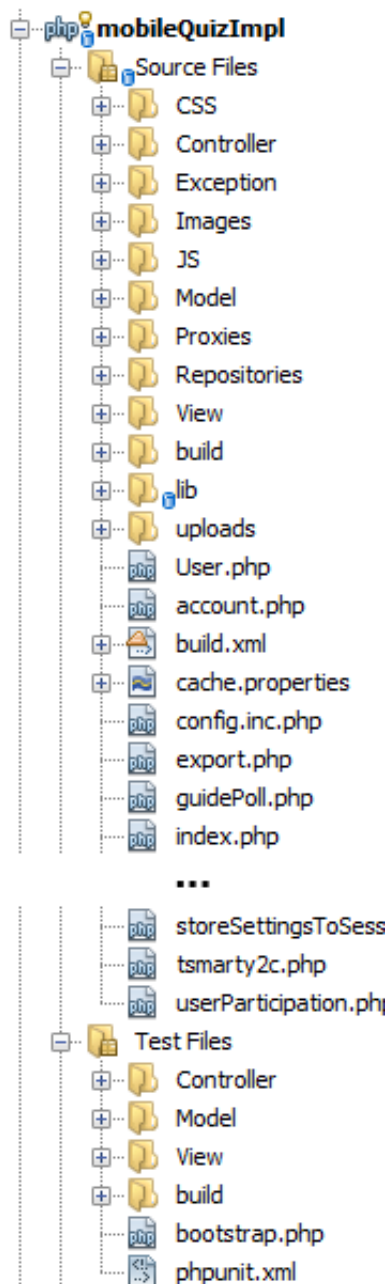


Abbildung 8.6.: Ordnerstruktur des Projekts

**CSS** Beinhaltet die CSS-Dateien.

**Controller** Enthält die Controller. Dazu gehört der Quiz-, Report-, Session- und UserController. Beinhaltet zusätzlich das setup.php mit dem das Datenbank-Schema erstellt wird.

**Exception** Beinhaltet Exception-Klassen.

**Images** Enthält Bilder, die in der Webanwendung eingesetzt werden.

**JS** Beinhaltet die JavaScript-Dateien.

**Model** Enthält die Model-Klassen. Diese sind in der Datenbank abgebildet.

**Proxies** Proxy Objekte von Doctrine werden hier abgelegt.

**Repositories** Enthält die Repositories. Repositories sind Klassen, die Daten aus der Datenbank extrahieren.

**View** Beinhaltet die TPL-Dateien und Helper-Klassen für die View.

**build** Enthält die build Log-Dateien.

**lib** Libraries, wie Smarty und Doctrine, sowie die Klassen der sozialen Netzwerke können hier gefunden werden.

**uploads** Enthält die Bilder und CSV-Dateien, die von den Benutzern hochgeladen werden.

**root** Beinhaltet die PHP-Dateien.

**Test Files** Enthält die Unit Tests der Controllers, Models und der View Helpers.

### 8.5.3. Zusammenspiel im View Layer

#### Navigationspfade im Bezug auf Dateien

Um eine Übersicht über die Dateien und ihre Zusammenhänge zu erhalten, werden auf den folgenden Abbildungen die Navigationspfade der Anwendung aufgezeigt.

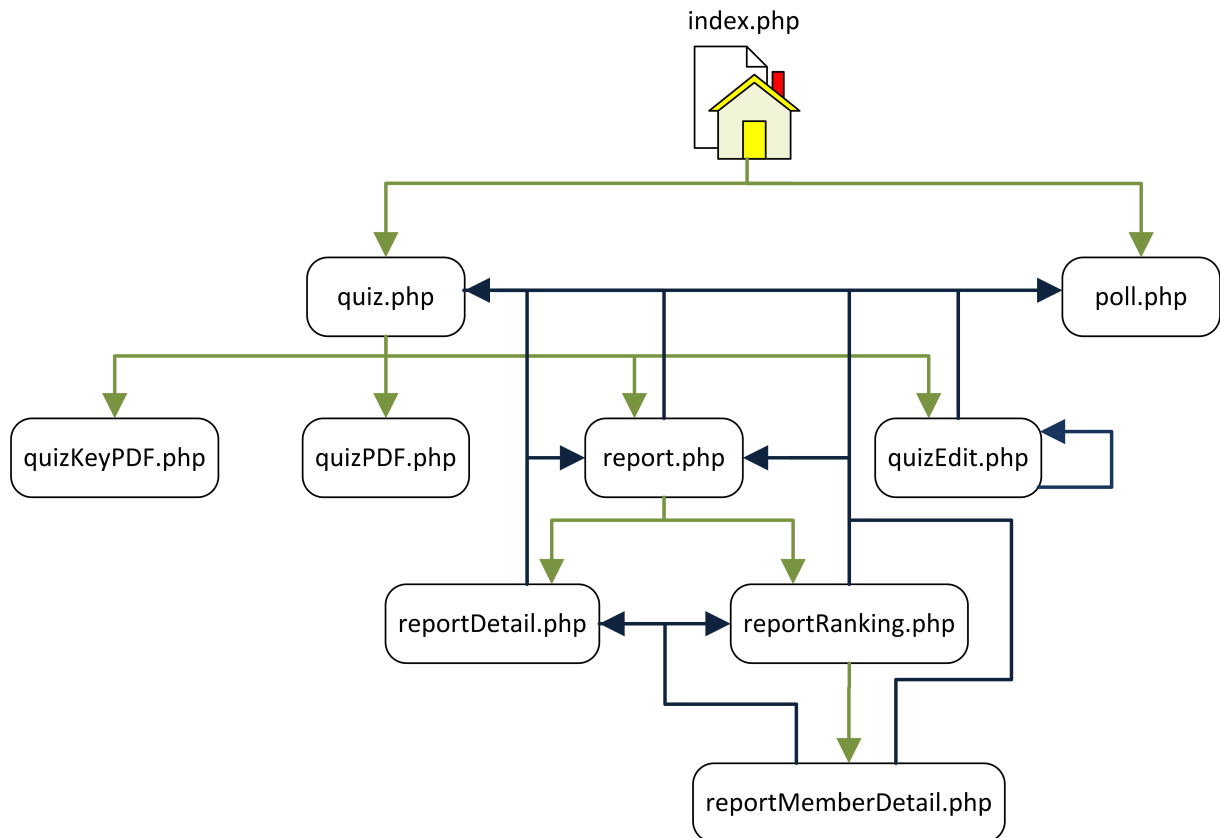


Abbildung 8.7.: Navigationspfade der Lernkontrolle

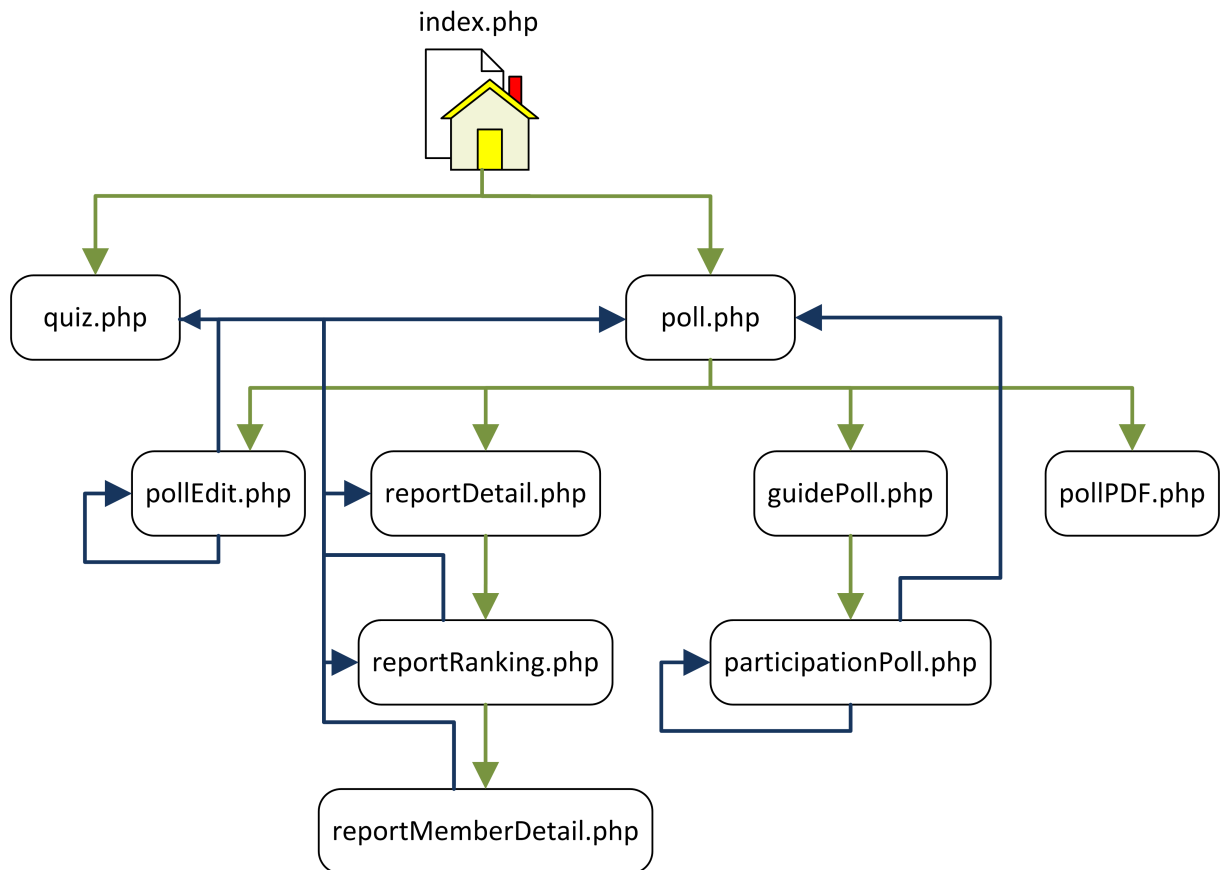


Abbildung 8.8.: Navigationspfade der Umfrage

### Administration

Wie in der Abbildung zu sehen ist, hat grundsätzlich jede TPL-Datei sein PHP Gegenstück.

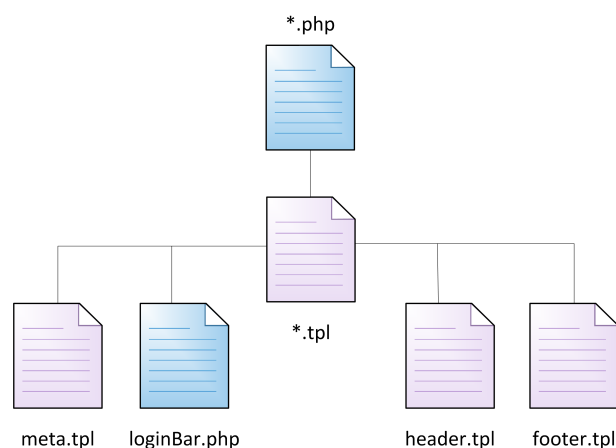
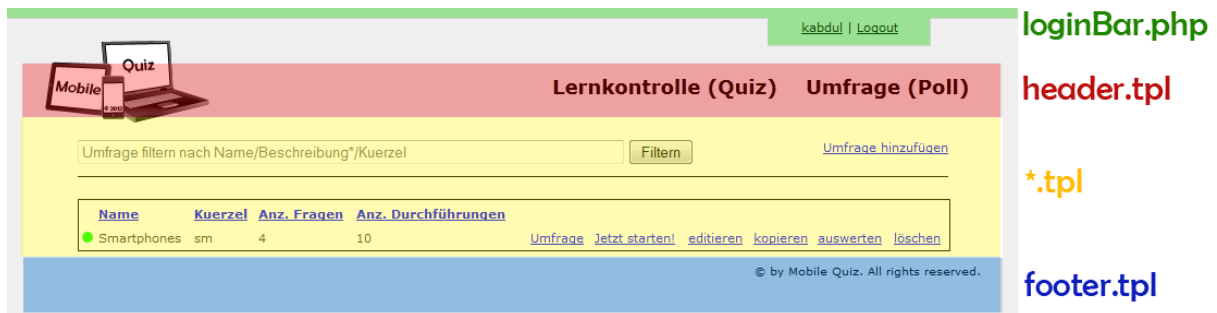


Abbildung 8.9.: Grundlegendes Zusammenspiel zwischen PHP- und TPL-Dateien im Code

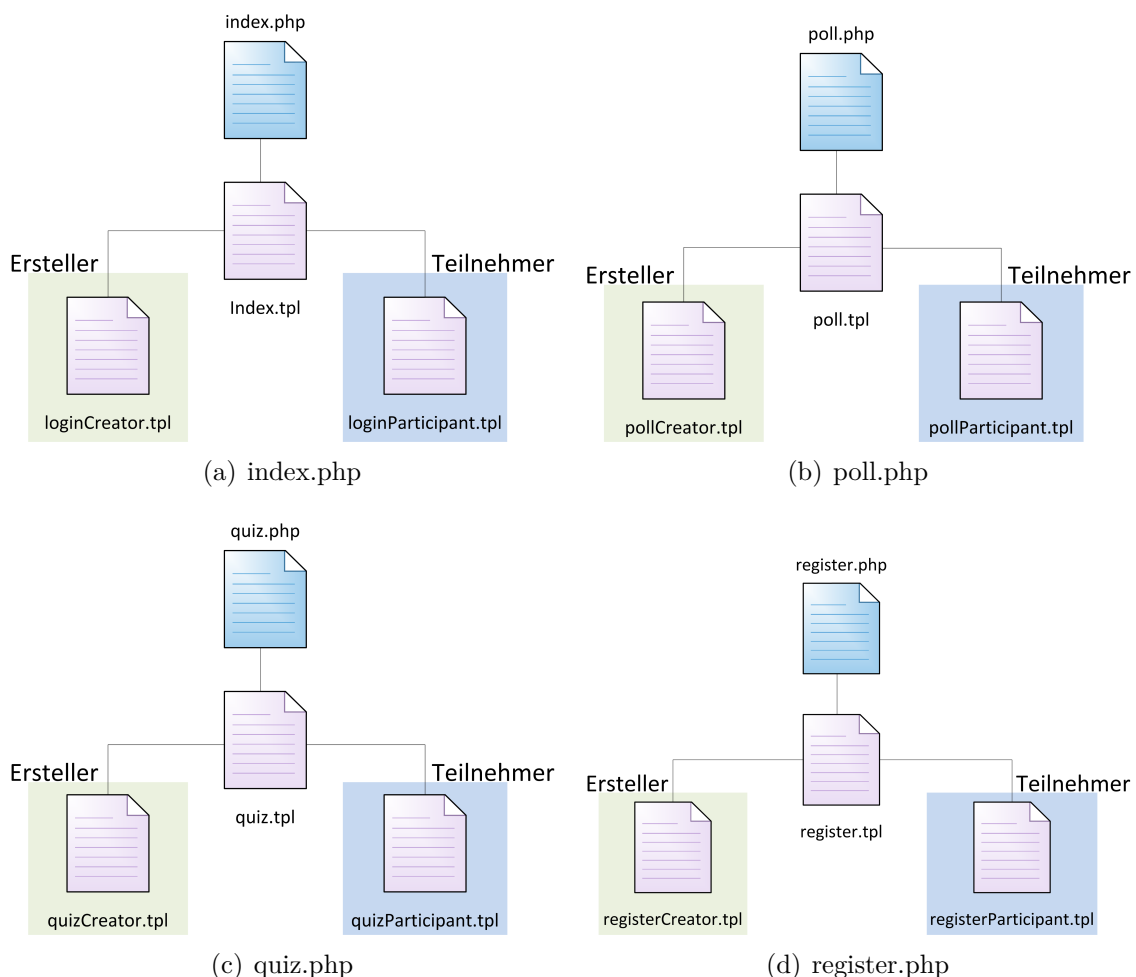
In der `meta.tpl` Datei sind die Angaben zur Website enthalten, wie Autor, Beschreibung, etc.. Bei der `loginBar` handelt es sich, im Gegensatz zu den anderen Dateien, um eine PHP-Datei. Dies aus dem Grund, weil zur Darstellung zusätzliche Informationen, wie der Name des Users, benötigt werden. Diese Information wird in der PHP-Datei aufbereitet. Die

Navigation befindet sich in der Datei header.tpl. Die Fussleiste ist in der Datei footer.tpl abgebildet. Die Abbildung 8.10 zeigt das Zusammenspiel dieser Dateien schematisch auf.



**Abbildung 8.10.:** Grundlegendes Zusammenspiel zwischen PHP- und TPL-Dateien im GUI

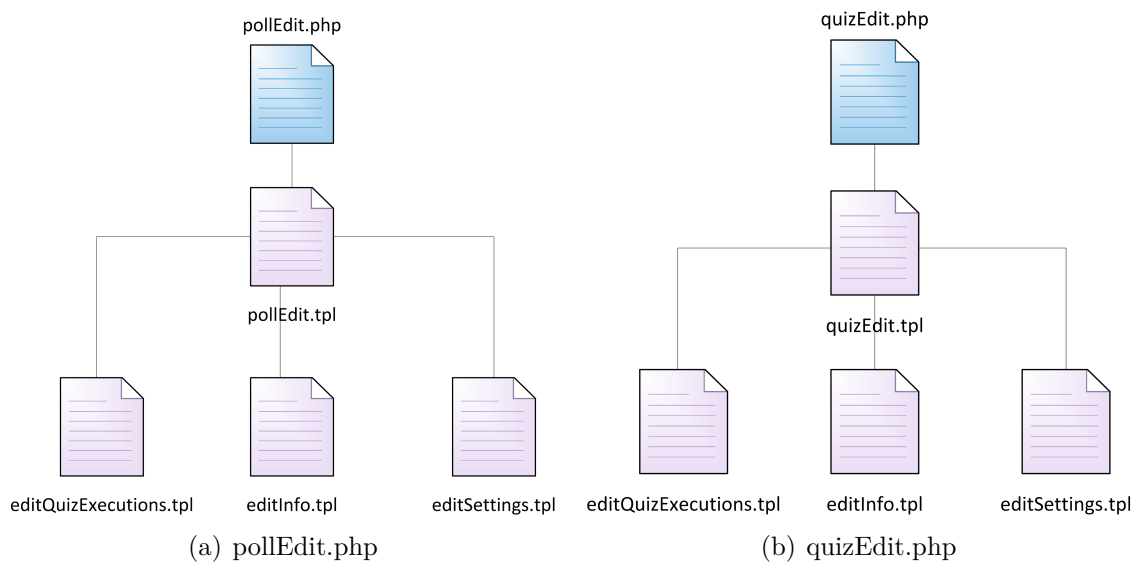
Es gibt auch Fälle, in denen eine TPL-Datei mehrere andere TPL-Dateien mit einbezieht:



**Abbildung 8.11.:** Zusammenspiel der Navigation

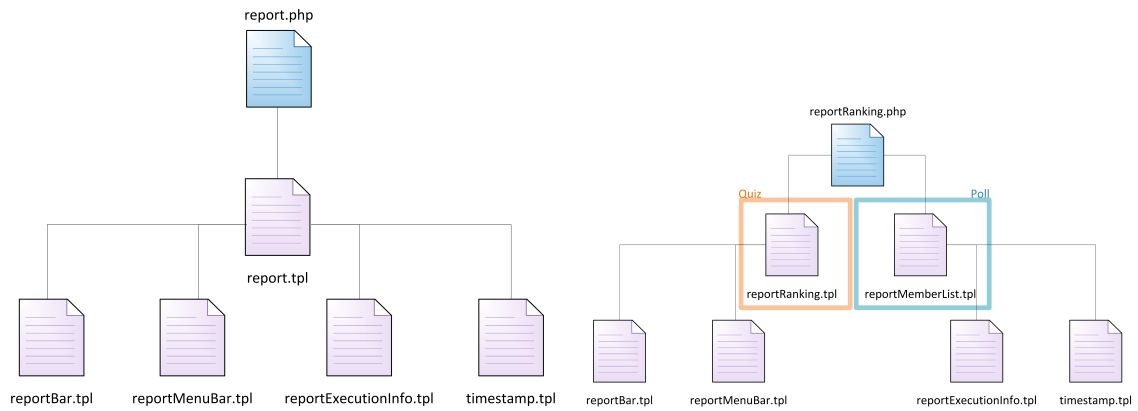
Die Abbildung 8.11 zeigt die vier PHP-Dateien index, poll, quiz und register. Die Zusammensetzung ist von der Rolle des Angemeldeten abhängig. Meldet sich ein Benutzer als Ersteller an, werden die TPL-Dateien des Erstellers angezeigt.





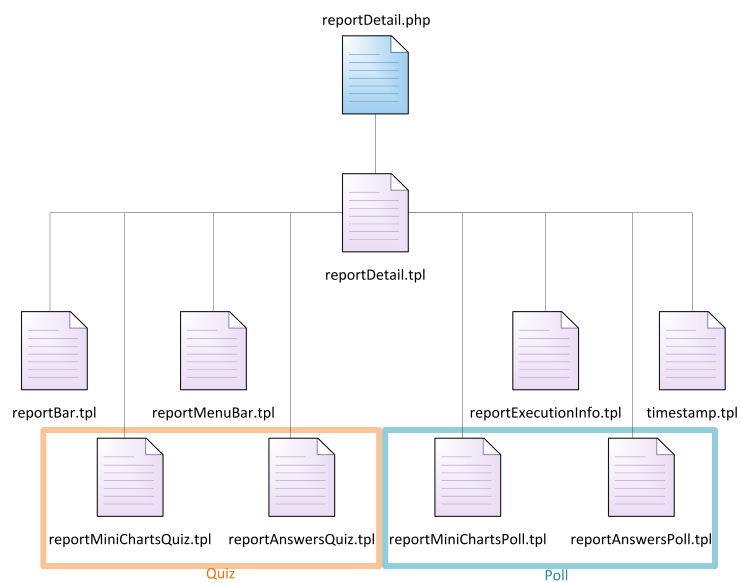
**Abbildung 8.12.:** Zusammenspiel der PHP- und TPL-Dateien in der Editierung

Das Editieren der Lernkontrolle unterscheidet sich grundsätzlich nicht vom Editieren der Umfrage. Die Unterschiede finden sich in den quizEdit- bzw. pollEdit-Dateien. Aus diesem Grund wurden die Bereiche der View ausgelagert, die bei beiden identisch sind. Damit bleibt die Struktur übersichtlich.

