

TeachEval

Web App für die Unterrichtsevaluation

Bachelorarbeit

Abteilung Informatik
Hochschule für Technik Rapperswil

Frühjahrssemester 2014

Autor(en):	Dominique Sorg, Benjamin Kehl
Betreuer:	Prof. Dr. Eduard Glatz
Projektpartner:	OdA G Zürich
Experte:	Roberto Pajetta
Gegenleser:	Prof. Dr. Olaf Zimmermann

Diese Seite wurde absichtlich leer gelassen

Aufgabenstellung zur Bachelorarbeit FS 2014 „TeachEval – Web App für die Unterrichtsevaluation (1)“

Gruppe: Benjamin Kehl / Dominique Sorg

Ausgangssituation

Die OdA G Zürich (<http://www.oda-g-zh.ch/>) ist der Branchenverband für Berufsbildung im Gesundheitswesen des Kantons Zürich. Die OdA G führt unter anderem Kurse im Lernbereich Training & Transfer (LTT) Praxis durch, die systematisch sowohl von den Lernenden und Dozierenden wie auch deren Supervisoren evaluiert werden. Zurzeit erfolgen diese Evaluationen mit Hilfe von gedruckten Formularen, die in manueller Arbeit aufwendig ausgewertet werden müssen.

Aufgabe

Ihre Aufgabe besteht darin eine Webapplikation zu entwickeln, die sowohl die Befragung als auch deren Auswertung automatisiert. Das Beantworten der Fragebögen soll mit Hilfe eines Web-Browsers möglich sein, der eine entsprechende Eingabemöglichkeit in benutzerfreundlicher Art sowohl auf PCs, Tablets als auch Smartphones anbietet.

Zudem besteht ein zweites Hauptziel der Arbeit darin, ansprechende Auswertungen zu realisieren, die auch Diagramme enthalten und als Excel- oder MS Word-Dateien bereitgestellt werden. Der Auswertungsteil der Applikation soll zudem die Möglichkeit bieten, die Antworten auf ausgewählte Fragen zu korrelieren, d.h. die Kovarianz numerischer Auswertungsergebnisse zu berechnen.

Die Webanwendung soll in JavaScript, HTML 5 und CSS erstellt werden. Benutzte Drittkomponenten (z.B. Datenbank) und Tools sollen als OSS vorliegen.

Dieses Thema bietet die vielversprechende Möglichkeit eine praktische Lösung für den Einsatz beim Kunden zu erarbeiten. Die Aufgabe wird parallel in zwei Gruppen in Konkurrenz erarbeitet.

Allgemeine Vorgaben

Die Arbeit ist gemäss den allgemeinen Vorgaben [1] durchzuführen. Dies beinhaltet auch Vorgaben zur Berichtsgestaltung.

Termine

17. Feb. 2014	Arbeitsbeginn
13. Juni 2014	Abgabe des Berichts an den Betreuer (bis 12:00)

Weitere Termine siehe Terminangaben auf dem HSR-Web (intern) bzw. nach Absprache, insbesondere Zwischenbesprechungen mit Experte und Gegenleser sowie für mündliche BASchlussprüfung.

Betreuung

Betreuer: Prof. Dr. Eduard Glatz, Email: eglatz@hsr.ch

Während der Durchführung der Arbeit findet nach Möglichkeit regelmässig jede Woche eine Besprechung mit dem Betreuer statt. Dazu werden entsprechende Termine bei Arbeitsbeginn festgelegt.

Referenzen

[1] Glatz, E., „Vorgaben zur Arbeitsdurchführung“, Ausgabe des 16. Februar 2014.

Rapperswil, den 17. Feb. 2014



Eduard Glatz

Erklärung

Wir erklären hiermit,

- dass wir die vorliegende Arbeit selber und ohne fremde Hilfe durchgeführt haben, ausser derjenigen, welche explizit in der Aufgabenstellung erwähnt ist oder mit dem Betreuer schriftlich vereinbart wurde,
- dass wir sämtliche verwendeten Quellen erwähnt und gemäss gängigen wissenschaftlichen Zitierregeln korrekt angegeben haben.
- dass wir keine durch Copyright geschützten Materialien (z.B. Bilder) in dieser Arbeit in unerlaubter Weise genutzt haben.

Rapperswil, 07.06.2014



Benjamin Kehl



Dominique Sorg

Abstract

Die Oda G Zürich ist der Branchenverband für Berufsbildung im Gesundheitswesen im Kanton Zürich. Sie ist der Ansprechpartner und aktive Mitgestalter in allen berufspolitischen Bildungsfragen. Die Oda G bildet in Form von Kursen und Weiterbildungen am Kurszentrum Zürich-Oerlikon angehende Fachfrauen und Fachmänner Gesundheit (FAGE), Assistentinnen und Assistenten Gesundheit und Soziales (AGS) und diplomierte Pflegefachfrauen und diplomierte Pflegefachmänner aus. Damit die Oda G Zürich ein qualitativ hochwertiges Schulungsangebot anbieten kann, ist das Qualitätsmanagement ein wichtiger Bestandteil. Die Qualität wird mit schriftlichen Feedbacks von Studierenden, Dozenten und Supervisoren fortlaufend überprüft, was in manueller Arbeit aufwändig ausgewertet werden muss.

In dieser Bachelorarbeit wurde eine Webapplikation namens TeachEval entwickelt, die sich vor allem darauf konzentriert, die Qualität der angebotenen Kurse der Oda G Zürich sicherzustellen. Der momentane Workflow für die Befragung und deren Auswertung wurde in die Applikation integriert und automatisiert. Die Teilnahme an einer Befragung ist mit Hilfe eines Web Browser möglich, dessen Bedienung sowohl auf PCs, Tablets als auch auf Smartphones benutzerfreundlich ist. Aus den Befragungen können ansprechende Auswertungen in Form von Diagrammen realisiert und anschliessend als Excel-Download bereitgestellt werden. Zudem können die Antworten auf ausgewählte Fragen miteinander korrelieren, d.h. die Kovarianz numerischer Auswertungsergebnisse berechnet werden.

Als Ergebnis entstand eine Web Applikation mit Java Jersey und AngularJS. Mit dem Open Source Framework Jersey lässt sich eine serviceorientierte Architektur basierend auf den REST Prinzipien entwickeln. Für die Persistierung der Daten wurde Hibernate als ORM-Framework für Java eingesetzt. Auf der Client Seite konnte mit AngularJS eine moderne, dynamische und einfach zu bedienende Web Applikation realisiert werden. TeachEval wurde bereits Anfang Mai von der Oda G Zürich produktiv eingesetzt. Durch die enge Integration von Microsoft Excel mit Apache POI hat die Bereichsleiterin die Möglichkeit ihre Umfragen und Auswertungen mit Excel zu importieren oder zu exportieren. Ein eingebundener CronJob überprüft laufend, ob Umfragen gestartet, beendet oder ob die Teilnehmer durch eine E-Mail benachrichtigt werden müssen. Ein selbstentwickeltes VBA Makro generiert dynamisch in Excel die Diagramme und Korrelation zu den vorhandenen Rohdaten.

TeachEval

Web App für die Unterrichtsevaluation

Herzlich Willkommen bei TeachEval

TeachEval ist eine Webapplikation für die Erstellung von Umfragen und Auswertungen, die speziell für die Bildungsveranstaltungen der Oda G ZH im Bereich Gesundheit gew.
Umfrage der Oda G Zürich teilnehmen zu können, wird eine Umfragekennung, die Sie per Post, per Email oder persönlich erhalten haben, benötigt. Die Umfragen werden Anonym durch
Teilnehmer von der LTT Praxis Psychiatrie verwenden bitte den folgenden Link "Zur Umfrage".
Weitere Informationen finden Sie direkt auf der Oda G ZH Seite.

LTT Praxis Allgemein



LTT Praxis Psychiatrie



Management Summary (MAS)

Autor(en):	Dominique Sorg, Benjamin Kehl
Betreuer:	Prof. Dr. Eduard Glatz
Projektpartner:	OdA G Zürich
Experte:	Roberto Pajetta
Gegenleser:	Prof. Dr. Olaf Zimmermann

Ausgangslage / Problemstellung

Die Organisation der Arbeitswelt Gesundheit (OdA G) ist der Branchenverband für Berufsbildung im Gesundheitswesen des Kantons Zürich. Die OdA G ZH ist der Ansprechpartner und aktive Mitgestalter in allen berufspolitischen Bildungsfragen. Sie führt unter anderem Kurse im Lernbereich Training und Transfer Praxis (LTT Praxis) für die diplomierte Pflegefachfrauen und Pflegefachmänner durch. Damit die OdA G ZH ein qualitativ hochwertiges Schulungsangebot anbieten kann, ist das Qualitätsmanagement ein wichtiger Bestandteil (OdA G ZH Offizielle Webseite, 2014). In der folgenden Abbildung soll der Workflow besser illustriert werden:

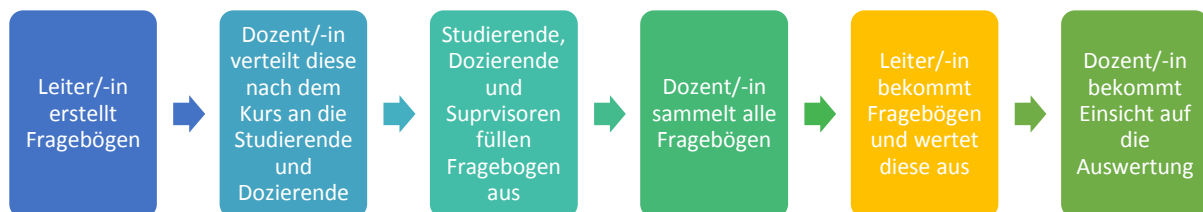


Abbildung 1 IST Situation

Die Qualität wird in Form von gedruckten Fragebögen von Studenten, Dozenten und Supervisoren nach jedem Kurs überprüft, die in manueller Arbeit aufwändig (gelber Kasten) ausgewertet werden muss.

Vorgehen / Technologien

In einer ersten Phase wurde mit Hilfe eines Fragebogens die Anforderungen der OdA G ZH abgeklärt. Daraus entstand die Projektplanung nach dem Arbeitsprozess RUP (Rational Unified Process), welche sich durch Iterationen von der Realisierung bis hin zur Abgabe erstreckte. Bei der Realisierung wurde grossen Wert auf Open Source Produkte gelegt. Daher setzten wir clientseitig JavaScript und serverseitig Java ein. Für die Verwaltung der Daten wurde eine Datenbank eingesetzt. Durch die enge Integration von Microsoft Excel mit TeachEval hat der/die Leiter/in die Möglichkeit, die Umfragen und Auswertungen mit Excel zu importieren oder zu exportieren. Ein eingebundener Job überprüft laufend, ob Umfragen gestartet, beendet oder ob die Teilnehmer durch eine E-Mail automatisch benachrichtigt werden müssen. Ein selbstentwickeltes Makro generiert dynamisch in der Excel-Datei die Diagramme und Korrelationen zu den vorhandenen Rohdaten.

Ergebnisse

Als Ergebnis entstand eine Webapplikation namens TeachEval, die sich vor allem darauf konzentriert, die Qualität der angebotenen Kurse der OdA G ZH sicherzustellen. Der momentane Workflow für die Befragung und deren Auswertung wurde in die Applikation integriert und automatisiert.

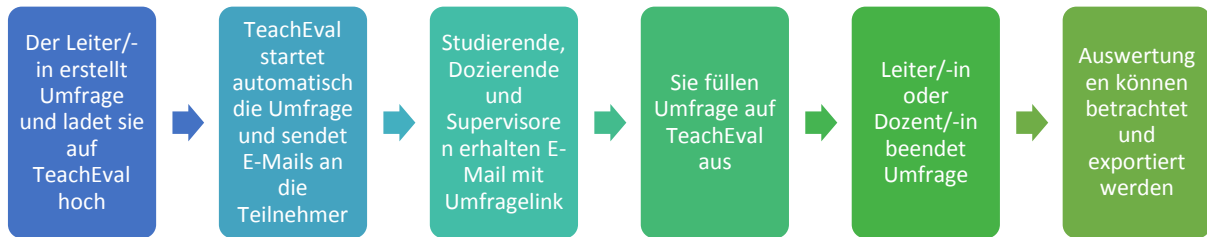


Abbildung 2 SOLL Situation

Durch die starke Integration mit Microsoft Excel ist der Leiter oder die Leiterin in der Lage, alle Angaben in der Excel-Datei abzulegen, um anschliessend die Umfrage auf TeachEval hochzuladen bzw. hinzuzufügen. Dies hat den Vorteil, dass die verwendete Excel-Datei für andere Umfragen wiederverwendet werden kann. Die Teilnahme an einer Umfrage ist mit Hilfe eines Web-Browsers möglich, die sowohl von PCs, Tablets als auch von Smartphones durchgeführt werden können. Aus den Befragungen können ansprechende Auswertungen in Form von Diagrammen realisiert werden und sind für Dozenten/innen und Leiter/innen zugänglich. Die Auswertung kann für die Weiterverwendung im Excel heruntergeladen und bearbeitet werden. Zudem können die Antworten von zwei beliebigen Fragen miteinander korreliert und exportiert werden. TeachEval wurde bereits Anfang Mai 2014 von der OdA G Zürich produktiv eingesetzt.

Ausblick

Durch unser Engagement und Motivation konnten wir eine qualitativ hochwertige Webapplikation entwickeln, welche den Bedürfnissen der OdA G Zürich entspricht. Das Projekt kann durch weitere Funktionen erweitert und optimiert werden. Die weiterführenden Ideen möchten wir in diesem Kapitel erläutern.

Umfrage verwalten

Das Verwalten von Umfragen wird in der aktuellen Version nicht unterstützt. Eine nachträgliche Bearbeitung einer Umfrage wäre aufgrund eines Benutzerfehlers eine hilfreiche Funktion, wenn z.B. ein Teilnehmer vergessen wurde. So könnte man diesen nachträglich zur Umfrage hinzufügen.

Umfrage Titel eindeutig

Während der Entwicklung haben wir uns darauf geeinigt, dass wir die Umfragen nicht löschen, sondern sie als archiviert setzen. Dadurch muss der Titel der Umfrage eindeutig sein, da sonst in der Liste mehrere Umfragen mit gleichem Titel erscheinen würde. Das ist aber nicht immer wünschenswert. Momentan erhält die Umfrage von der Leiterin den Namenszusatz FS<Jahr> oder HS<Jahr> und die Anzahl Semester.

Export als PDF

Eine Auswertung als PDF-Variante für den Druck würde sich gut eignen, wenn die Daten nicht für die Weiterverarbeitung benötigt werden. Diese Funktion könnte zusätzlich angeboten werden.

Inhaltsverzeichnis I

Abstract	6
Management Summary (MAS)	7
Technischer Bericht (TEB)	16
Schlussbericht (SCB)	35
Glossar (GLO)	39
Quellenverzeichnis (QUV)	43
Anhang	46
Projektplan (PMP)	47
Anforderungsspezifikation (SAS)	64
User Interface (UI)	82
Software Architektur Dokument (SDD)	91
Software Qualitätsmanagement (SQM)	124
Projektauswertung (PAU)	162

Inhaltsverzeichnis II

Abstract	6
Management Summary (MAS)	7
Technischer Bericht (TEB)	16
1. Einführung	17
2. Ausgangslage	17
2.1 Problemstellung	17
2.2 Anforderungen und Ziele	18
3. Konkurrenz-Produkte	19
3.1 SurveyMonkey	19
3.2 Lime Survey	20
3.3 Fazit	20
4. Chancen & Gefahren	21
5. Vorgehensweise	21
6. Ergebnisse	22
6.1 Erreichte Ziele	22
6.2 Optimierungen	31
7. Problemlösungen	32
7.1 Technologie	32
7.2 Kundennahe Lösung	32
7.3 Benutzerfreundlich	33
8. Schlussfolgerungen	33
9. Abbildungsverzeichnis	34
10. Tabellenverzeichnis	34
Schlussbericht (SCB)	35
1. Einführung	36
1.1 Zweck	36
2. Zielerreichung	36
3. Persönliche Erfahrungen	37
3.1 Dominique Sorg	37
4. Persönliche Erfahrungen	38
4.1 Benjamin Kehl	38
Glossar (GLO)	39
Quellenverzeichnis (QUV)	43
Anhang	46
Projektplan (PMP)	47
1. Einführung	48
1.1 Zweck	48

1.2 Gültigkeitsbereich	48
1.3 Übersicht	48
2. Projekt Übersicht.....	49
2.1 Zweck und Ziel.....	49
2.2 Einsatz	49
2.3 Lieferumfang	49
2.4 Annahmen und Einschränkungen	50
3. Projektorganisation	51
3.1 Organisationsstruktur	51
3.2 Team.....	51
3.3 Externe Schnittstellen	52
4. Management Abläufe.....	53
4.1 Kostenvoranschlag	53
4.2 Zeitliche Planung	53
4.3 Vorgehensmodell	53
4.4 Besprechungen.....	56
5. Risikomanagement.....	57
5.1 Risiken	57
5.2 Umgang mit Risiken	57
6. Arbeitspakete	58
7. Infrastruktur	60
8. Qualitätsmassnahmen.....	61
8.1 Dokumentation	61
8.2 Projektmanagement	61
8.3 Entwicklung	61
8.4 Testen.....	61
9. Abbildungsverzeichnis.....	63
10. Tabellenverzeichnis	63
Anforderungsspezifikation (SAS)	64
1. Einführung.....	65
1.1 Zweck	65
1.2 Beschreibung.....	65
1.3 Gültigkeitsbereich	65
1.4 Übersicht	65
2. Allgemeine Beschreibung.....	66
2.1 Produkt Perspektive	66
2.2 Produkt Funktion.....	66
2.3 Technologien.....	66
2.4 Benutzer Charakteristik.....	66

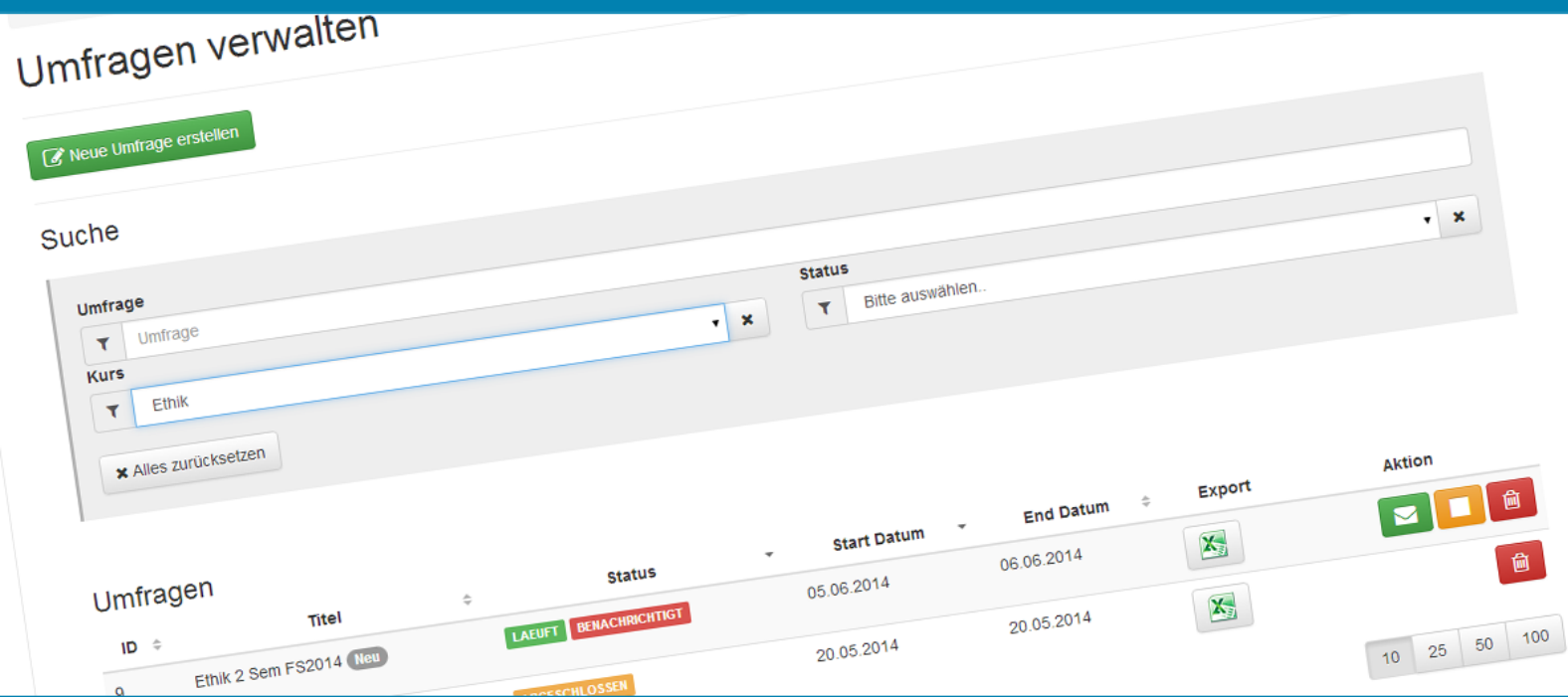
2.5 Benutzerfreundlichkeit	67
2.6 Kompatibilität.....	68
2.7 Einschränkungen	68
2.8 Abhängigkeiten	68
3. Use Cases.....	69
3.1 Priorisierung.....	69
3.2 Use Case Diagramm	70
3.3 Aktoren.....	71
3.4 Beschreibungen (Brief).....	71
3.5 Beschreibungen (Fully Dressed).....	73
4. Weitere Anforderungen	79
4.1 Qualitätsmerkmale.....	79
4.2 Zuverlässigkeit.....	79
4.3 Benutzbarkeit.....	79
4.4 Effizienz	79
4.5 Wartbarkeit	79
4.6 Übertragbarkeit.....	80
4.7 Schnittstellen.....	80
5. Abbildungsverzeichnis.....	81
6. Tabellenverzeichnis	81
User Interface (UI)	82
1. Einführung	83
1.1 Zweck	83
1.2 Gültigkeitsbereich	83
2. Wireframes.....	84
2.1 Grundgerüst	84
2.2 Mobile	84
2.3 Umfrage teilnehmen	85
2.4 Umfrage verwalten	86
2.5 Auswertung.....	88
3. Abbildungsverzeichnis.....	90
Software Architektur Dokument (SDD).....	91
1. Einführung.....	92
1.1 Zweck	92
1.2 Gültigkeitsbereich	92
1.3 Übersicht	92
2. Systemübersicht.....	93
2.1 Client	93
2.2 Server	94

3. Representational State Transfer (REST)	95
3.1 Richardson Maturity Model	95
4. Projektstruktur	96
4.1 Frontend.....	97
4.2 Backend.....	98
4.3 Verwendete Libraries.....	98
5. Physische Architektur.....	100
5.1 Datenbank.....	100
5.2 E-Mail	100
5.3 Deployment.....	100
6. Logische Architektur.....	102
6.1 Frontend.....	102
6.2 Backend.....	107
7. Open Source Software	118
7.1 Frontend.....	118
7.2 Backend.....	118
8. Datenspeicherung	119
8.1 User Bereich.....	120
8.2 Umfrage Bereich	121
8.3 Löschoptionen.....	122
9. Abbildungsverzeichnis.....	123
10. Tabellenverzeichnis.....	123
Software Qualitätsmanagement (SQM).....	124
1. Zweck.....	125
2. Einführung.....	125
3. Systemtestspezifikation	126
3.1 Systemtest.....	126
3.2 Unit Testing	129
4. Systemtestprotokoll	132
4.1 Angaben zur Durchführung.....	132
4.2 Protokoll.....	132
4.3 Verbesserungsmöglichkeiten.....	140
5. Leistungstests	141
6. Kompatibilitätstests	143
6.1 Einleitung	143
6.2 Angaben zur Durchführung.....	143
6.3 Einschränkung	143
6.4 Überblick aller Tests.....	143
6.5 Internet Explorer 8.....	144

6.6 Internet Explorer 9	144
6.7 Internet Explorer 10	145
6.8 Firefox	146
6.9 Chrome	146
6.10 Safari	147
6.11 Opera	147
6.12 Browserstack	148
7. Fazit	149
8. Usability Test	150
8.1 Ziel und Zweck	150
8.2 Testpersonen	150
8.3 Durchführung	150
8.4 Statistiken	150
8.5 Bewertung	156
8.6 Fazit aus den Testergebnissen	157
9. Verbesserungsmöglichkeiten	158
10. Abbildungsverzeichnis	160
11. Tabellenverzeichnis	161
Projektauswertung (PAU)	162
1. Zweck	163
2. Überblick	164
3. Ist / Soll-Vergleich	164
4. Team-Vergleich	165
5. Aktivitäten Vergleich	166
6. Fazit	166
7. Abbildungsverzeichnis	167

TeachEval

Web App für die Unterrichtsevaluation



Technischer Bericht (TEB)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Einführung

Die Organisation der Arbeitswelt Gesundheit Zürich (kurz Oda G ZH) ist der Branchenverband für Berufsbildung im Gesundheitswesen des Kantons Zürich. Sie bezweckt die Vertretung ihrer Mitglieder bei der Gestaltung, Entwicklung und Umsetzung der Gesundheitsausbildungen im Kanton Zürich. Der Verein ist eine kantonsweit tätige Organisation der Arbeitswelt im Sinne der Berufsbildungsgesetze sowie der Fachhochschulgesetzgebung von Bund und Kanton (Oda G ZH Offizielle Webseite, 2014).

Die Oda G ZH bildet in Form von überbetrieblichen Kursen und Weiterbildungen angehende Fachfrauen und Fachmänner Gesundheit, Assistentinnen und Assistenten Gesundheit und Soziales aus. Zudem bieten sie in der höheren Berufsbildung ein praxisnahes Modulangebot Lernbereich Training und Transfer Praxis (LTT Praxis) für Studierende der Pflege HF an. Die Leiterin von LTT Praxis Pflege ist unsere direkte Ansprechperson.

2. Ausgangslage

Für die Oda G ZH ist eine hohe Qualität der Bildungsangebote ein grosses Anliegen. Deshalb ist das Qualitätsmanagement einer der zentralen Schwerpunkte. Damit die Oda G ZH ein qualitativ hochwertiges Schulungsangebot anbieten kann, werden mit Umfragen schriftliche Feedbacks von den Studierenden, Dozenten und Supervisoren eingeholt und das Angebot fortlaufend überprüft und optimiert. Eine Umfrage enthält Einzel- & Mehrfachauswahl-Fragen sowie Textfelder. Eine spezielle Form von Fragen sind die IST-/SOLL-Einzelauswahl-Fragen. Die Befragten sollen anhand dieser IST-/SOLL-Fragen einen persönlichen Vergleich des Kurses ziehen. Die rund 20 bis 30 Fragen pro Student(in), Dozent(in) oder Supervisor(in) werden anschliessend von der Leiterin in eine Excel-Datei eingetragen, ausgewertet und daraus Diagramme und Korrelationen gebildet. Am Schluss entsteht ein Semester Report.

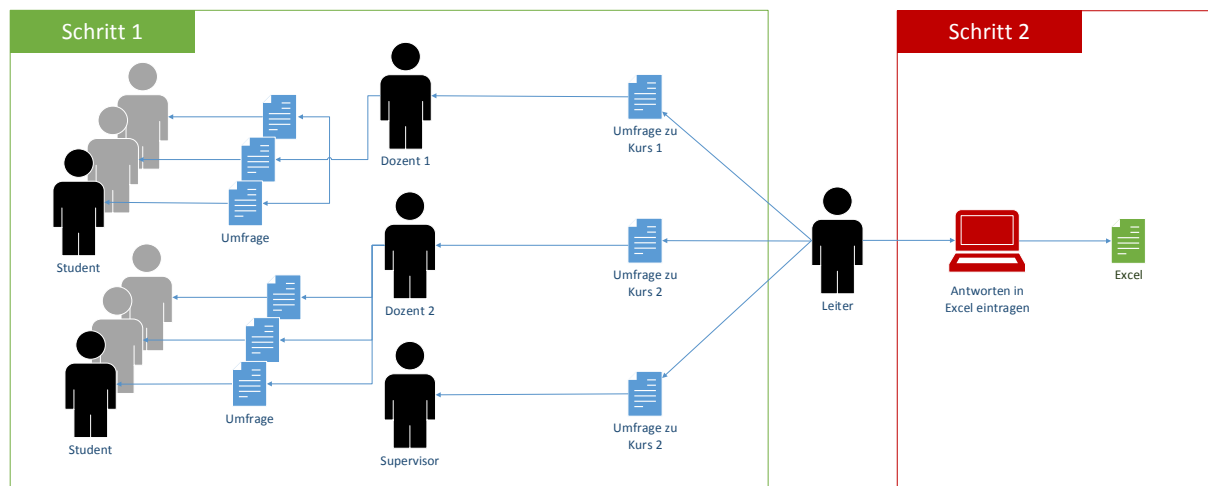


Abbildung 3 IST Workflow

2.1 Problemstellung

Die schriftlichen Umfragen und Auswertungen sind für das Qualitätsmanagement der Oda G ZH bereits ein fest verankerter Workflow, der von der Leiterin durchgeführt wird. Für die Bereiche LTT Praxis und Psychiatrie finden im Jahr 2014 ungefähr rund 25 Kurse statt. Nach jedem Kurs wird eine schriftliche Umfrage durchgeführt, die anschliessend von Hand ausgewertet wird. Wenn wir die Berechnung für einen einzigen Kurs weiterführen würden, dann wären es bei 10 Teilnehmern mit durchschnittlich 25 bis 30 Fragen schon gesamthaft 250 bis 300 Fragen, die bei der Auswertung berücksichtigt werden

müssen. Diese manuelle Arbeit ist nicht nur sehr zeitintensiv und aufwändig, sondern auch fehleranfällig.

2.2 Anforderungen und Ziele

In dieser Bachelorarbeit soll eine Webapplikation entwickelt werden, die sich vor allem darauf konzentriert, den momentanen Workflow der OdA G ZH für die Befragung von Studenten, Dozenten oder Supervisoren zu übernehmen und deren Auswertung zu automatisieren. Die Teilnahme an einer Umfrage ist mit Hilfe eines Webbrowsers möglich, die sowohl von PCs, Tablets als auch von Smartphones durchgeführt werden können. Die Webapplikation soll in JavaScript, HTML 5 und CSS erstellt werden. Benutzte Drittkomponenten (z.B. Datenbank) und Tools sollen als OSS (Open Source Software) vorliegen.

Ein zweites Hauptziel der Arbeit besteht darin, ansprechende Auswertungen zu realisieren, die auch Diagramme enthalten und als Microsoft Excel- oder als Word-Dateien bereitgestellt werden. Der Auswertungsteil der Applikation soll die Möglichkeit bieten, die Antworten auf ausgewählte Fragen zu korrelieren, d.h. die Kovarianz numerischer Auswertungsergebnisse zu berechnen.

Als Endergebnis soll eine Webapplikation bezüglich der Anforderungen erstellt werden, welche die Leiterin im Bereich LTT Praxis Pflege der OdA G ZH bei der Befragung und Auswertung unterstützt. Während einer Umfrage soll die Applikation unterschiedlichen Zielgruppen (Student, Dozent, Supervisor) unterschiedliche Fragen stellen können. Die Personen werden per E-Mail oder durch die Verteilung von Tokens zu einer Umfrage eingeladen. Neben der Leiterin, dürfen auch Dozenten auf die Auswertungen zugreifen. Dozenten sollten dies aber erst nach ihrer Teilnahme machen dürfen. Schlussendlich muss der jetzige, manuelle Workflow soweit wie möglich automatisiert werden, damit der Aufwand für die Beteiligten weniger wird. Zudem soll bei der Umsetzung eine kundennahe Lösung angeboten werden.

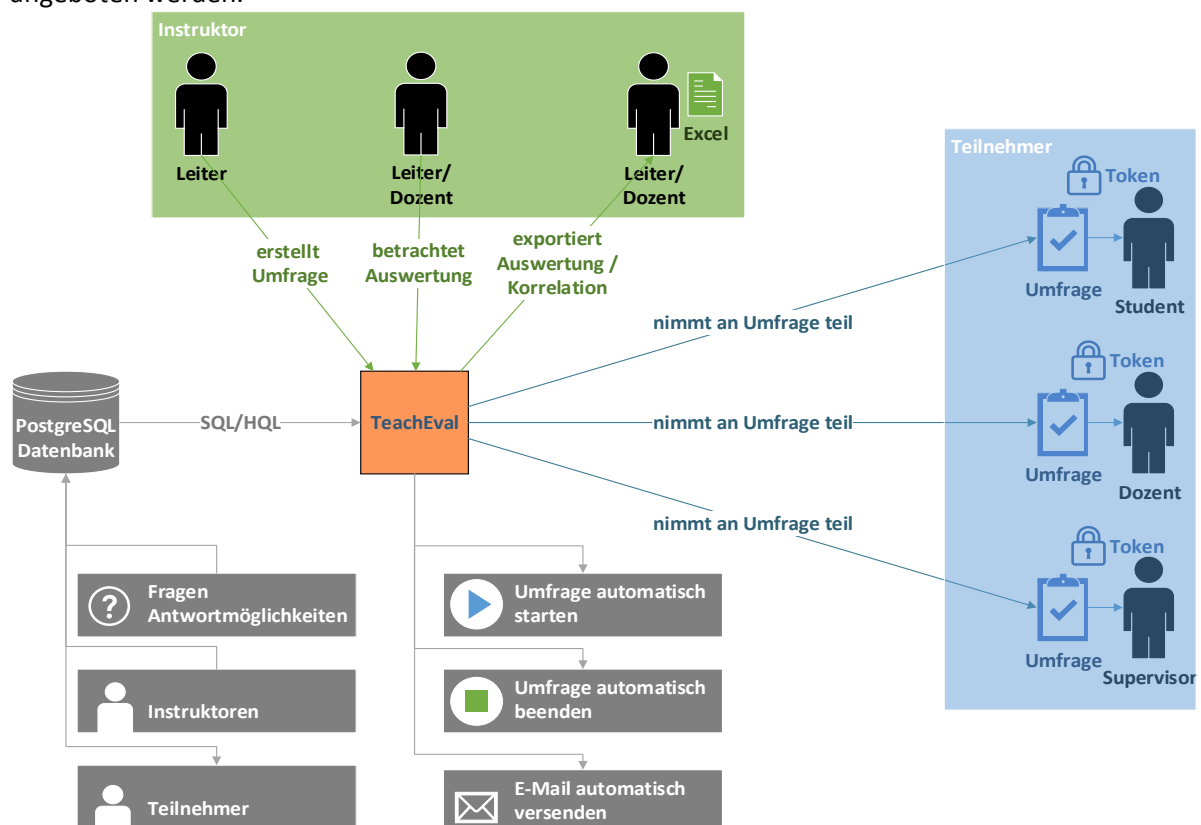


Abbildung 4 SOLL Workflow

3. Konkurrenz-Produkte

Umfragen sind ein effizientes Mittel um systematisch an Informationen über Meinungen, Wissen und Verhaltensweisen der Teilnehmer zu gelangen. Denn diese liefern sehr oft die entscheidenden Informationen, um auf die Bedürfnisse der Teilnehmer besser und gezielter einzugehen und das Angebot zu optimieren. Dies hat die OdA G ZH erkannt und sucht nun nach einem geeigneten Online Umfrage-Tool. Jedoch gibt es viele Online Umfrage Tools auf dem Markt, die entweder kostenlos oder mit einer Gebühr genutzt werden können. Daher stellt sich schnell die Frage, wieso an der HSR (Hochschule für Technik Rapperswil) eine weitere Umfrage-Lösung entwickelt werden soll?

Um diese Frage zu beantworten, wurde eine kleine Konkurrenz-Analyse von zwei bekannten Online Umfrage-Tools erstellt.

3.1 SurveyMonkey

SurveyMonkey ist ein amerikanisches Unternehmen und bietet als Online-Umfrage-Anbieter eine Vielzahl von Funktionen an. In der folgenden Tabelle wurden die wichtigsten Kern-Funktionen mit den anzubietenden Produkten zusammengetragen und verglichen:

Basic	Plus	Gold	Platinum
Design Features			
10 Fragen pro Umfrage 100 Beantwortungen pro Umfrage	Unlimitierte Fragen 1000 Antworten pro Monat plus zusätzliche Gebühr bei Übertretung	Unbegrenzte Fragen Unbegrenzte Beantwortungen	Unbegrenzte Fragen Unbegrenzte Beantwortungen
Benutzerfreundliches, webbasiertes Tool	Benutzerfreundliches, webbasiertes Tool	Benutzerfreundliches, webbasiertes Tool	Benutzerfreundliches, webbasiertes Tool
31 Umfragevorlagen	51 Umfragevorlagen	51 Umfragevorlagen	51 Umfragevorlagen
15 Fragetypen	15 Fragetypen	15 Fragetypen	15 Fragetypen
kein Unternehmenslogo	mit Unternehmenslogo	mit Unternehmenslogo	mit Unternehmenslogo
Validieren/anfordern von Umfragebeantwortungen	Validieren/anfordern von Umfragebeantwortungen	Validieren/anfordern von Umfragebeantwortungen	Validieren/anfordern von Umfragebeantwortungen
keine druckreife PDF-Version	keine druckreife PDF-Version	keine druckreife PDF-Version	keine druckreife PDF-Version
Erfassungsfeatures			
Senden der Umfrage per Weblink, E-Mail oder Twitter	Senden Ihrer Umfrage per Weblink, E-Mail oder Twitter	Senden Ihrer Umfrage per Weblink, E-Mail oder Twitter	Senden Ihrer Umfrage per Weblink, E-Mail oder Twitter
Senden der Umfrage mit einem E-Mail-Manager	Senden der Umfrage mit einem E-Mail-Manager	Senden der Umfrage mit einem E-Mail-Manager	Senden der Umfrage mit einem E-Mail-Manager
Keine erweiterte Sicherheit (SSL)	Erweiterte Sicherheit (SSL)	Erweiterte Sicherheit (SSL)	Erweiterte Sicherheit (SSL)
Analysefeatures			
Beantwortungen können nicht heruntergeladen werden	Herunterladen von Beantwortungen	Herunterladen von Beantwortungen	Herunterladen von Beantwortungen

Basic	Plus	Gold	Platinum
Es können keine benutzerdefinierten Diagramme erstellt & heruntergeladen werden	Erstellen & Herunterladen von benutzerdefinierten Diagrammen	Erstellen & Herunterladen von benutzerdefinierten Diagrammen	Erstellen & Herunterladen von benutzerdefinierten Diagrammen
kostenlos	35 CHF pro Monat	449 CHF pro Jahr	1'169 CHF pro Jahr

Tabelle 1 Produktevergleich von SurveyMonkey (Pricing SurveyMonkey, 2014)

Die *SurveyMonkey Basic* Variante ist sehr limitiert in der möglichen Anzahl Fragen und Beantwortungen. Zudem bietet sie keine Möglichkeit, die Auswertungsergebnisse als Datei herunterzuladen. Die *Gold* oder *Platinum* Varianten würden die Bedürfnisse der O&A GZH nur bedingt besser erfüllen, da in den 15 Fragetypen kein Fragetyp wie IST-/Soll-Einzelauswahl vorhanden ist (HelpCenter for Questoin Types SurveyMonkey, 2014). Der Auswertungsteil ist frei konfigurierbar und es ist möglich, die Daten als Excel, PDF oder als CSV-Datei zu exportieren. Diagramme werden jedoch nur als Bilder angeboten (HelpCenter for creating Charts SurveyMonkey, 2014), was für die Weiterverarbeitung der Daten nicht erwünscht ist. Ein weiterer Nachteil ist, dass für jeden Kurs drei Umfragen, nämlich für Studenten, Dozenten und für Supervisoren, erstellt werden muss.

3.2 Lime Survey

Der Funktionsumfang von *Lime Survey* kann etwa mit SurveyMonkey gleichgestellt werden. Jedoch bietet auch Lime Survey keine Fragetypen mit IST-/Soll-Einzelauswahl an (Manual Question Types in Lime Survey, 2014). Der Auswertungsteil ist ebenfalls als Excel exportierbar, jedoch sind diese nur Rohdaten.

Produkt (Antworten / Monat)	Preis ¹ in CHF
25	Kostenlos
100	9
250	20
500	31
1000	54
2500	112
5000	179

Tabelle 2 Produkteübersicht Lime Survey (Pricing LimeService, 2014)

Im Vergleich zu SurveyMonkey hat *Lime Survey* ein anderes Preiskonzept und zwar rechnen sie mit der Anzahl Antworten ab. Aus finanzieller Hinsicht wäre dieses Preismodell tragbar, da unter einer Umfrageantwort die Teilnahme an der kompletten Umfrage zählt, unabhängig von der Anzahl Fragen- und Antwortmöglichkeiten. Jedoch kann auch Lime Survey nicht die Anforderungen decken.

3.3 Fazit

Beide Produkte unterstützen eine Vielzahl an Funktionen. Der Kunde kann somit seine Umfrage nach seinen Wünschen anpassen. Jedoch lässt sich nicht eine komplette kundenzugeschnittene Umfrage mit diesen Produkten verwirklichen, ausser man verzichtet auf gewisse Funktionen. Aus diesem Grund

¹ Preise wurden am 04.06.14 von Dollar zu CHF mit dem Kurs 0.8959 umgerechnet

wurde entschieden, eine eigene Web Applikation zu entwickeln. Deren Stärken liegen vor allem in den folgenden Bereichen:

- Eine Umfrage deckt alle Fragen und Antwortmöglichkeiten für die Benutzergruppen (Student, Dozent, Supervisor) ab. Es soll nicht für jede Benutzergruppe eine individuelle Umfrage gestartet werden.
- Die Antworten der unterschiedlichen Benutzergruppen werden in die Auswertung einbezogen und einander gegenübergestellt.
- Die Leiterin möchte die bereits eingesetzten IST/SOLL Fragetypen weiterverwenden.
- Der Excel Export ermöglicht die automatisierte Berechnung und Generierung von Diagrammen die weiterverwendet werden können.
- Es können Korrelationen auf zwei ausgewählten Fragen erstellt und als Excel exportiert werden.
- Der Einsatz von TeachEval ist aus finanzieller Hinsicht tragbar.

4. Chancen & Gefahren

Nebst vielen Vorteilen bzw. Chancen eines Umfrage-Tools beherbergen diese auch Risiken. Wir unterscheiden zwischen den technischen (Applikation bezogenen) Risiken, die wir in einem separaten Dokument führen und den Risiken, die das Benutzerverhalten beeinträchtigen können. Die folgende Abbildung soll die Chancen und Gefahren, die für die Oda G ZH auftreten können, aufzeigen:

Chancen	Gefahren
<ul style="list-style-type: none"> - Kosten/Arbeitsaufwand wird durch die Automatisierung reduziert. - Umfragen werden schneller ausgefüllt. - Die Applikation kann von nicht IT Personen bedient werden. - Applikation ist ausbaubar und für die Auftraggeberin zugeschnitten. - Fokus (Befragung & Auswertung) konzentriert sich auf das Wesentliche - Zeitunabhängige Befragungsmöglichkeit 	<ul style="list-style-type: none"> - Bei einer schriftlichen Umfrage zwingt man die Studenten fast, diese auszufüllen, da diese von der Lehrperson wieder eingesammelt wird. Bei der neuen Web Applikation können die Teilnehmer die Befragung im Unterricht, unterwegs oder zu Hause ausfüllen. Durch die physische Präsenz der Lehrperson werden die Studenten fast gezwungen die Umfrage auszufüllen, was mit der Applikation verloren gehen kann. Dadurch könnten die Anzahl Teilnahmen abnehmen. Als Alternative sind die Tokens als Excel exportierbar und können im Unterricht verteilt werden. - Die Applikation bezieht sich nur auf die Anforderungen des Arbeitgebers und ist keine Universallösung. Erweiterungen oder Anpassungen benötigen einen Entwickler.

Tabelle 3 Chancen / Gefahren von TeachEval

5. Vorgehensweise

Als Arbeitsprozess wurde RUP gewählt. Der Grund, weshalb wir dieses Vorgehensmodell gewählt haben, liegt an den bereits erfolgreich gesammelten Erfahrungen in der Studienarbeit und im SE2-Projekt. Ausserdem können mittels RUP die Aufteilungen der Arbeit in einzelne Phasen und Iterationen mit Meilensteinen definiert werden. Einzelne Arbeitspakete können untereinander aufgeteilt werden.

In den ersten beiden Wochen wurden die Vorbereitungen für das bevorstehende Projekt und die Kick-Off Meetings mit dem Betreuer und der Auftraggeberin getroffen.

Ab der zweiten Woche startete die Elaboration-Phase, welche 3 Wochen lang dauerte. In der ersten Woche definierten wir die Risiken und erstellten den Projektmanagementplan. Zudem liefen die Einarbeitungen in die Technologien wie AngularJS, Java Jersey, Hibernate, Apache POI, usw. bereits auf Hochtouren.

In der zweiten Phase der Elaboration wurde vermehrt Zeit für die Erstellung der Architektur Prototypen investiert und die definierten Risiken laufend neu beurteilt. Auch ein erster Entwurf für das Datenbankmodell wurde erstellt und die Anforderungen in Use Cases ausgearbeitet.

Die Construction Phase enthält 4 Iterationen und dauerte insgesamt 9 Wochen. Eine Iteration geht im Schnitt 2 Wochen. Die letzte Woche konzentrierte sich vor allem auf das Refactoring und Testen der Applikation. In der ersten Iteration wurde das Datenbank Schema aufgesetzt, welche als Basis für die Web Applikation dient. Zusätzlich konzentrierten wir uns auf die Implementation des Backends und der Datenbeschaffung mit Hibernate. Durch die Einarbeitung hatten wir zu diesem Zeitpunkt bereits eine Grund-Architektur des Frontends, die vereinzelte Daten aus dem Backend holen konnte.

In der zweiten Iteration wurde die Grund Architektur verbessert und erweitert. Zudem wurde Im Frontend die Benutzerverwaltung, Umfrageverwaltung und deren Teilnahme realisiert.

Die 3te Iteration befasste sich mit einer ersten Auswertung und Korrelation der Umfragen. Zudem wurden in dieser Iteration viele Verbesserungen und Tests gemacht. Am Ende dieser Phase wurde ein erster Prototyp auf dem produktiven Server aufgeschaltet.

Die vierte und letzte Iteration der Construction-Phase diente für weitere Verbesserungen und um die Applikation benutzerfreundlicher für den Endkunden zu machen. Zudem wurden gefundene Bugs behoben, Code-Refactorings durchgeführt und viel getestet. Ein Support für die OdA G ZH wurde während des gesamten Projektes gewährleistet.

In der Schlussphase, auch Transition-Phase, wurden die Dokumente aktualisiert und für die Abgabe vorbereitet.

6. Ergebnisse

6.1 Erreichte Ziele

Als Ergebnis entstand eine erste Version der Web Applikation TeachEval. Sie ist in der Lage, Umfragen als Excel-Datei zu erstellen und zu verwalten. Eine Benutzerverwaltung regelt den Zugriff. Die Teilnehmer können die Umfrage ausfüllen und der Leiter(in) oder der Dozent(in) kann die Auswertung im Auswertungsbereich begutachten. Das Layout wurde mit Hilfe des Corporate Designs der OdA G ZH erstellt, dazu wurde uns ein Richtlinien-Dokument bereitgestellt.

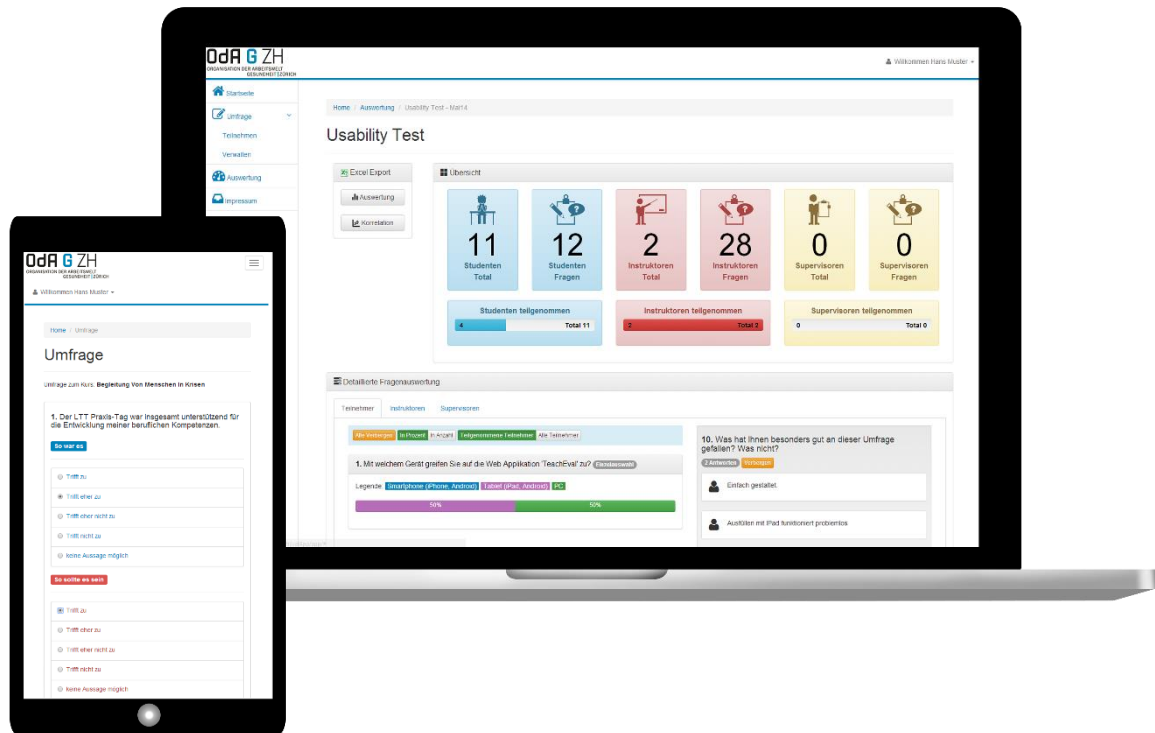


Abbildung 5 Endprodukt

In den folgenden Schritten werden die Ergebnisse detaillierter erläutert.

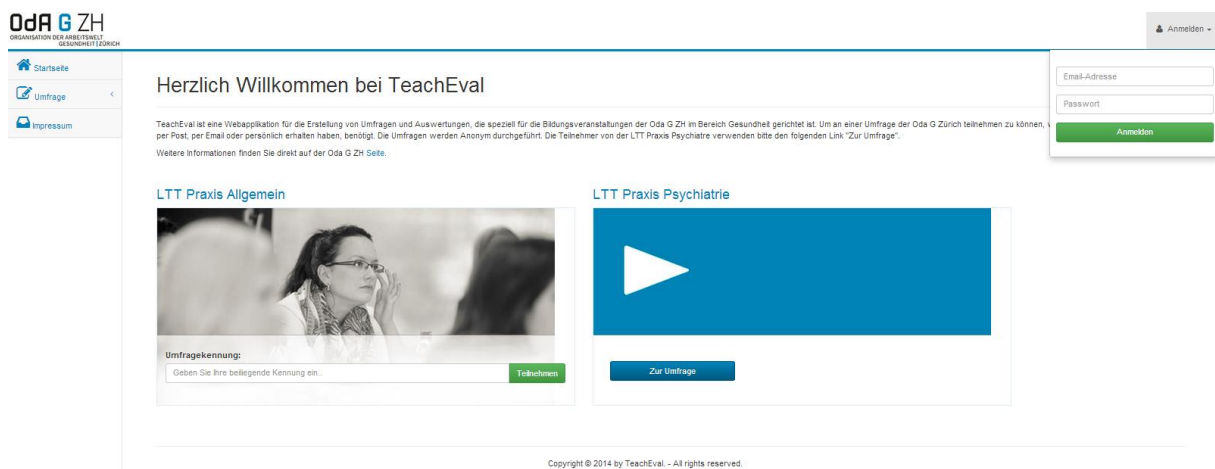


Abbildung 6 Startseite TeachEval

Die Startseite enthält zwei Fachbereiche. Die *LTT Praxis Allgemein* und die *LTT Praxis Psychiatrie*. Da TeachEval von zwei Bachelorarbeitsgruppen in Konkurrenz entwickelt wurde und jede Gruppe einen Fachbereich zugewiesen bekam, musste als Einstiegspunkt eine Startseite erstellt werden. Da wir bereits von Anfang an eine Startseite hatten, haben sich beide Gruppen auf unsere Startseite geeinigt. Für die Verwaltung von Umfragen oder Instruktoren wird ein Login-Zugriff benötigt, welcher im Abbild 4 oben rechts zu finden ist.



Abbildung 7 Profil

Jeder eingeloggte Instruktor hat ein Benutzerprofil. In diesem kann er seine E-Mail Adresse ändern oder sein Passwort zurücksetzen.

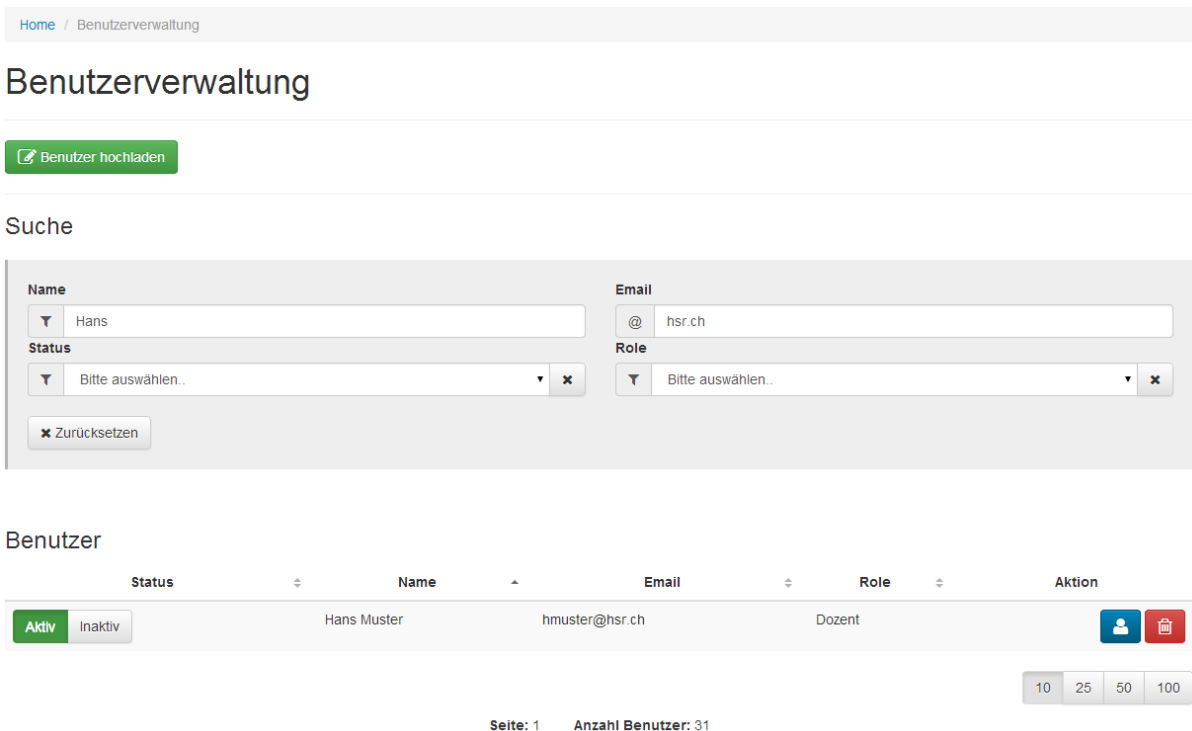


Abbildung 8 Benutzerverwaltung

Die Leiterin kann über die Benutzerverwaltung nach eingetragenen Benutzern suchen und diese verwalten. Neue Benutzer können mittels einer Excel-Vorlage hochgeladen werden und bereits existierende Benutzer werden aktualisiert. Durch den Excel-Import werden keine Benutzer gelöscht.

