

Administration von E-OSCE Prüfungen

Bachelorarbeit

Abteilung Informatik
Hochschule für Technik Rapperswil

Frühjahrssemester 2011

Autoren: Matthias Good, Philipp Marugg und Sven Lenz
Betreuer: Prof. Hans Rudin
Projektpartner: Institut für Medizinische Lehre der Universität Bern
Gegenleser: Prof. Beat Stettler

Inhaltsverzeichnis

1	Abstract	6
1.1	Ausgangslage	6
1.2	Vorgehen und Technologien	6
1.3	Resultate	6
2	Management Summary	7
2.1	Ausgangslage [00]	7
2.2	Vorgehen / Technologien	8
2.3	Ergebnisse	8
2.4	Ausblick	9
3	Erklärung	10
4	Technischer Bericht	11
4.1	Einleitung	11
4.2	Ergebnisse	11
4.2.1	Erreichte Ziele	11
4.2.2	Erklärung	12
4.2.3	Nicht erreichte Ziele	12
4.2.4	Ursachen	12
4.3	Schlussfolgerung	12
5	Analyse	14
5.1	Anforderungsspezifikation	14
5.1.1	Einführung	14
5.1.2	Personas	15
5.1.3	Soll-Scenarios	16
5.1.4	Use Cases	18
5.1.5	Nicht-funktionale Anforderungen	23
5.2	Domainanalyse	25
5.2.1	Domain Model	25
5.3	Technologieentscheid	27
5.3.1	Technologievorschlge	27
5.3.2	Beschluss	28
5.5	Paperprototyping	29
5.6	Heuristische Evaluation	34
5.6.1	Erkenntnisse heuristische Evaluation „E-OSCE Admin Tool	34
6	Software Architektur Dokument	37
6.1	Einführung	37
6.1.1	Zweck	37
6.1.2	Gltigkeitsbereich	37
6.1.3	Übersicht	37
6.2	Gesamtarchitekturübersicht	38
6.2.1	Komponenten	38
6.2.2	Schnittstellen	39
6.3	Strategien und Konzepte	40
6.3.1	Eingesetzte Technologien	40

6.3.2	Java EE Server	41
6.3.3	Datenbank	41
6.3.4	Logging	41
6.3.5	Internationalisierung	41
6.3.6	JSF und HTML Komponenten	41
6.3.7	Entwicklungsumgebung	41
6.3.8	Namenskonventionen	42
6.4	Logische Architektur	43
6.5	Klassendiagramme	45
6.6	Klassenbeschreibungen	48
6.6.1	eosceAdminApp	48
6.6.2	eosceAdminDaoInterface	51
6.6.3	eosceAdminDao	52
6.6.4	eosceAdminDomain	53
6.6.5	eosceAdminTest	54
6.7	Sequenzdiagramme	55
6.7.1	Sequenzdiagramm CRUD	55
6.7.2	Sequenzdiagramm „Plan Generieren“	56
6.7.3	Sequenzdiagramm „Export zum Prüfungsserver“	57
6.7.4	Sequenzdiagramm Monitor	58
6.8	Schnittstellenstruktur	59
6.8.1	Datenmodell	59
6.8.2	XML-Interface	66
6.8.3	Import/Export	67
7	Tests	68
7.1.1	Zweck	68
7.1.2	Gültigkeitsbereich	68
7.1.3	Übersicht	68
7.2	Integrations-/Performancetest Sprint 2 (und 3)	68
7.2.1	Testaufbau /-durchführung	68
7.2.2	Testreihe 1	69
7.2.3	Testreihe 2	69
7.2.4	Testreihe 3a	70
7.2.5	Testreihe 3b	70
7.2.6	Testreihe 4	70
7.2.7	Testreihe 5	71
7.3	Gesamtüberblick	72
8	Projektmanagement	73
8.1.1	Zweck	73
8.1.2	Gültigkeitsbereich	73
8.1.3	Übersicht	73
8.2	Projektorganisation	73
8.2.1	Projektmitarbeiter	73
8.2.2	Externe Schnittstellen	73

8.3	Managementabläufe	74
8.3.1	Projektkostenvoranschlag	74
8.3.2	Projektplan	74
8.3.3	Vorgehen nach Scrum & RUP	74
8.3.4	Besprechungen	74
8.4	Infrastruktur	74
8.4.1	Räumlichkeiten	74
8.4.2	Hardware	75
8.4.3	Software	75
8.4.4	Versionsverwaltung	75
8.4.5	Kommunikation	75
8.5	Qualitätsmassnahmen	75
8.5.1	Dokumente	75
8.5.2	Sitzungen	75
8.5.3	Heuristische Evaluation	75
8.5.4	Tests	76
8.6	Risikomanagement	77
8.6.1	Risiken	77
8.6.2	Hinzugefügte Risiken	79
8.6.3	Eingetretene Risiken und Massnahmen	79
9	Benutzerdokumentation	80
9.1	Installation	80
9.1.1	Systemanforderungen	80
9.1.2	Installation der Datenbank	80
9.1.3	Installation der E-OSCE-Adminkomponente	81
9.2	Erste Schritte	83
9.2.1	Starten der E-OSCE-Adminkomponente	83
9.2.2	Login	83
9.2.3	Startseite	83
9.3	Editor	83
9.3.1	Prüfungsbogen	83
9.3.2	Prüfungseditor	84
9.4	Basisdaten	87
9.4.1	Posten	87
9.4.2	Examinatoren	88
9.4.3	Kandidaten	89
9.4.4	Infrastruktur	91
9.5	Planer	91
9.5.1	Prüfungsplaner	92
9.5.2	Prüfungswizard	92
9.5.3	Prüfungsstatus	102
9.6	Monitor	103
9.6.1	Prüfung überwachen	103
9.6.2	Prüfung pausieren	103
9.6.3	Prüfung stoppen	104

9.6.4	Monitor auf dem Smartphone	104
10	Zeitauswertung	105
11	Erfahrungsberichte	106
11.1	Matthias Good	106
11.1.1	Allgemein	106
11.1.2	Ablauf des Projektes	106
11.1.3	Teamarbeit	106
11.1.4	Fazit	106
11.1	Sven Lenz	107
11.1.1	Allgemein	107
11.1.2	Ablauf des Projektes	107
11.1.3	Teamarbeit	107
11.1.4	Fazit	107
11.1	Philipp Marugg	108
11.1.1	Allgemein	108
11.1.2	Ablauf des Projektes	108
11.1.3	Teamarbeit	108
11.1.4	Fazit	108
12	Glossar	109
13	Literaturverzeichnis	110
13.1	Kapitel 2 (Management Summary)	110
13.2	Kapitel 5 (Analyse)	110
13.3	Kapitel 6 (Software Architektur)	110
13.4	Kapitel 8 (Projektmanagement)	110
13.1	Kapitel 9 (Benutzerdokumentation)	110
14	Anhangsverzeichnis	111
15	Tabellenverzeichnis	112
16	Abbildungsverzeichnis	113

1 Abstract

1.1 Ausgangslage

Das Institut für Software der HSR und das Institut für Medizinische Lehre der Universität Bern führen gemeinsam das SWITCH Projekt E-OSCE durch, welches das Ziel hat, die sogenannten Objective Structured Clinical Exams (OSCE) elektronisch durchzuführen. Die bisher auf Papier vorgenommene Bewertung der Kandidaten wird in Zukunft elektronisch über ein Mobile-Gerät von einem Examinator vorgenommen.

Diese Bachelorarbeit beinhaltet die Adminkomponente, welche das Interface zwischen dem Prüfungsverantwortlichen und dem bestehenden Prüfungsserver bildet.

1.2 Vorgehen und Technologien

Die Adminkomponente vereinfacht dem Verantwortlichen die Prüfungsvorbereitung durch eine elektronische Erfassung der Basisdaten (Examinatoren, Kandidaten, Posten und Geräte) sowie der Prüfungsformulare und Prüfungen.

Während der Prüfungsdurchführung erlaubt ein Monitor die Überwachung der Prüfung und gegebenenfalls kann eine Prüfung über den Monitor pausiert oder sogar gestoppt werden. Eingehende Fehler werden akustisch und visuell signalisiert.

Eine abgeschlossene Prüfung kann in der Datenbank der Adminkomponente für eine spätere Auswertung abgespeichert und jederzeit heruntergeladen werden.

Die Lösung ist als modulare Enterprise Application mit einem webbasierten User Interface entwickelt. Dieser Ansatz stellt sicher, dass die Businesslogik getrennt vom Client weiterentwickelt und getestet werden kann.

1.3 Resultate

Mit dieser Bachelorarbeit erhält der Auftraggeber eine Applikation zur Unterstützung des Projektes E-OSCE, mittels welcher ein Prüfungsverantwortlicher ohne erheblichem Zeit- und Organisationsaufwand eine Prüfung vorbereiten, durchführen und auswerten kann.

Dabei werden dem Verantwortlichen Werkzeuge wie ein Wizard, Drag & Drop und Auto-complete zur Verfügung gestellt, welche einen optimalen Komfort der Arbeit sicherstellen.

2 Management Summary

2.1 Ausgangslage [00]

An der Universität Bern werden in der medizinischen Abteilung jedes Jahr sogenannte OSCE-Prüfungen durchgeführt. OSCE ist die Abkürzung für "Objective Structured Clinical Examination", dabei handelt es sich um praktische Tests, bei welchen die Fähigkeiten der angehenden Ärzte geprüft werden.

Das Institut für medizinische Lehre (IML) der Universität Bern ist für die Vorbereitung der Prüfungen sowie deren Durchführung und Überwachung zuständig. Zusammen mit dem Institut für Software (IFS) Rapperswil haben sie das Projekt "E-OSCE" (also Electronic OSCE) gestartet. Bisher werden die OSCE-Prüfungen mittels papiergebundenem Prozess durchgeführt, d.h. die Prüfungsformulare werden einzeln am Computer geschrieben und danach den Experten, welche die Studenten bewerten, gedruckt übergeben. Diese Bewertung erfolgt ebenfalls von Hand. Die Ergebnisse werden dann wieder in den Computer eingegeben, wo sie dann korrigiert werden und die Resultate analysiert werden können. Das folgende Diagramm zeigt diesen Prozess auf:



Abbildung 1 - OSCE Prozess

Die Zeit bis die Prüfungsergebnisse veröffentlicht werden können beträgt 26 Tage, also knapp ein Monat. Dieser Prozess ist auch fehleranfällig, sowohl bei der Bewertung wie auch der Digitalisierung können Fehler entstehen, bis zu 50% der Prüfungen müssen von Hand nachbearbeitet werden. Zudem wird die Anzahl an OSCE-Prüfungen in den nächsten Jahren weiter ansteigen.

Diese Mängel will man mit dem Projekt E-OSCE beheben, wie das folgende Diagramm zeigt:



Abbildung 2 - E-OSCE Prozess

Durch den effizienteren elektronischen und weniger fehleranfälligen Bewertungsprozess entfallen mehrere kostenintensive Schritte und die Veröffentlichungszeit der Prüfungsergebnisse verringert sich auf 12 Tage. Dieser papierlose Prozess wird durch Tablets realisiert, mit welchen die Experten die Studenten bewerten können.

Das Projekt ist in drei Phasen gegliedert:

- Phase 1: (Dez 08 - Mai 09) iPhone Prototyp zur Erfassung der Resultate von medizinischen Assessments.
- Phase 2: (Aug 09 - Juli 10) effizientes und benutzerfreundliches Pilotsystem für papierlose Prüfungen und Prüfung der Machbarkeit.
- Phase 3: (Aug 10 - Okt 11) Produktiver Betrieb des E-OSCE-System.

Diese Arbeit befasst sich dabei nur mit der dritten Phase des Projekts. Dabei soll eine Adminkomponente (Prüfungs-Administration) erstellt werden, mit welcher die Prüfungsverantwortlichen die folgenden Aufgaben erledigen können:

- Elektronische Erfassung von Prüfungsformularen
- Planung einer Prüfung mithilfe der Erfassung von Stationen, Kandidaten und Experten
- Export der Prüfung zur Durchführung auf den Prüfungs-Server
- Zusammenstellung der Prüfungsergebnisse für die Korrektur und Analyse

Das folgende Diagramm gibt einen kleinen Überblick das gesamte E-OSCE-Projekt:

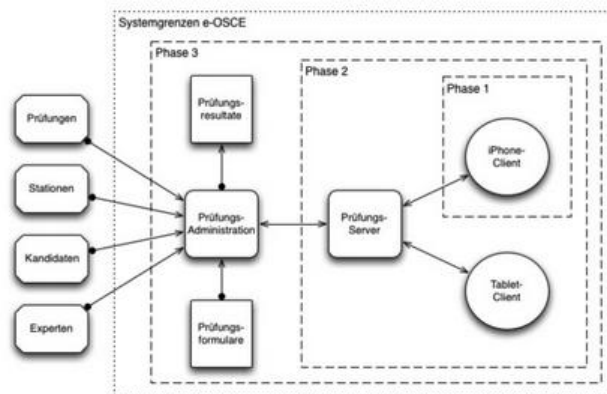


Abbildung 3 - Gesamtüberblick E-OSCE

2.2 Vorgehen / Technologien

In einer ersten Phase wurde ein Paperprototype erstellt, welcher zusammen mit dem Be-treuer-team analysiert wurde. Zudem wurde eine heuristische Evaluation mit zwei Experten durchgeführt, um frühzeitig mögliche Usability-Probleme zu beheben. Nach dieser Analyse-Phase folgte die Implementation des Prototypes, welche sich in folgende Aufgabengebiete gliedern lässt:

- Basisdaten: Import, resp. manuelle Eingabe von Daten (Stationen, Kandidaten, Experten)
- Prüfungseditor: Erfassen von Prüfungsformularen
- Prüfungsplaner: Erstellen von kompletten Prüfungen
- Monitor: Überwachung einer momentan durchgeführten Prüfung

Die E-OSCE-Adminkomponente wurde als Java Enterprise Application mit einem JSF-Frontend entwickelt, welche sich im Browser starten lässt. Als Ergänzung zu JSF wurde mit der Primefaces-Library gearbeitet.

Als Vorgehensmodell bei der Implementation wurde Scrum gewählt, dabei wurde die Arbeit in fünf Sprints von jeweils einer Dauer von zwei Wochen aufgeteilt. Die Dokumentation wurde dazu synchron nach dem RUP-Vorgehensmodell erstellt.

2.3 Ergebnisse

Die E-OSCE-Adminkomponente reduziert den Aufwand des Prüfungsprozesses enorm. Das Erstellen von Prüfungsformularen kann in wenigen Schritten durchgeführt werden, die Daten, wie z.B. die Kandidaten oder Experten, können direkt in das System importiert werden. Der Prüfungsverantwortliche wird bei der Planung der Prüfung durch einen Wizard, bei dem er Schritt für Schritt die benötigten Daten eingeben kann. Zusätzlich erhält er am Schluss eine Übersicht über die komplette Prüfung, wobei dort alle Komponenten mittels Drag & Drop untereinander ausgetauscht werden können.

E-OSCE

Electronic Observed Structured Clinical Examinations.

Prüfungsplaner
Prüfungseditor
Basisdaten
Logout

Informationen
Posten
Kandidaten
Parcours
Rotationen

Geben Sie einen Prüfungsnamen an und legen Sie das Startdatum der Prüfung fest. Falls die Anzahl der verfügbaren Prüfungsräume bekannt sind, können Sie diese unter "Anzahl Räume" eintragen. (Anhand der Anzahl Räume kann die optimale Anzahl an Rotationen berechnet werden)

Rotation hinzufügen

Examinatoren
Kandidaten
Formulare

Rotation 1		Rotation 2		Rotation 3	
10.06.2011 11:11 - 11:20		10.06.2011 11:30 - 11:39		10.06.2011 11:45 - 11:54	
Parcours 1	PHY Bob Brühler Dr. HNO Cox Perry Dr. KPW Müller Toni Prof. BREAK CBA 1 	Parcours 3	PHY Koch Peter Dr. HNO Kein Examinator ausgewählt KPW Kelso Bob Dr. BREAK CBA 1 	Parcours 6	PHY Bob Brühler Dr. HNO Cox Perry Dr. KPW Müller Toni Prof. BREAK CBA 1
Parcours 2	PHY Koch Peter Dr. HNO Stinson Barney Dr. KPW Kelso Bob Dr. BREAK CBA 1 	Parcours 7		BREAK HNO Kein Examinator ausgewählt KPW Kein Examinator ausgewählt PHY Koch Peter Dr. CBA 1 	
		Parcours 4	PHY Bob Brühler Dr. HNO Cox Perry Dr. KPW Müller Toni Prof. BREAK CBA 1 		

Fertig

Abbildung 4 - Übersicht über komplette Prüfung

Der Monitor benachrichtigt bei auftretenden Problemen oder Fehlern bei einer momentan durchgeführten Prüfung unverzüglich den Prüfungsverantwortlichen, sodass darauf frühzeitig reagiert werden kann. Der Monitor kann problemlos auch auf einem Smartphone gestartet werden, da er sich automatisch der Displaygröße anpasst.

2.4 Ausblick

Der papiergebundene Prozess besitzt, wie bereits erwähnt, viele Nachteile, welche durch E-OSCE behoben werden. Doch der neue elektronische Prozess birgt auch Gefahren: die Bedienbarkeit der Adminkomponente muss einfach zu erlernen sein und der neue Arbeitsvorgang die alte Vorgehensweise in allen Bereichen verbessern, ansonsten wird man den neuen Prozess dem Alten nicht vorziehen wollen.

Die zweite Phase des Projekts neigt sich zu diesem Zeitpunkt dem Ende zu, sodass die dritte Phase, welche die Ergebnisse dieser Arbeit enthält, in naher Zukunft in einem ersten Probe-Durchlauf durch das IML getestet werden kann. Dabei wird man sehen, wie gut der elektronische Prozess gegenüber dem papiergebundenen Prozess abschneidet.

3 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.

Matthias Good, Rapperswil, den 16. Juni 2011



Philipp Marugg, Rapperswil, den 16. Juni 2011



Sven Lenz, Rapperswil, den 16. Juni 2011



4 Technischer Bericht

4.1 Einleitung

Das Ziel dieser Bachelorarbeit war eine Applikation zu entwickeln, welche einem Prüfungsverantwortlichen die Administration von E-OSCE Prüfungen erleichtern soll. Das wichtigste Ziel war, ein einfaches und intuitiv zu bedienendes User Interface für die Phasen Prüfungsvorbereitung, Prüfungsdurchführung und Prüfungsauswertung zu erstellen.