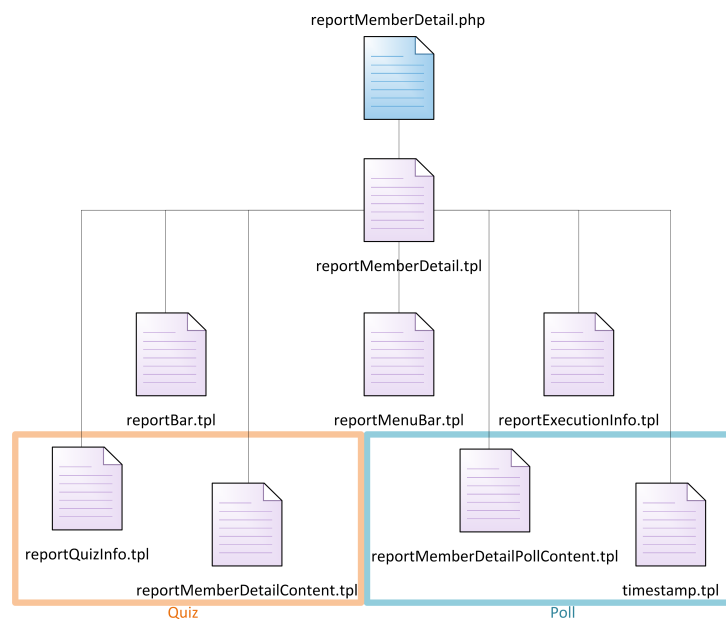


(a) report.php

(b) reportRanking.php



(c) reportDetail.php



(d) reportMemberDetail.php

Abbildung 8.13.: Zusammenspiel der PHP- und TPL-Dateien in der Auswertung

In der Abbildung 8.13 ist ersichtlich, wie die Auswertungsseiten zusammenspielen. Dabei werden, je nachdem ob sich die Auswertung auf eine Lernkontrolle oder Umfrage bezieht, andere Dateien inkludiert.

### Teilnahme

Die Teilnahme, welche mit jQuery Mobile realisiert wurde, enthält keine Verschachtelungen des User Interfaces. Pro PHP-Datei wird eine zugehörige TPL-Datei verwendet. Dies beinhaltet folgende Dateien:

- guidePoll.php - Startseite des Präsentationsmodus von Umfragen.
- participation.php - Startseite der Teilnahme.
- participationDecision.php - Wird im Präsentationsmodus benötigt, um diesen zu beenden.
- participationPoll.php - Resultatseite des Präsentationsmodus der Umfrage.
- participationPollEnd.php - Endseite der Teilnahme.
- participationPreview.php - Startseite der Vorschau.
- participationQuestion.php - Stellt die Fragen und Antworten der Lernkontrolle/Umfrage dar.
- participationQuestionPreview.php - Stellt die Fragen und Antworten in der Vorschau der Lernkontrolle/Umfrage dar.
- participationReport.php - Beinhaltet die Auswertung der Lernkontrolle.
- participationResult.php - Stellt die Resultate der einzelnen Fragen dar.
- participationResultPreview.php - Stellt die Resultate der einzelnen Fragen in der Vorschau dar.
- participationWaitScreen.php - Beinhaltet den Warteraum einer Umfrage im Präsentationsmodus.

### Sonstige Dateien

Ausserdem gibt es Druckansichten, die für sich alleine stehen. Diese stellen die folgenden Dateien dar:

- pollKeyPdf.php - Druckansicht der Umfrage.
- pollPdf.php - Druckansicht der Resultate der Umfrage.
- quizKeyPdf.php - Druckansicht der Lernkontrolle.
- quizPdf.php - Druckansicht der Resultate der Lernkontrolle.

Zusätzlich zu den oben behandelten Fällen enthält die Anwendung Dateien, die keine Anbindung an TPL-Dateien enthalten. Zu diesen gehören folgende:

- export.php - Wird für den Export der Fragen und Antworten benötigt.
- login-facebook.php - Beinhaltet die Logik des Logins per Facebook.
- login-google.php - Beinhaltet die Logik des Logins per Google.
- login-twitter.php - Beinhaltet die Logik des Logins per Twitter.
- participationDoRegister.php - Behandelt die Registrierung der Teilnehmenden.
- participationLogin.php - Behandelt das Anmelden der Teilnehmenden.
- participationLogout.php - Behandelt das Abmelden der Teilnehmenden.
- participationValidateAnswer.php - Speichert die Antworten der Teilnehmenden.
- register-facebook.php - Wickelt die Registrierung mit Facebook ab.
- storeSettingsToSession.php - Wird für die Speicherung der Einstellungen verwendet.
- smarty2c.php - Wird für die Internationalisierung benötigt.

## 8.6. Sequenzdiagramme

Die Sequenzdiagramme zu Mobile Quiz sind unter `100_Requirements/Sequenzdiagramme.pdf` zu finden.

## 9. Verwendete Technologien

### 9.1. Entwicklungsumgebung

Zu Beginn der Arbeit wurden zwei Entwicklungsumgebungen evaluiert. Eine davon war Aptana[8], die andere Netbeans[9]. Aptana ist eine Entwicklungsumgebung die speziell für Webanwendung erstellt wurde und bietet Code-Completion für HTML, CSS und JavaScript. Nach einigen Tests zusammen mit PHP, Smarty und Doctrine wurde ersichtlich, dass Netbeans als Entwicklungsumgebung komfortabler ist. Verschiedene Funktionalitäten, die in der Arbeit von Interesse waren, sind bereits in der Standardversion enthalten. Deshalb wurde als Entwicklungsumgebung die Netbeans IDE 7.1.1 zusammen mit XAMPP[10] 1.7.7 installiert.

#### 9.1.1. PHP, MySQL, HTML5 und CSS3

Die Anwendung ist mit der Programmiersprache PHP[11] realisiert. Zusammen mit MySQL[12] bildet diese die Grundlage für den restlichen Teil der Anwendung. Zum Zeitpunkt der Realisierung wurde die PHP Version 5.3.8 und die MySQL Version 5.5.16 eingesetzt. Der Inhalt der Webanwendung ist mit der Auszeichnungssprache HTML5[13] realisiert. Durch den Einsatz von CSS3[14], lassen sich Elemente der Darstellung individuell ausrichten und anpassen.

#### 9.1.2. JavaScript

Als JavaScript Library kam jQuery[17] zum Einsatz. Dies aus dem Grund, dass jQuery den Studierenden an der Hochschule für Technik Rapperswil im Rahmen des Moduls “Internettechnologien” näher gebracht wird. Zusätzlich bietet jQuery die Library jQuery UI[18] und das Framework jQuery Mobile[19] an. jQuery Mobile ist ein eigenständiges Framework, das auf jQuery und jQuery UI aufbaut. Dieses wird in der Anwendung auf der Teilnehmerseite angewendet, da es verschiedene mobile Endgeräte unterstützt.

### 9.2. Frameworks

#### 9.2.1. Smarty

Smarty[20] ist eine Template Engine, die es erlaubt, die Präsentationsschicht von der Applikationsschicht zu trennen. Dies bedeutet, dass die HTML Seite in einer Template (.tpl) Datei untergebracht wird und im Hintergrund eine PHP-Datei zu jedem dieser .tpl-Dateien besteht. Standardmässig werden die Zeichen { und } verwendet, um den Smarty Code zu kennzeichnen. Da dies zu Problemen führt, wenn in JavaScript Smarty eingesetzt wird, wurden diese Zeichen angepasst. Aus “{” wurde “<!--{” und aus “}” wurde “-->”.

### 9.2.2. PHPUnit

Um die Qualität der Anwendung zu garantieren, wurden Unit Tests geschrieben. PHPUnit ist ein Framework, das es einem erlaubt Unit Tests zu schreiben. Diese Tests können jederzeit gestartet werden, um zu überprüfen ob Erweiterungen oder Änderungen den bestehenden Quellcode beeinflussen. Informationen zu den Tests in der Anwendung können im Kapitel 10. Testing entnommen werden.

### 9.2.3. Doctrine

Bei Doctrine<sup>[22]</sup> handelt es sich um ein Framework zur objektrelationalen Abbildung für PHP. Mittels Annotationen werden in den Klassen die Einstellungen und Konfigurationen für die Datenbank definiert. Damit können Objekte persistiert werden.

# **Teil IV.**

## **Qualitätssicherung**

# 10. Testing

In diesem Kapitel werden die Methoden beschrieben, die zur Qualitätssicherung des Codes beigetragen haben.

## 10.1. Systemtests

### 10.1.1. Test Driven Development

Um möglichst alle Bereiche der Anwendung testen zu können, wurde auf die Vorgehensweise Test Driven Development gesetzt. Das bedeutet, dass vor der Implementierung einer Funktion, die Tests dazu geschrieben werden. Diese Tests umfassen die Logik der Funktion, sowie Grenzwerte für zulässige Übergabeparameter oder Zustände.

### 10.1.2. Unit Test

Der gesamte Application Layer wird mit 447 Unit Tests geprüft. Diese Tests prüfen die Model-Klassen und Controller-Klassen. Dabei wird die Logik der Funktionen getestet, sowie die Grenzwerte der jeweiligen Übergabeparameter für die Funktionen. Die Grenzwerte der unteren, wie auch der oberen Grenze, werden geprüft. Ausserdem wird überprüft, ob die Exceptions zum richtigen Zeitpunkt geworfen werden.

Bevor eine Funktion implementiert wird, werden die Unit Tests dazu geschrieben. Die Tests decken den Umfang der Funktion ab. Somit wird vor der Implementation festgehalten, welche Übergabeparameter welche Werte erzeugen. Zu Beginn resultieren die Tests in einem roten Balken, der das Scheitern der Tests kennzeichnet. Daraufhin wird Test für Test die Implementierung vorgenommen, bis alle Tests mit einem grünen Balken abschliessen.

Bei Unit Tests gibt es die Möglichkeit Objekte zu mocken. Somit können zeitaufwändige, situationsabhängige oder datenbankspezifische Angelegenheiten getestet werden, die sonst durch ihre Implementierung bei jeder Ausführung in einem anderen Resultat enden würden.

Über den gesamten Application Layer wurde eine Code Coverage von 92,68% erreicht. Es wurde eine Code Coverage von mindestens 90% angestrebt. Die restlichen 7,32% können nicht abgedeckt werden. Dies hat verschiedene Gründe:

- Wenn ein Objekt erzeugt werden soll und dies aus irgendwelchen Gründen nicht funktioniert, wird eine Exception geworfen. Diese Situation kann mit Unit Tests nicht überprüft werden, da sie nicht nachzubilden ist.
- Wenn eine Funktion mit ID's von Objekten arbeitet. Objekt ID's können mit Mock-Objekten nicht imitiert werden. Daher fallen diese Funktionen aus dem Testrahmen.



- Wenn eine Funktion eine Situation überprüft, die vom Zustand der Internetverbindung abhängig ist.
- Wenn eine Funktion nicht erreichbar ist, weil sie privat ist und die Methode, die sie aufruft aus einem der vorhin genannten Gründen nicht getestet werden kann.
- Unter Umständen kann es zu aufwändig sein, eine Basis für die Tests aufzubauen. Dies ist ebenfalls ein Grund, warum Funktionen nicht getestet werden.

Die Code Coverage für die einzelnen Klassen, aufgeteilt in Packages:

### Controller

Klasse	Code Coverage	Anzahl Tests
QuizController	95,93%	63
ReportController	91,74%	51
SessionController	96,23%	28
UserController	97,47%	28

### Model

Klasse	Code Coverage	Anzahl Tests
Answer	92,59%	15
Execution	83,13%	32
Question	91,07%	24
Quiz	90,53%	43
Session	91,09%	50
Settings	95,00%	17
User	94,70%	73

**Tabelle 10.1.:** Werte zu den Unit Tests

In der Tabelle 10.1 sind die Werte zu allen Klassen angegeben. Wie aus der Tabelle entnommen werden kann, hat die Klasse Execution das gesetzte Ziel der Code Coverage von 90% nicht erreicht. Die Methode `getState()` in der Klasse Execution kann nicht getestet werden, da die Logik vom heutigen Tag abhängig ist. Der Vergleich, in welchem Zustand sich die Durchführung befindet, basiert auf `new DateTime()`. Daher kann der Vergleich nicht beeinflusst werden und die Methode somit damit nicht testbar.

## 10.2. Informelles zum Code

In diesem Abschnitt wird Informelles zum Code aufgelistet. In der Statistik werden die Klassen, die sich im Unterordner lib befinden, nicht miteinberechnet.

Beschreibung	Wert
Anz. Zeilen:	9251
Kommentierte Anz. Zeilen:	2464
Nicht-Kommentierte Anz. Zeilen:	6787
Anz. Klassen:	44
Durchschnittliche Klassenlänge Anz. Zeilen:	97
Anz. Methoden	379
Durchschnittliche Methodenlänge Anz. Zeilen:	11

**Tabelle 10.2.:** Code Statistiken

Auf den ersten Blick sehen die Anz. Zeilen Code mit 9251 nach viel aus. Wie man aber in der Tabelle erfährt, sind davon 2464 Zeilen kommentiert, also PHPDoc. PHPDoc wurde bei jeder Methode eingesetzt, um die Übergabe- und Rückgabeparameter zu beschreiben, weil es in PHP nur var als Datentypen gibt.

Im Code wurden Teile von Drittanbietern bezogen. Zu diesen gehören die Klassen für die sozialen Netzwerke:

- Google[28]
- Facebook[29]
- Twitter[30]

Plugins für jQuery:

- jQuery Validation Engine[31]
- jQuery Countdown[32]

Für die QR-Codes wurde Google Chart Tools[33] verwendet. Um E-Mails zu versenden, wurde PHPMailer[34] eingesetzt.

## 10.3. Usability Tests

Da es sich um eine User Interface orientierte Webanwendung handelt, die auf verschiedenen Geräten einwandfrei laufen soll, sind Usability Tests von grosser Bedeutung.

### 10.3.1. Eingabemethoden

Weil man sich einer Frage enthalten kann, wurden zu Beginn der Arbeit verschiedene Methoden evaluiert, wie man eine Eingabe mit drei Zuständen realisieren kann. Dazu wurde eine Umfrage gemacht, die eine Lernkontrolle simulierte, die aus sechs Fragen bestand. Zu jeder Frage wurde eine andere Eingabemethode angezeigt. Die Fragen wurden in einer zufälligen Reihenfolge durchgearbeitet, damit die Reihenfolge der Fragen das Abstimmungsergebnis der Eingabemethode nicht beeinflusste. So konnte ausgeschlossen werden, dass eine Methode nicht nur anhand der Position in der Umfrage positiv oder negativ ausfiel. Am Ende der Umfrage konnten die Teilnehmenden aus den sechs verschiedenen Eingabemethoden die für sie Beste wählen. Dazu wurde um Tendenzen erkennen

zu können, festgehalten, mit welchem Gerät an der Umfrage teilgenommen wurde. Bei den Eingabemethoden handelte es sich um folgende:

**Eingabemethoden Evaluierung**  
Auf den folgenden Seiten werden Ihnen eine handvoll Multiple-Choice Fragen gestellt, bei denen mehr als eine Antwort richtig sein können.

Es ist möglich zu einer Frage keine Antwort zu geben. Dies aus dem Grund, dass eine richtige Antwort +1, eine Falsche -1 und keine Antwort 0 Punkte gibt.

In diesem Beispiel geht es lediglich darum verschiedene Eingabemethoden auf ihre Benutzerfreundlichkeit zu prüfen. Am Ende können Sie für die beste Eingabemethode stimmen.

weiter

(a) Start der Umfrage

**Welche Personen waren/sind Diktatoren?**

F   kA   R

☒ ☐ ☐ Silvio Berlusconi  
☐ ☐ ☒ Pervez Musharraf  
☐ ☐ ☒ Kim Jong-il  
☒ ☐ ☐ Barack Obama

Falsch   keine Antwort   Richtig

weiter

(b) Radiobuttons

**Welche Länder sind grösser (Fläche) als die Schweiz?**

kA

Deutschland ☐  
Liechtenstein ☐  
Luxemburg ☐  
Frankreich ☐

Falsch   Richtig   keine Antwort

weiter

(c) Buttons

**Welche Länder sind grösser (Fläche) als die Schweiz?**

R/F   kA

☒ Deutschland ☐  
☐ Liechtenstein ☐  
☐ Luxemburg ☐  
☒ Frankreich ☐

Richtig   Falsch   keine Antwort

weiter

(d) Checkbox ohne Hintergrund (CheckboxV1)

**Welche Tiere gehören zur Klasse der Säugetiere?**

R/F   kA

☐ Frosch ☐  
☐ Hai ☐  
☒ Katze ☐  
☐ Adler ☐

Richtig   Falsch   keine Antwort

weiter

(e) Checkbox mit Hintergrund (CheckboxV2)

**Welche Fahrzeuge haben vier Räder?**

☒ Dreirad  
☒ Quad  
☐ Lastwagen  
☒ Auto

☐ keine Antwort ☒ Richtig ☒ Falsch

weiter

(f) Checkbox mit drei Status (Tristate)

**Welche Länder grenzen an Österreich?**

F   kA   R   Deutschland  
F   kA   R   Kroatien  
F   kA   R   Frankreich  
F   kA   R   Ungarn

Falsch   keine Antwort   Richtig

weiter

(g) Toggle

**Dieser Teil wird ausgewertet**

**Welche der Eingabemethoden war am besten?**

☐ Radiobuttons  
☐ Checkbox V1  
☐ Checkbox V2  
☐ Tristate Checkbox  
☐ Buttons  
☐ Toggle  
☒ Keine der oben genannten

Kommentar oder Anregungen (optional):

abschicken

(h) Ende der Umfrage

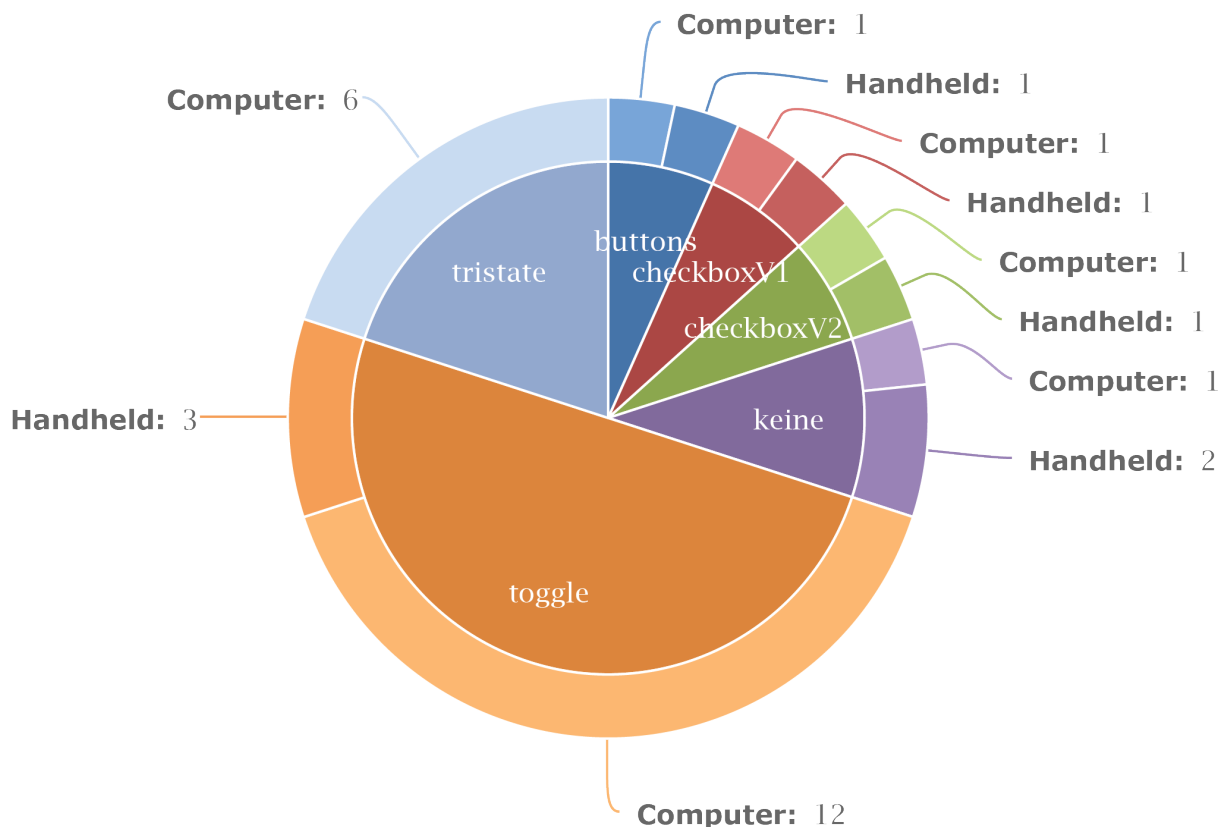
Abbildung 10.1.: Aufbau der Umfrage

Diese Umfrage wurde von insgesamt 30 Personen ausgefüllt, unter denen sich ebenfalls Leute mit weniger Computererfahrung befanden. Von den Probanden wurde zu Beginn ihrer Teilnahme der User Agent ausgelesen. Der User Agent liefert Informationen zum Browser und zum System des Users. Dies sieht zum Beispiel wie folgt aus:

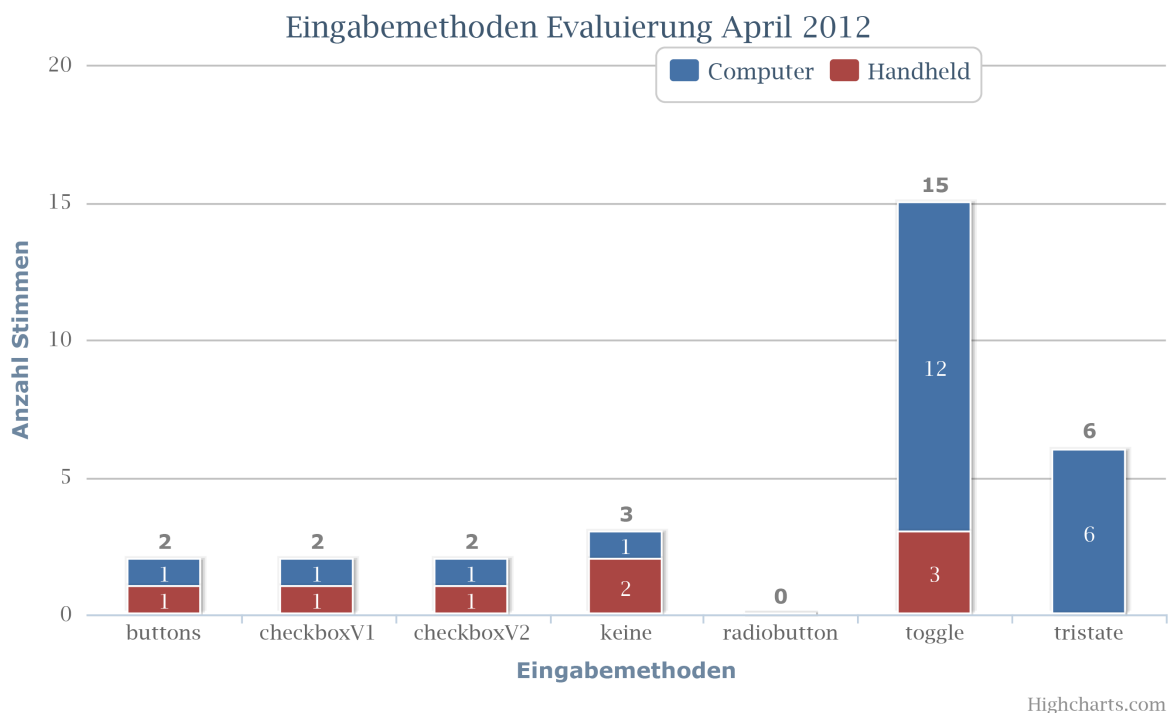
- Mozilla/5.0 (Linux; U; Android 2.3.7; de-ch; Optimus 2X Build/FRG83G) AppleWebKit/533.1 (KHTML, like Gecko) Version/4.0 Mobile Safari/533.1
- Mozilla/5.0 (BlackBerry; U; BlackBerry 9800; en) AppleWebKit/534.8+ (KHTML, like Gecko) Version/6.0.0.534 Mobile Safari/534.8+
- Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; .NET4.0C; .NET4.0E)
- Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:11.0) Gecko/20100101 Firefox/11.0

Mit diesen Informationen wurde eine Auswertung erstellt, welche die Teilnehmenden anhand ihrer Entscheidung und zusätzlich mit ihrem Eingabegerät unterteilt. Daraus entstand folgendes Resultat:

### Eingabemethoden Evaluierung April 2012



**Abbildung 10.2.:** Auswertung der Eingabemethoden-Umfrage als Kuchendiagramm



**Abbildung 10.3.:** Auswertung der Eingabemethoden-Umfrage als Balkendiagramm

Die Ergebnisse der Auswertung zeigen klar, dass der Toggle aus Abbildung 10.1(g) der Favorit ist. Die Auswertung der Kommentare der Teilnehmenden zeigen jedoch, dass Radiobuttons ebenfalls wünschenswert wären. Hier einige Kommentare[23]:

*«Der toggle ist, zusammen mit den Radiobuttons, auf dem Smartphone am intuitivsten zu bedienen. Bei den Checkboxes und den Buttons ist es relativ ungewohnt, dass man zweimal klicken muss um eine Antwort als Falsch zu kennzeichnen. Die Tristate checkbox ist ebenfalls eine nicht ganz so intuitive Lösung.»*

*«Toggle, Tristate und Radiobuttons sind die Varianten die am intuitivsten sind. Die Checkbox Varianten sind ein wenig verwirrend. Es ist ungewohnt eine neutrale Antwort abzugeben. Scheint mir jedoch fair zu sein.»*

*«Ausser Radiobuttons und Toggle sind alle anderen sehr verwirrend.»*

Die Teilnehmenden scheinen sich schliesslich für den Toggle entschieden zu haben, weil er originell und neuartig ist. Der Toggle ist dem heutigen Standard der Smartphonebedienung angepasst und intuitiv bedienbar.

### 10.3.2. Anwendung

Am Ende der Entwicklung wurde die Anwendung von mehreren Personen geprüft. Unter Beobachtung der Entwickler wurde den Testpersonen die Webanwendung vorgelegt und sie erhielten eine vordefinierte Aufgabe, welche sie zu bewältigen hatten. Diese Aufgaben reichten vom Lernkontrolle Erstellen mit Importieren von Fragen und Publizieren der Lernkontrolle bis zur Teilnahme an einer Lernkontrolle mit dem Eingabegerät nach Wahl.

Die Testpersonen wurden aufgefordert, alles was sie gerade dachten, auszusprechen, damit die Entwickler die einzelnen Schritte der Probanden nachvollziehen konnten. Durch diese Tests wurde das User Interface stetig verbessert, um den Benutzern eine möglichst intuitive Bedienung zu bieten.

# 11. Reviews

## 11.1. Code-Review

Der Code musste mindestens einmal durch den Projektpartner überprüft und wenn möglich verbessert werden. Ende jeder Woche wurden diese Reviews durchgeführt. Änderungen wurden mündlich kommuniziert und mussten nicht dokumentiert werden. Die durch den Revisor zu überprüfende Punkte waren folgende:

- Verständlichkeit des Code-Abschnittes
- Korrektheit des Code-Abschnittes
- Auftreten von “Code Smells”

## 11.2. Dokument-Review

Jedes geschriebene Kapitel, so wie jedes Dokument musste mindestens einmal durch den Projektpartner überprüft und wenn nötig verbessert werden. Die Kapitel und Dokumente wurden immer Ende Woche durch ein Review überprüft. Die durch den Revisor zu überprüfenden Punkte:

- Grammatik und Rechtschreibung
- Konsistenz des Textes
- Vollständigkeit des Dokuments



# **Teil V.**

## **Schlussfolgerung**

## 12. Schlussfolgerung

### 12.1. Zusammenfassung

#### 12.1.1. Ergebnis

Im Rahmen dieser Bachelorarbeit wurde ein webbasierter Prototyp erstellt, der es Dozierenden erlaubt, Lernkontrollen und Umfragen zu erstellen und ihre Durchführungen zu planen. Mittels portablen Geräten, wie Laptop, Tablet oder Smartphone kann an diesen Durchführungen teilgenommen werden.

Auf Funktionen, wie beispielsweise ein Lückentext zu erstellen, wurde während der Arbeit bewusst verzichtet, da dies die Übersichtlichkeit und Einfachheit verringert hätte. Dies wäre eine Funktion die in Zukunft angegangen werden könnte. Nicht implementiert wurde auch die Funktion, einem Teilnehmenden zu erlauben seine Antworten, bis zum Ablauf der Zeit einer Durchführung, zu korrigieren. Darauf wurde deswegen verzichtet, weil die Funktion zu einem späteren Zeitpunkt als unwichtig eingestuft wurde. Weiter wurde die Unterstützung von Spezialzeichen nicht vollumfänglich realisiert. Mathematische Formelzeichen sind nicht eingebettet, Sonderzeichen des deutschen Alphabets lassen sich jedoch ohne Probleme verwenden.

Allerdings wurden während der Bachelorarbeit Funktionen implementiert, welche von Mitbewerbern nicht oder unvollständig realisiert wurden. Zu diesen gehören beispielsweise das Verteilen von Negativpunkten bei Falschantworten. Dies wird auch in schriftlichen Multiple-Choice-Prüfungen der Hochschule für Technik Rapperswil so gehandhabt. Ebenfalls, wie an Multiple-Choice-Prüfungen der HSR, können Teilnehmende bei einer Antwort keinen Tipp abgeben, um keine Negativpunkte zu riskieren. Weiter ist es mit Mobile Quiz möglich, Umfragen im Präsentationsmodus zu starten. So können Dozierende die Teilnehmenden durch die Fragen führen, wobei die Resultate fortlaufend auf dem Präsentationsmedium aktualisiert werden. Zudem wurde der Bereich der Zeitkontrolle realisiert. Mobile Quiz bietet die Möglichkeit, keine Zeit, eine Zeit pro Frage oder eine Zeit für die gesamte Durchführung festzulegen. Eine wichtige Funktion, die Mobile Quiz von anderen Mitbewerbern abhebt, ist das Ausfüllen von Lernkontrollen oder Umfragen auf mobilen Geräten, wie Laptops, Tablets oder Smartphones. Dies ist beim Grossteil der Mitbewerbern nicht vollständig umgesetzt oder nicht implementiert.

Die vollständige Liste mit Funktionen im Vergleich mit den Mitbewerbern kann dem Kapitel 5. Funktionsliste entnommen werden.

### 12.1.2. Schlussfolgerung

#### PHP

Das Erstellen der Anwendung mittels PHP erwies sich als relativ einfach. Zusammen mit Smarty, welches die Trennung der Presentation- und Business Layers ermöglicht, erscheint die Anwendung auf den ersten Blick zwar unaufgeräumt und verwirrend, ist jedoch durch die Trennung viel einfacher zu verstehen und weiterzuentwickeln. Die Möglichkeit, eine objektrelationale Abbildung mit Doctrine zu erstellen, erwies sich zu Beginn als schwierig, half aber in der Entwicklung der Anwendung ungemein. So wird die gesamte Persistierung der Daten durch Doctrine verwaltet. Da der Umfang des Projekts zu Beginn klein erschien, wurde auf Frameworks wie Zend oder Symfony verzichtet. Rückblickend wäre dies der einfacherer Ansatz gewesen die Webanwendung zu entwickeln.

#### Test Driven Development

Das Test-Driven-Development-Verfahren stellte sich als guter Ansatz für die Qualitätssicherung heraus. Da die Tests vor der Implementierung der Methode geschrieben wurden, musste zu Beginn der Implementation immer zuerst festgelegt werden, was von der Methode benötigt und erwartet wird. Dieser Ansatz half, von Anfang an die Klassen genau zu definieren. Die grösste Herausforderung bei diesem Ansatz stellte das Mocken von Objekten dar. Zu Beginn war das Mocken unmöglich, da keine Repositories vorhanden waren. Diese mussten für die einzelnen Model-Klassen erstellt werden. Dazu kam, dass einzelne Abschnitte der Konfiguration in den ServiceLocator verschoben werden mussten. Durch diese Umstellungen wurde das Mocken der Objekte möglich. Auch wenn dies einiges an Zeit beanspruchte, hat es sich auszubezahlt, den Test-Driven-Development-Ansatz im Projekt zu verfolgen.

## 12.2. Ausblick

Von Prof. Dr. Peter Heinzmann wird die Anwendung in Zukunft im Unterricht eingesetzt, um Umfragen und Lernkontrollen durchzuführen. Im Laufe der Zeit werden eventuell weitere Dozierende der HSR die Anwendung einsetzen. Längerfristig könnte die Anwendung auch in anderen Organisationen, wie Berufs- oder Mittelschulen, zum Einsatz kommen.

Ein Ausbau der Teilnahmemodi, um eine total anonyme Verwendung wäre interessant. Ein Modus in dem die Teilnehmenden sich nicht zuerst registrieren, bzw. anmelden müssen. Dies würde den Dozierenden, die nicht an Auswertungen interessiert sind, mehr Zeit während den Vorlesungen schenken.

Die Erweiterung der Fragetypen wäre denkbar. Beispielsweise um Lückentexte, offene Fragen oder Drag-and-Drop-Fragen und -Antworten.

In Zukunft wäre es vorstellbar die Import- und Exportformate anzupassen und mit Formaten, die bei anderen Umfrage-Anwendungen Verwendung finden, zu erweitern. Somit würde ein breiteres Publikum angesprochen, da bestehende Fragen auf einfachem Weg migriert werden könnten. Als Beispiel wären die Formate GIFT[36], Hot Potatoes[37] oder Moodle XML[38] denkbar, wobei noch weitere Formate evaluiert werden müssten.

Die Auswertung zu Lernkontrollen und Umfragen könnte um einige Punkte erweitert werden. So bestünde die Möglichkeit, beispielsweise eine Auswertung zu implementieren, die es erlaubt, Teilnehmende untereinander zu vergleichen. Auch wäre es interessant, eine Funktion anzubieten, die unterschiedliche Durchführungen miteinander vergleicht. Auf diesem Weg könnten Tendenzen erfasst werden, die für einen Erstellenden von Interesse sein könnten.

Für Dozierende wäre das Erstellen von Klassen eine weitere nützliche Funktion. Diese Funktion würde es erlauben, Studierende in Klassen zu unterteilen und sie Lernkontrollen und Umfragen zuzuweisen. Mit dieser Funktion könnte besser nachvollzogen werden, wie hoch die Teilnahmequote ist. Für die Teilnehmenden würde eine Art Portal entstehen. Auf diesem sind alle für den Teilnehmenden gedachten Lernkontrollen und Umfragen, sowie weitere statistische Auswertungen ersichtlich.

Die Einführung einer multilingualen Benutzeroberfläche wäre ebenfalls denkbar. Die notwendigen Vorarbeiten sind allesamt bereits erfolgt. Es müssten nur noch die anzubietenden Sprachen ermittelt und die dazugehörenden Übersetzungen gemacht werden.

Neben der Benutzeroberfläche könnten Lernkontrollen und Umfragen multilingual erfasst werden. Somit wäre eine Auswertung über Sprachgrenzen hinaus denkbar.

Eine weitere Funktion die in Zukunft angegangen werden könnte, wäre die Darstellung der Lernkontrollen/Umfragen für Legastheniker anzupassen. Damit wäre die Teilnahme für Legastheniker bei zeitkritischen Durchführungen einfacher.

# **Teil VI.**

## **Verzeichnisse**

## 12.3. Quellenverzeichnis

- [1] Tips for Taking Multiple Choice Tests, <http://www.kvhigh.com/learning/testtips.html>, letzter Zugriff am 28.05.2012
- [2] Test-wiseness, <http://en.citizendium.org/wiki/Test-wiseness>, letzter Zugriff am 28.05.2012
- [3] Facebook for Websites - Facebook Developers, <http://developers.facebook.com/docs/guides/web/>, letzter Zugriff am 29.05.2012
- [4] Sign in with Twitter | Twitter Developers, <https://dev.twitter.com/docs/auth/sign-twitter>, letzter Zugriff am 29.05.2012
- [5] Federated Login for Google Account Users - Google Accounts Authentication and Authorization - Google Developers, <https://developers.google.com/accounts/docs/OpenID>, letzter Zugriff am 29.05.2012
- [6] how to get email from twitter user using OAuthTokens, <https://dev.twitter.com/discussions/4019>, letzter Zugriff am 29.05.2012
- [7] Blowfish, <http://de.wikipedia.org/wiki/Blowfish>, letzter Zugriff am 28.05.2012
- [8] Aptana, <http://www.aptana.com>, letzter Zugriff am 03.06.2012
- [9] Welcome to Netbeans, <http://www.netbeans.org>, letzter Zugriff am 03.06.2012
- [10] apache friends - xampp, <http://www.apachefriends.org/en/xampp.html>, letzter Zugriff am 03.06.2012
- [11] PHP: Hypertext Preprocessor, <http://www.php.net>, letzter Zugriff am 03.06.2012
- [12] MySQL :: The world's most popular open source database, <http://www.mysql.com>, letzter Zugriff am 03.06.2012
- [13] HTML5, <http://en.wikipedia.org/wiki/HTML5>, letzter Zugriff am 03.06.2012
- [14] Cascading Style Sheets, [http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets), letzter Zugriff am 16.04.2012
- [15] YUI Library, <http://www.yuilib.com>, letzter Zugriff am 16.04.2012
- [16] MooTools - a compact javascript framework, <http://en.wikipedia.org/wiki/HTML5>, letzter Zugriff am 16.04.2012
- [17] jQuery: The Write Less, Do More, JavaScript Library, <http://www.jquery.com>, letzter Zugriff am 30.05.2012
- [18] jQuery UI - Home, <http://www.jqueryui.com>, letzter Zugriff am 18.04.2012

- [19] jQuery Mobile | jQuery Mobile, <http://www.jquerymobile.com>, letzter Zugriff am 22.05.2012
- [20] PHP Template Engine, <http://www.smarty.net/>, letzter Zugriff am 03.06.2012
- [21] sebastianbergmann/phpunit · GitHub, <https://github.com/sebastianbergmann/phpunit>, letzter Zugriff am 03.06.2012
- [22] Welcome to the Doctrine Project - Doctrine-Project, <http://www.doctrine-project.org>, letzter Zugriff am 03.06.2012
- [23] MobileQuiz - Eingabemethoden Evaluation, <http://khalid1.myhostpoint.ch/SampleQuizSmarty/resultOverview.php>, letzter Zugriff am 03.06.2012
- [24] quiz-and-poll - Engage your students and get instant feedback - Google Project Hosting, <http://code.google.com/p/quiz-and-poll>, letzter Zugriff am 09.05.2012
- [25] Google Docs, <https://docs.google.com>, letzter Zugriff am 09.05.2012
- [26] quiz-and-poll Template, [https://docs.google.com/spreadsheet/ccc?key=0Aqrbx\\_ov7Sd5dF9YQXdMNTlQczRxLTR5ZDBVb2tuY0E](https://docs.google.com/spreadsheet/ccc?key=0Aqrbx_ov7Sd5dF9YQXdMNTlQczRxLTR5ZDBVb2tuY0E), letzter Zugriff am 09.05.2012
- [27] StatCounter Global Stats - Browser, OS, Search Engine including Mobile Market Share, <http://gs.statcounter.com/#browser-CH-monthly-201103-201203>, letzter Zugriff 22.03.2012
- [28] Lightweight OpenID library. - Google Project Hosting, <http://code.google.com/p/lightopenid/downloads/list>, letzter Zugriff am 14.06.2012
- [29] PHP SDK Overview, <http://developers.facebook.com/docs/reference/php/>, letzter Zugriff am 14.06.2012
- [30] abraham/twitteroauth GitHub, <https://github.com/abraham/twitteroauth>, letzter Zugriff am 14.06.2012
- [31] posabsolute/jquery-Validation-Engine GitHub, <https://github.com/posabsolute/jquery-Validation-Engine>, letzter Zugriff am 14.06.2012
- [32] jQuery Countdown, <http://keith-wood.name/countdown.html>, letzter Zugriff am 14.06.2012
- [33] Google Chart Tools: Infographics, [https://developers.google.com/chart/infographics/docs/qr\\_codes](https://developers.google.com/chart/infographics/docs/qr_codes), letzter Zugriff am 14.06.2012
- [34] UPDATE: Send email with PHP and GMail hosted for your domain, <http://www.thedanglybits.com/2006/10/13/update-send-email-with-php-and-gmail-hosted-for-your-domain>, letzter Zugriff am 14.06.2012
- [35] Moodle.org: open-source community-based tools for learning, <http://moodle.org>, letzter Zugriff am 07.03.2012
- [36] GIFT format, <http://docs.moodle.org/22/en/GIFT>, letzter Zugriff am 07.03.2012

- [37] Hot Potatoes - Deutsche Adaption, <http://www.hotpotatoes.de/>, letzter Zugriff am 07.03.2012
- [38] Moodle XML format, [http://docs.moodle.org/22/en/Moodle\\_XML\\_format](http://docs.moodle.org/22/en/Moodle_XML_format), letzter Zugriff am 07.03.2012
- [39] Detlef Detailreich, <http://goo.gl/6d8Nx>, letzter Zugriff am 09.03.2012
- [40] Bernhard Begeisterungsfähig, <http://goo.gl/vo4eY>, letzter Zugriff am 09.03.2012
- [41] JavaScript Frameworks, <http://wappalyzer.com/categories/javascript-frameworks>, letzter Zugriff am 12.06.2012
- [42] Symfony2 PEAR channel, <http://pear.symfony.com/>, letzter Zugriff am 13.06.2012
- [43] Doctrine PEAR channel, <http://pear.doctrine-project.org/>, letzter Zugriff am 13.06.2012
- [44] Arbeiten Mit Dem PHP-ORM Doctrine 2, <http://www.mind2code.com/blog/2010/07/11/arbeiten-mit-dem-php-orm-doctrine-2/>, letzter Zugriff am 13.06.2012



## 13. Glossar

A	
Admin	Kurzform für Administrator
Android	Betriebssystem von Google für Smartphones
Apple iOS	Betriebssystem von Apple für Smartphones und Tablets

B	
Build	Erstellungsvorgang einer bestimmten Version einer Software
Bug	Fehler in der Anwendung

C	
Checkbox	Auswahlkasten bzw. Kontrollkästchen
Client	Computer der einen angebotenen Dienst beansprucht
Code	siehe Source Code
Code Review	siehe Review
CSS	Cascading Style Sheet - deklarative Sprache für Stilvorlagen von strukturierten Dokumenten

D	
Dokument Review	siehe Review
Drag-and-Drop	eine Methode zur Bedienung des GUI, bei dem Elemente gezogen und losgelassen werden können

E	
ERM	Entity Relationship Modell, für Datenmodellierung in Datenbanken

F	
Feature	Funktionalität

G	
Git	Eine Software zur verteilten Versionsverwaltung von Dateien
GUI	Grafische Benutzeroberfläche (Graphical User Interface)

H	
HTML	Hyper Text Markup Language ist eine textbasierte Auszeichnungssprache zur Strukturierung von Inhalten in Dokumenten

I	
IDE	Integrated development environment
Interface	Schnittstelle
Internet-anwendung	Anwendung die auf dem Internet läuft
Iteration	Wiederholung

J	
JavaScript	ist eine Skriptsprache, die hauptsächlich in Web-Browsern eingesetzt wird.