[Home](#) / Umfragen verwalten

Umfragen verwalten

[Neue Umfrage erstellen](#)

Suche

Umfrage

Kurs

Status

Umfragen

ID	Titel	Status	Start Datum	End Datum	Export	Aktion
12	Begleitung Von Menschen In Krisen FS14 Neu	ERSTELLT	13.06.2014	13.06.2014		
4	Begleitung Von Menschen In Krisen FS2014	ABGESCHLOSSEN	08.05.2014	08.05.2014		

10 25 50 100

Seite: 1 Anzahl Umfragen: 14

Abbildung 9 Umfrageverwaltung

Im Bereich *Umfrage verwalten* können wie bei der Benutzerverwaltung, neue Umfragen über eine Excel Datei hochgeladen und erstellt werden. Dabei muss eine Umfrage einen eindeutigen Umfragetitel haben. Erstellte Umfragen werden entweder durch die Leiterin oder durch den CronJob automatisch gestartet und beendet. Der Status informiert, ob die jeweilige Umfrage erstellt, läuft oder abgeschlossen wurde. Die Umfragetokens werden nach dem Hinzufügen einer Umfrage den Teilnehmern zugewiesen und werden für den Excel-Export hinzugefügt.

Erfolgreich! Umfragen wurden erfolgreich hinzugefügt x

Neue Umfrage erstellen

Bitte laden Sie Ihre Umfragen als Excel Datei hoch. Dabei muss folgende Vorlage verwendet werden: [Excel Vorlage](#)

No file chosen

Datei	Dateigröße	Fortschritt	Status	Aktion
Begleitung von Menschen in Krisen FS14.xlsx	0,05 MB		Erfolgreich	

Anzahl Dateien: 1

Abbildung 10 Umfrage als Excel importieren

Eine neue Umfrage wird mittels einer Excel Vorlage, welche heruntergeladen werden kann, hochgeladen. Es kann auch eine existierende Umfrage exportiert werden und die Änderungen dann als neue Umfrage hochgeladen werden. Die Vorlage stellt die richtige Strukturierung des Imports sicher. Es können mehrere Umfragen gleichzeitig hochgeladen werden.

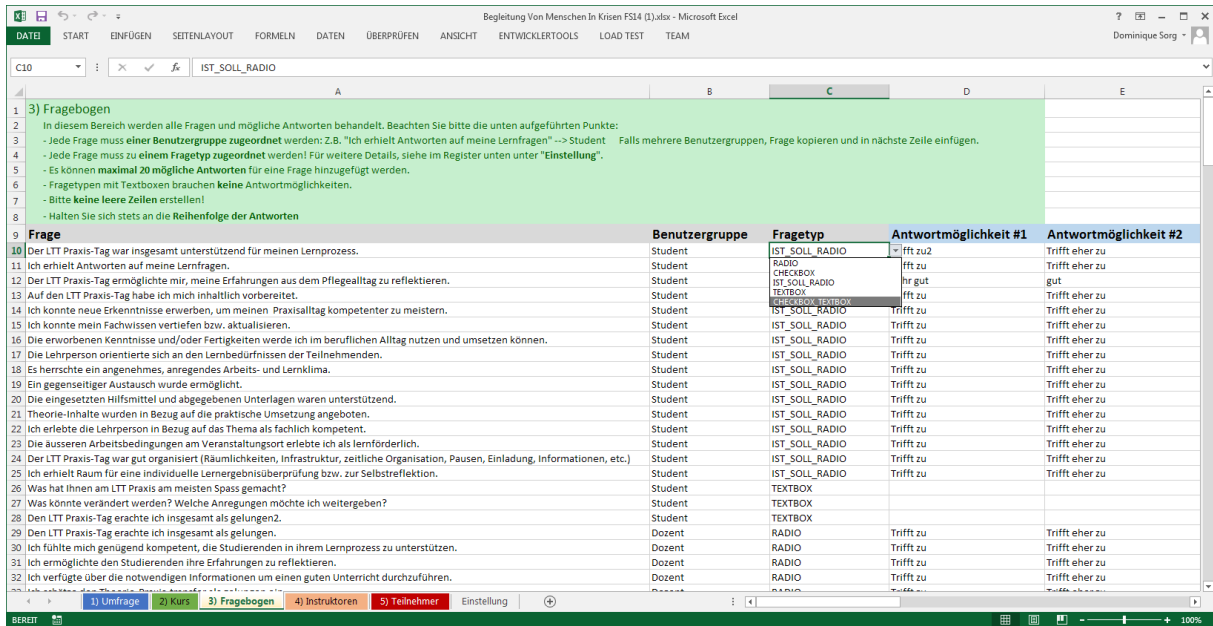


Abbildung 11 Umfrage als Excel importieren

Wie in der Abbildung 11 ersichtlich ist, besitzt die Excel Vorlage mehrere Arbeitsblätter. Jedes dieser Arbeitsblätter repräsentiert einen bestimmten Bereich. Im Bereich *Umfrage* wird der Titel und die Dauer der Umfrage eingetragen. Das Register *Kurs* enthält den Namen für die Kategorisierung der Umfragen. Die Fragen mit dem jeweiligen Fragetyp und zu welchem Benutzer (Student, Dozent oder Supervisor) sie zugewiesen bekommt, werden im Register *Fragebogen* erfasst. Im Bereich *Instruktor* werden die Dozenten, die den Kurs leiten und unter *Teilnehmer* die Studenten mit Vornamen, Nachnamen und E-Mail eingetragen.

Die obige Abbildung zeigt eine ausgefüllte Excel Vorlage. Die Benutzergruppen oder Fragetypen werden mit Dropdown Menüs unterstützt um Fehleingaben schon frühzeitig zu vermeiden.

LTT Praxis Allgemein

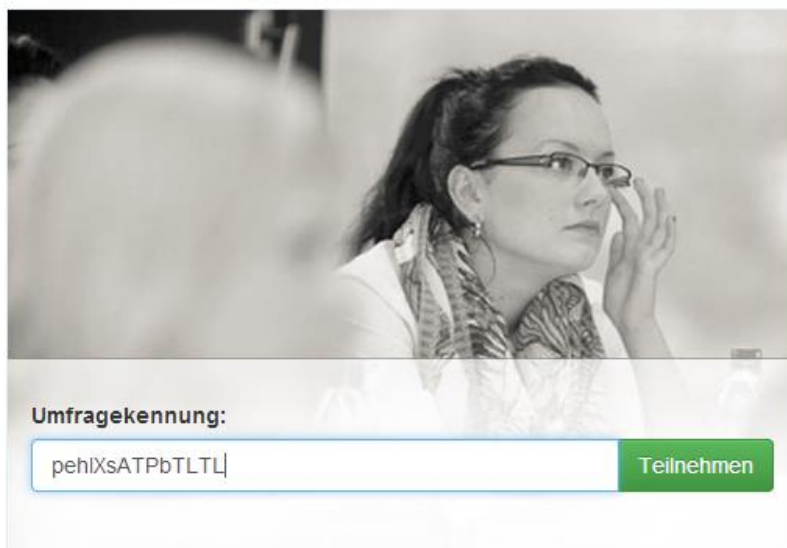


Abbildung 12 An Umfrage teilnehmen

Auf der Startseite können Teilnehmer mit einem 14 Stelligen Umfragetoken, das sie per Post, E-Mail oder persönlich erhalten haben, an der Umfrage teilnehmen.

[Home](#) / [Umfrage](#)

Umfrage

Umfrage zum Kurs: **Begleitung Von Menschen In Krisen**

1. Der LTT Praxis-Tag erachte ich insgesamt als unterstützend für die Entwicklung von beruflichen Kompetenzen

- Trifft zu
- Trifft eher zu
- Trifft eher nicht zu
- Trifft nicht zu
- keine Aussage möglich

2. Die übergeordneten Lern-Ziele des Tages werden bekannt gegeben

- Trifft zu
- Trifft eher zu
- Trifft eher nicht zu
- Trifft nicht zu
- keine Aussage möglich

Abschicken Abbrechen

Abbildung 13 Umfrage ausfüllen

Nachdem das Token erfolgreich einer Umfrage zugeordnet worden ist, gelangt der Teilnehmer auf seine Umfrage.

TeachEval unterstützt folgende Fragetypen:

- Eine Einzelauswahl (RADIO-Typ, gut geeignet für numerische Fragen)
- IST-/SOLL-Einzelauswahl (RADIO-Typ mit IST-/SOLL-Situationen)
- Kommentar (Textboxes für offene Fragen)
- Mehrfachauswahl (Checkboxes, gut geeignet für kategorische Fragen)
- Mehrfachauswahl mit Kommentar (gut geeignet für kategorische Fragen mit Kommentar)

Umfrage abgeschlossen

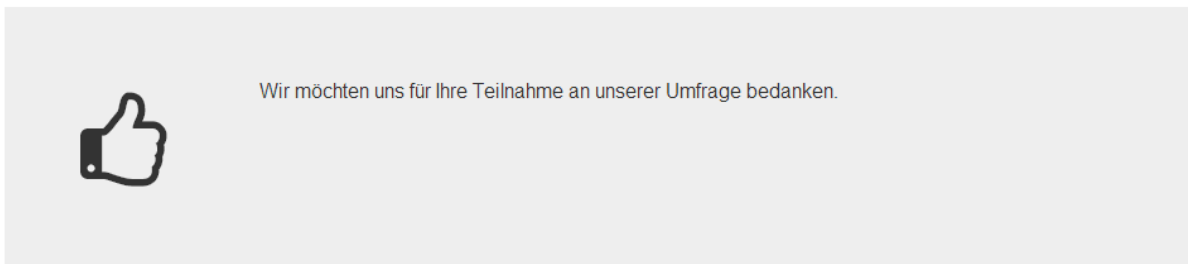


Abbildung 14 Umfrage beendet

Der Teilnehmer hat die Umfrage erfolgreich abgeschlossen.

Umfrage geschlossen

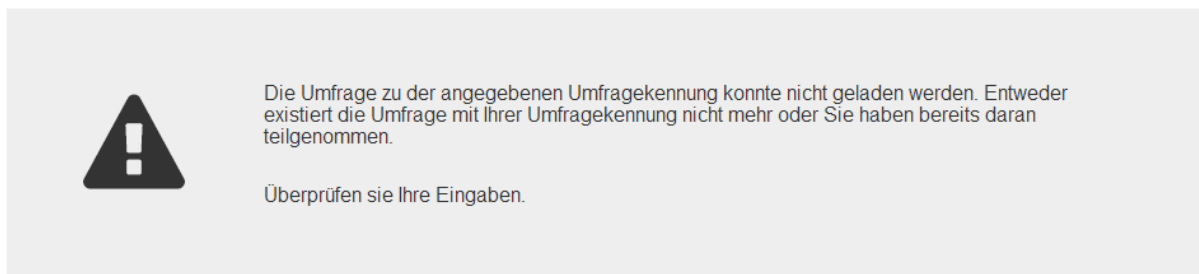


Abbildung 15 Umfrage bereits geschlossen

Falls der Teilnehmer mit der gleichen Umfragekennung oder über die Dauer der Umfrage zugreifen möchte, dann erhält der Benutzer eine Fehlermeldung und wird dazu aufgefordert, seine Angaben nochmals zu überprüfen.

Auswertung

Suche

Umfrage

Umfrage

Kurs

Status

Umfragen

ID	Titel	Status	Teilgenommen / Total	Start Datum	End Datum	Aktion
10	Infusionen 3. Sem FS2014	LAUFT	16 / 20	25.05.2014	13.06.2014	<input type="button" value="x"/>
7	Multimorbidität 2. Sem. FS2014	ABGESCHLOSSEN	20 / 20	12.05.2014	14.05.2014	<input type="button" value="Bar chart"/>

10 25 50 100

Seite: 1 Anzahl Umfragen: 2

Abbildung 16 Übersicht aller Umfragen

Für eine(n) Leiter(in) werden im Auswertungsteil alle Umfragen als Liste angezeigt. Als Dozent(in) nur die Umfragen, die er/sie auch leitet. Der Benutzer kann über den Zwischenstand, wie viele momentan

teilgenommen haben, entscheiden, ob er/sie die laufende Umfrage beenden möchte, um dann die Auswertungsdetails anzusehen.

Home / Auswertung / Begleitung Von Menschen In Krisen FS14

Begleitung Von Menschen In Krisen FS14

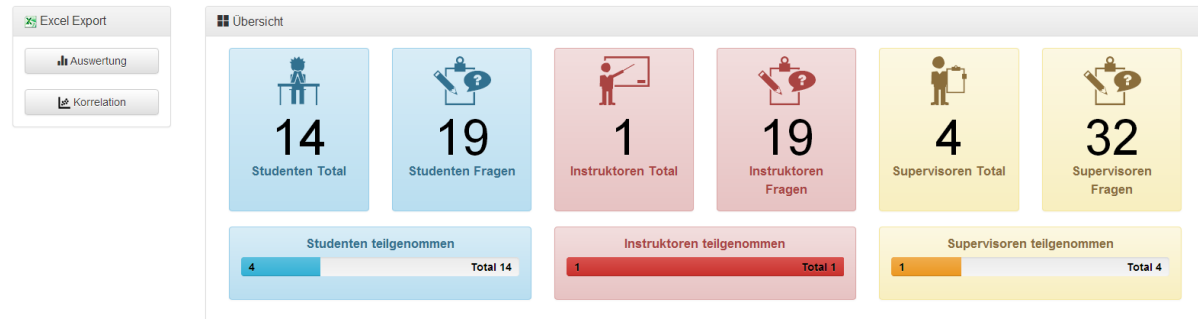


Abbildung 17 Übersicht der Auswertung

In den Auswertungsdetails soll die Anzahl Teilnahmen genauer aufgezeigt werden. Als Export-Optionen kann die komplette Auswertung mit Details als Excel heruntergeladen werden oder eine Korrelation zwischen zwei Fragen generiert werden.

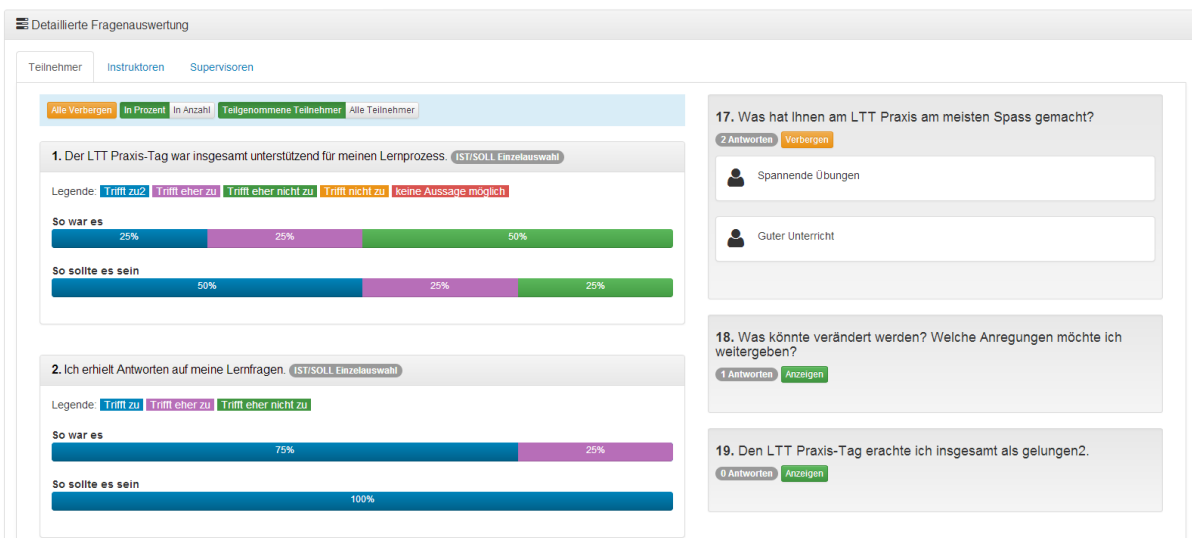


Abbildung 18 Detaillierte Auswertung auf Fragen

Wie in *Abbildung 18* ersichtlich ist, wird die *detaillierte Fragenauswertung* in zwei Hälften geteilt. In der linken Hälfte werden die Einzel- und Mehrfachauswahl-Fragen mit gestapelte Balken und mit Balkendiagrammen ausgewertet. In der rechten Hälfte werden die Kommentare untereinander aufgelistet.

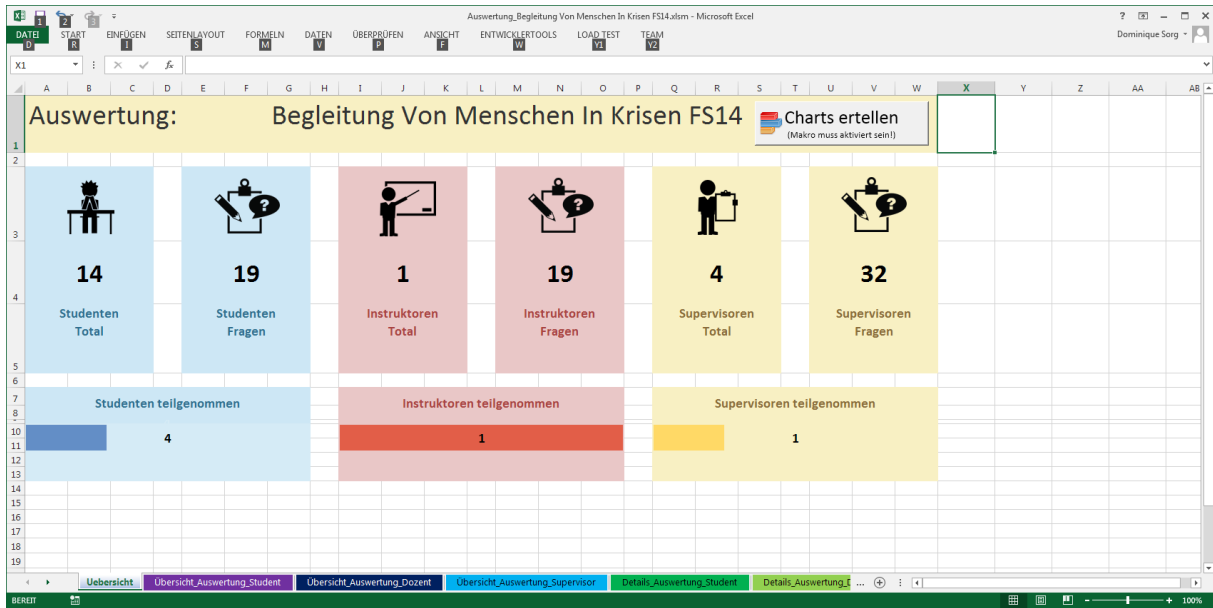


Abbildung 19 Auswertung als Excel

Die Auswertung kann als Excel-Datei heruntergeladen werden. Mit Hilfe eines selbstgeschriebenen VBA Makros werden die Diagramme und Berechnungen automatisch generiert.

Fragen vergleichen

Auswahl 1

Umfrage: Begleitung Von Menschen In Krisen FS14

Rolle:

Frage:

Auswahl 2

Umfrage:

Rolle:

Frage:

Abbildung 20 Korrelation von 2 Fragen

Zudem können Korrelationen zwischen zwei Fragen generiert und als Excel heruntergeladen werden.

	A	B	C	D	E	F	G	H
1	Korrelation							
2								
3	Übersicht		So war es					
4								
5	Umfrage X:	Multimorbidität 2. Sem. FS2014						
6	Frage X:	Ein Bezug zur beruflichen Praxis wurde hergestellt.						
7	Rolle X:	Student						
8								
9	Umfrage Y:	Multimorbidität 2. Sem. FS2014						
10	Frage Y:	Der LTT Praxis-Tag war insgesamt unterstützend für die Entwicklung meiner beruflichen Kompetenzen.						
11	Rolle Y:	Student						
12								
13								
14	Werte							
15	Korrelationskoeffizient (R)	0.90508776	R					
16	Bestimmtheitsmass (R²)	0.819183853	R²					
17	Mittelwert X (xm)	0.9375	xm	xm liegt in der Nähe von:		Trifft eher zu		
18	Mittelwert Y (ym)	1.1875	ym	ym liegt in der Nähe von:		Trifft eher zu		
19								
20								
21	Interpretation Korrelationskoeffizient							
22	negativer Zusammenhang	R = -1	oder -100%					
23	kein Zusammenhang	R = 0	oder 0%					
24	positiver Zusammenhang	R = 1	oder 100%					
25								
26	Interpretation Bestimmtheitsmass							
27	kein linearer Zusammenhang	R ² = 0	oder 0%					
28	perfekt linearer Zusammenhang	R ² = 1	oder 100%					

Abbildung 21 Korrelation als Excel

Die generierte Excel Datei berechnet mit Hilfe eines selbstgeschriebenen VBA Makros die Korrelation aus den Rohdaten und zeigt diese als Diagramm an.

6.2 Optimierungen

In diesem Kapitel wollen wir die Ideen und Optimierungen aus unserer Sicht festhalten, für deren Umsetzung die Zeit fehlte.

Die entwickelte Applikation deckt alle definierten Anforderungen der OdA G ZH. Nichts desto trotz sehen wir Verbesserungspotenzial in den folgenden Bereichen:

Umfrage bearbeiten

Die Umfragen können erstellt und durchgeführt werden. Bearbeitungen der erstellten Umfragen werden in diesem Release nicht unterstützt.

Umfrage Titel eindeutig

Der Titel der Umfrage muss eindeutig sein. Da wir uns entschieden haben, Umfragen zu archivieren, statt diese endgültig zu löschen, bleiben die Titel vorhanden und können nicht doppelt hinzugefügt werden. Momentan erhält die Umfrage von der Leiterin den Namenszusatz FS<Jahr> oder HS<Jahr>.

Export als PDF

Der Export als Excel ist eine gute Variante, um mit Diagrammen weiter zu arbeiten. Die Rohdaten und Diagramme benötigen jedoch sehr viel Platz, was meist für den Druck nicht ausreicht. Deshalb würde sich eine PDF-Variante besonders gut eignen, wenn die Diagramme nicht mehr weiter bearbeitet werden müssen.

7. Problemlösungen

Bei der Umsetzung der Web Applikation tauchten unterschiedliche Probleme auf, die es zu lösen galt. In den folgenden Kapiteln gehen wir auf die Probleme ein und beschreiben, wie wir diese gelöst haben.

7.1 Technologie

Das Webprojekt soll mit OSS (Open Source Software) realisiert werden. Welche Technologie dabei eingesetzt wird, ist uns Entwicklern überlassen. Damit wir für das Web Projekt die richtigen Technologien wählen können, müssen zuerst die Anforderungen analysiert werden.

- Open Source (Wo immer möglich)
- Browser unabhängig
- E-Mail Support für die Benachrichtigung der Teilnehmer
- Excel oder CSV Support
- Fragebögen sollen importiert werden können
- Aussagekräftige Auswertung der Umfrage

Im Verlauf der Bachelorarbeit war uns schnell klar, welche Technologien wir im Rahmen dieser Arbeit verwenden wollen. Im Folgenden eine Auflistung davon:



Abbildung 22: Verwendete Technologien

TeachEval wurde im Backend Server mit Java entwickelt. Dieser Bereich stellt mit Jersey eine REST-Schnittstelle für den Client zur Verfügung. Zur Kommunikation zwischen Java und PostgreSQL wurde Hibernate verwendet. Dieses Framework baut auf JPA und JDBC auf. Im Bereich Client wurde auf AngularJS und Bootstrap für das Responsive Design gesetzt. Damit kann eine breite Palette an Geräten, unabhängig ob PC, Tablet oder Smartphones, abgedeckt werden.

7.2 Kundennahe Lösung

Bei den Sitzungen mit unserer Auftraggeberin haben wir gemerkt, dass Sie viel mit Excel arbeitet. Sie erfasst die Teilnehmerliste und erstellt Auswertungen in Form von Diagrammen mit Excel. Darum legten wir einen speziellen Fokus für eine gute Integration einer Excel Library für Java. Zudem sollte

für den Semester Report die Möglichkeit bestehen, die exportierten Auswertungen und die Korrelationen ausgewählter Fragen in Form von Diagrammen zu bearbeiten. Dies ist ein weiteres Argument, weshalb wir uns für eine Lösung mit Excel statt mit CSV entschieden haben.

7.3 Benutzerfreundlich

Aus dem Software Qualitätsmanagement-Dokument wird ersichtlich, dass für die Messung der Benutzerfreundlichkeit Usability Tests in Form von Online Umfragen durchgeführt wurden. Mit den gestellten Fragen, versuchten wir den Einsatz von TeachEval aus unterschiedlichen Blickwinkel nachzuvollziehen. Die Studenten nahmen mehrheitlich mit dem Smartphone teil und konnten diese effizient innerhalb max. 10min ausfüllen. Die Auswertungen wurden mit den ISO 9241-110 Kriterien wie *Verfügbarkeit*, *Aufgabenangemessenheit*, *Übersichtlichkeit*, *Selbstbeschreibungsfähigkeit*, *Erwartungskonformität*, *Fehlertoleranz*, *Erlernbarkeit*, *Individualisierbarkeit* oder *Steuerbarkeit* bewertet. Mit den vorhandenen Daten konnten wir die vereinbarte Benutzerfreundlichkeit aus der Anforderungsspezifikation von 60% erfüllen.

8. Schlussfolgerungen

Entstanden ist eine moderne und dynamische Web Applikation namens TeachEval, die trotz der aus dem Kapitel 6.2 *Optimierungen* aufgelisteten Verbesserungsmöglichkeiten eine gute Basis schafft, um das Qualitätsmanagement der OdA G ZH zu vereinfachen. Der gesamte manuelle Workflow wurde automatisiert. Nur die Umfragen müssen noch von Hand in einer Excel-Vorlage erstellt und hochgeladen werden. Anschliessend überprüft TeachEval automatisch, ob die Umfrage gestartet, beendet oder ob E-Mail Einladungen zu einer Umfrage als Link an die Teilnehmer verschickt werden müssen. Nachdem die meisten Teilnehmer die Fragebögen ausgefüllt haben, kann die Leiterin oder der/die Dozent/in die Umfrage auch manuell beenden, um als Nächstes die Auswertungen anzusehen. Da die Auftraggeberin viel mit Microsoft Office, darunter Word und Excel arbeitet, können die Auswertungen per Mausklick direkt als Excel exportiert werden. Mittels selbst entwickelten Makros werden die Diagramme beim Öffnen der Excel-Datei automatisch generiert und können für den weiteren Gebrauch verwendet werden. Auch kann sie Korrelationen auf zwei Fragen durchführen und diese ebenfalls als Excel exportieren. Neue Umfragen können durch bestehende Umfragen als Excel einfach überschrieben und wieder hochgeladen werden.

Mit der Integration von Apache POI konnte eine kundennahe Lösung mit Excel realisiert werden. Jedoch muss auch erwähnt werden, dass Apache POI eventuell nicht das beste Excel Framework ist, da es viel Memory für die Generierung der Excel Dateien benötigt, was wir erst während der Entwicklung erkannten. Mit Hilfe von Memory Analysis Programme, wie Visual VM, konnte eine stabile Applikation realisiert werden.

Für die Realisierung war Java Jersey und AngularJS eine gute Wahl. Da dies unser erstes Projekt mit den gewählten Technologien ist, können sicherlich noch Optimierungen im Programmcode durchgeführt werden. Nichts desto trotz sind wir mit dem Endergebnis sehr zufrieden und würden diese Technologien für weitere Projekte nochmals wählen. Serverseitig besteht auch ein grosser Reiz bei einem weiteren Projekt das Java Spring Framework einzusetzen.

Wir hoffen, mit dem Abschluss dieser Arbeit den Support für die OdA G ZH aufrecht zu erhalten und eventuell Verbesserungen und Erweiterungen auch nach dieser Arbeit selbst realisieren und anbieten zu können.

9. Abbildungsverzeichnis

Abbildung 3 IST Workflow	17
Abbildung 4 SOLL Workflow	18
Abbildung 5 Endprodukt.....	23
Abbildung 6 Startseite TeachEval.....	23
Abbildung 7 Profil.....	24
Abbildung 8 Benutzerverwaltung.....	24
Abbildung 9 Umfrageverwaltung	25
Abbildung 10 Umfrage als Excel importieren.....	25
Abbildung 11 Umfrage als Excel importieren.....	26
Abbildung 12 An Umfrage teilnehmen.....	26
Abbildung 13 Umfrage ausfüllen.....	27
Abbildung 14 Umfrage beendet	28
Abbildung 15 Umfrage bereits geschlossen.....	28
Abbildung 16 Übersicht aller Umfragen.....	28
Abbildung 17 Übersicht der Auswertung	29
Abbildung 18 Detaillierte Auswertung auf Fragen.....	29
Abbildung 19 Auswertung als Excel.....	30
Abbildung 20 Korrelation von 2 Fragen.....	30
Abbildung 21 Korrelation als Excel.....	31
Abbildung 22: Verwendete Technologien.....	32

10. Tabellenverzeichnis

Tabelle 1 Produktevergleich von SurveyMonkey (Pricing SurveyMonkey, 2014).....	20
Tabelle 2 Produkteübersicht Lime Survey (Pricing LimeService, 2014)	20
Tabelle 3 Chancen / Gefahren von TeachEval.....	21

TeachEval

Web App für die Unterrichtsevaluation

Auswertung

Suche

The screenshot shows the 'Auswertung' (Evaluation) section of the TeachEval web application. It features a search bar and several filter dropdowns for 'Umfrage' (Survey), 'Kurs' (Course), and 'Status'. Below the filters is a table listing surveys with columns for ID, Titel (Title), Status, Teilgenommen / Total (Participants / Total), Start Datum (Start Date), End Datum (End Date), and Aktion (Action). The table contains three rows of data.

ID	Titel	Status	Teilgenommen / Total	Start Datum	End Datum	Aktion
9	Ethik 2 Sem FS2014	LAEUFT	0 / 14	05.06.2014	06.06.2014	[Icon]
8	Usability Test - Juni14	LAEUFT	0 / 15	04.06.2014	13.06.2014	[Icon]
	Kommunikation2 FS2014	ABGESCHLOSSEN	9 / 9	23.05.2014	23.05.2014	[Icon]

Schlussbericht (SCB)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Einführung

1.1 Zweck

Dieses Dokument beschreibt die Erfahrungen der Teammitglieder von TeachEval bei ihrer Arbeit. Diese werden mit je einem persönlichen Bericht und einer (selbst-)kritischen Reflexion beschrieben.

2. Zielerreichung

Während der Bachelorarbeit haben wir uns folgende persönlichen Ziele gesetzt:

- Aufwandschätzung aufstellen und einhalten
- Die Risiken einschätzen und bewältigen
- Eine kundennahe Lösung planen und realisieren
- Die Kommunikation im Team, mit unserem Betreuer und mit der Auftraggeberin meistern
- Eine komplexe Webapplikation realisieren
- Neue Projekt- und Programmiererfahrungen sammeln sowie neue Technologien kennen lernen.

Grundsätzlich konnten wir unsere Ziele alle erreichen. Das Endprodukt kann produktiv eingesetzt und betrieben werden. Da wir in der Studienarbeit mit einer reinen serverseitigen Technologie, namens ZF2 (Zend Framework 2) gearbeitet haben, wollten wir in der Bachelorarbeit eine eher Client orientierte Technologie einsetzen. Die Inhalte sollten dynamisch geladen werden. Dies wäre bei einer serverseitigen Technologie nicht möglich, da bei jeder Aktion des Besuchers die Seite neu geladen wird.

Trotz der Erfahrungen aus der Studienarbeit konnten wir als Team viele neue Kenntnisse und Erfahrungen mitnehmen. Zudem waren die eingesetzten Technologien für uns Neuland und wir mussten uns erst einarbeiten. Wir sind zudem sehr zuversichtlich, dass die nächsten Projekte mit den gewählten Technologien wie AngularJS und Java Jersey besser werden. Zudem sehen wir einen grossen Reiz für das nächste Projekt mit Spring statt mit Jersey zu arbeiten.

3. Persönliche Erfahrungen

3.1 Dominique Sorg

In meiner diesjährigen Bachelorarbeit ging es darum für die OdA G ZH eine Webapplikation namens TeachEval zu implementieren. Ich konnte während der Bachelorarbeit neue Projekt- und Programmiererfahrungen sammeln, aber auch neue Technologien wie das AngularJS, Jersey, Apache POI oder Hibernate kennenlernen. Zudem lernte ich im Webbereich die Java Welt mit dem Java Jersey Framework in Kombination mit Maven und Tomcat 7 kennen. Mir machte das Arbeiten mit dem Java Jersey Framework richtig Spass. Durch die Annotation konnte einfacher und übersichtlicher Code geschrieben werden. Die Zusammenarbeit mit der Auftraggeberin erwies sich als Herausforderung, da Sie häufig eine Umfrage kurz vor dem stattfindenden Kurs hinzufügen wollte. Somit gab es keine Reservezeit mehr, falls ein Problem auftrat. Wir merkten ziemlich schnell, dass die Benutzerfreundlichkeit laufend verbessert werden musste, da wir Entwickler die Webapplikation anders wahrnehmen. Die Besprechungen mit unserem Betreuer waren immer konstruktiv und sehr angenehm. Er nahm sich immer die benötigte Zeit um unsere Fragen zu beantworten und gab uns Verbesserungsvorschläge.

3.1.1 Verlauf

Während der Bachelorarbeit war ich sehr motiviert, da wir eine Webapplikation entwickeln konnten, die auch produktiv eingesetzt wird. Um die Fragebögen oder die Antworten der Teilnehmer vernünftig abzuspeichern, um daraus aussagkräftige Auswertungen zu machen, wurde zu Beginn des Projektes viel Zeit in die Datenbankmodellierung investiert. Viele Anforderungen konnten wir erst im Gespräch mit der Auftraggeberin erkennen. Aus diesem Grund setzten wir von Anfang an auf eine skalierbare Lösung. Anschliessend entwickelten beide Teammitglieder am Backend an der REST Schnittstelle. Im Laufe der Entwicklung ging ich immer mehr in den Clientteil mit AngularJS über, was auch mein Zuständigkeitsbereich wurde. Als Erstes entwickelte ich eine Token basierte Authentisierung für den Login Bereich. Der Token wird bei jedem Request im HTTP Header mitgeschickt. Zusätzlich wird das Passwort clientseitig mit einem SHA256 gehashed und übertragen. In einem zweiten Schritt wurde die ganze Benutzerverwaltung, Umfrageverwaltung sowie die Teilnahme an Umfragen realisiert. Der weitere Verlauf der Arbeit bestand darin, die Evaluation mit Diagrammen zu visualisieren. Die Auswertung einzelner Umfragen kann auf einer Detailansicht betrachtet werden. Nebst dem Frontend, gab es immer wieder auch Einsätze im Backend wie z.B. für das Exportieren von Excel-Dateien mit Apache POI.

3.1.2 Probleme

Die Kompatibilität mit dem Internet Explorer erwies sich im Laufe des Projektes als eine echte Herausforderung. Im Frontend traten immer wieder Cache Probleme oder Probleme mit der Anzeige von CSS Style auf. Diese konnten wir mit den richtigen Kompatibilitätsbibliotheken und einem serverseitigen Cache Control lösen. Da die Bachelorarbeit mein erstes Projekt mit AngularJS war, musste viel Zeit für Einarbeitung aufgewendet werden. Zudem entschieden wir uns für den Einsatz von requireJS für die Verwaltung von Abhängigkeiten um eine bessere Projektstruktur zu erhalten.

Von der Auftragsgeberin erhielten wir ab Mitte Mai keine Antwort oder Feedback mehr, bis wir erfahren hatten, dass sie drei Wochen krank und in den Ferien war. Dies erschwerte unsere Arbeit sehr, da wir zum Auswertungsteil nie einschätzen konnten, ob es den Bedürfnissen der OdA G ZH wirklich entspricht. Zwei Tage vor der Abgabe erhielten wir ihr Feedback!

3.1.3 Fazit

Ich persönlich bin sehr zufrieden mit dem Ergebnis. Natürlich können immer noch Verbesserungen und Erweiterungen implementiert werden, die auch die Benutzerfreundlichkeit verbessern. Ich arbeite gerne an Projekten, welche auch im realen Umfeld eingesetzt werden können. Ich wäre auch bereit für die OdA G ZH das Projekt TeachEval weiterzuführen.

4. Persönliche Erfahrungen

4.1 Benjamin Kehl

Für meine Laufbahn als Informatiker suchte ich nach einer Bachelorarbeit, bei der ich mich in den Bereich Webapplikation vertiefen kann. Durch die erfolgreiche Studienarbeit, bei der auch eine Webapplikation entwickelt werden musste, wollte ich unbedingt wieder etwas Ähnliches durchführen. Deshalb erfüllte „TeachEval – Webapp für die Unterrichtsevaluation“ meine Ansprüche als Thema für die Bachelorarbeit. In der Studienarbeit machte ich gute Erfahrungen mit Zend Framework 2, welches ein objektorientiertes MVC-Framework für PHP darstellt. Dennoch fand ich anhand der Anforderungen für die Bachelorarbeit dieses Framework nicht als gut geeignet, da für die Webapplikation anhand verstärkter Integration von JavaScript mehr Interaktion seitens des Clients erforderte. Deshalb wollten wir die Webapplikation mit einer guten Mischung aus Server- und Clientteil entwickeln und so entschieden wir uns für Jersey und für AngularJS. Dieser Entscheid erwies sich als erfolgsversprechend. Wir konnten eine gute Webapplikation entwickeln, welche für die Bedürfnisse des Kunden zugeschnitten ist.

4.1.1 Verlauf

Zu Beginn des Projektes definierten wir unsere Verantwortungen und Aufgabenbereiche. Da Dominique sich bereits früh mit AngularJS auseinander gesetzt hatte, übernahm ich den Backend-Bereich und stellte die Ressourcen zum Client mit Jersey und die Verbindung zur Datenbank mit Hibernate zur Verfügung. Dank unseres vierseitigen Fragebogens konnten wir bereits am ersten Treffen mit unserer Auftraggeberin wichtige Informationen sammeln, wie der Stand des momentanen Workflows ist, welche Technologien im Einsatz sind und welche Wünsche sie unbedingt realisiert haben möchte. Einer ihrer Wünsche war z.B. der Export als Excel mit Rohdaten und Diagrammen, was wir später mit Apache POI und mit VBA-Makro realisierten. An diesem Treffen haben wir auch erfahren, dass eine erste Version bereits anfangs Mai produktiv an einem Kurs getestet werden muss. Dadurch haben Dominique und ich zu Beginn viel Wert auf ein gutes Datenbankschema gelegt. Danach wurde Hibernate als die Persistence-Schicht, Jersey für die REST-Schicht und Service als die Zwischenschicht mit Apache POI entwickelt. Ab diesem Standpunkt konnten die Funktionalitäten wie Umfrage oder Benutzer verwalten oder auch die Auswertungen erzeugen, realisiert werden.

4.1.2 Probleme

Die Vorzüge von Hibernate erwiesen sich zu Beginn als sehr praktisch, da er Objekte direkt als Tabellen in der Datenbank abbildet (OR-Mapping). Dennoch benötigte ich mit Hibernate viel Einarbeitungszeit wie Operationen richtig durchgeführt werden. Zu Beginn war das Session-Handling ein Problem, welche Schicht konkret für das Session-Handling verantwortlich ist. Das Verfahren mit Eager-Loading wurde später ebenfalls zu einem Problem, da Eager-Loading jedes Objekt mit seinen Abhängigkeiten zu laden versucht, was schliesslich irgendwann dazu führt, dass die gesamte Datenbank im Cache landet. Dem konnten wir jedoch erfolgreich mit dem Lazy-Loading Verfahren entgegenwirken.

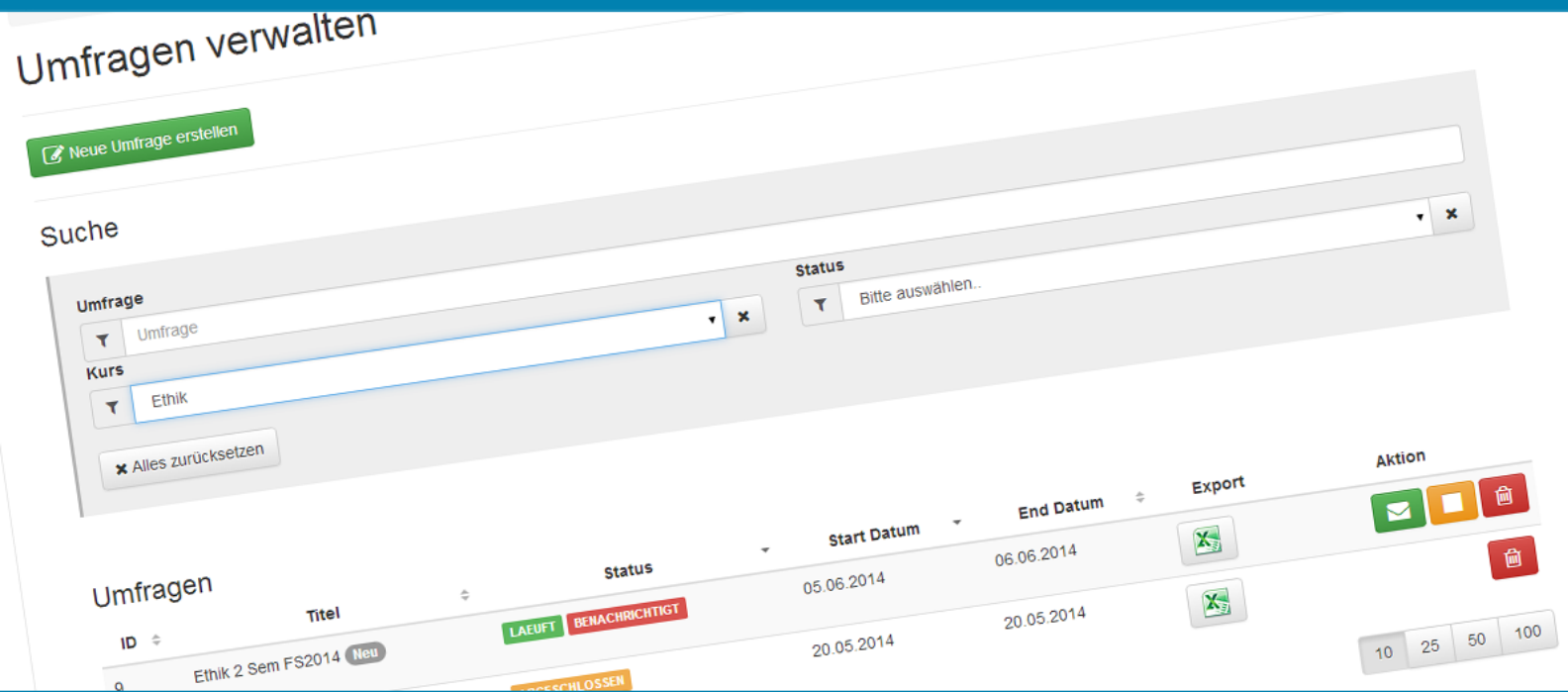
Ab dem letzten Drittel des Projektes empfand ich es als schwierig, dass die Auftraggeberin häufig abwesend war und wir zum Teil bis zu zwei Wochen nichts von ihr hörten. So erhielten wir sehr spät Feedbacks um die Applikation nach ihren Wünschen zu verbessern.

4.1.3 Fazit

Mit viel Engagement, Freude und Zuversicht habe ich die Bachelorarbeit durchgeführt. Eine gute Zusammenarbeit und ein gutes Arbeitsklima unterstützten den Entwicklungsprozess. Dass TeachEval auch produktiv verwendet wird, motivierte mich umso mehr ein gutes Produkt abzuliefern. Ich bin froh, dass wir die Bachelorarbeit erfolgreich abschliessen konnten.

TeachEval

Web App für die Unterrichtsevaluation



Glossar (GLO)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

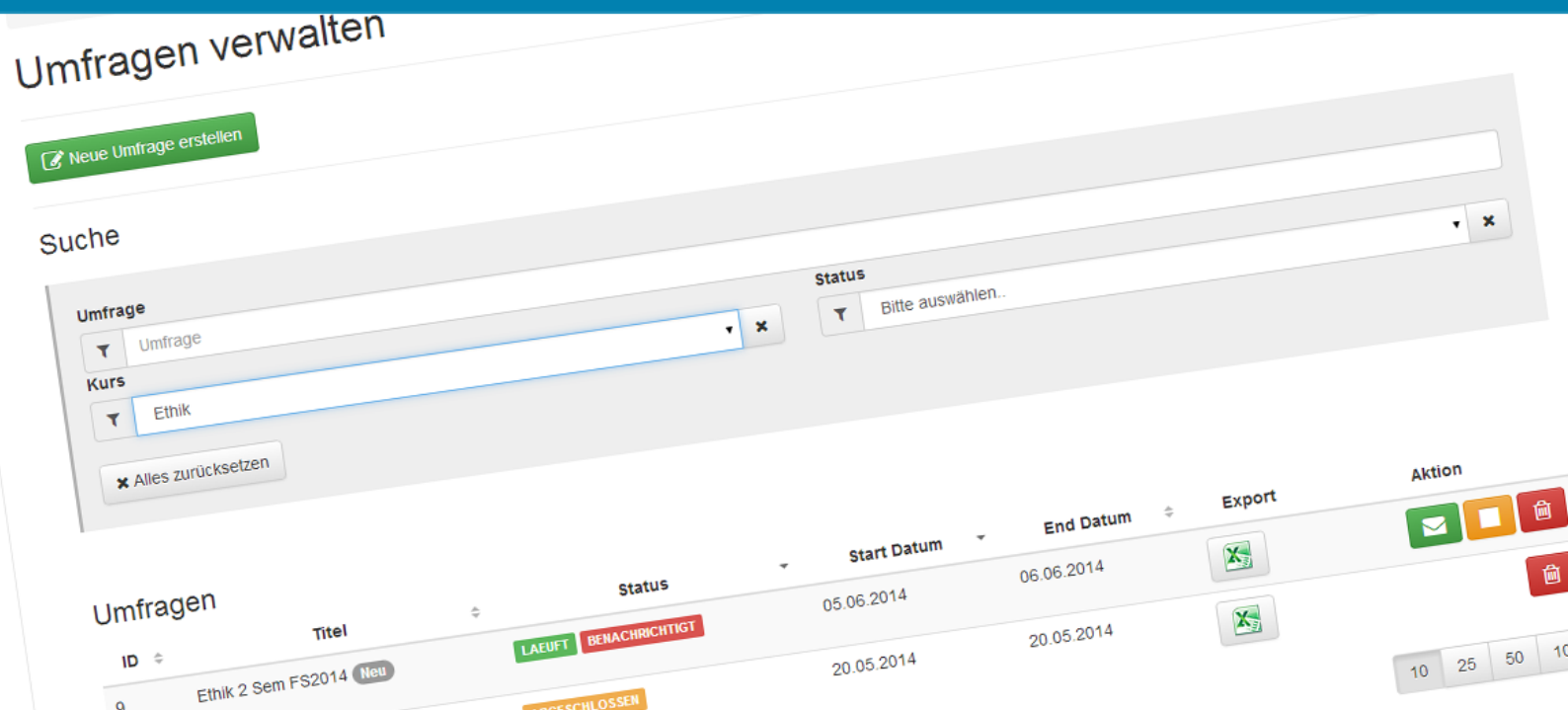
Begriff	Beschreibung
AngularJS	<i>AngularJS</i> ist ein Open Source Framework von Google, welches die Erstellung von browserbasierten Single-Page-Anwendungen mit einem MVC-Modell unterstützt.
API	Als <i>API</i> wird eine Programmierschnittstelle verstanden, der von einem Softwaresystem anderen (externen) Programmen zur Anbindung an das System zur Verfügung stellt. In unserem Fall handelt es sich um die REST-Ressourcen, die für die Kommunikation mit dem Frontend zur Verfügung gestellt werden.
Backend	Das <i>Backend</i> ist näher am System und somit auch näher an der Verarbeitung oder Ausgabe. Bei TeachEval wird hier der Web Service verstanden der auf dem Server läuft.
Benutzeroberfläche	Synonym GUI
Client, Clientseitig	Eine <i>clientseitige</i> Anwendung wird auf dem Rechner des Nutzers ausgeführt. Das Gegenstück eines Clients ist der Server. Bei Web Anwendungen sind es vor allem die Sprachen HTML, CSS, JavaScript die clientseitig ausgeführt werden
CRUD	<i>CRUD</i> umfasst die grundlegenden Operationen Create (Datensatz anlegen), Read oder Retrieve (Datensatz lesen), Update (Datensatz aktualisieren) und Delete oder Destroy (Datensatz löschen).
CSS	CSS Steht für Cascading Style Sheets und ist eine deklarative Sprache für Stilvorlagen von strukturierten Dokumenten (z.B. in HTML oder in XML).
DB	Abkürzung für Database bzw. Datenbank.
DRY	Abkürzung für Don't Repeat Yourself. <i>DRY</i> ist ein Prinzip und besagt, dass Redundanz in Codes vermieden werden sollte (keine Duplizierungen).
Frontend	Das <i>Frontend</i> ist näher am Benutzer und somit auch näher an der Eingabe. Darunter wird bei TeachEval die Benutzeroberfläche gemeint, worüber der Benutzer seine Eingaben tätigen kann.
GIT	<i>GIT</i> ist eine freie Software zur verteilten Versionsverwaltung von Daten.
GUI	Das <i>GUI</i> (Graphical User Interface) ist die grafische Benutzeroberfläche einer Applikation. Es stellt die Schnittstelle mit grafischen Elemente zum System dar.
HSR	Hochschule für Technik Rapperswil (HSR)
HTML, HTML5	<i>HTML</i> steht für Hyper Text Markup Language. Sie dient zur Strukturierung von Inhalten in Webseiten. <i>HTML5</i> ist momentan der aktuellste Standard.
Interaktion	Wird als aufeinander bezogenes Handeln zwischen Mensch und System verstanden.
ISO	<i>ISO</i> steht für International Organization for Standardization und ist verantwortlich für Standardisierungen in verschiedensten Bereichen.
ISO 9126	Das ISO 9126 ist eine empfohlene Richtlinie von ISO, die sich speziell auf Qualitätsmerkmale im Bereich Software richtet.
Jasmine	Jasmine ist ein Unit Test Framework für AngularJS
JavaScript	<i>JavaScript</i> ist eine Skriptsprache auf der Client-Ebene und dient der Entwicklung von dynamischen HTML in Webbrowsern.
JAX-RS	<i>JAX-RS</i> steht für Java API for RESTful Web Services. Es handelt sich um eine Programmierschnittstelle (API) in der Programmiersprache JAVA, das auf die Prinzipien von REST basiert und das Anbieten von Web Services ermöglicht.
JDBC	<i>JDBC</i> steht für Java Database Connectivity. JDBC ist eine Datenbankschnittstelle der JAVA Plattform, die eine einheitliche Schnittstelle zu Datenbanken verschiedener Hersteller bietet und speziell auf relationale Datenbanken ausgerichtet ist.
Jersey	<i>Jersey</i> ist ein JAVA Open Source Projekt, welches auf JAX-RS basiert.

JPA	<i>JPA</i> steht für Java Persistence API. JPA ist eine Schnittstelle für Java-Anwendungen, die die Zuordnung und die Übertragung von Objekten zu Datenbankeinträgen vereinfacht.
JQuery	<i>JQuery</i> ist eine freie JavaScript-Bibliothek.
JUnit	<i>JUnit</i> ist ein Framework zum automatisierten Testen von Java Anwendungen.
KISS	Steht für <i>Keep it Simple</i> und ist ein Prinzip in der Programmierung, das besagt, man solle seine Lösung so einfach wie möglich umsetzen.
Logische Architektur	In der <i>logischen Architektur</i> werden Modelle, Klassen oder Abhängigkeiten innerhalb eines Systems beschrieben
Modifikationen	Synonym für <i>Änderungen</i>
MVC	Steht für <i>Model View Controller</i> und ist ein Pattern, welches für die Strukturierung der Software-Entwicklung weit verbreitet ist. Das MVC unterteilt die Software in die drei Einheiten: Model, View und Controller.
ODA G	Organisation der Arbeitswelt Gesundheit
OdA G ZH	Organisation der Arbeitswelt Gesundheit des Kantons Zürich
Persistieren	<i>Persistenz</i> ist der Fachbegriff für das Verwalten von Daten in einer Software. Dafür werden die Daten in einem nichtflüchtigem Speichermedium, bzw. Datenbank gesichert.
Physische Architektur	In der Physischen Architektur werden die Komponenten, Hardware oder Netze beschrieben.
Postgres, PostgreSQL	Postgres ist eine populäre Open-Source Datenbank.
Produktions-Server	Auf dem produktiven Server läuft eine stabile Version der Applikation. Diese muss eine hohe Verfügbarkeit und Zuverlässigkeit sicherstellen.
Redmine	<i>Redmine</i> ist ein Projektmanagement Tool für die Planung und Zeiterfassung von IT Projekte.
Responsive Webdesign	Unter dem Begriff <i>Responsive Webdesign</i> versteht man einen gestalterischen und technischen Vorgang für die Erstellung von Webseiten. Dabei reagiert es auf Eigenschaften des jeweils benutzten Endgeräts. Der Aufbau der Webseite basiert dann auf den Anforderungen des jeweiligen Gerätes wie Mobile, Tablet, PC, etc. Dies betrifft vor allem Elemente wie die Navigation, Aufteilung von Seitenspalten und Text, wie Anreihungen von Bildern. Responsive Webdesign kann mit den Webstandards HTML5, CSS3 und JavaScript realisiert werden. Ausserdem gibt es auch diverse Frameworks wie beispielsweise Bootstrap von Twitter, die diesem Prinzip nachgehen.
REST	<i>REST</i> steht für Representational State Transfer. Es bezeichnet einen Programmierstil und Software Architekturstil für Webanwendungen. Im Grunde bezeichnet REST die Idee, dass eine URL genau einen Seiteninhalt als Ergebnis einer serverseitigen Aktion darstellt. Im HTTP werden die Operationen GET, POST, PUT, DELETE verwendet.
RESTful	<i>RESTful</i> wird verwendet um Web Services nach der REST Architektur zu realisieren.
RUP	Abkürzung für Rational Unified Process. RUP ist ein Vorgehensmodell in der Softwareentwicklung.
Server, Serverseitig	Eine serverseitige Anwendung wird auf dem Server ausgeführt und die Ergebnisse dem Client zugestellt. Das Gegenstück eines Servers ist der Client. Bei Web Anwendungen sind es vor allem Sprachen wie PHP, Ruby, Java, C die serverseitig zum Einsatz kommen.
Single Page Anwendung	Als Single-page Web Application (SPA), oder Einzelseiten Webanwendung wird eine Website bezeichnet, die aus einem einzigen HTML-Dokument besteht und deren Inhalte dynamisch nachgeladen werden. Diese Art von Website-Design

	steht im Gegensatz zu klassischen Websites, die aus mehreren, untereinander verlinkten HTML-Dokumenten bestehen.
SPA	<i>SPA</i> steht für Single Page Anwendung oder auch Single-page Web Application.
Testserver	Der <i>Testserver</i> wird mit der gleichen Applikation wie der Produktion Server aufgesetzt. Auf dem Testserver werden Tests an die Applikation durchgeführt. Dabei muss die Verfügbarkeit der Applikation nicht sichergestellt werden.
Tomcat	Apache <i>Tomcat</i> ist ein Java Open Source HTTP Web Server die von der ASF (Apache Software Foundation) entwickelt worden ist.
UC	Abkürzung für <i>Use Case</i> .
UML	Die Unified Modeling Language, kurz <i>UML</i> , ist eine grafische Modellierungssprache zur Spezifikation, Konstruktion und Dokumentation von Software-Teilen und anderen Systemen.
URI	Die <i>URI</i> (Unique Resource Identifier) ist ein Identifikation für Web Ressourcen. Diese müssen im Netzwerk eindeutig sein.
URL	Die <i>URL</i> (Uniform Resource Locator) identifiziert und lokalisiert Ressourcen einer Webseite über die Protokolle HTTP oder FTP.
Web App	Kurzform für <i>Web Applikation</i>
Web Applikation	Unter einer Web Applikation (oder auch Webapplikation) versteht man eine Software, die speziell für das Web/Internet konzipiert wurde. Diese Applikationen werden mit Hilfe von HTML, CSS, etc. umgesetzt.
Web Browser	Der Web Browser (oder auch Webbrowser) ist ein Computerprogramm für die Darstellung von Webseiten aus dem World Wide Web (WWW). Die Darstellung von Webseiten wird mit HTML, CSS, JavaScript, Bilder, etc. realisiert. Bekannte Webbrowser sind Internet Explorer, Chrome, Firefox, Safari, Opera und viele mehr.
Web Service	Ein Web Service oder Webdienst ist eine Softwareanwendung, die über ein Netzwerk für die direkte Maschine-zu-Maschine-Interaktion bereitgestellt wird. Jeder Webservice besitzt einen Uniform Resource Identifier (URI), über den er eindeutig identifizierbar ist. Die Kommunikation kann über Protokolle wie HTTP, HTTPS laufen, basierend auf Übertragungsformate wie XML oder JSON

TeachEval

Web App für die Unterrichtsevaluation



Quellenverzeichnis (QUV)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gelesen: Prof. Dr. Olaf Zimmermann

Quellen

- Benutzerfreundlichkeit Wikipedia*. (08. 05 2014). Abgerufen am 04. 03 2014 von <http://de.wikipedia.org/wiki/Benutzerfreundlichkeit>
- Browserstack Webbased Browser Testing*. (2014). Abgerufen am 19. 05 2014 von <http://www.browserstack.com/>
- DBSchema Offizielle Webseite*. (2014). Abgerufen am 14. 05 2014 von <http://www.dbschema.com/>
- Documentation Jersey Test Framework*. (2014). Abgerufen am 26. 05 2014 von <https://jersey.java.net/documentation/latest/test-framework.html>
- Entwicklertagebuch AngularJS Structure*. (2014). Abgerufen am 08. 04 2014 von <http://entwicklertagebuch.com/blog/2013/10/how-to-structure-large-angularjs-applications/>
- Fielding, R. T. (2000). *ICS University of Science*. Abgerufen am 27. 05 2014 von https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf
- Hanselman, S. (2014). *Hanselman HTTP PUT DELETE*. Abgerufen am 22. 05 2014 von <http://www.hanselman.com/blog/HTTPPUTOrDELETENotAllowedUseXHTTPMethodOverrideForYourRESTServiceWithASPNETWebAPI.aspx>
- HelpCenter for creating Charts SurveyMonkey*. (2014). Abgerufen am 04. 06 2014 von http://help.surveymonkey.com/articles/en_US/kb/Can-I-create-custom-charts
- HelpCenter for Questoin Types SurveyMonkey*. (2014). Abgerufen am 01. 06 2014 von http://help.surveymonkey.com/articles/en_US/kb/Available-question-types-and-formatting-options
- Java RESTful Web Service with AngularJS*. (13. 07 2013). Abgerufen am 05. 03 2014 von <http://draptik.github.io/blog/2013/07/13/angularjs-example-using-a-java-restful-web-service/>
- JBoss Documentation Hibernate Session Factory*. (2014). Abgerufen am 23. 05 2014 von <http://docs.jboss.org/hibernate/orm/3.3/reference/en/html/session-configuration.html#configuration-sessionfactory>
- Licence Apache Log4J*. (2014). Abgerufen am 11. 06 2014 von <https://logging.apache.org/log4j/1.2/license.html>
- Licence Apache Math*. (01 2014). Abgerufen am 11. 06 2014 von <http://www.apache.org/licenses/LICENSE-2.0>
- Licence Apache POI*. (01 2014). Abgerufen am 11. 06 2014 von <http://www.apache.org/licenses/LICENSE-2.0>
- Licence Apache Quartz*. (01 2014). Abgerufen am 11. 06 2014 von <http://www.apache.org/licenses/LICENSE-2.0>
- Licence Apache Tomcat*. (2014). Abgerufen am 11. 06 2014 von <http://www.apache.org/licenses/LICENSE-2.0>
- Manual Question Types in Lime Survey*. (2014). Abgerufen am 04. 06 2014 von http://manual.limesurvey.org/Question_types/de
- Messen von Usability*. (2007). (Uni Bremen) Abgerufen am 11. 03 2014 von http://www.informatik.uni-bremen.de/st/lehre/Re-Architect/Seminar/Praesentationen/04_Messen_von_Usability.pdf
- MSDN HTTP-Verbs*. (2014). Abgerufen am 06. 06 2014 von <http://msdn.microsoft.com/en-us/library/dd541471.aspx>
- Oda G ZH Offizielle Webseite*. (2014). Abgerufen am 07. 06 2014 von <http://oda-g-zh.ch/>
- Pingdom Website Speed Test*. (2014). Abgerufen am 30. 05 2014 von <http://tools.pingdom.com/fpt/>
- Pricing LimeService*. (2014). Abgerufen am 04. 06 2014 von <http://www.limeservice.com/de/preise>
- Pricing SurveyMonkey*. (2014). Abgerufen am 01. 06 2014 von https://de.surveymonkey.com/pricing/details/?t=sm_g_p&ut_source=header
- RequireJS Offizielle Webseite*. (2014). Abgerufen am 31. 05 2014 von <http://requirejs.org/>
- REST Wikipedia*. (26. 05 2014). Abgerufen am 09. 06 2014 von https://de.wikipedia.org/wiki/Representational_State_Transfer

Vision der Oda G ZH. (2014). Abgerufen am 25. 02 2014 von http://www.oda-g-zh.ch/de-ch/organisation_1/vision_und_leitbild.html

Wireframe Tool Balsamiq Mockups. (2014). Abgerufen am 25. 02 2014 von <http://balsamiq.com/products/mockups/>

Anhang

TeachEval

Web App für die Unterrichtsevaluation

Herzlich Willkommen bei TeachEval

TeachEval ist eine Webapplikation für die Erstellung von Umfragen und Auswertungen, die speziell für die Bildungsveranstaltungen der Oda G ZH im Bereich Gesundheit gew.
Umfrage der Oda G Zürich teilnehmen zu können, wird eine Umfragekennung, die Sie per Post, per Email oder persönlich erhalten haben, benötigt. Die Umfragen werden Anonym durch
Teilnehmer von der LTT Praxis Psychiatrie verwenden bitte den folgenden Link "Zur Umfrage".
Weitere Informationen finden Sie direkt auf der Oda G ZH Seite.