Das User Interface wurde in enger Zusammenarbeit mit dem Institute für Software der HSR entworfen und durch eine heuristische Evaluation auf die Benutzbarkeit überprüft.

4.2 Ergebnisse

4.2.1 Erreichte Ziele

Prüfungsvorbereitung

- Prüfungsverantwortliche sollen für die Prüfung relevante Daten über Kandidaten und Examinatoren (Ärzte) aus einem bestehenden System importieren können.
- Prüfungsverantwortliche sollen die Posten einer Rotation erfassen können.
- Prüfungsverantwortliche sollen die Examinatoren den jeweiligen Posten einer Rotation zuweisen können.
- Prüfungsverantwortliche sollen die Kandidaten zunächst auf die verschiedenen Rotationen und danach auf die verschiedenen Posten innerhalb einer Rotation aufteilen können.
- Prüfungsverantwortliche sollen Pausenposten in Rotationen planen können.
- Prüfungsverantwortliche sollen Kandidatenzuteilungen noch bis kurz vor Start einer Prüfung verändern können.
- Prüfungsverantwortliche sollen bestehende papiergebundene Prüfungsbogen in elektronische umwandeln können, welche von den E-OSCE Server und Client Komponenten unterstützt werden. Dazu ist ein Formulareditor zu entwickeln.
- Prüfungsverantwortliche sollen elektronische Prüfungsbogen bis kurz vor Start einer Prüfung verändern können.
- Prüfungsverantwortliche sollen Unterstützung beim Aufbau des Prüfungsnetzwerks und der Konfiguration der Clients erhalten.
- Prüfungsverantwortliche sollen Keycodes (der iPads) für die Examinatoren verwalten können.

Prüfungsdurchführung

- Prüfungsverantwortliche sollen jederzeit den Status der Prüfung abrufen können.
- Prüfungsverantwortliche sollen bei Problemen von Server oder Clients umgehend informiert werden.
- Prüfungsverantwortliche sollen Rotationen starten und beenden können.
- Prüfungsverantwortliche sollen Rotationen pausieren und neu ansetzen können.

Prüfungsauswertung:

- Prüfungsverantwortliche sollen alle Daten der Prüfung in geeignetem Format exportieren können.

- Prüfungsverantwortliche sollen nach dem Export der Prüfungsdaten alle Daten vom Server löschen können.

4.2.2 Erklärung

Mittels der E-OSCE Adminkomponente kann ein Prüfungsverantwortlicher zu Beginn der Prüfungsvorbereitung alle benötigten Basisdaten für eine anstehende Prüfung erfassen. Dies beinhaltet, dass Kandidaten, Examinatoren, Geräte und Posten, welche während der Prüfung eingesetzt werden auf der Datenbank abgespeichert und auch wieder geändert werden können. Nach der Erfassung der Basisdaten können über einen Formulareditor die einzelnen Prüfungsbogen für die Posten erstellt werden. Anhand der Basisdaten und Formulare kann der Prüfungsverantwortliche über einen Wizard, der ihn Schritt für Schritt durch die Prüfungsvorbereitung führt, eine Prüfung erstellen. Algorithmen, wie zum Beispiel das Berechnen der optimalen Anzahl an Parcours oder die automatische Zuweisung von Startkandidaten an die Posten, unterstützen dabei den Verantwortlichen bei der Planung der Prüfung.

Für die Phase der Prüfungsvorbereitung ist zudem ein Export der Daten auf den Prüfungsserver entwickelt worden. Aus den im Admin-Tool erfassten Daten werden die benötigten XML-Strukturen erstellt, mittels dem Admin-Zertifikat signiert und über eine geschützte HTTPS-Verbindung an den Prüfungsserver übermittelt.

In der Phase der Prüfungsdurchführung ist ein Monitor entwickelt worden, welcher dem Verantwortlichen bereits eine halbe Stunde vor der Prüfung eine Überwachung der Aktivitäten ermöglicht. Während der Prüfungsdurchführung werden entdeckte Fehler dargestellt und bei dringenden Problemen kann die Prüfung pausiert oder gegebenenfalls gestoppt werden. Zudem ist ein Import der Prüfungsergebnisse in die Admin-Datenbank implementiert worden. Anschliessend kann via Admin-Tool ein Zip-File mit allen XML-Files der Prüfung für eine spätere Auswertung in einem externen System heruntergeladen werden.

4.2.3 Nicht erreichte Ziele

- Prüfungsverantwortliche sollen elektronische Prüfungsbogen austauschen können (Import/Export).
- Prüfungsverantwortliche sollen Ausnahmesituationen behandeln können (z.B. Auswechseln Experte).

4.2.4 Ursachen

Da die Entwicklung des User Interface basierend auf einer Web-Applikation Schwierigkeiten, wie zum Beispiel die komplexe Drag & Drop Ansicht für die Planung der Rotation, bereitet hat, mussten die in Kapitel 4.2.3 aufgeführten Punkte in Absprache mit dem Betreuer-Team tiefer Priorisiert werden. Man einigte sich darauf, dass so viele Features wie möglich noch implementiert werden.

Der Austausch der elektronischen Prüfungsbogen konnte aus zeitlichen Gründen gar nicht implementiert werden. Jedoch besteht zu diesem Ziel der Workaround, die Formulare aus der Datenbank zu exportieren und in einem anderen System wieder einzuspeisen.

Das Auswechseln eines Experten soll während einer Prüfungsdurchführung nicht möglich sein. Das Ziel wurde zusammen mit dem Betreuer Team im Nachhinein geändert.

4.3 Schlussfolgerung

Die E-OSCE Adminkomponente ist soweit vorbereitet, dass alle im Product Backlog [ANH06] mit Priorität 1 oder 2 gekennzeichneten Ziele implementiert sind. Die Ziele mit der Priorität 3 sind soweit wie möglich implementiert. Die Adminkomponente ist für den Einsatz bei E-OSCE-Prüfungen optimiert, sie kann jedoch auch für andere Prüfungen verwendet werden.

Dem Betreuerteam wurden im letzten Sprint die Funktionalitäten der Adminkomponente präsentiert und eine Probeprüfung auf dem Prüfungsserver durchgeführt. Das Feedback darauf ist sehr positiv ausgefallen.

5 Analyse

5.1 Anforderungsspezifikation

Rev.	Datum	Wer	Änderung
0.1	14.03.2011	Matthias Good	Dokument erstellt
0.2	15.03.2011	Matthias Good	Erste Bearbeitung aller Themen
0.3	13.06.2011	Matthias Good	Review des Dokumentes

5.1.1 Einführung

5.1.1.1 Zweck

Dieses Dokument analysiert die Anforderungen und zeigt die Ergebnisse auf.

5.1.1.2 Gültigkeitsbereich

Das Dokument ist für die komplette Bachelorarbeitsdauer gültig

5.1.1.3 Übersicht

Die Anforderungsspezifikation beschreibt die wesentlichen Punkte der funktionalen sowie nicht-funktionalen Anforderungen an die Applikation. Zu Beginn werden zwei Personas vorgestellt, welche in die folgenden Soll-Szenarios eingebunden sind. Das Use-Case Model gibt einen Überblick über Interaktionen mit dem System. Die Domain wird mittels eines Domain Model dargestellt.

Die Analyse beinhaltet zudem den Technologieentscheid, die Paperprototypes sowie die heuristische Evaluation, welche anhand der Paperprototypes durchgeführt wurde.

5.1.2 Personas

5.1.2.1 Dr. Bob Brühler



Dr. Bob Brühler

63 Jahre

Verheiratet, keine Kinder

Medizinstudium
Seit 5 Jahren Chefarzt

Schlechte
Computerkenntnisse

Bob arbeitet seit mehr als 30 Jahre auf dem Beruf als Arzt. Er hat sich in den Jahren zu einem kompetenten Chefarzt entwickelt und ist nebenbei auch Dozent an der Universität für Medizin.

Dr. Bob Brühler gehört dem Gremium der Prüfungsverantwortlichen an. Er ist dabei mitunter für die Prüfungszusammenstellung verantwortlich. Nur widerwillig hat er sich vor ein paar Jahren dazu überreden lassen, die Prüfungsdaten nicht wie bisher mit Stift und Papier, sondern mittels Computer und Excel-Dateien zu bearbeiten. Bob hat sich mittlerweile an das elektronische Erstellen von Prüfungen gewöhnt, kann aber ansonsten dem Computer nichts Gutes abgewinnen. *„Mit meinem Kugelschreiber bin ich 1000x schneller als mit solch einer Kiste und zudem weiss ich dann auch, wo ich meine Papiere wiederfinde.“*

Bob arbeitet jeweils sehr exakt und gut organisiert, darum ist es ihm wichtig, dass die Prüfung gut strukturiert ist und mögliche Änderungen kurz vor Prüfungsbeginn kein grosses Problem darstellen.

Dr. Bob Brühler ist mit dem jetzigen Arbeitsvorgang zufrieden und ist gegenüber neuen Technologien skeptisch eingestellt. *„Neue Technologien bringen auch neue Probleme mit sich!“*

Abbildung 5 - Persona 1 [01]

5.1.2.2 Dr. Peter Koch



Dr. Peter Koch

45 Jahre

Geschieden, 2 Kinder

Peter arbeitet zusammen mit anderen Dozenten an der Universität. Er unterrichtet vor allem Module aus dem Gebiet der Inneren Medizin. Dr. Peter Koch ist sehr streng und erwartet von seinen Studenten so einiges. Von sich selber verlangt er jedoch am meisten, ist stets sehr engagiert und setzt sich für seine Schützlinge ein. *„Es wäre unfair, wenn ich nicht mehr leisten würde als die Studenten, schliesslich erhalte ich im Gegensatz zu ihnen dafür Lohn.“*

Nebenbei ist Peter im Gremium der Prüfungsverantwortlichen Mitglied. Er arbeitet gerne mit den anderen Experten zusammen an den Prüfungen und leistet auch dort grossen Einsatz. Leider nimmt diese Zweitbelastung viel Zeit in Anspruch, vor allem der ganze Papierkram macht ihm zu schaffen. *„In der heutigen Zeit sollte man solche Vorgänge elektronisch erfassen können.“*

Peter, selbst immer perfekt organisiert, ist ein wenig unzufrieden mit der Übersicht während der Prüfungsdurchführung.

Medizinstudium Oberarzt und Dozent	<i>„Wenn es bei einem Examiner Probleme gibt, dauert es eine halbe Ewigkeit bis man dies als Verantwortlicher mitbekommt und reagieren kann.“</i>
---------------------------------------	---

Abbildung 6 - Persona 2 [02]

5.1.3 Soll-Scenarios

5.1.3.1 Prüfung planen

Die nächsten OSCE-Prüfungen stehen in geraumer Zeit an und Dr. Peter Koch möchte nun den genauen Ablauf dieser Prüfungen planen. Dazu meldet er sich mittels Login bei der Adminkomponente von E-OSCE an und wählt im Planner-Tab die bevorstehende Prüfung aus. Da es sich nur um eine kleine Zwischenprüfung handelt, werden lediglich zwei Rotationen mit je zwei Parcours benötigt. Zuerst erstellt er die erste Rotation und fügt die dazu benötigten Angaben wie Name, Startzeit, Dauer und Pausenlänge ein. Nun erweitert Peter Koch die Rotation um die beiden Parcours blau und rot. Jedem der Parcours teilt er die dafür vorgesehenen Posten zu. Die meisten Daten, welche er für einen Posten benötigt, sind bereits importiert worden, ein paar Examinatoren/Studenten muss er jedoch noch per Hand unter den entsprechenden Tabs erstellen. Für diejenigen Posten, für die noch keine Prüfungsbogen existieren, kann der Prüfungsverantwortliche diese noch zu einem späteren Zeitpunkt kreieren.

Nachdem er die erste Rotation abgeschlossen hat, geht er nun genau nach dem gleichen Prinzip für die zweite Rotation vor. Nach dem erfolgreichen Erstellen aller Rotationen inkl. Parcours für die Prüfung, sieht Peter Koch, dass noch nicht alle Posten einen Prüfungsbogen haben. Diese Aufgabe will er aber zu einem späteren Zeitpunkt erledigen.

5.1.3.2 Prüfungsrelevante Daten importieren

Nach der Anmeldefrist für die Herbst-Prüfung erstellt Dr. Bob Brühler ein Excel-Sheet mit den Namen und den Studentennummern der angemeldeten Studenten. Nachdem er auch die Zusagen der Examinatoren erhalten hat, fertigt er ebenfalls eine Liste mit den Namen der Examinatoren an. Diese beiden Listen importiert Bob Brühler in die Adminkomponente, wo er nun die Prüfung mit diesen Studenten und Examinatoren füllt.

5.1.3.3 Prüfungsbogen erstellen

Dr. Peter Koch möchte zwei Wochen vor den Prüfungen die noch fehlenden Prüfungsbogen erstellen. Dazu loggt er sich bei der Adminkomponente von E-OSCE ein und wählt den Editor-Tab aus. Er erstellt einen neuen Bogen und gelangt automatisch in den Editor-Modus. Nachdem er dem Prüfungsbogen den Namen "PSY v 1.0" gegeben hat, will er die erste Frage zusammenstellen. Dazu wählt er als Erstes das "Ja/Nein"-Template aus den Vorlagen aus und tippt die dazugehörige Frage ein. Bei der nächsten Frage handelt es sich um eine oft gestellte Frage, wie Peter Koch weiss, deshalb sucht er im Fragenpool nach "Freundliche Begrüssung" und wählt die gewünschte Frage aus, welche automatisch in das Formular übertragen wird. Er fährt nach diesem Schema weiter bis er den Prüfungsbogen komplettiert hat.

Peter Koch will die erstellten Prüfungsbogen den jeweiligen Posten hinzufügen. Dazu wählt im Planner-Tab die entsprechende Prüfung aus und sieht auf den ersten Blick die Posten, welche noch keinen Prüfungsbogen haben. Er fügt die fehlenden Bogen den entsprechenden Posten hinzu und beendet die heutige Arbeit indem er sich von der Adminkomponente abmeldet.

5.1.3.4 Prüfungsdurchführung überwachen

Die Frühlingssemester-Prüfung hat soeben begonnen. Der für diese Prüfung verantwortliche Dozent ist Dr. Bob Brühler. Er sitzt in seinem Büro und ist mit dem Korrigieren von Hausarbeiten beschäftigt. Neben ihm liegt sein Smartphone, auf welchem er sich in den Monitor-Client eingeloggt hat. Nach etwa 20 Minuten wird Bob Brühler durch ein akustisches Signal von seinem Smartphone darauf hingewiesen, dass der Examinator Josef Werter seit längerer Zeit inaktiv ist. Bob Brühler begibt sich zum Prüfungszimmer von Herrn Werter und wartet vor der Tür, bis der momentane Kandidat fertig ist. Als dieser Kandidat aus dem Zimmer herauskommt, kann Bob Brühler den Examinatoren nach seinem Problem fragen. Es stellt sich heraus, dass der Examinator vergessen hatte, seine Bewertungen einzugeben und holt dies jetzt nach.

Bob Brühler kann die Warnmeldung im Monitor-Client als "Behandelt" markieren und geht wieder zurück in sein Büro um weiterzuarbeiten.

5.1.3.5 Prüfungsdaten exportieren

Die Herbstsemester-Prüfung ist erfolgreich durchgeführt worden. Dr. Peter Koch will nun die Resultate zu sich auf den Computer speichern. Dazu meldet er sich bei der Adminkomponente an und sieht gleich in der Übersicht die erfolgreich beendete Prüfung. Er kann diese Prüfung nun als Zip-File herunterladen und es lokal auf dem Computer speichern. Dadurch hat Peter Koch zu jeder Zeit die Möglichkeit die Prüfung auszuwerten oder diese an eine andere Stelle weiterzugeben

5.1.4 Use Cases

5.1.4.1 Use Case Model

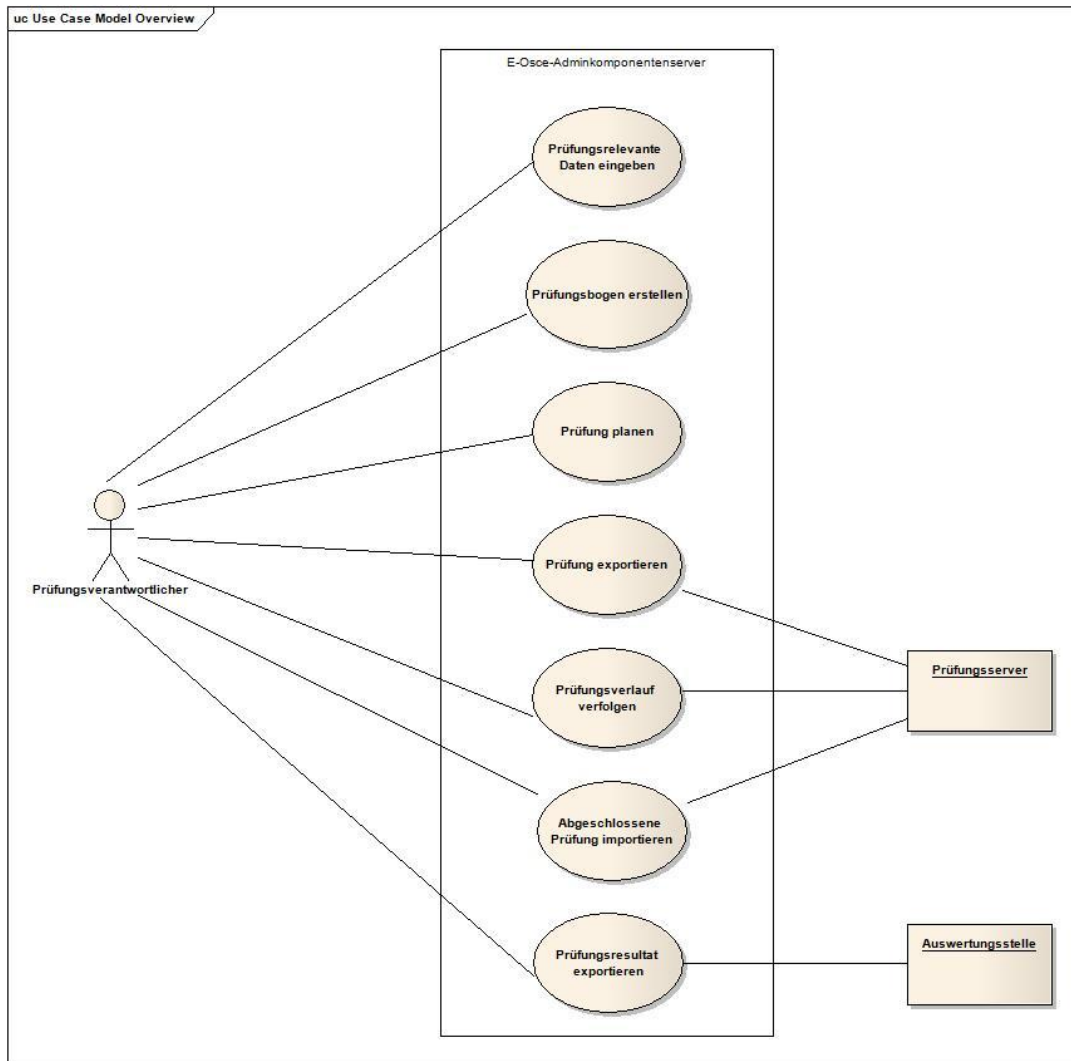


Abbildung 7 - Use Case Model

Prüfungsverantwortlicher: Wird durch eine (oder mehrere) Person verkörpert, welche für die Durchführung einer E-OSCE-Prüfung verantwortlich ist. Dazu gehören die Prüfungsvorbereitung (Dateneingabe, Prüfungsbogenerstellung und Planung), die –Überprüfung (Verlaufkontrolle) sowie der Export der Prüfung an eine Auswertungsstelle.