<b>M</b>	
Meeting	Englisch für Besprechung
Mobile	Englisch für mobil

<b>O</b>	
Online	Im Internet befindend

<b>P</b>	
PDF	Portable Document Format, Dateiformat
Plugin	Eigenständige Software, die eine andere um Funktionen erweitert
Poll	Englisch für Abstimmung
Python	Programmiersprache

<b>Q</b>	
QR-Code	Quick Response Code
Quiz	Englisch für Befragung

<b>R</b>	
Review	Manuelle Überprüfung von Arbeitsergebnissen

<b>S</b>	
Sheet	Englisch für Blatt
Smartphone	ein Mobiltelefon, das eine deutlich gesteigerte Computerfunktionalität und Konnektivität zur Verfügung stellt
Software	Programm
Software Engineering	Herstellung bzw. Entwicklung von Software
Source Code	Den für Menschen lesbaren, in einer Programmiersprache geschriebenen, Text eines Computerprogrammes

<b>T</b>	
Tablet	Ist ein mobiler Computer in Form eines Touchscreens
Templates	Vorlagen

<b>U</b>	
Upgrade	Aktualisierung, Höhere Stufung
URL	Uniform Resource Locator, identifizieren und lokalisieren eine Ressource.
User Interface	Open Systems Interconnection

<b>Z</b>	
Zeiterfassungstool	Programm zur Erfassung der Zeit

## 14. Bilderverzeichnis

0.1. Anwendung der Applikation . . . . .	7
0.2. Anwendung der Applikation . . . . .	9
1.1. Das Kontextdiagramm der Anwendung . . . . .	16
2.1. Tristate Toggle Button . . . . .	19
2.2. Beispiel der Punktevergabe . . . . .	19
2.3. Login via Google, Facebook oder Twitter . . . . .	20
2.4. Kennwortverschlüsselung bei Registrierung . . . . .	21
6.1. Symbole für den Status der Einträge. . . . .	31
6.2. Symbole für den Status der Durchführungen . . . . .	32
7.1. Einstiegspunkte Administration (links), Teilnahme (rechts) der Webanwen- dung . . . . .	39
7.2. Das Storyboard der User Interface . . . . .	40
8.1. Das Deployment Diagramm mit den unterschiedlichen Geräten . . . . .	41
8.2. Das Domainmodell der Problemdomäne . . . . .	42
8.3. Das Klassendiagramm der Anwendung . . . . .	44
8.4. Das Entity-Relationship-Diagramm der Anwendung . . . . .	47
8.5. Das Layer-Diagramm der Anwendung . . . . .	48
8.6. Ordnerstruktur des Projekts . . . . .	49
8.7. Navigationspfade der Lernkontrolle . . . . .	50
8.8. Navigationspfade der Umfrage . . . . .	51
8.9. Grundlegendes Zusammenspiel zwischen PHP- und TPL-Dateien im Code . . . . .	51
8.10. Grundlegendes Zusammenspiel zwischen PHP- und TPL-Dateien im GUI . . . . .	52
8.11. Zusammenspiel der Navigation . . . . .	52
8.12. Zusammenspiel der PHP- und TPL-Dateien in der Editierung . . . . .	53
8.13. Zusammenspiel der PHP- und TPL-Dateien in der Auswertung . . . . .	54
10.1. Aufbau der Umfrage . . . . .	64
10.2. Auswertung der Eingabemethoden-Umfrage als Kuchendiagramm . . . . .	65
10.3. Auswertung der Eingabemethoden-Umfrage als Balkendiagramm . . . . .	66
A.1. Anwendung der Applikation . . . . .	83
A.2. Use-Case-Diagramm der Anwendung . . . . .	86
B.1. Quiz & Poll Logo . . . . .	97
B.2. Quiz & Poll Plugin . . . . .	97
C.1. Tri-State Toggle Button . . . . .	106

C.2. Beispiel der Punkteverteilung . . . . .	106
C.3. Statistik über die Marktanteile der Browser vom 22.03.2012 . . . . .	108
E.1. Meilensteine der Bachelorarbeit . . . . .	112
E.2. Auswertung der Arbeitsstunden . . . . .	115
E.3. Übersicht der Phasen . . . . .	116
E.4. Gegenüberstellung der einzelnen Phasen in der IST- und SOLL-Ansicht . .	116
E.5. Soll- und Ist-Zeiten der einzelnen Iterationen . . . . .	117
E.6. Arbeitsaufwand der einzelnen Arbeitspakete . . . . .	118
E.7. Gegenüberstellung der einzelnen Kategorien . . . . .	119
F.1. Evaluation der JavaScript Frameworks . . . . .	121

## 15. Tabellenverzeichnis

10.1. Werte zu den Unit Tests . . . . .	61
10.2. Code Statistiken . . . . .	62
E.1. Die Iterationen in der Übersicht . . . . .	114
G.1. Die verfügbaren Annotations . . . . .	126
G.2. Wichtige Parameter . . . . .	127

# **Teil VII.**

## **Anhang**

## A. Vision

### A.1. Einführung

Dozierende sollen eine Anwendung erhalten, die sie in ihrem Lehrauftrag unterstützt. Diese Anwendung soll anhand von Umfragen und Lernkontrollen den Lernfortschritt der Studenten reflektieren. Somit können Dozierende jederzeit die Lernkurve der Studierenden überwachen und die Teilnehmenden können durch die Umfragen oder Lernkontrollen besser abschätzen, wie ihr Lernfortschritt aussieht.

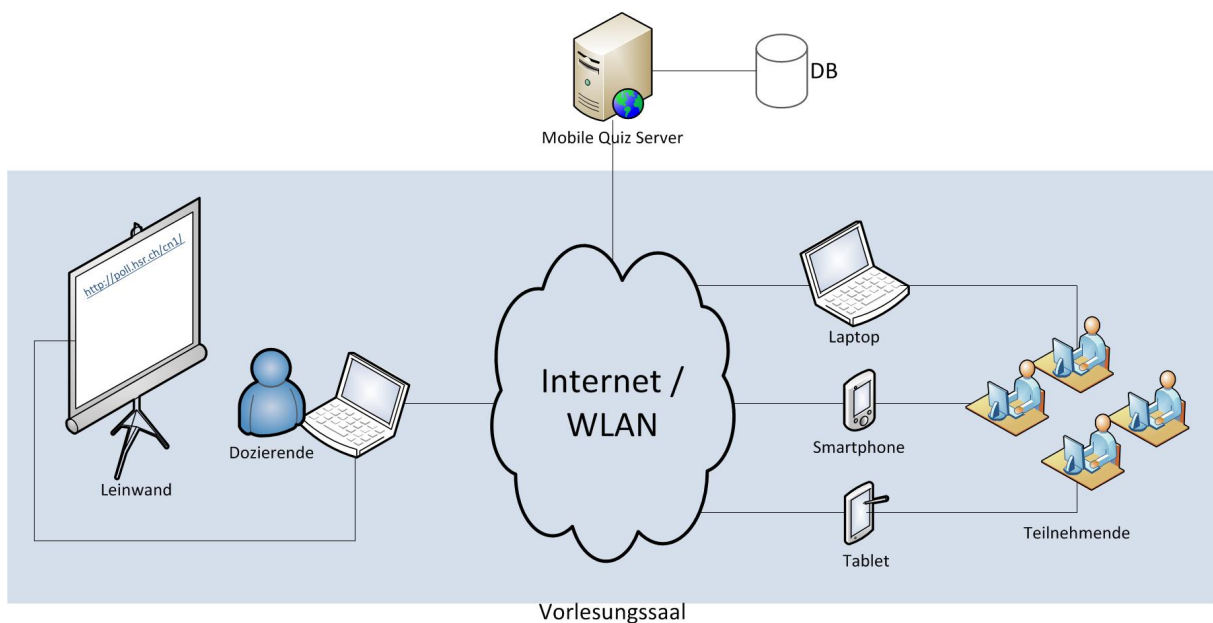


Abbildung A.1.: Anwendung der Applikation

### A.2. Positionierung

#### Geschäftsmöglichkeiten

Momentan können Dozierende während des Semesters nur mit grossem Aufwand, zum Beispiel mit schriftlichen Zwischenprüfungen, die Lernkurve der Studierenden überprüfen. Zudem haben die Teilnehmenden während des Semesters keine Möglichkeit ihren Wissensstand zu überprüfen und wissen nicht, wie sie im Vergleich zu ihren Kommilitonen stehen. Der Markt fordert ein Produkt, welches diese Lücken füllt.

### Problembeschreibung

Die traditionelle Unterrichtsmethode basiert darauf, dass Dozierende ihre Vorlesungen halten und die Teilnehmenden den Dozierenden zuhören. Eine Interaktion zwischen Dozierenden und Teilnehmenden findet selten statt. Ein Austausch ist schwierig zu erreichen bzw. aufrecht zu halten.

Die Ursache der fehlenden Interaktion während des Unterrichts kommt daher, dass sich die Teilnehmenden nicht blossstellen und ihre Meinung oder Ansichten offen preisgeben wollen.

Aus persönlicher Erfahrung ist es für einen Studenten unangenehm, Fragen während des Unterrichts zu beantworten. Dies, weil Studierende das Gefühl haben, dass sich eine falsche Antwort negativ auf die Notengebung oder die Reputation auswirken könnte.

### Aussage der Produktpositionierung

Die Anwendung ist für Dozierende gedacht, welche die Lernkurve der Teilnehmenden überprüfen und auf diesem Weg die Fortschritte verfolgen wollen. Zudem ist es das Ziel, Teilnehmende zum aktiven Mitmachen in der Vorlesung zu motivieren. Die Anwendung ermöglicht das einfache und schnelle Erstellen und Ausführen von Umfragen und Lernkontrollen, das Im-/Exportieren und die Analyse der Durchführungen. Die Umfragen können während des Unterrichts im Präsentationsmodus gestartet werden. Somit wird der Unterrichtsfluss nicht wesentlich beeinträchtigt. Die Anwendung ist klein, simpel und deckt die Bedürfnisse der Dozierenden ab.

### A.2.1. Alternativen und Wettbewerb

Es gibt einige Mitbewerber auf dem Markt, Kleine, Grosse und Alleskönner. Alle Plattformen haben eines gemeinsam: der Funktionsumfang ist entweder zu klein, oder zu schwerfällig. Es gibt kein Produkt, welches den Anforderungen der Dozierenden gerecht wird. Zu den Konkurrenten gehören beispielsweise:

- Quiz & Poll
- RationalSurvey
- FluidSurveys
- ClassMarker
- SurveyGizmo
- QuizStar
- Moodle

Der gesamte Funktionsumfang der Produkte kann in der Produkte-Matrix nachgeschlagen werden. Diese zeigt den Funktionsumfang der Konkurrenz, wie auch die Anforderungen der Dozierenden auf. Aus dieser Evaluation entstand die Liste der Funktionen für die Anforderungsspezifikationen.



### A.3. Allgemeine Schlüsselziele und Probleme der Dozierenden und Teilnehmenden

Allgemeines Ziel	Priorität	Probleme und Bedenken
Schnelle und robuste Abwicklung von Umfragen, bei bis zu 150 Teilnehmenden.	hoch	Erhöhte Antwortzeit, wenn die Anzahl Studenten zunimmt. Verringerte Verarbeitungsgeschwindigkeit, wenn die Anzahl paralleler Umfragen zunimmt. Bei grosser Anzahl Studenten ein Feedback in vertretbarer Zeit zu gewährleisten.
Einstellungen zu Umfragen verändern können.	hoch	Komplexe Darstellung der Einstellmöglichkeiten. Unklarheit welche Einstellungen standardmässig gelten. Ob Einstellungen für alle Umfragen gleich, oder nur pro Durchführung gelten.
Darstellung der Inhalte auf unterschiedlichen Geräten.	hoch	Die Lösung kann nur auf einer gewissen Gruppe von Geräten verwendet werden. Für jedes Gerät ist eine eigene grafische Darstellung nötig. Ein eventuell zum Einsatz kommendes Framework für Cross-Plattformen ist nicht performant genug.
Schnelle und klar verständliche Auswertung der Umfrage.	hoch	Verringerte Verarbeitungsgeschwindigkeit bei einer grossen Anzahl an Teilnehmenden. Wegen der Geschwindigkeit nicht verfügbar innerhalb des Unterrichts.
Exportieren von Umfragen.	hoch	Format des Exports. Soll der Datenaustausch zwischen unterschiedlichen Plattformen ermöglicht werden? Welche Formate sind gängig? Wie lange bewähren sich die einzelnen Formate? Welches sind die verwendeten Formate der grossen Plattformen?
Importieren von Umfragen.	hoch	Die Architektur der Datenhaltung muss die nächsten fünf Jahre beständig bleiben. Soll der Datenaustausch mit anderen Plattformen möglich sein?
Teilnehmende können sowohl anonym, wie auch identifizierbar antworten.	hoch	Teilnehmende vertrauen der Anwendung nicht. Dozierenden ist es unklar ob standardmässig anonym oder identifiziert beantwortet wird.

### A.4. Produktübersicht

#### A.4.1. Produktperspektive

Die Anwendung wird primär für Umfragen während des Präsenzunterrichts der Fachhochschule für Technik in Rapperswil angewendet, kann genauso für das Selbststudium eingesetzt werden. Die Anwendung ist webbasiert und somit von überall her, wo ein In-

ternetzugang besteht, erreichbar. Dozierende haben die Möglichkeit, die Umfragen und Lernkontrollen zuhause oder unterwegs zu erstellen und zu bearbeiten. Die Anwendung soll in Zukunft ebenfalls anderen Organisationen zur Verfügung gestellt werden. Das folgende Use-Case-Diagramm zeigt die Funktionen der Anwendung:

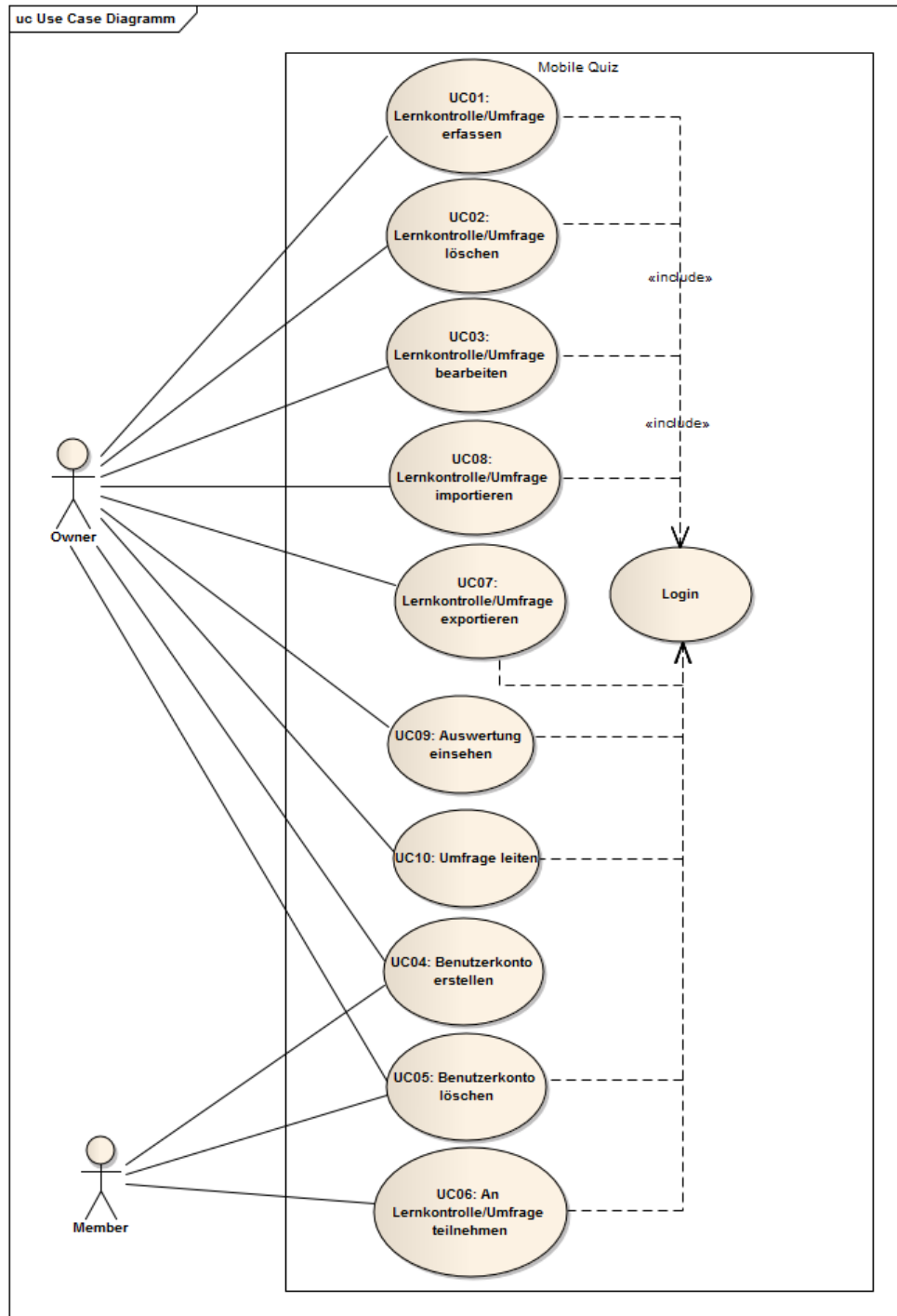


Abbildung A.2.: Use-Case-Diagramm der Anwendung

#### A.4.2. Zusammenfassung von Vorteilen

Funktion	Stakeholder-Nutzer
Lernkontrollen und Umfragen innerhalb des Unterrichts machen können.	Dozierende haben direktere Rückmeldung über die Lernkurve der Teilnehmenden und motivieren diese zur aktiven Teilnahme.
An einer Umfrage teilnehmen.	Teilnehmende können Rückschlüsse aus ihren Lernerfolg ziehen.

## B. Analyse der Mitbewerber

In erster Linie geht es darum, die Vielzahl der bereits bestehenden Internetanwendungen, welche auf den selben Anwendungszweck abzielen, zu analysieren und zu kategorisieren.

### B.1. Analyse

#### B.1.1. Rational Survey



##### Link

<http://www.rationalsurvey.com/>

##### Feature

Befragung/Umfrage (X) Prüfung () Statistik (X)

##### Preis

1. Free, \$ 0.00/Monat
2. Basic, \$ 20.00/Monat
3. Pro, \$ 30.00/Monat

##### Beschreibung

RationalSurvey ist eine Webanwendung für die Erstellung von Umfragen. Die Auswertung der Umfragen ist gesamthaft, wie auch auf einzelne Teilnehmende möglich. Eine Umfrage wird Element für Element zusammengesetzt. Die einzelnen Elemente stellen die unterschiedlichen Arten der Fragestellung dar. Diese können von Befragung zu Befragung variieren und damit wird automatisiertes Antworten verhindert. Die Fragen können in unterschiedliche Seiten gepackt werden. Somit wird eine Gliederung nach Thema möglich. Die jeweiligen Elemente können kopiert, bearbeitet, umgeordnet und gelöscht werden.

Die Einstellungen zu einer Befragung sind in folgende Kategorien unterteilt:

### Validierung

1. Müssen vor dem Ausfüllen der Befragung Bedingungen akzeptiert werden?
2. Muss die Frage zwingend beantwortet werden?

### Zeit

1. Zeitlimit für die Beantwortung der Fragen.
2. Periode, in der die Befragung ausgefüllt werden kann.

### Inhalt

1. Das Setzen eines Starttextes.
2. Das Setzen eines Endtextes.

**Präsentation** Es stehen viele Einstellmöglichkeiten für die Darstellung einer Befragung zur Verfügung, wobei nur einzelne Elemente im Design angepasst werden können.

Die Befragung kann in einer Vorschau begutachtet werden.

Die Analyse zeigt auf, wann wie viele Personen die Befragung ausgefüllt haben. Es ist ersichtlich, wie viele Befragungen abgeschlossen und wie viele Befragungen nicht abgeschlossen wurden. Die Auswertung kann über die gesamte Umfrage, wie auch auf einzelne Teilnehmende gemacht werden.

### Eindruck

RationalSurvey hat ein breites Spektrum an Funktionen und ist auf Anhieb einfach zu bedienen. Die Anwendung wirkt aufgeräumt. Die Funktionalität beschränkt sich jedoch lediglich auf die Befragung von Personen.

### Interessante Funktionen

1. Anpassbarkeit des Designs
2. Link zur Umfrage
3. Umfrage-Download als PDF
4. Exportmöglichkeit der Auswertung
5. Anzahl der Teilnehmenden, die Umfrage abgeschlossen haben
6. Anzahl Teilnehmenden, die Umfrage nicht abgeschlossen haben
7. Auswertung der einzelnen Teilnehmenden

## B.1.2. FluidSurveys



### Link

<http://www.fluidsurveys.com/>

### Feature

Befragung/Umfrage (X) Prüfung () Statistik (X)

### Preis

1. Free, \$ 0.00/Monat
2. Pro, \$ 19.00/Monat
3. Ultra, \$ 59.00/Monat
4. Enterprise, Preis auf Anfrage

### Beschreibung

FluidSurveys ist eine Internetanwendung um Umfragen zu erstellen. Bei der Erstellung von Umfragen wird man durch den Prozess begleitet. Es stehen verschiedene Fragetypen zur Auswahl, die per Drag-and-Drop auf den Fragebogen gezogen werden können. Jede dieser Fragen kann im Anschluss editiert werden. Dies reicht von einfachen Änderungen der Frage- und Antwortfelder, bis hin zu Validatoren, die man auf die Textboxen anwenden kann. Bei FluidSurveys werden Statistiken für die erstellten Umfragen möglich, welche auch auf einzelne Teilnehmende heruntergebrochen werden können.