LTT Praxis Allgemein



LTT Praxis Psychiatrie



Projektplan (PMP)

Autor(en):	Dominique Sorg, Benjamin Kehl
Betreuer:	Prof. Dr. Eduard Glatz
Projektpartner:	Oda G Zürich
Experte:	Roberto Pajetta
Gegenleser:	Prof. Dr. Olaf Zimmermann

1. Einführung

1.1 Zweck

Dieses Dokument beschreibt den Projektplan für die Bachelorarbeit TeachEval Web App.

1.2 Gültigkeitsbereich

Der Projektplan gilt als Grundlage des Projektes und ist daher über die gesamte Projektdauer gültig. Damit der Projektplan immer auf dem neusten Stand ist, wird er laufend angepasst.

1.3 Übersicht

Im Kapitel *Projekt Übersicht* soll die aktuelle Problematik für das Thema genauer beschrieben und aufgezeigt werden. Zudem werden der Zweck und das Ziel des Projektes beschrieben. Anschliessend wird aufgezeigt, was der Lieferumfang ist und welche Einschränkungen es gibt.

Im Kapitel *Projektorganisation* werden die Mitarbeiter und deren Motivation für das Projekt vorgestellt und als Organisationsstruktur aufgezeigt.

Wichtige Themen wie unser Vorgehen, Zeitplanung wie Meilensteine, mögliche Risiken und Schaden, Arbeitspakete, usw. werden in den Kapiteln *Management Abläufe*, *Risikomanagement* und *Arbeitspakete* näher vorgestellt. Gegen Ende des Dokumentes werden die Infrastruktur und Qualitätsmassnahmen des Projektes TeachEval in den entsprechenden Kapiteln *Infrastruktur und Qualitätsmassnahmen* beschrieben.

2. Projekt Übersicht

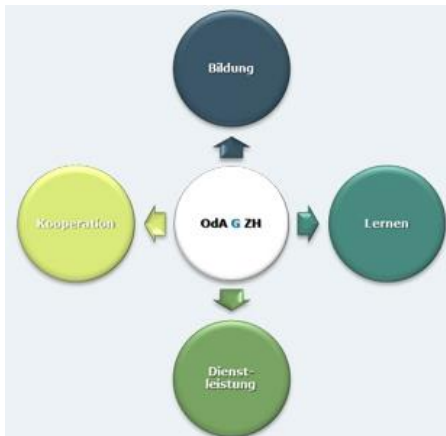


Abbildung 24 Oda G Vision & Leitbild
(Vision der Oda G ZH, 2014)

Die Organisation der Arbeitswelt Gesundheit Zürich (Oda G) ist der Branchenverband für Berufsbildung im Gesundheitswesen des Kantons Zürich. Sie ist der Ansprechpartner in allen berufspolitischen Bildungsfragen und übernimmt die Verantwortung für die kompetente Durchführung überbetrieblicher Kurse, Lerntage Training und Transfer (LTT) und ähnliche Bildungsveranstaltungen auf allen Bildungsstufen. Die Oda G übernimmt auch die fachliche und organisatorische Unterstützung rund um die praktische Ausbildung (Oda G ZH Offizielle Webseite, 2014).

Die Kurse im LTT werden von den Lernenden und Dozierenden wie auch deren Supervisoren evaluiert. Zurzeit erfolgen diese Evaluationen mit Hilfe von gedruckten Formularen, die in manueller Arbeit aufwendig ausgewertet werden müssen. Deshalb soll eine Webapplikation namens TeachEval

entwickelt werden, die sowohl die Befragung als auch deren Auswertung automatisiert. Die Webapplikation soll Fragebögen und Auswertungen mit Hilfe eines Webbrowsers in benutzerfreundlicher Art auf PCs, Tablets als auch auf Smartphones anzeigen können. Der Auswertungsteil soll mit ansprechenden und verschiedenen Diagrammen visualisiert und als Excel oder Word Export bereitgestellt werden. Zudem soll sie auf ausgewählte Fragen korrelieren, das heisst, die Kovarianz numerischer Auswertungsergebnisse berechnen.

2.1 Zweck und Ziel

Mit Hilfe von TeachEval soll der momentane Zeitaufwand des bestehenden Arbeitsprozesses stark vermindert werden, indem die Webapplikation die Befragung als auch deren Auswertung automatisiert. Zudem soll die Evaluation auf den heute gängigen Geräten wie Smartphone, Tablets oder PCs zugänglich sein und benutzerfreundlich (siehe Anforderungsspezifikation) angezeigt werden.

2.2 Einsatz

TeachEval wird an der Hochschule für Technik Rapperswil (HSR) entwickelt. Oda G ZH erneuert zurzeit Ihre Server-Infrastruktur. Wenn der benötigte Server bei Oda G ZH verfügbar ist, wird die Webapplikation migriert. Bis dahin wird die Applikation auf einem HSR oder externen Server laufen.

2.3 Lieferumfang

Folgende Punkte werden am Schluss der Bachelorarbeit abgeliefert:

- Webapplikation TeachEval zur Verwaltung und Ausführung von Umfragen und Auswertungen
 - o Sprache: Deutsch
 - o Die Umfrage wird automatisch über eine Excel-Datei erstellt. Die Excel-Datei enthält mehrere Arbeitsmappen wie Fragen, Teilnehmern, Benutzerrollen (z.B. Dozent, Student, Supervisor oder Leiter) usw.
 - o Auswertung in Form von Diagrammen anzeigen
 - o Auswertung als Excel-, Word- oder als PDF-Export angeboten
→ Weiterverarbeitungsmöglichkeit wird durch Excel/Word bereitgestellt (Diagramme nicht als Bilder exportieren).
 - o Korrelation einzelner Fragen oder Umfragen

- vernünftige Kompatibilität gewährleisten*
- gute Benutzerfreundlichkeit gewährleisten*
- Benutzerhandbuch zur Installation und Verwendung der Webapplikation
- Technischer Bericht für die Beurteilung der eingesetzten Technologien
- Vollständige Software Engineering Dokumente

** siehe Anforderungsspezifikation für genauere Definition*

2.4 Annahmen und Einschränkungen

Für den Zugang zu der Benutzer- und Umfrageverwaltung ist ein Login notwendig, um die verschiedenen Benutzer auseinander zu halten.

3. Projektorganisation

Die Bachelorarbeit wird von Dominique Sorg und Benjamin Kehl durchgeführt. Jedes Mitglied hat seinen eigenen Verantwortlichkeits- und Aufgabenbereich. Die Arbeitspakete werden je nach Aufgabenbereich und Verantwortlichkeit zugeteilt. Dabei können auch nicht zugewiesene Arbeitspakete bearbeitet werden. Eduard Glatz ist über der gesamten Bachelorarbeit unser Projektbetreuer und die direkte Ansprechperson. Während Brigitte Keller unsere Auftraggeberin der OdA G ZH ist.

3.1 Organisationsstruktur

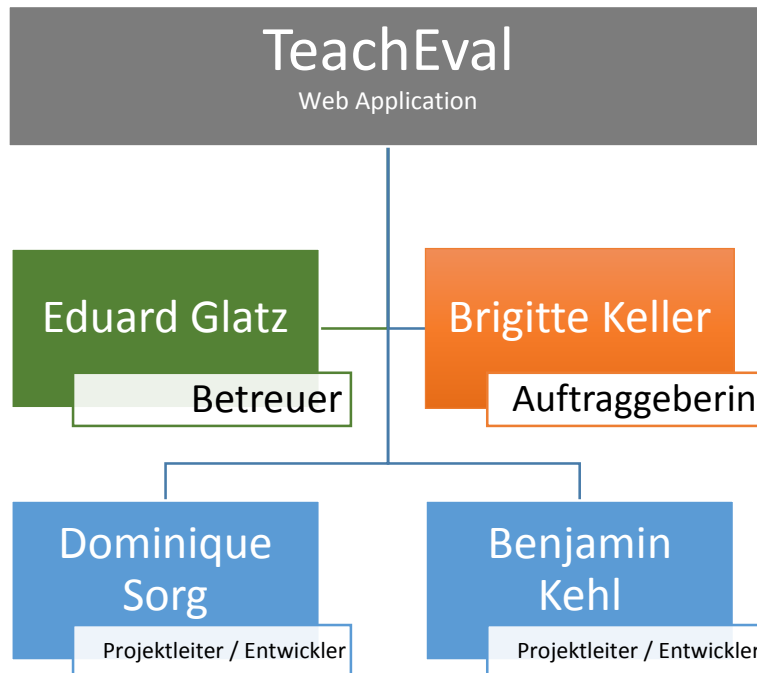


Abbildung 25 Organisationsstruktur

3.2 Team

Benjamin Kehl	
Motivation	Ich finde es sehr spannend mich mit neuen Technologien auseinanderzusetzen. Ich kann durch dieses Projekt mein Wissen über das Umsetzen von Internetseiten vertiefen. Durch die Webapplikation kann beim Auswerten der Evaluationen viel Zeit gespart werden. Das empfinde ich als motivierend. Zudem kann ich mir durch das Projekt neues Wissen aneignen und dies gleich umsetzen. Mein Ziel ist es, eine hilfsbereite und brauchbare Webapplikation anbieten zu können und ich freue mich bereits auf die neuen Herausforderungen in diesem Projekt.
Aufgabenbereich	<ul style="list-style-type: none"> - Datenbank (JDBC, JPA, Hibernate) - Backend (Rest, Web Services, Hibernate, PostgreSQL)
Verantwortlichkeiten	<ul style="list-style-type: none"> - Protokollierung der Meetings - Überwachung Qualität der Dokumente und Code

Tabelle 4 Teammitglied Benjamin Kehl

Dominique Sorg	
Motivation	Die Entwicklung an einem komplexen Webprojekt stellt mich vor neuen Herausforderungen und ermöglicht meiner Kreativität freien Lauf zu lassen. Des Weiteren lerne ich neue Technologien und neue Umsetzungskonzepte im Webbereich kennen. Eine Webapplikation zu entwickeln, die anschliessend auch produktiv eingesetzt wird motiviert mich sehr.
Aufgabenbereich	<ul style="list-style-type: none"> - Frontend (AngularJS, Bootstrap, Charts, Layout / Design, Login) - Backend (User Authentication & Authorization, Rest, Apache POI)
Verantwortlichkeiten	<ul style="list-style-type: none"> - Koordination des Projektes - Zeitmanagement (Planung des Projektes wie Arbeitspakete, Zeitplan, ...)

Tabelle 5 Teammitglied Dominique Sorg

3.3 Externe Schnittstellen

Während unserem Projekt sind nur Eduard Glatz und Brigitte Keller als externe Personen involviert. Da Brigitte Keller unsere Auftraggeberin ist, kann Sie uns helfen Usability Tests mit Mitarbeiter durchzuführen, von denen wir uns Feedback erhoffen.

4. Management Abläufe

In der Grobplanung wurden zuerst die Meilensteine definiert und in einem weiteren Schritt in unser Projektmanagement Tool namens Redmine erfasst. Die Grobplanung leitet sich von den Bedürfnissen und Prioritäten der OdA G ZH (siehe Anforderungsspezifikation) ab, die von Eduard Glatz und Brigitte Keller repräsentiert werden. In der Feinplanung wurden Arbeitspakete zu den jeweiligen Meilensteinen erfasst und an den jeweiligen Teammitgliedern zugeteilt.

4.1 Kostenvoranschlag

Für die Bachelorarbeit sind 17 Wochen geplant. Sie beginnt mit dem Kickoff-Meeting am Dienstag den 18.02.2014 und endet mit der Abgabe der Bachelorarbeit am Freitag den 13.06.2014.

Pro Teammitglied muss mindestens 360 Arbeitsstunden im Sinne der BA investiert werden. Daraus rechnen wir mit einem Arbeitsaufwand von 21.2 Stunden ($12 \text{ ECTS} * 30 \text{ h} / 17 \text{ Wochen}$) pro Teammitglied und pro Woche. Wir gehen von einer maximalen Obergrenze an Arbeitsstunden von 20% aus, das heisst zusätzliche 144 Stunden.

Sollte der Projektumfang bis Abgabe schneller als geplant fertiggestellt werden, so kann die Applikation durch optionale Funktionen ergänzt werden.

4.2 Zeitliche Planung

Unsere Bachelorarbeit besteht aus 9 Meilensteinen, die zwingend einzuhalten sind. Zusätzlich werden wir wöchentlich mit unserem Betreuer ein Meeting führen. Mit der Auftraggeberin führen wir, nach Bedarf, jede zweite Woche ein Meeting. Die Reviews werden immer nach Ende des letzten Meilensteins geführt.

Falls wir als Team frühzeitig schon realisieren, das wir ein Meilenstein nicht einhalten können, dann müssen wir die Arbeit mit Prioritäten abstrahieren. Die nicht realisierten Aufgaben werden entweder im nächsten Meilenstein, oder in der geplanten Zeitreserve nachgeliefert. Falls ein Meilenstein gar nicht eingehalten werden kann, muss unmittelbar nach Erkenntnis Kontakt mit unserem Betreuer oder Auftraggeberin aufgenommen werden.

Die Meilensteine werden in Redmine als Versionen abgebildet und die groben Arbeitspakete als Tickets angegeben. Redmine bietet zusätzlich ein integriertes Zeitplan-Diagramm (Gantt-Diagramm). Dies liefert uns immer einen direkten IST-/SOLL-Vergleich zu unserer Zeitplanung. Die Arbeitspakete werden im Kapitel 6 *Arbeitspakete*6. detaillierter beschrieben.

4.3 Vorgehensmodell

Unsere Bachelorarbeit wird nach dem Vorgehensmodell Rational Unified Process (RUP) entwickelt. RUP beinhaltet verschiedene Disziplinen (wie z.B. Business Modeling, Requirements, Analyse & Design, Implementation, usw.), die in jeder Phase und in eine oder mehrere Iterationen vorkommen. Dadurch ist auch ein projektbegleitendes Qualitätsmanagement möglich. Zudem werden die Komponenten isoliert entwickelt als auch getestet, was zur Wiederverwendbarkeit und der Qualitätssteigerung des Produktes beiträgt. Zum besseren Problemverständnis nutzen wir für die visuelle Modellierung die Notationssprache UML.

4.3.1 Phasen / Iterationen

TeachEval wird in den Phasen Inception, Elaboration, Construction und Transition unterteilt. Jede Phase enthält eine oder mehrere Iterationen mit einer Grössenordnung von etwa 2 Wochen. Dazu beachten Sie unsere Zeitplanung in Redmine oder in unserem Meilenstein Dokument.

Die Bachelorarbeit beginnt mit der Inception Phase und endet dann nach dem Durchlauf aller RUP Phasen mit der Transition Phase. Die Iterationen sind ebenfalls in Redmine als Meilensteine definiert.

4.3.1.1 Inception

Die Inception Phase dient als Konzeptionsphase, in der die Vision, die Ziele und die Risiken des Projektes wie auch die wichtigsten Funktionalitäten beschrieben werden. Vom Projektbeginn begann die Inception Phase zu Beginn des Kickoff-Meetings am 18.02.2014 und endet zwei Woche später am 01.03.2013.

4.3.1.2 Elaboration

Anschliessend läuft die Elaboration Phase 3 Wochen, somit mit 2 Iterationen. In der ersten Iteration werden 80% der Anwendungsfälle mittels Artefakten beschrieben und der Projektplan wird endgültig beim Betreuer vorgelegt. In der zweiten Iteration werden mehrere Architekturprototypen erstellt, um mögliche Risiken bereits zu bearbeiten.

4.3.1.3 Construction

Nachdem die Architektur ausgearbeitet worden ist, beginnen wir in der Construction Phase mit der Entwicklung und dem Testen. Die Construction Phase enthält 4 Iterationen mit insgesamt 9 Wochen. Nach Ende jeder Iteration wird ein Architekturprototyp implementiert und ausgebaut.

4.3.1.4 Transition

In der Transition Phase bleiben 3 Wochen übrig. Hier können parallel letzte Anpassungen und Fehlerbehebungen durchgeführt werden, sowie für die Abgabe des A0-Posters und der Dokumentation vorbereitet werden.

4.3.2 Meilensteine

Das Projekt wurde in 9 Meilensteine definiert. In der folgenden Abbildung eine grobe Übersicht aller Meilensteine.

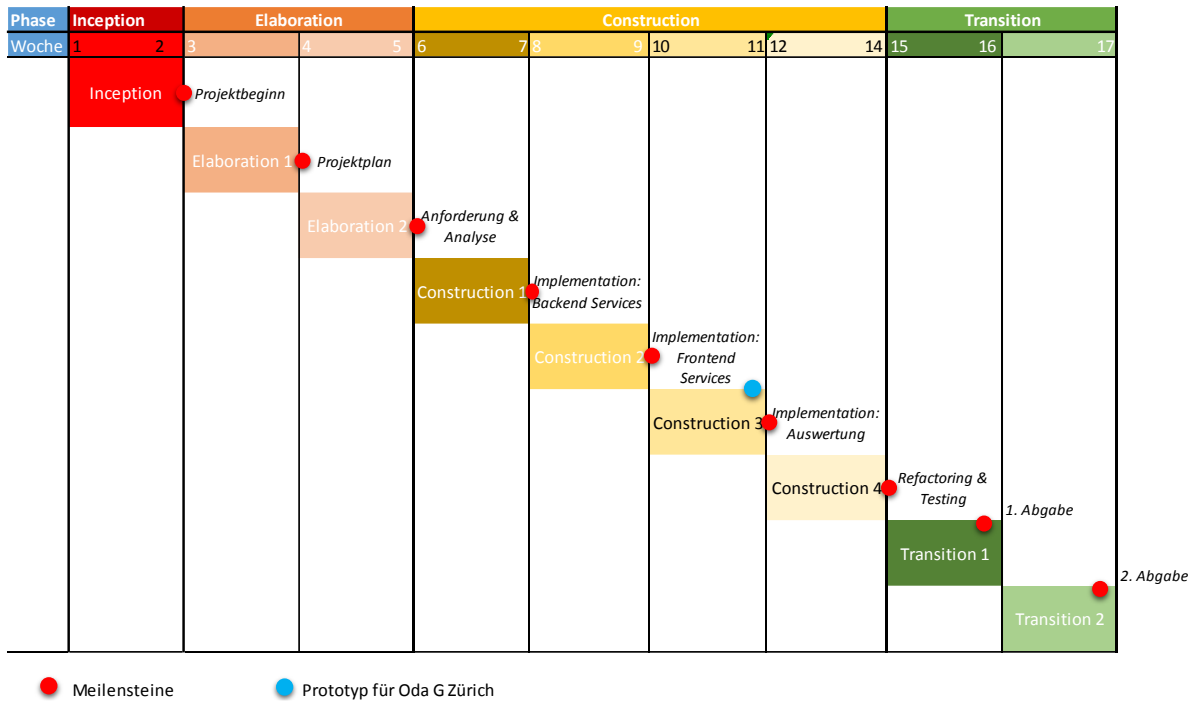


Abbildung 26 Übersicht der Phasen und Meilensteine

Beachten Sie unser Meilenstein-Dokument im Anhang oder Redmine für die detailliertere Planung.

4.3.2.1 M1 Projektbeginn

In den ersten zwei Wochen werden mit den Kick-Off Meetings die Vorabklärungen und die Aufgabenstellung mit unserem Betreuer Eduard Glatz und der Auftraggeberin der Oda G Brigitte Keller besprochen. Ausserdem wird eine erste Grobplanung erstellt.

4.3.2.2 M2 Projektplan

Die technischen Risiken werden im TechnischeRisiken-Dokument analysiert und abgearbeitet. Die Meilensteine werden definiert und eine Detailplanung in Redmine mit Arbeitspakete ausgearbeitet, die fortlaufend abgearbeitet und angepasst wird.

4.3.2.3 M3 Anforderung & Analyse

Die Anforderungsspezifikationen werden abgearbeitet und ein Datenbank-Modell wird erstellt. Ein Use Case-Diagramm zeigt die Requirements der Auftraggeberin für die Webapplikation auf. Zudem werden die nicht funktionalen Anforderungen beschrieben und ein Design bzw. Architektur ausgearbeitet. Kleine Prototypen stellen die Machbarkeit der Applikation in Bezug zu den technischen Risiken sicher. Mit den Anforderungsspezifikationen und den Wireframes wird ein Lösungsvorschlag ausgearbeitet.

4.3.2.4 M4 Implementation: Backend Service

Das ausgearbeitete Datenbank Modell wird in einer Datenbank aufgesetzt und realisiert. Mit Hilfe der Prototypen und der ausgearbeiteten Architektur wird serverseitig eine RESTful API implementiert und geschützt (durch User-Authentication). Ein weiteres Feature wird der Import/Export von Excel, Word Dokumente sein.

4.3.2.5 M5 Implementation: Frontend Umfrage

Im Frontend wird die Umfrage realisiert. Die Umfrage bezieht ihre Daten, abhängig der User-Tokens, von unserem Webserver per Ajax. In diesem Meilenstein werden folgende Bereiche abgedeckt:

- Frontend Architektur
- Erstellen von Umfragen (Import von Excel Datei)
- Bereitstellung und Teilnahme an Umfragen (Einsatz von anonyme, eindeutige Tokens)
- Verwaltung von Umfragen (Übersicht, Archivierung, Liste aller Tokens)

4.3.2.6 M6 Implementation: Frontend Auswertung

Als zweiter Teil der Applikation wird das Dashboard entwickelt. Dieser enthält die Daten und Diagramme zu den jeweiligen Umfragen. Hier wird der Export der Auswertung als Excel Datei realisiert. Zudem wird in diesem Meilenstein die Korrelation mit unterschiedlichen Fragen entwickelt. Am Ende von Meilenstein 6 wird ein Prototyp für die OdA G ZH aufgesetzt, damit sie diese für Testzwecken einsetzen kann.

4.3.2.7 M7 Refactoring & Testing

Ein eigener Meilenstein befasst sich mit Refactoring und Testing. Die bestehende Lösung wird überarbeitet und ausgearbeitet. Da sich die OdA G ZH mit dem M6 Meilenstein (Anfangs Mai) ein funktionstüchtiger Prototyp erhofft, dient dieser Meilenstein auch für die realisieren von fehlenden Features. Ausserdem wollen wir mit Refactoring und Testing eine bessere Usability und eine zuverlässige Applikation gewährleisten.

4.3.2.8 M8 Abgabe Kurzfassung & A0-Poster

Ein vordefinierter Meilenstein der HSR ist die Abgabe einer Kurzfassung und einem A0 Poster an unserem Betreuer. Diese Abgabe findet am 06.06.14 statt.

4.3.2.9 M9 Abgabe der Bachelorarbeit & Präsentation

Fertigstellung der Bachelorarbeit und letzte Anpassungen an Dokumenten bis am 13.06.14 bis 12.00 Uhr. Die Abgabe umfasst alle Dokumente auf Papier und CD und wird an unser Betreuer, unserer Auftraggeberin und an die HSR Studiengangsekretariat abgegeben.

Am selben Tag zwischen 16.00 und 20.00 Uhr erhalten wir mit der Präsentation der Bachelorarbeit die Gelegenheit unser Projekt vorzustellen und unser Betreuer bzw. Auftraggeberin von unserer Arbeit / Applikation zu überzeugen.

4.4 Besprechungen

Da beide Teammitglieder denselben Stundenplan bzw. die gleichen Module besuchen, werden die Besprechungen sehr oft in den Zwischenzeiten durchgeführt.

Die Besprechungen mit Eduard Glatz soll wöchentlich am Freitag zwischen 12:00-13:00 Uhr in der Cafeteria durchgeführt werden. Dabei sollen insbesondere mögliche Probleme, das weitere Vorgehen und Erledigtes besprochen werden. Im Laufe des Projekts müssen nicht mehr zwingend wöchentliche Meetings durchgeführt werden.

Meetings mit der Auftraggeberin Brigitte Keller werden laufend nach Vereinbarungen durchgeführt. Diese können entweder in der HSR oder bei der OdA G in Zürich stattfinden.

5. Risikomanagement

Das Risikomanagement wird in einem separaten Dokument (siehe TechnischeRisiken.xlsx) ausführlicher beschrieben und aufgelistet. Dabei wurden insbesondere die technischen Risiken und deren geschätzten Schaden für unsere Bachelorarbeit analysiert und erläutert.

5.1 Risiken

Die Risiken werden im TechnischeRisiken-Dokument aufgelistet, beschrieben und bewertet.

5.2 Umgang mit Risiken

Mit Risiken und Problemen in einem Projekt muss immer gerechnet werden. Aus diesem Grund ist es wichtig genügend Zeitreserven einzuplanen, um die kritischsten Risiken zu beseitigen und die Kernarchitektur zu bestimmen. Die Einarbeitung in diesen Themenbereichen ist schon sehr früh im vollen Gange.

Die Risiken müssen bis zum Abschluss des Projektes im Auge behalten werden. Falls Probleme auftauchen, dienen Sitzungen für die Besprechung und anschliessend deren Lösungssuche. Zusätzlich appellieren wir an unsere Eigeninitiative, dass bei Problemen sofort die anderen Teammitglieder informiert werden, damit wir als Team die Risiken und Probleme in Griff bekommen. Zudem müssen laufend neue Risiken protokolliert werden.

Zusätzlich werden in der Elaboration-Phase kleine Technologieprototypen entwickelt, um allfällige Risiken, die aufgrund unbekannter Technologien auftauchen könnten, zu minimieren.

6. Arbeitspakete

Alle Arbeitspakete werden in Redmine definiert und abgearbeitet. Hier werden die Arbeitspakete pro Meilenstein als Screenshot gezeigt.

M1 Projektbeginn

Zugehörige Tickets

Backend-Feature-#5: Pre-Study: Java Restful Jersey
Backend-Feature-#11: Pre-Study: Postgres DB
Backend-Feature-#17: Pre-Study: JDBC, JPA, Hibernate
Backend-Feature-#27: Pre-Study: Excel-CRUD with Java
Frontend-Feature-#4: Pre-Study: AngularJS
Meeting-#2: Sitzung: eglatz 18.02.14
Meeting-#3: Sitzung: bkeller 26.02.14
Meeting-#10: Sitzung: eglatz 28.02.14
Documentation-#6: Requirements: Fragen mit Fragebogen/Anwortbogen vorbereiten
Administrative-#1: Mail: Kontaktaufnahme Frau Keller von Oda G Zürich

Tabelle 6 Arbeitspakete Projektbeginn

M2 Projektplan

Zugehörige Tickets

Meeting-#26: Sitzung: eglatz 07.13.14
Documentation-#7: Projektplan: Einleitung & Projektübersicht
Documentation-#8: Projektplan: Projektorganisation & Management Abläufe
Documentation-#12: Projektplan: Risikomanagement & Technische Risiken
Documentation-#13: Projektplan: Infrastruktur
Documentation-#16: Projektplan: Qualitätsmassnahmen & Testen
Administrative-#14: Zeitplanung: Meilensteine und Arbeitspakete in Redmine erfassen
Administrative-#15: Meilensteine: Grobplanung erstellen

Tabelle 7 Arbeitspakete Projektplan

M3 Anforderung & Analyse

Zugehörige Tickets

Backend-Feature-#34: Rest: UserRest
Backend-Feature-#35: Rest: SurveyRest
Backend-Feature-#36: Rest: LoginRest
Backend-Feature-#58: Persistence: Entities erstellen
Meeting-#28: Sitzung: bkeller 12.03.14
Meeting-#29: Sitzung: eglatz 14.03.14
Meeting-#30: Sitzung: eglatz 21.03.14
Documentation-#9: Anforderungsspezifikation: Nicht-Funktionale Anforderungen
Documentation-#18: Anforderungsspezifikation: Use Case beschreiben
Documentation-#19: Anforderungsspezifikation: Use Case Diagramm
Documentation-#20: Wireframes
Documentation-#21: Datenbank Modell
Documentation-#22: Software Architektur Dokument: Einleitung & Systemübersicht
Documentation-#23: Software Architektur Dokument: Logische Architektur
Documentation-#24: Software Architektur Dokument: Projektstruktur & Funktionen
Documentation-#25: Software Architektur Dokument: REST & Datenspeicherung (DB)
Documentation-#32: Projektplan: Überarbeitung
Documentation-#33: Anforderungsspezifikation: Einführung (Kompatibilität, Benutzerfreundlichkeit, etc.)

Tabelle 8 Arbeitspakete Anforderung & Analyse

M4 Implementation: Backend Service

Zugehörige Tickets

Backend-Feature-#37: Persistence: UserQuery erstellen
Backend-Feature-#38: Persistence: SurveyQuery erstellen
Backend-Feature-#40: Authentication: SecurityFilter, TokenManager, Token
Backend-Feature-#41: Persistence: LoginQuery erstellen
Backend-Feature-#51: Import Excel Benutzer (Dozenten & Leiter)
Backend-Feature-#52: Import Excel Umfrage
Backend-Feature-#60: Junit Test Framework Entity Schicht
Backend-Feature-#61: Persistence: Session Handling
Backend-Feature-#105: Persistence: Lazy-Loading for Entities
Frontend-Feature-#42: AngularJS: Authentication
Frontend-Feature-#43: AngularJS: Frontend Layout
Frontend-Feature-#44: AngularJS: Seiten Struktur
Frontend-Feature-#45: AngularJS: Error Handling
Frontend-Feature-#48: AngularJS: Profile
Frontend-Feature-#49: AngularJS: UserManagement
Meeting-#31: Sitzung: eglatz 28.03.14
Meeting-#115: Sitzung: bkeller 26.03.14

Tabelle 9 Arbeitspakete Implementation Backend Service

M5 Implementation: Frontend Umfrage

Zugehörige Tickets

Backend-Feature-#63: Rest: downloadSurvey
Backend-Feature-#64: Umfrage exportieren
Backend-Feature-#65: Rest: Umfrage teilnehmen mittels Token
Backend-Feature-#66: Rest: Umfrage teilnehmen - Antworten parsen und speichern
Backend-Feature-#67: Hibernate: Umfrage teilnehmen - Antworten speichern & Benutzer deaktivieren
Backend-Feature-#70: Umfrage starten
Backend-Feature-#71: Umfrage stoppen
Backend-Feature-#72: Umfrage entfernen
Frontend-Feature-#46: AngularJS: Survey Management
Frontend-Feature-#47: AngularJS: Survey & Survey Participate
Frontend-Feature-#73: AngularJS: Suchfeld
Frontend-Feature-#106: Umfrage: Download als Excel-Datei
Meeting-#116: Sitzung: eglatz 11.04.14
Bug-#53: Persistence: Import SurveyQuestion
Documentation-#54: Architektur: ServerArchitektur Phase1
Documentation-#56: Architektur: ClientArchitektur Phase1
Administrative-#107: Server: Testserver einrichten

Tabelle 10 Arbeitspakete Implementation Frontend Umfrage

M6 Implementation: Frontend Auswertung

Zugehörige Tickets

Backend-Feature-#68: Junit Test Framework Query Schicht
Backend-Feature-#69: Junit Test Framework Service Schicht
Backend-Feature-#78: Quartz: Automatisches Starten und Beenden von Umfragen
Backend-Feature-#79: MemoryAnalysis: Refactoring, Hibernate LazyLoading, ApachePoi Refactoring, Tomcat configuration
Backend-Feature-#108: Hibernate: Export Auswertung Details
Backend-Feature-#109: Hibernate: Export Auswertung Überblick
Frontend-Feature-#74: AngularJS: Auswertung Überblick
Frontend-Feature-#75: AngularJS: Auswertung Detailliert
Frontend-Feature-#76: AngularJS: Auswertung Korrelation
Frontend-Feature-#77: AngularJS: Refactoring & Verbesserung der Umfragen
Meeting-#117: Sitzung: eglatz 25.04.14
Documentation-#55: Architektur: ServerArchitektur Phase2
Documentation-#57: Architektur: ClientArchitektur Phase2

Tabelle 11 Arbeitspakete Implementation Frontend Auswertung

M7 Refactoring & Testing

Zugehörige Tickets

Backend-Feature-#83: Jersey: Exception Handling Refactoring
Backend-Feature-#88: Jersey: Projectstructure Refactoring, better Dependencies
Backend-Feature-#89: Jersey: Swagger
Backend-Feature-#103: Java: UnitTest erweitern & Refactoring
Frontend-Feature-#80: AngularJS: Projectstructure Refactoring
Frontend-Feature-#90: AngularJS: Detail Auswertung Refactoring
Frontend-Feature-#111: AngularJS: Test, Refactoring, Support(OdA G), Bug fixes
Meeting-#84: Sitzung: bkeller 07.05.14
Meeting-#118: Sitzung: bkeller 07.05.14
Meeting-#119: Sitzung: eglatz 09.05.14
Meeting-#120: Sitzung: eglatz 16.05.14
Meeting-#121: Sitzung: eglatz 23.05.14
Documentation-#85: JavaDoc: Einsatz und Dokumentation mit JavaDoc
Documentation-#96: REST Spezifikation
Administrative-#101: Produktiver Server einrichten (SSL, Tomcat, PostgreSQL)

Tabelle 12 Arbeitspakete Refacotring & Testing

M8 Abgabe Kurzfassung/Abstract & A0-Poster

Zugehörige Tickets

Meeting-#122: Sitzung: eglatz 06.06.14
Documentation-#81: Abstract
Documentation-#82: A0 Poster
Documentation-#98: Glossar
Documentation-#99: Quellen
Documentation-#100: Installationsdokument: Überarbeitung und Verbessern
Documentation-#112: Benutzerhandbuch

Tabelle 13 Arbeitspakete Abgabe 1

M9 Abgabe Bachelorarbeit & Präsentation

Zugehörige Tickets

Documentation-#86: TechnischerBericht: Dokument schreiben und überarbeiten
Documentation-#91: Qualitätsmanagement: Usabilitytest
Documentation-#92: Qualitätsmanagement: Unittest
Documentation-#93: Qualitätsmanagement: Kompatibilitätstest
Documentation-#94: Titelblatt, Dokument Layout & Struktur
Documentation-#95: Wireframe: Überarbeitung & Verbesserungen
Documentation-#97: Management Summary
Documentation-#104: Fertigstellung Dokumentation
Documentation-#123: Überarbeitung Software Architektur
Documentation-#124: Überarbeitung Software Qualitätsmanagement
Documentation-#126: Überarbeitung Benutzerhandbuch
Administrative-#113: Plakat drucken
Administrative-#114: Dokumentation drucken
Administrative-#125: CD erstellen
Administrative-#127: Überarbeitung Benutzerhandbuch

Tabelle 14 Arbeitspakete Abgabe 2

7. Infrastruktur

Eine Anforderung unseres Projektes ist es, wenn möglich Open-Source Software (OSS) zu verwenden. Dabei haben wir folgende Tools im Einsatz:

Software / Systeme	Beschreibung / Einsatzbereich
OS Entwicklungsumgebung	Entwicklungsumgebung: Windows 7 / 8 & Ubuntu Linux
OS Webserver	Ubuntu Linux der HSR
Eclipse Kepler	Backend-Entwicklung: Java Web Service
Webstorm/Aptana Studio	Frontend-Entwicklung: HTML, CSS, JavaScript
Postgres	Backend: Datenbank für das Persistieren der Daten
Java Jersey Restful	Backend: RESTful Jersey
JDBC, JPA, Hibernate	Backend: Java Datenbankbindung
AngularJS	Frontend: Browserbasiertes Single-Page-Anwendung mit MVC-Modell
HTML, CSS	Frontend: Die Seitenarchitektur wird mit HTML und das Design/Layout mit CSS realisiert.
Twitter Bootstrap	Frontend: Framework für Responsive Web Design mit CSS und HTML. (Mobile freundlich)
Java, Javascript	Java als serverseitige und Javascript als clientseitige Programmiersprache
Git	Versionsverwaltung von Dateien und Quellcode. Link zum Remote-Repository: https://git.hsr.ch/git/TeachEval
Redmine	Realbasiertes Projektmanagementtool. Für das Projektmanagement, das Wiki und den Stundenrapport. Link: http://sinv-56043.edu.hsr.ch/redmine
MS Office	Für die Dokumentation
Dropbox	Austausch von Dokumenten
Astah Community	UML-Modellierung für die Requirements und Software Architektur
DBSchema	Abbildung des Datenbankschemas

Tabelle 15 eingesetzte Tools

8. Qualitätsmassnahmen

Nach jedem Abschluss eines Arbeitspakets, muss das andere Teammitglied die Arbeit überprüfen. Durch die ständige Zusammenarbeit an der HSR werden abgeschlossene Arbeiten angeschaut und besprochen. Durch die wöchentlichen Meetings mit unserem Betreuer erhalten wir Feedback zu der geleisteten Arbeit.

8.1 Dokumentation

Die Dokumentation wird zwischen den Teammitgliedern auf Dropbox geteilt. Zusätzlich befindet sich auf jedem privaten Rechner eine Kopie als Backup.

8.2 Projektmanagement

Als Projektmanagement Tool setzen wir Redmine ein. Dabei hat besitzt jedes Teammitglied ein Login.

8.3 Entwicklung

Der Source Code befindet sich auf dem GIT Remote-Repository der HSR <https://git.hsr.ch/git/TeachEval> und lokal auf unseren Workspaces.

8.3.1 Vorgehen

Sobald das Projekt um eine weitere Funktionalität erweitert wird, muss sinnvoll getestet werden, ob die Erweiterung auch den Erwartungen entspricht. Dies führen wir serverseitig wie auch clientseitig.

8.3.2 Unit Testing

Für den Server Teil kommt Java zum Einsatz. Für Java gibt es ein Unit Test Framework namens JUnit. Clientseitig bietet AngularJS das Unit Test Framework namens Jasmine an.

8.3.3 Code Reviews

Mindestens einmal pro Iteration findet ein Codereview durch das andere Teammitglied statt. Damit behalten beide den Überblick über das Projekt und der Code wurde von mindestens einer anderen Instanz überprüft.

8.3.4 Code Style Guidelines

Das Backend wird mit Eclipse entwickelt, während das Frontend mit Webstorm und Aptana Studio entwickelt wird. Wir verwenden jeweils den Standard Code Style Guideline der Entwicklungsumgebung.

8.4 Testen

8.4.1 Integrationstest

Unser Web Service hat mehrere Schnittstellen von denen die Daten mit Hilfe der HTTP Befehlen wie GET und POST abgerufen werden. Diese müssen laufend bei jeder Iteration getestet werden.

8.4.2 Systemtest

Die Webapplikation wird in der Construction Phase nach jeder Iteration einem Systemtest unterlaufen, damit sichergestellt wird, ob alle Requirements der Auftraggeberin implementiert sind und funktionieren.

8.4.3 Usability

Mit Usability Tests werden Meinungen eingeholt, um die Webapplikation zu verbessern. Dabei appellieren wir vor allem auf die Kooperation der OdA G ZH. Anhand der Usability Tests kann die Effektivität, Effizienz, Erwartungskonform, Fehlerrobustheit oder Steuerbarkeit der App gemessen werden.

9. Abbildungsverzeichnis

Abbildung 24 OdA G Vision & Leitbild (Vision der OdA G ZH, 2014)	49
Abbildung 25 Organisationsstruktur	51
Abbildung 26 Übersicht der Phasen und Meilensteine.....	55

10. Tabellenverzeichnis

Tabelle 4 Teammitglied Benjamin Kehl	51
Tabelle 5 Teammitglied Dominique Sorg	52
Tabelle 6 Arbeitspakete Projektbeginn	58
Tabelle 7 Arbeitspakete Projektplan	58
Tabelle 8 Arbeitspakete Anforderung & Analyse	58
Tabelle 9 Arbeitspakete Implementation Backend Service	58
Tabelle 10 Arbeitspakete Implementation Frontend Umfrage.....	59
Tabelle 11 Arbeitspakete Implementation Frontend Auswertung	59
Tabelle 12 Arbeitspakete Refacotring & Testing	59
Tabelle 13 Arbeitspakete Abgabe 1.....	59
Tabelle 14 Arbeitspakete Abgabe 2.....	59
Tabelle 15 eingesetzte Tools	60

1. Einführung

1.1 Zweck

Dieses Dokument beschreibt die Anforderungen für die Webapplikation TeachEval.

1.2 Beschreibung

Die *Organisation der Arbeitswelt Gesundheit (Oda G ZH)* bietet diverse Kurse im Bereich Gesundheit an. Für diese Kurse werden Evaluationen durchgeführt um die Qualität des Unterrichts zu messen. Da im jetzigen Standpunkt alle Evaluationen von Hand durchgeführt werden, benötigt dies ein grosser Mehraufwand. Aus diesem Grund soll eine Webapplikation für die Oda G entwickelt werden, die die Befragungen als auch Evaluationen automatisiert.

1.3 Gültigkeitsbereich

Die Anforderungsspezifikation gilt als Grundlage des Projektes und ist daher über die gesamte Projektdauer gültig. Damit das Dokument immer auf dem neusten Stand ist, wird es laufend angepasst.

1.4 Übersicht

Dieses Dokument gibt eine Übersicht über die Anforderungen für unsere Bachelorarbeit TeachEval. Im ersten Kapitel werden die Anforderungen allgemein beschrieben, damit die Aufgabenstellung der Arbeit besser ersichtlich ist. Danach werden die Anforderungen als Use Cases erfasst und als grobe Übersicht in einem Use Case Diagramm aufgezeigt. Die wichtigsten Use Cases werden zusätzlich als UC Fully Dressed detaillierter beschrieben. Zum Schluss werden weitere Anforderungen in Form von nichtfunktionalen Anforderungen aufgezählt und beschrieben.

2. Allgemeine Beschreibung

2.1 Produkt Perspektive

TeachEval ist eine öffentlich zugängliche Webapplikation, für die primär Leiter/innen, Dozenten/innen, Supervisoren/innen oder Studenten/innen Zugang erhalten. TeachEval automatisiert den bislang manuell ausgeführten Umfrageprozess für die Qualitätssicherung der Kurse bei der OdA G Zürich.

2.2 Produkt Funktion

Der/die Leiter/in der OdA G Zürich erfasst mit Hilfe eines Excel-Sheet die Umfragen in der Webapplikation. Mit der Umfrageverwaltung ist er/sie in der Lage, die erfassten Umfragen zu verwalten und die Tokens bzw. Teilnehmerkennungen der Teilnehmer als Excel zu exportieren. Dozierende, Supervisoren und Studenten können an unterschiedlichen Umfragen teilnehmen, sofern sie dazu berechtigt sind. Die Fragen können unter den unterschiedlichen Benutzerrollen abweichen. Der/die Leiter/in wie auch der/die Dozent/in ist zudem in der Lage, die Auswertungen zu ihren Kursen anzusehen und als Excel zu exportieren.

2.3 Technologien

Die einzusetzende Technologie wird uns Projektleitern / Entwicklern überlassen. Die einzige Bedingung ist, dass sie mit Open Source Software entwickelt werden muss (siehe genaue Technologiewahl im Projektplan).

2.4 Benutzer Charakteristik

Für die TeachEval Webapplikation werden vier Benutzerrollen unterschieden:

Benutzer	Beschreibung
Leiter/-in	Der Leiter bzw. die Leiterin ist zugleich der Administrator der Webapplikation. Er/sie hat Einsicht auf alle Umfragen und Auswertungen. Zudem kann er/sie neue Umfragen erstellen. Die Fragen können unabhängig von den Umfragen beliebig angepasst und erweitert werden. Wenn jedoch eine Umfrage abgeschlossen wurde, sind Bearbeitungen an den Fragen und Teilnehmern nicht mehr möglich. Für die Auswertung können Reports (Excel oder PDF) generiert und mit unterschiedlichen Filterungen miteinander verglichen und eingeschränkt werden.
Dozent/-in	Der Dozent füllt nach seinem Unterricht, wie die Studenten die Umfrage am Ende des Kurstages aus. Die Fragen sind spezifisch auf den Dozent gerichtet. Wenn der Dozent seine Befragung abgeschlossen hat, kann er die Umfrage abschliessen und die Ergebnisse der Studenten einsehen. Eine abgeschlossene Umfrage kann aber nicht wiedereröffnet werden.
Supervisor/-in	Der Supervisor füllt wie der Dozent und die Studierenden eine Umfrage aus. Die Fragen können anders gestellt sein als beim Dozent oder Student. Im Gegensatz zu den Studenten und dem Dozenten erhält der Supervisor bis zu 2 Wochen Zeit, um die Fragen auszufüllen, jedoch erhält er keinen Einblick in die Auswertungen.
Student/-in	Der Student bzw. Teilnehmer füllt die Umfrage am Ende des Kurses aus. Er erhält keinen Einblick in die Auswertungen.

Tabelle 16 Benutzer Charakteristik

Die Personen in den genannten Benutzerrollen stehen im Zentrum der Konzeption, da sie diejenigen sind, die letztendlich mit der Applikation arbeiten. Die Webapplikation muss so aufgebaut sein, so dass sie benutzerfreundlich (siehe Kapitel Benutzerfreundlichkeit) ist und dass die gewünschten Informationen so schnell wie möglich abgerufen werden können. Zudem soll die Befragung anonym und über das Internet durchführbar sein.

2.5 Benutzerfreundlichkeit

Unter Benutzerfreundlichkeit wird die vom Nutzer erlebte Nutzungsqualität bei der Interaktion mit der TeachEval Webapplikation verstanden. Eine einfache, zum Nutzer und seinen Aufgaben passende Bedienung wird als benutzerfreundlich angesehen. Faktoren, die die Benutzerfreundlichkeit beeinflussen können, sind die Wahrnehmung (bekannte Strukturen zeigen), die Aufmerksamkeit (Element hervorheben, Grosse/Kleine Elemente aufzeigen, Farbe, Kontrast), der Lesestil (unterschiedlichen Leseanforderungen gerecht werden), die Leserlichkeit (Schrift, Hintergrund, Kontrast, etc.) und die Navigation auf der Webseite (Benutzerfreundlichkeit Wikipedia, 2014). Wobei vieles dieser Elemente entnehmen wir aus dem CI-Styleguide der OdA G ZH.

Damit wir der OdA G Zürich eine benutzerfreundliche Applikation liefern können, definieren wir die Benutzerfreundlichkeit mit folgenden messbaren Kriterien.

- Die Benutzeroberfläche wird mit dem ISO 9241-110 Standard gemessen. Die Kriterien werden mit Fragebögen, Meetings und Usability auf eine Anzahl von der OdA G Zürich bereitgestellten Testpersonen durchgeführt. Die Mehrheit entscheidet über den tatsächlichen Messwert. Ein Messwert **über 60%** wird als benutzerfreundlich angesehen.

Kriterium	Beschreibung
Verfügbarkeit	Ein Dialog ist verfügbar, wenn er dem Benutzer bereitsteht, sobald er diesen braucht. Eine Internetverbindung wird vorausgesetzt.
Aufgabenangemessenheit	Ein Dialog ist aufgabenangemessen, wenn er den Benutzer unterstützt, seine Arbeitsaufgabe effektiv und effizient zu erledigen.
Übersichtlichkeit	Ein Dialog ist übersichtlich, wenn der Benutzer seine Aufgabe effektiv und effizient erledigen kann.
Selbstbeschreibungsfähigkeit	Ein Dialog ist selbstbeschreibungsfähig, wenn jeder einzelne Dialogschritt durch Rückmeldungen des Dialogsystems unmittelbar verständlich ist oder dem Benutzer eine Anfrage erklärt wird.
Erwartungskonformität	Ein Dialog ist erwartungskonform, wenn er konsistent ist und den Merkmalen des Benutzers entspricht, z.B. seinen Kenntnissen aus dem Arbeitsgebiet, seiner Ausbildung und seiner Erfahrung sowie den allgemein anerkannten Konventionen.
Fehlertoleranz	Ein Dialog ist fehlertolerant, wenn das beabsichtigte Arbeitsergebnis trotz erkennbar fehlerhafter Eingaben entweder mit keinem oder minimalem Korrekturaufwand seitens des Benutzers erreicht werden kann.
Erlernbarkeit	Ein Dialog ist lernförderlich, wenn er den Benutzer beim Erlernen des Dialogsystems unterstützt und anleitet.
Individualisierbarkeit	Ein Dialog ist individualisierbar, wenn das Dialogsystem Anpassungen an die Erfordernisse der Arbeitsaufgabe

	sowie an die individuellen Fähigkeiten und Vorlieben des Benutzers zulässt.
Steuerbarkeit	Ein Dialog ist steuerbar, wenn der Benutzer in der Lage ist, den Dialogablauf zu starten sowie seine Richtung und Geschwindigkeit zu beeinflussen, bis das Ziel erreicht ist.

Tabelle 17 ISO 9241-110

- Neben der Benutzeroberfläche wird auch die Effizienz gemessen. Für die Messung werden die Teilnehmer an den geplanten Usability Tests befragt.

2.6 Kompatibilität

Da die Zielgruppe der Webapplikation mit den unterschiedlichsten Smartphones und Computer auf TeachEval zugreifen können, muss eine breite Palette von Browser unterstützt werden. Für TeachEval werden fortlaufend die folgenden Browser während der Entwicklung getestet:

PC / Notebook	
Windows 7/8	<ul style="list-style-type: none"> - Internet Explorer 10 - Internet Explorer 8, 9 → Fokus auf Umfrageteilnahme - Chrome >=25 - Firefox >= 23 - Safari >= 5.1 - Opera >= 12.14
Mac OS	<ul style="list-style-type: none"> - Safari >= 5.1 - Chrome >= 25 - Opera >= 12.14 - Firefox >= 23

Smartphone	
iPhone IOS >= 4	Safari, Chrome
Android Phone	Chrome
Windows Phone	Internet Explorer

Tablet	
Tablet Android	Chrome
Tablet Windows	Internet Explorer
Tablet Ipad	Safari, Chrome

2.7 Einschränkungen

In der ersten Version von TeachEval können Umfragen nur über ein Excel Template mit den Formaten *.xls oder *.xlsx erstellt werden.

2.8 Abhängigkeiten

- Der Server muss die serverseitigen Technologien (Java, Postgres) unterstützen.
- Die erste Version von TeachEval erlaubt es, eine Umfrage nur über ein Excel-Sheet zu erstellen. Der Leiter/ -in ist damit an Excel gebunden.

3. Use Cases

In diesem Projekt müssen folgende funktionale Anforderungen umgesetzt werden:

ID	Use Case Titel	Priorität
01	An- / Abmelden	1
02	Umfrage CRUD	1
03	Token generieren	1
04	Token exportieren	1
05	Token per Mail versenden	2
06	Excel Template importieren	1
07	Auswertungen ansehen	1
08	Korrelation von Fragen	1
09	Export als Excel oder PDF	1
10	Umfrage teilnehmen	1
11	Benutzer verwalten	1

3.1 Priorisierung

Wie in der Tabelle oben ersichtlich ist, wurden die Use Cases in Priorität 1 und 2 festgelegt. Priorität 1 steht für wichtig und Priorität 2 für normal. Use Cases mit Priorität 1 müssen entwickelt werden und solche mit Priorität 2 können bei guter Entwicklung zusätzlich umgesetzt werden.