Prüfungsserver: Der Prüfungsserver kommuniziert einerseits mit dem E-OSCE-Adminserver und andererseits mit den iPads der Examinatoren. Der Server nimmt eine komplette Prüfung entgegen und stellt dem Adminserver Informationen zu einer laufenden Prüfung bereit

Auswertungsstelle: Der Auswertungsstelle werden die Resultate einer abgeschlossenen Prüfung zugestellt, welche diese dann auswertet.

Das Use Case Model besteht aus den folgenden Use Cases:

Use Case	Beschreibung
<i>UC01_Prüfungsrelevante Daten eingeben</i>	Der Prüfungsverantwortliche gibt Daten, welche für eine Prüfung relevant sind, ein. Dazu gehören die Kandidaten, die Examinatoren und die Geräte. Zusätzlich besteht für ihn die Möglichkeit, eine komplette Liste von Kandidaten resp. Examinatoren zu importieren.
<i>UC02_Prüfungsbogen erstellen</i>	Der Prüfungsverantwortliche erstellt mithilfe des Formulareditors einen Prüfungsbogen, welcher später auf dem iPad des Examinators dargestellt wird.
<i>UC03_Prüfung planen</i>	Der Prüfungsverantwortliche erstellt eine neue Prüfung und weist ihr die gewünschten Posten, Kandidaten und Examinatoren zu. Die Anzahl der Parcours und Rotationen kann er ebenfalls ändern und zu einem späteren Zeitpunkt alles bearbeiten, resp. neu zuteilen.
<i>UC04_Prüfung exportieren</i>	Der Prüfungsverantwortliche kann eine Prüfung, sofern diese von der Adminkomponente als gültig eingestuft wurde, an den Prüfungsserver exportieren.
<i>UC05_Prüfungsverlauf verfolgen</i>	Der Prüfungsverantwortliche kann den Status mithilfe des Monitors der Adminkomponente betrachten. Bei allfälligen Störungen/Problemen, welche die Adminkomponente vom Prüfungsserver erhält, hat der Prüfungsverantwortliche die Möglichkeit, die Prüfung für eine gewisse Zeit zu pausieren oder diese komplett zu stoppen.
<i>UC06_Absgeschlossene Prüfung importieren</i>	Sobald eine Prüfung beendet ist, kann der Prüfungsverantwortliche diese wieder importieren.
<i>UC07_Prüfungsdaten exportieren</i>	Der Prüfungsverantwortliche kann eine abgeschlossene, importierte Prüfung an die gewünschte Auswertungsstelle exportieren.

Tabelle 1 - Use Cases

5.1.4.2 User Stories

Die oben genannten Use Cases werden für die Implementation nach Scrum in User Stories aufgeteilt. Dabei gibt es vier Oberkategorien:

- UC_DATEN (UC01)
- UC_PLANEN (UC03, UC04, UC06, UC07)
- UC_EDITIEREN (UC02)
- UC_MONITOR (UC05)

User Story	Beschreibung
User Stories UC_DATEN	
UC_DATEN1	Der Prüfungsverantwortliche erfasst einen neuen Posten
UC_DATEN2	Der Prüfungsverantwortliche bearbeitet einen bestehenden Posten
UC_DATEN3	Der Prüfungsverantwortliche löscht einen bestehenden Posten
UC_DATEN4	Der Prüfungsverantwortliche betrachtet alle bestehenden Posten
UC_DATEN5	Der Prüfungsverantwortliche exportiert alle bestehenden Posten
UC_DATEN6	Der Prüfungsverantwortliche weist einem bestehenden Posten einen bestehenden Prüfungsbogen zu
UC_DATEN7	Der Prüfungsverantwortliche erfasst einen neuen Examinator
UC_DATEN8	Der Prüfungsverantwortliche bearbeitet einen bestehenden Examinator
UC_DATEN9	Der Prüfungsverantwortliche löscht einen bestehenden Examinator
UC_DATEN10	Der Prüfungsverantwortliche betrachtet alle bestehenden Examinatoren
UC_DATEN11	Der Prüfungsverantwortliche exportiert alle bestehenden Examinatoren
UC_DATEN12	Der Prüfungsverantwortliche erfasst einen neuen Kandidaten
UC_DATEN13	Der Prüfungsverantwortliche bearbeitet einen bestehenden Kandidaten
UC_DATEN14	Der Prüfungsverantwortliche löscht einen bestehenden Kandidaten
UC_DATEN15	Der Prüfungsverantwortliche betrachtet alle bestehenden Kandidaten
UC_DATEN16	Der Prüfungsverantwortliche exportiert alle bestehenden Kandidaten
UC_DATEN17	Der Prüfungsverantwortliche erfasst ein neues Geräte
UC_DATEN18	Der Prüfungsverantwortliche bearbeitet ein bestehendes Gerät
UC_DATEN19	Der Prüfungsverantwortliche löscht ein bestehendes Gerät
UC_DATEN20	Der Prüfungsverantwortliche betrachtet alle bestehenden Geräte

UC_DATEN21	Der Prüfungsverantwortliche exportiert alle bestehenden Geräte
UC_DATEN22	Der Prüfungsverantwortliche druckt eine Liste aller Geräte
UC_DATEN23	Der Prüfungsverantwortliche importiert eine bestehende Prüfung aus dem Excel-File
UC_DATEN24	Der Prüfungsverantwortliche importiert eine CSV-Datei der Kandidaten
UC_DATEN25	Der Prüfungsverantwortliche navigiert in der Datenübersicht
User Stories UC_PLANEN	
UC_PLANEN1	Der Prüfungsverantwortliche öffnet die Übersicht aller bestehenden Prüfungen
UC_PLANEN2	Der Prüfungsverantwortliche erfasst eine neue Prüfung
UC_PLANEN3	Der Prüfungsverantwortliche bearbeitet eine bestehende Prüfung
UC_PLANEN4	Der Prüfungsverantwortliche löscht eine bestehende Prüfung
UC_PLANEN5	Der Prüfungsverantwortliche weist der Prüfung bereits bestehende Posten zu
UC_PLANEN6	Der Prüfungsverantwortliche weist der Prüfung einen neuen Posten zu
UC_PLANEN7	Der Prüfungsverantwortliche weist der Prüfung bereits bestehende Kandidaten zu
UC_PLANEN8	Der Prüfungsverantwortliche weist der Prüfung einen neuen Kandidaten zu
UC_PLANEN9	Der Prüfungsverantwortliche erhält eine Übersicht der optimalen Anzahl an Parcours
UC_PLANEN10	Der Prüfungsverantwortliche kann einen Pausenposten allen Parcours hinzufügen
UC_PLANEN11	Der Prüfungsverantwortliche kann einen Non-EOSCE Posten benennen und allen Parcours hinzufügen
UC_PLANEN12	Der Prüfungsverantwortliche kann die Anzahl der Parcours ändern
UC_PLANEN13	Der Prüfungsverantwortliche kann die Farbe eines Parcours ändern
UC_PLANEN14	Der Prüfungsverantwortliche erhält eine Übersicht aller Rotationen
UC_PLANEN15	Der Prüfungsverantwortliche kann Rotationen hinzufügen und löschen
UC_PLANEN16	Der Prüfungsverantwortliche kann für jede Rotation ein Prüfungsdatum/-zeit setzen
UC_PLANEN17	Der Prüfungsverantwortliche kann die Reihenfolge eines Parcours innerhalb einer Rotation ändern
UC_PLANEN18	Der Prüfungsverantwortliche kann einen Parcours einer anderen Ro-

	tation zuweisen
UC_PLANEN19	Der Prüfungsverantwortliche kann die Reihenfolge eines Posten innerhalb eines Parcours ändern
UC_PLANEN20	Der Prüfungsverantwortliche kann einem Posten einen Examiner zuweisen
UC_PLANEN21	Der Prüfungsverantwortliche kann in die Kandidaten Ansicht wechseln
UC_PLANEN22	Der Prüfungsverantwortliche kann einem Posten einen anderen Startkandidaten zuweisen
UC_PLANEN23	Der Prüfungsverantwortliche kann in die Formular Ansicht wechseln
UC_PLANEN24	Der Prüfungsverantwortliche weist einem Posten eines Parcours einen anderen Prüfungsbogen zu
UC_PLANEN25	Der Prüfungsverantwortliche speichert die neue Prüfung
UC_PLANEN26	Der Prüfungsverantwortliche betrachtet den Zustand aller Prüfungen
UC_PLANEN27	Der Prüfungsverantwortliche wählt eine Prüfung aus und exportiert diese auf den Prüfungsserver
UC_PLANEN28	Der Prüfungsverantwortliche wählt eine exportierte Prüfung aus und importiert diese in ein CSV-File
UC_PLANEN29	Der Prüfungsverantwortliche gibt die Anzahl zur Verfügung stehender Räume ein und erhält einen Hinweis wie viele Rotationen daraus resultieren.
UC_PLANEN30	Der Prüfungsverantwortliche kopiert eine bestehende Prüfung
UC_PLANEN31	Der Prüfungsverantwortliche navigiert direkt zum entsprechenden Fehler
UC_PLANEN32	Der Prüfungsverantwortliche sieht die Anzahl hinzugefügter Posten / Kandidaten
UC_PLANEN33	Der Prüfungsverantwortliche sucht im Pool der Kandidaten nach einer Studentenummer
UC_PLANEN34	Der Prüfungsverantwortliche wird darauf hingewiesen, dass bereits ein generierter Plan existiert
UC_PLANEN35	Der Prüfungsverantwortliche gibt pro Prüfung die Bewertungs- und Pausendauer ein
User Stories UC_EDITIEREN	
UC_EDITIEREN1	Der Prüfungsverantwortliche öffnet die Übersicht der Prüfungsbogen
UC_EDITIEREN2	Der Prüfungsverantwortliche will einen neuen Prüfungsbogen erfassen
UC_EDITIEREN3	Der Prüfungsverantwortliche will einen bestehenden Prüfungsbogen bearbeiten oder löschen

UC_EDITIEREN4	Der Prüfungsverantwortliche fügt einen neuen Bereich hinzu
UC_EDITIEREN5	Der Prüfungsverantwortliche ordnet die Bereiche neu an
UC_EDITIEREN6	Der Prüfungsverantwortliche fügt eine neue Frage hinzu
UC_EDITIEREN7	Der Prüfungsverantwortliche ordnet die Fragen neu an
UC_EDITIEREN8	Der Prüfungsverantwortliche fügt einen neuen Fragetyp hinzu
UC_EDITIEREN9	Der Prüfungsverantwortliche kann Kriterien hinzufügen / löschen
UC_EDITIEREN10	Der Prüfungsverantwortliche speichert den Prüfungsbogen als neue Version
UC_EDITIEREN11	Der Prüfungsverantwortliche navigiert in der Editorübersicht
UC_EDITIEREN12	Der Prüfungsverantwortliche kann einen erstellten Fragetyp löschen
User Stories UC_MONITOR	
UC_MONITOR1	Der Prüfungsverantwortliche öffnet das Monitoring und sieht den Status der aktiven Prüfung
UC_MONITOR2	Der Prüfungsverantwortliche bearbeitet einen Fehlerfall
UC_MONITOR3	Der Prüfungsverantwortliche stoppt eine Prüfung
UC_MONITOR4	Der Prüfungsverantwortliche pausiert eine Prüfung
UC_MONITOR5	Der Prüfungsverantwortliche beendet die Pausierung einer Prüfung

Tabelle 2 - User Stories

5.1.5 Nicht-funktionale Anforderungen

5.1.5.1 Funktionalität

Interoperabilität

Die E-OSCE Adminkomponente soll über die Möglichkeit verfügen, mit dem bereits vorhandenen Prüfungs-Server zu kommunizieren (XML-Requests). Ausserdem soll der Import und Export von prüfungsrelevanten Daten möglich sein. Diese Daten können in Form von XML- oder Excel-Dateien auftreten.

Sicherheit

Die Sicherheitsanforderungen an die E-OSCE Adminkomponente sind enorm hoch, da die Prüfungsdaten auf keinen Fall von unbefugten Dritten eingesehen werden dürfen. Der Prüfungs-Server arbeitet bereits mit Server- und Client-Zertifikaten und übermittelt nur in einem extra dafür erstellten Netzwerk über einen sicheren Kanal (https). Die Adminkomponente soll deshalb ebenfalls in dieses Netzwerk eingebunden werden und nicht von ausserhalb ansprechbar sein. Zudem darf mit dem Prüfungsserver nur über verschlüsselte Kanäle kommuniziert werden und die Verbindungen benötigen jeweils eine Zertifikat-Authentifizierung.

Die Logindaten für die Web-basierte Adminkomponente sollen nicht im Klartext gespeichert werden, zudem wäre eine Rollen-basierte Authentifizierung von Vorteil.

5.1.5.2 Zuverlässigkeit

Fehlertoleranz

Dem Benutzer sollen bei Problemen Fehlermeldungen angezeigt werden. Der Fehler soll dann in einem Log-File gespeichert werden. Handelt es sich um einen Error bezüglich Import und Export soll der Benutzer darüber informiert werden und noch einmal die Möglichkeit erhalten die vorherige Aktion zu starten. Falls ein interner Fehler auftritt, soll der Benutzer wenn möglich mit einer Ausgabe des entsprechenden Fehlers auf den Startbildschirm geleitet werden.

Auf Netzwerk-Fehler hat die E-OSCE Adminkomponente keinen Einfluss und daher besteht kein zusätzlicher Handlungsbedarf.

5.1.5.3 Benutzbarkeit

Verständlichkeit

Die E-OSCE Adminkomponente soll für den Benutzer leicht verständlich sein. Die Paperprototypes des externen Designs sollen durch Reviews auf deren Verständlichkeit überprüft werden.

Erlernbarkeit

Die Oberfläche der E-OSCE Adminkomponente soll für den Benutzer einfach erlernbar und intuitiv bedienbar sein. Die UI-Elemente sollten sich so gut wie möglich auf die aktuellen Elemente (z.B. Formulare) beziehen und diese in ähnlicher Form wiedergeben.

Die interne Komplexität der Applikation soll vor dem Benutzer verborgen werden.

Dem Benutzer soll ein Benutzerhandbuch zur Verfügung stehen, welches ihn bei seinen ersten Schritten und bei auftretenden Problemen unterstützt.

Bedienbarkeit

Die Bedienung der E-OSCE Adminkomponente soll durch Maus und Tastatur erfolgen. Weitere Eingabegeräte werden nicht explizit unterstützt. Mittels (Paper-)Prototypen soll die Bedienbarkeit der Applikation getestet werden.

5.1.5.4 Effizienz

Verbraucherverhalten

Der Benutzer soll während der Prüfungsplanung, respektive dem Editieren von Formularen keine Performance-Einbrüche erfahren.

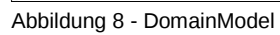
5.1.5.5 Änderbarkeit

Prüfbarkeit

Die E-OSCE Adminkomponente soll durch geeignete Unit-Tests überprüfbar gemacht werden. Dabei soll vor allem die Business-Logik und Datenbank-Anbindung getestet werden, das User Interface benötigt hingegen keine Unit-Tests.

5.2.1 Domain Model

5.2.1 Domain Model



Das Domain Model stellt die gesamte Funktionalität der Adminkomponente dar. Die einzelnen Domain-Objekte werden im Folgenden genauer beschrieben:

Domain Objekt	Beschreibung
Person	Umfasst alle Personen, welche mit der E-OSCE-Adminkomponente zusammenhängen
Proband	Student (Kandidat), welcher eine E-OSCE-Prüfung absolviert
Examinator	Experte, welcher die Leistung der Probanden bewertet
Prüfungsverantwortlicher	Verantwortlich für das Planen und Durchführen einer Prüfung
Gerät	Umfasst alle Geräte, welche mit der E-OSCE-Prüfung zusammenhängen
Monitorgerät	Gerät, mit welchem eine E-OSCE-Prüfung überwacht wird
Prüfungsgerät	Gerät, mit welchem eine E-OSCE-Prüfung durch den Examinatoren bewertet wird
Prüfungsvorlage	Ein leerer Prüfungsbogen, welcher der Examinator zur Bewertung eines Probanden verwendet
Bereich	Kategorisiert mehrere Fragen einer Prüfungsvorlage in einen Bereich
Frage	Eine Prüfungsfrage innerhalb einer Prüfungsvorlage
Kriterium	Bewertungskriterium welches der Kandidat während der Prüfung erfüllen muss.
Optionen Set	Gruppe von mehreren zu einer Frage gehörenden Bewertungsmöglichkeiten
Option	Eine Option, resp. Bewertungsmöglichkeit für eine Frage
Prüfung	Durchführung einer E-OSCE-Prüfung
Rotation	Ein Durchgang einer kompletten E-OSCE-Prüfung
Parcours	Ein Durchgang aller Posten einer E-OSCE-Prüfung
Posten	Ein Posten wird während der Prüfung von einem Kandidaten besucht. Entweder handelt es sich um einen Prüfungs- oder Pausenposten.
Pausenposten	Posten, an dem der Kandidat eine Pause hat (benötigt keinen Examinatoren)
Prüfungsposten	Posten, an dem der Kandidat durch den Examinator bewertet wird.