**Präsentation** Es stehen viele Einstellmöglichkeiten für die Darstellung einer Befragung zur Verfügung, wobei nur einzelne Elemente im Design angepasst werden können.

Die Befragung, bzw. die Abstimmung, kann in einer Vorschau angeschaut werden, ohne dass die Statistik zu dieser Umfrage beeinflusst wird.

Die Analyse zeigt auf, wann wie viele Personen die Befragung ausgefüllt haben. Dabei wird ersichtlich, wie viele Befragungen abgeschlossen, bzw. wie viele Befragungen nicht abgeschlossen wurden. Die Auswertung kann über die gesamte Umfrage gemacht werden, so wie auf einzelne Teilnehmende.

### Eindruck

Durch die gut implementierte Drag-and-Drop-Funktion bietet FluidSurveys eine simple Handhabung. Das User Interface ist ansprechend und die Anwendung sieht organisiert und durchdacht aus. Leider stehen viele Features, die interessant wären, erst durch ein zahlungspflichtiges Upgrade zur Verfügung.

### Interessante Funktionen

1. Drag-and-Drop
2. Auswahl von Validatoren
3. Archivieren von Umfragen
4. Link zur Umfrage/Quiz
5. Exportmöglichkeit der Auswertung
6. Auswertung auf einzelne Teilnehmende
7. Anpassen der Umfrage

### B.1.3. SurveyGizmo



#### Link

<http://www.surveymozmo.com/>

#### Feature

Befragung/Umfrage (X) Prüfung (X) Formular (X) Statistik (X)

#### Preis

1. Professional Plan, \$ 49.00/Monat
2. Enterprise Plan, \$ 159.00/Monat
3. Dedicated Plan, \$ 7'888.00/Jahr

#### Beschreibung

Mit SurveyGizmo können Umfragen, Prüfungen, Formulare wie auch Abstimmungen erstellt werden. Jeder dieser Typen kann analysiert werden. Dabei variieren die Möglichkeiten der Analyse von Typ zu Typ. Bei der Erstellung der unterschiedlichen Typen wird ein Projekt erzeugt, welches als Rahmen für die Verwaltung dient. Es wird eine Art Wizard durchlaufen, der die möglichen Optionen anzeigt. Die Optionen ermöglichen das Erstellen und Kopieren eines Projekts oder die Verwendung eines Templates. Zudem werden unterschiedliche Designs zur Verfügung gestellt, wobei auch eigene Designs erstellt werden können. Bei der Zusammenstellung unterscheidet sich SurveyGizmo nicht wesentlich von anderen Produkten. Es können Elemente unterschiedlichster Art für die Fragestellung benutzt werden. Das Erfassen der Fragen und Antworten ist intuitiv gelöst und es werden vordefinierte Antwortmöglichkeiten (Zahlen von 1-100, alle Monate etc...) angeboten. Das Setzen der richtigen Antworten ist jedoch umständlich realisiert. Es müssen zuerst alle Fragen und Antworten erfasst werden, bevor die richtige Antwort angegeben werden kann. Dieses Vorgehen unterbricht den Fluss des Erfassens, bzw. ist fehleranfällig. Im Zusammenhang mit Prüfungen bietet SurveyGizmo eine interessante Funktion. Es kann bestimmt werden, wie viele Prozent der Antworten richtig sein müssen, um die Prüfung zu bestehen. Damit kann den Teilnehmenden das Resultat direkt nach der Prüfung präsentiert werden.

Die erfassten Fragen können in eine Bibliothek von Fragen aufgenommen werden, welche beim Erstellen von anderen Projekten wiederverwendet werden können. Damit lässt sich eine Sammlung von Fragen erstellen.

Die Auswertung der einzelnen Elemente wird grafisch, wie auch in Zahlenform, präsentiert. Es stehen viele unterschiedliche Arten der Auswertung zur Verfügung. Diese können



jederzeit aktualisiert und in den Formaten Word, Excel und PDF exportiert werden. Eine Auswertung auf einzelne Teilnehmende wird erst in der Beta-Version angeboten.

### **Eindruck**

Die Anwendung SurveyGizmo wirkt aufgeräumt und strukturiert. Das Design ist ansprechend und lädt ein, Umfragen oder Prüfungen zu erstellen. Die Bedienung ist nicht intuitiv. Oft sucht man in der Anwendung nach Funktionen. Es befremdet dass die Fragen und Antworten vorhanden sein muss, um in einem späteren Schritt die richtige Antwort zu definieren. Zudem ist die Anwendung langsam. Wartezeiten von einigen Sekunden sind eher die Regel als die Ausnahme.

### **Interessante Funktionen**

1. Umfangreiche Auswertung, verschiedene Möglichkeiten und Darstellungen
2. Kopieren von Projekten
3. Projekt-Templates erstellen
4. Prozentangabe für Prüfungserfolg
5. Eigene Definition der Kategorien
6. Unterschiedliche Designs
7. Erstellung eigener Designs

### B.1.4. ClassMarker



#### Link

<http://www.classmarker.com/>

#### Feature

Befragung/Umfrage (X) Prüfung (X) Statistik (X)

#### Preis

1. Free, \$ 0.00/Monat
2. Professional 1, \$ 16.50/Monat
3. Professional 2, \$ 33.00/Monat

#### Beschreibung

ClassMarker ist eine einfach zu bedienende Internetanwendung, die sich im Gegensatz zu SurveyMonkey oder FluidSurveys nicht auf Umfragen, sondern auf Tests konzentriert. Tests, die der Dozent mittels ClassMarker erstellt, können im Nachhinein kategorisiert werden. Dasselbe gilt für Benutzer. Es besteht die Möglichkeit, Benutzergruppen zu erstellen und zu verwalten. So kann beim Publizieren der Tests, eine Benutzergruppe angegeben werden, die den Test ausfüllen darf. Die Teilnehmenden müssen sich dabei auf der Plattform anmelden und werden bei einem anstehenden Test benachrichtigt.

#### Eindruck

ClassMarker ist professionell gestaltet und im Gegensatz zu FluidSurveys einfach gehalten. Die Bedienung ist gewöhnungsbedürftig, da beim Erstellen oder Editieren von Tests alles auf einer Seite dargestellt wird und man viele Menüs erst aufklappen muss. Die Free Version zeigt viele Funktionen auf, welche durch ClassMarker genutzt werden können. Die Auswertung gehört nicht zum Funktionsumfang der Free Version. Trotzdem überzeugt das Produkt durch seine Funktionalität.

#### Interessante Funktionen

1. Erstellung von Zertifikaten für einen Test
2. Kategorisieren von Tests
3. Erstellen von Benutzergruppen
4. Link zur Umfrage/Quiz
5. Import Templates für Fragen
6. Anpassen des Umfragedesigns

### B.1.5. QuizStar



#### Link

<http://quizstar.4teachers.org/>

#### Feature

Befragung/Umfrage (X) Prüfung (X) Statistik (X)

#### Preis

1. Free, \$ 0.00/Monat

#### Beschreibung

Ähnlich wie ClassMarker ist auch QuizStar auf Tests ausgelegt. Dozierende können Lektionen, Klassen und Tests erstellen. Diese Funktionalitäten erlauben es den Dozierenden, ganze Benutzergruppen für einen Test anzumelden. Studierende müssen bei QuizStar einen Login besitzen, um an Tests teilnehmen zu können. Dies hat den Vorteil, dass die Antworten der Studierenden während eines ganzen Semesters ausgewertet werden können.

#### Eindruck

Das User Interface von QuizStar ist veraltet und sieht auf den ersten Blick unprofessionell aus. Dies fällt im Vergleich zu den Funktionalitäten die es bietet nicht weiter ins Gewicht. QuizStar ermöglicht es, ein Zeitfenster festzulegen, während dem die Studierenden das Quiz ausfüllen können. Dabei können Dozierende dieses Zeitfenster auf eine Sekunde genau einstellen. Ein Anwendungszweck dieser Funktion wäre zum Beispiel, wenn Dozierende die Studierenden zu Beginn der Vorlesung als Repetition einige Fragen beantworten lassen. Im Gegensatz zu anderen Umfrage-/Quiz-Tools können bei QuizStar Bilder als Fragen hinzugefügt werden.

#### Interessante Funktionen

1. Lektionen erstellen
2. Klassen erstellen
3. Zeitfenster für Test festlegen
4. Bilder als Fragen
5. Auswertung der Studierenden über das ganze Semester
6. Lösung des Quiz/Test als PDF exportierbar

### B.1.6. Moodle



#### Link

<http://moodle.org/>

#### Feature

Befragung/Umfrage (X) Prüfung (X) Statistik (X)

#### Preis

1. Free, \$ 0.00/Monat

#### Beschreibung

Moodle ist ein Verwaltungssystem für Kurse. Um den Dozierenden und Studierenden zur Verfügung zu stehen muss es auf einem privaten Server installiert werden. Es handelt sich um eine Open Source Anwendung, die von jedem frei bezogen werden kann. Moodle ist eine umfassende Software, die es erlaubt, Dokumente auf die Plattform zu laden, sich auszutauschen und Kurse komplett online anzubieten.

#### Eindruck

Dadurch, dass Moodle eine Demo-Seite anbietet, konnten die verschiedenen Rollen, Administrator, Student und Dozent problemlos getestet werden. Jeder Eintrag des Verwaltungssystems kann nach Belieben editiert werden. Dadurch wirkt das User Interface überladen. Somit ist das Erstellen einer Lernkontrolle etwas schwierig. Es gibt viele Möglichkeiten, Einstellungen für das Quiz zu definieren. Dies reicht vom Zeitraum, in dem das Quiz von Studierenden ausgefüllt werden kann, bis hin zu Anzahl Versuche die sie dafür maximal aufwenden dürfen. Auch die Auswertung lässt sich einstellen. Zum Beispiel gibt es die Möglichkeit bei mehreren Versuchen, den besten, schlechtesten, den letzten Versuch oder den Durchschnitt der Versuche zu werten. Auch hier ist es möglich, Bilder hinzuzufügen.

#### Interessante Funktionen

1. Zeitfenster für Test festlegen
2. Bilder als Fragen
3. Auswertungsmöglichkeiten
4. Kategorien erstellen (Kurse)
5. Individuelle Punktevergabe
6. Abzüge bei falschen Antworten
7. Individuelles Feedback

## B.2. Detailanalyse

### B.2.1. Quiz & Poll

Eines der ersten Produkte, welches zu Beginn der Arbeit unter die Lupe genommen wurde, war Quiz & Poll[24]. Mit Quiz & Poll können Dozierende Lernkontrollen oder Umfragen mit Hilfe von Google Spreadsheets[25] erstellen.

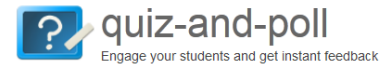


Abbildung B.1.: Quiz & Poll Logo

#### Funktionsweise

Die Funktionsweise des Plugins ist einfach. Benötigt wird ausschliesslich ein Google-Account, um ein neues Spreadsheet zu erstellen. Quiz & Poll stellt ein Template Quiz[26] bereit, welches kopiert und angepasst werden kann. Dabei muss beachtet werden, dass dem Dokumentnamen [Q] für Quiz oder [P] für Poll vorangestellt wird. In der ersten Spalte des Dokuments stehen die Fragen. In den nachfolgenden stehen die jeweiligen Antworten. Richtige Antworten werden mit einem \* am Ende gekennzeichnet. Falls nur eine Antwort korrekt ist, werden die Antworten als Radiobuttons (Single-Choice) dargestellt. Wenn mehrere Antworten korrekt sind, werden die Antworten als Checkboxes (Multiple-Choice) dargestellt. Sobald die Fragen und Antworten eingetragen sind, kann man das Dokument mit quiz-n-poll@appspot.gserviceaccount.com teilen. Dies wird benötigt, um das Quiz in der Endfassung zu erstellen. Unter dem Menüpunkt "Quiz Games", welcher auf Abbildung B.2 zu sehen ist, kann das Quiz oder die Umfrage via "Share with Students" den Studierenden per E-Mail oder QR-Code geschickt werden. Das Dokument besteht aus vier Sheets. Dem Questions-Sheet, in dem die Fragen und Antworten enthalten sind. Dem Information-Sheet, in dem die Informationen zum Quiz oder Poll eingetragen sind. Diese Informationen werden den Studierenden zu Beginn der Durchführung angezeigt. Im Leaderboard-Sheet sind die Punkte der Teilnehmenden enthalten, während im Statistics-Sheet eine genauere Auswertung der Antworten der Teilnehmenden dargestellt wird.

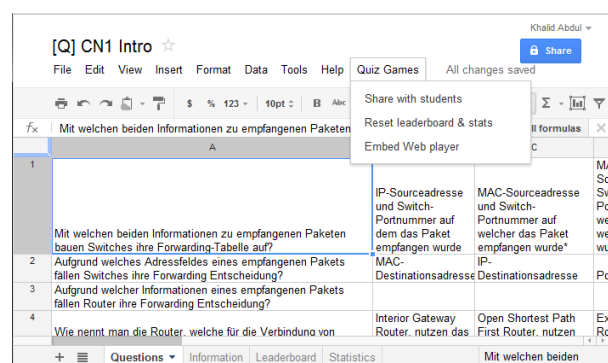


Abbildung B.2.: Quiz & Poll Plugin

Das Plugin bietet einige Features.

#### Features

- Einfaches Handling.
- Übersichtliches User Interface.
- Auswertung der Umfragen und der Quiz werden im selben Dokument dargestellt.

- **Android-Anwendung.** Für Teilnehmende ohne Android Smartphone steht ein Web Interface zur Verfügung, welches via Laptop, iPhone oder iPad aufgerufen werden kann.

### Mängel/Unschönheiten

Ausführlichen Tests ergaben kleinere Mängel. Zum einen fiel auf, dass beim Quiz maximal zehn Fragen gestellt werden, obwohl im Dokument unlimitiert viele erfasst sein können. Vom Plugin werden zehn Fragen zufällig ausgewählt, die dann gestellt werden. Zum anderen bleibt das Sheet mit den Statistiken manchmal leer, obwohl Teilnehmende den Fragebogen ausgefüllt haben. Dies scheint der Fall zu sein, wenn Umlaute wie ä, ö, ü in den Fragen enthalten sind. In den Antworten scheint dies kein Problem darzustellen.

### Erweiterbarkeit

Da der Quellcode des Plugins, welcher in Python geschrieben wurde, offen zugänglich ist, besteht die Möglichkeit, diesen mit neuen Funktionen zu erweitern. Quiz & Poll bietet eine begrenzte Vielzahl an Möglichkeiten, da es auf einem Spreadsheet basiert. Zum Beispiel stellt sich die Frage, wie Bilder in Fragen integriert werden können. Quiz & Poll bietet diesbezüglich nur auf der Begrüssungsseite die Einbindung eines Bildes an. Dies funktioniert nur über eine URL, was den Arbeitsfluss beim Erstellen eines Quiz behindert.

**Server** Auf der Wiki-Seite von Quiz & Poll wird Schritt für Schritt beschrieben, wie ein eigener Server in Betrieb genommen wird. Ausserdem wird erklärt, wie in der Android-Anwendung der Server anzugeben ist. Der Server muss direkt im Code angegeben werden. Damit hat man eine eigene App, welche mit dem eigenen Server interagiert. Zum Abschluss muss das Spreadsheet angepasst werden. Die Abhandlung in der Anleitung ist detailliert verfasst.

**Datenschutz und Datensicherheit** Da für die Verwendung von Quiz & Poll ein Google-Account nötig ist, wird garantiert, dass es sich um die angemeldete Person handelt. Es wird vorausgesetzt, dass Teilnehmende über einen Google-Account verfügen. Als Basis eines Quiz wird ein Spreadsheet vorausgesetzt, was ein Teil von Google Docs ist, und somit nur mit einem Account zugänglich ist. Daher wird der Datenschutz für die Quiz-Ergebnisse gewährleistet.

## B.3. Auswertung

Auch wenn eine Weiterentwicklung des Google Plugins verlockend gewesen wäre, da Google weltweit einen hohen Bekanntheitsgrad genießt, wurde entschieden, eine eigene Implementierung der Problemdomäne zu realisieren. Dies aus dem Grund, sich kleine Mängel und Unschönheiten zwar durch Erweiterungen entfernen lassen, jedoch ein grosser Teil der Anforderungen nur schwer realisiert werden könnte. Dadurch, dass die Anwendung auf Google Docs basiert, ist die Erfassung von Fragen und Antworten selbsterklärend und einfach handzuhaben. Es muss befürchtet werden, dass die Erweiterbarkeit durch diese Kopplung begrenzt ist. Einstellungen bezüglich der Umfragen oder Lernkontrollen liessen sich höchstens mittels Parameter in einem zusätzlichen Sheet verwirklichen. Die Anhäufung der Sheets würde die Übersichtlichkeit und Einfachheit der Anwendung verringern.

Zusätzlich wird bei Quiz & Poll mit Sonderzeichen, wie dem \*, gearbeitet, beispielsweise um Antwort als korrekt zu kennzeichnen. Dies könnte zu Problemen führen. Eine Funktionalität die für Dozierende interessant ist, ist das Einfügen von Bildern bei Fragen. Dies wäre ein weiterer schwer realisierbarer Punkt. Das Plugin erlaubt es momentan, Bilder auf der Startseite des Quiz anzuzeigen. Dieses muss vorher auf einen anderen Server geladen werden und die daraus resultierende URL wird in das Spreadsheet übernommen. Diese Arbeitsschritte wären bei Bildern in den Fragen vermutlich ebenfalls nötig, was einen Mehraufwand für die Dozierenden bedeuten und den Arbeitsfluss stören würde. Des Weiteren fehlen bei Quiz & Poll die graphischen Auswertungen zu den Quiz. Bei Umfragen sind sie im Präsentationsmodus hingegen vorhanden.

Ausserdem ist die Kopplung an Google zu streng, sodass sich jeder selbst bei anonymen Umfragen, mit seiner Google-Mail-Adresse anmelden muss, ohne dass dies einen Mehrwert für den Dozierenden darstellt. Dies ist für Dozierende nicht tragbar, da man von den Studierenden nicht erwarten kann, dass jeder einen Google-Account besitzt. Dies stellt zugleich eine Hemmschwelle für die Studierenden dar. Sie können sich nie sicher sein, ob die Umfrage wirklich anonymisiert stattfinden, wenn sie mit ihren persönlichen GMail-Adressen angemeldet sind. Diese Einschränkung lässt sich selbst bei einer Erweiterung des bestehenden Plugins nicht umgehen. Weiter kommt die Frage auf, ob der Datenschutz durch die enge Kopplung an Google gewährleistet ist, da nicht bekannt ist wo und wie die Daten abgelegt werden. Sowohl für Dozierende wie auch Studierende könnte dies ein weiterer problematischer Punkt darstellen.

In einer realen Testphase, in der Prof. Dr. Heinzmann Quiz & Poll für die Vorlesung Internetsicherheit eingesetzt hat, zeigten sich weitere Mängel der Anwendung. Einerseits gab es Probleme mit der Statistik, die durch die Verwendung von Umlauten in den Fragen ausblieb. Andererseits haben sich Teilnehmende darüber beschwert, dass der Browser auf ihren Smartphones, aus noch ungeklärten Gründen, abstürzte. Ein weiteres Defizit wurde zu einem späteren Zeitpunkt der Tests bekannt. Im Internet Explorer, also auch auf dem Windows Phone, wurden die Fragen zwar angezeigt, es bestand jedoch keine Möglichkeit eine Antwort zu geben. So erhielten die Studenten man beim Abschicken der Frage immer die volle Punktzahl.

Die anderen Anwendungen, die evaluiert wurden, haben aufgezeigt, dass es auch ohne Google Docs möglich ist, eine selbsterklärende und übersichtliche Anwendung zu entwickeln, welche die grundlegenden Bedürfnisse der breiten Masse erfüllt. Es gilt zu erwähnen, dass keine der getesteten Anwendungen den gesamten Funktionsumfang der Anforderungsspezifikationen abdeckt. Deshalb, und weil diese nicht erweiterbar sind, wird ihm Rahmen dieser Bachelorarbeit eine eigene Anwendung implementiert. Die daraus resultierende Funktionsliste ist im Kapitel 5. Funktionsliste zu finden.

## C. Anforderungsspezifikation

### C.1. Einführung

Dieses Kapitel dient dazu, eine Übersicht der Funktionen wie der Zielgruppen zu geben. Es erläutert, welche Funktionen zur Verfügung stehen und wie diese voneinander abhängig sind. Welche Bedingungen vor der Ausführung der Funktionen erfüllt sein müssen und was nach der Ausführung der Funktionen als Resultat vorliegt. Zudem werden die User Interface Elemente erfahrbar gemacht, die Zusammenhänge zwischen verschiedenen Elementen erklären.



## C.2. Allgemeine Beschreibung

### C.2.1. Personas

#### Dr. Detlef Detailreich, Informatik Dozent



Alter:	45
Funktion:	Dozent
Verpflichtungen:	Familie, drei Kindern
EDV-Kenntnisse:	sehr gut
Werdegang:	Informatik Studium, Dissertation

Detlef Detailreich ist Informatik Dozent an der Fachhochschule in Rapperswil und interessiert sich für Technik, Statistiken und Auswertungen.

Er hat eine schnelle Auffassungsgabe, was beim Erlernen und Verstehen von neuen Aufgabengebieten von Nutzen ist.

Da er ein Informatikstudium abgeschlossen hat, ist ihm der Computer vertraut und ist seine Spielwiese. Er verfügt über einen Laptop, sowie ein Smartphone. Zu Hause hat er eine Workstation. Detlef Detailreich trennt Privates vom Geschäftlichen.