3.2 Use Case Diagramm

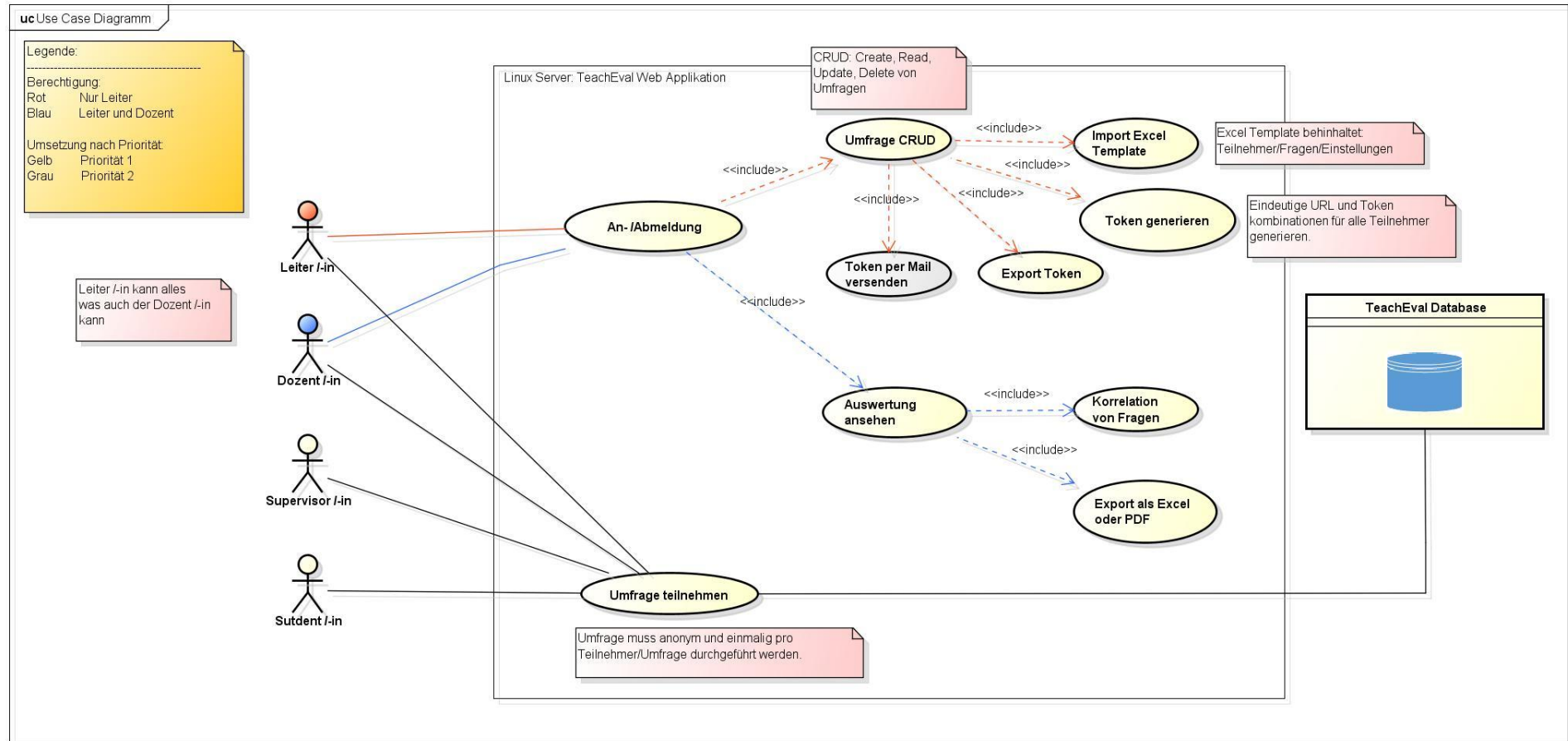


Abbildung 27 Use Case Diagramm für Oda G ZH

powered by Astah

3.3 Aktoren

Die Aktoren entsprechen den Personen, die im Kapitel 2.4 Benutzer Charakteristik erwähnt wurden. Im Verlaufe des Dokuments werden die Begriffe Benutzer und Teilnehmer verwendet. Unter dem Benutzer werden allgemein alle Personen (Leiter und Dozent) bezeichnet, die Zugriff auf die geschützten Bereiche der Webapplikation haben und die Teilnehmer sind alle Personen (Leiter, Dozent, Supervisor und Student), die für eine Umfrage berechtigt sind.

3.4 Beschreibungen (Brief)

UC 01 An- / Abmelden	
Nicht angemeldete Benutzer dürfen keine Anpassungen der Inhalte oder der Umfragen auf der Webapplikation machen. Mit einem Login schützen wir die kritischen Operationen wie auch die Auswertung. Für Dozenten gilt zusätzlich, dass Umfrageauswertungen erst angezeigt werden, wenn erstens die Umfrage der Studenten abgeschlossen wurde und zweitens der Dozent selbst seine Umfrage ausgefüllt hat. Die Teilnahme an der Umfrage benötigt keine Authentisierung.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 18 UC Brief An-/Abmelden

UC 02 Umfrage CRUD	
Der angemeldete Leiter hat das Recht die Umfrage zu verwalten. Das heisst, er kann neue Umfragen erstellen, noch nicht veröffentlichte Umfragen bearbeiten oder löschen oder bereits abgeschlossene Umfragen löschen.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 19 UC Brief Umfrage CRUD

UC 03 Token generieren	
Für jeden Teilnehmer muss eine eindeutige URL (oder Token) erzeugt werden. Dieser Token gewährt dem Teilnehmer zu der Umfrage eine Session mit der er die Umfrage abschliessen kann. Der Teilnehmer kann jedoch nicht zweimal an der Umfrage teilnehmen.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 20 UC Brief Token generieren

UC 04 Token exportieren	
Die zufällig generierten Tokens müssen als Liste (in einer Excel, Word oder PDF) exportiert werden können.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 21 UC Brief Token exportieren

UC 05 Token per Mail versenden	
Neu erstellte Tokens werden automatisch an die Teilnehmer per Mail versendet.	
Priorität	2
Grund	Da die Server Infrastruktur der HSR keinen automatisierten Emailversand aus Sicherheitsgründen zulässt, kann diese Funktion bis zur Aufschaltung der Webapplikation nicht verwendet werden. Aus diesem Grund, entschieden wir uns, diesen Use Case später zu realisieren falls dafür Zeit vorhanden ist.

Tabelle 22 UC Brief Token per Mail versenden

UC 06 Excel Template importieren	
Für die Erstellung einer Umfrage werden die Fragen, Titel, Teilnehmer und weitere Kursdaten in einem Excel Template abgelegt. Der Leiter kann diese Informationen für seine neue Umfrage ausfüllen und bei der Prozedur zur Erstellung einer neuen Umfrage importieren bzw. hochladen.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 23 UC Brief Excel Template importieren

UC 07 Auswertungen ansehen	
Auswertungen von erstellten Umfragen dürfen für die Benutzer (Leiter und Dozenten) eingesehen werden. Dabei darf nur der Leiter alle Umfragen, ob abgeschlossen oder nicht, einsehen. Während der Dozent nur die abgeschlossene Umfrage für seinen Kurs einsehen darf.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 24 UC Brief Auswertungen ansehen

UC 08 Korrelation von Fragen	
Um einen möglichen Zusammenhang zwischen zwei Fragen zu finden, soll für angemeldete Benutzer eine Funktion für die Korrelation bereitstehen.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 25 UC Brief Korrelation von Fragen

UC 09 Export als Excel oder PDF	
Die Auswertungen müssen mit ihren Diagrammen als Excel oder PDF exportierbar sein.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 26 UC Brief Export als Excel oder PDF

UC 10 Umfrage teilnehmen	
Die Teilnehmer (Studenten, Dozenten, Supervisoren und Leiter) können anhand Ihres Tokens an der Umfrage teilnehmen und ausfüllen. Wobei der Student andere Fragen gestellt bekommt als ein Dozent oder Supervisor.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 27 UC Brief Umfrage teilnehmen

UC 10 Benutzer verwalten	
Der Leiter kann die Benutzer von TeachEval (Dozenten und Leiter) verwalten, die ebenfalls erweiterten Zugriff auf der Seite benötigen. Unter erweitertem Zugriff ist gemeint, dass Benutzer mit der Funktion als Leiter erfasst werden können, die ebenfalls Zugriff auf alle Ressourcen bekommen dürfen, oder Dozenten, die nur auf Ihre zugewiesenen Umfragen Zugriff auf deren Auswertungen haben.	
Priorität	1
Grund	Gehört zur Aufgabenstellung

Tabelle 28 UC Brief Benutzer verwalten

3.5 Beschreibungen (Fully Dressed)

UC 01	An- / Abmelden	
Umfang:	TeachEval Applikation	
Ebene:	Security, Anwendung	
Primärer Akteur:	Benutzer (Leiter und Dozent)	
Vorbedingungen:	Anmelden: <ul style="list-style-type: none"> - Der Benutzer hat bereits ein Account - Der Dozent wurde als Teilnehmer definiert und ein Token wurde für ihn generiert Abmelden: <ul style="list-style-type: none"> - Der Benutzer sind angemeldet bzw. haben eine aktive Session 	
Nachbedingung:	Anmelden: <ul style="list-style-type: none"> - Eine neue Session wird erstellt - Die Inhalte werden anhand der Benutzerrolle sichtbar / unsichtbar Abmelden: <ul style="list-style-type: none"> - Die Session wird gelöscht - Der Benutzer wird weitergeleitet 	
Hauptszenario: (User- /Systemverantwortlichkeiten)	Akteure, Aktionen: Anmelden mit Email & Passwort 1. Der Benutzer meldet sich mit Emailadresse und Passwort an Anmelden mit Token 1. Der Benutzer benutzt für die Anmeldung ein Token 3a. Benutzer gibt sein Passwort und bestätigt es in einem zweiten Feld 6. Bei Erfolg wird er direkt zu der Umfrage weitergeleitet. Falls Dozent und noch kein Passwort definiert, so muss er zuerst ein Passwort erstellen.	Systemverantwortlichkeiten: Anmelden mit Email & Passwort 2. Die angegebene Emailadresse wird anhand des Formats überprüft 3. Die Emailadresse wird in der Datenbank gesucht 4. Das Passwort wird verschlüsselt und mit dem verschlüsseltem Passwort in der DB verglichen 5. Session wird erstellt und der Benutzer wird zur Startseite weitergeleitet. Der Dozent sieht nur seine berechtigten Umfragen. Der Leiter sieht alle. Anmelden mit Token: 2. Das Token wird in der Datenbank gesucht 3. Bei Übereinstimmung wird der Benutzer in der DB überprüft, ob das Passwort –Feld nicht null ist. Falls null, siehe Punkt 3a. Sonst gehe zu Punkt 6. 4. Passworteingabe wird verschlüsselt und verglichen 5. Session wird erstellt und der Benutzer wird zu der Umfrage weitergeleitet

	<p>Abmelden</p> <p>1. Der Benutzer meldet sich ab</p>	<p>Abmelden</p> <p>3. Der Benutzer wird anhand der Emailadresse/Token identifiziert</p> <p>4. Session wird zerstört</p>
	<p>Anmelden mit Email & Passwort</p> <p>2a. Das Emailformat stimmt nicht (@ vergessen, Domain falsch oder Punkt vergessen) → Eine Meldung wird angezeigt, dass das Format falsch sei</p> <p>3a. Die Emailadresse konnte in der Datenbank nicht gefunden werden → Eine Meldung wird angezeigt, dass die Emailadresse nicht gefunden wurde</p> <p>5a. Das Passwort stimmt nicht mit dem Passwort in der Datenbank überein → Eine Meldung wird angezeigt, dass das Passwort nicht korrekt ist</p> <p>Anmelden mit Token</p> <p>2a. Das Token existiert in der DB nicht → Eine Meldung wird angezeigt, dass das Token nicht gefunden wurde</p> <p>2b. Das Token ist ungültig → Eine Meldung wird angezeigt, dass das Token nicht mehr gültig ist (Umfrage abgelaufen)</p> <p>4a. Passwörter stimmen nicht überein → Eine Meldung wird angezeigt, dass die Passworteingabe nicht übereinstimmt</p> <p>Abmelden</p> <p>2a. Der Benutzer konnte nicht identifiziert werden → Die Session des Benutzers ist abgelaufen ist somit bereits abgemeldet. → Er wird zur Startseite weitergeleitet</p>	
Häufigkeit des Auftretens:	An- / Abmeldungsprozedur	

Tabelle 29 UC Fully Dressed An-/Abmelden

UC 02	Umfrage CRUD	
Umfang:	TeachEval Applikation	
Ebene:	Anwendung	
Primärer Akteur:	Leiter	
Vorbedingungen:	<ul style="list-style-type: none"> - Der Leiter ist angemeldet - Für die Bearbeitung oder für das Löschen einer Umfrage darf sie entweder noch nicht veröffentlicht worden sein oder sie muss abgeschlossen sein. 	
Nachbedingung:	<ul style="list-style-type: none"> - Die zu bearbeitende Umfrage wurde in der Datenbank aktualisiert - Die Änderungen müssen übernommen werden 	
Hauptscenario: (User-/Systemverantwortlichkeiten)	<p>Aktoren, Aktionen:</p> <p>Umfrage hinzufügen</p>	<p>Systemverantwortlichkeiten:</p> <p>Umfrage hinzufügen</p> <p>4. Die Angaben in der Excel-Datei wie die</p>

	<ol style="list-style-type: none"> 1. Der Leiter navigiert zu der Umfrageseite und klickt auf „Umfrage erstellen“ 2. Ein neues Fenster für den Upload Prozess erscheint. 3. Für den Upload wählt der Leiter das Excel-Template und klickt dann auf „Umfrage hochladen“ <p>Umfrage bearbeiten/löschen</p> <ol style="list-style-type: none"> 1. Der Leiter navigiert zu der Umfrageseite und sieht eine Auflistung aller Umfragen mit Status 2. Leiter wählt eine Umfrage aus, die noch nicht veröffentlicht wurde oder bereits abgeschlossen ist und wählt bearbeiten oder löschen 	<p>Teilnehmerliste, Fragen, Umfragetitel, usw. werden herausgelesen und in die Datenbank gespeichert.</p> <ol style="list-style-type: none"> 5. Wenn erfolgreich, wird die neue Umfrage in der Umfrageliste neu angezeigt <p>Umfrage bearbeiten/löschen</p> <ol style="list-style-type: none"> 3. Die geänderten Daten werden in der Datenbank gespeichert und falls löschen in der DB als inaktiv markiert
	<p>3/4a. Es konnte nicht hinzugefügt, bearbeitet oder gelöscht werden, da die benötigten Angaben nicht korrekt oder nicht vorhanden sind. → Eine Fehlermeldung wird dem Benutzer aufmerksam machen → Benutzer muss die Angaben im Excel-File oder auf TeachEval überprüfen</p> <p>3b. Es konnte nicht bearbeitet oder gelöscht werden, da das ICS gar nicht existiert → Die Seite muss neu geladen werden</p>	
Häufigkeit des Auftretens:	Nach jedem Hinzufügen, Bearbeiten oder Löschen von Umfragen	

Tabelle 30 UC Fully Dressed Umfrage CRUD

UC 06		Excel Template importieren
Umfang:	TeachEval Applikation	
Ebene:	Anwendung, Datenbank	
Primärer Akteur:	Leiter	
Vorbedingungen:	<ul style="list-style-type: none"> - Der Leiter ist angemeldet - Es werden nur Excel-Files unterstützt (*.xls oder *.xlsx) - Ein vorgegebenes Excel-Template wurde verwendet 	
Nachbedingung:	<ul style="list-style-type: none"> - Die Daten aus dem Excel-Template werden herausgelesen und in die Datenbank gespeichert 	
Hauptscenario: (User- /Systemverantwortlichkeiten)	<p>Aktoren, Aktionen:</p> <ol style="list-style-type: none"> 1. Der Leiter wählt die Datei für den Importvorgang aus und bestätigt diese 	<p>Systemverantwortlichkeiten:</p> <ol style="list-style-type: none"> 2. Die Excel-Datei wird auf dem Server hochgeladen 3. Der Inhalt der Datei wird herausgelesen und als Entitäten dargestellt 4. Die Entitäten werden in die Datenbank gespeichert

		<p>5. Tokens werden generiert (UC 03) und in die hochgeladene Excel-Datei gespeichert</p> <p>6. Excel mit den Tokens wird als Download angeboten</p>
	<p>2a. Die Datei konnte nicht hochgeladen werden, da das Format nicht unterstützt wird → Eine Meldung wird angezeigt, dass nur Excel Formate mit der Endung .xls und .xlsx unterstützt werden</p> <p>3a. Es konnte nicht der ganze Inhalt herausgelesen werden → eine Meldung wird angezeigt, dass Informationen nur im markierten Bereich gespeichert werden</p>	
Häufigkeit des Auftretens:	Bei jedem Upload	

Tabelle 31 UC Fully Dressed Excel Template importieren

UC 07		Auswertungen ansehen	
Umfang:	TeachEval Applikation		
Ebene:	Anwendung		
Primärer Akteur:	Benutzer (Leiter und Dozent)		
Vorbedingungen:	<ul style="list-style-type: none"> - Der Benutzer muss angemeldet sein - Dozent muss für die Auswertung zu der Umfrage berechtigt sein - Umfrage muss abgeschlossen sein 		
Nachbedingung:	-		
Hauptszenario: (User-/Systemverantwortlichkeiten)	<p>Aktoren, Aktionen:</p> <ol style="list-style-type: none"> 1. Der Benutzer navigiert zu der Auswertung 	<p>Systemverantwortlichkeiten:</p> <ol style="list-style-type: none"> 2. Die Daten werden aus der Datenbank herausgelesen 3. Die Daten werden mittels Javascript als Grafiken dargestellt 	
	<p>3a. Die Charts werden falsch dargestellt → Die Datenbankabfragen werden falsch durchgeführt</p>		
Häufigkeit des Auftretens:	Bei jedem Zugriff zur Auswertung		

Tabelle 32 UC Fully Dressed Auswertung ansehen

UC 08		Korrelation von Fragen	
Umfang:	TeachEval Applikation		
Ebene:	Anwendung		
Primärer Akteur:	Benutzer (Leiter und Dozent)		
Vorbedingungen:	<ul style="list-style-type: none"> - Der Benutzer muss angemeldet sein - Dozent muss für die Auswertung zu der Umfrage berechtigt sein - Umfrage muss abgeschlossen sein 		
Nachbedingung:	-		
Hauptszenario: (User-/Systemverantwortlichkeiten)	<p>Aktoren, Aktionen:</p> <ol style="list-style-type: none"> 1. Der Benutzer navigiert zu der Auswertung 2. Er klickt auf die Option Korrelieren 3. Benutzer wählt zwei Fragen aus 	<p>Systemverantwortlichkeiten:</p> <ol style="list-style-type: none"> 4. Die Daten werden aus der Datenbank herausgelesen 5. Die Kovarianz wird mittels Java berechnet 	

	5a. Die Kovarianz wird falsch berechnet → Die Datenbankabfragen werden falsch durchgeführt
Häufigkeit des Auftretens:	Bei jeder Korrelation

Tabelle 33 UC Fully Dressed Korrelation von Fragen

UC 09	Export als Excel oder PDF	
Umfang:	TeachEval Applikation	
Ebene:	Anwendung	
Primärer Akteur:	Benutzer (Leiter und Dozent)	
Vorbedingungen:	<ul style="list-style-type: none"> - Der Benutzer muss angemeldet sein - Dozent muss für die Auswertung zu der Umfrage berechtigt sein - Umfrage muss abgeschlossen sein 	
Nachbedingung:	<ul style="list-style-type: none"> - Auswertungen werden als Datei im Format Excel oder PDF exportiert 	
Hauptszenario: (User- /Systemverantwortlichkeiten)	Akteure, Aktionen: 1. Der Leiter oder Dozent navigiert zu der Auswertung 2. Für das Exportieren zu dem gewünschten Format klicken sie auf das richtige Icon (z.B. Excel-Icon)	Systemverantwortlichkeiten: 3. Die Daten werden aus der Datenbank herausgelesen 4. Mittels Excel-Library wird ein neues Excel-File mit den Grafiken erzeugt und zum Download angeboten
	4a. Die Grafiken werden nicht richtig erzeugt	
Häufigkeit des Auftretens:	Bei jedem Export	

Tabelle 34 UC Fully Dressed Export als Excel oder PDF

UC 10	Umfrage teilnehmen	
Umfang:	TeachEval Applikation	
Ebene:	Anwendung	
Primärer Akteur:	Teilnehmer → Leiter, Dozenten, Supervisoren oder Studenten	
Vorbedingungen:	<ul style="list-style-type: none"> - Ein Token ist für jeden Teilnehmer zugewiesen 	
Nachbedingung:	<ul style="list-style-type: none"> - Die Teilnehmer haben die Umfrage abgeschlossen - Die Daten wurden gespeichert 	
Hauptszenario: (User- /Systemverantwortlichkeiten)	Akteure, Aktionen: 1. Die Teilnehmer greifen mit ihrem Token oder mit ihrer Emailadresse und Passwort (gilt nur für Dozenten und Leiter) auf die Umfrage zu 2. Die Teilnehmer füllen die Umfrage aus 3. Die Umfrage wird am Schluss beendet	Systemverantwortlichkeiten: 4. Die Daten werden in die DB gespeichert
	4a. Die Daten werden falsch in die DB gespeichert → Auf die Reihenfolge der Datenpersistierung achten	
Häufigkeit des Auftretens:	Bei jeder Umfrage	

Tabelle 35 UC Fully Dressed Umfrage teilnehmen

UC 11		Benutzer verwalten	
Umfang:	TeachEval Applikation		
Ebene:	Anwendung, Sicherheit		
Primärer Akteur:	Leiter		
Vorbedingungen:	- Leiter muss angemeldet sein		
Nachbedingung:	-		
Hauptzenario: (User- /Systemverantwortlichkeiten)	<p>Aktoren, Aktionen:</p> <ol style="list-style-type: none"> 1. Der Leiter geht zur „Benutzerverwaltung“ 3. Er kann die Benutzerliste begutachten oder weitere Benutzer über eine Excel Datei hinzufügen 5. Ein Benutzer macht eine längere Pause und soll deaktiviert werden. Der Leiter kann den Hebel aktiv auf inaktiv setzen 7. Ein anderer Benutzer weiss sein Passwort nicht mehr. Somit kann der Leiter dem Benutzer ein neues Passwort setzen. 9. Ein weiterer Benutzer wird nicht mehr Kurse unterrichten. Der Benutzer kann aus der Liste gelöscht werden. 	<p>Systemverantwortlichkeiten:</p> <ol style="list-style-type: none"> 2. Eine Liste aller Benutzer wie Leiter und Dozenten erscheint 4. Beim Hinzufügen wird die Excel Datei importiert und nicht existierende Benutzer hinzugefügt 6. Der Benutzer wird über eine POST Request in der Datenbank geändert 8. Passwörter werden gehasht und verglichen. Bei Übereinstimmung wird das neue Passwort in die DB gespeichert. 10. Der Benutzer wird als Instruktor in der Datenbank entfernt und wird von der Liste gestrichen. 	
		<ol style="list-style-type: none"> 2. Die Benutzerliste wird nicht angezeigt oder aktualisiert → Problemen in der DB Abfrage, Daten werden im Cache gespeichert (keine Aktualisierung mehr möglich) 3. Falsche Angaben im Excel, Importvorgang wird abgebrochen → Zuerst alle Angaben überprüfen, dann in die DB speichern <p>6./8./10. Änderungen werden nicht in die DB gespeichert → Probleme bei der DB Abfrage</p>	
Häufigkeit des Auftretens:	Bei jeder Umfrage		

Tabelle 36 UC Fully Dressed Benutzer verwalten

4. Weitere Anforderungen

4.1 Qualitätsmerkmale

Die TeachEval Webapplikation stützt sich auf die Qualitätsmerkmale gemäss ISO 9126. Nachfolgend werden alle Punkte erläutert und entschieden, inwiefern die Merkmale für unsere Applikation wichtig sind.

4.2 Zuverlässigkeit

Die Befragung muss zuverlässig laufen können, damit die Dozenten, Supervisoren, Studierende und Leiter an den Umfragen teilnehmen können. Eine Befragung kann im Nachhinein nicht mehr nachgeholt werden. Die Leiterin muss Umfragen erstellen und verwalten können, damit Befragungen überhaupt möglich sind. Die Auswertungen selber müssen korrekt ausgelesen und visualisiert werden. Ein Login Bereich sichert den zuverlässigen Einsatz der Applikation vor nicht autorisiertem Zugriff.

4.3 Benutzbarkeit

Die Webapplikation soll benutzerfreundlich und schnell erlernbar sein (siehe Kapitel 2.5 Benutzerfreundlichkeit). Die Applikation hält sich an den Web Standards wie W3C und ist mit Maus und Tastatur bedienbar. Zudem ist die Applikation mit dem Tablet und Handy bedienbar. Bei der Konzeption haben die Auftraggeberin und unser Betreuer ein Mitspracherecht an der Webapplikation.

4.4 Effizienz

Die Effizienz der Webapplikation wird anhand von diversen Studien gemessen. Dabei ist der maximale Richtwert 7 Sekunden. Antwortzeiten einer Webapplikation, die über 3 bis 4 Sekunden gehen, werden schon als störend empfunden. Bei Antwortzeiten über 8 bis 10 Sekunden versagt nicht nur die Geduld der Besucher, es bricht auch deren roten Faden, welche Tätigkeit er gerade nachgehen wollte.

Damit wir anständige Antwortzeiten erreichen können, werden während der Implementierung Refactorings vorgenommen. Grosse Datenmengen werden nicht auf einem Mal geladen und werden wenn notwendig komprimiert. Wir setzen uns daher selber den Richtwert von 3 Sekunden, die eine Seite beim Laden maximal brauchen darf.

4.5 Wartbarkeit

Die Wartbarkeit der TeachEval Applikation ist ein wichtiger Bestandteil, da die Endversion in den Arbeitsprozess der Qualitätssicherung für Kurse ihren Einsatz findet. Dies bedeutet, dass eine Weiterführung sehr wahrscheinlich ist. Aus diesem Grund sind auch die Modifizierbarkeit und die Skalierbarkeit wichtig. Dies betrifft einerseits die Programmierung der Applikation selber, jedoch auch die dahinterliegende Datenbank, die wir mit dem Einsatz unserer Technologien weit möglichst sicherstellen. Ausserdem sollte die Architektur der Applikation so konzipiert sein, dass unerwartete Wirkungen von Änderungen keinen Einfluss auf andere Teilkomponenten haben. Dadurch soll Instabilität vermieden werden und Fehler in ihrem Problembereich isoliert bleiben.

4.6 Übertragbarkeit

TeachEval soll betriebssystemunabhängig betrieben werden können. Dabei ist die Übertragung der Applikation auf andere Web Server durchaus möglich. Ein Installationshandbuch soll dem Administrator beim Einrichten der Applikation behilflich sein.

4.7 Schnittstellen

4.7.1 Benutzerschnittstelle

Die Schnittstelle zwischen dem Benutzer und den Daten ist das grafische Web Frontend, welches auf HTML, CSS, JavaScript, JQuery, etc. basiert. Über das Frontend kann der Benutzer seine Anpassungen und Aktionen in Form eines Excel-Sheets hinaufladen. Umfragen und Auswertungen können über das Frontend direkt angesprochen und bedient werden.

4.7.2 Hardwareschnittstelle

Die Datenbank und die Webapplikation werden auf demselben Web-Server in Betrieb genommen.

4.7.3 Softwareschnittstelle

Es bestehen keine externen Schnittstellen. Alle externen Libraries werden lokal in unser Webprojekt hinzugefügt. Damit stellen wir sicher, dass wir nicht abhängig sind von Drittanbietern bzw. Drittkomponenten.

5. Abbildungsverzeichnis

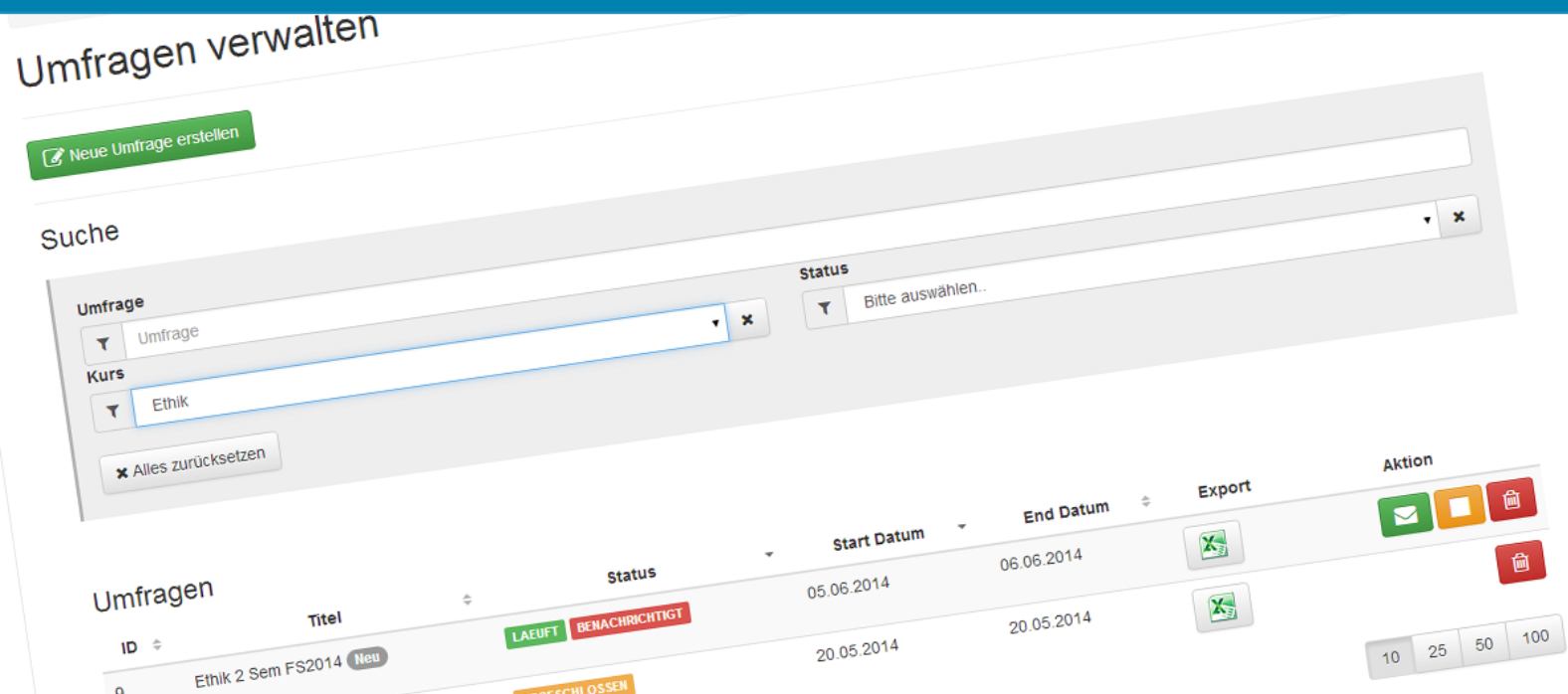
Abbildung 27 Use Case Diagramm für OdA G ZH	70
---	----

6. Tabellenverzeichnis

Tabelle 16 Benutzer Charakteristik	66
Tabelle 17 ISO 9241-110.....	68
Tabelle 18 UC Brief An-/Abmelden	71
Tabelle 19 UC Brief Umfrage CRUD.....	71
Tabelle 20 UC Brief Token generieren.....	71
Tabelle 21 UC Brief Token exportieren	71
Tabelle 22 UC Brief Token per Mail versenden	71
Tabelle 23 UC Brief Excel Template importieren	72
Tabelle 24 UC Brief Auswertungen ansehen.....	72
Tabelle 25 UC Brief Korrelation von Fragen	72
Tabelle 26 UC Brief Export als Excel oder PDF	72
Tabelle 27 UC Brief Umfrage teilnehmen.....	72
Tabelle 28 UC Brief Benutzer verwalten	72
Tabelle 29 UC Fully Dressed An-/Abmelden	74
Tabelle 30 UC Fully Dressed Umfrage CRUD	75
Tabelle 31 UC Fully Dressed Excel Template importieren.....	76
Tabelle 32 UC Fully Dressed Auswertung ansehen	76
Tabelle 33 UC Fully Dressed Korrelation von Fragen	77
Tabelle 34 UC Fully Dressed Export als Excel oder PDF.....	77
Tabelle 35 UC Fully Dressed Umfrage teilnehmen.....	77
Tabelle 36 UC Fully Dressed Benutzer verwalten.....	78

TeachEval

Web App für die Unterrichtsevaluation



User Interface (UI)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Einführung

1.1 Zweck

Dieses Dokument beschreibt das User Interface für die TeachEval Web Applikation mit Hilfe von Wireframes. Die Wireframes wurden mit dem Wireframe Tool Balsamiq Mockups erstellt (Wireframe Tool Balsamiq Mockups, 2014).

1.2 Gültigkeitsbereich

Das UserInterface Dokument dient als Grundlage für das Frontend (siehe Glossar) und ist über die gesamte Projektdauer gültig.

2. Wireframes

Damit wir unsere Auftraggeberin besser verstehen können, erstellten wir Wireframes. Mit den Wireframes wollen wir bereits einen frühen konzeptuellen Papier-Prototyp erstellen um eine gemeinsame Diskussionsbasis zu schaffen. Dabei zeigt der Papier-Prototyp vor allem die wichtigsten Inhaltselemente, die Seitenstruktur, wie auch die Struktur der Informationen, auf. Zudem visualisiert es die Benutzerschnittstellen, mit denen der Benutzer darauf agieren kann.

Wireframes beschreiben nicht nur das visuelle Design, sondern sie helfen uns, wichtige Funktionen von den Unwichtigen zu trennen. Damit wollen wir mit allen Beteiligten das ungefähre Layout für die Web Applikation TeachEval festlegen und erhoffen uns so, dass die meisten Änderungen bereits frühzeitig geklärt sind.

2.1 Grundgerüst

Bei der Realisierung von TeachEval entschieden wir uns für eine *Single Page Applikation*. Das Grundgerüst ist bei jedem Seitenaufruf vorhanden.

- **HEADER:** Der Header beinhaltet das Logo der OdA G Zürich und stellt ein Login zu Verfügung.
- **MAIN MENU:** Das Menu ist für die Navigation zuständig.
- **CONTENT:** Im Content Bereich wird dynamisch der jeweilige Inhalt geladen. Die einzelnen Inhalte werden in den folgenden Kapiteln detaillierter beschrieben.
- **FOOTER:** Der Footer beinhaltet einen Copyright Text.

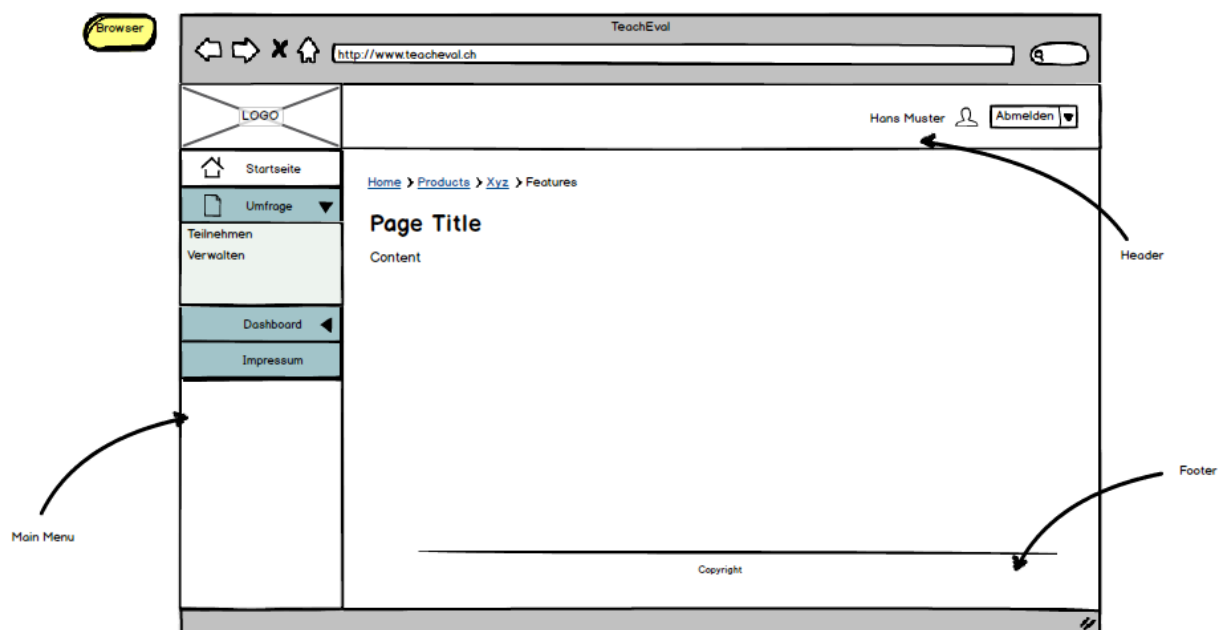


Abbildung 28 Grundgerüst Desktop

2.2 Mobile

Die Umfragen werden von Studenten, Dozenten und Supervisoren ausgefüllt (siehe Anforderungsspezifikation). Damit TeachEval auch auf kleinen Geräten bedienbar ist, wurde eine mobile Version ausgearbeitet.



Abbildung 29 Grundgerüst Mobile

2.3 Umfrage teilnehmen

Jeder Teilnehmer (siehe Anforderungsspezifikation) erhält eine E-Mail oder einen Brief, um an einer Umfrage teilzunehmen. In dieser E-Mail, bzw. Brief wird dem Teilnehmer eine eindeutige, anonyme Umfragekennung zugewiesen und als URI verpackt. Damit hat der Teilnehmer zwei Möglichkeiten an einer Umfrage teilzunehmen. Entweder kann er direkt auf den zugeteilten URI klicken oder er navigiert im *Main Menu* auf *Umfrage teilnehmen* und er gibt seine Kennung in die Eingabemaske ein.

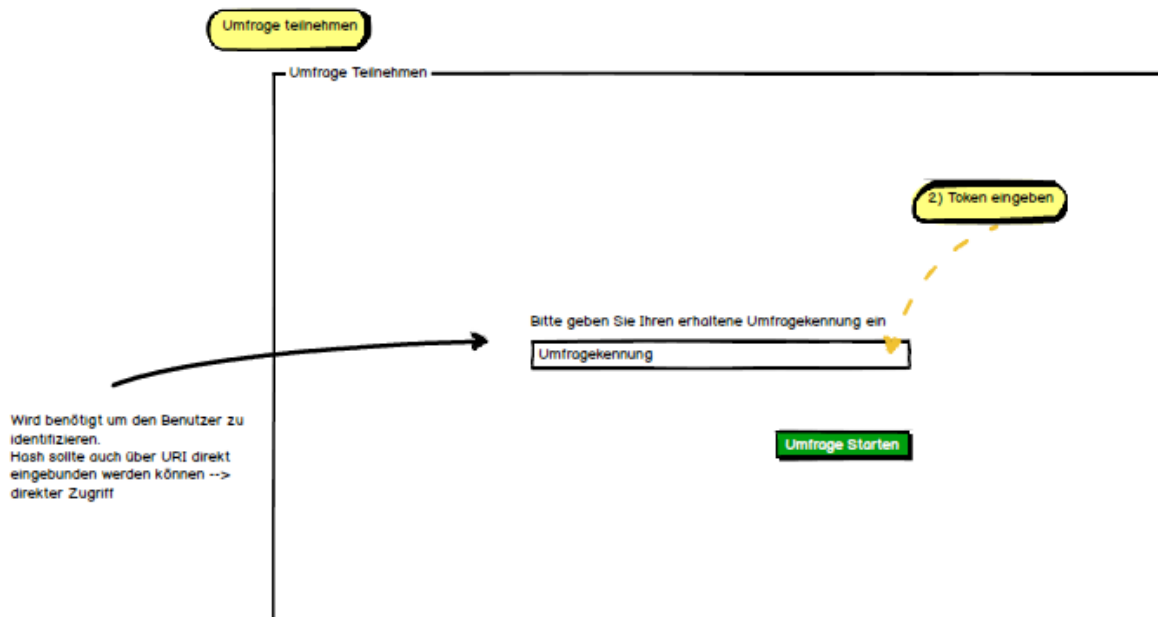


Abbildung 30 Zugang Umfrage

Wenn der Teilnehmer eine gültige Umfragekennung an TeachEval sendet, erhält er Zugang zu seiner Umfrage. Die Fragen der Umfragen werden untereinander aufgelistet. Die Antworten auf die Fragen werden je nach Fragetyp als Texteingabe oder als Auswahl (Checkbox, Radio) angezeigt. Eine Progress Bar soll dem Teilnehmer aufzeigen, wie lange seine Umfrage noch geht.

Umfrage Teilnehmen

Progress Bar

Fertig

Frage 1: Ich erhielt die Möglichkeit mir meine (Lern)Fragen zu beantworten.

So war es So sollte es sein

Antwort 1 Antwort 1

Antwort 2 Antwort 2

Antwort 3 Antwort 3

Frage 2: Ich fühlte mich genügend kompetent, die Studierenden in ihrem Lernprozess zu unterstützen.

Antwort 1

Antwort 2

Antwort 3

Frage 3: Was ist mir gut gelungen?

Abbrechen Next

Abbildung 31 Umfrage

2.4 Umfrage verwalten

Der Leiter/-in kann neue Umfragen als Excel hochladen und erstellen. Eine Tabelle listet alle erstellten Umfragen auf.

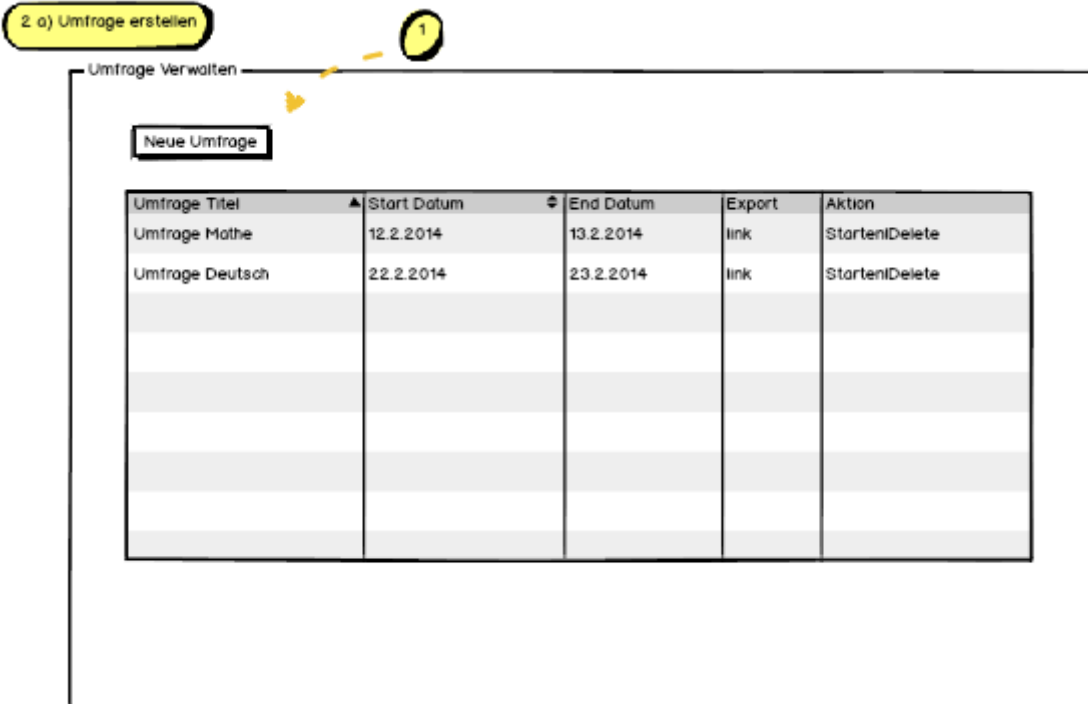


Abbildung 32 Umfrage Übersicht

Eine neue Umfrage kann auf der gleichen Seite wie die Umfrage Übersicht hochgeladen und erstellt werden. Nachdem eine Umfrage erfolgreich hochgeladen wurde, wird sie in der Tabelle On-The-Fly hinzugefügt.

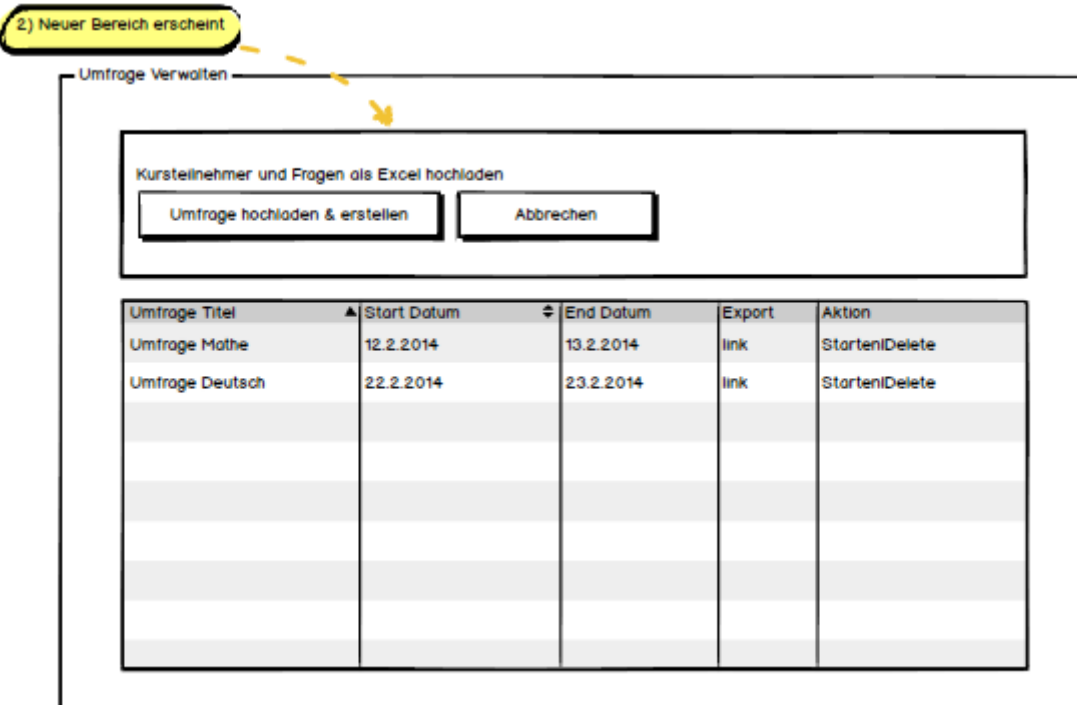


Abbildung 33 Neue Umfrage erstellen

2.5 Auswertung

Eine Übersicht zeigt alle vorhandenen Auswertungen auf. Diese Übersicht ist nebst dem Leiter auch den Dozenten zugänglich. Es können nur Auswertungen von abgeschlossenen Umfragen eingesehen werden.

3) Auswertung

Die definitive Planung der Auswertung wird während der Realisierung der Umfrage ausgearbeitet.

Dashboard Übersicht

Übersicht

Umfrage Titel ▲	Status ▲	Anzahl Teilnehmer▲	Anzahl Instruktoren▲	mitgemachte Teilnehme▲	Aktion
Umfrage Mathe	ABGESCHLOSSEN	9	1	9	Details
Umfrage Deutsch	ARCHIVIERT	18	2	15	Details

Abbildung 34 Auswertung Übersicht

Eine detaillierte Auswertung zeigt eine Übersicht der Umfrage und listet alle Fragen mit einem Balkendiagramm auf. Die Auswertung soll als Excel exportierbar sein.

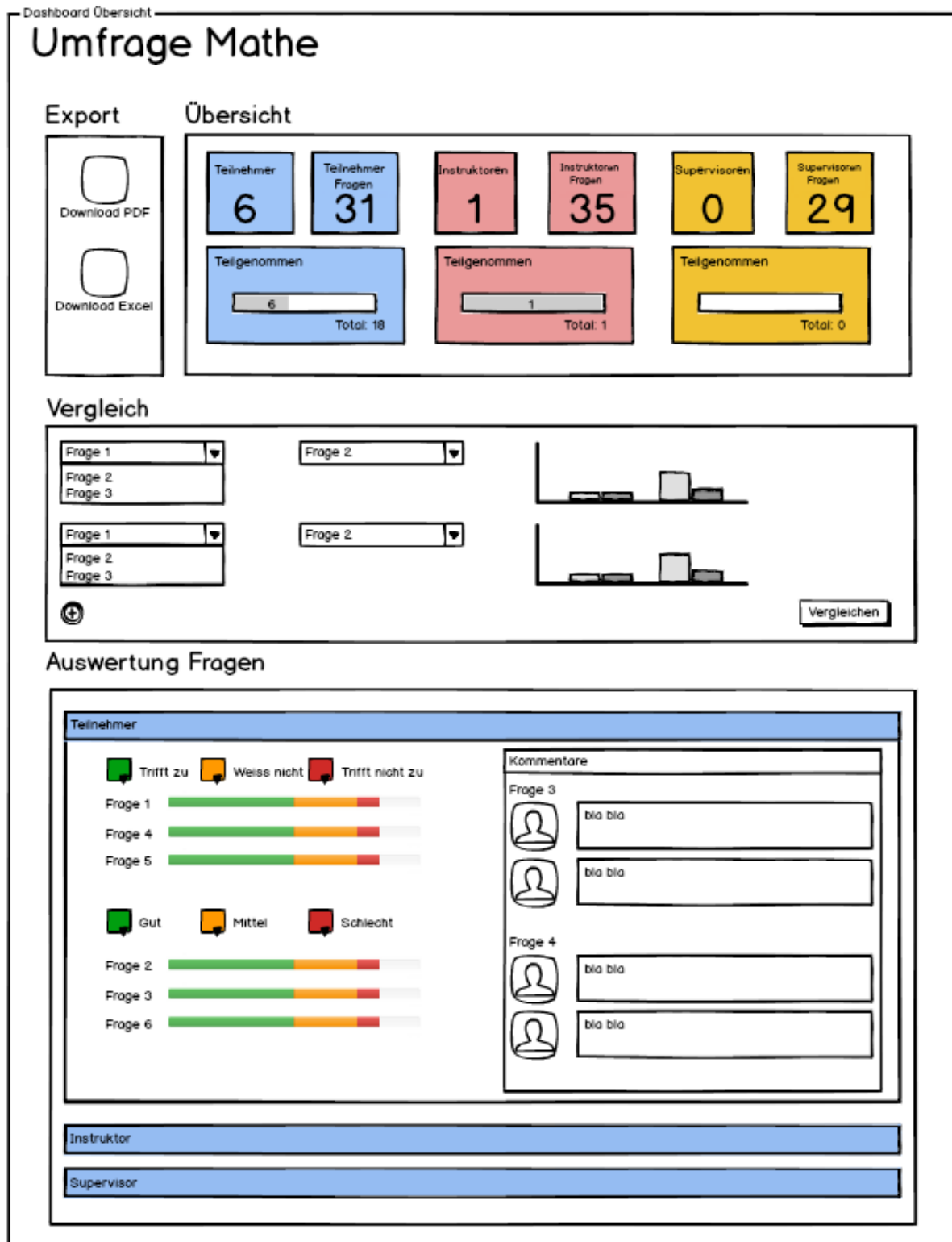


Abbildung 35 Detaillierte Auswertung

3. Abbildungsverzeichnis

Abbildung 28 Grundgerüst Desktop.....	84
Abbildung 29 Grundgerüst Mobile.....	85
Abbildung 30 Zugang Umfrage.....	85
Abbildung 31 Umfrage	86
Abbildung 32 Umfrage Übersicht.....	87
Abbildung 33 Neue Umfrage erstellen.....	87
Abbildung 34 Auswertung Übersicht	88
Abbildung 35 Detaillierte Auswertung.....	89

TeachEval

Web App für die Unterrichtsevaluation

Auswertung

Suche

The screenshot displays the 'Auswertung' (Evaluation) section of the TeachEval web application. It features a search bar with a 'Suche' (Search) button and a 'Status' dropdown menu. Below the search bar, there are filters for 'Umfrage' (Survey) and 'Kurs' (Course), both with dropdown menus and a 'Bitte auswählen..' (Please select..) prompt. A button labeled 'Alles zurücksetzen' (Reset all) is also visible. The main content area shows a table with columns for 'Teilgenommen / Total' (Participated / Total), 'Start Datum' (Start Date), 'End Datum' (End Date), and 'Aktion' (Action). The table contains several rows of data, including dates like 05.06.2014, 04.06.2014, 23.05.2014, 21.05.2014, and 20.05.2014. Overlaid on the bottom left of the screenshot is a code snippet in Java, showing annotations like @GET, @Path, @Produces, and @PathParam, along with a public method named 'getInstructorById'.

Software Architektur Dokument (SDD)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Einführung

1.1 Zweck

Dieses Dokument ist ein Teil des Designs und beschreibt die Webarchitektur von TeachEval. Hier werden die wichtigsten architektonischen Entscheidungen und Abläufe beschrieben.

1.2 Gültigkeitsbereich

Das Softwarearchitektur Dokument gilt für die gesamte Projektdauer.

1.3 Übersicht

Dieses Dokument gibt eine Übersicht über die Softwarearchitektur von TeachEval. Im Kapitel Systemübersicht wird die physische und logische Architektur und deren Unterschiede vorgestellt. Als Nächstes wird der Architekturstil REST von TeachEval definiert. Die Kapitel Projektstruktur, physische und logische Architektur geben Einblicke in die Projektstruktur und in die physische und die logische Abhängigkeiten zwischen Komponenten. Im Kapitel Open Source Software werden alle Technologien und deren Lizenzen aufgelistet. Zum Schluss soll im Kapitel Datenbankspeicherung das Datenbankschema näher beschrieben werden.

2. Systemübersicht

TeachEval ist eine REST-basierte Webapplikation (siehe Kapitel REST), welche mit Java Jersey und AngularJS entwickelt worden ist. Bei Java Jersey handelt es sich um eine Referenzimplementierung von *Java API for RESTful Web Services* (kurz JAX-RS), die die Verwendung des Softwarearchitekturstils *Representational State Transfer* (REST) im Rahmen von Webservices ermöglicht. Um die Dynamik des Webbrowsers besser auszunutzen, wurde auf der Clientseite AngularJS oder auch einfach Angular, verwendet. Angular ist ein von Google entwickeltes Open-Source Framework, welches Single-Page-Anwendungen sowie das MVC-Modell unterstützt.

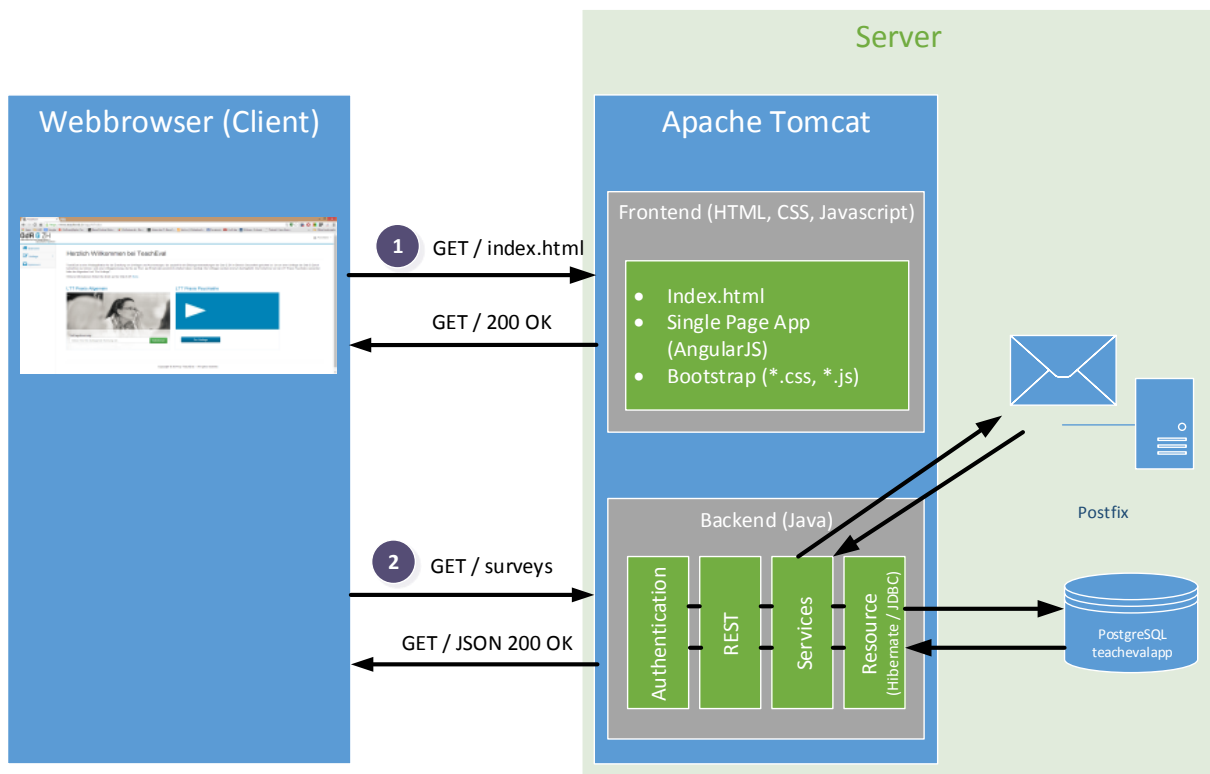


Abbildung 36 Systemübersicht TeachEval

Die Webapplikation TeachEval wird im Verlaufe dieser Dokumentation als *physische* und *logische* Architektur beschrieben. Die physische Architektur beschreibt das Zusammenspiel zwischen *Client* und *Server*, während die *logische* Architektur das Projekt TeachEval in Frontend und Backend beschreibt. Das Projekt TeachEval wurde auf dem Server eingerichtet (siehe Abbildung 36). Die logische Architektur wird im Kapitel 6. näher eingegangen. In den folgenden Abschnitten werden die einzelnen physischen Schichten genauer beschrieben.