Tabelle 3 - Domain Model

5.3 Technologieentscheid

Für den Technologieentscheid wurden die in Kapitel 5.3.1 aufgelisteten Technologien dem Betreuer-Team vorgelegt. Diese Vorschläge dienten als Grundlage für den Beschluss in Kapitel 5.3.2.

5.3.1 Technologievorschläge

Thin Client-Server-Architektur

Vorgeschlagen wird eine Client-Server-Architektur mit Einsatz des Java Server Faces Framework, welches mittels JavaScript und Ajax um dynamische Client-Interaktionen erweitert werden kann. Laufen würde die Applikation auf dem Application-Container JBoss, welcher frei erhältlich ist.

Weitere Möglichkeiten neben JSF sind ASP.NET, Ruby on Rails, aber auch weitere Frameworks basierend auf Python oder Perl. Diese Frameworks sind jedoch nur bedingt für die Adminkomponente geeignet, da sie alle eine weitere von Java unterschiedliche Technologie darstellen. Für das IML (Institut für Medizinische Lehre) ist jedoch die Verwendung einer einzigen Technologie von grossem Vorteil, da sie einen sehr grossen Wert auf eine simple (in diesem Sinne auch einfach wartbare) Lösung legen.

Vorteile

- Ein Prüfungsverantwortlicher ist nicht an einen bestimmten Arbeitsplatz gebunden
- Mehrere Prüfungsverantwortliche können die Adminkomponente bedienen
- Zugriffsberechtigungen können via Rollensystem des JBoss gelöst werden
- Die Adminkomponente kann auch mit mobilen Geräten bedient werden (z.B. für das Monitoring).
- Es ist nur eine Serverinstallation notwendig, da der Client in einem Browser läuft und somit von diversen Geräten angesprochen werden kann.
- Lastverteilung und Ausfallsicherheit kann mittels redundanten JBoss-Installationen sichergestellt werden

Nachteile

- Benutzerfeeling, wenn eine Applikation über Localhost oder eine IP-Adresse gestartet werden muss
- Falls Adminkomponente nicht lokal auf einem PC läuft ist ein Offline-Modus nicht möglich
- Bedienung des User Interface an einem Browser unterliegt immer gewissen Einschränkungen

Fat Client-Server-Architektur

Eine weitere Möglichkeit ist eine Rich Client-Server-Architektur, wobei der Rich Client einer in Java, Silverlight oder Flex entwickelten Applikation entspricht.

Vorteile

- User kann eine normale Applikation vom Desktop starten
- Mehrere Prüfungsverantwortliche können die Adminkomponente bedienen
- Zugriffsberechtigungen könnten theoretisch via Rollensystem des JBoss gelöst werden.
- Lastverteilung und Ausfallsicherheit kann mittels redundanten JBoss-Installationen sichergestellt werden

Nachteile

- Keine Bedienung mit Mobilgeräten. Für das Monitoring müsste eine zusätzliche Technologie verwendet werden
- Neben der Serverinstallation ist auch eine Clientinstallation notwendig
- Falls Adminkomponente nicht lokal auf einem PC läuft ist ein Offline-Modus nicht möglich
- Höherer Aufwand, da Swing-Programmierung ziemlich umständlich ist, und zudem noch eine Kommunikationskomponente evaluiert oder entwickelt werden muss

Desktopapplikation

Denkbar wäre auch eine reine Java-Swing Desktopapplikation, welche nur mit dem Prüfungsserver kommuniziert.

Vorteile

- Simple Installation auf dem gewünschten Rechner
- User kann eine normale Applikation vom Desktop starten
- Zugriffsberechtigungen müssen, wenn überhaupt nur lokal überprüft werden.
- Lastverteilung und Ausfallsicherheit kann mittels redundanten JBoss-Installationen sichergestellt werden
- Adminkomponente ist sowohl Online als auch Offline (eingeschränkt) verfügbar.

Nachteile

- Es kann nur ein Prüfungsverantwortlicher die Adminkomponente bedienen
- Keine Lastverteilung und Ausfallsicherheit
- Keine Bedienung mit Mobilgeräten. Für das Monitoring müsste eine zusätzliche Technologie verwendet werden
- Datenbank müsste auf demselben Rechner installiert werden wie die Desktopapplikation
- Höherer Aufwand, da Swing-Programmierung ziemlich umständlich ist

5.3.2 Beschluss

In Absprache mit dem Betreuerteam wurde beschlossen, die Adminkomponente als eine Enterprise Application mit einem JSF-Frontend und einer relationalen Datenbank zu entwickeln.

5.5 Paperprototyping

Die folgende Tabelle gibt einen Überblick über die Entwicklung der E-OSCE Adminkomponente. In der linken Spalte befinden sich die ersten Versionen der Paperprototypes. Diese wurden nach einem Review mit den Betreuern überarbeitet. In der rechten Spalte werden die finalen Paperprototypes angezeigt. Alle Paperprototypes der Versionen 1 und 2 befinden sich im Anhang [ANH02]

Paperprototypes Version 1	Paperprototypes Version 2
Prüfungsübersicht	

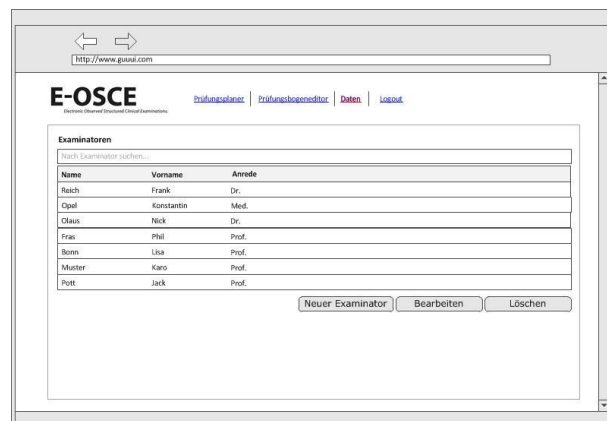
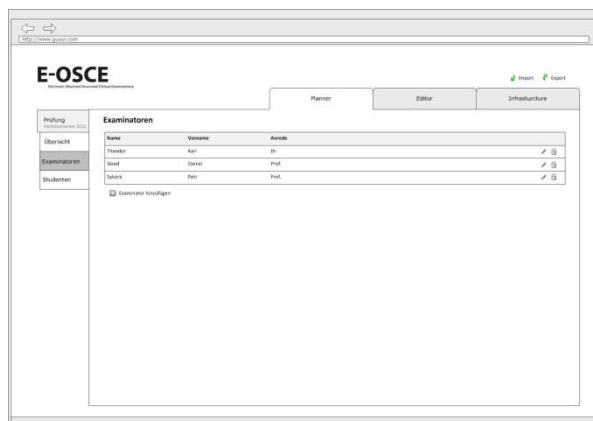
Die Prüfungsübersicht enthält nun mehr Informationen zu den einzelnen Prüfungen. Neben der Anzahl an Rotationen, Parcours, Posten und Kandidaten wird zusätzlich ein Prüfungszustand angezeigt.

Navigation

--	--

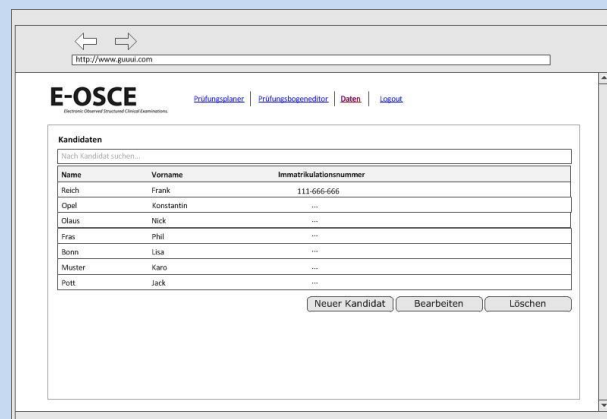
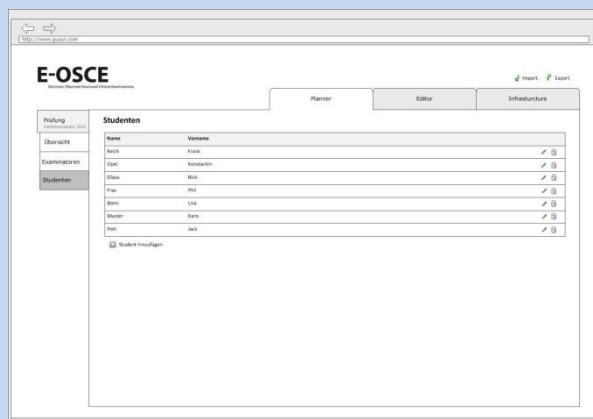
Anstelle von Tabs wurde eine normale Hauptnavigation neben dem Logo eingeführt.

Examinatoren



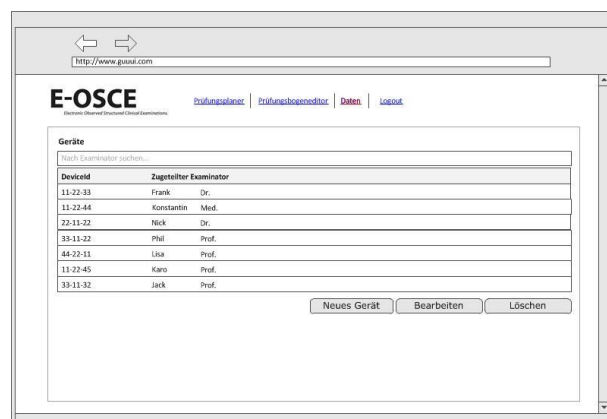
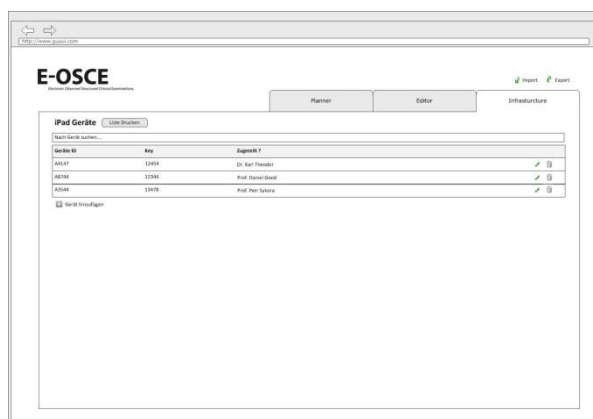
Die Icons zum Bearbeiten und Löschen von Examinatoren wurden durch Buttons unterhalb der Datensätze ersetzt.

Kandidaten



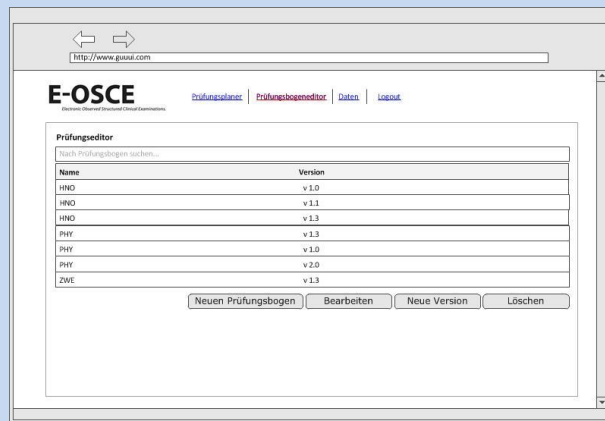
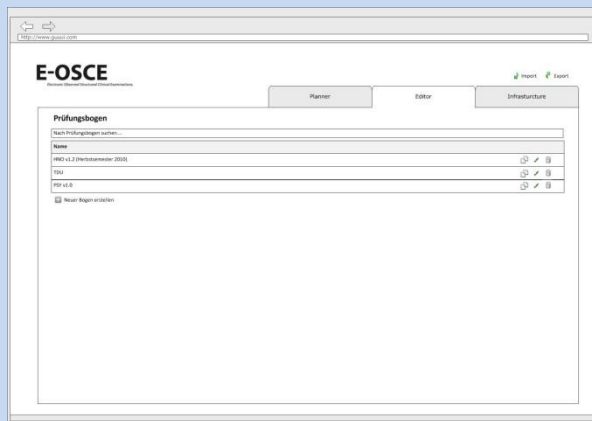
Die Icons zum Bearbeiten und Löschen von Kandidaten wurden durch Buttons unterhalb der Datensätze ersetzt. Zusätzlich wurde neben dem Name und Vornamen eine Studentennummer eingeführt.

Infrastruktur

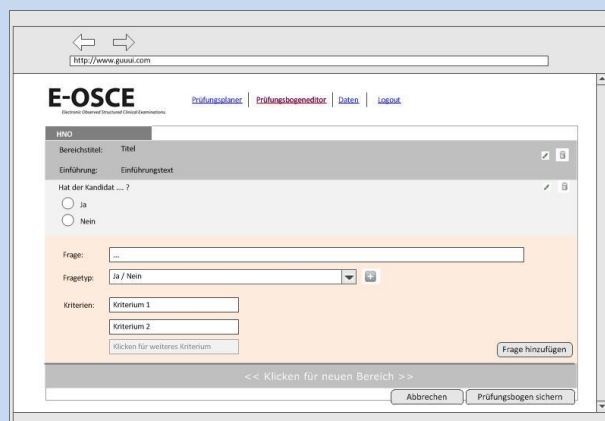
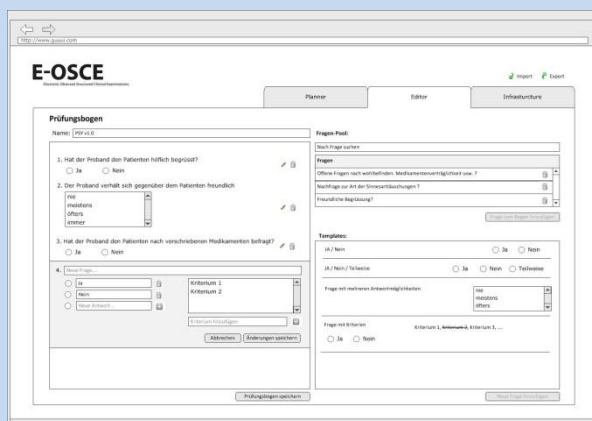
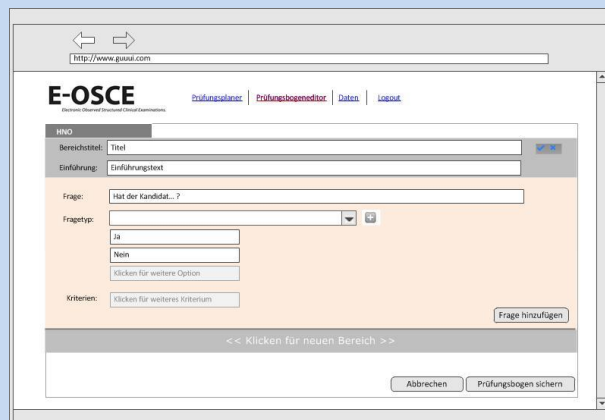
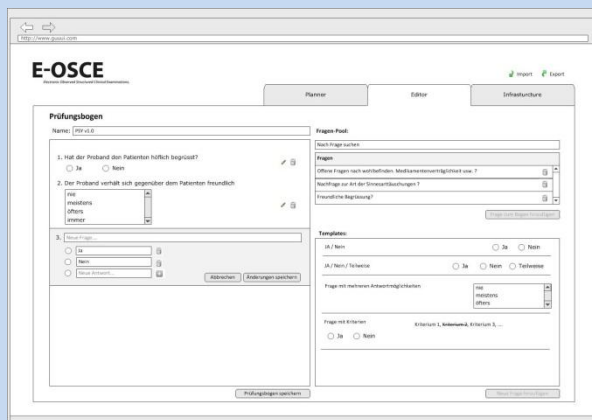


Die Icons zum Bearbeiten und Löschen von Geräten wurden durch Buttons unterhalb der Datensätze ersetzt.