Ihm liegt der Lernfortschritt seiner Studenten am Herzen. Daher versucht er seine Studenten zu animieren, am Unterricht aktiv teilzunehmen. Um dies zu erreichen, arbeitet er im Unterricht mit Fragen, die die Studenten für sich beantworten und allenfalls in Gruppen besprechen.

Für Detlef Detailreich muss eine Anwendung möglichst intuitiv und performant sein.

## Bernhard Begeisterungsfähig, Informatik Student



Alter:	22
Funktion:	Student
Verpflichtungen:	keine
EDV-Kenntnisse:	gut
Werdegang:	Sekundarschule, Berufsmaturität

Bernhard Begeisterungsfähig ist Informatik-Vollzeitstudent und hat vor seinem Studium eine Informatik-Lehre in der Firma Siemens Schweiz AG absolviert. Er interessiert sich für Mathematik und Action-Spiele.

Seine Auffassungsgabe ist schnell, was beim Erlernen von neuem Stoff im Studium nützlich ist.

Da sich Bernhard Begeisterungsfähig in der IT-Branche bewegt, ist ihm der Umgang mit Computern und technischen Systemen vertraut. Er tüftelt gerne und erforscht neue Technologien.

Er ist eher zurückhaltend und beteiligt sich mehrheitlich passiv im Unterricht. Für Bernhard Begeisterungsfähig ist es wichtig, die Zusammenhänge der einzelnen Themen zu verstehen.

Für Bernhard muss eine Anwendung funktional, strukturiert und nur aufs Wesentliche beschränkt sein.

Arbeiten am Computer bewerkstelligt er meist selber. Bei schwierigen Fällen versucht Bernhard Begeisterungsfähig das Problem mit Hilfe des Internets zu lösen.

### C.2.2. Benutzergruppen

Die Benutzergruppen für diese Anwendung beschränken sich auf Studenten/Schüler und Dozenten/Lehrer. Der Dozierende nimmt dabei die Rolle des Erstellers von Lernkontrollen und Umfragen ein, während der Studierende die Rolle des Teilnehmenden einnimmt.

### C.2.3. Szenarien

#### Day-In-The-Life-Szenarien

Day-In-The-Life-Szenarien sind Beschreibungen des Alltags eines fiktiven Nutzers mit dem fiktional bereichernden Produkt.

### Detlef Detailreich

Detlef Detailreich geht wie jeden Tag zur Fachhochschule um seine Vorlesung zu halten. Da Heute seine erste Vorlesung bereits um acht Uhr morgens stattfindet, geht er direkt in den Vorlesungssaal. Er richtet sich ein und bereitet die Umfrage zum Stoff der letzten Vorlesung vor. Nun hat die Glocke zum Unterricht geläutet und er begrüsst die Studierenden. Dabei fordert er die Studierenden auf, die Umfragen auf dem Gerät ihrer Wahl zu starten. Nach rund zehn Fragen schaut er die Antworten der Studierenden an und bespricht die Fragen mit ihnen. Dabei macht er sich Notizen, welcher Inhalt des Stoffes nochmals behandelt werden muss, bzw. was für die nächste Durchführung der Vorlesung geändert werden sollte.

### Bernhard Begeisterungsfähig

Am frühen Morgen steuert Bernhard Begeisterungsfähig Richtung Vorlesungssaal. Heute steht eine seiner Lieblingsvorlesungen auf dem Programm. Dies nicht nur, weil er den Stoff spannend findet, sondern weil der Dozent eine wöchentliche Lernkontrolle mit einer Umfrage-Anwendung durchführt. Bernhard nimmt im Vorlesungssaal Platz und bereitet sich für die Vorlesung vor. Er packt seine Notizen von der letzten Vorlesung aus und hält bereits sein Smartphone bereit. Die Vorlesung beginnt und der Dozent fordert die Studenten auf, die Umfrage auszufüllen. Bernhard Begeisterungsfähig füllt ebenfalls die Fragen aus. Dabei muss er oft grübeln und ist sich der Antworten nicht sicher. Nach rund sieben Minuten hat er alle Fragen beantwortet. Nun wartet er bis seine Kommilitonen die Fragen beantwortet haben. Nach rund zehn Minuten löst der Dozent die Fragen auf. Bei Fragen wendet sich Bernhard Begeisterungsfähig an einen Mitstudenten und notiert sich die Themengebiete seiner falschen Antworten. Bernhard liebt diese Vorgehensweise des Dozenten. Somit kann er jede Woche selbst überprüfen, wo er steht und kann so bereits unter dem Semester Lerndefizite ausfindig machen.

### Ist-Szenarien

**Detlef Detailreich** Detlef Detailreich geht wie jeden Tag zur Fachhochschule um seine Vorlesung zu halten. Da er heute seine erste Vorlesung um acht Uhr morgens hat, geht er direkt in den Vorlesungssaal. Unterwegs zum Saal ist er in Gedanken über die Studenten und seine Unterrichtsmethoden. Er fragt sich, ob alle Studenten mitkommen und ob der Unterricht optimal ausgelegt ist. Er denkt an seine Studienzeit zurück. Detlef richtet sich ein und wartet bis die Vorlesung beginnt. Die Studenten füllen allmählich den Saal und öffnen alle ihre Laptops. Die Studenten wirken unmotiviert und nicht bei der Sache. Die Tatsache, dass Fragen gestellt werden, ändert nichts daran.

**Der Assistent** Für Detlef Detailreich beginnt das neue Semester und die neue Vorlesung. Er hat sich während den Semesterferien Gedanken darüber gemacht, wie er die Übungen und die Vorlesung attraktiver gestalten kann. Er denkt sich, dass die Studenten mehr Freude am Lernen hätten, wenn er mit ihnen interaktiv zusammenarbeitet. Zusammen mit den Studenten will er im Unterricht einige Prüfungsfragen durchgehen. Da er einen fleissigen Assistenten hat, dem er solche Arbeiten abtreten kann, erklärt er ihm die Aufgabe. Dieser erledigt diese in kurzer Zeit und übergibt Detlef Detailreich die Folien mit den Fragen.

**Bernhard Begeisterungsfähig** Am frühen Morgen steuert Bernhard Richtung Vorlesungssaal. Heute steht eine seiner Lieblingsvorlesungen auf dem Programm. Er findet das Thema interessant, kann jedoch nicht einordnen, wie gut er den bereits behandelten Stoff beherrscht. Die Thematik scheint ihm komplex und er hat keine Anhang, wo er mit Lernen ansetzen soll. Bernhard nimmt im Vorlesungssaal Platz und bereitet sich für die Vorlesung vor. Er packt seine Notizen von der letzten Vorlesung aus. Als die Vorlesung beginnt, herrscht noch immer Hektik im Saal. Der Unterricht fängt nur schleppend an. Bernhard Begeisterungsfähig versucht motiviert bei der Vorlesung zu bleiben. Schliesslich schweift er mit seinen Gedanken ab und beginnt im Internet zu surfen. Die Zeit verfliegt und bis zum Vorlesungsende schnappt Bernhard nur noch kleine Teile der Vorlesung auf.

**Der Fragensteller** Bernhard Begeisterungsfähig beginnt die Woche mit der Vorlesung, die er am wenigsten mag. Da er denkt, dass der Vorlesungsstoff wichtig ist, geht er trotzdem hin. Der Dozent ist einer dieser permanenten Fragesteller. Auch wenn er zwei Minuten warten muss bis jemand aufstreckt oder er jemanden auswählen muss um die Frage zu beantworten, hört er nicht auf immer wieder Fragen zu stellen. Heute hat der Zug leider Verspätung und Bernhard kommt einige Minuten später in den Saal. Er merkt wie der Dozent ihn ins Auge gefasst hat. Einige Minuten vergehen und der Dozent stellt eine Frage in die Runde. Um die Aufmerksamkeit nicht auf sich zu lenken versteckt sich Bernhard hinter seinem aufgeklappten Laptop. Der Dozent, der sich Bernhard beim Eintreten in den Saal gemerkt hat, fragt ihn ihn dran. Bernhard Begeisterungsfähig zuckt zusammen und beantwortet die Frage falsch. Die Vorlesung geht weiter.

### C.2.4. Produktübersicht

#### Kernfunktionalität

- Dozierende müssen neue Lernkontrollen und Umfragen erstellen können.
- Dozierenden muss es möglich sein, Fragen hinzuzufügen, zu löschen und zu editieren.
- Dozierende müssen eine Lernkontrolle oder Umfrage mitsamt den Fragen und Antworten duplizieren können.
- Es muss möglich sein anzugeben, ob die Fragen bei der endgültigen Erstellung der Lernkontrolle oder Umfrage in eine zufällige Reihenfolge gebracht werden sollen.
- Den Dozierenden muss es möglich sein, eine oder mehrere korrekte Antworten anzugeben.
- Eingaben der Teilnehmenden müssen ausgewertet werden können.
- Es muss ersichtlich sein, wie viele Teilnehmende die Umfrage oder die Lernkontrolle ausgefüllt haben. Dies sowohl in absoluten Zahlen, wie prozentual.
- Die erstellte Umfrage muss als Link via E-Mail, oder als QR-Code verschickt werden können.
- Den Dozierenden muss es möglich sein, Fragen und Antworten in einem Excel vorzubereiten und zu importieren.

- Eine Vorschau der erstellten Lernkontrolle oder der Umfrage sollte für die Dozierenden zur Verfügung stehen.
- Es muss möglich sein, dass Dozierende angeben können, wann die Auflösung der Frage für die Teilnehmenden sichtbar ist. Dozierende können die Auflösung auf nach der Frage, am Ende der Lernkontrolle oder auf nie setzen.
- Es muss ersichtlich sein, wie lange die Teilnehmenden noch Zeit haben die Lernkontrolle auszufüllen.
- Teilnehmenden muss es möglich sein, die Fragestellung und Lösung, der teilgenommenen Umfragen und Lernkontrollen, zu beziehen.
- Die Lernkontrolle bzw. Umfrage muss auf Smartphones und auf Laptops ausfüllbar sein.
- Die Integration von Bildern muss möglich sein.

### Erweiterungen

- Quiz/Umfrage kann als PDF heruntergeladen werden.
- Internationalisierung des User Interface und der Fragen und Antworten.
- Zeit für die Fragen kann individuell angepasst werden.
- Zeit für den ganzen Fragebogen kann individuell angepasst werden.
- Visuelle Signale bei Ablauf der Zeit und eingehenden Antworten.
- Dozent sieht, wie viele Teilnehmer die Umfrage nicht abgeschlossen haben.

### C.2.5. Annahmen

Der Laptop des Dozierenden, sowie die Endgeräte der Teilnehmenden verfügen über eine Netzwerkverbindung.

### C.2.6. Abhängigkeiten

Die Anwendung hängt vom Faktor Internet ab. Ohne Internet ist es zwar möglich Fragen und Antworten in einem Excel vorzubereiten, jedoch ist es nicht, die definitive Lernkontrolle oder Umfrage zu erstellen oder durchzuführen.

## C.3. Funktionale Anforderungen

### C.3.1. Allgemeine Anforderungen

#### Punkteverteilung

Bei der Beantwortung von Fragen muss für die Teilnehmenden die Möglichkeit bestehen, keine Antwort abzugeben. Dies bedeutet, dass anstatt Checkboxes oder Radio-buttons ein Schalter angezeigt wird, der drei Status, richtig, falsch und neutral, darstellt. Die Punkte berechnen sich daraus, dass bei einer Multiple-Choice-Frage mit fünf Antworten, in der drei richtig sind und zwei falsch, minimal -5 Punkte und maximal 5 Punkte erzielt werden können. Wenn der Schalter auf neutral ist, enthält sich der Student der Stimme und erhält weder Plus- noch Minuspunkte.



Abbildung C.1.: Tri-State Toggle Button

	Richtige Antwort #1	+1		Richtige Antwort #1	+1		Richtige Antwort #1	+1
	Richtige Antwort #2	+1		Richtige Antwort #2	+1		Richtige Antwort #2	+1
	Richtige Antwort #3	+1		Richtige Antwort #3	-1		Richtige Antwort #3	+/- 0
	Falsche Antwort #1	+1		Falsche Antwort #1	+1		Falsche Antwort #1	+1
	Falsche Antwort #2	+1		Falsche Antwort #2	+1		Falsche Antwort #2	+1
5 Punkte			3 Punkte			4 Punkte		

Abbildung C.2.: Beispiel der Punkteverteilung

### C.3.2. Use Cases

Die detaillierten Use Cases sind unter [100\\_Requirements/Use\\_Cases.pdf](#) zu finden.

## C.4. Nichtfunktionale Anforderungen

### C.4.1. Leistungsanforderungen

Die Anwendung soll für die meisten Funktionen innerhalb von drei Sekunden ein Feedback zurück geben. Somit wird gewährleistet, dass der Anwender zu jeder Zeit weiss, ob eine Funktion ordnungsgemäss ausgeführt wurde, oder nicht. Diese Zeitgrenze soll eine Vereinheitlichung des Systemverhaltens bewirken.

### C.4.2. Mengenanforderungen

Im ersten Release von Mobile Quiz sollen zehn parallele Umfragen mit jeweils 150 Leuten ohne spürbare Verzögerungen durchgeführt werden können. Ausserdem sollen von zehn Modulen mit rund 150 Personen die Antworten der wöchentlichen Umfragen über zwei Semester hinweg gespeichert werden.

### C.4.3. Anforderungen an Schnittstellen

#### Hardware-Schnittstellen

Als Schnittstelle zur Hardware dient der Anwendung der Webserver. Der Prozessor des Systems muss dabei eine Leistung von mindestens 1.5 GHz erbringen. Das Minimum an Arbeitsspeicher muss 1 GB betragen und das System muss über 1 GB Speicher, sowie über eine permanente Internetverbindung verfügen.

### **Software-Schnittstellen**

Für die Anwendung Mobile Quiz ist kein Application Programming Interface vorgesehen. Daten werden anhand der Import-/Export-Funktion ausgetauscht.

### **Kommunikationsschnittstellen**

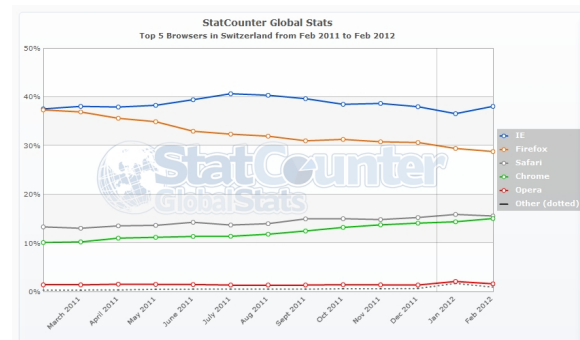
Um die Kommunikation zwischen den Dozenten bzw. den Teilnehmenden mit dem System zu erstellen, kommt das Hypertext-Transfer-Protokoll zur Anwendung.

### **Datenbank**

Es sind keine Anforderungen an das Datenbanksystem, bzw. an die Datenbankschnittstelle vorhanden.

### Benutzerschnittstelle

Als Benutzerschnittstelle wird den Benutzenden eine einheitliche Oberfläche in Form einer Webseite bereitgestellt. Diese bietet je nach Rolle unterschiedliche Funktionalitäten. Sie soll intuitiv sein, sodass keine Schulung der Anwender erforderlich ist. Zu den unterstützten Browsern gehört der Internet Explorer 9, Mozilla Firefox 12+ und Google Chrome 18+, die Statistik in Abbildung C.3 zeigt die meist genutzten Browser in der Schweiz. Auf den mobilen Geräten werden die Standardbrowser unterstützt.



**Abbildung C.3.:** Statistik über die Marktanteile der Browser vom 22.03.2012

## C.5. Qualitätsmerkmale

### C.5.1. Sicherheit

Die Webanwendung soll über ein Anmeldeformular zugänglich sein und Nichtregistrierten den Zutritt verwehren. Die Antworten zu den Lernkontrollen müssen vor unberechtigtem Zugriff geschützt werden. Für die Teilnehmenden muss der Datenschutz garantiert werden.

### C.5.2. Benutzbarkeit

Für Dozierende, die Lernkontrollen und Umfragen erstellen, lässt sich die Anwendung mittels Maus und Tastatur verwenden. Für die Teilnehmenden wird die Anwendung auf ihren mobilen Geräten verwendbar sein.

### C.5.3. Übertragbarkeit

Da die Anwendung auf einem Server im Internet läuft, ist diese von überall her zugänglich. Sie muss auf jedem Server installierbar sein, der den Mindestanforderungen gerecht wird.

### C.5.4. Zuverlässigkeit

Weil sich die Anwendung im Internet befindet, ist es wichtig, dass der Server eine hohe Verfügbarkeit garantiert.

## C.6. Andere Anforderungen

### C.6.1. Inbetriebnahme / Installation

Die Webanwendung wird während der Implementation und am Ende der Projektphase auf einen Server migriert.



### **C.6.2. Konfigurierbarkeit**

Mobile Quiz wird keinerlei Möglichkeit zur Konfiguration bieten.

### **C.6.3. Betrieb**

Die Anwendung läuft auf einem Server, auf den von überall her zugegriffen werden kann.

## D. Erfahrungsberichte

### D.1. Khalid Abdul

Die 16 Wochen der Bachelorarbeit sind schneller vorbeigegangen als zunächst gedacht. Ich blicke gerne auf diese interessante und lehrreiche Zeit zurück.

Zu Beginn der Bachelorarbeit war ich hoch motiviert. Ich habe mich persönlich schon immer für Webanwendung und die Gestaltung von User Interfaces interessiert. Da kam es gerade recht, dass wir uns um eine solche Bachelorarbeit beworben haben. Den Lehrinhalt, der uns in Modulen wie User Interface I und II beigebracht wurde, konnten wir gut für die Realisierung der Anwendung verwenden. Wir hatten beide schon von Anfang an gute Ideen, was die Umsetzung anbelangte. Diese haben wir zusammen mit unserem Betreuer in den ersten Phasen des Projekts ausgearbeitet und in der Anforderungsspezifikation festgehalten.

Wie sich herausstellte, blieb selbst nach Erstellung der Anforderungsspezifikation immer noch ein gewisser Spielraum für eigene Interpretationen. Dies machte sich dadurch bemerkbar, dass die Sitzungen immer länger dauerten als geplant. Trotzdem konnten wir in diesen wertvollen Stunden alle Unklarheiten beseitigen und uns schliesslich vollumfänglich der Implementierung widmen.

Gegenüber der Anwendung von TDD war ich zunächst skeptisch. Dies aus dem Grund, weil ich dachte, dass das Vorgehen mehr Zeit benötigt als eine normale Entwicklung. Mein Arbeitspartner hat mich jedoch schnell vom Gegenteil überzeugt und aufgezeigt, welchen Mehraufwand wir ohne die Anwendung von TDD hätten. Weiter habe ich in dieser Arbeit sehr gut aufgezeigt bekommen, wie wichtig es ist, sich frühzeitig mit dem Kunden abzusprechen und ihm die ersten Versionen der Anwendung vorzustellen. Als Entwickler ist der Ablauf der Anwendung nach einer gewissen Zeit klar und die Eingaben die man tätigt werden schnell zur Routine. Dabei vergisst man schnell, dass am Ende jemand die Anwendung bedient, der den Prozess der Entwicklung nicht durchgemacht hat und die Abläufe nicht in- und auswendig kennt.

Zusammenfassend kann ich sagen, dass die Erarbeitung des Projekts sehr lehrreich für mich war und ich mit unserer Leistung sehr zufrieden bin. Mit dieser Arbeit konnte ich viele Erfahrungen sammeln, die einen grossen Nutzen für meine zukünftigen beruflichen Laufbahn darstellen.

An dieser Stelle möchte ich mich bei Prof. Dr. Heinzmann für die gute Zusammenarbeit bedanken. Es hat Spass gemacht mit ihm zu arbeiten. Ausserdem möchte ich mich bei meinem Arbeitskollegen Patrik Naef für seine tolle Mitarbeit und für die gute Zeit, die wir während des Projektes hatten, bedanken.

### D.2. Patrik Naef

Die Zeit, in der wir an unserer Bachelorarbeit gearbeitet haben, ist bereits vorüber. Hiermit möchte ich meine Erfahrungen und Erkenntnisse kurz zusammenfassen.

Als wir die Aufgabenstellung von Prof. Dr. Peter Heinzmann erhielten, hatte ich zunächst viele Ideen. Zugleich war vieles unklar und ich konnte mir nicht wirklich vorstellen, wie unsere Arbeit schliesslich aussehen sollte. Selbst nach dem Kickoff-Meeting gab es noch einige Unklarheiten, welche wir in eigener Arbeit klärten und in der Anforderungsspezifikation festhielten. Das realitätsnahe Szenario hat mich motiviert an der Bachelorarbeit zu arbeiten. In meinen Augen ist dies eine gute Übung für den Einstieg ins Berufsleben. In Zukunft werde ich eine engere Zusammenarbeit mit dem Kunden suchen, um näher an die Anforderungen zu kommen. Die Kooperation mit dem Kunden ist elementar und entscheidend für den Erfolg eines Projekts. So hat uns Prof. Dr. Peter Heinzmann in den Sitzungen immer wieder Tipps und Ratschläge zum Vorgehen gegeben.

Schwierig an der Arbeit war es, die unterschiedlichen Ideen, Ansichten und Verständnisse auf einen gemeinsamen Nenner zu bringen. Schliesslich sollten die einzelnen Prozesse einfach sein. Die Analyse der Mitbewerber bot uns eine solide Basis des Funktionsumfangs. Die Bewertung von Funktionen und die Arten der Realisierung haben mein Verständnis für die Anwendung geschärft.

Die Anwendung von TDD hat mein Verständnis aus Sicht der Implementation erweitert. Viele Probleme oder nicht zu Ende gedachte Prozesse traten bereits beim Schreiben der Tests auf. Dies führte zu einer Risikominderung in früher Phase. Allgemein werde ich daher in Zukunft mehr Wert auf das Testing legen. Vor allem im Bereich der Benutzeroberfläche, bzw. der Tests durch Dritte. Auch hier findet sich ein weiterer Berührungspunkt mit der Kundenzusammenarbeit. Die Projektplanung nach RUP bot uns den Rahmen, all unsere Überlegungen zu vervollständigen und geordnet auf das Papier zu bringen.