2.1 Client

Der Client greift mit seinem Webbrowser auf die TeachEval Webapplikation zu. Damit der Client die Benutzeroberfläche angezeigt bekommt, wird diese beim Frontend, welches auf dem Server liegt, angefragt und übermittelt. Anschliessend kann der Client Interaktionen über die Benutzeroberfläche tätigen. Seine Modifikationen werden anschliessend an das Backend, welches sich ebenfalls auf dem Server befindet, übertragen. Damit die Darstellung der Web App beim Client erfolgreich sein kann, benötigt dieser einen Webbrowser mit der Unterstützung von HTML 5, CSS 3 und JavaScript.

2.2 Server

Während der Entwicklung von TeachEval läuft die Applikation auf einem von der HSR zur Verfügung gestellten virtuellen Testserver, welcher mit Java, Apache Tomcat 7 und PostgreSQL aufgerüstet wurde. Tomcat 7 ist der Web Container und für die Persistierung der Daten wird PostgreSQL eingesetzt. Bereits vorinstalliert war Redmine, welches im Rahmen dieser Arbeit als Projektmanagement-Tool eingesetzt wurde.

Zusätzlich zum Testserver musste auch ein produktiver Server für die Anwendung installiert werden. Da sich die IT Infrastruktur der OdA G ZH momentan in der Restrukturierung befindet, wurde entschieden, den produktiven Server bei einem externen Anbieter laufen zu lassen. In Zukunft könnte es sein, dass die Applikation auf einen internen Webserver der OdA G ZH gewechselt wird. Für die produktive Umgebung wurden dieselben Technologien verwendet wie auf dem Testserver. Zusätzlich wurde eine Domäne namens *teacheval.ch* bzw. *www.teacheval.ch* registriert, welche HTTP-Request Anfragen entgegen nimmt und diese auf HTTPS umleitet. Die Aufsetzung des Servers wird im Kapitel Deployment genauer beschrieben.

Auf dem Server ist die Webapplikation abgelegt, die in zwei Unterprojekte aufgeteilt wurde. Einfachheitshalber werden diese als Frontend und Backend bezeichnet. Im Frontend befindet sich die Benutzeroberfläche, die bei Anfragen übermittelt wird. Das Backend ist für die Verarbeitung der Daten zuständig.

3. Representational State Transfer (REST)

Rest ist ein Architekturstil, mit dem Webservices realisiert werden können und ist eine Bezeichnung für ein Programmierparadigma, welches von Dr. Thomas Roy Fielding in seiner Dissertation „*Architectural Styles and the Design of Network-based Software Architectures*“ beschrieben ist. Es benutzt Prinzipien, die in grossen, verteilten Anwendungen wie bspw. im World Wide Web eingesetzt werden. Ressourcen werden durch URIs (Unique Resource Identifier) im Netzwerk eindeutig identifiziert. Eine Ressource ist eine Informationsquelle, die beispielsweise Text, Bilder oder Dateien zurückliefert. Eine Ressource kann aber auch ein abstraktes Konstrukt sein, das intern einen Workflow mit Operationen und Funktionen auslöst, bspw. das Versenden von E-Mails (Thomas Roy Fielding, 2000, S. 76).

Für die TeachEval Applikation setzen wir eine REST API ein, die es den Benutzern erlaubt, ihre benötigten Informationen anzufordern und diese in einer leserlichen Form auf der Webseite angezeigt zu bekommen. Vorausgesetzt die Benutzer sind dazu berechtigt.

3.1 Richardson Maturity Model

Das Richardson Maturity Model beschreibt den Reifegrad der REST Schnittstelle. Die REST API wird dabei in vier Stufen unterschieden:

Level 0: POX

- *Verwendet XML-RPC oder SOAP*
- *Der Service wird über eine einzelne URI adressiert (REST, 2014)*
- *Verwendet eine einzelne HTTP-Methode (POST)*

Level 1: Resources

- *Verwendet verschiedene URIs und Ressourcen*
- *Verwendet eine einzelne HTTP-Methode (POST)*

Level 2: HTTP Verbs

- *Verwendet verschiedene URIs und Ressourcen*
- *Verwendet mehrere HTTP-Verben wie bspw. GET, POST, PUT, DELETE*

Level 3: Hypermedia Controls

- *Basiert auf HATEOAS; verwendet Hypermedia für Navigation*
- *Verwendet verschiedene URIs und Ressourcen*
- *Verwendet mehrere HTTP-Verben*

Zu Beginn des Projektes war noch nicht klar, wo der produktive Server für die Oda G befinden wird und so haben wir aufgrund diverser Quellen auf die HTTP Verben PUT und DELETE verzichtet, da diese typischerweise von Firewalls blockiert werden (MSDN HTTP-Verbs, 2014) (Hanselman, 2014). Deshalb verwenden wir für TeachEval eine Level 2 REST-Schnittstelle, mit der die Daten mit GET geholt und mit POST verschickt werden.

4. Projektstruktur

TeachEval wurde als Maven Projekt über das Maven Eclipse-Plugin erstellt. Somit entstand folgende Projektstruktur:

- Java Resources
Enthält alle Java Klassen und Ressourcen. Sie repräsentiert den **Backend Bereich** von TeachEval.
- pom.xml
POM steht für *Project Object Model* und repräsentiert in XML das TeachEval Maven Projekt. Darin werden alle Abhängigkeiten, Plugins, Eigenschaften und weitere Informationen eines Projektes definiert.
- WebContent
Der WebContent ist mit Ausnahme von *WEB-INF* der **Frontend Bereich**. Im *WEB-INF* werden diverse Konfigurationen und Libraries als Dateien hinterlegt:
 - *web.xml*
Konfigurationsdatei für Tomcat. Darin werden Servlets für Jersey definiert, Ressourcen für die Anbindung der Datenbank oder E-Mail Sessions oder Listeners.
 - *hibernate.cfg.xml*
Konfigurationsdatei für Hibernate. Die wichtigsten Parameter sind Parameter für Verbindungskonfigurationen und die Mappings zu den POJO-Klassen.
 - *Log4j.properties*
Konfigurationsdatei für Log4J. Darin können noch spezifischere Logging-Einstellungen konfiguriert werden als das Logging Framework von Tomcat.
 - *lib*
Im Verzeichnis *lib* werden alle Libraries als JAR Datei abgelegt.

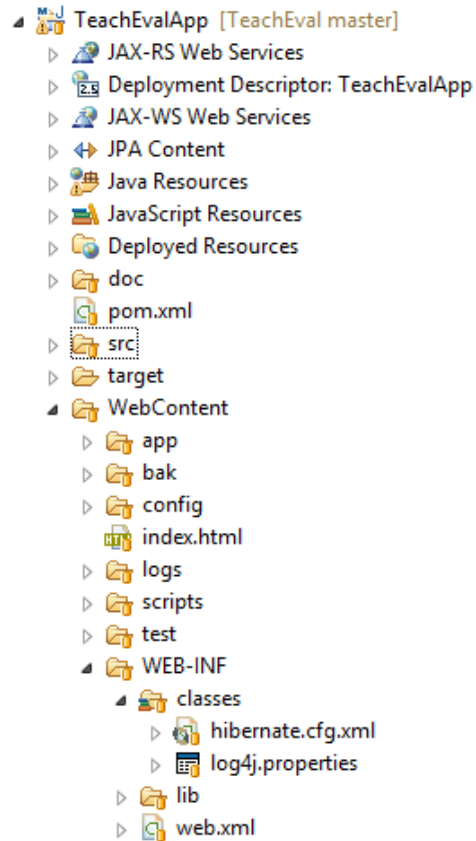


Abbildung 37 Projektstruktur TeachEval

4.1 Frontend

Die Projektstruktur im Client besteht aus mehreren Verzeichnissen. Sie dient der logischen Gliederung von Aufgaben und Features.

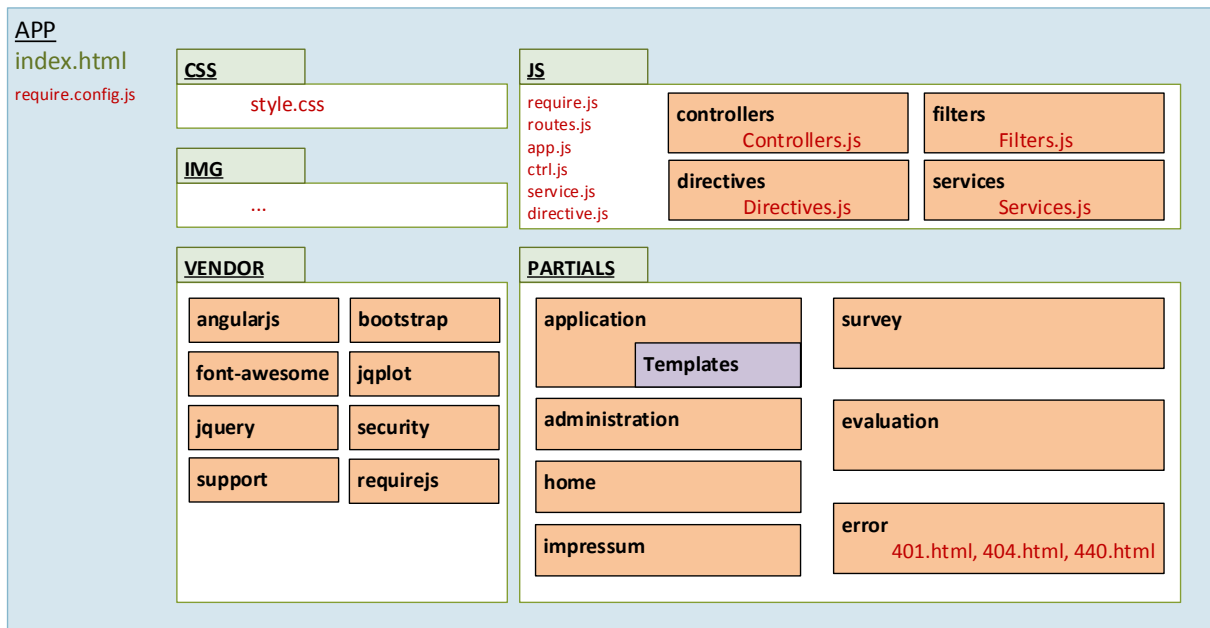


Abbildung 38 Projektstruktur Client

- /app
Dies ist das Hauptverzeichnis und enthält die eigentliche Projektstruktur im Clientteil. Es enthält alle öffentlich abrufbaren Dateien. Dies sind Grafiken, CSS- und JavaScript Dateien und die einzelnen HTML Seiten. Im App Verzeichnis sind zudem die index.html, welche der Startpunkt der Webapplikation ist und eine require.config.js Datei. Die require.config.js2 wird von der index.html geladen und ist für die Abhängigkeiten innerhalb der Web App verantwortlich. Die Abhängigkeiten zu externen oder internen Ressourcen werden asynchron geladen.
- /vendor
Im Vendor-Verzeichnis werden jegliche Fremdmodule abgelegt. Das Fremdmodul „angularjs“ ermöglicht uns unsere Webapplikation mit angularjs zu realisieren. Angular läuft mit der Unterstützung von JQuery. Das Bootstrap Verzeichnis ist ein Responsive Front-End Framework, welches uns bei der Kompatibilität unterschiedlichsten Geräten und Webbrowsern unterstützt. Das Font-Awesome Plug-In liefert uns ein breites Set an Open Source Icons. RequireJS ist das darunter laufende Plug-In für das Verwalten der Abhängigkeiten in der App. Im Verzeichnis Support befinden sich unter anderem die Dateien html5shiv 3und es5-shim4 für den IE8 Support.
- /css
Im CSS Ordner ist ein eigenes Stylesheet abgelegt, das auf dem Bootstrap Framework aufbaut.

² basiert auf dem requirejs Plug-In: (RequireJS Offizielle Webseite, 2014)

³ Liefert die Unterstützung von HTML5 Elemente für den IE8

⁴ Behebt die nicht unterstützten JavaScript Funktionen für IE8 (z.B. indexOf-Funktion)

Anpassungen oder Neuerungen am CSS sollten in dieser Datei vorgenommen werden. Dabei bleibt Bootstrap selbst unberührt und kann später durch neuere Versionen ersetzt werden.

- /img
Alle Bilder sind im Verzeichnis img abgelegt.
- /js
Das JS-Verzeichnis enthält die Logik der Webapplikation. Darunter befinden sich die Angular Controller, Direktiven, Filter und Services. Die require.js Datei lädt das App Modul, das Controller Modul und das Direktive Modul. Die Navigation auf der Web App wird im routes.js konfiguriert.
- /partials
Im partials Ordner werden die HTML Seiten und die HTML Templates abgelegt.

4.2 Backend

Durch Maven ist der Backend-Bereich in folgende vier Hauptverzeichnisse unterteilt:

- *src/main/java*
Der gesamte Source Code von TeachEval.
- *src/test/java*
Der gesamte Test Code von TeachEval.
- *src/test/resources*
Wurde im Rahmen des Projektes nicht benötigt.
- *src/main/resources*
Darin befinden sich Dateien wie Datenbank-Skript, Datenbank-Schema und Templates.

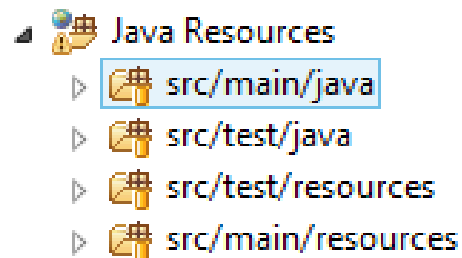


Abbildung 39 Projektstruktur im Backend

4.3 Verwendete Libraries

Library	Version	Verwendung
Jersey	1.18.1	Für die Verwendung des Software-Architekturstils REST im Rahmen von Webservices.
Hibernate	4.3.1	Objekt-Relationales Mapping zwischen Java und Datenbank.
Apache POI	3.10	Für das Lesen und Schreiben von Microsoft Dateien. Für unseren Zweck wird nur Excel verwendet.
Java Mail	1.5.2	Framework für das Versenden von Mails.
Quartz	2.2.1	Eine Library für <i>Job Scheduling</i> . Wird für das Automatisieren bzw. für das Starten oder Beenden der Umfragen verwendet.
Log4J	1.2.13	Log4J ist ein Logging-Framework zum Loggen von Anwendungsmeldungen.
Bootstrap	3.1.0	CSS und JavaScript Framework für die Gestaltung von TeachEval
Bootstrap UI		JavaScript Teil werden als AngularJS Directive angeboten
Font-Awesome	4.0.3	Liste von Open Source Icons.
JQuery	1.11.0	JavaScript Bibliothek
metisMenu		Seitenmenu, dessen Entwicklung auf JQuery basiert.
RequireJS	2.1.13	Verwaltet die Projekt-Abhängigkeiten zu internen und externen Ressourcen.

Library	Version	Verwendung
domReady		Plugin erkennt wann der DOM fertig geladen ist, damit als nächstes die App gestartet werden kann.
passwordmeter	-	Plugin für die Messung der Passwortqualität
sha256	0.8	Plugin hasehd Passwort als sha256.
es5-shim	-	Kompatibilitätsbibliothek für EcmaScript 5 Funktionen, die für IE6-9 unterstützt werden.
html5shiv	3.7.1	Kompatibilitätsbibliothek für HTML5 Element, die für IE6-9 unterstützt werden.
respond	-	CSS3 Kompatibilitätsbibliothek für IE6-8.
placeholder	-	JQuery-Plugin für den Support von Placeholder in IE8 und IE9

Tabelle 37 Eingesetzte Libraries

5. Physische Architektur

Die physische Architektur von TeachEval sieht vor, dass der produktive Webserver HTTP Anfragen über SSL/TLS entgegennimmt. Die Clients greifen mit ihren PCs, Tablets oder Smartphones auf den Webserver zu. Die folgende Abbildung zeigt eine Übersicht über das Zusammenspiel der verschiedenen Komponenten.

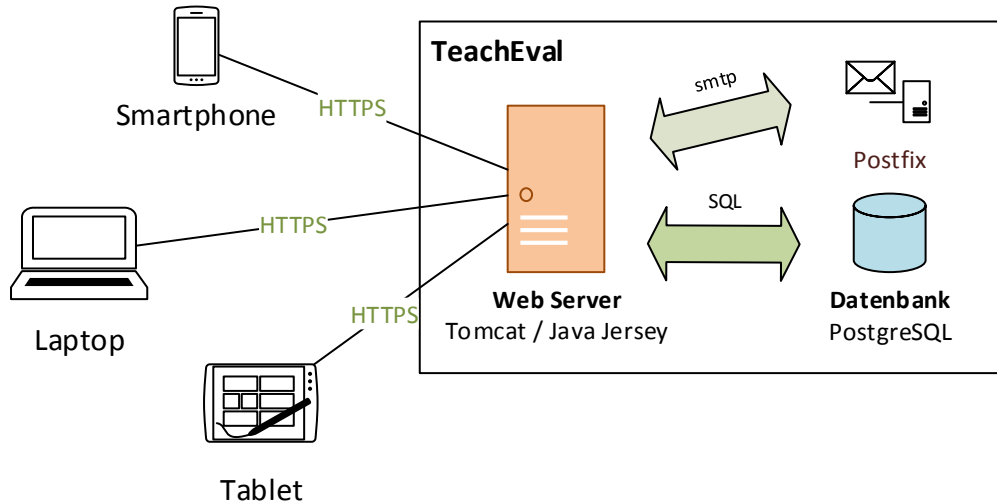


Abbildung 40 Physische Architektur

5.1 Datenbank

Die Datenbank ist für die Speicherung und Archivierung der TeachEval Daten zuständig. Dabei wird die Open-Source Plattform *PostgreSQL 9.3.4* verwendet. Als Schnittstelle zwischen Java EE und Datenbank kommt Hibernate zum Einsatz. Für das Abspeichern und Holen von Daten wird über SQL oder HQL (SQL für Hibernate) durchgeführt.

5.2 E-Mail

Als E-Mail Dienst wurde auf dem produktiven Server POSTFIX vorinstalliert. Er wird für die automatische Einladung der Teilnehmer zu einer Umfrage benötigt.

Auf dem Testserver steht uns aus Sicherheits- und Spamgründen kein E-Mail Dienst zur Verfügung, da dieser die E-Mails von und an nicht HSR E-Mail-Adressen blockiert. Daher muss der Absender und der Empfänger eine HSR E-Mailadresse haben, um an der Firewall vorbeizukommen. Für den produktiven Einsatz von TeachEval ist der Testserver der HSR daher ungeeignet.

5.3 Deployment

Der produktive Server wird von der Firma *Serverbase* gehostet. Es handelt sich um einen virtuellen Server, auf dem Debian 3.2.0 mit Intel Xeon 1.8 GHz und mit 2 GB Arbeitsspeicher läuft.

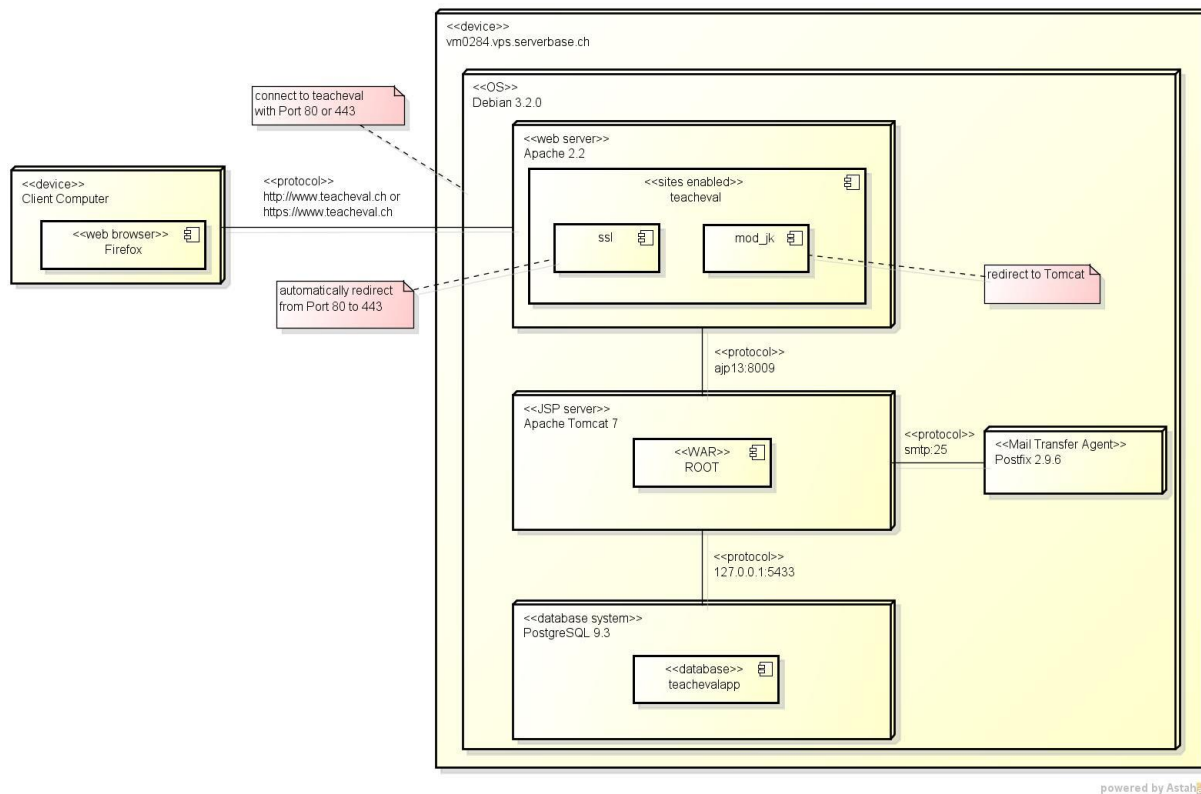


Abbildung 41 Deployment Diagramm

In Abbildung 41 wird die detaillierten Abhängigkeiten für einen erfolgreichen Zugriff aufgezeigt. Der Client greift mit seinem Webbrowser auf die TeachEval App über die Domäne teacheval.ch zu. In Apache wurde SSL so konfiguriert, so dass der Zugriff auf die Domäne teacheval.ch automatisch mit Port 443 umgeleitet wird. Mittels dem Modul *mod_jk* lässt sich Apache Tomcat einfach mit Apache verbinden. Die Kommunikation untereinander erfolgt mit dem Protokoll *ajp13* auf Port *8009*. Die Webapplikation wurde als WAR-Datei mit dem Namen *ROOT* im Ordner */var/lib/tomcat7/webapps* hinzugefügt. Tomcat entpackt automatisch alle nicht Java-Ressourcen wie (HTML, CSS, XML, usw.) aus dem *ROOT.war* Datei. Eine der entpackten Daten ist die *hibernate.cfg.xml* Datei, die Verbindungskonfigurationen, SQL-Dialekte und Mappings zu der Datenbank *teachevalapp* beinhaltet. Eine weitere Datei ist die Datei *log4j.properties*, welches Logging-Konfigurationen für Tomcat bereitstellt. Mit Postfix als Mail Transfer Agent (MTA) kann auf Port 25 E-Mails nach aussen versendet werden.

6. Logische Architektur

In diesem Kapitel wird die logische Projektstruktur von TeachEval und deren Architektur beschrieben.

6.1 Frontend

Das Frontend beschreibt die Benutzeroberfläche als AngularJS Applikation unter Einsatz moderner Technologien wie HTML5 und CSS3, die zum Client gesendet wird, wenn dieser die Webapplikation über seinen Webbrowser aufruft. Die *index.html* ist der Eingangspunkt in die Single Page App. Für die dynamischen Darstellung der Elemente auf verschiedene Displaygrößen und auf verschiedenen Webbrowsern wurde Bootstrap 3 und Angular UI Bootstrap verwendet.

6.1.1 AngularJS

AngularJS ist ein modernes clientseitiges MVC (Model-View-Controller) Framework, das auf JavaScript und JQuery basiert. Es läuft daher in einem Webbrowser und hilft uns bei der Entwicklung von dynamischen Single Page Apps die mit AJAX Aufrufe die Daten vom Server holt. Die folgende Abbildung beschreibt die Funktionsweise von Angular (Entwicklertagebuch AngularJS Structure, 2014).

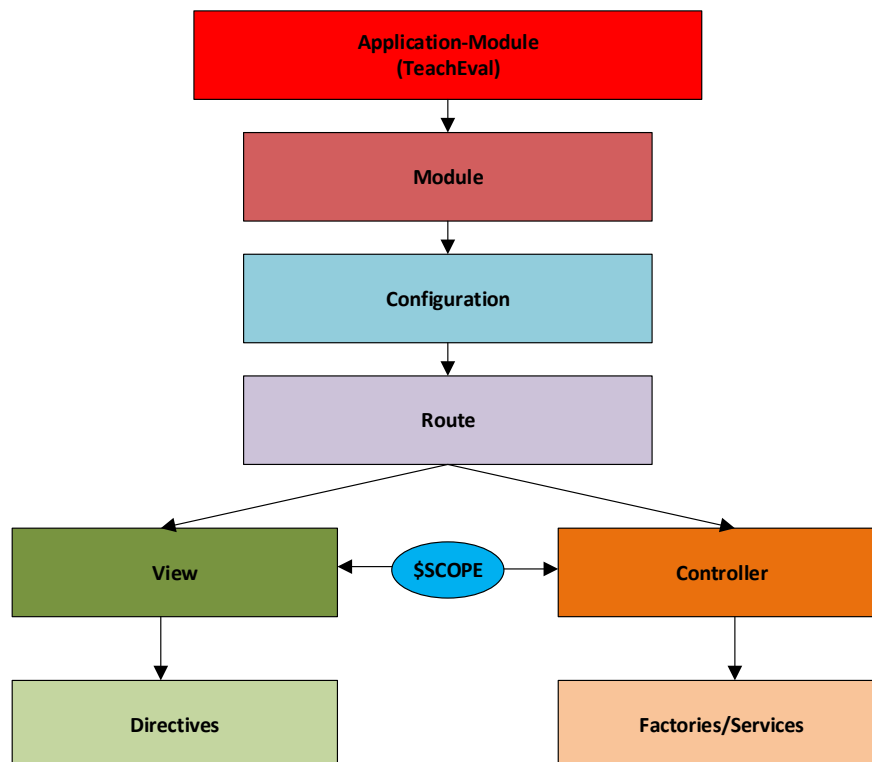


Abbildung 42 AngularJS Architektur (Entwicklertagebuch AngularJS Structure, 2014)

- Module
Ein Modul ist ein Container für die unterschiedlichen Bereiche der Web App wie bspw. controllers, services, filters, directives, etc.
- Configuration
Die Module sind in unterschiedlichsten Arten konfigurierbar. Wir haben für unsere App, einen Debug Mode, der automatisch Logs auf der Konsole ausgibt oder diese vor dem Benutzer verbirgt.

- **Route**
Die Seiten-Navigation ist als Route konfiguriert. Es werden jeweils die Webseite, der dazugehörige Controller und die Autorisierung angegeben. Auf der Client-Seite dient die Autorisierung nur für die Anzeige der richtigen Seiten für die richtigen Benutzer. Für einen Angreifer kann dies leicht umgangen werden. Damit aber die Webseite dem Besucher auch einen Mehrwert bringt, müssen die Daten vom Server beantragt werden. Diese sind durch einen Token serverseitig geschützt.
- **View**
Die View wird mit HTML Templates zusammengebaut. Das AngularJS Framework erweitert das Vokabular von HTML und aktualisiert einzelne Bereiche in der Seite automatisch, ohne manueller Intervention.
- **Controller**
Die Primäre Aufgabe des Controllers ist die Initialisierung von Scope-Variablen, die in der View verwendet werden können.
- **Scope**
Das Scope Objekt stellt der View die dynamischen Daten zur Verfügung. Die Scope Instanz kann sogenannte Scope Variablen mit Werten oder Funktionen initialisieren.
- **Directives**
Direktive greifen auf den DOM Tree zu und erweitern den HTML Code. AngularJS bringt standardmässig schon einige mit. In der Regel fangen diese immer mit ,ng' an.
- **Factories/Services**
Services dienen dazu Funktionalitäten zu kapseln, die an mehreren Stellen der Applikation benötigt werden. Definiert werden diese über die factory Funktion.

6.1.1.1 Die App

Die AngularJS Applikation wird über das App Modul gestartet, bzw. das App Module wird von require.js geladen. Das folgende Sequenz Diagramm soll den Ablauf bis zum Start der Webapplikation aufzeigen. Wie bereits erwähnt, werden die require.config.js und require.js Dateien von der index.html gestartet und dienen vor allem für das Verwalten von internen und externen Abhängigkeiten. Darunter werden alle *Module (rot)* und *Konfigurationen (violett)* geladen. Anschliessend wird die Applikation gestartet und die einzelnen Controller, Services oder Direktive für die Seitenansicht geladen.

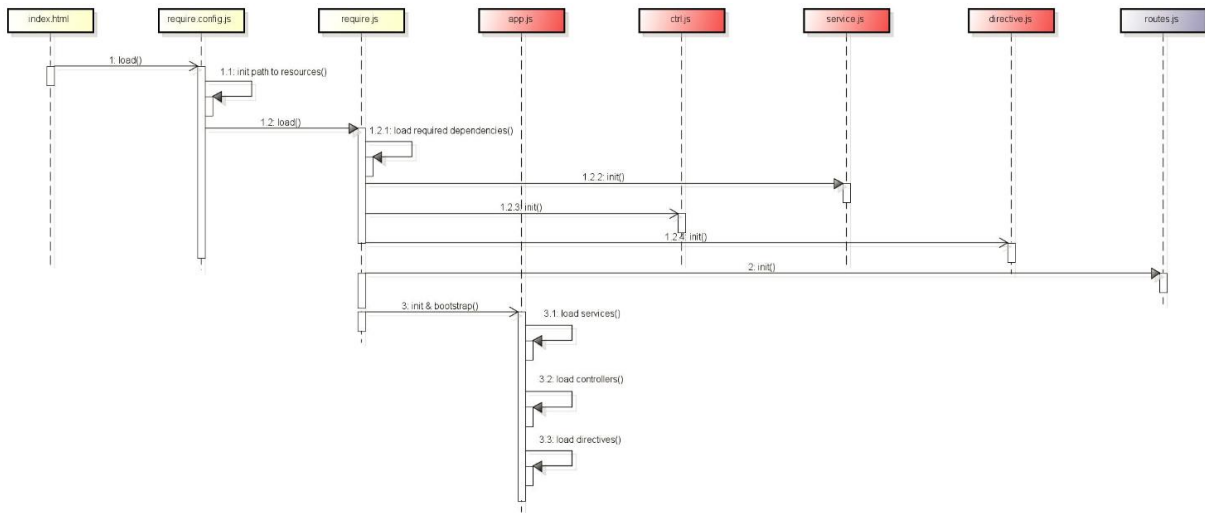


Abbildung 43 TeachEval Start Prozess

Es werden folgende Controller geladen:

Controller	Beschreibung
AdministrationController	Der <i>AdministrationController</i> ist mit der Benutzerverwaltung gekoppelt. Dieser beantragt für die Anzeige aller Benutzer (Leiter & Dozenten) beim <i>UserService</i> die nötigen Daten und stellt diese der View in Form von Scope Variablen/Funktionen zur Verfügung. Über den Administration Controller können die Daten innerhalb der Tabelle gefiltert werden.
AuthenticationController	Der <i>AuthenticationController</i> wird über die gesamte Web App angebunden und verwaltet das Anmelden und Abmelden eines Benutzers an der Applikation.
DashboardController	Für die Übersicht aller Auswertungen wird der <i>Dashboard Controller</i> verwendet.
ErrorController	Der <i>ErrorController</i> liefert die Daten zu den Fehlerseiten.
EvaluationReportController	Der <i>EvaluationReport Controller</i> holt die detaillierten Auswertung einer Umfrage vom Evaluation Service und stellt die Statistiken und Fragen der View zur Verfügung.
DatepickerController	Der <i>DatepickerController</i> ist ein Zusatz zum UI Bootstrap und stellt Funktionen und Daten für die Anzeige eines Datepicker zu Verfügung.
HandlerController	Über den <i>HandlerController</i> wird die Navigation innerhalb der einzelnen Seiten, bzw. über eine Link, Button, etc. verwaltet. Zudem behandelt der Handler Controller das Error Handling.
HomeController	Der <i>HomeController</i> ist zu der Startseite gekoppelt.
ImpressumController	Der <i>ImpressumController</i> ist zum Impressum gekoppelt.
SurveyInfoController	Über den <i>Survey Info Controller</i> wird die entsprechende Anzeige, ob ein Teilnehmer bereits an einer Umfrage teilgenommen hat, geliefert.
LoginController	Der <i>LoginController</i> ist zu der Login Seite gekoppelt.
ManageSurveyController	Der <i>Manage Survey Controller</i> ist mit der Umfrageverwaltung verbunden und liefert die Daten vom Survey Service für deren Verwaltung. Über diesen Controller können die Daten durch ein Suchfeld gefiltert werden.

ParticipateSurveyController	Der <i>Participate Survey Controller</i> ist für das Mapping der Umfrage zum eingegebenen Token zuständig. Dieser holt die Daten vom Survey Service und stellt diese der View zur Verfügung.
ProfileController	Im <i>Profile Controller</i> wird die Verwaltung des Benutzerprofils behandelt. Benutzereingaben werden über diesen Controller an den User Service übergeben.
SurveyController	Der <i>Survey Controller</i> ist mit der eigentlichen Umfrage gekoppelt und liefert die nötigen Daten aus dem Survey Service.
WrapperController	Der <i>Wrapper Controller</i> ist über die gesamte Applikation gültig und stellt die Verlinkung zu einzelnen HTML Templates zu Verfügung.

Tabelle 38 Controllers

Es werden folgende Services geladen:

Service	Beschreibung
AdminService	Im <i>AdminService</i> befindet sich die Logik für die Verwaltung der einzelnen Benutzerprofile.
AuthenticationService	Der <i>AuthenticationService</i> ist die clientseitige Schnittstelle zum Server. Dieser meldet einen Benutzer beim Server an oder ab.
EvaluationService	Über den <i>EvaluationService</i> werden die einzelnen Evaluationen vom Server beantragt.
SessionService	Der <i>SessionService</i> verwaltet den angemeldeten Benutzer und speichert diesen mit einem Cookie ab. Dadurch bleibt dieser bei einem Neustart oder einer Aktualisierung über die Taste <i>F5</i> des Webbrowsers angemeldet. Zudem wird der empfangene Login Token, für weitere Anfragen auf serverseitigen Ressourcen im Session Service abgespeichert.
SurveyService	Über den <i>SurveyService</i> werden Umfragen geladen und verwaltet. Ein CookieStore stellt sicher, das während einer Umfrage die ausgefüllten Antworten zwischengespeichert werden, damit ein Neustart oder eine Aktualisierung des Webbrowsers kein Verlust der bereits ausgefüllten Fragen mit sich bringt.
SystemService	Der <i>SystemService</i> liefert spezifisch Informationen über das Projekt, wie bspw. die aktuelle TeachEval Version oder das aktuelle Datum.
UserService	Über den <i>UserService</i> wird auf die User Ressourcen auf dem Server zugegriffen. Es können einzelne Anfragen nach Benutzer getätigt oder auch Aktualisierungen übermittelt werden.

Tabelle 39 Services

Es werden folgende Direktiven geladen:

Service	Beschreibung
EnterDirective	<p>Die EnterDirektive dient für die Benutzung der Enter Taste nach einer Eingabe in einem Input Feld. Direktive werden in der VIEW verwendet und beginnen normalerweise mit ng. Folgender Code zeigt den Einsatz der EnterDirective bzw. ng-enter in der View:</p> <pre><input id="userInputToken" type="text" class="form-control" ng-enter="userTokenInput()" placeholder="Geben Sie Ihre beliebige Kennung ein."/></pre> <p>Abbildung 44 ng-enter Direktive</p>

Tabelle 40 Direktive

6.1.2 Bootstrap 3

Für das Webdesign wird das Frontend Framework Twitter Bootstrap 3 eingesetzt. Mit Bootstrap lässt sich ein responsives Design für die unterschiedlichen Geräte wie PC, Tablet oder Smartphone entwickeln, welches auf einem eingebauten Grid-System basiert. Für die Gestaltung der bereits mitgelieferten Elemente wie Tabellen, Buttons, Modals, etc. wurde anhand des Corporate Designs der OdA G ZH das Design von TeachEval erstellt.

6.2 Backend

Das Backend ist die Java EE Applikation, die als WAR-File in Tomcat gehostet wird. Im WAR-File wurden zwei wesentliche Technologien verwendet: Zum einen die Referenzimplementation namens Jersey für die REST-Implementierung und zum anderen Hibernate als ORM-Framework für die Persistierung (OR-Mapping).

TeachEval wurde in drei Schichten unterteilt. In der obersten Schicht befinden sich alle Ressourcen, die für den Client über URIs ansprechbar sind.

Die Service Schicht repräsentiert die gesamte Business Logik, wie die Daten zu verstehen und zu interpretieren sind. Dieser leitet REST-Anfragen in weitere Subpackages für spezifischere Aufgaben weiter und greift direkt auf die Persistence Schicht zu, um die Verbindungen zur Datenbank zu eröffnen oder zu schliessen.

Die unterste Schicht ist die *Persistence Layer* und sie beinhaltet die Verbindung zur Datenbank, alle *Abfrage*- und alle *POJO* (Plain Old Java Object) – Klassen. In den Abfrage-Klassen wurden *prepared Statements* mit *Hibernate Query Language* (kurz HQL) geschrieben, um so CRUD-Operationen auszuführen. Die POJO-Klassen werden mit der Annotation *@Entity* auf der Klassenebene deklariert. Diese POJO Klassen werden auch als Entitäten bezeichnet, welche die gesamte Datenbank repräsentieren.

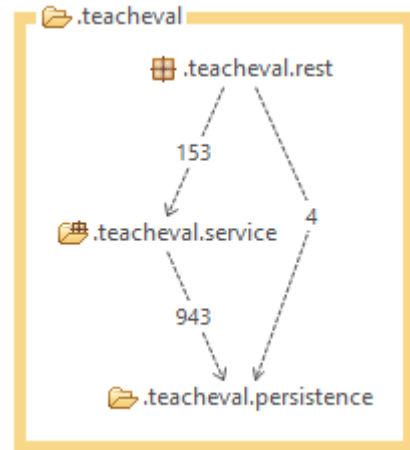


Abbildung 45
3-Schichten-Architektur

6.2.1 REST-Schicht

Die REST-Schicht besteht aus einem Package mit allen Ressourcen, die für die Webapplikation TeachEval benötigt werden.

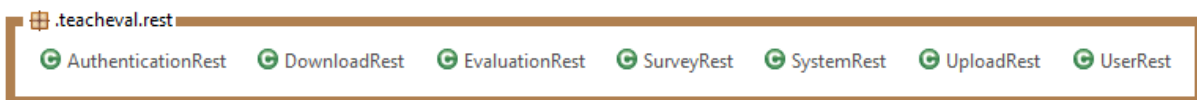


Abbildung 46 Struktur REST Package

Ressourcen	Funktion
AuthenticationRest	Im AuthenticationRest werden alle Zugriffe geregelt, für die eine Authentifizierung benötigt werden. Das heisst, das An- und Abmelden als auch die Überprüfung, ob das Session Token für den eingeloggten User noch gültig ist.
SurveyRest	Der SurveyRest stellt CRUD-Ressourcen und weitere benötigte Ressourcen zur Verfügung. Die meisten Ressourcen stellen ein Objekt <i>Survey</i> zur Verfügung.
UserRest	Der <i>UserRest</i> stellt ebenfalls CRUD-Ressourcen zur Verfügung und stellt die meisten Ressourcen als Objekt <i>Instructor</i> dar und paar wenige als Objekt <i>Participant</i> . Der Instructor ist der Benutzer, der ein Login-Account besitzt.
DownloadRest	Der DownloadRest übernimmt alle Download-Anfragen wie das Herunterladen eines Excel-Templates und das Exportieren von Daten als Excel.

UploadRest	Wie im DownloadRest werden im UploadRest nur Upload-Operationen abgearbeitet wie z.B. das Hochladen von Benutzern oder Umfragen als Excel Datei.
EvaluationRest	Im EvaluationRest geht es darum, die ausgefüllten Umfragen auszuwerten und diese zur Verfügung zu stellen.
SystemRest	Diese Ressource liefert dem Client die aktuelle TeachEval Version und das Datum.

Tabelle 41 REST-Klassen

6.2.2 Service Schicht

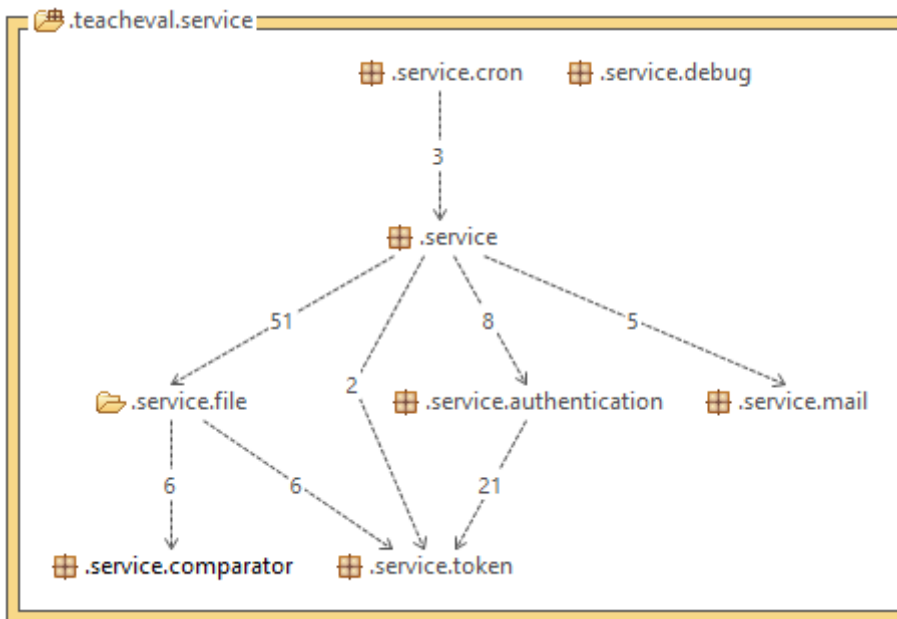


Abbildung 47 Struktur Service Schicht

Service	Funktion
service	Das <i>Service</i> Package ist der Kern aller Anfragen, die von der REST-Schicht kommen. Sie übernimmt weitere Aufgaben, wie das Weiterleiten oder das Session Handling.
service.authentication	Im Authentication Package wurden die Client-Sessions mittels Session Tokens gesetzt und an dem eingeloggten Benutzer für die Wiedererkennung zugewiesen. Zudem wurden hier die Berechtigungsstufen für den Zugriff auf die Ressourcen definiert.
service.token	Im <i>Token</i> Package wurden zwei verschiedene Tokens definiert. Zum einen das Session Token für den Login-Zugriff auf TeachEval und zum anderen ein Token für die Teilnahme an einer Umfrage.
service.mail	Die Umfrage Tokens für die Teilnehmer können über die Mail-Funktion versendet werden.
service.file	Dieses Package <i>File</i> beinhaltet alle Import- und Export Vorgänge.
service.comparator	Dieses Package ist eine Ansammlung an Comparators um Objekte richtig zu sortieren.
service.debug	Beinhaltet Debugging Tools wie die MemoryAnalyzis.

Tabelle 42 Service Packages

6.2.2.1 File Package

Für das Importieren und Exportieren von Daten in die Applikation oder in eine Excel-Datei, wurde eine Java Library namens *Apache POI* verwendet. Mit dieser Library ist es möglich, Microsoft Dateien in Java zu bearbeiten.

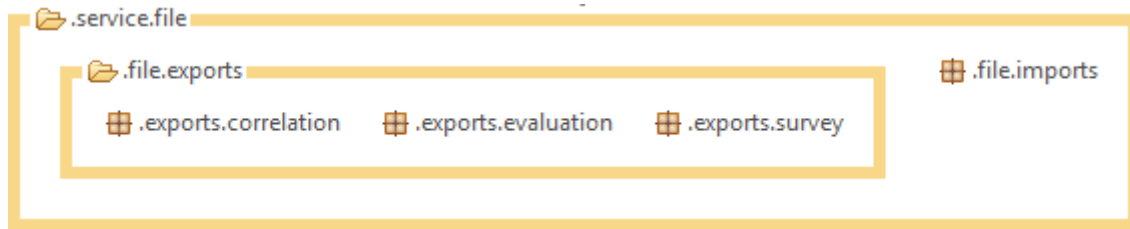


Abbildung 48 Struktur File Package

Package	Funktion
file.imports	Regelt alle Importvorgänge wie das Hinzufügen einer Umfrage oder der Instrukturen.
file.exports	Regelt alle Exportvorgänge wie das Exportieren von Korrelationen zweier Fragen, Auswertung einer Umfrage oder alle Daten für eine Umfrage.

6.2.3 Persistence Schicht

Die Persistence-Schicht agiert mit der Datenbank und führt CRUD-Operationen aus. Dabei erstellt er eine Verbindung zur Datenbank, führt eine Abfrage aus und bekommt als Resultat eine Entität zurück.

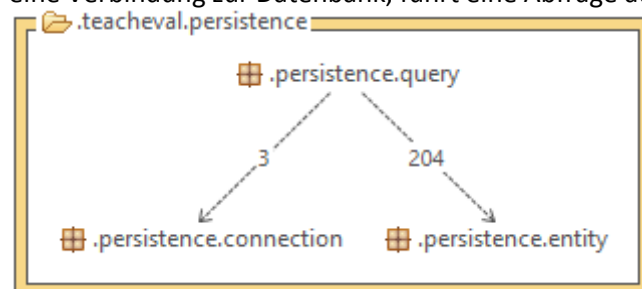


Abbildung 49 Struktur Persistence Schicht

Package	Funktion
persistence.entity	Im <i>persistence.entity</i> wurden alle POJO-Klassen definiert. Diese Klassen bilden die gesamte TeachEval Datenbank ab.
persistence.query	Die Service-Schicht greift direkt auf dieses Package zu um bestimmte Abfragen auszuführen. Dasselbe gilt auch für das Package <i>persistence.connection</i>
persistence.connection	Im <i>persistence.connection</i> wird die Verbindung zur Datenbank und mit ihren Einstellungen zur Verfügung gestellt. Um globalen Zugriff auf die Verbindung zu erhalten, wurde eine Helper-Klasse namens <i>HibernateUtil</i> erstellt. Diese Klasse erstellt einmalig ein globales <i>SessionFactory</i> Objekt, welches <i>thread-safe</i> ist, aber während dem Starten relativ viel Zeit benötigt. Dazu instanziiert es <i>single-threaded Session</i> -Objekte um alle Arbeitseinheiten schnell durchzuführen (JBoss Documentation Hibernate Session Factory, 2014).

6.2.4 Schnittstellen

In TeachEval wurden einige Schnittstellen verwendet um die Skalierbarkeit zu erhöhen und um duplizierten Code zu vermeiden.

6.2.4.1 AbstractQuery

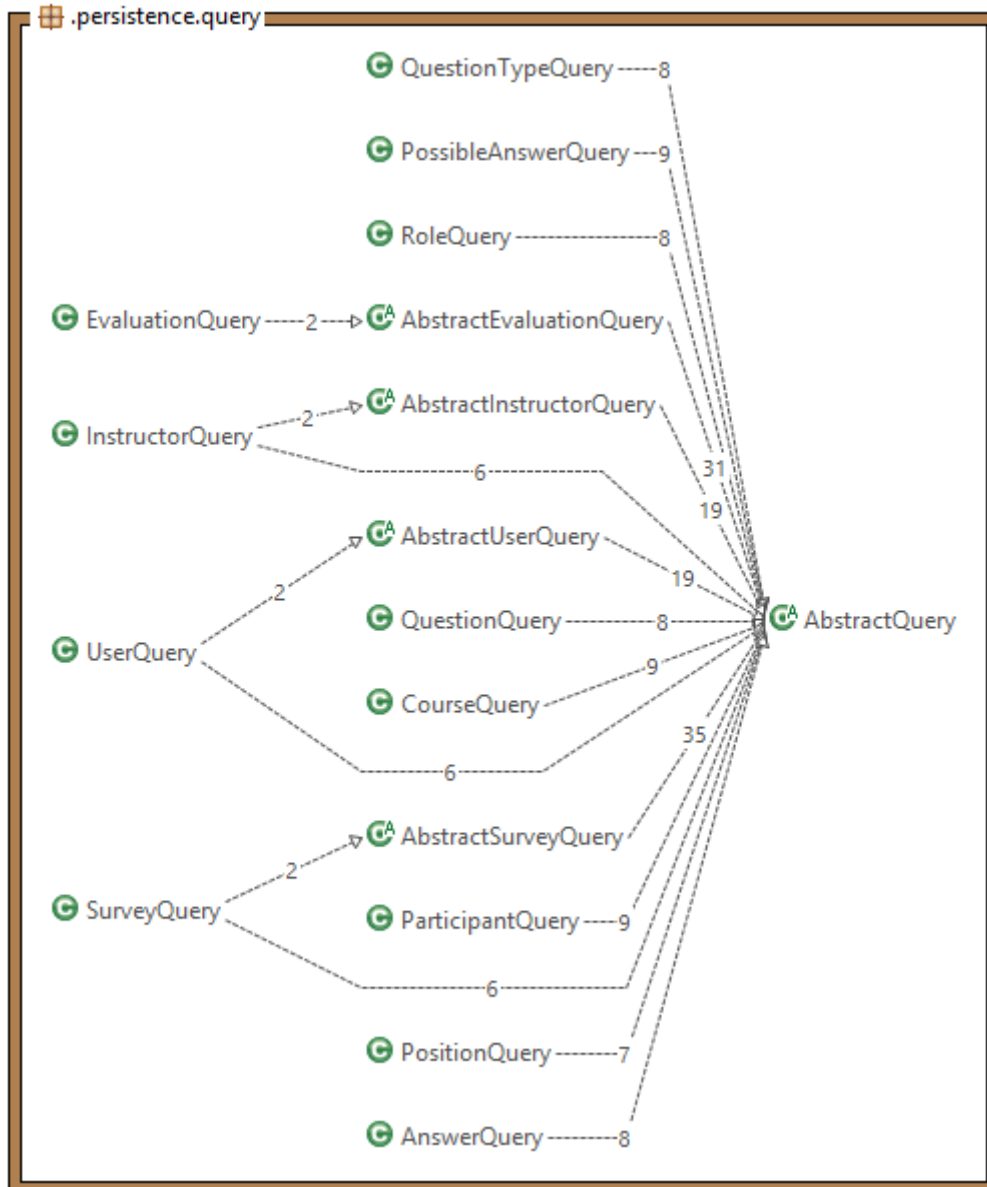


Abbildung 50 Die AbstractQuery mit Decorator

Die `AbstractQuery` ist die Basis Klasse aller Query-Klassen. Sie beinhaltet ein `Session` Objekt und stellt einige CRUD-Methoden für die Implementation zur Verfügung. Um noch spezifischere Abfragen erstellen zu können, wurden mehrere Decorators wie bspw. `AbstractInstructorQuery`, `AbstractSurveyQuery` oder `AbstractEvaluationQuery` erstellt.

6.2.4.2 Collector

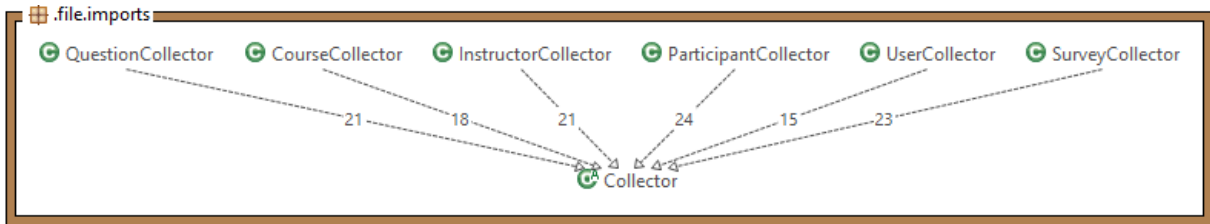


Abbildung 51 Der Collector mit Unterklassen

Der *Collector* wird während eines Importvorganges verwendet und sammelt alle Informationen in einer Excel-Datei. Der Collector ist eine abstrakte Klasse, die Felder und Methoden für die Unterklassen zur Verfügung stellt.

6.2.4.3 Token

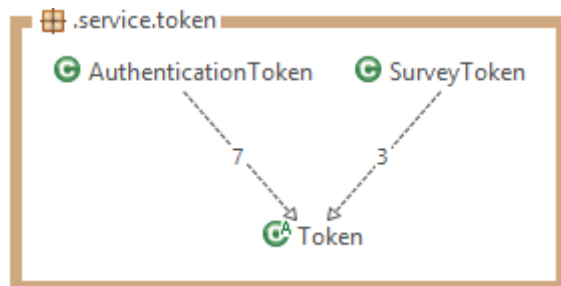


Abbildung 52 Token Varianten

Das Token Objekt ist ebenfalls eine abstrakte Klasse, die bereits Felder und Methoden für die verschiedenen Token-Varianten zur Verfügung stellt.

6.2.4.4 AbstractSheet

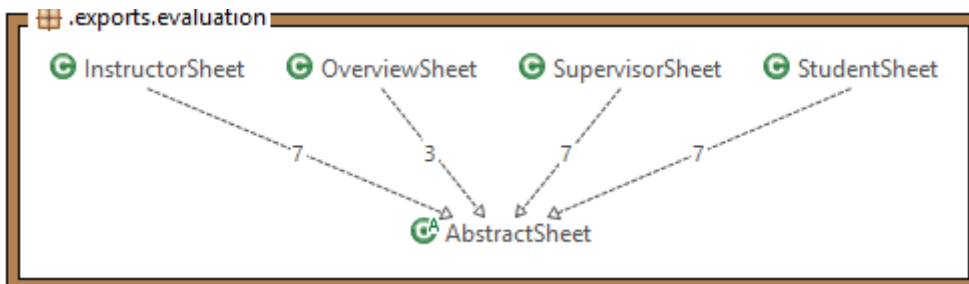


Abbildung 53 AbstractSheet für EvaluationExport

Das *AbstractSheet* wird für den Export der Evaluationen benötigt. Da die Daten in Student, Dozent und Supervisor unterteilt werden und diese je ein eigenes Arbeitsblatt im Excel erhalten, wurde für jedes Arbeitsblatt ein eigenes *Sheet*-Objekt erstellt. Das *AbstractSheet* gewährt den Subklassen Format-Styles und einige definierte Implementierungen, die für alle Subklassen verwendet werden können.

6.2.5 Token Authentication

Um einen angemeldeten Benutzer identifizieren zu können, muss der Server den Benutzer kennen. Dieser Benutzer muss sich mit seiner E-Mail und seinem Passwort anmelden. Das Passwort wird als SHA256 clientseitig gehashed und zum Server übertragen. Anschliessend erhält er ein Session Token, welches serverseitig und clientseitig gespeichert wird. Bei jedem Request des Clients sendet dieser sein Token im HTTP Header mit. Nachdem der Token erfolgreich vom Server überprüft worden ist, erhält der Client die angefragten Daten als JSON.