Editor

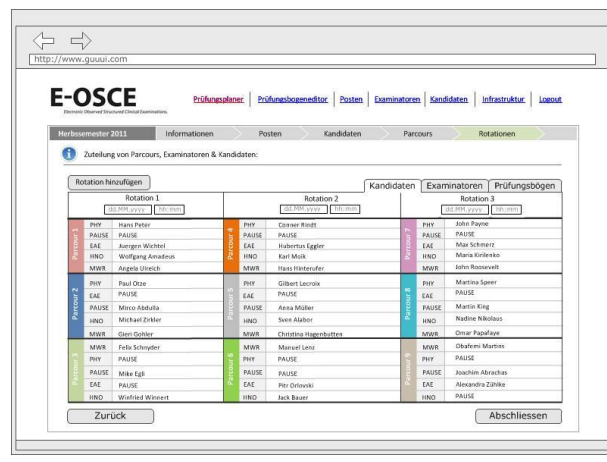
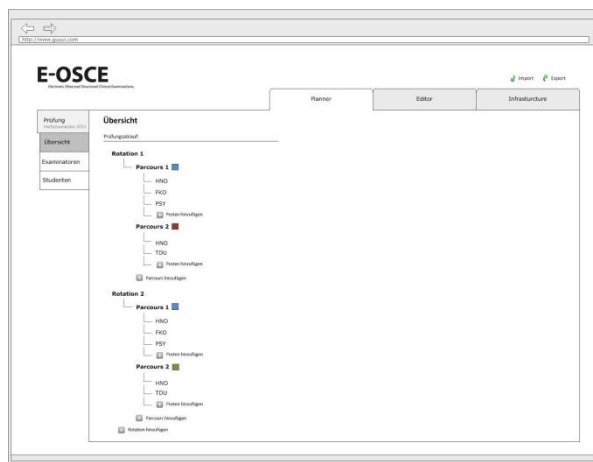
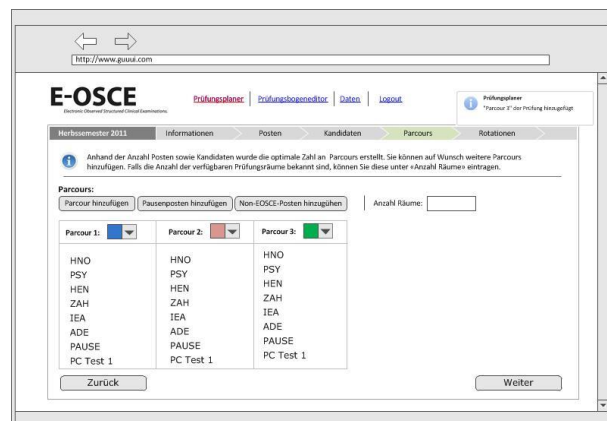
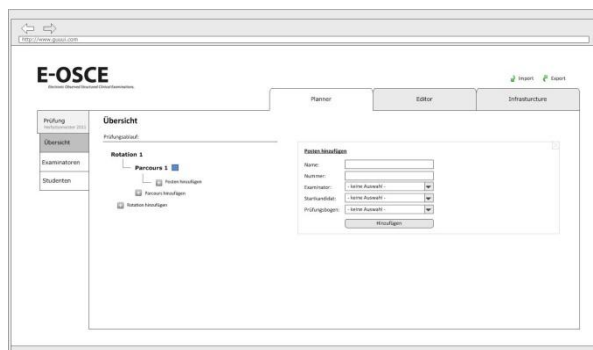
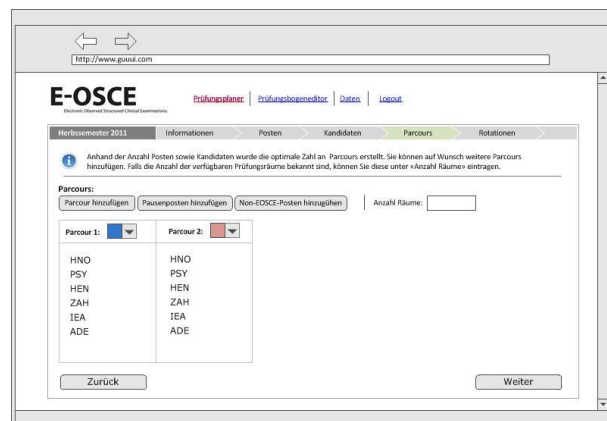
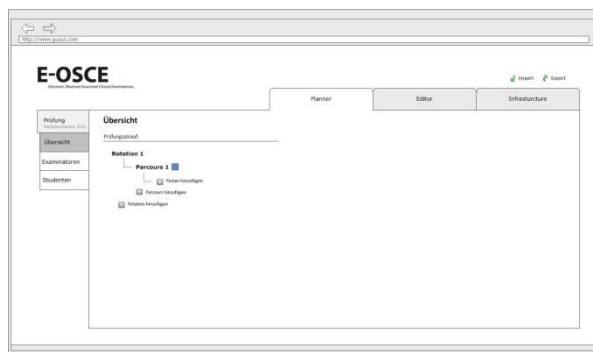


Die Icons zum Bearbeiten und Löschen von Prüfungsbogen wurden durch Buttons unterhalb der Datensätze ersetzt. Zusätzlich wurde der Button „Neue Version“ hinzugefügt. Mit dessen Hilfe es möglich ist einen bestehenden Prüfungsbogen zu kopieren.



Der Editor wurde komplett überarbeitet. Nun ist es möglich den Fragebogen in verschiedene Bereiche aufzuteilen, sowie Fragen zwischen den einzelnen Bereichen per Drag & Drop zu verschieben. Zusätzlich wurde der Pool mit bestehenden Fragen entfernt.

Prüfungsplaner



Die Baumstruktur für die Prüfungsplanung wurde durch einen Wizard basierten Planungsvorgang ersetzt. Der Wizard vereinfacht die Planung einer Prüfung und bietet eine bessere Gesamtübersicht. Nachdem grundsätzliche Prüfungsinformationen eingegeben wurden, können Kandidaten sowie Posten zur Prüfung hinzugefügt werden. Im nächsten Schritt können die Parcours bearbeitet werden. Abgeschlossen wird die Prüfungsplanung mit der Rotationsübersicht. Diese erlaubt das Zuteilen von Kandidaten, Examinatoren sowie Prüfungsformulare zu den einzelnen Posten.

Tabelle 4 - Paperprototypes

5.6 Heuristische Evaluation

Mit den überarbeiteten Paperprototypes in der Version 2 [ANH02] wurde eine heuristische Evaluation durchgeführt. Als Testpersonen stellten sich Michael Graf sowie Prof. Markus Stolze der HSR zur Verfügung. Die daraus resultierenden Erkenntnisse flossen in die Finale Version der Adminkomponente ein.

In Anhang [ANH13] befindet sich das Testskript, welches als roter Faden für die Durchführung der heuristischen Evaluation benutzt worden ist. Während der Evaluation wurden die Probanden für eine spätere Auswertung gefilmt und im Anhang dieser Arbeit beigefügt [ANH14] und [ANH15].

Die Resultate und Erkenntnisse der heuristischen Evaluation sind in Kapitel 5.6.1 festgehalten.

5.6.1 Erkenntnisse heuristische Evaluation „E-OSCE Admin Tool

5.6.1.1 Testperson Michael Graf

Wann: 14.04.2011, 10:00-11:00

Wo: HSR Zimmer 1.223

Unterlagen: Paperprototype Version 2 [ANH02], Testskript [ANH13]

Video: [ANH13]

Prüfung planen

- Direktsprung von Übersicht zu Rotationen
- Growlnachrichten nicht oben rechts platzieren
- „X“ bei Fehlernachricht sieht wie „schliessen“ aus
- „Neuen Posten erfassen“ anstatt „Neuer Posten hinzufügen“
- „Neuen Kandidaten erfassen“ anstatt „Neuer Kandidat hinzufügen“
- Kürzerer Hinweis bei „Parcours“
- Genauerer Hinweistext, bei Dialog „Prüfung nicht vollständig ausgefüllt“

Daten erfassen

- Versionierung des Prüfungsbogens ist nicht genügend aussagekräftig. Besser z.B. nur aktuellsten Bogen anzeigen, und weitere Versionen auf einem neuen Screen, oder Checkbox, mit der alte Bogen ausgeblendet werden können.
- Breadcrump für Daten

Prüfungsbogeneditor

- Kriterien und neue Fragetypen löschen?

Monitor

- Nachfragen bei „Prüfung stoppen“

5.6.1.2 Testperson Prof. Markus Stolze

Wann: 20.04.2011, 09:00-10:00

Wo: HSR Zimmer 1.223

Unterlagen: Paperprototype Version 2 [ANH02], Testskript [ANH13]

Video: [ANH14]

Prüfung planen

- Bestehende Prüfung kopieren
- Falls mehrere Fehler, diese auflisten und Button „zum ersten Fehler“
- Label „Start“ der Prüfung in „Durchführungstermin“ ändern
- Fehlernachrichten, wo möglich nur bei Inputfeld anzeigen. Growl nur verwenden, wenn dies nicht möglich ist.
- Klickbare Breadcrumb-Items sollten andere Farbe haben
- Zähler, wie viele Posten/Kandidaten zur Prüfung hinzugefügt worden sind.
- Was, wenn 5 Hans Muster vorhanden sind? Es müsste z.B. die Immatrikulationsnummer angezeigt werden und in der Suche nicht nur nach Name sondern auch nach Immatrikulationsnummer gesucht werden können.
- Excelimport der Kandidaten und direkte Zuweisung zur Prüfung.
- Sind die Farben nicht auf der Rotation?
- Falls Räume ausgefüllt werden, ein Hinweis, wie viele Rotationen generiert werden
- Button „Weiter“ in „Plan generieren“ ändern.
- Falls bereits ein generierter Plan besteht, ein Dialog, „Achtung, ein neuer Plan wird generiert“

Daten erfassen

- „Daten“ in „Basisdaten“ ändern
- Breadcrumb für Daten
- Manuelle Erfassung zu mühsam. Excelimport besser.

Prüfungsbogeneditor

-

Monitor

-

5.6.1.3 Resultierende Userstories

UC_DATEN25	Der Prüfungsverantwortliche weiss wo er sich befindet
UC_EDITIEREN11	Der Prüfungsverantwortliche weiss wo er sich befindet
UC_EDITIEREN9	Der Prüfungsverantwortliche kann Kriterien hinzufügen / löschen
UC_EDITIEREN12	Der Prüfungsverantwortliche kann einen erstellten Fragetyp löschen
UC_MONITOR6	Der Prüfungsverantwortliche erhält ein Nachfragedialog bei „Prüfung stoppen“
UC_PLANEN30	Bestehende Prüfung kopieren
UC_PLANEN31	Der Prüfungsverantwortliche navigiert direkt zum entsprechenden Fehler
UC_PLANEN32	Der Prüfungsverantwortliche sieht die Anzahl hinzugefügter Posten / Kandidaten
UC_PLANEN33	Der Prüfungsverantwortliche sucht im Pool der Kandidaten nach einer Studentenummer
UC_DATEN24	Der Prüfungsverantwortliche importiert eine CSV-Datei der Kandidaten
UC_PLANEN29	Der Prüfungsverantwortliche gibt die Anzahl zur Verfügung stehender Räume ein und erhält einen Hinweis wie viele Rotationen daraus resultieren.
UC_PLANEN34	Der Prüfungsverantwortliche wird darauf hingewiesen, dass bereits ein generierter Plan existiert

5.6.1.4 Labelanpassungen

- „Neuen Posten erfassen“ anstatt „Neuer Posten hinzufügen“
- „Neuen Kandidaten erfassen“ anstatt „Neuer Kandidat hinzufügen“
- Genauerer Hinweistext, bei Dialog „Prüfung nicht vollständig ausgefüllt“
- Kürzerer Hinweis bei „Parcours“
- Label „Start“ der Prüfung in „Durchführungstermin“ ändern
- Button „Weiter“ in „Plan generieren“ ändern.
- „Daten“ in „Basisdaten“ ändern

6 Software Architektur Dokument

Rev.	Datum	Wer	Änderung
0.1	06.05.2011	Sven Lenz	Dokument erstellt
0.2	17.05.2011	Sven Lenz	Klassendiagramme hinzugefügt
0.3	11.06.2011	Sven Lenz	Klassendiagramme und Beschreibungen überarbeitet
0.4	13.06.2011	Matthias Good	Review

6.1 Einführung

6.1.1 Zweck

Dieses Dokument beschreibt die Software Architektur des Grundsystems.

6.1.2 Gültigkeitsbereich

Das Dokument ist für die komplette Bachelorarbeitsdauer gültig

6.1.3 Übersicht

Zu Beginn des Dokumentes wird die Architekturübersicht beschrieben.

Danach werden die eingesetzten Technologien vorgestellt und die einzelnen Packages sowie das Datenmodell genauer betrachtet.

6.2 Gesamtarchitekturübersicht

Die Abbildung 9 bietet einen Überblick der verwendeten Komponenten und Interaktionen. In den nachfolgenden Abschnitten werden die einzelnen Komponenten und die Schnittstellen genauer beschrieben.

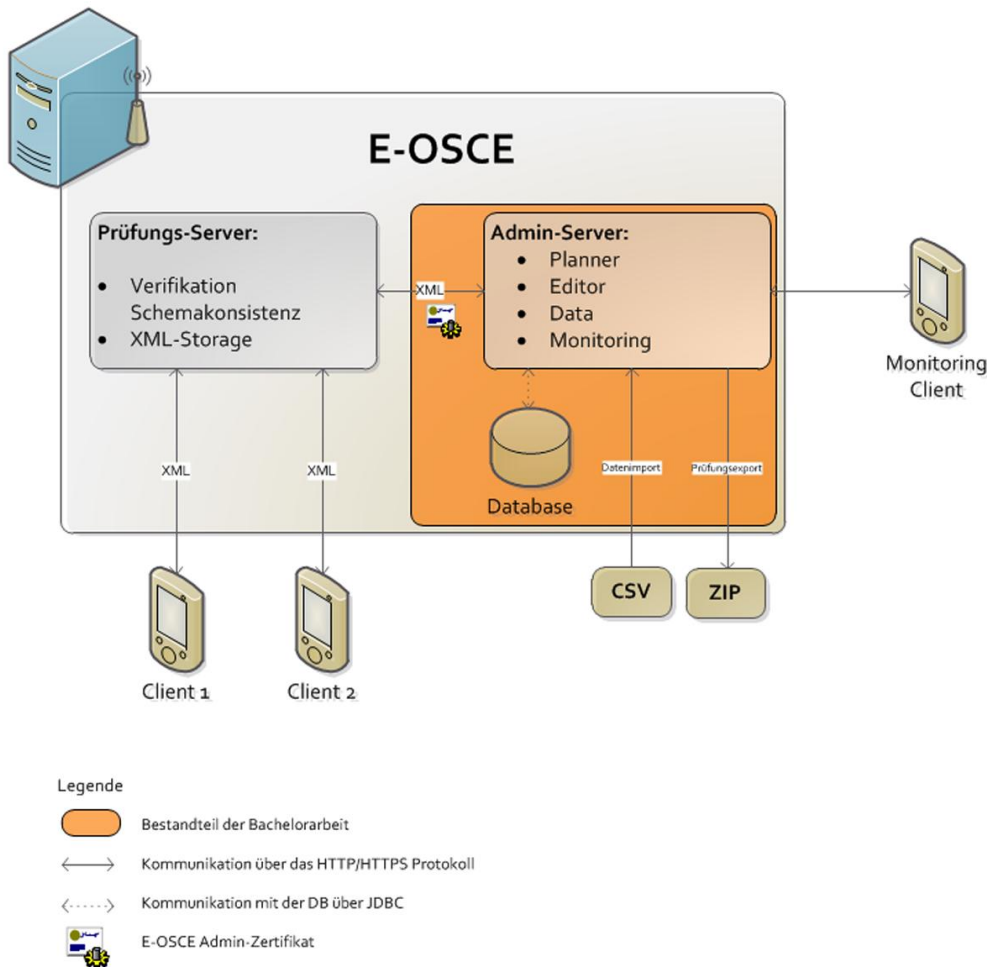


Abbildung 9 - Überblick der verwendeten Komponenten

6.2.1 Komponenten

6.2.1.1 Prüfungs-Server

Der Prüfungs-Server stellt einen RESTful-Server dar, auf welchem während der Prüfungszeit alle Informationen in Form von XML-Files abgelegt sind. Die Clients laden sich die entsprechenden XML-Files und stellen die editierten Formulare wieder zurück auf den Server.

Vor und nach der Prüfung befinden sich keine Daten auf dem Server. Die Prüfungsdaten werden von der Adminkomponente an den Server geliefert. Der Export der Prüfung geschieht ebenfalls über den Admin.

6.2.1.2 Admin-Server

Die Administrationskomponente wird vor und nach der Prüfungszeit von einem oder mehreren Prüfungsverantwortlichen bedient. Sie dient der Erstellung und Koordination von Prüfungen und Prüfungsbogen.

Zusätzlich kann noch eine Liste der verfügbaren Clients erfasst werden.

6.2.1.3 Client

Der Client ist ein Gerät, welches während der Prüfung von einem Examiner bedient wird. Auf den Clients werden die Ergebnisse eines Probanden, welcher einen Posten besucht hat, erfasst und dem Server mitgeteilt.

6.2.1.4 Monitoring Client

Monitoring Clients sind Geräte, die während der Prüfung von einem Prüfungsverantwortlichen bedient werden. Neben Statusinformationen zu einer Prüfung kann eine Prüfung über den Monitoring Client auch pausiert und sogar gestoppt werden.

6.2.2 Schnittstellen

6.2.2.1 Übertragungsprotokoll

Zwischen allen beschriebenen Komponenten wird das sichere Hypertext-Übertragungsprotokoll HTTPS für die Kommunikation verwendet.

Für die Kommunikation zwischen dem Prüfungsserver und dem Administrationsserver wird zusätzlich die zertifikatbasierte Authentifizierung eingesetzt.

6.2.2.2 Admin zu Server

Die Schnittstelle zwischen dem Admin- und dem Prüfungs-Server wird mittels XML Schemas (XSD) beschrieben. Da die Serverkomponente RESTful arbeitet, muss die Adminkomponente die Services jeweils gegen das Schema validieren und das Marshalling/Unmarshalling der HTTP-Requests und -Responses vornehmen. Die Nutzdaten befinden sich dabei jeweils im Body des HTTP-Requests beziehungsweise der HTTP-Response.

6.2.2.3 Admin zu Datenbank

Mittels Einsatzes eines Container-Managed Persistence Model, welches der EJB 3.0 Standard zur Verfügung stellt, kann via dem Java Naming and Directory Interface (JNDI) auf die Datenbank zugegriffen werden.

6.3 Strategien und Konzepte

Die grundlegende Technologie dieser Bachelorarbeit stellt eine Java EE Applikation dar.

Diese Technologie ermöglicht einen „Separation of Concerns“ (SoC) Ansatz, wodurch der Zugriff auf die Datenbank, die Businesslogik und die Aufbereitung des User Interface vollkommen losgelöst umgesetzt werden können.

Der Vorteil dieses Ansatzes ist, dass jede Applikation innerhalb einer Schicht durch eine andere ausgetauscht oder ergänzt werden kann. Beispielsweise ist es möglich, anstelle des Thin Client (Web-oberfläche) einen Fat Client einzusetzen.

Die Abbildung 10 illustriert die Aufteilung in der E-OSCE Adminkomponente in die verschiedenen Schichten.