Rückblickend auf die gesamte Bachelorarbeit kann ich sagen, dass ich in allen Belangen sehr zufrieden bin. Die Arbeit war anspruchsvoll und hat uns einiges abverlangt, doch sie war stets interessant und lehrreich. Dank der Arbeit habe ich neue Technologien und Konzepte kennengelernt, welche ich für meine zukünftige Tätigkeit als Softwareingenieur brauchen werde.

Ich möchte mich bei Prof. Dr. Peter Heinzmann herzlich für die vorbildliche Betreuung bedanken. Seine Vorschläge waren konstruktiv und er motivierte uns. Im Weiteren danke ich meinem Bachelorarbeitspartner Khalid Abdul für die ausgezeichnete Zusammenarbeit, die interessanten und anregenden Diskussionen und seine geleistete Arbeit.

## E. Projektmanagement

### E.1. Management Abläufe

#### E.1.1. Projekt Zeitaufwand

Der Aufwand für die Bachelorarbeit beträgt pro Student 360 Stunden. Somit liegt der Gesamtaufwand der Arbeit mit zwei, an der Arbeit beteiligten Studenten bei 720 Stunden. Die Bachelorarbeit startete am 20. Februar 2012 und dauerte 16 Wochen bis zum 15. Juni 2012. Der wöchentliche Arbeitsaufwand teilte sich in den 16 Wochen wie folgt auf:

- Die ersten 14 Wochen: pro Student 20 Stunden
- In den Wochen 15 und 16: jeweils 40 Stunden pro Student

Zwischen Woche sieben und acht liegen die Osterferien.

#### E.1.2. Projektplan

##### Meilensteine

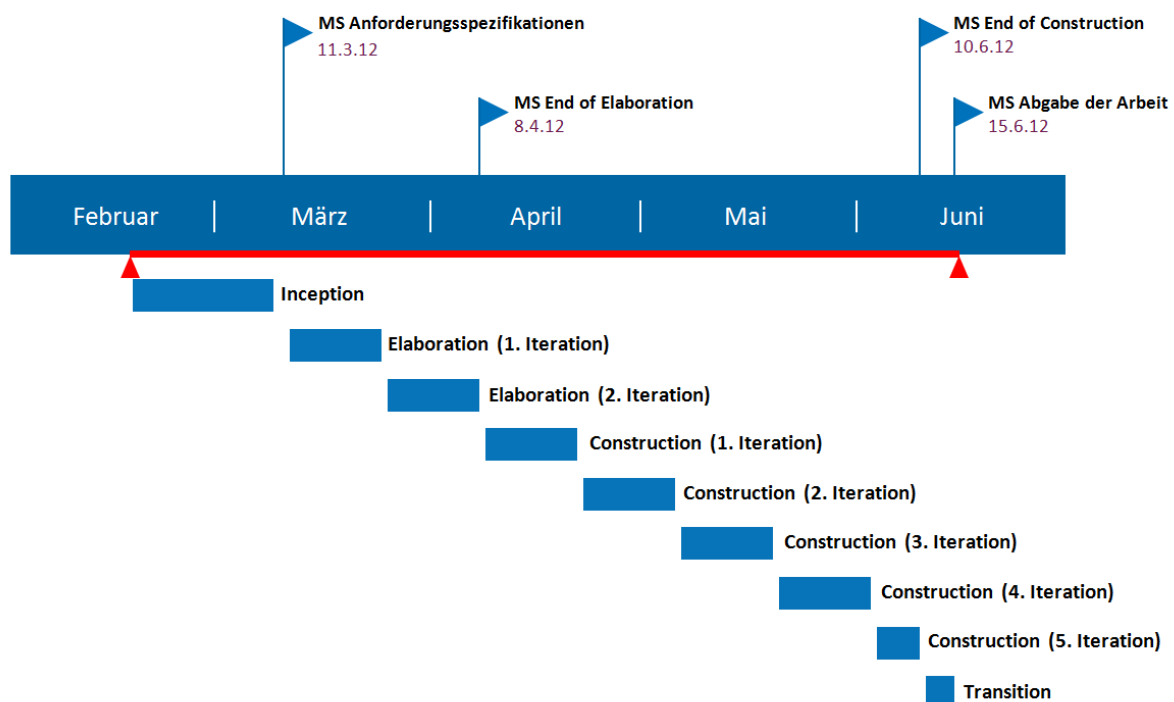


Abbildung E.1.: Meilensteine der Bachelorarbeit

In Abbildung E.1 sind die vier Meilensteine des Projektes zu sehen. Der erste Meilenstein beinhaltet die Anforderungsspezifikation und wird am Ende der Inception-Phase fertiggestellt. Ab diesem Zeitpunkt sollten die Anforderungen für Betreuer und Studierende klar sein. Der nächste Meilenstein ist auf Ende der Elaboration-Phase gesetzt. Dieser beinhaltet den konzeptionellen Prototypen, der dazu dient, die Machbarkeit des Projekts zu definieren. Zwischen dem Meilenstein der Elaboration und der, der Construction gibt es keine weiteren Meilensteine. Während dieser Zeitspanne gibt es zu jeder Iteration einen Release, der im Kapitel Releases E.1.2 genauer erläutert wird. Der letzte Meilenstein befindet sich am Schluss des Projektes und beinhaltet die abgabebereite Arbeit.

### Iterationsplanung

Im Rational Unified Process sind vier Phasen definiert. Die Phasen Inception, Elaboration, Construction und Transition werden in mehreren Iterationen durchlaufen. In der folgenden Tabelle wird eine Übersicht der einzelnen Iterationen mit einer Beschreibung in Stichworten aufgezeigt.

Iteration	Beschreibung	Ende	Dauer [W]
Inception 1	Kickoff-Meeting, Risikomanagement, Beginn Projektplan, Entwicklungsumgebung aufsetzen, Analyse der Mitbewerber, Umfrage bei Dozenten, Anforderungsspezifikation evaluieren	W03	3
Elaboration 1	Domainmodell, Paper Prototype, Projektplan erweitern, Datenmodell, Prototyp Server, Prototyp Client	W05	2
Elaboration 2	Projektplan erweitern, Doctrine 2 einrichten, PHPUnit einrichten, Smarty einrichten, Prototyp Server erweitern, Prototyp Client erweitern, Technologiestudium, Präsentation Prototyp	W07	2
Construction 1	Aufbau Software Architektur Server, GUI Entwurf Client, Zwischenpräsentation, technischer Bericht erweitern	W09	2
Construction 2	Features Lernkontrolle und Umfrage Server- und Clientseitig erweitern, Technologiestudium, technischer Bericht erweitern, Usability-Test durchführen, Review durchführen	W11	2
Construction 3	Auswertung Lernkontrolle und Umfrage erstellen, Previewmöglichkeit bei Umfragen erstellen, technischer Bericht erweitern, Usability Test durchführen, Review durchführen	W13	2
Construction 4	Features Lernkontrolle und Umfrage Server- und Clientseitig erweitern, technischer Bericht erweitern, Review durchführen	W15	2
Construction 5	Bug Fixing, Abstract erstellen, Management Summary erstellen, Qualitätssicherung erstellen	W16	1

Iteration	Beschreibung	Ende	Dauer [W]
Transition 1	Poster, Abstract verbessern, Management Summery verbessern, Qualitätssicherung erweitern, technischer Bericht fertigstellen	W17	1

**Tabelle E.1.:** Die Iterationen in der Übersicht

### Besprechungen

Es findet alle sieben bis 14 Tage ein Meeting mit dem Betreuer Prof. Dr. Peter Heinzmann im cnlab statt, um allfällige Fragen zu klären, den Projektstand zu besprechen oder Reviews durchzuführen.

### Releases

Im Kapitel 5. Funktionsliste ist ersichtlich, in welchem Release welche Funktionen implementiert werden sollen. Dabei dient Release 5 als Reserve, um Funktionen mit wichtiger Priorität fertigstellen zu können.

## E.2. Risikomanagement

Im Risikomanagement werden potentielle Risiken, die das Projekt bedrohen könnten, aufgelistet. Diese werden bewertet und analysiert, um ihr Ausmass einzudämmen. Das Risikomanagement ist unter [000\\_Projektmanagement/050\\_Risikomanagement.pdf](#) zu finden.

## E.3. Zeitauswertung

Der Arbeitsaufwand der gemäss Vorgabe durch die einzelnen Studenten geleistet werden muss, beträgt 360 Stunden. Diese verteilen sich über 16 Wochen, wobei in den letzten beiden Arbeitswochen jeweils 40 Stunden aufgewendet werden. Die Diagramme wurden aus dem Zeiterfassungstool Redmine extrahiert. Folgende Abbildung zeigt einen Überblick über die aufgewendeten Stunden pro Person.

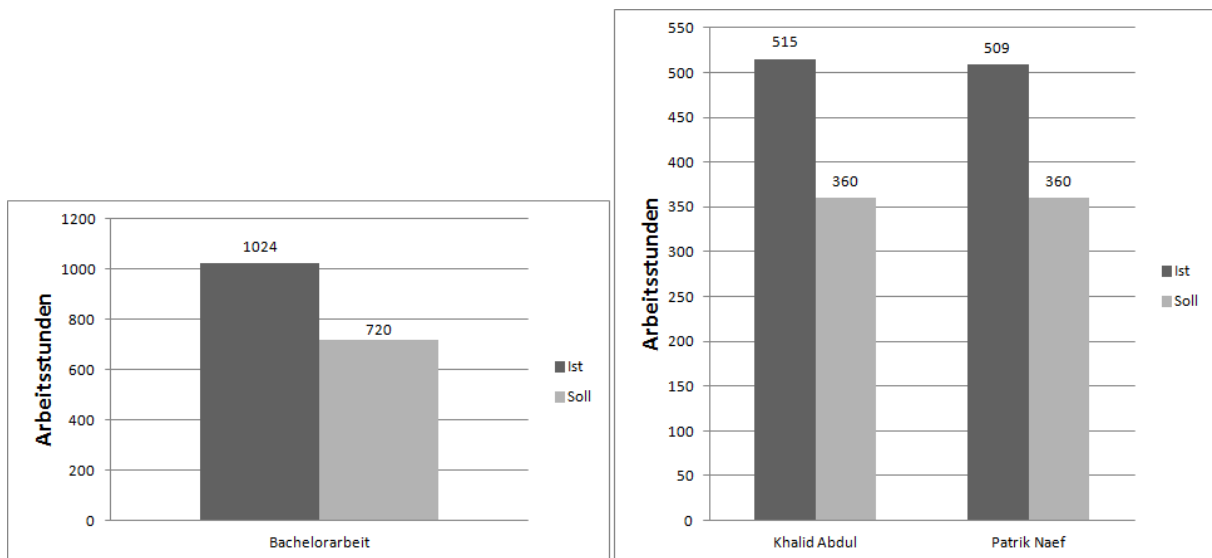


Abbildung E.2.: Auswertung der Arbeitsstunden

Auf die einzelnen Phasen aufgeteilt, sieht dies wie folgt aus:

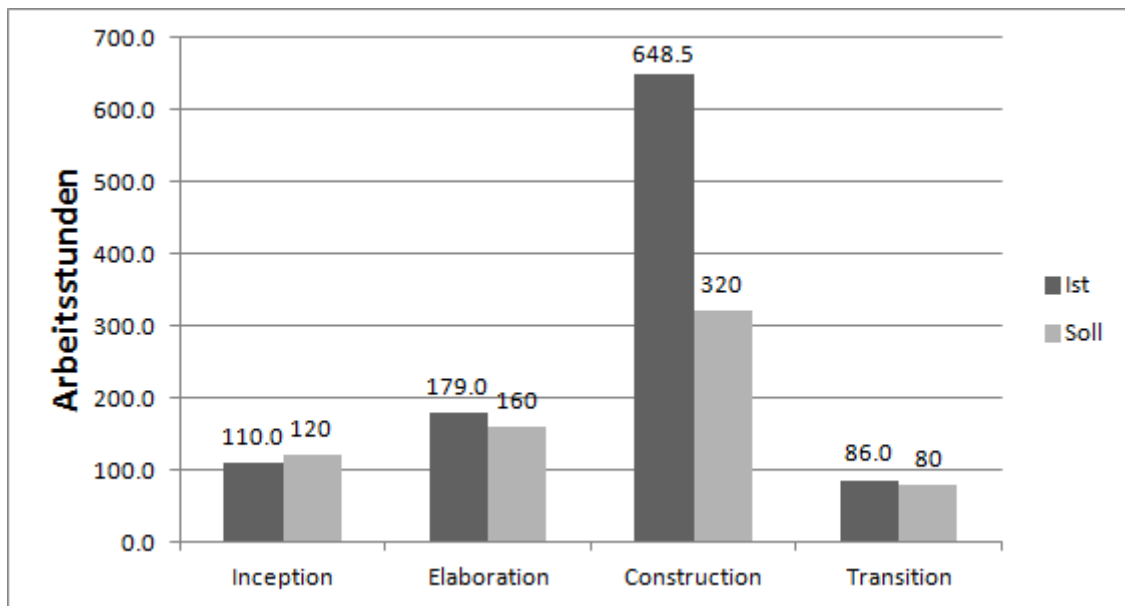


Abbildung E.3.: Übersicht der Phasen

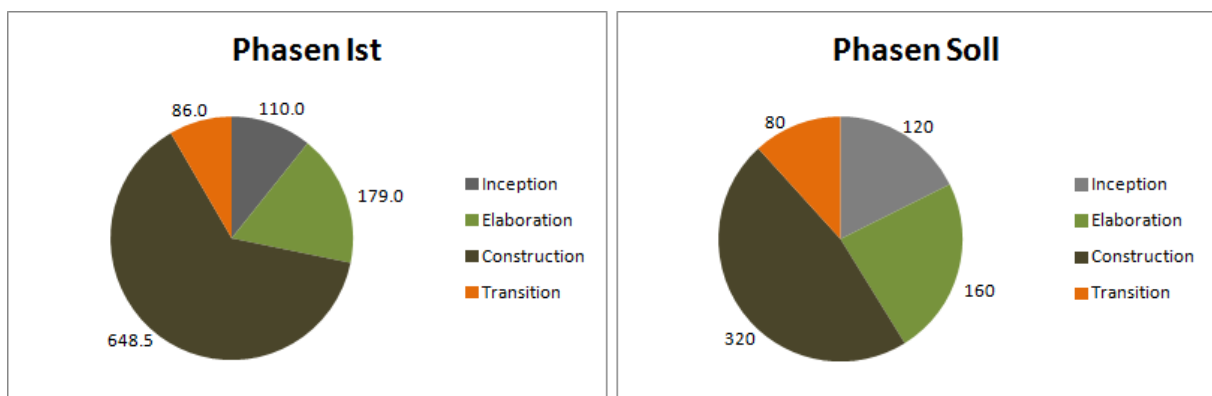
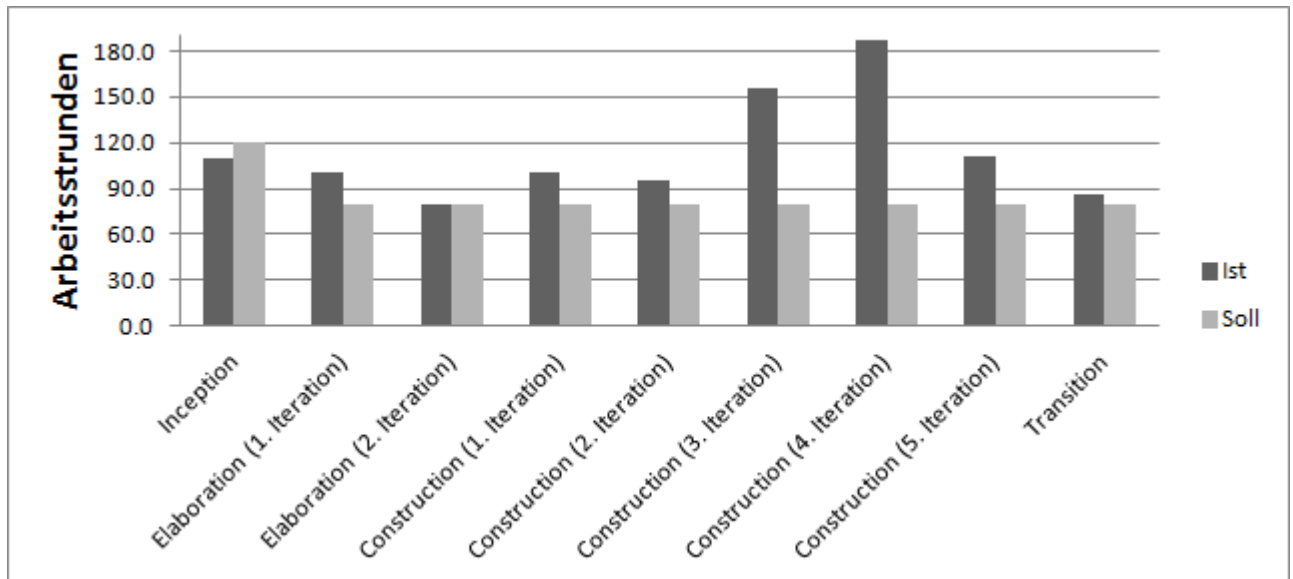


Abbildung E.4.: Gegenüberstellung der einzelnen Phasen in der IST- und SOLL-Ansicht

Die Abbildung E.5 zeigt die Aufteilung der Arbeitsstunden auf die jeweiligen Iterationen.





**Abbildung E.5.:** Soll- und Ist-Zeiten der einzelnen Iterationen

Auf folgender Abbildung ist die Aufteilung der Arbeitsstunden auf die einzelnen Arbeitspakete aufgeschlüsselt.

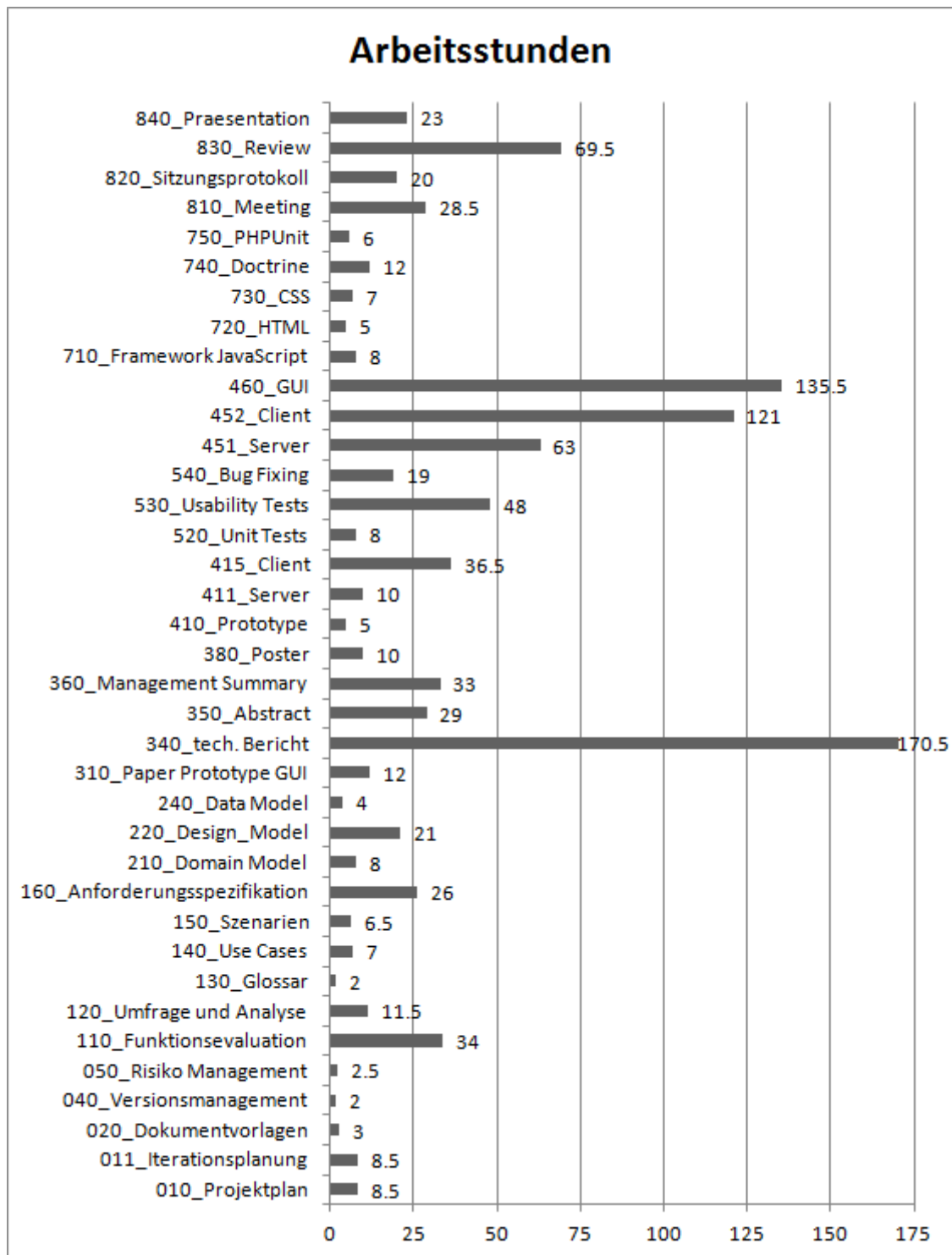


Abbildung E.6.: Arbeitsaufwand der einzelnen Arbeitspakete

Die Abbildung E.7 zeigt die Arbeitsstunden der einzelnen Kategorien.