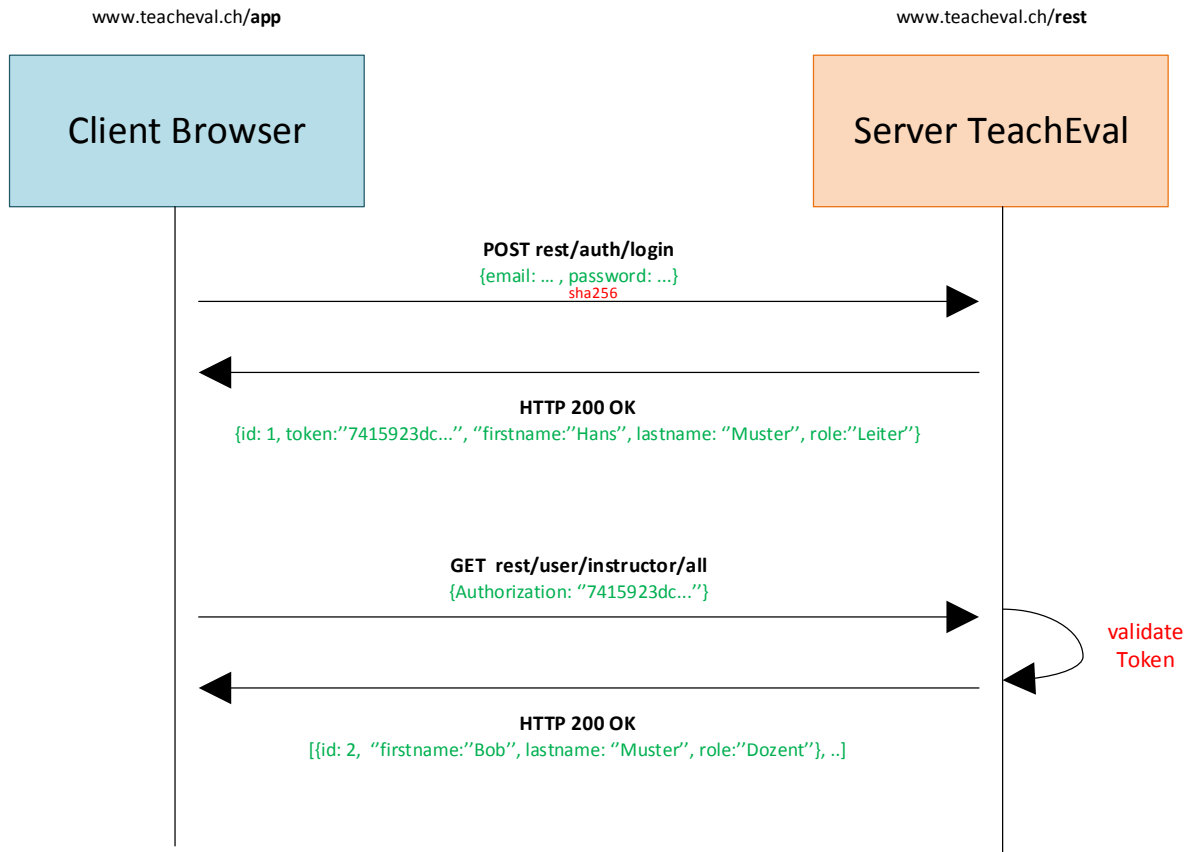


Abbildung 54 Token Based Authentication

Die serverseitige Validierung läuft mit einem *SecurityFilter*. Der SecurityFilter ist ein Servlet und wird im *web.xml* registriert.

```
<init-param>
  <param-name>com.sun.jersey.spi.container.ContainerRequestFilters</param-name>
  <param-value>com.sun.jersey.api.container.filter.LoggingFilter;ch.odag.teacheval.service.authentication.SecurityFilter</param-value>
</init-param>
```

Abbildung 55 Security Filter im web.xml

Jeder Request durchläuft, bevor er Zugriff auf eine Ressource erhält den SecurityFilter. Dieser überprüft ob der mitgelieferte Token gültig ist und gewährt, bzw. verhindert den Zugang zur Ressource. Der folgende Code demonstriert die Implementation des Security Filter.


```
public class SecurityFilter implements ContainerRequestFilter {
    @Override
    public ContainerRequest filter(ContainerRequest request) throws
    WebApplicationException {
        //Code
        return request;
    }
}
```

Tabelle 43 Security Filter als Code

Anschliessend kann mit folgenden Befehl der Token aus dem Header gelesen werden.

```
// Get the authentication-data passed in HTTP headers parameters
String stringToken = request.getHeaderValue("Authorization");
```

Abbildung 56 GetHeaderValue

Falls der Token nicht gültig sein sollte wird eine `WebApplicationException` geworfen, die automatisch eine HTTP Response mit dem Status Code `UNAUTHORIZED` an den Client schickt.

```
throw new WebApplicationException(Status.UNAUTHORIZED);
```

Abbildung 57 WebApplicationException Unauthorized

Um am Ende den Zugriff nach der Benutzerrolle zu unterscheiden, kommt eine `RoleAuthenticator` Klasse die vom `SecurityContext` abgeleitet wird zum Einsatz. Dieser `SecurityContext` kann anschliessend im REST folgendermassen verwendet werden. Wir verweisen hier, für den vollständigen Code auf unser JavaDoc und auf den Source Code.

```
@GET
@Path("/participant/all")
@Produces(MediaType.APPLICATION_JSON + ";charset=utf-8")
public Response getParticipants() {

    if (securityContext.isUserInRole(Roles.LEITER.toString())) {

        //Code

    }
    return Response.status(Status.UNAUTHORIZED).cacheControl(cc).build();
}
```

Abbildung 58 SecurityContext isUserInRole

6.2.6 Wichtige Abläufe

6.2.6.1 Login

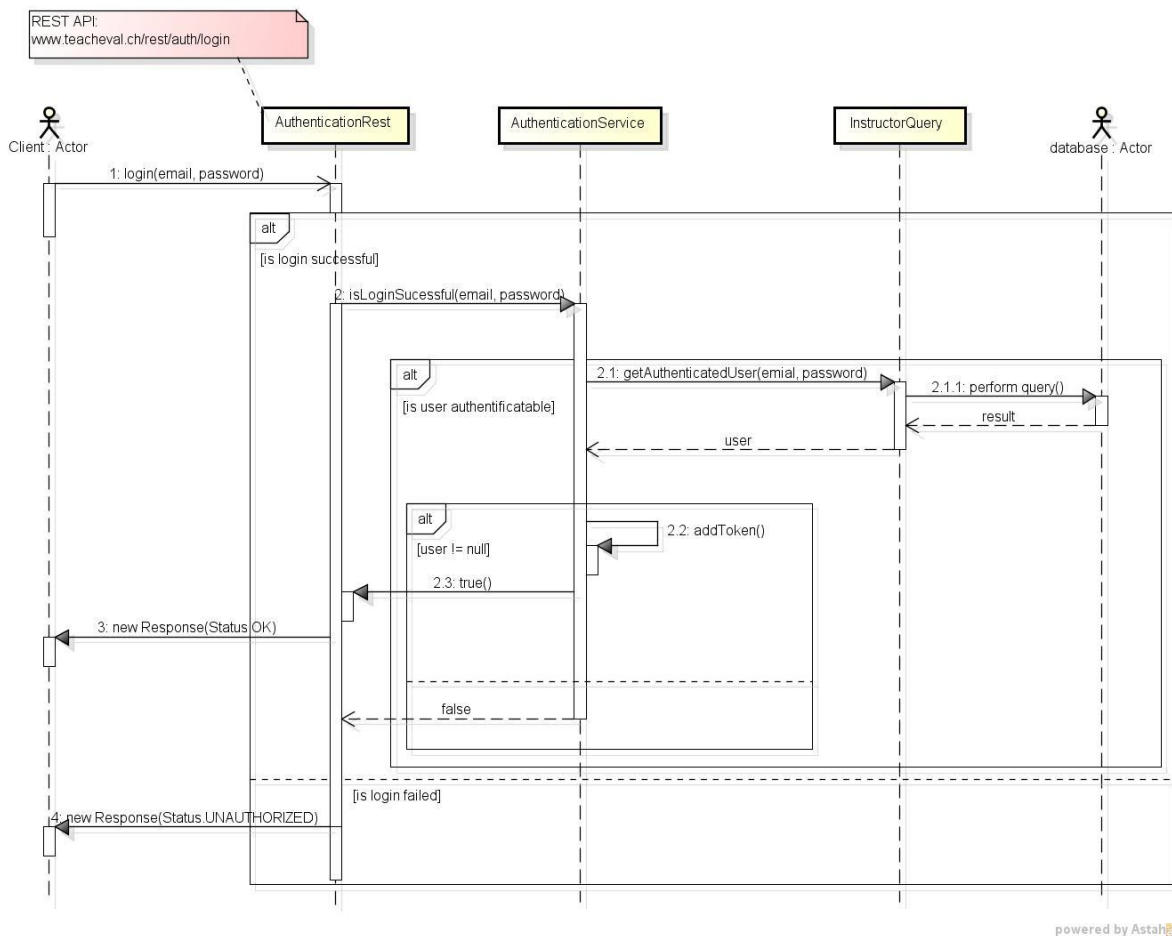


Abbildung 59 Ablauf Login

Um die kritischeren Funktionalitäten wie die Verwaltung von Benutzer und Umfragen zu schützen, ist es zwingend, sich an der Webapplikation anzumelden. Der Client meldet sich mit Email und Passwort an. Nach der Bestätigung übergibt der Client die Emailadresse und das Passwort als Hash an die Ressource `/rest/auth/login`. Die Klasse `AuthenticationRest` ist verantwortlich für die Ressource und übergibt die Informationen mittels der Methode `isLoginSuccessful` weiter an den `AuthenticationService`. Dieser erstellt eine Session zur Datenbank und ruft über die `InstructorQuery` die Methode `getAuthenticatedUser` auf, welche eine Abfrage in der Datenbank erstellt und die Emailadresse und Passwort abgleicht. Falls der Benutzer existiert, antwortet der Server mit dem Response Status `OK`, ansonsten mit Status `UNAUTHORIZED`.

6.2.6.2 Berechtigungsüberprüfung

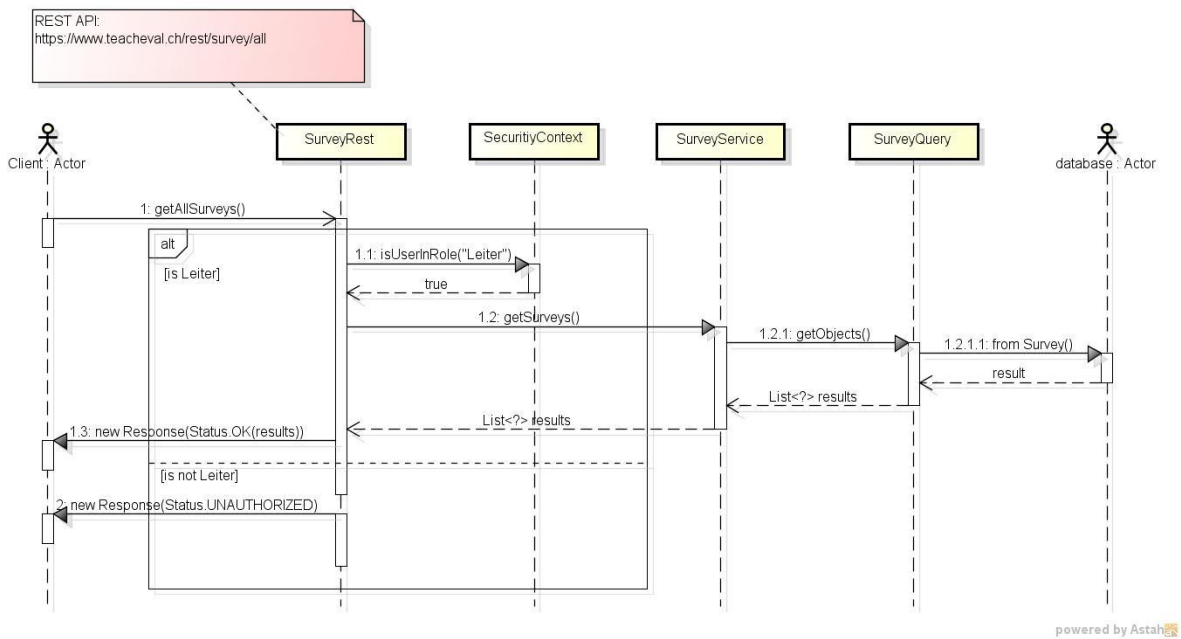


Abbildung 60 Rollenprüfung auf Ressourcen

Um unerlaubten Zugriff auf bestimmte Ressourcen zu vermeiden, werden in jede nicht öffentliche Ressourcen nach der Rolle überprüft. Diese Rollen sind die *Leiter* und *Dozenten*. Wenn der Client auf die Ressource */rest/survey/all* zugreifen möchte um eine Liste aller Umfragen zu bekommen, dann wird immer zuerst nach der Rolle überprüft. Das heisst, wenn der Client die Rolle als *Leiter* besitzt, dann wird als Nächstes die gewünschte Operation durchgeführt, sonst erhält der Client eine *UNAUTHORIZED* Meldung.

6.2.6.3 Umfrage importieren

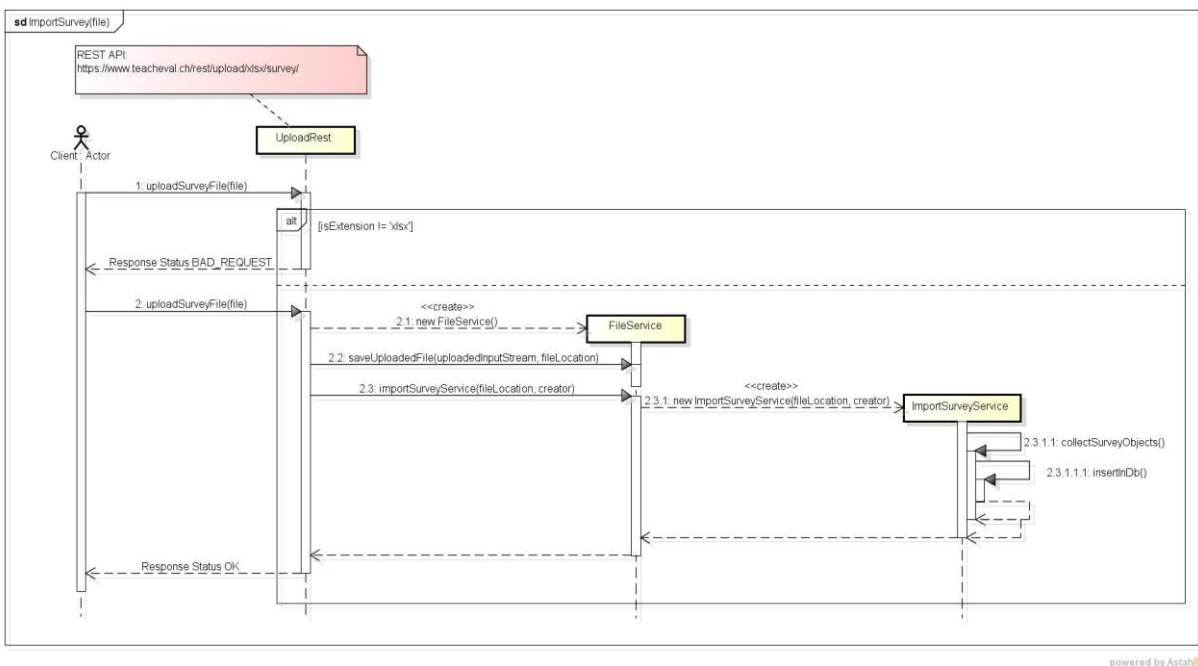


Abbildung 61 Ablauf Importieren einer Umfrage

Der Importvorgang besteht aus zwei Vorgängen. Der erste Vorgang ist das Hochladen der Datei und die Datei temporär auf dem Server abspeichern. Als Zweites werden die Informationen in der Datei

gesammelt und überprüft. Beim ersten Versuch hat der Client in Abbildung 61 Ablauf Importieren einer Umfrage eine Datei hochgeladen, bei der die Dateierweiterung nicht stimmt. Der Client erhält dann eine Fehlermeldung. Beim zweiten Versuch lädt der Client eine Datei anhand der Vorlage hoch. Der UploadRest überprüft wieder nach der Dateierweiterung und gibt bei Erfolg den Importvorgang weiter hinunter zum *FileService*. Dieser speichert als Erstes die Datei lokal ab. Als Nächstes werden die Informationen gesammelt und nach Korrektheit und Vollständigkeit geprüft. Falls alles in Ordnung ist, dann speichert *ImportSurveyService* die Informationen in die Datenbank und der Client erhält eine Erfolgsmeldung.

6.2.6.4 Umfrage exportieren

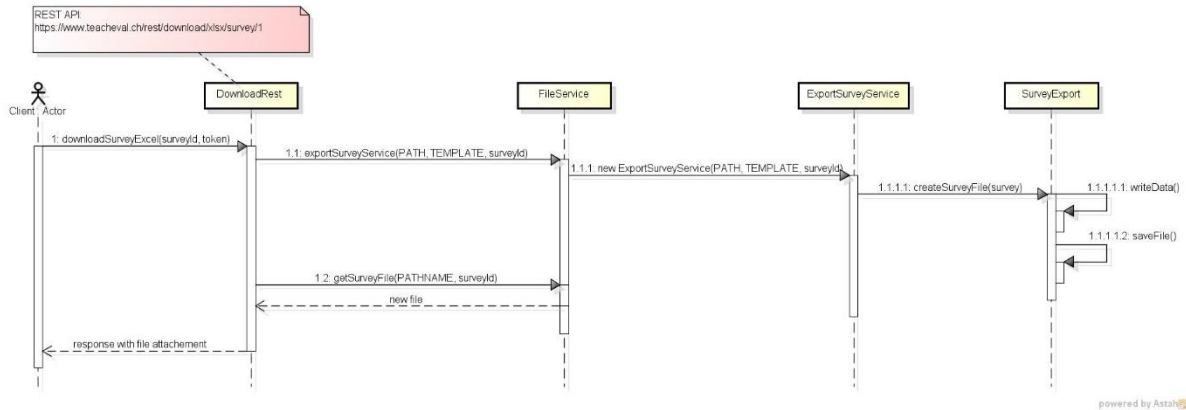


Abbildung 62 Ablauf Umfrage exportieren

Der Client möchte eine Umfrage exportieren, damit er/sie die Tokens für die Teilnehmer erhält. Über die Ressource `/rest/download/xlsx/survey/{id: [0-9]*}` kann der Client eine Umfrage herunterladen. Die ID kann eine beliebige ganze Zahl sein. Der DownloadRest nimmt die Anfrage über die URL entgegen und gibt die Anfrage an den *FileService*. Dieser erstellt ein neues *ExportSurveyService* Objekt, welches dann das Template einliest, die Umfragedaten in das richtige Arbeitsblatt schreibt und am Ende als neue Datei speichert. Der *FileService* holt die Datei vom *ExportSurveyService* und gibt die Datei an dem *DownloadRest* zurück. Dieser baut eine Response mit einem Attachment auf und gibt die Datei als Response an dem Client zurück.

6.2.7 Cache Control

Der Cache ist eine wichtige Komponente des Webbrowsers um bereits geladene Ressourcen, wie z.B. Bilder oder Text auf den Computer des Besuchers abzuspeichern. Sobald eine Ressource wieder benötigt wird, dann ist der Inhalt über den Cache schneller abrufbar. Es traten jedoch Cache Probleme bei Internet Explorer 8 bis 10 auf, die das automatische Aktualisieren der Tabellen, bspw. nach dem hochladen von Umfragen oder Benutzer verhindert. Mit einem serverseitigen Cache Control wird im Header der Parameter „no-cache“ mitgeschickt. Der Internet Explorer erkennt den Cache Controller Header und überspringt das Cachen der beantragten Ressource. Aus diesem Grund wurde bei GET Abfragen der *no-cache* Parameter mitgeschickt.

Der Cache Control wurde mit folgenden Code im REST initialisiert.

```

public UserRest() {
    this.userService = new UserService();

    cc = new CacheControl();
    cc.setNoCache(true);
}
  
```

Abbildung 63 Cache Controller initialisieren

Anschliessend kann der Cache Control mit dem *Response* Objekt am Client zurückgesendet werden.

```
return Response.status(Status.OK).entity(instructor).cacheControl(cc).build();
```

Abbildung 64 Cache Control im Response

7. Open Source Software

TeachEval muss als Anforderung mit Open Source Software (OSS) realisiert werden. Damit man sich einen Überblick über die eingesetzten Open Source Komponenten oder auch Fremdkomponenten verschaffen kann, werden diese in der folgenden Tabelle aufgelistet.

7.1 Frontend

Fremdkomponente *	Lizenz
angularjs	MIT license
bootstrap	MIT license
bootstrap UI	MIT license
font-awesome	MIT License
jquery	MIT license
metisMenu	MIT license
requirejs	new BSD or MIT licensed
domReady	Dual-licensed -- new BSD or MIT
passwordmeter	MIT license
sha256	BSD license
es5-shim	MIT license
html5shiv	MIT license
respond	MIT license
placeholder	MIT license
flaticon – Autor wurde im Impressum vermerkt	CC BY 3.0

Tabelle 44 Open Source Software Frontend

*In der `require.config.js` werden die einzelnen Fremdbibliotheken beschrieben.

7.2 Backend

Fremdkomponente	Lizenz
Tomcat	Apache License, Version 2.0_(Licence Apache Tomcat, 2014)
Jersey	Dual licensed – CDDL or GPL
Hibernate	Dual licensed: LGPL 2.1 or ASL 2.0
Apache POI	Apache License, Version 2.0 (Licence Apache POI, 2014)
Java Mail	CDDL or GPLv2+CE
Apache Math	Apache License, Version 2.0 (Licence Apache Math, 2014)
Quartz	Apache License, Version 2.0_(Licence Apache Quartz, 2014)
log4j	Apache License, Version 2.0 (Licence Apache Log4J, 2014)

Tabelle 45 Open Source Software Backend

8. Datenspeicherung

TeachEval verwendet für die Datenspeicherung die Open-Source Datenbank PostgreSQL 9.3. PostgreSQL ist ein objektrelationales Datenbankmanagementsystem (ORDBMS).

Das Datenbankschema von TeachEval sieht wie folgt aus:

Legende	
	Primärschlüssel
#	Spalte mit Primärschlüssel (Serial Datentyp)
	Spalte mit Fremdschlüssel
	Spalte, das auf ein Fremdschlüssel von einer anderen Tabelle zeigt
	Spalte mit Index
	Spalte mit Unique Index
	Kardinalität (1 zu n)
	Kardinalität (n zu 1)
	Kardinalität (n zu 1)
0 1	Boolean Wert

Tabelle 46 Legende DbSchema

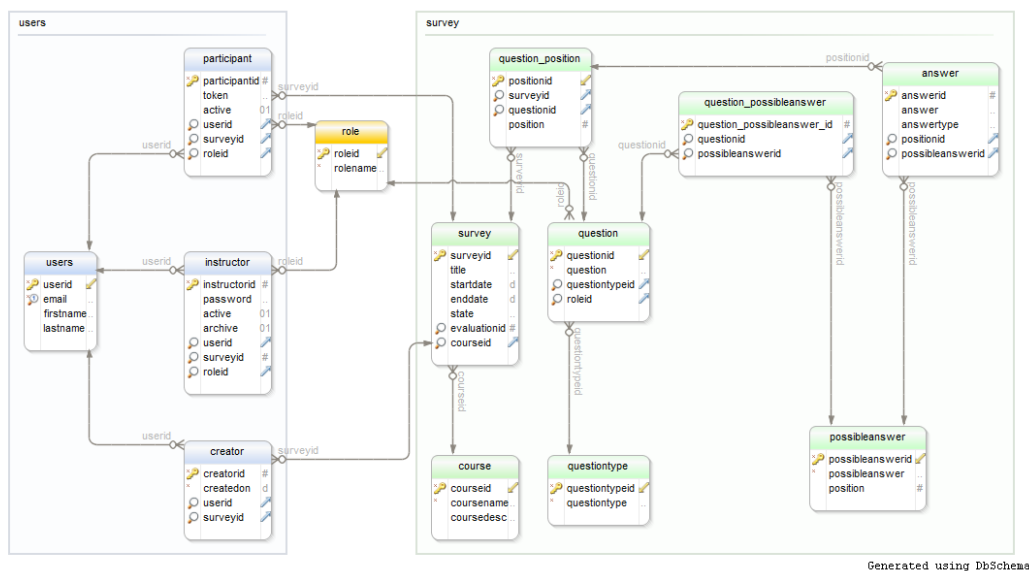


Abbildung 65 Datenbankschema TeachEval

Das Datenbankschema wurde mit DB Schema (DBSchema Offizielle Webseite, 2014) gezeichnet und lässt sich in die Bereiche *Users* und *Survey* unterteilen. Wobei im Bereich *Users* insbesondere die Benutzerverwaltung und im Bereich *Survey* die Umfrageverwaltung stattfindet.

8.1 User Bereich

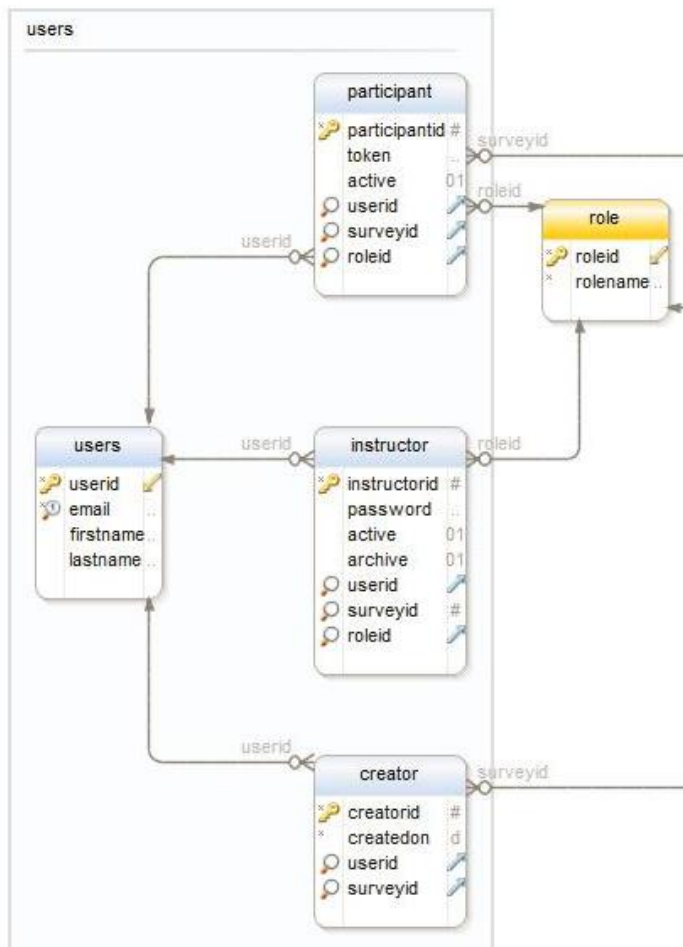


Abbildung 66 Bereich User

- **Users**
Jeder Benutzer in TeachEval wird in der Tabelle Users mit Vorname, Nachname und Email-Adresse erfasst.
 - **Participant**
Zusätzlich bekommt jeder Benutzer mindestens eine bestimmte Funktion zugewiesen. Wenn der Benutzer nur an einer Umfrage teilnehmen möchte, dann wird er als Teilnehmer (Tabelle Participant) erfasst und erhält ein Token. Um gewährleisten zu können, dass jeder Teilnehmer nur einmal an der Umfrage teilnehmen kann, wird das Feld `active` standardmässig nach der Teilnahme auf `false` gesetzt.
 - **Instructor**
Falls es sich aber um ein Benutzer handelt, der Login-Zugriff auf die Webapplikation benötigt, dann wird der Benutzer als Instruktor (Tabelle Instructor) erfasst. Der Instruktor hat entweder die Rolle als Dozent oder als Leiter. Der Dozent darf nur auf die Webapplikation zugreifen, um Umfragen über sein Kurs zu beenden und um auf die Auswertung zuzugreifen. Der Leiter ist im wesentlichen Sinne der Administrator, der die Benutzer verwaltet und Umfragen verwaltet.
- Ein Benutzer kann als Umfrageteilnehmer oder als Instruktor agieren. Das heisst, jeder Umfrageteilnehmer, Instruktor oder Ersteller einer Umfrage wird erfasst.

- Creator
Der Creator ist der Ersteller einer Umfrage. Der Ersteller ist immer ein Instruktor mit der Rolle als Leiter.
- Role
Jeder Benutzer kann eine Rolle haben. Ein Teilnehmer kann die Rolle Student, Dozent, Supervisor oder Leiter haben. Ein Instruktor nur Dozent oder Leiter.

8.2 Umfrage Bereich

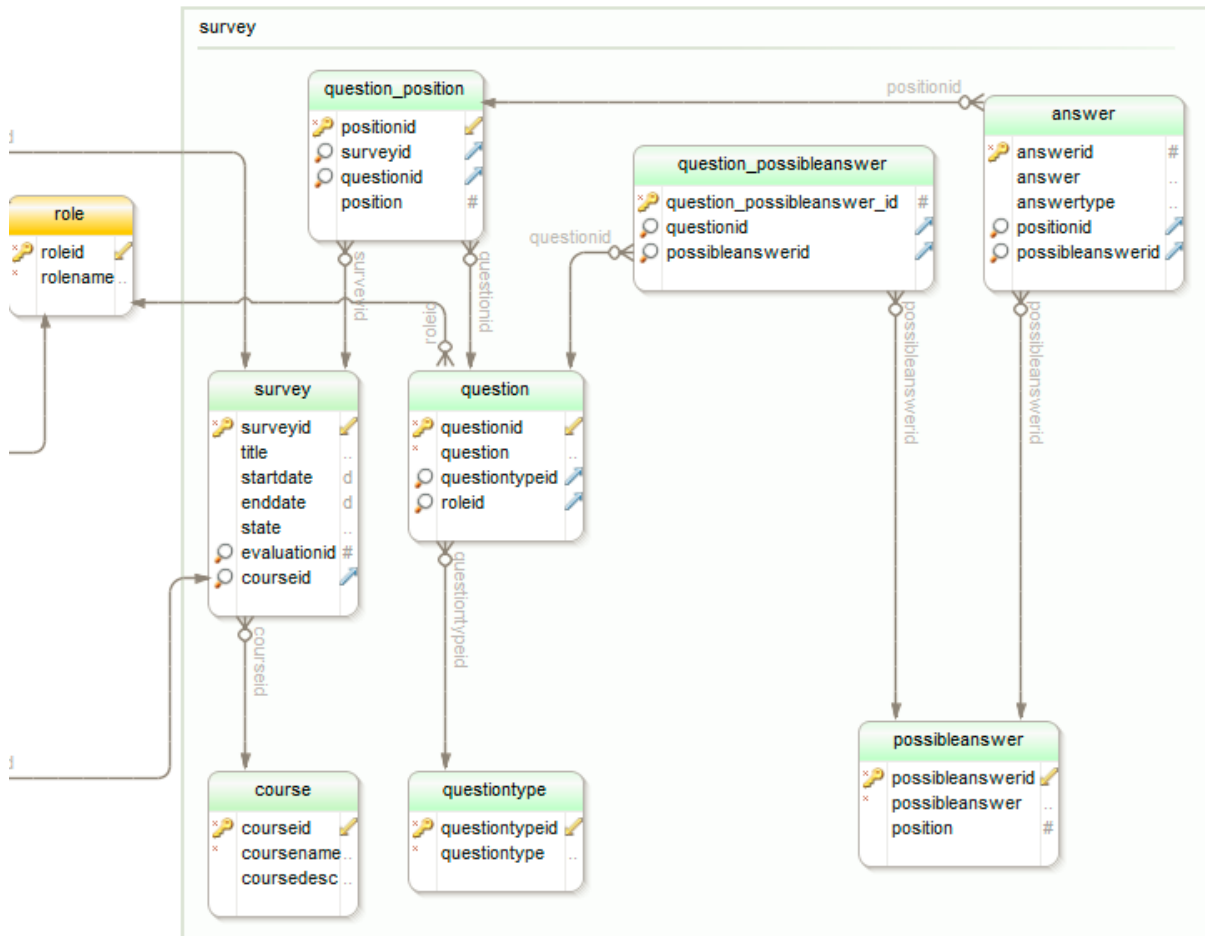


Abbildung 67 Bereich Umfrage

- Survey
Jede Umfrage besitzt ein Titel, Start- und Enddatum und der aktuelle Status.
- Course
Um Umfragen nach Kurse zu kategorisieren, wird für die Tabelle Course den Kursnamen und eine Beschreibung benötigt.
- Question
Für jede Umfrage können eine oder mehrere Fragen erstellt werden. Die Fragen können in verschiedenen Typen (Tabelle QuestionType) und in verschiedene Rollen (Tabelle Role) existieren.

- Question_Position
Question_Position ist eine Zwischentabelle und sie dient zur eindeutigen Bestimmung der Fragen zu einer Umfrage und der exakten Position (Reihenfolge) der Fragen (Tabelle Question).
- Questiontype
Eine Frage kann in eine der folgenden Fragetypen zugeordnet werden: RADIO, IST_SOLL_RADIO, TEXTBOX, CHECKBOX und CHECKBOX_TEXTBOX.
- Role
Jede Frage kann in eine der folgenden Rollen zugeordnet werden: Student, Dozent und Supervisor.
- PossibleAnswer
Zu einer Frage können mehrere Antwortmöglichkeiten existieren und eine Antwortmöglichkeit kann wiederum in mehreren Fragen vorkommen (siehe Tabelle Question_PossibleAnswer). Jede Antwortmöglichkeit besitzt ebenfalls eine Position zur Bestimmung der Reihenfolge.
- Answer
Die Antworten der Teilnehmer werden je nach Fragetyp in der Spalte Typ (IST, SOLL, RADIO, CHECKBOX oder TEXTBOX) gespeichert.

8.3 Löschooperationen

In der Datenbank werden laufend neue Daten hinzugefügt und bestehende Daten geändert. Löschooperationen werden selten genutzt. Wenn, dann nur beim Instruktor. Die Umfragen werden nicht gelöscht, sondern mit dem Feld *State* auf *Archiviert* gesetzt. So wird verhindert, dass die Daten bei einem versehentlichen Löschen unwiderruflich weg sind.

9. Abbildungsverzeichnis

Abbildung 36 Systemübersicht TeachEval.....	93
Abbildung 37 Projektstruktur TeachEval.....	96
Abbildung 38 Projektstruktur Client.....	97
Abbildung 39 Projektstruktur im Backend	98
Abbildung 40 Physische Architektur	100
Abbildung 41 Deployment Diagramm.....	101
Abbildung 42 AngularJS Architektur (Entwicklertagebuch AngularJS Structure, 2014).....	102
Abbildung 43 TeachEval Start Prozess.....	104
Abbildung 44 ng-enter Direktive	105
Abbildung 45 3-Schichten-Architektur	107
Abbildung 46 Struktur REST Package	107
Abbildung 47 Struktur Service Schicht	108
Abbildung 48 Struktur File Package.....	109
Abbildung 49 Struktur Persistence Schicht	109
Abbildung 50 Die AbstractQuery mit Decorator	110
Abbildung 51 Der Collector mit Unterklassen.....	111
Abbildung 52 Token Varianten.....	111
Abbildung 53 AbstractSheet für EvaluationExport.....	111
Abbildung 54 Token Based Authentication.....	112
Abbildung 55 Security Filter im web.xml.....	112
Abbildung 56 GetHeaderValue.....	113
Abbildung 57 WebApplicationException Unauthorized.....	113
Abbildung 58 SecurityContext isUserInRole.....	113
Abbildung 59 Ablauf Login	114
Abbildung 60 Rollenprüfung auf Ressourcen.....	115
Abbildung 61 Ablauf Importieren einer Umfrage	115
Abbildung 62 Ablauf Umfrage exportieren	116
Abbildung 63 Cache Controller initialisieren.....	116
Abbildung 64 Cache Control im Response	117
Abbildung 65 Datenbankschema TeachEval	119
Abbildung 66 Bereich User	120
Abbildung 67 Bereich Umfrage	121

10. Tabellenverzeichnis

Tabelle 37 Eingesetzte Libraries.....	99
Tabelle 38 Controllers	105
Tabelle 39 Services	105
Tabelle 40 Direktive.....	105
Tabelle 41 REST-Klassen	108
Tabelle 42 Service Packages	108
Tabelle 43 Security Filter als Code	113
Tabelle 44 Open Source Software Frontend	118
Tabelle 45 Open Source Software Backend	118
Tabelle 46 Legende DbSchema	119

TeachEval

Web App für die Unterrichtsevaluation

The screenshot displays the TeachEval web application interface. On the left, there is a survey form with fields for 'Frage:' (Question), 'Rolle:' (Role), and 'Umfrage:' (Survey). The survey title is 'Multimorbidität 2. Sem. FS2014'. The role is 'Student'. The question is '13 Ein Bezug zur beruflichen Praxis wurde hergestellt'. Below this, another survey entry is visible with the question '1 Der LTT Praxis-Tag war insgesamt unterstützend für die Entwicklung meiner beruflichen Praxis'. On the right, a table titled 'Korrelation' (Correlation) is shown. The table has two main sections: 'Übersicht' (Overview) and 'Werte' (Values). The 'Übersicht' section contains a table with columns 'Übersicht' and 'So war es'. The 'Werte' section contains a table with columns 'Werte' and 'Interpretation Korrelationskoeffizient'. The 'Werte' table shows the following data:

Werte	Interpretation Korrelationskoeffizient
Korrelationskoeffizient (R)	0.90508776 R
Bestimmtheitsmass (R ²)	0.819183853 R ²
Mittelwert X (xm)	0.9375 xm
Mittelwert Y (ym)	1.1875 ym

The 'Interpretation Korrelationskoeffizient' table shows the following data:

Interpretation Korrelationskoeffizient	Interpretation
R = -1	oder -100%
R = 0	oder 0%
R = 1	oder 100%

Software Qualitätsmanagement (SQM)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: ODA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Zweck

Dieses Dokument gilt als Grundlage für die einzelnen Testdurchführungen und zeigt auf, welche Arten von Tests ausgeführt wurden.

2. Einführung

Für die Webapplikation TeachEval wurden Systemtests, Kompatibilitätstests, Usability Tests und Unit Tests gemacht um den Einsatz der Use Cases zu überprüfen und deren Qualität zu messen. Diese Tests erstrecken sich vom Backend bis zum Frontend. Im Backend wurde vor allem mit JUnit4 und dem Jersey Test Framework getestet, damit die REST Schnittstellen auch die Daten liefern die verlangt werden. Im Frontend wurden eher Usability Tests und Kompatibilitätstests durchgeführt.

3. Systemtestspezifikation

3.1 Systemtest

Jede Funktion muss nach seiner Funktionsweise getestet und bewertet werden. Dabei wird bewertet, ob die Funktion die definierte Anforderung erfüllt.

Die folgende Tabelle zeigt alle Systemtests als Übersicht auf:

Use Case Titel	Priorität	Implementiert	Fehler/Unschönheiten	Status
An- / Abmelden	1	Ja		✓
Umfrage CRUD	1	Ja	kein nachträgliches Bearbeiten einer Umfrage möglich*	(✓)
Token generieren	1	Ja		✓
Token exportieren	1	Ja		✓
Token per Mail versenden	2	Ja		✓
Excel Template importieren	1	Ja		✓
Auswertungen ansehen	1	Ja		✓
Korrelation von 2 Fragen	1	Ja		✓
Export als Excel oder PDF	1	Ja		✓
Umfrage teilnehmen	1	Ja		✓
Benutzer verwalten	1	Ja		✓

*einfaches Bearbeiten der Umfrage möglich und alle anderen Operationen wie Umfrage hinzufügen, ansehen oder löschen werden unterstützt. Bearbeiten einer Umfrage mit Fragen und Teilnehmer usw. konnte jedoch aus Zeitgründen nicht realisiert werden.

Die implementierten Use Cases wurden mit der Webapplikation getestet und werden im Folgenden tabellarisch dargestellt:

3.1.1 An- / Abmelden

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Login:		
- Formular und Validierung		✓
- Passwort-Hash Vergleich		✓
- Session Token wird erstellt		✓
Logout:		
- Session Token wird gelöscht		✓

3.1.2 Umfrage CRUD

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Hinzufügen (siehe UC Excel Template importieren)		
Bearbeiten	Bis jetzt können Umfrage gestartet und beendet werden. Nachträgliches Bearbeiten von Teilnehmer und Fragen könnte eine wünschenswerte Erweiterung sein.	(✓)
Löschen	Wird in der Datenbank auf ARCHIVIERT gesetzt	✓
Ansehen (siehe UC Export als Excel)		✓

3.1.3 Token generieren

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Benutzer mit Email und Passwort aus der DB wird gefunden		✓
Session Token mit 14 Zeichen wird zufällig generieren		✓
Session Tokens wird aufbewahrt	Beim Neustart von Apache Tomcat sind alle Tokens weg. Benutzer muss sich neu einloggen.	✓

3.1.4 Token exportieren

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Umfrage Token mit 14 Zeichen zufällig generieren		✓
Umfrage Token als Excel exportieren		✓

3.1.5 Token per Mail versenden

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
E-Mail wird erfasst		✓
Teilnehmer mit Token in Mail wird hinzugefügt		✓
E-Mail wird versandt	Ein Teilnehmer mit ungültiger E-Mail (syntaktisch jedoch korrekt) kann zu einer Fehlermeldung führen. Die restlichen Teilnehmer werden dann nicht mehr benachrichtigt.	✓

3.1.6 Excel Template importieren

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Excel wird hochgeladen		✓
Informationen werden gesammelt und überprüft		✓
Umfrage wird in der DB erstellt		✓

3.1.7 Auswertungen ansehen

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Informationen werden aus der DB gesammelt		✓
Informationen werden als Diagramme angezeigt		✓

3.1.8 Korrelation mit 2 Fragen

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Auswählen von zwei Fragen über verschiedene Umfragen		✓
Beide Fragen werden exportiert		✓
Fragen werden im Excel korreliert		✓

3.1.9 Export als Excel oder PDF

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Umfrage exportieren		✓
Auswertung exportieren		✓
- mit Rohdaten		✓
- mit Diagrammen		✓
Korrelation exportieren		✓

3.1.10 Umfrage teilnehmen

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Umfrage Token überprüfen		✓
Umfrage teilnehmen		✓
Fragen validieren		✓
Antworten speichern		✓

3.1.11 Benutzer CRUD

Subtasks	Fehler/Unschönheiten/Verbesserungen	Status
Hinzufügen (siehe UC Excel Template Import)	Hinzufügen nicht nur als Excel, sondern auch mit Formulardaten	✓
Bearbeiten		
- Email bearbeiten		✓
- Passwort zurücksetzen		✓
- inaktiv / aktiv setzen		✓
Löschen		✓

3.2 Unit Testing

Unit Test kam vor allem serverseitig mit JUnit und dem Java Jersey Framework zum Einsatz. Aus diesem Grund wurde über alle Schichten getestet, die in der folgenden Abbildung visualisiert werden. Die Schichten *Persistence* und *Service* können als TestSuite Bundle ausgeführt werden und die *REST*-Schicht wird zusätzlich mit Jersey Test Framework mit der parallelen Ausführung von Tomcat getestet (Documentation Jersey Test Framework, 2014).

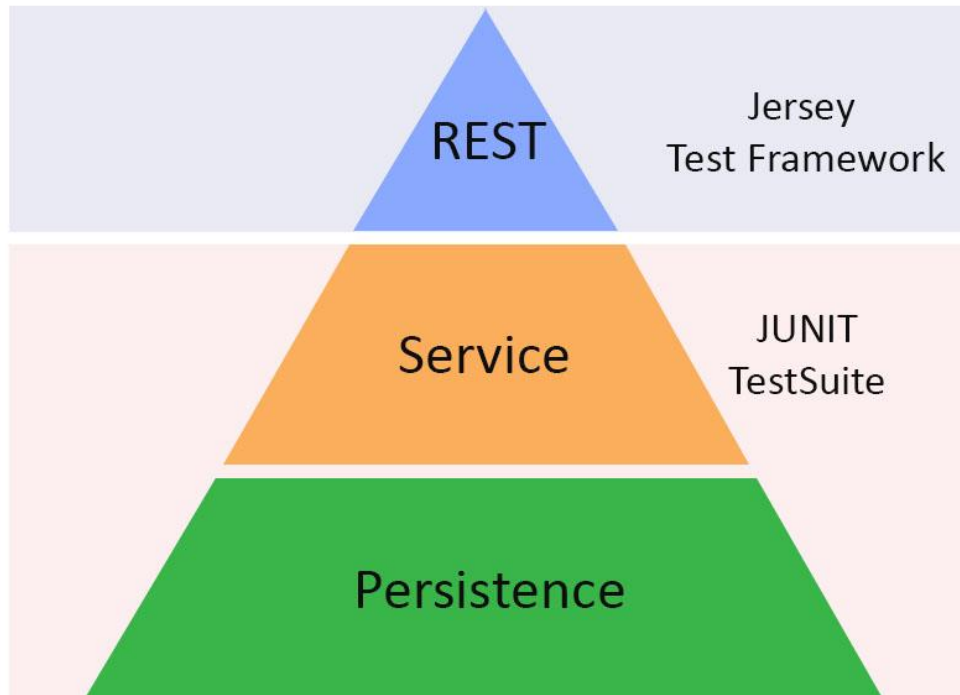


Abbildung 68 Test Frameworks

Das Jersey Test Framework eignet sich sehr gut um Clients zu simulieren, die dann auf die REST Schnittstellen von TeachEval Tests durchführen. Alle Tests wurden mit vorbereiteten Testdaten getestet. Dafür wurden 6 Umfragen als Basis für alle Tests durchgeführt.

3.2.1 JUnit TestSuite

Der grösste Teil der Unit Tests wurden mit JUnit 4 mittels TestSuite getestet. Der TestSuite erstreckt sich auf die *Persistence* und auf die *Service* Schicht. Da wir schon im Mai 2014 den ersten produktiven Prototyp aufschalten mussten, der vor allem die Umfrageverwaltung und die ganze Befragung umfasste, setzten wir den Fokus auf eine saubere und funktionstüchtige Ablegung der Daten in der Datenbank.

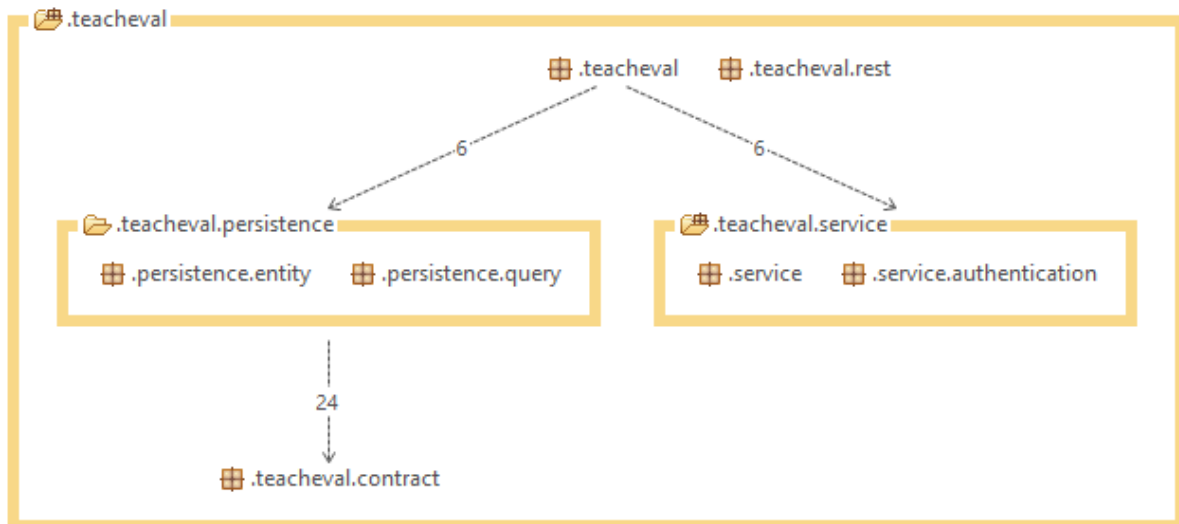


Abbildung 69 Package Struktur im Test Verzeichnis

Dabei wurden in der Persistence Schicht die folgenden Packages getestet:

- persistence.entity
Im *persistence.entity* wurde das Speichern, Löschen, Bearbeiten und Abfragen von Entitäten mit Hibernate getestet. Dieses Package dient hauptsächlich dazu, den korrekten Einsatz von Hibernate als Lazy Loading sicherzustellen.
- persistence.query
Im Package *persistence.query* wurden alle CRUD Abfragen auf die Datenbank definiert, die für TeachEval benötigt werden.

Im Service Schicht wurden folgende Services getestet:

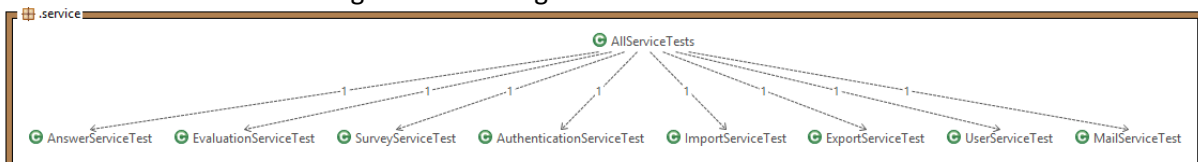


Abbildung 70 Service Schicht

Service	Use Case	Bemerkung
Answer Service	Umfrage teilnehmen	Speichern der Antworten.
Authentication Service	An- / Abmelden	In diesem Service wird die Authentifizierung eines Benutzers getestet.
Evaluation Service	Auswertungen ansehen	Die wichtigsten Methoden wurden getestet.
Export Correlation Service	Korrelation mit 2 Fragen	Export mit allen Fragetypen-Variationen.
Export Evaluation Service	Export als Excel	Export einer Auswertung.
Export Survey Service	Export als Excel	Export einer Umfrage.
Import Survey Service	Umfrage CRUD („Create“-Teil)	Importieren/Hinzufügen einer Umfrage.
Import User Service	Benutzer CRUD („Create“-Teil)	Importieren/Hinzufügen eines oder mehreren Benutzers.

Service	Use Case	Bemerkung
Mail Service	Token per Mail versenden	Testen von Emails mit Token mittels einem Dummy Mail Server.
Survey Service	Umfrage CRUD	Testen von allen Survey CRUD-Methoden.
User Service	Benutzer CRUD	Testen von allen User CRUD-Methoden.

Tabelle 47 getestete Services

3.2.2 Test Coverage

Zu Beginn des Projektes setzten wir uns keinen mindestprozentualen Wert für die Testüberdeckung. Wir testen vor allem kritische Bereiche, wie bspw. die sorgfältige Ablegung der Daten in die Datenbank. Zudem planten wir aufgrund des frühen Prototyps von Beginn weg Reservezeit für die Tests und Refactorings. Dennoch wollten wir am Schluss des Projektes die Qualität steigern und eine Mindestabdeckung von 60% erreichen. Der Hauptgrund für diesen niedrigen Wert liegt darin, dass wir uns hauptsächlich auf *Anweisungsüberdeckung* und nicht auf *Zweig-* oder *Pfadüberdeckung* konzentrierten.

4. Systemtestprotokoll

4.1 Angaben zur Durchführung

Für die Durchführung der Tests muss die Datenbank *teachevalapp* mit Schema (siehe Installationsdokumentation) bereits installiert sein. Eine korrekte Verbindung der Datenbank mit Hibernate wird vorausgesetzt. Um alle Tests in den Schichten Persistence und Service zu testen, muss der Test Runner *AllTests.java* ausgeführt werden. Er befindet sich im Source Folder *src/test/java* unter dem Package *ch.odag.teacheval* und er lässt sich mit *Rechtsklick > Run As > JUnit Test* ausführen (siehe Abbildung 71).

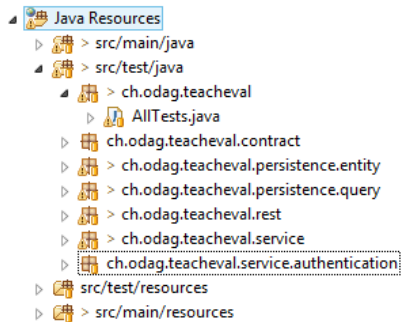


Abbildung 71 Test Runner *AllTests.java*

4.2 Protokoll

4.2.1 Stand 21.04.2014

Use Case Titel	Priorität	Implementiert	Fehler/Unschönheiten	Status
An- / Abmelden	1	Ja		✓
Umfrage CRUD	1	Ja	kein nachträgliches Bearbeiten einer Umfrage möglich*	✓
Token generieren	1	Ja		✓
Token exportieren	1	Ja		✓
Token per Mail versenden	2	Nein	In Entwicklung	✗
Excel Template importieren	1	Ja		✓
Auswertungen ansehen	1	Ja		✓
Korrelation von 2 Fragen	1	Nein	Funktion geplant	✗
Export als Excel oder PDF	1	Nein	In Entwicklung: Evaluation Export	✓
Umfrage teilnehmen	1	Ja		✓
Benutzer verwalten	1	Ja		✓

Abbildung 72 Stand April

*einfaches Bearbeiten der Umfrage möglich und alle anderen Operationen wie Umfrage hinzufügen, ansehen oder löschen werden unterstützt. Bearbeiten einer Umfrage mit Fragen und Teilnehmer usw. jedoch nicht.

Zum Zeitpunkt 21. April 2014 wurde TeachEval ausgiebig getestet und diverse Bugfixes unterzogen. In Entwicklung waren hauptsächlich die E-Mail Funktion für das Versenden von Tokens an die Teilnehmer/innen und die Funktion für das Exportieren der Auswertung.

4.2.1.1 Durchgeführte Unit Tests

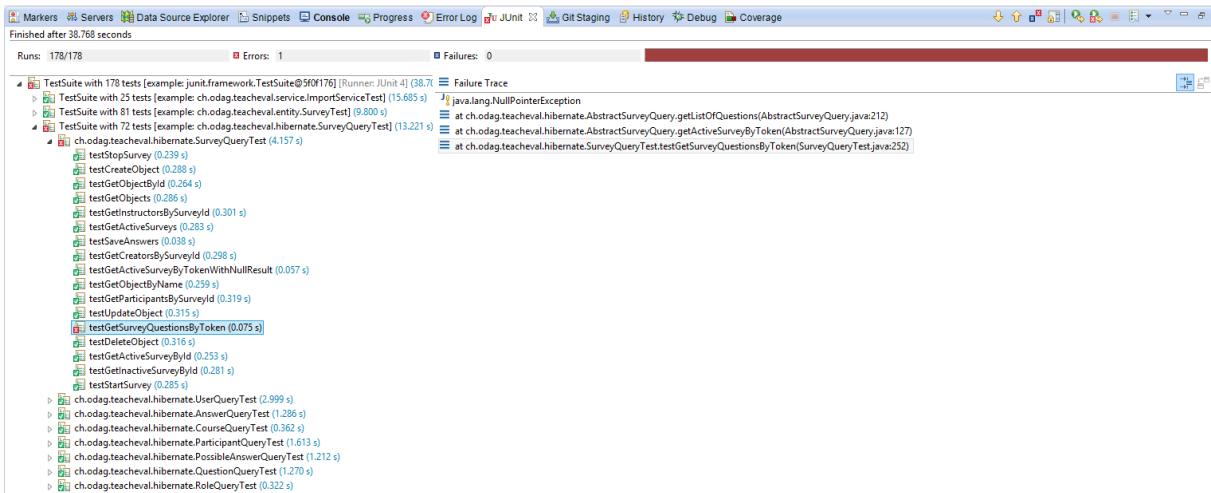


Abbildung 73 Alle Tests bis Service Schicht (21.04.2014)

Insgesamt wurden 178 Tests erstellt und ausgeführt, wobei ein Test fehlschlug, nämlich `testGetSurveyQuestionsByToken()`. In der Praxis schickt der Teilnehmer dem Server ein Umfrage-Token. Wenn dieser Token für eine Umfrage gültig ist, dann kann der Teilnehmer an der Umfrage teilnehmen. Ansonsten wird eine `NullPointerException` zurückgeworfen bzw. am Client eine Fehlermeldung angezeigt, dass die Umfrage nicht gefunden wurde oder der Token falsch sei.