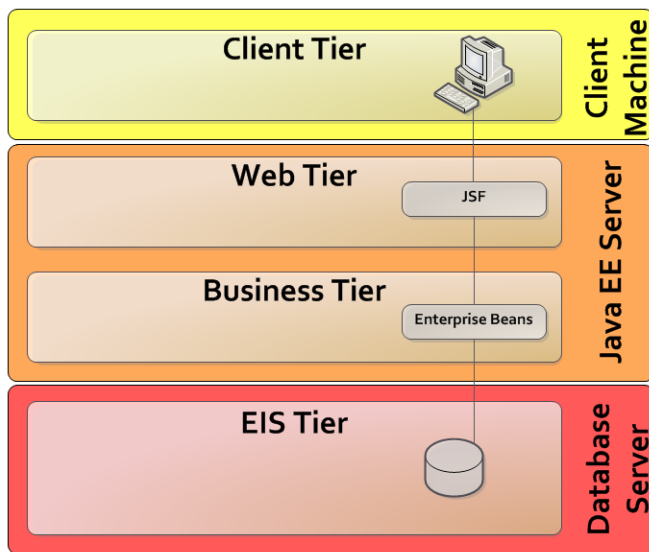


Abbildung 10 - technische Architektur

6.3.1 Eingesetzte Technologien

Die Tabelle 5 bietet, aufgeteilt nach den verschiedenen Schichten, einen Überblick über die eingesetzten Technologien.

Tier	Technologie
Client	- Extensible HyperText Markup Language (XHTML) - jQuery [03]
Web Tier	- Java Server Faces (JSF) 2.0 (Mojorra 2.0.2) [04] - Primefaces 2.2.1 [05]
Business Tier	- Enterprise Java Beans (EJB) 3.0 [06] - Java Architecture for XML Binding (JAXB) 2.0 [07]
EIS Tier	- Java Persistence API (JPA) 2.0

Tabelle 5 - Technologien

6.3.2 Java EE Server

Für diese Bachelorarbeit wird der frei erhältliche JBoss 6.0 [08] verwendet, da dieser einer der am weitesten verbreiteten Java EE Server ist und somit auch ein gewisses Mass an Standardkonformität bietet. Zusätzlich ermöglicht der JBoss auch eine Integration von weiteren Open-Source-Projekten (wie z.B. Hibernate).

6.3.3 Datenbank

Als Datenbank wird die relationale Datenbank PostgreSQL 9.0 [09] eingesetzt, welche frei erhältlich ist. Durch den gewählten Architekturansatz können aber auch relationale Datenbanken von anderen Herstellern wie z.B. Oracle eingesetzt werden.

6.3.4 Logging

Das Logging der Benutzerinteraktionen übernimmt das log4j Framework welches standardmässig im JBoss enthalten ist. Bei der Entwicklung wurde viel Wert auf das Loggen aller wichtigen Einstiegs- und Checkpunkte in Methoden, und Fehlermeldungen gelegt.

6.3.5 Internationalisierung

Eine spezielle Internationalisierung der Applikation wurde nicht vorgenommen. Jedoch wurde streng darauf geachtet, dass alle Labels und Meldungen in einem Property-File ausgelagert sind. Das JSF-Framework ermöglicht es, weitere Property-Files einer gewünschten Locale bereitzustellen und so weitere Sprachen für die Applikation freizuschalten.

6.3.6 JSF und HTML Komponenten

Bei der Entwicklung werden JSF Standard, Primefaces und jQuery Komponenten verwendet. Da Primefaces einige nützliche Komponenten wie die sortier- und filterbare Tabelle, Benutzerdialoge und Growl-Meldungen enthält, wurde Primefaces als Zusatz zu den bestehenden JSF Komponenten ausgewählt. Da nicht alle Anforderungen an die UI Elemente mit JSF und Primefaces umgesetzt werden können (z.B. beim Drag & Drop im Formulareditor) werden auch einige Komponenten aus der jQuery-Bibliothek verwendet.

6.3.7 Entwicklungsumgebung

Als Entwicklungsumgebung wird die Eclipse IDE for Java EE Developers [10] eingesetzt, da diese die Entwicklung einer Enterprise Applikation optimal unterstützt. Vorteilhaft sind dabei vor allem folgende Tools:

- Starten/Stoppen eines Servers direkt aus der IDE
- Deployment der Applikation auf den Server direkt aus der IDE
- Integration eines JUnit-Plugin um die Code-Qualität zu sichern
- Integration von Java Architecture for XML Binding (JAXB) 2.0, um aus dem XSD-Interface Java-Klassen für das Marshalling und Unmarshalling der XML-Daten zu generieren.

6.3.8 Namenskonventionen

Zusätzliche zu der logischen Trennung aller Klassen in verschiedene Packages werden auch Namenskonventionen für die Klassen eingeführt. Dies hat den Vorteil, dass eine Ressource auch während Meetings und in der Dokumentation eindeutig adressiert werden kann und zusätzlich bei der Entwicklung die entsprechende Klasse intuitiv gefunden werden kann.

Was	Namenskonvention	Beispiel
Managed Bean (JSF)	Eine Managed Bean endet auf Form.	AssessmentForm
Remote Stub	Die Schnittstelle endet auf Remote.	AssessmentRemote
Enterprise Java Bean	Das EJB, welches ein Remote Stub implementiert endet auf Bean.	AssessmentBean
Entity	Ein Entity, welches auf der DB persistiert wird, endet auf Entity.	AssessmentEntity
Interface-Testklasse	Eine Testklasse setzt sich aus dem Namen des Remote Stub und der Endung Test zusammen.	AssessmentRemoteTest

Tabelle 6 - Namenskonventionen

6.4 Logische Architektur

Die Adminkomponente wird in die sechs verschiedenen Applikationen

- eosceAdmin
- eosceAdminApp
- eosceAdminTest
- eosceAdminDaoInterface
- eosceAdminDao
- eosceAdminDomain

aufgeteilt. Die Abbildung 11 zeigt die Abhängigkeiten unter den Applikationen und die Tabelle 7 liefert eine Beschreibung der Applikationen. Alle Geräte aus dem Deployment Diagramm von Abbildung 11 können physisch auch dasselbe Gerät sein.

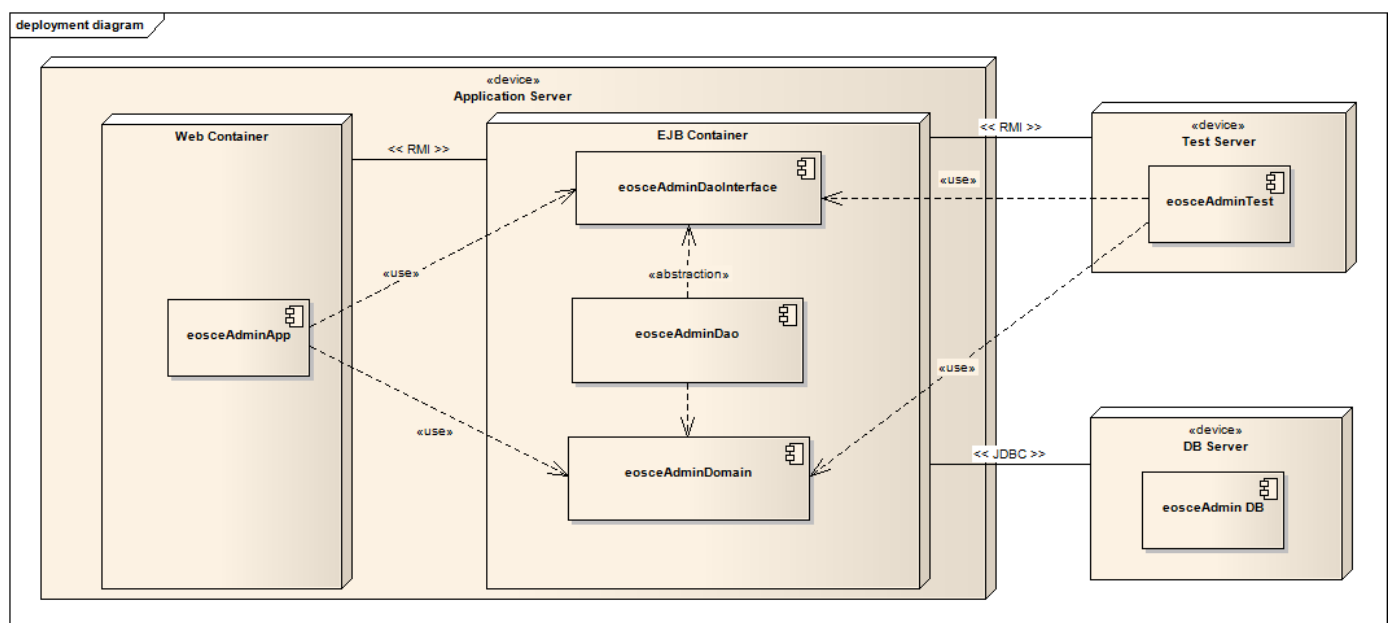


Abbildung 11 – Deployment Diagramm

Applikation	Beschreibung
eosceAdmin	Die Applikation eosceAdmin stellt das Enterprise Application Archive (EAR) dar, welches die Applikationen des Web und EJB Container zu einer logischen Applikation zusammenfasst.
eosceAdminApp	Die eosceAdminApp stellt das Web Application Archiv (WAR) dar, welches die Daten der Domain liest und diese grafisch aufbereitet.
eosceAdminDaoInterface	Das eosceAdminDaoInterface stellt eine Abstraktion des eosceAdminDao dar. Es beinhaltet die abstrakten Remote-Stubs, welche im eosceAdminDao von den Data Access Objects (Enterprise Java Beans) implementiert werden müssen. Die Remote-Stubs entsprechen somit den Services, welche einen Mechanismus für den Zugriff auf die Datenbank anbieten.
eosceAdminDao	Die Data Access Objects entsprechen Enterprise Java Beans, welche das vorgegebene Interface aus der Applikation eosceAdminDaoInterface implementieren und mittels eines Persistence-Managers auf die Domain-Objekte zugreifen.
eosceAdminTest	Clientapplikation, welche über das eosceAdminDaoInterface und dem Einsatz von JUnit die Businesslogik testet.

Tabelle 7 - Applikationen

6.5 Klassendiagramme

Nachfolgend werden mehrere Klassendiagramme abgebildet und erläutert.

Die Abbildung 12 beinhaltet eine Übersicht der Zusammenhänge zwischen den wichtigsten Klassen der Applikationen eosceAdminApp, esoceAdminDaoInterface und esoceAdminDao. Das Klassendiagramm der Abbildung 12 befindet sich in besserer Qualität im Anhang [ANH18]. Übersichtshalber sind keine privaten Variablen und die Accessor-Methoden (Getter und Setter) in diesem Klassendiagramm und die Entity-Klassen der Domain werden in Abbildung 13 dargestellt.

In der Abbildung 14 sind alle Testklassen mit den jeweiligen Testfällen aufgeführt.

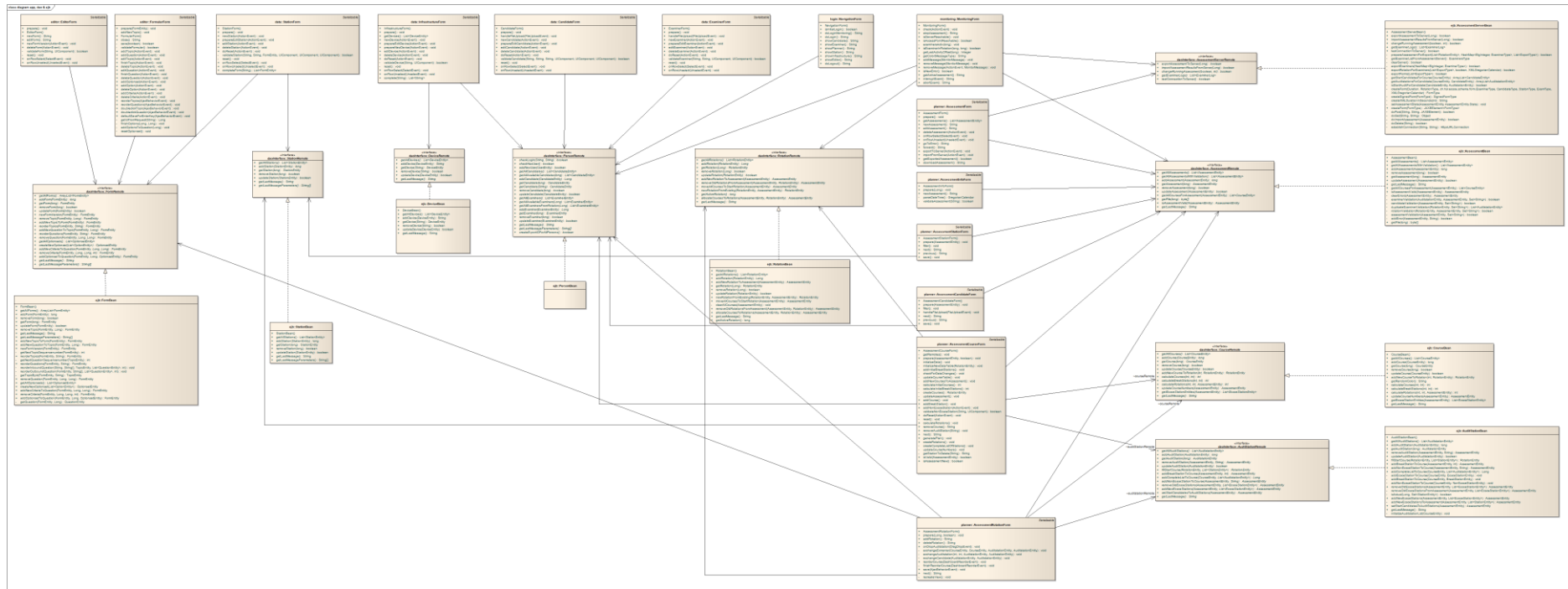
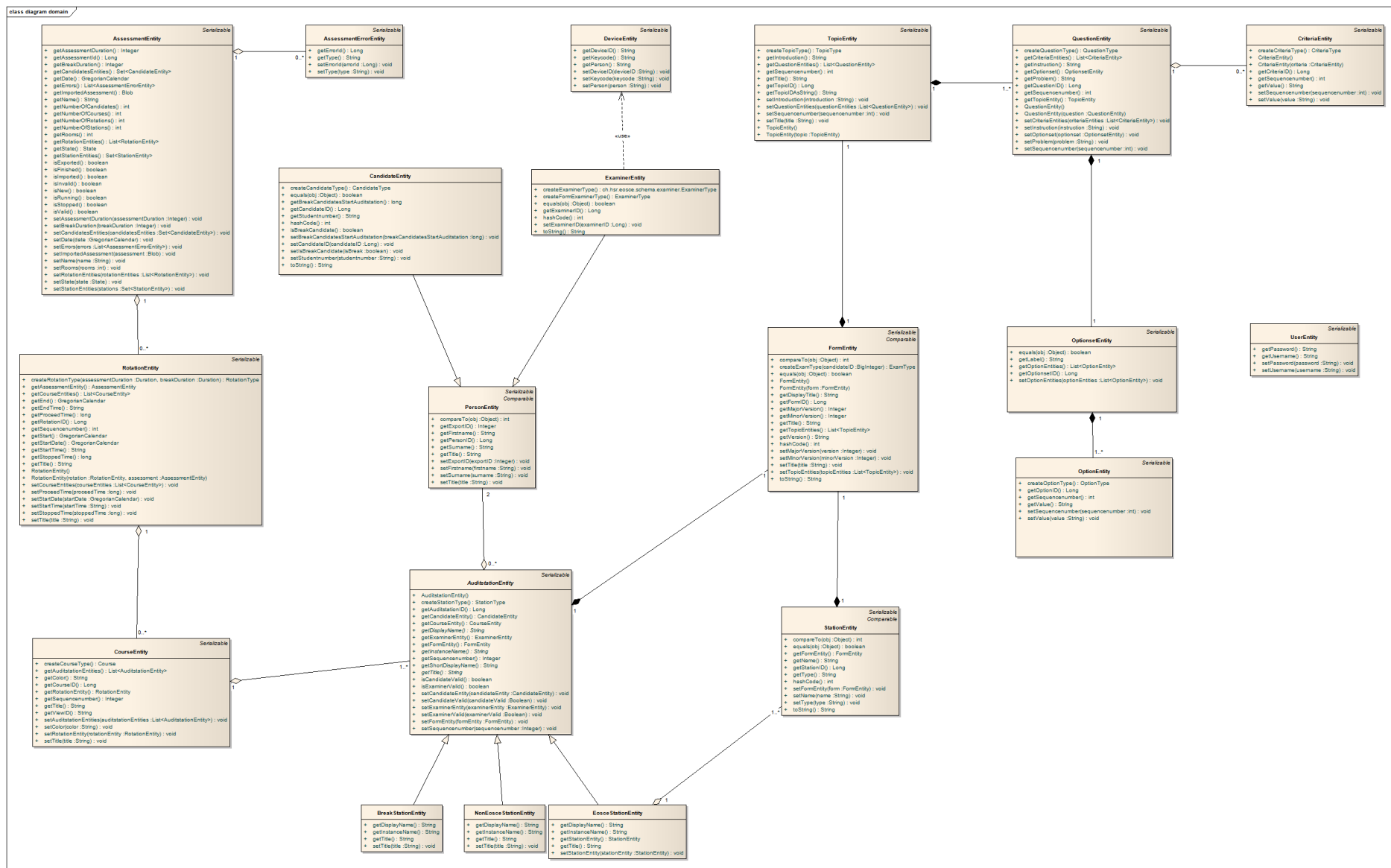


Abbildung 12 - Übersicht App, DaoInterface und Dao



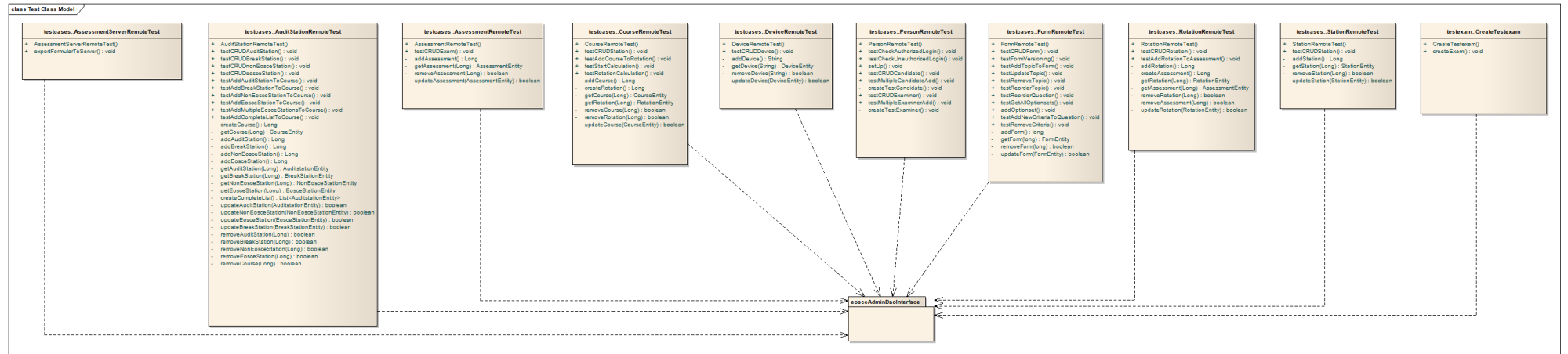


Abbildung 14- Klassendiagramm Test (eosceAdminTest)

6.6 Klassenbeschreibungen

In den nachfolgenden Kapiteln werden die einzelnen Klassen der Applikation kurz beschrieben.