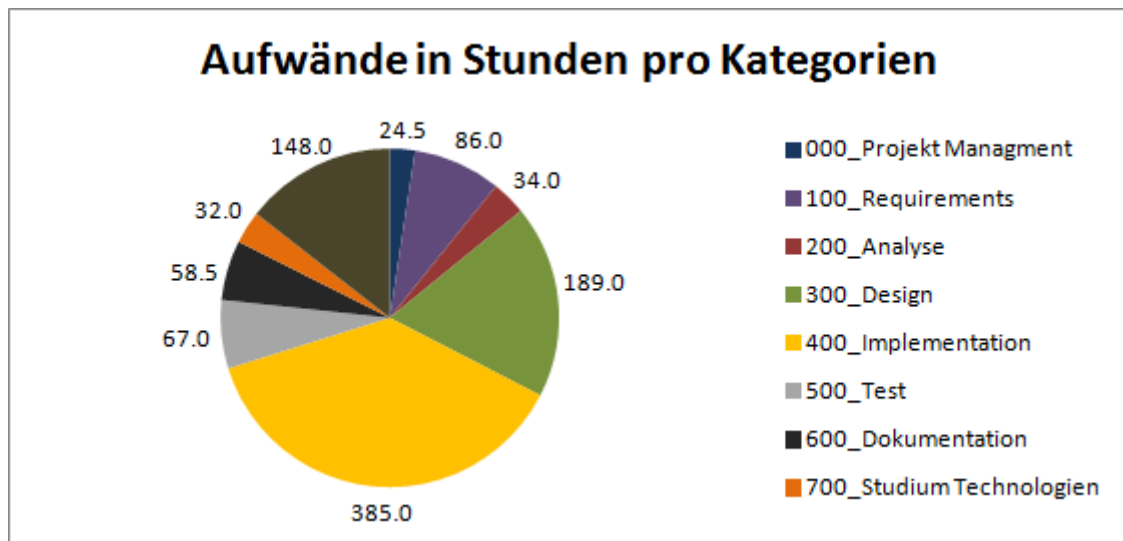


Abbildung E.7.: Gegenüberstellung der einzelnen Kategorien

### E.3.1. Zusammenfassung

In Abbildung E.4 ist zu sehen, dass die Phase der Construction mehr Zeit in Anspruch genommen hat, als zunächst geplant. Dies daher, da die Verwendung von Doctrine und PHPUnit mehr Einarbeitungszeit benötigte als gedacht. Weiter ist auf der Abbildung E.6 zu sehen, dass das Arbeitspaket des technischen Berichts mit Abstand am meisten Zeit benötigt hat. Wenn die Zeiten der Implementierung, die in Client-, Server- und GUI-Pakete aufgeteilt wurden, zusammenrechnet und dem Arbeitspaket des technischen Berichtes gegenüberstellt, ist wesentlich mehr Zeit für die Implementierung der Anwendung aufgewendet worden. Dies ist in Abbildung E.7 ersichtlich. Insgesamt wurden 300 Stunden mehr für das Projekt aufgewendet. Dieses Mehr an Stunden ist daraus entstanden, dass einzelnen Features mehr Zeit benötigten. Die Verlängerte die Zeit für die Implementierung. Dazu kommt die längere Einarbeitungsphase für PHPUnit und Doctrine.

## F. Werkzeuge und Tools

### F.1. Werkzeuge

#### F.1.1. IDE

Als IDE wurde Netbeans 7.1 eingesetzt. Dies daher, weil sich Netbeans gut für Webanwendungen eignet. Bevor Netbeans als Entwicklungsumgebung eingesetzt wurde, wurde Aptana Studio 3 erprobt. Um mit Aptana Studio eine Webanwendung zu realisieren bedarf es jedoch vielen Plugins und Einstellungen. Dies fällt bei der Anwendung von Netbeans weg. Netbeans bietet eine solide Basis zur Entwicklung von Webanwendungen. Beispielsweise ist eine Git-Anbindung bereits enthalten.

#### F.1.2. Frameworks und Libraries

In diesem Abschnitt werden die verschiedenen JavaScript Frameworks und Libraries verglichen. Im Vordergrund stand der Umfang der Funktionen, Performanz und die Anwendungsfreundlichkeit. Zudem war auch die Verbreitung[41] ein wichtiger Punkt für die Auswahl.

##### Analyse

**YUI (Yahoo! User Interface)** YUI[15] ist ein JavaScript Framework, welches von Mitarbeitern des Unternehmens Yahoo! entwickelt wurde. Dabei bietet das Yahoo! User Interface viele Möglichkeiten, um die Webseite zu manipulieren. Es wird beispielsweise eine erweiterte Drag-and-Drop-Funktion unterstützt oder auch das Erstellen von Diagrammen vereinfacht.

**MooTools (My Object-Oriented Tools)** Bei MooTools[16] handelt es sich, wie beim YUI, um ein Framework. Wie der ausgeschriebene Name schon sagt, ist es ein Objekt-orientiertes Framework. MooTools wird von einer breiten Masse genutzt, unter anderem auch auf Webseiten wie Joomla, Vimeo und Bing. Es bietet jedoch nicht so viele Hauptfunktionalitäten.

**jQuery** Die jQuery[17] Library ist eine der bekanntesten JavaScript Libraries. jQuery ist kostenlos und bietet verschiedene Möglichkeiten zur Manipulation der Website. Es ist die verbreitetste Library und wird von einer breiten Community unterstützt, die Plugins entwickelt.

## Auswertung

Priorität*	Bewertungspunkte	jQuery	MooTools	YUI
<b>Kenndaten</b>				
	Version	1.7.1	1.4.1	3.4
	Source language	JavaScript	JavaScript	Javascript, HTML, CSS
	Marktanteil*	62%	9%	3%
	Anzahl Websites	1'054'619	159'415	56'960
	Seitenzugriff	16'005'252	1'463'482	1'188'476
	Erweiterungen	Plugin jQuery Mobile jQuery UI	Plugin	Plugin
	License	MIT & GPL	MIT	BSD
<b>Unterstützte Browser</b>				
A	Internet Explorer	6+	6+	6+
A	Chrome	1+	1+	--
A	Mozilla Firefox	2+	2+	3+
<b>Feature</b>				
A	Feature detection	X	X	X
A	Event handling	X	X	X
A	JSON data retrieval	X	X	X
A	Server push data retrieval	X	O	via Plugin
B	XML data retrieval	X	X	O
B	HTML data retrieval	X	X	O
A	Drag and drop	X	X	X
C	Autocompletion tools	via Plugin	via Plugin	X
A	Rich text editor	via Plugin	X	X
B	Simple visual effects	X	X	X
B	advanced visual effects	X	X	X
B	Animation	X	X	X

Priorität    A - Für Projekt sehr relevant  
               B - Für Projekt eventuell relevant  
               C - Für Projekt nicht relevant

Abbildung F.1.: Evaluation der JavaScript Frameworks

Für die Bachelorarbeit wurde die JavaScript Bibliothek jQuery eingesetzt, weil jQuery den Studenten an der Hochschule für Technik Rapperswil im Modul "Internettechnologien" näher gebracht wird. jQuery bietet zusätzlich zur Standard Bibliothek, die Library jQuery UI[18] und das Framework jQuery Mobile[19] an. Die jQuery UI Library stellt verschiedene User Interface Komponenten zur Verfügung, während jQuery Mobile ein eigenständiges Framework darstellt, dass auf jQuery und jQuery UI aufbaut. Dieses Framework wird in der Anwendung auf der Seite der Teilnahme angewendet, da es verschiedene mobile Eingabegeräte unterstützt.

### F.1.3. Git

Git wurde zur Versionierung des Quellcodes und der Dokumente eingesetzt. Die Repositories wurden auf dem Server, der von der HSR zur Verfügung gestellt wurde, aufgesetzt.

### F.1.4. Redmine

Für die Zeiterfassung der Bachelorarbeit wurde Redmine eingesetzt. Dieses wurde auf dem Server der Schule bereits vorinstalliert und konnte ohne Weiteres genutzt werden. Während den einzelnen Iterationen wurde der Iterationsplan erstellt und die Arbeitspakete erfasst. Durch die Projektmitglieder wurden die Zeiten auf die einzelnen Arbeitspakete gebucht.

### F.1.5. Jenkins

Jenkins wurde für die Distribution auf den Server benötigt und um Code-Statistiken zu generieren. Dabei wurden folgende Plugins installiert:

- Checkstyle - um den Stil des Codes zu prüfen
- Clover PHP - für PHPUnit
- DRY - prüft auf "duplicated" Code
- HTML Publisher - um PHPUnit Code Coverage zu publizieren
- JDepend - liefert Statistiken zum Code bezüglich der Abhängigkeiten und der Komplexität
- Plot - liefert Statistiken zum Code bezüglich Anzahl Linien, Klassen, Interfaces etc.
- PMD - zeigt mögliche Fehler im Code und suboptimalen Code an
- Violations - generiert Auswertungen für Checkstyle, PMD, CPD etc.
- xUnit - um PHPUnit logfiles zu bearbeiten

### F.1.6. WinShell

Die Dokumentation wurde in Latex geschrieben. Um die Dateien zu bearbeiten, wurde die Anwendung WinShell eingesetzt. Zu Beginn der Arbeit wurden verschiedene Latex-Anwendungen erprobt, WinShell stellte sich schliesslich als beste Lösung heraus, da es möglich ist, Dateien in Projekte zu strukturieren und so für eine saubere Darstellung zu sorgen.

## F.2. Tools

### F.2.1. Smarty 3

Um die Trennung von Code und View zu garantieren, wurde die Template Engine Smarty 3 eingesetzt. Smarty ermöglicht es, zu einer View, einer sogenannten TPL-Datei, ein Code-Behind zu erstellen. Dabei werden im Code-Behind Variablen und Objekte deklariert, auf

die die View zugreifen kann. Dabei generiert Smarty zur Laufzeit eine “.tpl.php”-Datei, welche die View und den Code verflechtet. Die Trennung zwischen View und Code-Behind ist nötig, um die View zu einem späteren Zeitpunkt ohne Probleme austauschen zu können.

### **F.2.2. Doctrine 2**

Doctrine erlaubt die objektrelationale Abbildung, was die Handhabung der Datenbank vereinfacht. Durch Doctrine werden die einzelnen Klassen in der Datenbank abgebildet und Beziehungen zwischen den Klassen in der Datenbank reflektiert. Dies bedeutet, dass das Design der Datenbank durch die Klassen bestimmt wird.

### **F.2.3. PHPUnit**

Mit PHPUnit lassen sich Unit Tests für die PHP-Klassen schreiben. Durch die Tests kann zu einem späteren Zeitpunkt der Code überarbeitet werden und regelmässig auf seine Funktionstüchtigkeit überprüfen lassen. Die automatisierten Tests überprüfen die Resultate, zu bestimmten Übergabeparametern. Entsprechen die Resultate nicht den erwarteten Werten, schlägt der Test fehl.

## G. Einrichtung Produktivumgebung

Bevor Doctrine 2 installiert werden kann, müssen einige Pakete von Symfony2 installiert werden, weil Doctrine 2 von einigen Paketen abhängig ist.

### G.0.4. Installation Symfony2

Um die Installation vollziehen zu können, muss ein Kommandozeilenprogramm gestartet werden. Anschliessend muss in das PEAR Verzeichnis navigiert werden. Im PEAR Verzeichnis angelangt, kann mit der Installation der Symfony2<sup>[42]</sup> Pakete begonnen werden.

Listing G.1: Registrierung des Channels:

```
1 pear channel-discover pear.symfony.com
```

Listing G.2: Verfügbare Pakete anzeigen:

```
1 pear remote-list -c symfony2
```

Listing G.3: Paket installieren:

```
1 pear install symfony2/package_name
```

Listing G.4: Doctrine 2 benötigt die Pakete Console und Yaml.

```
1 pear install symfony2/Console
2 pear install symfony2/Yaml
```

Listing G.5: Erscheint die folgende Meldung, ist das Pakete bereits installiert.

```
1 Ignoring installed package symfony2/package_name
2 Nothing to install
```

### G.0.5. Installation Doctrine 2

Um Doctrine 2 zu installieren, muss man mit einem Kommandozeilenprogramm ins PEAR Verzeichnis wechseln. Im PEAR Verzeichnis angelangt, kann mit der Installation von Doctrine 2<sup>[43]</sup> begonnen werden.

Listing G.6: Registrierung des Channels:

```
1 pear channel-discover pear.doctrine-project.org
```

Listing G.7: Verfügbare Pakete anzeigen:

```
1 pear remote-list -c doctrine
```



Listing G.8: Paket installieren:

```
1 pear install doctrine/package_name
```

Listing G.9: Alle verfügbaren Pakete müssen installiert werden. Der PEAR Installer nimmt automatisch die aktuellste Version des angegebenen Paketes.

```
1 pear install doctrine/Doctrine
2 pear install doctrine/DoctrineCommon
3 pear install doctrine/DoctrineDBAL
4 pear install doctrine/DoctrineORM
5 pear install doctrine/DoctrineSymfonyConsole
6 pear install doctrine/DoctrineSymfonyYaml
```

Sollte ein Paket von einem anderen abhängig sein, welches sich zur Zeit der Installation nicht auf dem Webserver befindet, wird eine Fehlermeldung angezeigt. Diese gibt an, welches Paket fehlt und in welchem Channel es sich befindet.

### G.0.6. Konfiguration von Doctrine 2<sup>[44]</sup>

**Proxies** Dieses Verzeichnis wird von Doctrine 2 für die Generierung der Proxy-Klassen benötigt.

**setup.php** Hier wird das Datenbankschema erzeugt.

#### **config.inc.php**

Hier wird die Datenbank-Verbindung konfiguriert, es ist zugleich das zentrale Element von Doctrine 2.

Listing G.10: In diesen Zeilen werden benutzte Doctrine-Klassen in den aktuellen Namespace importiert.

```
1 use Doctrine\Common\ClassLoader ,
2     Doctrine\ORM\Configuration ,
3     Doctrine\ORM\EntityManager ,
4     Doctrine\Common\Cache\ApcCache;
```

Listing G.11: Mit der require-Anweisung wird die Autoloader-Klasse von Doctrine 2 geladen.

```
1 require '../.. / php/PEAR/Doctrine/Common/ClassLoader.php';
```

Listing G.12: In diesen Zeilen wird das Autoloading für die restlichen Klassen definiert. Autoloading bedeutet, dass Doctrine 2 alle Klassen automatisch bei Bedarf geladen werden. Jeder Namespace wird getrennt konfiguriert.

```
1 // Autoloading für Doctrine
2 $doctrineClassLoader = new ClassLoader('Doctrine', realpath(__DIR__ . '
    ../../php/PEAR'));
3 $doctrineClassLoader->register();
4 // Autoloading für unsere Modelle
5 $entitiesClassLoader = new ClassLoader('Models', __DIR__);
6 $entitiesClassLoader->register();
7 // Autoloading für die Doctrine-Proxy-Klassen
8 $proxiesClassLoader = new ClassLoader('Proxies', __DIR__);
9 $proxiesClassLoader->register();
```

Listing G.13: Hiermit wird angegeben, dass mit Annotationen die Entitäten beschrieben werden. Alternativ zu Annotationen kann die Entitätsstruktur mit einer XML- oder Yaml-Datei angegeben werden. In dieser Beschreibung wird nur die Annotations-Notation vorgeführt.

```
1 $driverImpl = $config->newDefaultAnnotationDriver(array(__DIR__ . "/Models"
  ));
2 $config->setMetadataDriverImpl($driverImpl);
```

Listing G.14: Angabe in welchem Verzeichnis und in welchem Namespace die Proxy-Modelle abgelegt werden sollen.

```
1 $config->setProxyDir(__DIR__ . '/Proxies');
2 $config->setProxyNamespace('Proxies');
```

Listing G.15: Damit wird die Datenbankverbindung beschrieben.

```
1 $connectionOptions = array(
2     'driver' => 'pdo_mysql',
3     'dbname' => 'doctrine2_test',
4     'host' => 'localhost',
5     'user' => 'root',
6     'password' => '',
7 );
```

### Annotations

- @Entity - Die Klasse soll von Doctrine als Entity verwaltet werden.
- @Table(name="personen") - optional, legt den Namen der zugehörigen Tabelle in der Datenbank fest. Ansonsten wird der Klassenname verwendet.
- @Id - Definition des Primärschlüssels.
- @Column(type="integer") - legt den Datentyp fest.

#### 1. Datentypen:

Name	Datenbank	PHP
string	VARCHAR	string
integer	INT	integer
smallint	SMALLINT	integer
bigint	BIGINT	integer
boolean	BOOLEAN oder INT - je nach Datenbank	boolean
decimal	DECIMAL	double
date	DATE	DateTime-Objekt
time	TIME	DateTime-Objekt
datetime	TIMESTAMP	DateTime-Objekt
text	CLOB	string

**Tabelle G.1.:** Die verfügbaren Annotations

#### 2. sonstige wichtige Parameter:

name	optional, legt den Namen der Spalte in der DB fest.
length	optional, bei Strings kann die Länge festgelegt werden.
unique	optional, legt fest, ob das Attribut UNIQUE ist.
nullable	optional, legt fest, ob das Attribut den Wert NULL annehmen darf.

**Tabelle G.2.:** Wichtige Parameter

- @GeneratedValue - Wert des Primärschlüssels soll automatisch erzeugt werden.

Listing G.16: Beispiel einer annotierten Klasse

```
1 /**
2  * @Entity
3  * @Table(name="adressen ")
4  */
5 class Adresse
6 {
7     /**
8      * @Id @Column(type="integer ")
9      * @GeneratedValue
10     */
11     private $id;
12     /** @Column(type="string ", length=150) */
13     private $strasse;
```

### Abbildung von Beziehungen

Mit Doctrine 2 können Datenbank-Beziehungen auf Model-Klassen abgebildet werden. Es stehen die Standard-Kardinalitäten 1:1, 1:n und n:m zur Verfügung.

Das folgende Beispiel, in dem eine Person mehrere Adressen haben darf, bildet eine 1:n-Beziehung ab:

Listing G.17: @OneToMany beschreibt die 1:n-Beziehung zwischen zwei Klassen. In unserem Fall zwischen der Klasse Person und Adresse. Dies bedeutet, dass das Objekt Person, n Objekte der Adresse enthält. Das Objekt Adresse wiederum enthält genau eine Person. Der Parameter targetEntity legt fest, auf welche Entität abgebildet wird. Mit mappedBy wird angegeben, wie das Objekt der aktuellen Klasse in der anderen heisst. In der Klasse Adresse heisst dieses Attribut \$person.

```
1 /** @OneToMany(targetEntity="Adresse ", mappedBy="person ") */
2 private $adressen;
```

Listing G.18: In der Klasse Adresse wird angegeben, dass \$person auf ein Objekt der Klasse Person, targetEntity="Person", verweist. Die aktuelle Klasse Adresse wird in Person über \$adressen, inversedBy="adressen" abgebildet.

```
1 /** @ManyToOne(targetEntity="Person ", inversedBy="adressen ") */
2 private $person;
```

### Datenbankschema erzeugen

Nachdem die Beziehungsinformationen in den Klassen angegeben wurden, kann die Datenbank angelegt werden.

Listing G.19: Mit dieser Zeile wird das SchemaTool erzeugt und die aktuelle Datenbankverbindung, dem EntityManager \$em übergeben. Anschliessend werden die Metadaten ausgelesen.

```
1 $schemaTool = new DoctrineORMToolsSchemaTool($em);
```

Listing G.20: Nach dem Auslesen der Metadaten werden diese dem SchemaTool übergeben.

```
1 $metadata = $em->getMetadataFactory()->getAllMetadata();
```

Listing G.21: Die Methode updateSchema() vergleicht den aktuellen Stand der Datenbank mit dem vorliegenden Schema und führt bei Bedarf CREATE/ALTER/DROP Anweisungen aus, um den Stand im Schema zu erreichen.

```
1 $schemaTool->updateSchema($metadata);
```

### Objekte persistieren

Die Persistierung von Objekten ist einfach gestaltet. Wenn ein Objekt in der Datenbank gespeichert werden soll, wird es der persist()-Methode des EntityManagers übergeben. Entsprechend dem Data-Mapper-Entwurfsmuster übernimmt der EntityManager das Speichern.

Listing G.22: Doctrine führt die Datenbank-Operationen nicht sofort aus, sondern erst am Ende eines PHP-Skriptes oder wenn die Methode flush() auf dem EntityManager aufgerufen wird. Sämtliche INSERT/UPDATE Anweisungen werden in einer einzigen Transaktion ausgeführt, was die Operation effizienter macht.

```
1 $per = new Person();
2 $per->setVorname('Hans')
3   ->setNachName('Muster')
4   ->setEMail('hans.muster@muster.ch');
5
6 $em->persist($per);
```

### Datenbank erstellen und konfigurieren

Nachdem alle Vorbereitungen getroffen wurden, können die Dateien und Ordner in das Webserver-Verzeichnis kopiert werden. Dabei muss darauf geachtet werden, dass der Webserver Lese- und Schreibrecht auf die Ordner uploads und den Unterordner temp besitzt. Dies aus dem Grund, weil in diesen Ordnern die Logos und die CSV-Dateien hochgeladen werden. Als nächstes muss die Datenbank und der Datenbank Benutzer erstellt werden. Dies lässt sich per Konsole bewerkstelligen. Dabei gibt man folgende Befehle ein:

Listing G.23: Datenbank und Benutzer erstellen

```
1 CREATE DATABASE mobilequiz;
2 CREATE USER 'username' IDENTIFIED BY 'yourpassword';
3 GRANT ALL PRIVILEGES ON 'mobilequiz'.* TO 'username'@\%';
```

Als nächstes müssen die Verbindungsparameter, Doctrine bekannt gemacht werden. In der Datei config-inc.php stehen im Array \$connectionOptions, wie in Listing G.15 festgehalten, die nötigen Informationen zur Verbindung. Dort müssen die einzelnen Attribute angepasst werden. Im ServiceLocator.php wird die Konfiguration für die Datenbankverbindung vollzogen. Diese Attribute müssen mit der Angaben der Datenbank übereinstimmen.