Zudem wurde zu diesem Zeitpunkt im `testGetSurveyQuestionsByToken()` statisch ein Token String gesetzt, statt dynamisch aus der Datenbank ein bestehendes Token zu holen. Deshalb schlug dieser Test fehl.

4.2.1.2 Erreichte Test Coverage

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
TeachEvalApp	60.1 %	9,695	6,432	16,127
src/main/java	53.9 %	5,166	4,416	9,582
ch.odag.teacheval.authentication	0.0 %	0	416	416
ch.odag.teacheval.entity	68.4 %	1,288	595	1,883
ch.odag.teacheval.fileexport	100.0 %	296	0	296
ch.odag.teacheval.fileimport	68.3 %	1,746	809	2,555
ch.odag.teacheval.hibernate	61.3 %	852	537	1,389
ch.odag.teacheval.rest	0.0 %	0	1,032	1,032
ch.odag.teacheval.service	50.1 %	940	936	1,876
ch.odag.teacheval.token	32.6 %	44	91	135
src/test/java	69.2 %	4,529	2,013	6,542
src/test/resources	0.0 %	0	3	3

Abbildung 74: Test Coverage Stand 28.04.13

Insgesamt konnten wir das Minimum von 60% Test Coverage erreichen.

4.2.1.3 Metriken

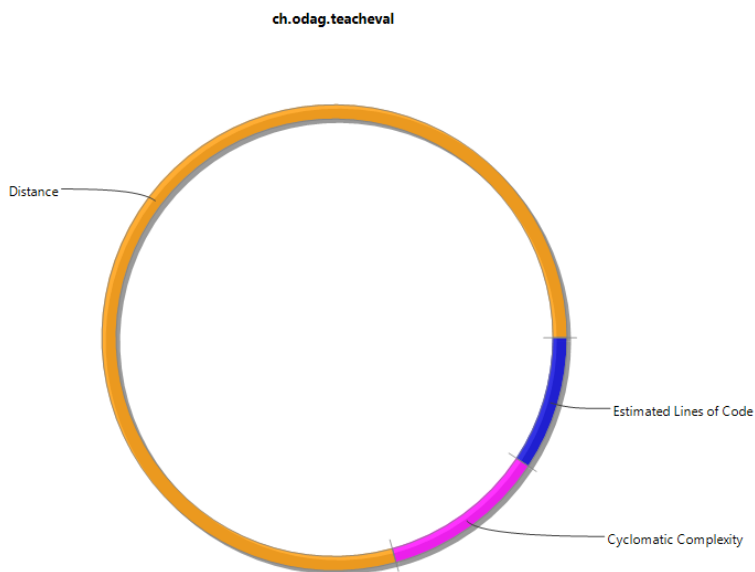


Abbildung 75: Pollution (Stand 21.04.2014)

Metrics	Value
Units	59
Classes / Class	0.02
Methods / Class	9.28
Fields / Class	2.26
ELOC (Estimated Lines of Code)	4097
ELOC / Unit	69.44
CC (Cyclomatic Complexity)	1.57
Fat	19
Tangled	0.00%

Tabelle 48 Metriken

4.2.1.4 Violations

Package	Metric	Value
entity	Distance	-1
fileimport	CC	29
	ELOC	85

Tabelle 49 Violation

Package fileimport

Der hohe *Cyclomatic Complexity* Wert entstand dadurch, dass beim Importieren einer Umfrage mit Excel in jeder Spalte überprüft werden musste, ob die in der Zelle enthaltene Information richtig gesetzt wurde.

Dabei kann der Zugriff auf eine Zelle vier unterschiedliche Stadien haben.

- Gültige Zelle enthält gültigen Wert
- Gültige Zelle enthält leeren Wert (bspw. bei Zellenformatierungen)
- Ungültige Zelle liefert NULL
- Ungültige Zelle wirft NullPointerException

Leider kann nicht vorhergesagt werden, in welcher Situation z.B. die Zelle NULL oder eine Exception zurückwirft. Um die Zelle bestmöglich zu überprüfen, wurde eine Switch-Case Anweisung verwendet (siehe Abbildung 76 CC durch Switch Anweisung).

```
switch (cell.getColumnIndex()) {
case 0:
    throwInvalidCellTypeException(cell,
        "Bitte geben Sie eine Frage ein! "
        + " \nFehlerquelle: \nExcel-Arbeitsmappe:\t"
        + "\"\" + sheet.getSheetName() + "\"\"
        + " auf Zeile:\t"
        + "\"\" + cell.getRowIndex() + "\"");

    position.getQuestion().setQuestion(cell.getStringCellValue());
    break;
case 1:
```

Abbildung 76 CC durch Switch Anweisung

Ein Problemfall besteht in der Excel-Arbeitsmappe 3) *Fragen*. Darin befinden sich mehr als 20 Spalten, die mit der Klasse *QuestionCollector* gesammelt und mit Switch-Case überprüft werden. Dies bedeutet, dass mindestens 20 Case Anweisungen für jede Spaltenüberprüfung verwendet wurden, was schliesslich ein sehr hoher ELOC Wert verursachte.

```
Position position = (Position) object;

if(cell.getColumnIndex() == 0) {
    checkQuestionCell(cell, position);
}

if (cell.getColumnIndex() == 1) {
    checkRoleCell(cell, position);
}

if (cell.getColumnIndex() == 2) {
    checkQuestionTypeCell(cell, position);
}

if (cell.getColumnIndex() > 2 && cell.getColumnIndex() < 23) {
    checkPossibleAnswerCell(cell, position);
}
```

Abbildung 77 Reduzierung des CC-Wertes mit IF-Statements

Um den hohen CC-Wert zu reduzieren, wurde in der *Abbildung 77* die Switch-Case-Anweisungen durch IF-Statements ersetzt und der Code für die bessere Lesbarkeit neustrukturiert.

Package entity

Der negative *Distance* Wert von -1 im *Package entity* berechnet sich durch die folgende Formel:

$$D = |A + I - 1|$$

Wobei *A* für *Abstractness*, d.h. die Anzahl abstrakter Klassen und *I* für *Instability* steht. Beide Werte haben je den Wert 0, was somit zu der Distanz von -1 führt. Dieser Wert lässt sich nicht verbessern, da die POJO-Klassen nur Felder und Getter-/Setter-Methoden besitzen.

4.2.2 Stand 01.06.2014

Use Case Titel	Priorität	Implementiert	Fehler/Unschönheiten	Status
An- / Abmelden	1	Ja		✓
Umfrage CRUD	1	Ja	kein nachträgliches Bearbeiten einer Umfrage möglich*	✓
Token generieren	1	Ja		✓
Token exportieren	1	Ja		✓
Token per Mail versenden	2	Ja		✓
Excel Template importieren	1	Ja		✓
Auswertungen ansehen	1	Ja		✓
Korrelation von 2 Fragen	1	Ja		✓
Export als Excel oder PDF	1	Ja		✓
Umfrage teilnehmen	1	Ja		✓
Benutzer verwalten	1	Ja		✓

Abbildung 78 Stand Juni

*einfaches Bearbeiten der Umfrage möglich und alle anderen Operationen wie Umfrage hinzufügen, ansehen oder löschen werden unterstützt. Bearbeiten einer Umfrage mit Fragen und Teilnehmer usw. jedoch nicht.

Am 01.06.2014 konnten weitgehend alle Use Cases implementiert werden. Das Exportieren der Auswertung als auch der Korrelation ist nun möglich. Zusätzlich können die Teilnehmer per E-Mail für eine Umfrage eingeladen werden. Das nachträgliche Bearbeiten von Umfragen mit ihren Fragen und Teilnehmer kann nicht mehr unterstützt werden, da dieses Feature sehr viel Zeit in Anspruch nehmen würde bis alles reibungslos läuft.

4.2.2.1 Durchgeführte Unit Tests

Finished after 28.119 seconds

Runs: 258/258 Errors: 0 Failures: 0

▲ TestSuite with 258 tests [example: junit.framework.TestSuite@105670d2] [Runner: JUnit 4] (28.058 s)

- ▶ TestSuite with 53 tests [example: ch.odag.teacheval.service.ImportServiceTest] (23.398 s)
 - ▶ ch.odag.teacheval.service.ImportServiceTest (11.661 s)
 - ▶ ch.odag.teacheval.service.ExportServiceTest (5.770 s)
 - ▶ ch.odag.teacheval.service.UserServiceTest (0.076 s)
 - ▶ ch.odag.teacheval.service.SurveyServiceTest (0.424 s)
 - ▶ ch.odag.teacheval.service.AuthenticationServiceTest (0.047 s)
 - ▶ ch.odag.teacheval.service.EvaluationServiceTest (0.305 s)
 - ▶ ch.odag.teacheval.service.AnswerServiceTest (0.394 s)
 - ▶ ch.odag.teacheval.service.MailServiceTest (4.721 s)
- ▶ TestSuite with 89 tests [example: ch.odag.teacheval.persistence.entity.SurveyTest] (0.786 s)
 - ▶ ch.odag.teacheval.persistence.entity.SurveyTest (0.312 s)
 - ▶ ch.odag.teacheval.persistence.entity.UserTest (0.031 s)
 - ▶ ch.odag.teacheval.persistence.entity.CourseTest (0.015 s)
 - ▶ ch.odag.teacheval.persistence.entity.CreatorTest (0.020 s)
 - ▶ ch.odag.teacheval.persistence.entity.InstructorTest (0.020 s)
 - ▶ ch.odag.teacheval.persistence.entity.ParticipantTest (0.039 s)
 - ▶ ch.odag.teacheval.persistence.entity.QuestionAnswerTest (0.011 s)
 - ▶ ch.odag.teacheval.persistence.entity.QuestionPossibleAnswerTest (0.016 s)
 - ▶ ch.odag.teacheval.persistence.entity.QuestionTest (0.299 s)
 - ▶ ch.odag.teacheval.persistence.entity.QuestionTypeTest (0.011 s)
 - ▶ ch.odag.teacheval.persistence.entity.RoleTest (0.012 s)
- ▶ TestSuite with 109 tests [example: ch.odag.teacheval.persistence.query.SurveyQueryTest] (3.720 s)
 - ▶ ch.odag.teacheval.persistence.query.SurveyQueryTest (0.507 s)
 - ▶ ch.odag.teacheval.persistence.query.UserQueryTest (0.227 s)
 - ▶ ch.odag.teacheval.persistence.query.AnswerQueryTest (0.246 s)
 - ▶ ch.odag.teacheval.persistence.query.CourseQueryTest (0.226 s)
 - ▶ ch.odag.teacheval.persistence.query.ParticipantQueryTest (0.246 s)
 - ▶ ch.odag.teacheval.persistence.query.EvaluationQueryTest (0.817 s)
 - ▶ ch.odag.teacheval.persistence.query.InstructorQueryTest (0.250 s)
 - ▶ ch.odag.teacheval.persistence.query.QuestionTypeQueryTest (0.261 s)
 - ▶ ch.odag.teacheval.persistence.query.PossibleAnswerQueryTest (0.204 s)
 - ▶ ch.odag.teacheval.persistence.query.QuestionQueryTest (0.514 s)
 - ▶ ch.odag.teacheval.persistence.query.RoleQueryTest (0.222 s)
- ▶ TestSuite with 7 tests [example: ch.odag.teacheval.service.authentication.AuthTokenManagerTest] (0.154 s)
 - ▶ ch.odag.teacheval.service.authentication.AuthTokenManagerTest (0.154 s)

Failure Trace

Abbildung 79 Stand 02.06.14

Insgesamt wurden in den Schichten *Persistence* und *Service* 258 Unit Tests durchgeführt. Der grösste Teil der REST-Schnittstellen konnte leider aus Zeitgründen nicht mehr richtig getestet werden.

4.2.2.2 Erreichte Test Coverage

AllTests (Jun 2, 2014 4:25:34 PM)

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
TeachEvalApp	67.0 %	18,705	9,197	27,902
src/main/java	57.1 %	11,001	8,250	19,251
ch.odag.teacheval.persistence.connection	82.9 %	29	6	35
ch.odag.teacheval.persistence.entity	64.2 %	1,283	714	1,997
ch.odag.teacheval.persistence.query	68.4 %	1,960	907	2,867
ch.odag.teacheval.rest	0.0 %	0	1,969	1,969
ch.odag.teacheval.service	52.5 %	2,965	2,681	5,646
ch.odag.teacheval.service.authentication	72.2 %	351	135	486
ch.odag.teacheval.service.comparator	100.0 %	9	0	9
ch.odag.teacheval.service.cron	0.0 %	0	108	108
ch.odag.teacheval.service.debug	0.0 %	0	75	75
ch.odag.teacheval.service.file.exports.correlation	88.1 %	407	55	462
ch.odag.teacheval.service.file.exports.evaluation	92.0 %	1,278	111	1,389
ch.odag.teacheval.service.file.exports.survey	100.0 %	328	0	328
ch.odag.teacheval.service.file.imports	61.3 %	2,154	1,362	3,516
ch.odag.teacheval.service.mail	55.5 %	96	77	173
ch.odag.teacheval.service.token	73.8 %	141	50	191
src/test/java	89.1 %	7,704	947	8,651

Abbildung 80 Erreichte Test Coverage 01.06.2014

Trotz mehr Unit Tests konnte die Testüberdeckung nicht markant erhöht werden. Der Grund dafür liegt darin, dass wir hauptsächlich Tests mit Anweisungsüberdeckung durchführten und nicht jeden Zweig und Pfad berücksichtigten.

4.2.2.3 Metriken

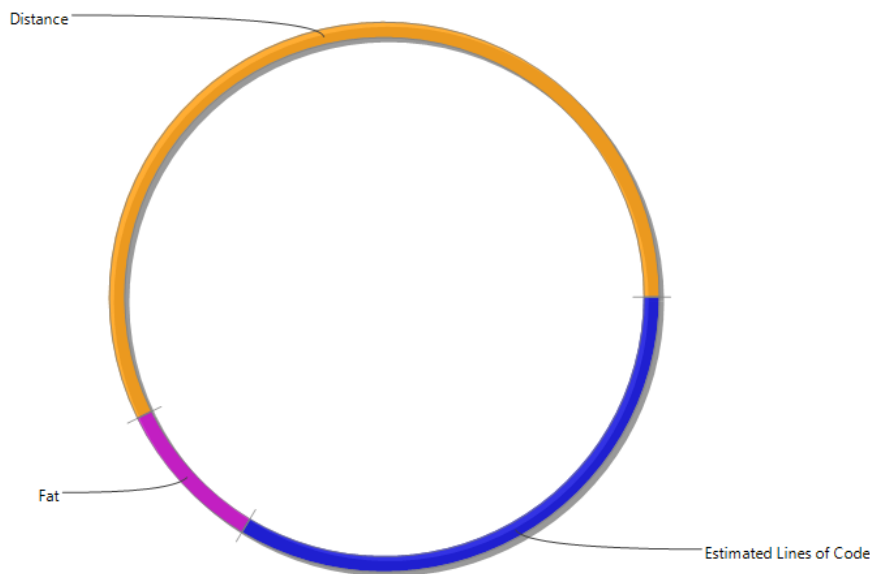


Abbildung 81 Pollution Stand 01.06.2014

Metrics	Value
Units	78
Classes / Class	0.01
Methods / Class	910.79
Fields / Class	3.34
ELOC (Estimated Lines of Code)	7692
ELOC / Unit	98.62
CC (Cyclomatic Complexity)	1.84
Fat	3
Tangled	0.00%

Abbildung 82 Metriken im /src/main/java

Diese Metriken stammen aus dem Source Verzeichnis /src/main/java. Im Test-Verzeichnis /src/test/java entsprechen die Werte wie folgt:

Metrics	Value
Units	44
Classes / Class	0
Methods / Class	9.57
Fields / Class	2.25
ELOC (Estimated Lines of Code)	3246
ELOC / Unit	73,77
CC (Cyclomatic Complexity)	1.17
Fat	0
Tangled	0.00%

Tabelle 50 Metriken im /src/test/java

4.2.2.4 Violations

In diesem Kapitel werden die Violations im Main-Verzeichnis (/src/main/java) aufgezeigt.

Package / Class	Metric	Value
persistence.entity	Distance	-1
persistence.connection		
service.mail		
service.debug		
service.comparator		
rest.SurveyRest	Fat	66
service.SurveyService	ELOC	343
service.file.exports.evaluation.AbstractSheet		
	ELOC	352

Tabelle 51 Violations /src/main/java

Distance

Im Mai haben wir die Packages auf das 3-Schichten-Modell umstrukturiert. Dies führte jedoch dazu, dass sich die *Distance* Metrik auf weitere Packages ausgeweitet hat.

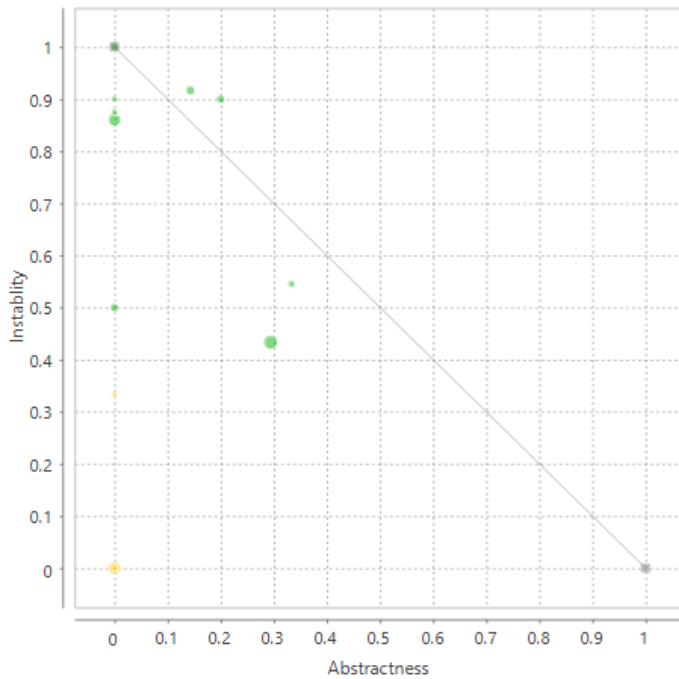


Abbildung 83 Distance Verteilung der Packages

Die Abbildung 83 zeigt, wie die Verteilung der Packages zur Haupt-Sequenz (diagonale Linie) liegt. Der Ordinatenabschnitt ist die *Instability* I und beschreibt, wie viele Packages auf das Package P abhängen und wie viele Packages P selber verwendet. Der auszugebende Wert I liegt zwischen 0 und 1, wobei 0 für nur einkommende Abhängigkeiten und 1 für nur auskommende Abhängigkeiten steht. Die Abszissenachse ist die *Abstractness* und zählt die Anzahl abstrakte Typen in einem Package P . Der Wert 0 bedeutet, dass nur konkrete Klassen im Package P liegen und 1 nur abstrakte Klassen oder Interface. Durch beide Werte lässt sich aussagen, wie weit entfernt (die Distance) ein Package von der Hauptsequenz liegt.

Die Distance Wert liegt zwischen -1 und +1 und lässt sich mit der Formel wie folgt berechnen:

$$D = |A + I - 1|$$

Die *Violations* aus Tabelle 51 *Violations* haben alle mit Ausnahme des Comparators einen Distance-Wert von -1. Diese entsprechen aus Abbildung 83 den grossen gelben Punkt (0|0). Der Comparator liegt ungefähr bei (0|3.3).

Als Lösung sehen wir momentan nur 2 Möglichkeiten, um die Distanzen zu reduzieren:

- Zusammenführen mehrere Packages in ein einziges Package um die Abhängigkeiten zueinander auszuhebeln
- Die Packages für die Weiterentwicklung so belassen, da in Zukunft vielleicht weitere Klassen in diese Packages hinzukommen.

Die Package-Struktur ist nach unserer Meinung her gut aufgebaut und es gibt keinen Anlass aufgrund dieser Metrik alle in ein Package zu verschieben.

ELOC

Der nächstgrösste Bereich ist die *Estimated Lines of Code (ELOC)*. Dabei haben zwei Klassen mit 352 und mit 343 Zeilen einen leichten Übertritt von den empfohlenen 300 Zeilen. Diese wurden mit *Extract Method* neustrukturiert.

4.3 Verbesserungsmöglichkeiten

Wie bereits erwähnt, konnten aus Zeitgründen nicht mehr alle REST-Schnittstellen mit Jersey Test Framework getestet werden. Für die Weiterführung oder als Folgearbeit müsste dies nachgeführt werden.

5. Leistungstests

Aus den nichtfunktionalen Anforderungen wurde für die Ladezeit der Applikation einen Richtwert von 3 Sekunden definiert. Die Ladezeiten von TeachEval wurden mit Pingdom Speed₁ gemessen. Dabei wird die Antwortzeit aus unterschiedlichen Regionen gemessen. Mit Pingdom konnten wir testen, wie lange TeachEval aus Amsterdam oder New York für den Aufruf der Webseite benötigt. Die Leistung wurde bewusst nicht nur aus der Schweiz getestet.

In den folgenden Ladezeiten, wird das komplette Laden der Webapplikation gemessen. Da es sich bei TeachEval um eine Single Page App handelt, wurden bei den 63 Request der grösste Teil der Applikation geladen und gecached. Unterseiten sind daher schneller zugreifbar, da sie bereits vorhanden sind.

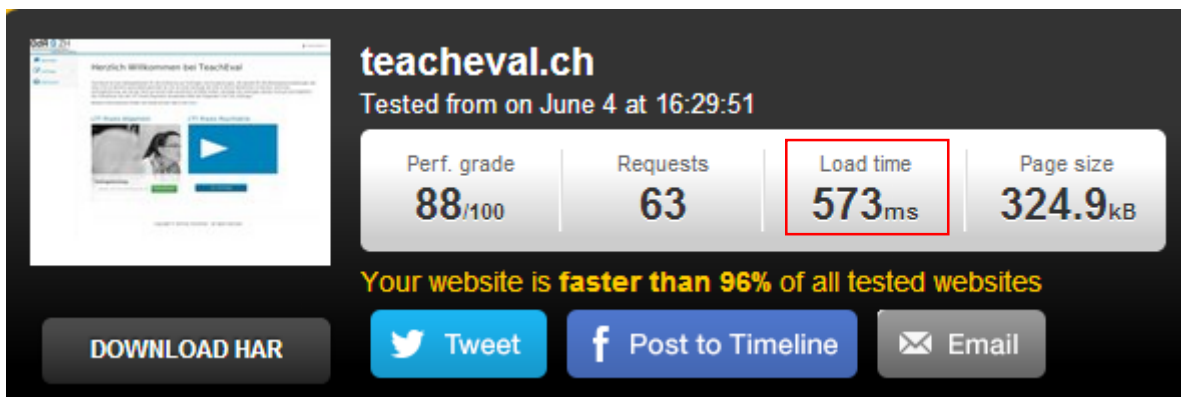


Abbildung 84 Ladezeit Startseite Amsterdam, Holland

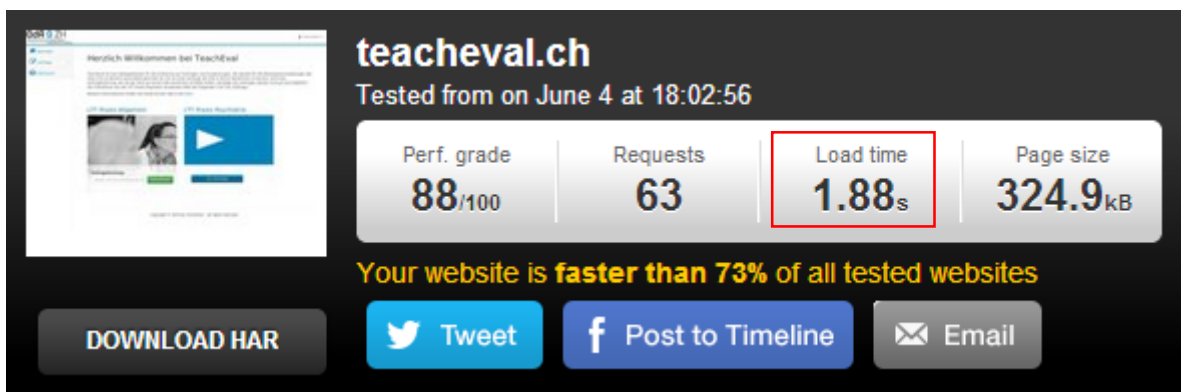


Abbildung 85 Ladezeit Startseite New York, USA

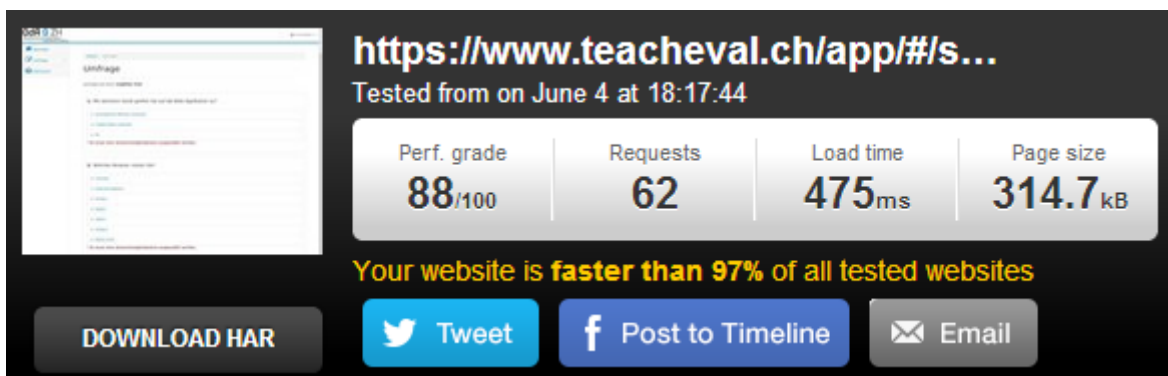


Abbildung 86 Ladezeit Umfrage Amsterdam, Holland

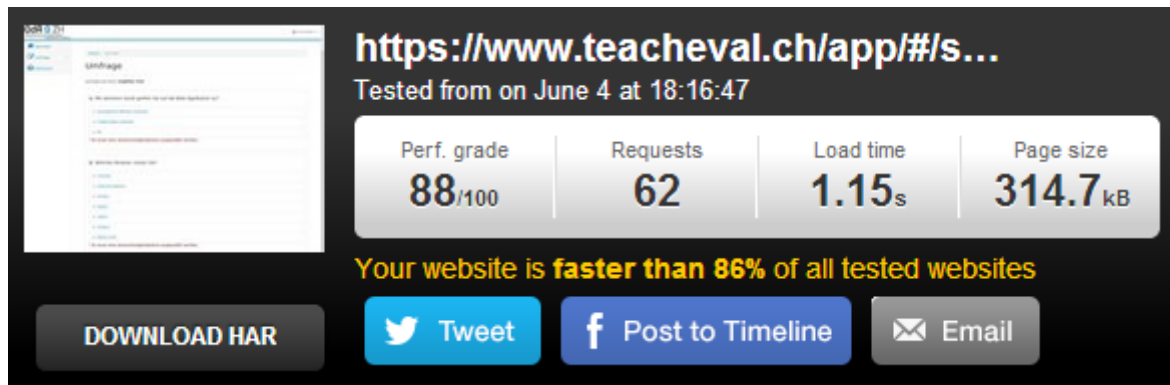


Abbildung 87 Ladezeit Umfrage New York, USA

Da wir grundsätzlich keine grosse Datenmenge zwischen Client und Server verschicken, hat die Webapplikation gute Ladezeiten. Durch das asynchrone Laden der Skripte konnte viel Performance herausgeholt werden. Natürlich könnten noch weitere Optimierungen vorgenommen werden. Beispielsweise könnten die selbstgeschriebenen Skripte mit einer JavaScript Library, wie r.js, minimiert und komprimiert werden. Zudem könnten die Anzahl Requests besser auf die einzelnen Bereiche, bzw. Module verteilt werden, was aus Zeitgründen nicht mehr drin lag.

6. Kompatibilitätstests

6.1 Einleitung

Heutzutage gibt es eine breite Palette von Browsern auf dem Markt wie bspw. dem Internet Explorer, Chrome, Firefox, Opera, etc. Durch das W3C-Konsortium gibt es einen offiziellen Standard, um die Browserhersteller dazu zu animieren die Darstellung von HTML, CSS, etc. Seiten zu vereinheitlichen. Jedoch halten sich nicht immer alle Hersteller an den Standard, oder bieten eigene zusätzliche Funktionen an, die zu Unstimmigkeiten oder auch zu Browserkompatibilitätsprobleme führen kann. Dies hat sich zwar in den letzten Jahren um einiges verbessert, jedoch kann es trotzdem noch zu Browserinkompatibilitäten führen wenn z.B. eine Webapplikation auch ältere Browser unterstützen muss. In diesem Kapitel werden die Kompatibilitätsergebnisse protokolliert.

6.2 Angaben zur Durchführung

Der Kompatibilitätstest wurde mit den Browsern IE8, IE9, IE10, IE11, Firefox 23, Chrome 25, Safari 5.1 und Opera 12.14 durchgeführt. Der Internet Explorer hat eine eingebaute Rückwärtskompatibilitätsfunktion, um zwischen den IE Versionen zu wechseln. Zudem wurden mit Test Geräte (Tablets und Smartphones) die Bedienung, das Ausfüllen einer Umfrage und die Verwaltung getestet. Darunter fallen ein iPad2, iPad3, iPhone4, iPhone5, Android Nexus S und ein HTC One. Das Android Nexus erhielten wir von der HSR als Testgerät, um die Applikation auch mit Android vollständig testen zu können.

Für die Kompatibilität anderer Browser wurden Screenshots über Drittanbieter wie *Browsershots* und *Browserstack* gemacht. Damit können aber keine Kompatibilitätsüberprüfungen im internen Bereich von TeachEval gemacht werden.

6.3 Einschränkung

Für Kompatibilitätsüberprüfungen konnten wir ein Android Testgerät der HSR ausleihen. Andere Testgeräte werden mit den oben genannten öffentlichen, kostenlosen Diensten simuliert.

6.4 Überblick aller Tests

Browser	Fehler/Unschönheiten/Verbesserungen	Status
Internet Explorer 8	Teilnehmer können ohne Probleme mit dem IE8 an einer Umfrage teilnehmen. Es wurden kürzlich Probleme mit dem Import von Dateien festgestellt.	(✓)
Internet Explorer 9	Teilnehmer können ohne Probleme mit dem IE9 an einer Umfrage teilnehmen. Es wurden kürzlich Probleme mit dem Import von Dateien festgestellt. Zudem können bei IE9 die Cookies nachdem sie gesetzt worden sind nicht mehr entfernt werden.	(✓)
Internet Explorer 10	IE10 läuft einwandfrei nachdem serverseitig ein Cache Control bei GET Abfragen genutzt wird.	✓
Internet Explorer 11	IE11 läuft einwandfrei.	✓
Firefox 29.0.1	Firefox läuft einwandfrei.	✓
Chrome 35.0.1916.114	Chrome läuft einwandfrei	✓
Safari 5.1.7	Safari läuft einwandfrei	✓
Opera	Opera läuft einwandfrei	✓

Abbildung 88 Überblick der Kompatibilitätstests

6.5 Internet Explorer 8

Mit dem IE8 entstanden am Anfang die meisten Probleme, da viele JavaScript Funktionen erst in den späteren IE Versionen unterstützt werden. Dank der Kompatibilitätsbibliotheken *html5shiv* und *es5shim* wird die Webapplikation auch für IE8 unterstützt.

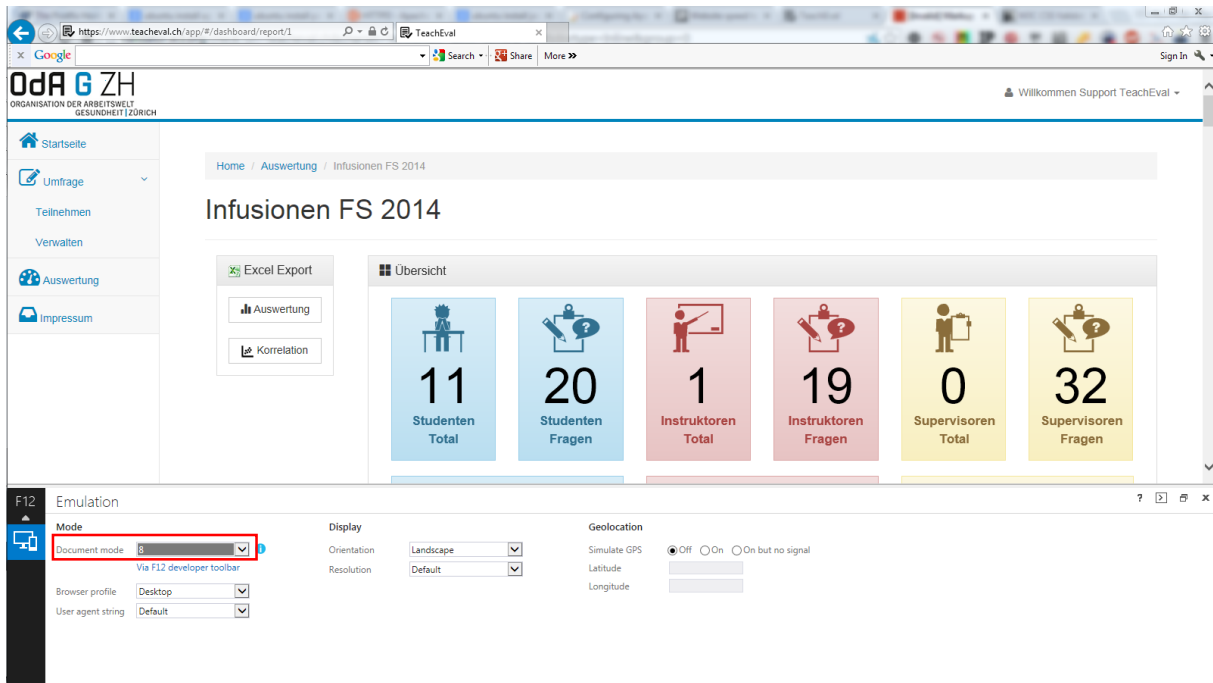


Abbildung 89 TeachEval mit IE8 Support

Mit einer JQuery Placeholder Bibliothek wird der Vorgabetext bei einer Eingabemaske auch für den IE8 oder IE9 unterstützt. Die folgende Abbildung zeigt die Problematik:

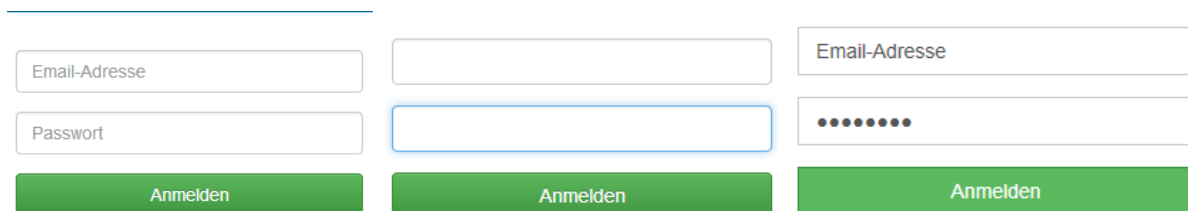


Abbildung 90 IE10 Placeholder Support

Abbildung 91 IE8 – IE9 ohne Placeholder Bibliothek

Abbildung 92 IE8 – IE9 mit JQuery Placeholder Bibliothek

6.6 Internet Explorer 9

Der grösste Teil der Applikation läuft ohne Kompatibilitätsbibliotheken relativ stabil. Jedoch traten Probleme mit dem Cache vom IE9 auf, die serverseitig mit einem Cache Control gelöst wurden.

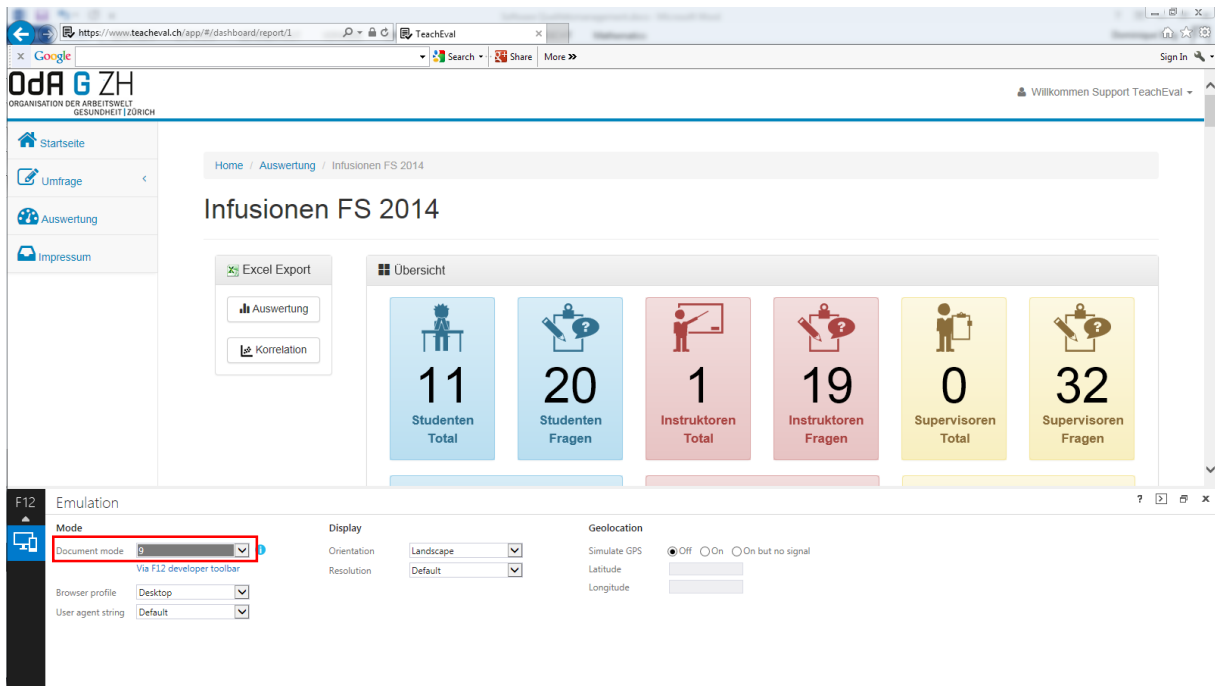


Abbildung 93 TeachEval mit IE9 Support

6.7 Internet Explorer 10

IE10 läuft stabil. Mit einem Cache Control wird serverseitig das Cachen von dynamischen Inhalte bei GET Abfragen unterbunden.

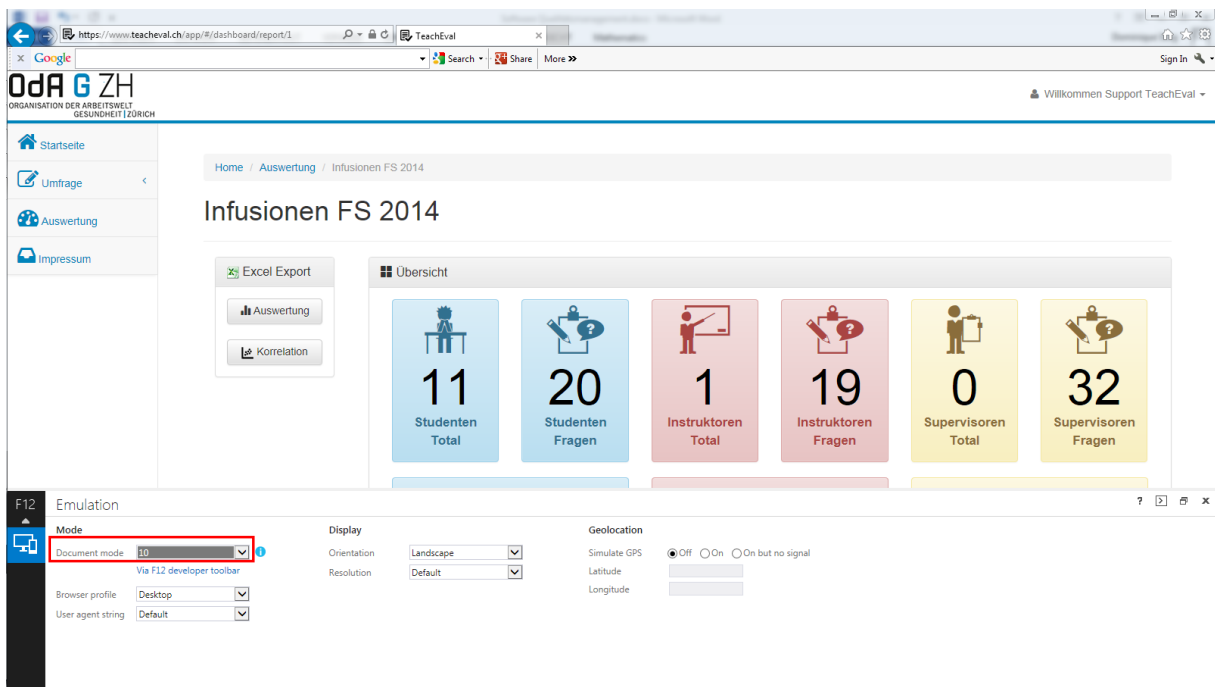


Abbildung 94 TeachEval mit IE10 Support

6.8 Firefox

Innerhalb von Firefox läuft TeachEval problemlos.

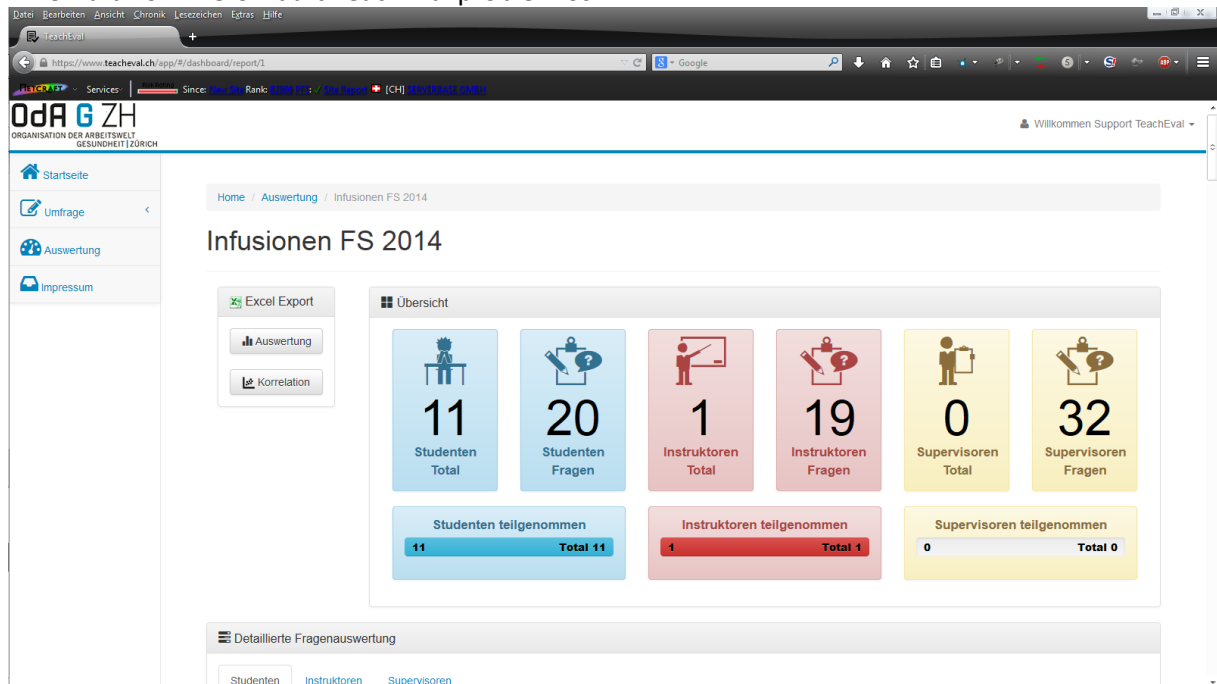


Abbildung 95 TeachEval mit Firefox 29.0.1 Support

6.9 Chrome

In Chrome gibt es keine bekannten Probleme mit TeachEval.

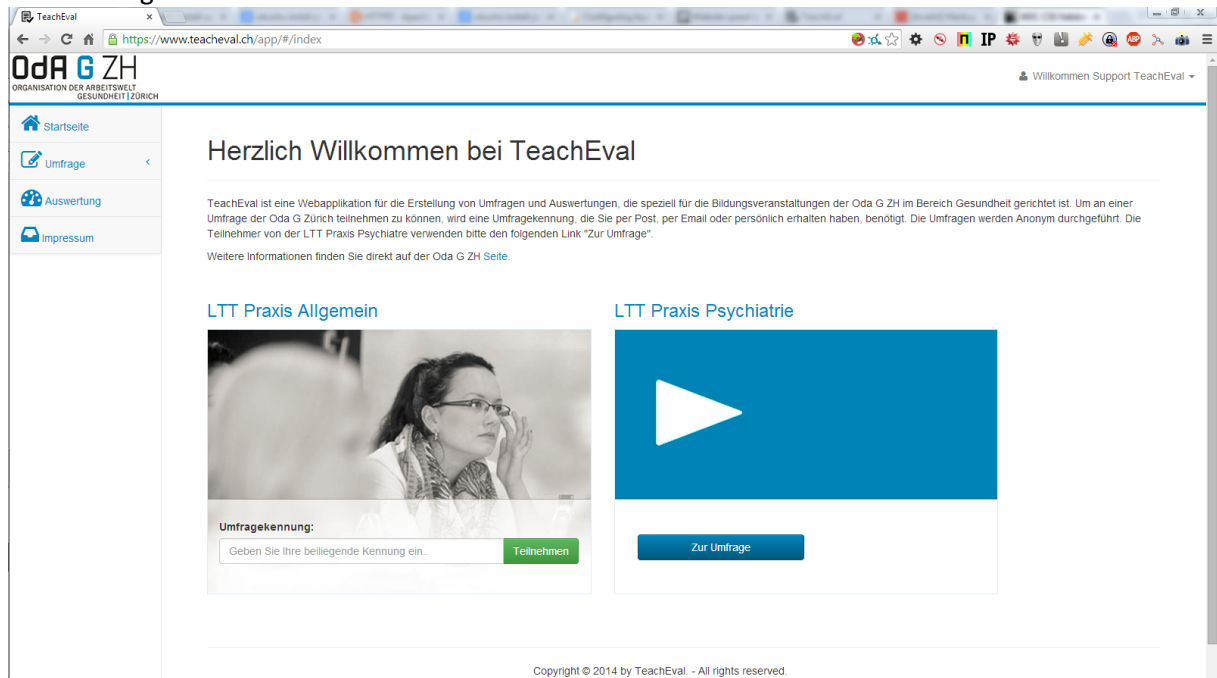


Abbildung 96 TeachEval Chrome 35.0 Support

6.10 Safari

Da Safari den „`<input required>`“ Tag nicht kennt, werden die Formulare unvalidiert übermittelt. Damit dies unterstützt wird, implementieren wir mit AngularJS eine eigene Validierung für das absenden von Umfragen.

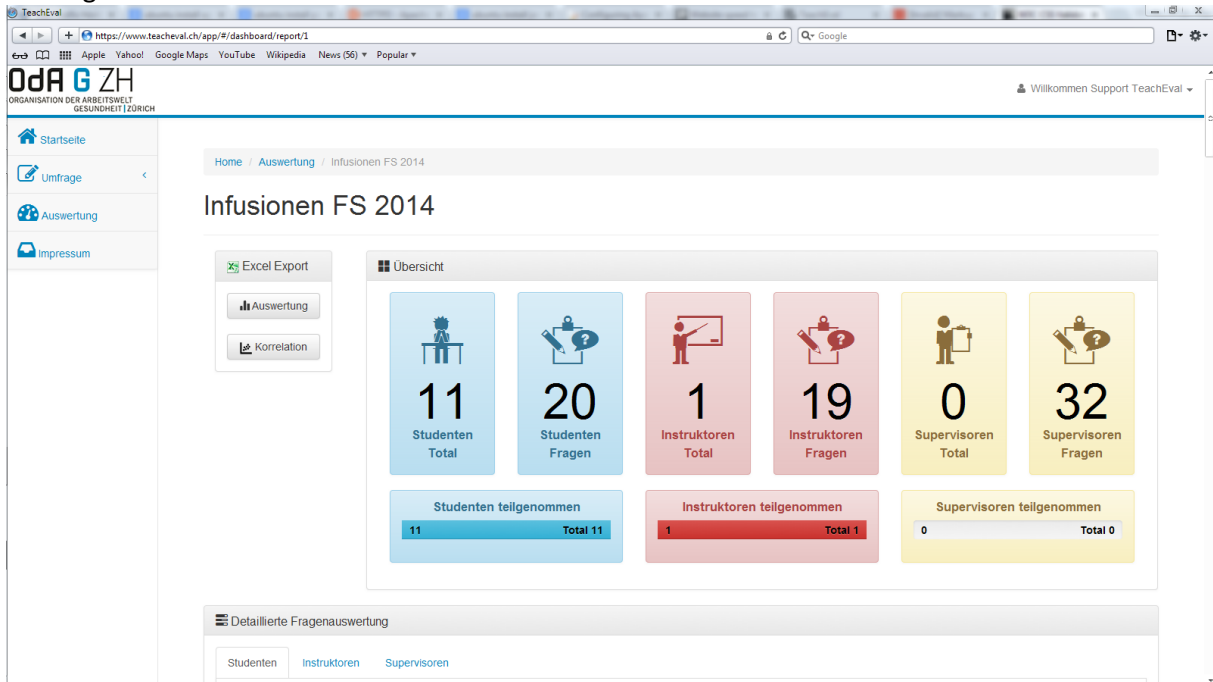


Abbildung 97 TeachEval Safari 5.1.7 Support

6.11 Opera

Mit Opera konnten wir eine stabile Version von TeachEval testen.

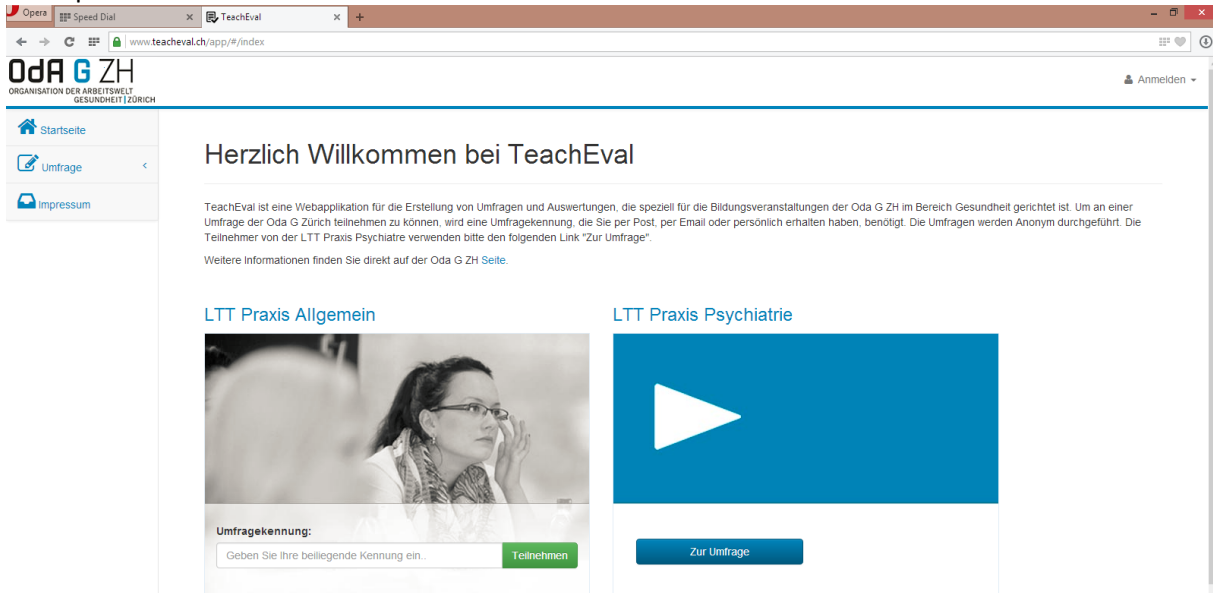


Abbildung 98 TeachEval Opera 21.0

6.12 Browserstack

Vor allem für die mobilen Testgeräte wurde ein online Kompatibilitätstest mit Screenshots durchgeführt. Damit erhalten wir einen Überblick, auf welchen Geräte TeachEval läuft.

Rapidly test your website for cross browser compatibility across 300+ browsers.

www.teacheval.ch History ▾

To test your internal web server or design files, setup local testing. Start local testing or Use binaries

23/25 browsers selected ×

OS	Device	Browser	Version					
iOS	v5 iPad 2	v5.1 iPhone 4S iPad 3	v6 iPhone 4S iPhone 5 iPad 3 iPad Mini					
	v7 iPhone 5S iPad 3rd							
	Android		Amazon Kindle Fire 2 Kindle Fire HD 8.9	Samsung Galaxy S Galaxy S II Galaxy S III Galaxy Tab 2 Galaxy Note Galaxy Note II Galaxy Note 10.1 Galaxy Nexus	Motorola Razr Droid Razr Atrix HD	Sony Xperia Tipo Google Nexus 7	LG Nexus 4 Optimus 3D HTC One X Wildfire Evo 3D	
			Windows XP	6 7	3.6 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12.10 12.14 12.15 12.16	5.1
			Windows 7	8 9 10 11	3.6 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12.10 12.14 12.15 12.16	5.1
			Windows 8	10 Desktop	16 17 18 19 20 21 22 23 24 25	22 23 24 25 26 27 28 29 30 31	12 12.10 12.14 12.15 12.16	5.1
			Windows 8.1	11 Desktop	16 17 18 19 20 21 22 23 24 25	22 23 24 25 26 27 28 29 30 31	12 12.10 12.14 12.15 12.16	5.1
			Mac OS X Snow Leopard	4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12 12.12 12.14 12.15	5.1	
			Mac OS X Lion	3.6 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12 12.12 12.14 12.15	6	
			Mac OS X Mountain Lion	3.6 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12 12.12 12.14 12.15	6.1	
Mac OS X Mavericks		3.6 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	14 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	11.6 12 12.12 12.14 12.15	7			

Abbildung 99 Browsertest über Browserstack



Abbildung 100 iPhone 5



Abbildung 101 iPhone 4s

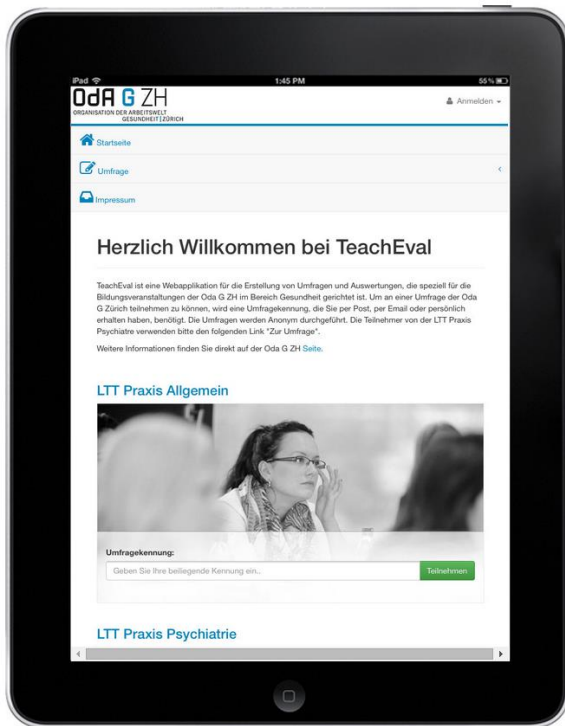


Abbildung 102 iPad 3 – Portrait

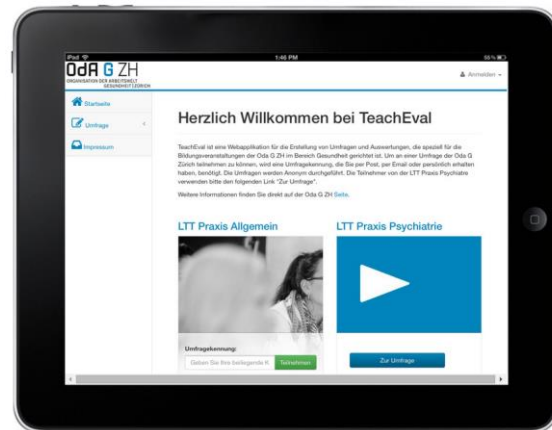


Abbildung 103 iPad 3 – Landscape

Weitere Screenshots wurden in einem separaten Ordner unter „Qualitätsmanagement“ gesammelt. Leider wurden bei *Browsershots* und *Browserstack* bei einigen Geräten die Screenshots zu schnell ausgelöst. Somit erscheint nur ein weisser Bildschirm, da bei der kostenlosen Variante keine Verzögerung eingestellt werden kann.