6.6.1 eosceAdminApp

Die meisten Klassen der eosceAdminApp entsprechen Managed Beans, welche von einer JSF-Seite aufgerufen werden.

Einerseits liefern die Methoden die Daten, welche auf der Webseite angezeigt werden müssen, andererseits werden die Methoden auf eine Aktion vom Client hin aufgerufen. Die Kommunikation mit der Datenbank geschieht dabei ausschliesslich über die im eosceAdminDaoInterface zur Verfügung gestellten Interfaces.

Klasse	Package	Beschreibung
SessionFilter	*.filter	Servlet Filter, der jeden eingehenden Request, welcher nicht dem Login-Request oder einem Ressourcen-Request entspricht, auf eine gültige Session überprüft. Besteht keine gültige Session, wird der Request auf die Startseite umgeleitet.
NavigationForm	*.login	Managed Bean der Seite login.xhtml. Ist für das Erstellen eines neuen E-OSCE Admin-User und Anmelden am System zuständig.
StationForm	*.data	Managed Bean der Seite station.xhtml. Ist für die Darstellung und der bereits erfassten Posten und die Erstellung von neuen Posten zuständig.
ExaminerForm	*.data	Managed Bean der Seite examiner.xhtml. Ist für die Darstellung und der bereits erfassten Examinatoren und die Erstellung von neuen Examinatoren zuständig.
CandidateForm	*.data	Managed Bean der Seite candidate.xhtml. Ist für die Darstellung und der bereits erfassten Kandidaten und die Erstellung von neuen Kandidaten zuständig.
InfrastructureForm	*.data	Managed Bean der Seite infrastructure.xhtml. Ist für die Darstellung und der bereits erfassten Geräte und die Erstellung von neuen Geräten zuständig.
EditorForm	*.editor	Managed Bean der Seite editor.xhtml. Ist für die Darstellung und der bereits erfassten Formulare und die Erstellung von neuen Formularen zuständig.
FormularForm	*.editor	Managed Bean der Seite formular.xhtml. Über dieses Bean wird ein Formular mit seinen Bereichen, Fragen, Optionen und Kriterien erstellt. Zusätzlich werden Methoden für das Drag & Drop und Sortieren bereitgestellt.
BreadCrumbForm	*.planner	Managed Bean des Prüfungs-Wizard. In diesem Bean wird die momentan bearbeitete Prüfung zwischengespeichert und die Navigation inner-

		halb des Wizard behandelt. Dazu gehört unter anderem auch die Validierung, ob ein Wizard-Schritt bereits aufgerufen werden kann.
AssessmentForm	*.planner	Managed Bean der Seite assessment.xhtml. Ist für die Darstellung und der bereits erfassten Prüfungen und die Erstellung von neuen Prüfungen zuständig.
AssessmentInfoForm	*.planner	Managed Bean der Seite assessmentInfo.xhtml und zuständig für den ersten Schritt des Prüfungs-Wizard. In diesem Bean wird initial eine neue Prüfung angelegt oder eine bestehende Prüfung von der Datenbank geladen. Der Prüfung wird in diesem Bean der Name, die Bewertungs-/Pausendauer und das Datum zugewiesen.
AssessmentStationForm	*.planner	Managed Bean der Seite station.xhtml und zuständig für den zweiten Schritt des Prüfungs-Wizard. Der Prüfung wird vom Bean eine Auswahl an Posten zugewiesen.
AssessmentCandidateForm	*.planner	Managed Bean der Seite candidate.xhtml und zuständig für den dritten Schritt des Prüfungs-Wizard. Der Prüfung wird vom Bean eine Auswahl an Kandidaten zugewiesen.
AssessmentCourseForm	*.planner	Managed Bean der Seite course.xhtml und zuständig für den vierten Schritt des Prüfungs-Wizard. Wird dieses Bean für eine Prüfung das erste Mal aufgerufen, wird Anhand der Anzahl Posten und Kandidaten die optimale Anzahl an Parcours berechnet. Bei jedem weiteren Aufruf, werden allfällige Änderungen nachträglich an der Prüfung vorgenommen. Neben der Generierung der Parcours werden auch die nötigen Rotationen erstellt (die Anzahl ist dabei von der Anzahl Räume abhängig, welche der Benutzer optional eingeben kann)
AssessmentRotationForm	*.planner	Managed Bean der Seite rotation.xhtml und zuständig für den fünften und letzten Schritt des Prüfungs-Wizard. Das Bean bietet Methoden, um neue Rotationen zu erstellen sowie Bestehende zu löschen. Des Weiteren werden Methoden für die Zuteilung von Kandidaten, Examinatoren oder Prüfungsbogen zu einer Prüfungsstation angeboten. Neben den oben genannten Aktionen können mittels diesem Bean per Drag & Drop Examinatoren, Kandidaten, Stationen und Parcours untereinander ausgetauscht werden.
MonitoringForm	*.monitoring	ManagedBean der Seite monitoring.xhtml. Ist für die Aufbereitung der Daten für das Monitoring zuständig und stellt Aktionen bereit, mittels welchen eine laufende Prüfung pausiert oder gestoppt werden

		kann.
BeanFinder	*.util	Hilfsklasse, um von einem Managed Bean auf ein anderes zugreifen zu können. Diese Klasse wird beispielsweise für das Lesen der aktuellen Prüfung innerhalb des Wizard verwendet.
CandidateConverter	*.util	Konverter für ein Kandidaten-Dropdown oder -Autocomplete.
ExaminerConverter	*.util	Konverter für ein Examinatoren-Dropdown oder -Autocomplete.
FormConverter	*.util	Konverter für ein Formular-Dropdown oder -Autocomplete.
StationConverter	*.util	Konverter für ein Posten-Dropdown oder -Autocomplete.
OptionsetConverter	*.util	Konverter für ein Optionset-Dropdown oder -Autocomplete.
CsvReader	*.util	Dem CsvReader kann ein kommasepariertes File übergeben werden, um Kandidaten oder Examinatoren in das System zu importieren.
GrowlMessages	*.util	Hilfsklasse um Info-, Warn- und Fehlermeldung in die JSF Message-Queue zu stellen. Diese werden via Growl angezeigt, oder falls eine ClientID übergeben wird, direkt bei der entsprechenden Komponente eingeblendet.

Tabelle 8 - eosceAdminApp Klassen

6.6.2 eosceAdminDaoInterface

Diese Klassen stellen das Interface der Business-Logik dar.

Klasse	Package	Beschreibung
AssessmentRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein AssessmentEntity bietet.
AssessmentServerRemote	*.daoInterface	Interface, welches Methoden für die Kommunikation zwischen Admin- und Prüfungsserver zur Verfügung stellt.
RotationRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein RotationEntity bietet.
CourseRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein CourseEntity bietet.
AuditStationRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein AuditstationEntity bietet.
FormRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein FormEntity bietet.
PersonRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein PersonEntity bietet.
StationRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein StationEntity bietet.
DeviceRemote	*.daoInterface	Interface, welches Methoden für den Zugriff auf ein DeviceEntity bietet.

Tabelle 9 - eosceAdminDaoInterface Klassen

6.6.3 eosceAdminDao

Implementation der Remote-Interfaces der eosceAdminDaoInterface-Applikation.

Klasse	Package	Beschreibung
AssessmentBean	*.ejb	Implementation des AssessmentRemote-Interface
AssessmentServerBean	*.ejb	Implementation des AssessmentServerRemote-Interface
RotationBean	*.ejb	Implementation des RotationRemote-Interface
CourseBean	*.ejb	Implementation des CourseRemote-Interface
AuditStationBean	*.ejb	Implementation des AuditStationRemote-Interface
FormBean	*.ejb	Implementation des FormRemote-Interface
PersonBean	*.ejb	Implementation des PersonRemote-Interface
StationBean	*.ejb	Implementation des StationRemote-Interface
DeviceBean	*.ejb	Implementation des DeviceRemote-Interface

Tabelle 10 - eosceAdminDao Klassen

6.6.4 eosceAdminDomain

Die Entities bilden jeweils eine Datenbanktabelle ab.

Klasse	Package	Beschreibung
AssessmentEntity	*.domain	Abbildung der Datenbank-Tabelle assessment
RotationEntity	*.domain	Abbildung der Datenbank-Tabelle rotation
CourseEntity	*.domain	Abbildung der Datenbank-Tabelle course
AuditstationEntity	*.domain	Abbildung der Datenbank-Tabelle auditstation
StationEntity	*.domain	Abstrakte Abbildung der Datenbank-Tabelle station
EosceStationEntity	*.domain	Konkrete Abbildung der Datenbank-Tabelle station für den Typ eosce
NonEosceStationEntity	*.domain	Konkrete Abbildung der Datenbank-Tabelle station für den Typ NonEosce
BreakStationEntity	*.domain	Konkrete Abbildung der Datenbank-Tabelle station für den Typ Break
FormEntity	*.domain	Abbildung der Datenbank-Tabelle form
TopicEntity	*.domain	Abbildung der Datenbank-Tabelle topic
QuestionEntity	*.domain	Abbildung der Datenbank-Tabelle question
OptionsetEntity	*.domain	Abbildung der Datenbank-Tabelle optionset
OptionEntity	*.domain	Abbildung der Datenbank-Tabelle option
CriteriaEntity	*.domain	Abbildung der Datenbank-Tabelle criteria
PersonEntity	*.domain	Abstrakte Abbildung der Datenbank-Tabelle person
CandidateEntity	*.domain	Konkrete Abbildung der Datenbank-Tabelle person für den Typ candidate
ExaminerEntity	*.domain	Konkrete Abbildung der Datenbank-Tabelle person für den Typ examiner
UserEntity	*.domain	Abbildung der Datenbank-Tabelle eosce_user
DeviceEntity	*.domain	Abbildung der Datenbank-Tabelle device.

Tabelle 11 - eosceAdminDomain Klassen

6.6.5 eosceAdminTest

In den nachfolgenden Testklassen werden jeweils folgende Testfälle abgedeckt:

- CRUD der zugehörigen Entity
- Alle Methoden aus dem Interface

Klasse	Package	Beschreibung
EosceAdminTestSuite	*.testsuite	Suite, welche alle Testfälle zusammenfasst.
AssessmentRemoteTest	*.testcases	Testklasse für das Dao-Interface AssessmentRemote
AssessmentServerRemoteTest	*.testcases	Testklasse für das Dao-Interface AssessmentServerRemote
AuditStationRemoteTest	*.testcases	Testklasse für das Dao-Interface AuditStationRemote
CourseRemoteTest	*.testcases	Testklasse für das Dao-Interface CourseRemote
DeviceRemoteTest	*.testcases	Testklasse für das Dao-Interface DeviceRemote
FormRemoteTest	*.testcases	Testklasse für das Dao-Interface FormRemote
PersonRemoteTest	*.testcases	Testklasse für das Dao-Interface PersonRemote
RotationRemoteTest	*.testcases	Testklasse für das Dao-Interface RotationRemote
StationRemoteTest	*.testcases	Testklasse für das Dao-Interface StationRemote
CreateTestexam	*.testexam	Erstellen und Überprüfen einer komplett erfassten Prüfung.

Tabelle 12 - eosceAdminTest Klassen

6.7 Sequenzdiagramme

In den nachfolgenden Unterkapiteln werden einige ausgewählte Bereiche der implementierten Remote-Aufrufe mittels Sequenzdiagrammen veranschaulicht. Übersichtshalber wurden nur die wichtigsten Aufrufe in die Diagramme aufgenommen.

6.7.1 Sequenzdiagramm CRUD

Die Abbildung 15 beschreibt das allgemeingültige Vorgehen für das Anlegen, Lesen, Ändern und Löschen einer Entity. In diesem Beispiel wird die CandidateEntity verwendet.

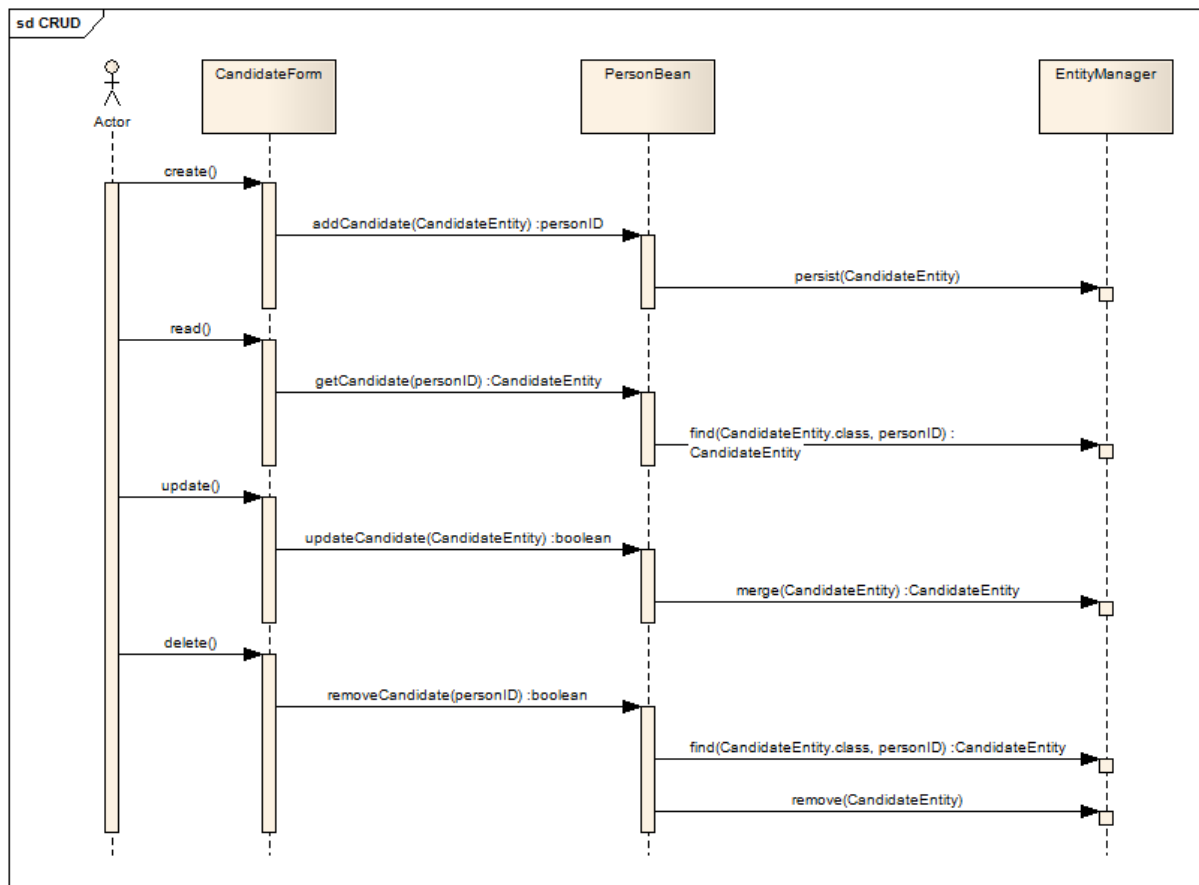


Abbildung 15 - Sequenzdiagramm CRUD

6.7.2 Sequenzdiagramm „Plan Generieren“

Die Abbildung 16 beschreibt die Abläufe, welche durch das Drücken des „Plan generieren“-Buttons im Wizard ausgelöst werden.

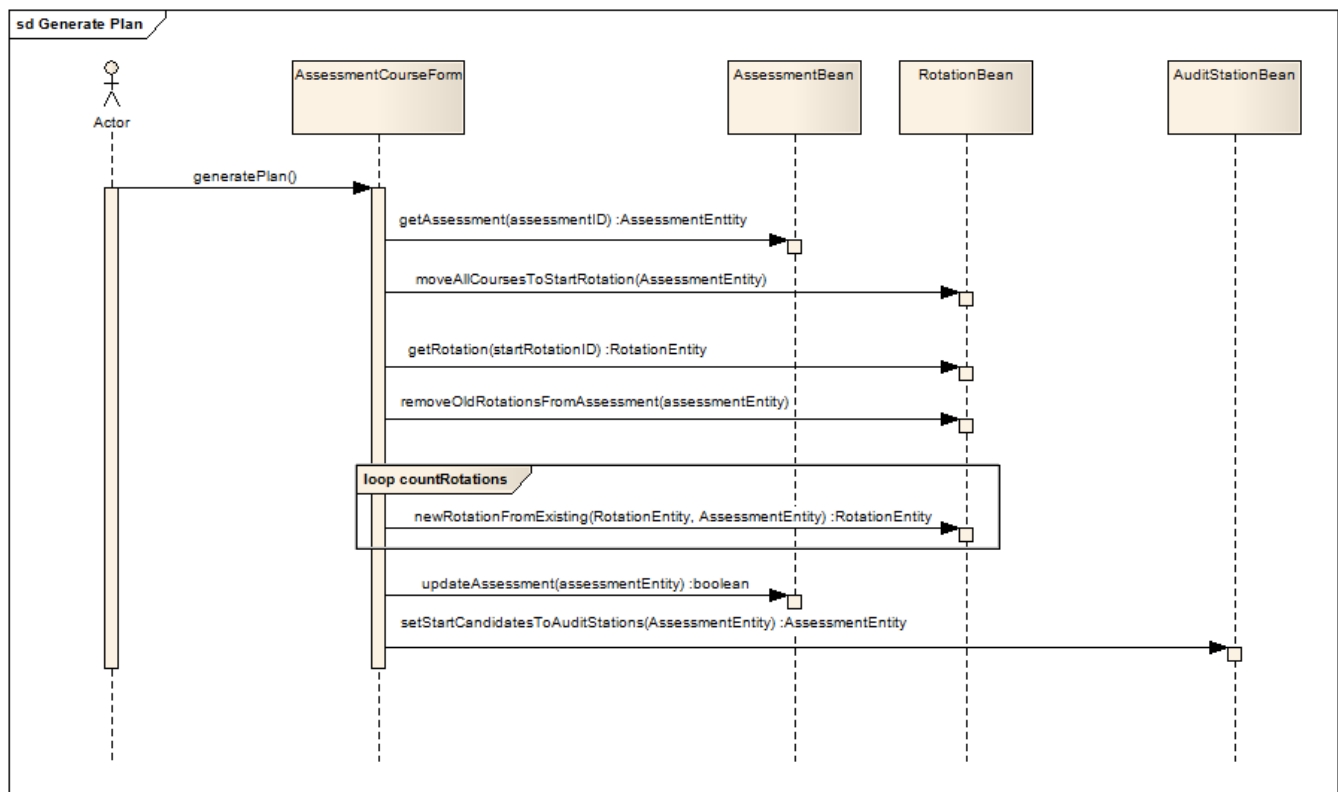


Abbildung 16 - Sequenzdiagramm Plan generieren

6.7.3 Sequenzdiagramm „Export zum Prüfungsserver“

Die Abbildung 17 beschreibt die Abläufe, welche von einem Actor durch das Exportieren einer Prüfung auf den Prüfungsserver ausgelöst werden.