7. Fazit

Dank dem Twitter Bootstrap Framework konnten die meisten Browser unterstützt werden, welche heutzutage häufig genutzt wurden. Mit dem Responsive Design können zudem unterschiedliche Geräte, wie PC, Tablet oder Smartphone unterschieden und individuell angesprochen werden. Mit dem Einsatz von Kompatibilitätsbibliotheken kann auch Internet Explorer 8 oder 9 unterstützt werden. Zudem wird mit einer *JQuery Placeholder Bibliothek* der Vorgabetext auch für IE8 und IE9 angezeigt.

8. Usability Test

8.1 Ziel und Zweck

Die Usability Tests sind Teil des Qualitätsmanagements. Sie wird durchgeführt, um die Benutzerfreundlichkeit und die Gebrauchstauglichkeit von TeachEval mit den Teilnehmern zu überprüfen. Das heisst, die Teilnehmer eines Kurses füllen zusätzlich die Usability Umfrage aus. Dafür wurde die Umfrage mit der Excel Vorlage erstellt und hochgeladen. Ziel ist es, anhand von gezielten Fragen die möglichen Stärken und Schwächen in Bezug auf die Benutzerfreundlichkeit der Webapplikation aufzuzeigen.

Eine zweite Art von Usability Test konnten wir bei einer kleinen Schulung bei der OdA G ZH durchführen. Wir erklärten der Leiterin die Applikation und sie bediente diese zugleich. Dabei konnten wir ihr über die Schulter schauen und analysieren, wie sie die Applikation bedient und auf was wir noch achten müssen.

8.2 Testpersonen

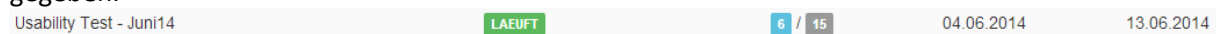
Als Testperson nehmen einerseits die Leiterin und andererseits die ausgewählten Studenten teil. Die Testpersonen sollen möglichst unbelastet und neutral sein. Das bedeutet, dass diese keine Konkurrenzprodukte zuvor getestet haben oder zu den Entwicklern von TeachEval gehören. Bei den Testpersonen, nehmen wir an, dass es sich mehrheitlich um Studenten im Bereich Medizin handelt.

8.3 Durchführung

Der Usability Test wurde während eines Unterrichts durch die Leiterin durchgeführt. Dabei haben die Studenten als Erstes die Umfrage zum Unterricht und anschliessend den Usability Test ausgefüllt. Die Leiterin selber hat am Usability Test als Dozent mitgewirkt.

8.4 Statistiken

An den Usability Tests haben insgesamt 6 von 15 Personen teilgenommen und uns ein Feedback gegeben.



Usability Test - Juni14	LAEUFT	6 / 15	04.06.2014	13.06.2014
-------------------------	--------	--------	------------	------------

Abbildung 104 Übersicht Usability Test

Warum weniger als die Hälfte mitgemacht haben, können wir nur spekulieren. Die Umfrage war schliesslich nicht für alle verbindlich, was vermutlich das geringe Interesse der Anzahl Teilnehmenden erklärt.

Die Auswertungen wurden als Excel-Datei exportiert und anschliessend wurden die Diagramme in den nächsten Abschnitten eingefügt.

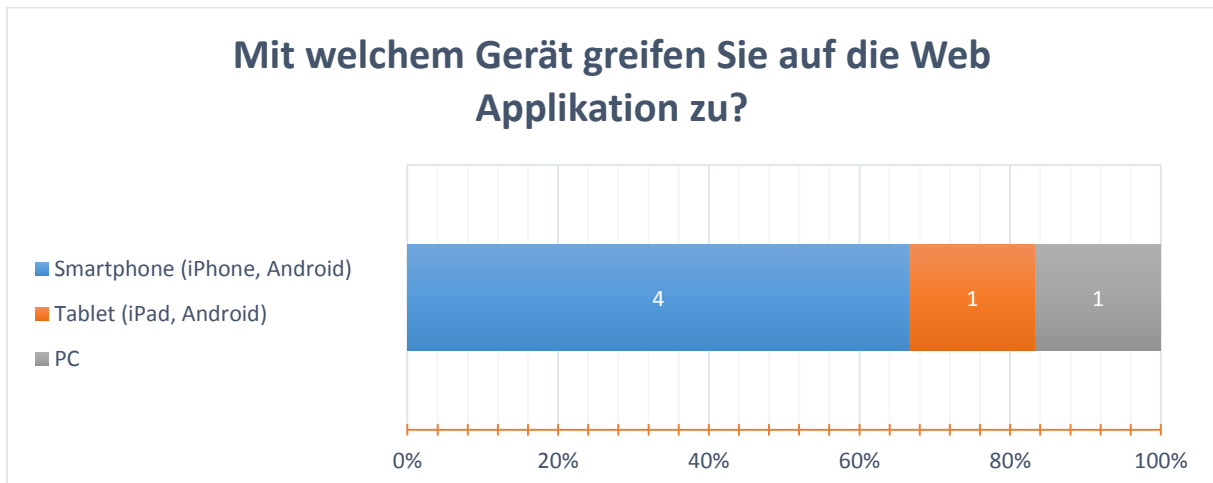
Frage 1:

Abbildung 105 Resultate von Frage 1

Aus den Usability Tests wird ersichtlich, dass die Teilnehmer mehrheitlich mit den Smartphones zugreifen. Daher konnten wir die Benutzerfreundlichkeit vor allem auf Smartphones gut testen.

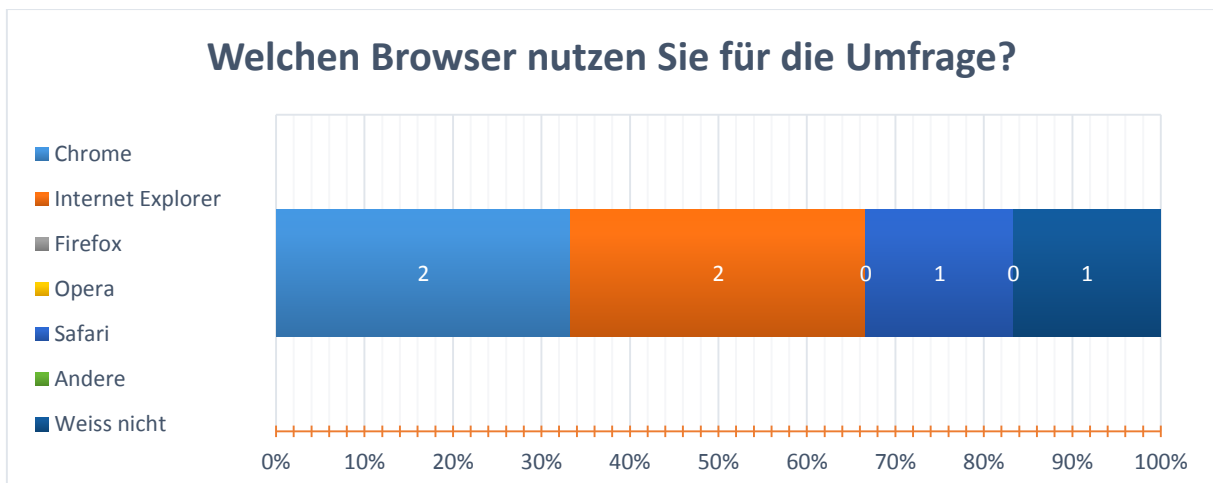
Frage 2:

Abbildung 106 Resultate von Frage 2

Für den Zugriff auf die Umfragen wurden hauptsächlich der Internet Explorer oder der Chrome Browser verwendet. Eine Testperson enthielt sich bei dieser Frage.

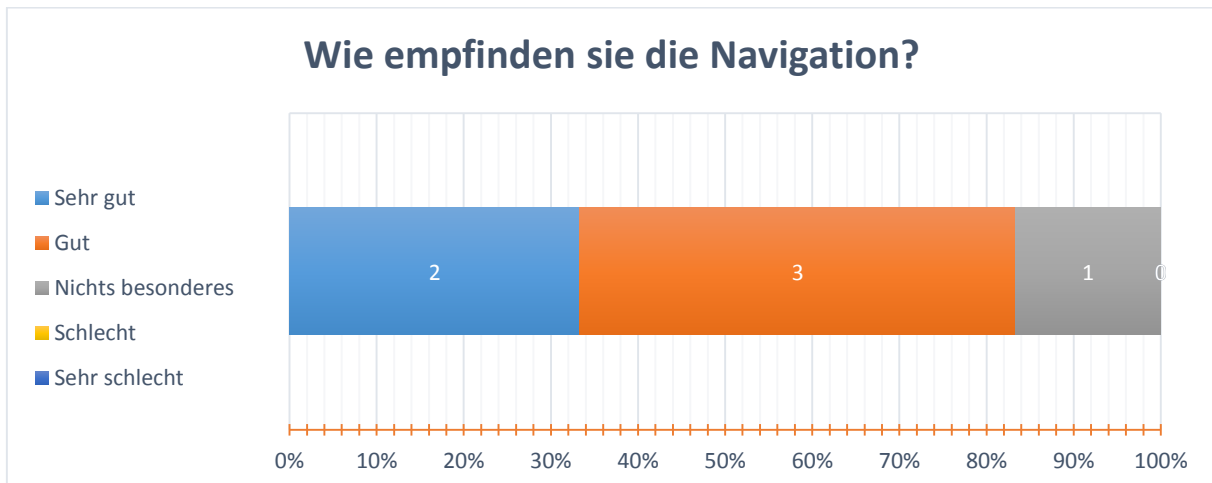
Frage 3:

Abbildung 107 Resultate von Frage 3

Die Navigation kam bis auf eine Stimme mehrheitlich gut an.

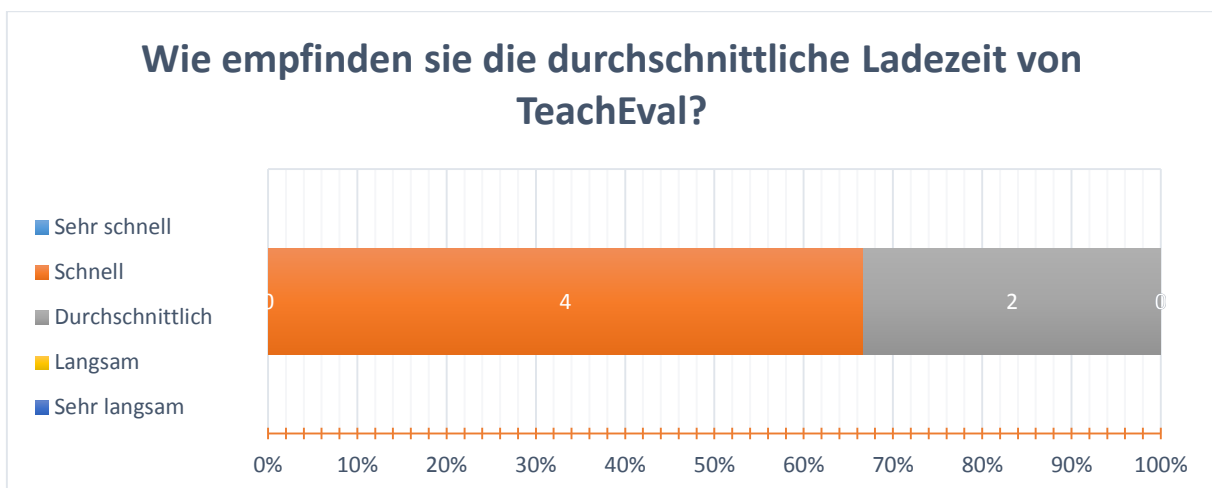
Frage 4:

Abbildung 108 Resultate von Frage 4

Die meisten Teilnehmer empfinden die Ladezeit als schnell und 2 haben es als durchschnittlich bewertet. Warum sie es nur als durchschnittlich empfanden, können wir nur darauf spekulieren, dass entweder alte Hardware (die meisten haben mit Smartphone zugegriffen) oder eine alte Version eines Webbrowsers verwendet wurde.

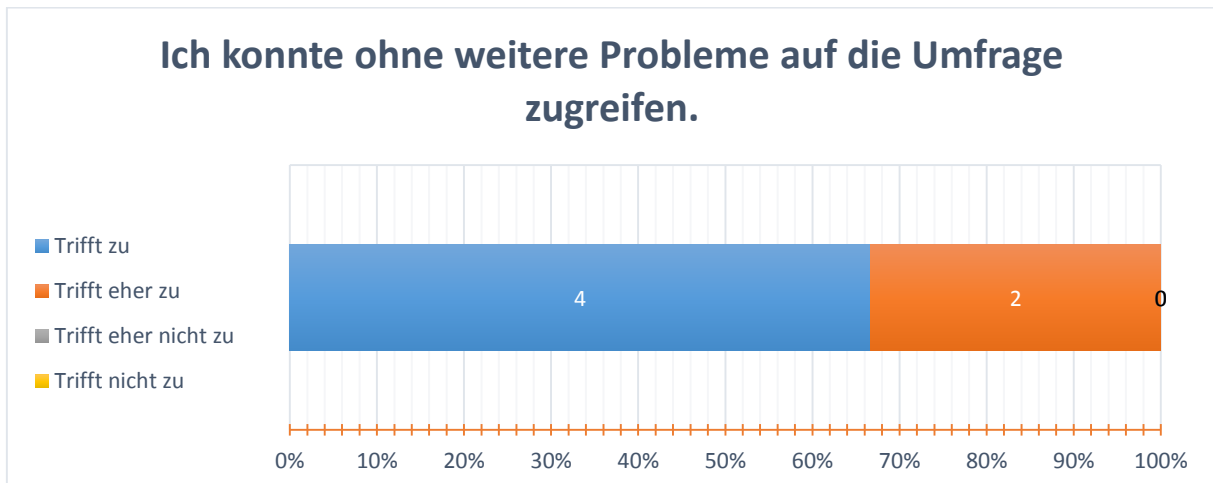
Frage 5:

Abbildung 109 Resultate von Frage 5

Es konnten alle Teilnehmer an der Umfrage teilnehmen, wobei zwei anscheinend Probleme damit hatten.

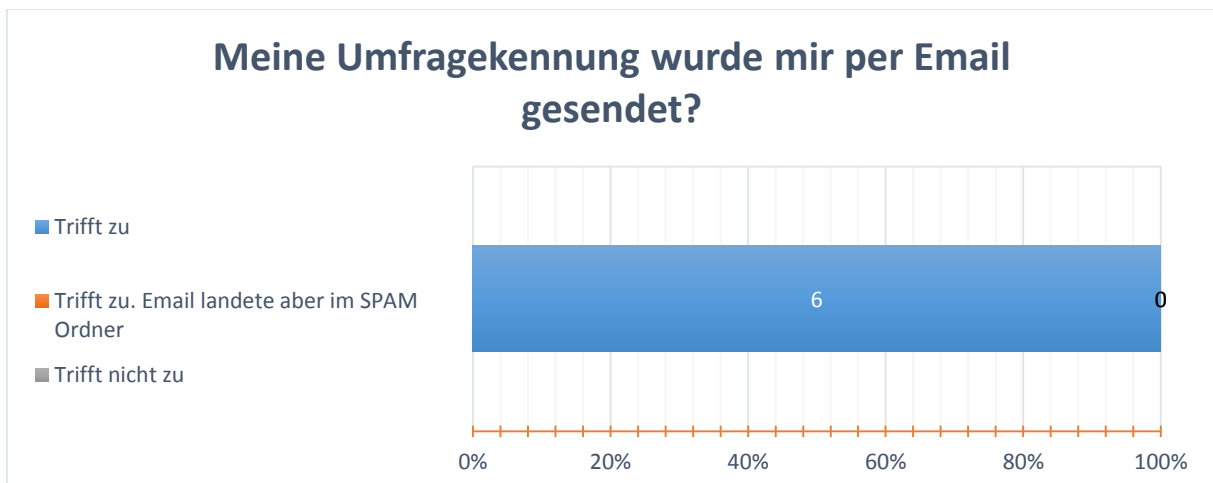
Frage 6:

Abbildung 110 Resultate von Frage 6

Alle Teilnehmer bekamen eine E-Mail mit Umfragelink und keine der E-Mails wurde als Spam deklariert.

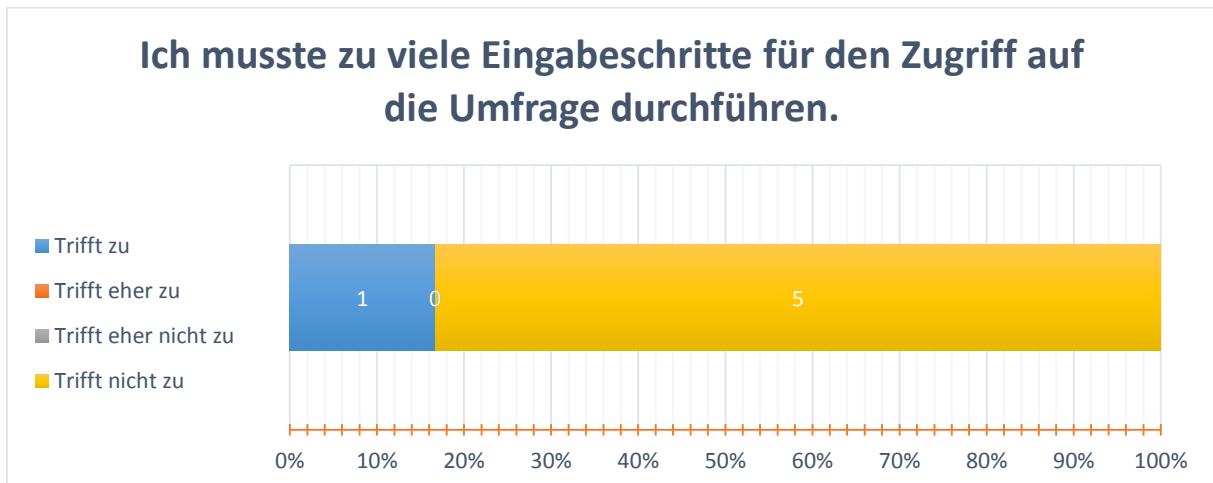
Frage 7:

Abbildung 111 Resultate zur Frage 7

Die meisten hatten keine Probleme um auf die Umfrage zuzugreifen und sie mussten wenige Eingabeschritte unternehmen. Eine Person empfand die Anzahl Eingabeschritte als zu viel, obwohl sie laut der Frage 6 ebenfalls eine E-Mail Einladung mit der Umfragelink erhielt.

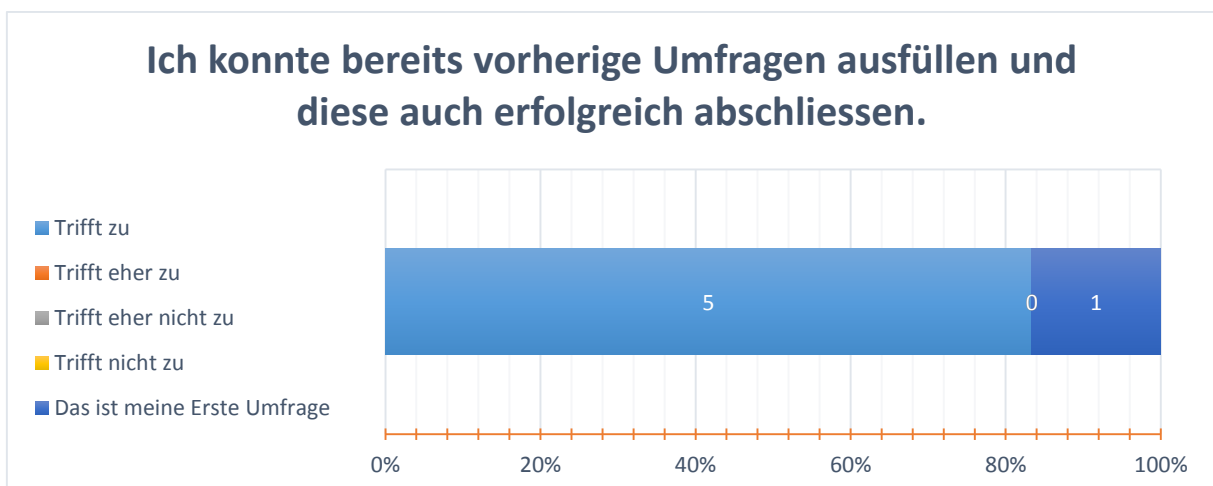
Frage 8:

Abbildung 112 Resultate zur Frage 8

Es fällt auf, dass es für eine Person die erste Umfrage mit unserem Tool war. Vielleicht war es auch die Person, die zu Beginn Probleme hatte.

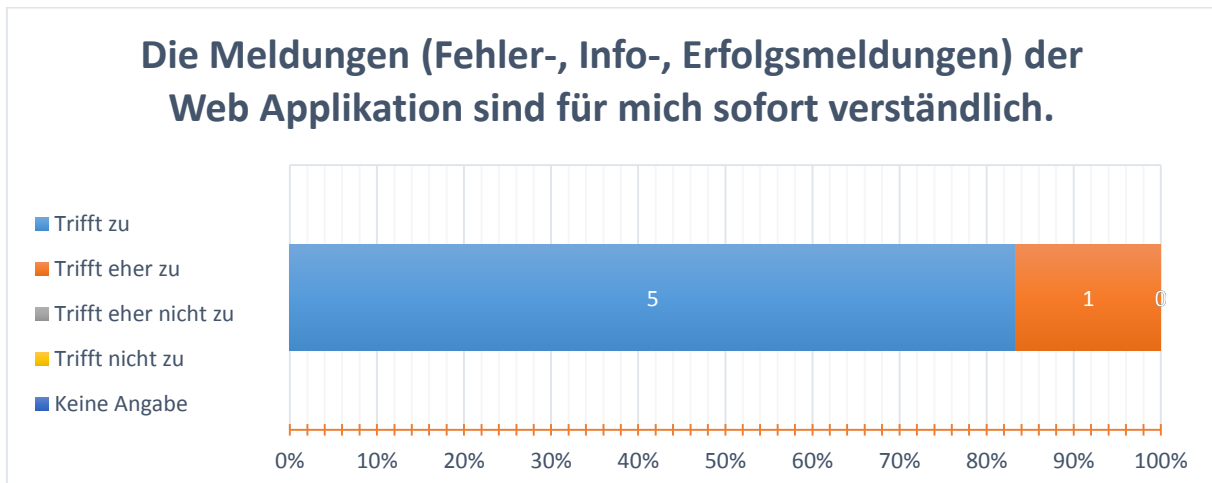
Frage 9:

Abbildung 113 Resultate zur Frage 9

Die Meldungen von TeachEval wurden für die meisten als verständlich wahrgenommen.

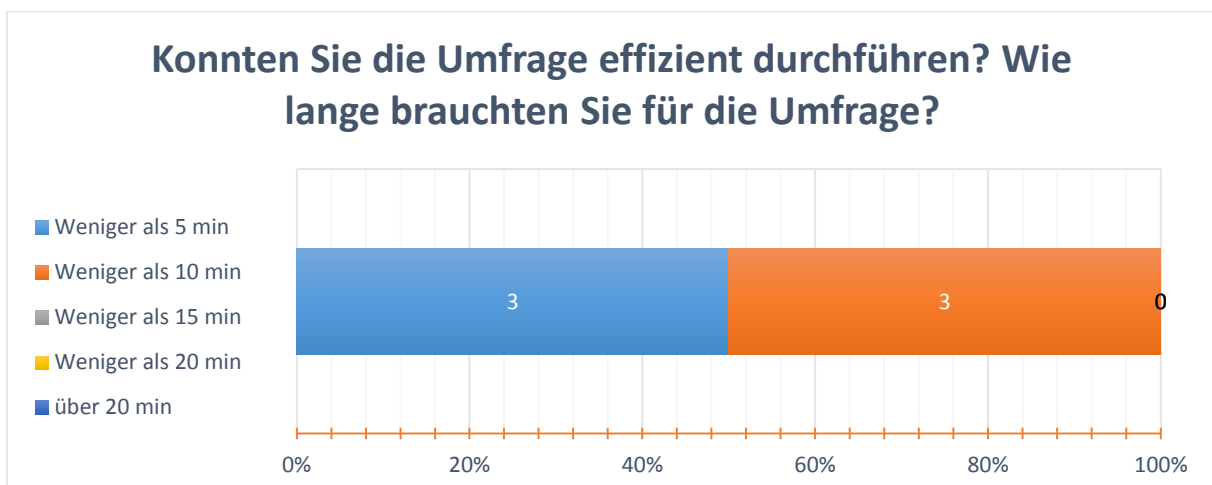
Frage 10:

Abbildung 114 Resultate zur Frage 10

Der Usability Test konnte von allen teilgenommenen innerhalb von 10 min ausgefüllt werden. Man kann sagen, dass die Teilnehmer die Umfrage effizient ausfüllen konnten.

8.5 Bewertung

Die Bewertung der Statistiken werden zu den einzelnen ISO 9241-110 Kriterien gegenübergestellt. Mit einem Dreisatz wird aus den Statistiken die prozentuale Bewertung angegeben. In der folgenden Tabelle werden die Gewichtung zu den einzelnen Antwortmöglichkeiten angegeben um die Antworten in Zahlen ausrechnen und ausgeben zu können.

Definition Gewichtung:

Antwort			Gewichtung
Trifft zu	Sehr gut	Sehr schnell	2
Trifft eher zu	Gut	Schnell	1
keine Aussage möglich	Nichts besonderes	Durchschnittlich	0
Trifft eher nicht zu	Schlecht	Langsam	-1
Trifft nicht zu	Sehr schlecht	Sehr langsam	-2

Tabelle 52 Übersicht Gewichtung der Antwortmöglichkeiten

Um die maximale Prozentzahl der einzelnen Kriterien zu berechnen, wurde aus Tabelle 52 die Gewichtung mit Wertung 2 als 100% gewählt. Der Best Case ist, wenn alle Teilnehmer die beste Wertung (Gewichtung 2) abgeben. Für die Berechnung wurde folgende Formel angewendet:

$$\frac{\sum \text{Anzahl Antworten} * \text{Gewichtung}}{\text{Total Anzahl Teilnehmer}} = x$$

8.5.1 Verfügbarkeit

Fragen	Interpretation	in Prozent
Mit welchem Gerät greifen Sie auf die Webapplikation zu?	Alle konnten auf die Webapplikation zugreifen	100%
Welchen Browser nutzen Sie für die Umfrage?		
Meine Umfragekennung wurde mir per Email gesendet?	Alle bekamen eine E-Mail	100%
Total		<u>100%</u>

Von den 6 Teilnehmern konnten alle ohne grössere Probleme auf die Umfrage zugreifen. Wir wurden nicht angeschrieben, dass gewisse Teilnehmer nicht teilnehmen können.

8.5.2 Aufgabenangemessenheit

Fragen	Interpretation	Gewichtung	Mittelwert
Ich konnte ohne weitere Probleme auf die Umfrage zugreifen.	4x Trifft zu	2	1.332
	2x Trifft eher zu	1	0.333
Total 2 = 100% => 1.665 = 2x			83%
Meine Umfragekennung wurde mir per Email gesendet	6x Trifft zu	2	2.00
Total 2 = 100% => 2.00 = 2x			100%
Ich musste zu viele Eingabeschritte für den Zugriff auf die Umfrage durchführen.	5x Trifft nicht zu	2	1.667
	1x Trifft zu	-2	-0.334
Total 2 = 100% => 1.333 = 2x			67%
Gesamtresultat auf alle Fragen bezogen			<u>83%</u>

Es konnten 83% aller Aufgaben mit guten Erwartungen erfüllt werden.

8.5.3 Selbstbeschreibungsfähigkeit

Fragen	Interpretation	Gewichtung	Mittelwert
Die Meldungen (Fehler-, Info-, Erfolgsmeldungen) der Webapplikation sind für mich sofort verständlich.	5x Trifft zu	2	1.666
	1x Trifft eher zu	1	0.167
Total 2 = 100% => 1.833 = 2x			<u>92%</u>
Gesamtergebnis auf alle Fragen bezogen			<u>92%</u>

Durch die verständlichen Fehler oder Informationsmeldungen wird der Benutzer zu 92% durch die Umfrage geleitet.

8.5.4 Erwartungskonformität

Fragen	Interpretation	Gewichtung	Mittelwert
Wie empfinden sie die Navigation?	2x Sehr gut	2	0.8
	3x Gut	1	0.6
	1x Nichts besonderes	0	
Total 2 = 100% => 1.4 = 2x			70%
Wie empfinden sie die durchschnittliche Ladezeit von TeachEval?	4x Schnell	1	0.5
	2x Durchschnittlich	0	
Total 2 = 100% => 0.5 = 2x			50%
Gesamtergebnis auf alle Fragen bezogen			<u>60%</u>

Durch das effiziente Ausfüllen der Umfragen interpretieren wir die Erwartungskonformität der Teilnehmer mit 60% als hoch ein. Die durchschnittliche Ladezeit mit 50% könnte besser sein.

8.5.5 Erlernbarkeit

Fragen	Interpretation	Gewichtung	Mittelwert
Ich konnte bereits vorherige Umfragen ausfüllen und diese auch erfolgreich abschliessen.	5x Trifft zu	2	2
	1x Meine Erste Umfrage (enthaltene Stimme)	-	-
Total 2 = 100% => 2.00 = x			100%
Gesamtergebnis auf alle Fragen bezogen			<u>100%</u>

8.6 Fazit aus den Testergebnissen

Als die Usability Tests von der OdA G Zürich online geschaltet worden sind, konnten innerhalb von 1-2 Tagen die Feedbacks der Studenten ausgewertet werden. Viele gestellte Fragen geben Aufschluss, wie die Webapplikation bei der OdA G ZH eingesetzt wird. Einzelne Usability Fragen führen aber wieder zu neuen Fragen. Zum Beispiel, erhielten alle aufgeforderten Teilnehmer den Zugriff zur Umfrage über eine E-Mail. Bei den Teilnehmern, welche am Usability Test nicht mitgemacht haben, stellt sich schnell die Frage, weshalb diese nicht an der Usability Umfrage teilgenommen haben. Erhielten sie keine E-Mail? Wollten sie nicht teilnehmen? Jedenfalls konnten wir den erfolgreichen Versand aller E-Mails serverseitig in den Log-Dateien bestätigen.

Da es sich bei den Testpersonen um Studenten im Bereich Medizin handelt, testeten wir vor allem die Durchführung der Umfrage.

Zudem wurde die Usability Test parallel zu einer weiteren Umfrage gestartet. Die Teilnehmer waren bei beiden Umfragen dieselben. Jedoch nahmen bei den Usability Tests nicht so viel Teilnehmer teil wie beim Ethik Kurs:

Titel	Status	Teilgenommen / Total	Start Datum
Usability Test - Juni14	LAEUFT	6 / 15	04.06.2014
Ethik 2 Sem FS2014	ABGESCHLOSSEN	12 / 14	05.06.2014

Abbildung 115 Vergleich Teilnehmer Ethik und Usability Test

Aus den Bewertungen konnte die vereinbarte Mindestprozentzahl von 60% erreicht werden, damit die Applikation als benutzerfreundlich gilt. Jedoch ist die Auswertung mit Vorsicht zu geniessen, da sich bei den Testpersonen um eine geringe Anzahl von Studenten handelt. Im Nachhinein, hätten wir vielleicht bessere Fragen für die Studierenden ausdenken können, um aussagekräftigere Auswertungen zu erhalten.

Leider nahm die Leiterin der OdA G ZH, trotz mehrerer Aufrufe erst zwei Tage vor Abgabe an den Usability Tests teil. Daher reichte die Zeit nicht um das Feedback detailliert zu analysieren und die Problembereiche mit der Auftragsgeberin genauer zu besprechen, geschweige davon, noch Verbesserungen zu implementieren. Das Feedback befindet sich im Anhang. Aus dem Feedback ist ersichtlich, dass die Leiterin erfolgreich an der Umfrage teilnehmen konnte. Sie erhielt eine E-Mail mit Ihrem Umfragetoken. Damit konnte sie schnell auf die Umfrage zugreifen. Die Meldungen der Applikation sind für sie verständlich und sie kann diese immer dem auslösenden Vorgang zuordnen. Bei der Bedienung der Applikation treten keine Probleme auf und ist für sie verständlich. Die Auswertung und Korrelation der Fragen könnten übersichtlicher gestaltet werden und sollten bearbeitbar sein.

9. Verbesserungsmöglichkeiten

Die Systemtests zeigen unsere kritische Sichtweise und Einschätzung der umgesetzten Use Cases. Wie bereits schon im Kapitel 3.1 *Systemtest* erwähnt worden ist, können folgende Verbesserungen gemacht werden:

- Statt Umfrage über eine Excel-Datei zu erstellen, könnte die Umfrage auch direkt in der Webapplikation erstellt und verwaltet werden.
- Die Umfrage, wie auch die Auswertung könnten nachträglich bearbeitbar sein.
- Ein anderes Feature könnte eine Art Vorschau der Umfrage sein.
- Weitere Dateiformate als Export anbieten. Vielleicht kann der Semester Report als Word automatisiert werden.
- Statt die Session Tokens im Memory zu speichern, können dies in der Datenbank abgelegt werden.

Da wir uns vor allem auf die Kompatibilitätstests und Systemtests konzentrierten, kann die Testabdeckung durch Unit Tests verbessert werden, indem die REST-Schnittstelle besser abgedeckt wird.

Durch den Usability Test ist ersichtlich geworden, dass ein grosser Teil der Studenten die Umfrage über ein Smartphone ausfüllen. Dies bedeutet, dass grosse Bereiche der mobilen Geräte unterstützt werden müssen. Dies müsste laufend weitergeführt werden, weil wir während dieser Arbeit nicht alle Geräte abdecken können. Zudem könnten die JavaScript-Dateien während der Übertragung komprimiert werden um die Performance zu verbessern.

Insgesamt entstand eine Webapplikation, die ausgiebig getestet wurde. Aus Zeitgründen konnten nicht alle Bereiche endgültig ausgebaut werden. Zudem lieferten wir der Leiterin der OdA G ZH jederzeit Support, falls Probleme auftauchten. Die Webapplikation läuft stabil und stellt der OdA G ZH ein praktisches Tool für ihre Qualitätssicherung dar.

10. Abbildungsverzeichnis

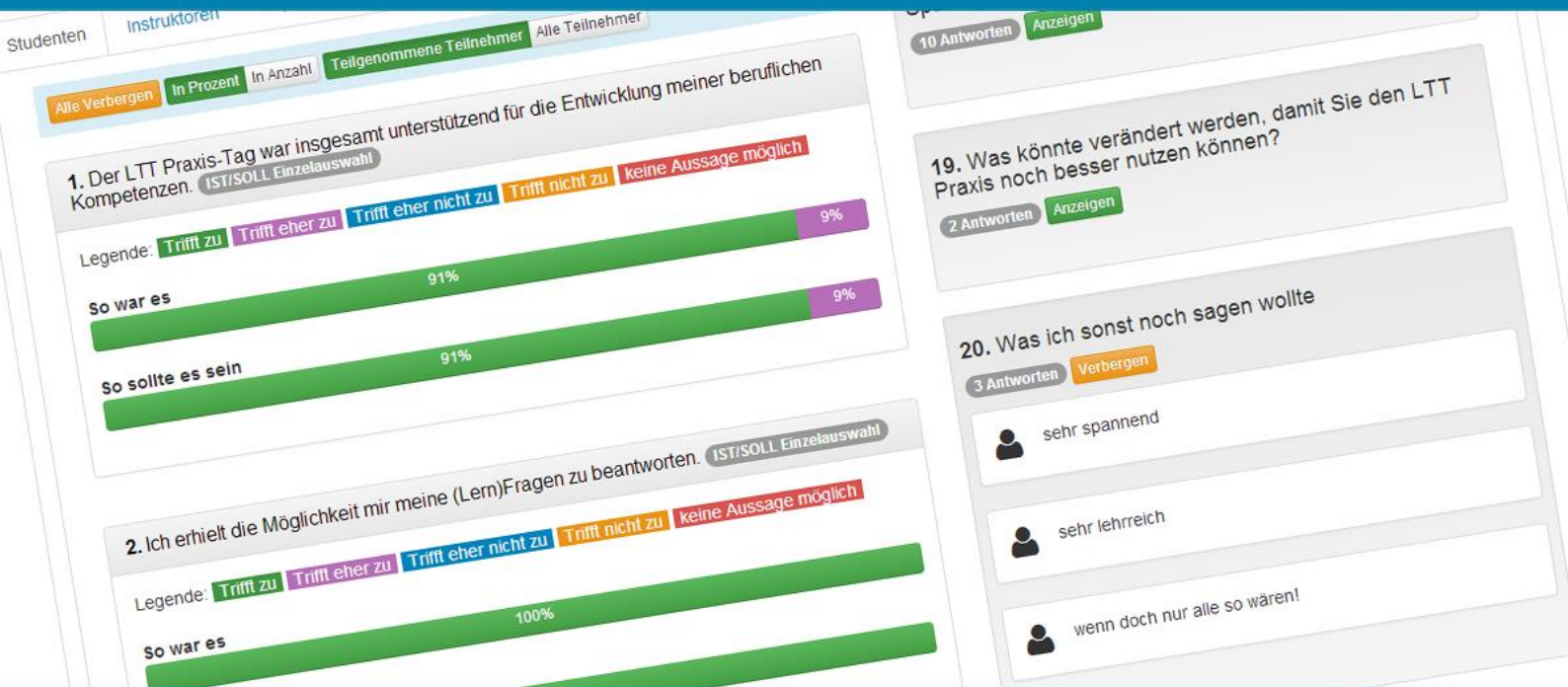
Abbildung 68 Test Frameworks	129
Abbildung 69 Package Struktur im Test Verzeichnis	130
Abbildung 70 Service Schicht	130
Abbildung 71 Test Runner AllTests.java	132
Abbildung 72 Stand April.....	132
Abbildung 73 Alle Tests bis Service Schicht (21.04.2014)	133
Abbildung 74: Test Coverage Stand 28.04.13	133
Abbildung 75: Pollution (Stand 21.04.2014)	134
Abbildung 76 CC durch Switch Anweisung.....	135
Abbildung 77 Reduzierung des CC-Wertes mit IF-Statements.....	135
Abbildung 78 Stand Juni	136
Abbildung 79 Stand 02.06.14	136
Abbildung 80 Erreichte Test Coverage 01.06.2014	137
Abbildung 81 Pollution Stand 01.06.2014.....	137
Abbildung 82 Metriken im /src/main/java.....	138
Abbildung 83 Distance Verteilung der Packages.....	139
Abbildung 84 Ladezeit Startseite Amsterdam, Holland	141
Abbildung 85 Ladezeit Startseite New York, USA.....	141
Abbildung 86 Ladezeit Umfrage Amsterdam, Holland	141
Abbildung 87 Ladezeit Umfrage New York, USA	142
Abbildung 88 Überblick der Kompatibilitätstests	143
Abbildung 89 TeachEval mit IE8 Support	144
Abbildung 90 IE10 Placeholder Support.....	144
Abbildung 91 IE8 – IE9 ohne Placeholder Bibliothek	144
Abbildung 92 IE8 – IE9 mit JQuery Placeholder Bibliothek	144
Abbildung 93 TeachEval mit IE9 Support	145
Abbildung 94 TeachEval mit IE10 Support	145
Abbildung 95 TeachEval mit Firefox 29.0.1 Support	146
Abbildung 96 TeachEval Chrome 35.0 Support.....	146
Abbildung 97 TeachEval Safari 5.1.7 Support	147
Abbildung 98 TeachEval Opera 21.0	147
Abbildung 99 Browsertest über Browserstack.....	148
Abbildung 100 iPhone 5 Abbildung 101 iPhone 4s	148
Abbildung 102 iPad 3 – Portrait Abbildung 103 iPad 3 – Landscape.....	149
Abbildung 104 Übersicht Usability Test	150
Abbildung 105 Resultate von Frage 1.....	151
Abbildung 106 Resultate von Frage 2.....	151
Abbildung 107 Resultate von Frage 3.....	152
Abbildung 108 Resultate von Frage 4.....	152
Abbildung 109 Resultate von Frage 5.....	153
Abbildung 110 Resultate von Frage 6.....	153
Abbildung 111 Resultate zur Frage 7	154
Abbildung 112 Resultate zur Frage 8	154
Abbildung 113 Resultate zur Frage 9	155
Abbildung 114 Resultate zur Frage 10	155
Abbildung 115 Vergleich Teilnehmer Ethik und Usability Test	158

11. Tabellenverzeichnis

Tabelle 49 getestete Services	131
Tabelle 50 Metriken	134
Tabelle 51 Violation.....	134
Tabelle 52 Metriken im /src/test/java	138
Tabelle 53 Violations /src/main/java	138
Tabelle 54 Übersicht Gewichtung der Antwortmöglichkeiten.....	156

TeachEval

Web App für die Unterrichtsevaluation



Projektauswertung (PAU)

Autor(en): Dominique Sorg, Benjamin Kehl
Betreuer: Prof. Dr. Eduard Glatz
Projektpartner: OdA G Zürich
Experte: Roberto Pajetta
Gegenleser: Prof. Dr. Olaf Zimmermann

1. Zweck

Dieses Dokument wertet und zeigt den Projektaufwand auf. Die Aufwände wurden aus Redmine exportiert und in einem Excel mit eigenen Diagrammen visualisiert.

2. Überblick

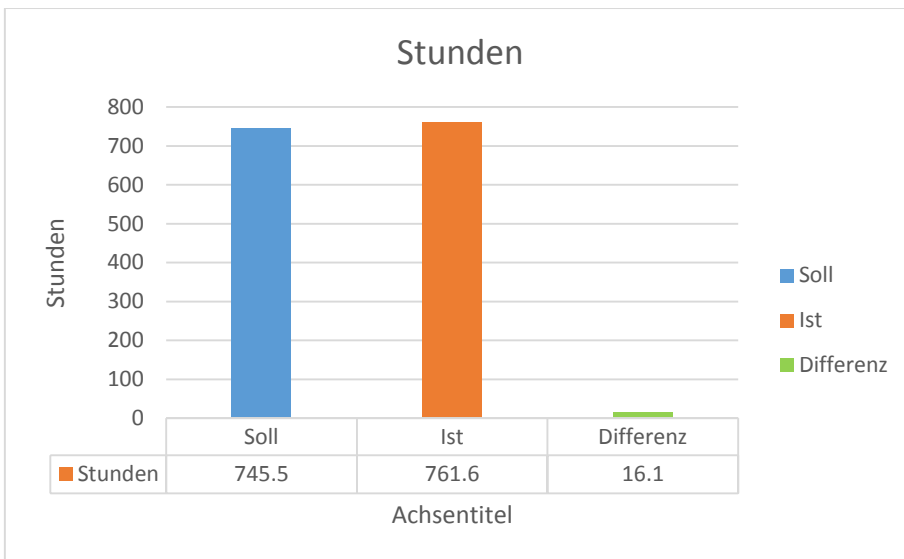


Abbildung 116 Zeitauswertung Überblick

Für diese Bachelorarbeit wurde 761.6 h aufgewendet. Dabei setzten wir uns, wie im Projektplan beschrieben, einen max. Zeitaufwand von $2 * 360 = 720$ h + 20% Abweichung. Daraus ergibt sich maximal 864 h, die nicht überschritten werden dürfen. Die Arbeitspakete wurden grosszügig definiert und jeweils am Freitag jeder Woche verbucht.

3. Ist / Soll-Vergleich

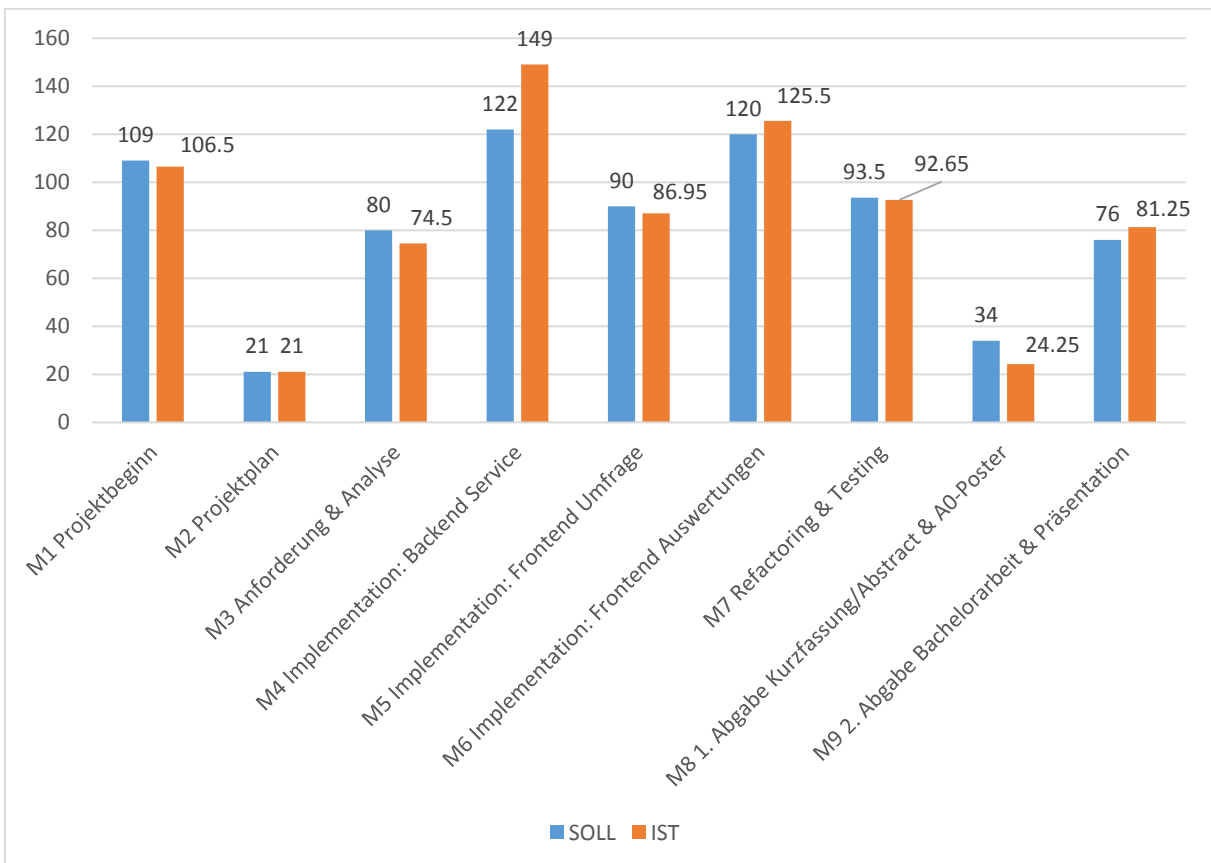


Abbildung 117 Zeitauswertung Meilensteine

Der IST-/SOLL-Vergleich visualisiert den Projektaufwand von TeachEval. Ab Projektbeginn (M1) starteten wir mit grosser Motivation und investierten viel Zeit in die Einarbeitung und Auswahl von Technologien. In der ersten Woche hatten wir das Kick-Off-Meeting mit unserem Betreuer Herrn Prof. Dr. Eduard Glatz. In der zweiten Woche fand das Kick-Off Meeting mit Frau Brigitte Keller von der Oda G ZH statt.

Der Projektplan und die Anforderungsanalyse wurden bereits in den ersten beiden Wochen begonnen. Somit wurde in den Meilensteinen *Projektplan (M2)* und *Anforderung & Analyse (M3)* im Verhältnis zu Meilenstein 1 (M1) weniger geplant.

Ab M4 begann aus den erarbeiteten Prototypen die richtige Entwicklung der Webapplikation. Die Erarbeitung des Datenbankmodells und die REST-Schnittstelle nahmen im Backend-Bereich sehr viel Zeit ein. Da ein erster Prototyp der Applikation bereits Anfang Mai live geschaltet werden musste, war ein gutes Datenbankmodell das A und O. Danach unterteilten wir das Frontend in zwei Meilensteine:

Im ersten Meilenstein (M5) ging es vermehrt um die Erarbeitung eines Layouts nach dem CI Styleguide der Oda G ZH und der Realisierung des Umfrage-Teils. Darunter werden die *Benutzerverwaltung*, die *Umfrageverwaltung* und die *Teilnahme an Umfragen* verstanden. Aus den Einarbeitungen konnten wir viele Elemente bereits wiederverwenden.

Im zweiten Meilenstein (M6) ging es anschliessend um die Auswertung der Umfragen. Natürlich wurde im Backend fortlaufend weiterentwickelt. Um eine qualitativ hochwertige Applikation anbieten zu können, wurde in M8 Zeit für Refactoring, Test, Verbesserungen, Erweiterungen und Support geplant. Die letzten Meilensteine befassen sich mit der Fertigstellung der Dokumente und den letzten Verbesserungen an der Applikation.

4. Team-Vergleich

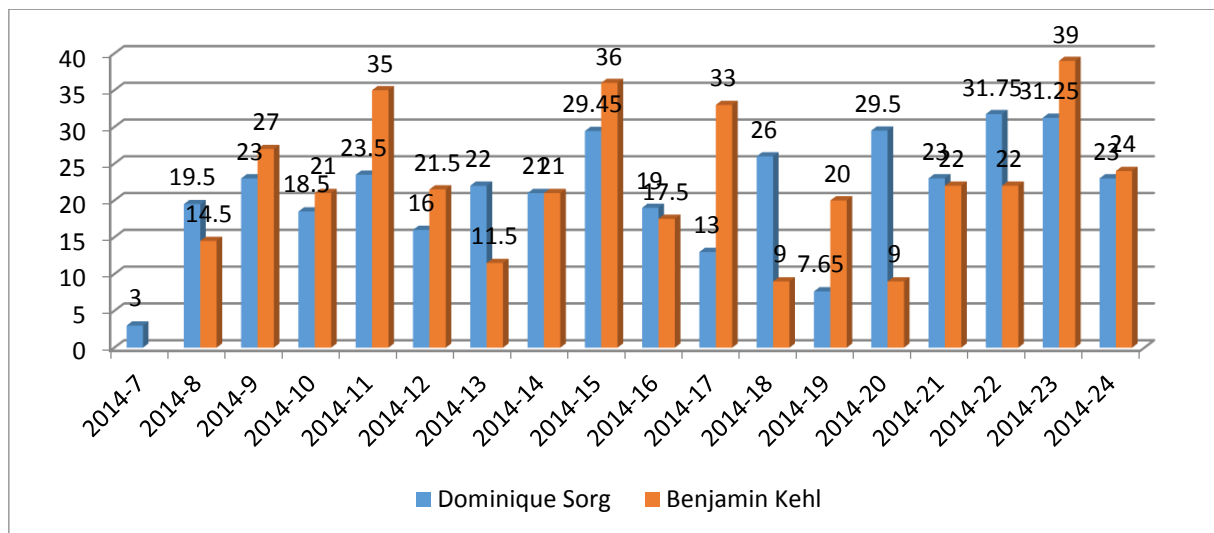


Abbildung 118 Zeitauswertung Team

Im oberen Diagramm wird der Zeitaufwand pro Teammitglied und pro Woche gezeigt.

5. Aktivitäten Vergleich

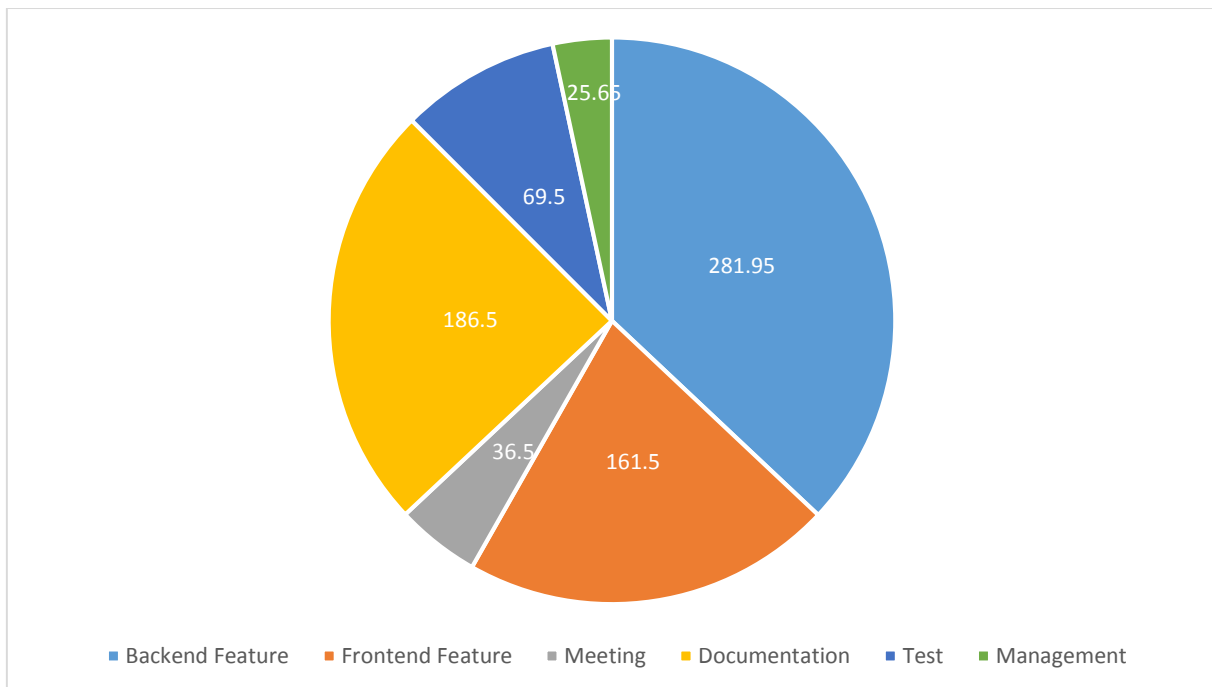


Abbildung 119 Zeitauswertung Aktivitäten

Die Arbeitspakete wurden in 6 Aktivitäten verbucht. Dabei nahm das Backend die meiste Zeit ein. Zu vermerken ist, dass im Backend Bereich das Schreiben von Tests auch als Implementation bzw. als Backend Feature (siehe Abbildung 119 Zeitauswertung Aktivitäten) verbucht wurde, da sich in manchen Situationen das Vorgehensmodell Test-Driven Development (TDD) sehr gut eignete. Das Frontend nahm im Vergleich zum Backend viel weniger Zeit in Anspruch, weil wir im Frontend die Daten entweder vom Server holen oder die Benutzereingaben an den Server direkt weiterleiten.

Die Dokumentation nahm vor allem die letzten zwei bis drei Wochen viel Zeit in Anspruch. Unter *Management* wird das Aufsetzen der Server oder die Planung verstanden. Dabei wurde die Dokumentation der Planung überwiegend als *Dokumentation* erfasst. Die Meetings zeigen den Zeitaufwand für die Sitzungen mit unserem Betreuer oder der Auftraggeberin auf.

6. Fazit

Für diese Bachelorarbeit wurde insgesamt 761.6 h aufgewendet. Durch die gesetzte Reservezeit, kamen wir mit der Arbeit nicht in Verzug und konnten bei jedem Meilenstein ein funktionstüchtiges Ergebnis liefern. Die Reservezeit wurde vor allem für das Testen und für Verbesserungen benötigt. Leider wurde unsere Webapplikation von der Oda G ZH nie richtig durchgetestet. Somit erhielten wir wenig Feedback, was dazu führte, dass wir laufend mehr Integrationstests, Kompatibilitätstests und Systemtests unsererseits durchführten, was zusätzliche Zeit kostete. Gefundene Fehler wurden direkt behoben und unmittelbar aufgeschaltet.

Die Differenz zwischen den geleisteten Aufwänden (Abbildung 116 Zeitauswertung Überblick) kamen im Vergleich zur Planung aufgrund des Backend zu Stande (Abbildung 117 Zeitauswertung Meilensteine Meilenstein 4). Da wir Hibernate und Jersey das erste Mal eingesetzt haben, nahmen diese mehr Zeit in Anspruch als wir dachten.

7. Abbildungsverzeichnis

Abbildung 116 Zeitauswertung Überblick.....	164
Abbildung 117 Zeitauswertung Meilensteine	164
Abbildung 118 Zeitauswertung Team	165
Abbildung 119 Zeitauswertung Aktivitäten.....	166