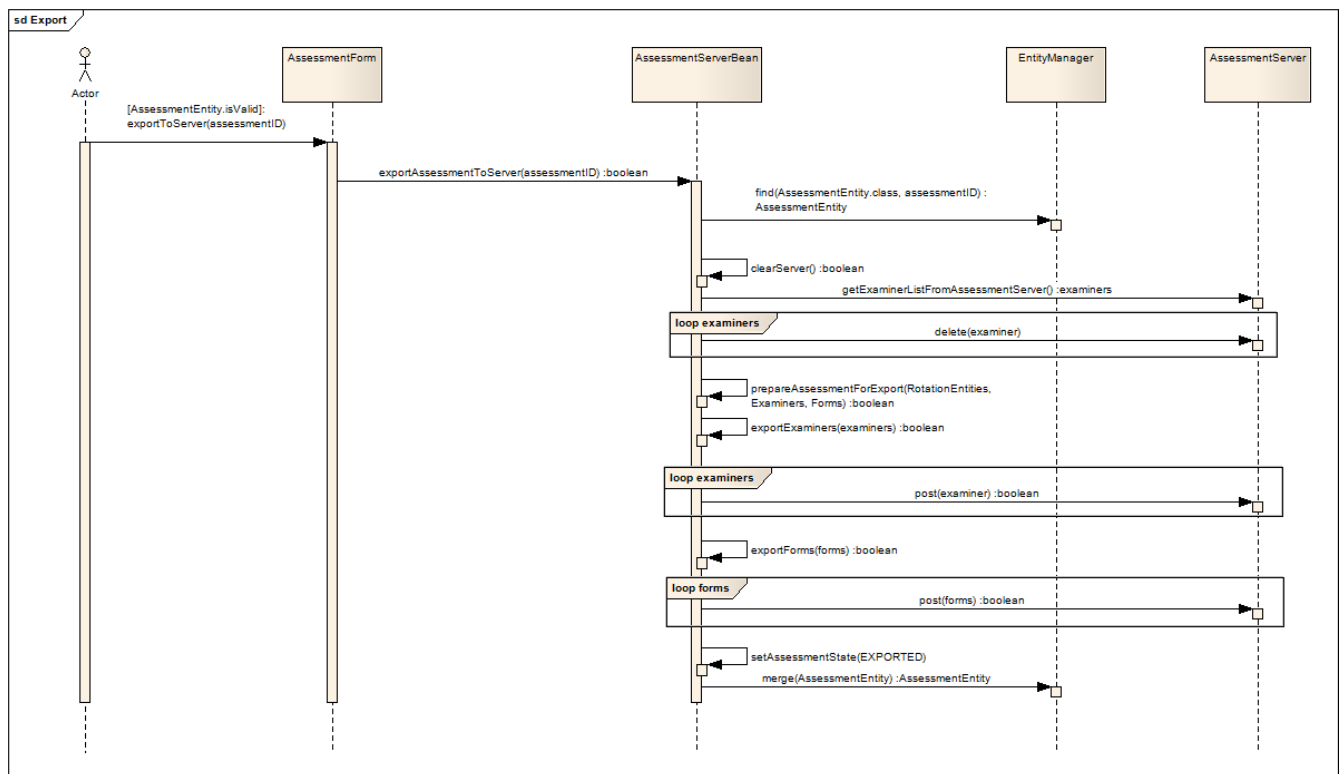


Abbildung 17 - Sequenzdiagramm Export

6.7.4 Sequenzdiagramm Monitor

Die Abbildung 18 illustriert die Abfragen des Monitors an den Prüfungsserver und die Auswertung der zurückgelieferten Log-Files pro Examiner.

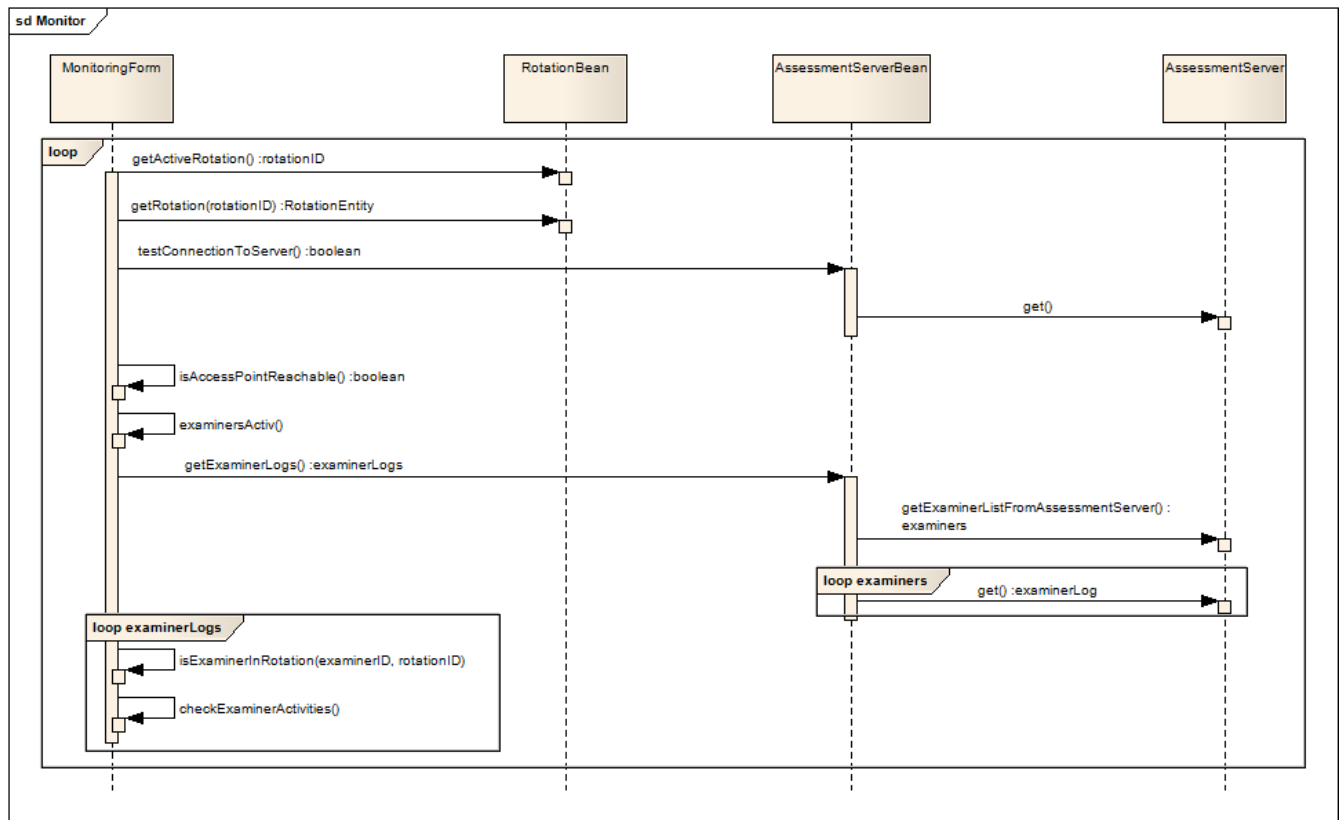


Abbildung 18 - Sequenzdiagramm Monitor

6.8 Schnittstellenstruktur

6.8.1 Datenmodell

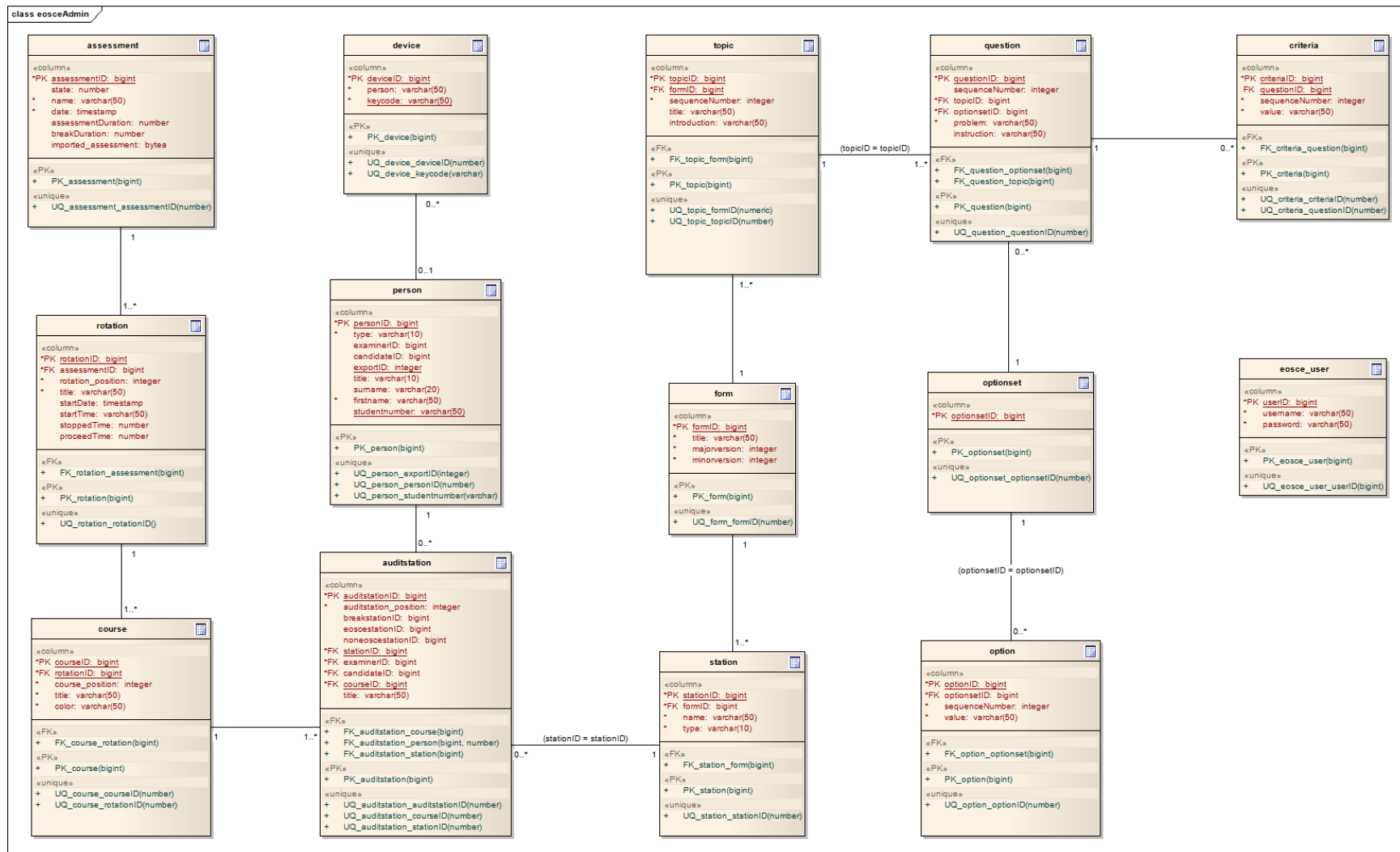


Abbildung 19 - Datenmodell

Tabelle	Beschreibung														
assessment	<i>Datenhaltung:</i> Allgemeine Prüfungsinformationen. <i>Datenfelder:</i> <table> <tr> <td>assessmentID</td><td>Primärschlüssel</td></tr> <tr> <td>name</td><td>Prüfungsname</td></tr> <tr> <td>date</td><td>Prüfungsdatum</td></tr> <tr> <td>assessmentDuration</td><td>Dauer der Prüfung</td></tr> <tr> <td>breakDuration</td><td>Dauer der Pause zum Wechseln</td></tr> <tr> <td>state</td><td>Status der Prüfung: neu – unvollständig erfasst gültig – ohne Fehler erfasst ungültig – mit Fehler erfasst exportiert – an Prüfungsserver übertragen importiert – vom Prüfungsserver importiert gestoppt – Prüfung abgebrochen</td></tr> </table> imported_assessment vom Prüfungsserver importierte Prüfung <i>Beziehung:</i> Jeder gültigen Prüfung ist mindestens eine Rotation zugeordnet.	assessmentID	Primärschlüssel	name	Prüfungsname	date	Prüfungsdatum	assessmentDuration	Dauer der Prüfung	breakDuration	Dauer der Pause zum Wechseln	state	Status der Prüfung: neu – unvollständig erfasst gültig – ohne Fehler erfasst ungültig – mit Fehler erfasst exportiert – an Prüfungsserver übertragen importiert – vom Prüfungsserver importiert gestoppt – Prüfung abgebrochen		
assessmentID	Primärschlüssel														
name	Prüfungsname														
date	Prüfungsdatum														
assessmentDuration	Dauer der Prüfung														
breakDuration	Dauer der Pause zum Wechseln														
state	Status der Prüfung: neu – unvollständig erfasst gültig – ohne Fehler erfasst ungültig – mit Fehler erfasst exportiert – an Prüfungsserver übertragen importiert – vom Prüfungsserver importiert gestoppt – Prüfung abgebrochen														
rotation	<i>Datenhaltung:</i> Informationen zu einer Rotation innerhalb einer Prüfung. <i>Datenfelder:</i> <table> <tr> <td>rotationID</td><td>Primärschlüssel</td></tr> <tr> <td>assessmentID</td><td>Fremdschlüssel</td></tr> <tr> <td>rotation_position</td><td>Position innerhalb der Prüfung</td></tr> <tr> <td>title</td><td>Titel der Rotation</td></tr> <tr> <td>startDate</td><td>Prüfungsdatum (falls Prüfung über mehrere Tage durchgeführt wird)</td></tr> <tr> <td>startTime</td><td>Startzeit der Rotation</td></tr> <tr> <td>stoppedTime</td><td>Timestamp, falls die Prüfung via Monitorclient angehalten werden muss.</td></tr> </table>	rotationID	Primärschlüssel	assessmentID	Fremdschlüssel	rotation_position	Position innerhalb der Prüfung	title	Titel der Rotation	startDate	Prüfungsdatum (falls Prüfung über mehrere Tage durchgeführt wird)	startTime	Startzeit der Rotation	stoppedTime	Timestamp, falls die Prüfung via Monitorclient angehalten werden muss.
rotationID	Primärschlüssel														
assessmentID	Fremdschlüssel														
rotation_position	Position innerhalb der Prüfung														
title	Titel der Rotation														
startDate	Prüfungsdatum (falls Prüfung über mehrere Tage durchgeführt wird)														
startTime	Startzeit der Rotation														
stoppedTime	Timestamp, falls die Prüfung via Monitorclient angehalten werden muss.														

	proceedTime Timestamp, falls eine pausierte Prüfung fortgesetzt werden muss. <i>Beziehung:</i> Einer gültigen Rotation ist immer mindestens ein Course zugeordnet.
course	<i>Datenhaltung:</i> Informationen zu einem Parcours innerhalb einer Rotation. <i>Datenfelder:</i> courseID Primärschlüssel rotationID Fremdschlüssel course_position Position innerhalb der Rotation title Titel des Parcour color Farbe des Parcour <i>Beziehung:</i> Einem gültigen Parcours ist immer mindestens eine Auditstation zugeordnet.
auditstation	<i>Datenhaltung:</i> Zuweisungstabelle für einen Prüfungsposten. <i>Datenfelder:</i> auditstationID Primärschlüssel courseID Fremdschlüssel stationID Fremdschlüssel – im System erfasste Station examinerID Fremdschlüssel – Examiner candidateID Fremdschlüssel - Startkandidat auditstation_position Position innerhalb des Parcours title Titel des Prüfungsposten <i>Beziehung:</i> Auf der Tabelle Auditstation werden die Beziehung zwischen dem Examiner, dem Startkandidaten und der im System erfassten Station abgebildet.

station	<i>Datenhaltung:</i> Vorlage eines Prüfungsposten. <i>Datenfelder:</i> <table> <tr> <td>stationID</td><td>Primärschlüssel</td></tr> <tr> <td>formID</td><td>Fremdschlüssel</td></tr> <tr> <td>name</td><td>Name Prüfungsstation</td></tr> <tr> <td>type</td><td>Art der Station eosce – E-OSCE Posten mit einem Formular NonEosce – nicht E-OSCE Posten (z.B. PC) Break - Pausenposten</td></tr> </table> <i>Beziehung:</i> Handelt es sich um einen eosce Posten, ist diesem ein Prüfungsformular zugewiesen.	stationID	Primärschlüssel	formID	Fremdschlüssel	name	Name Prüfungsstation	type	Art der Station eosce – E-OSCE Posten mit einem Formular NonEosce – nicht E-OSCE Posten (z.B. PC) Break - Pausenposten
stationID	Primärschlüssel								
formID	Fremdschlüssel								
name	Name Prüfungsstation								
type	Art der Station eosce – E-OSCE Posten mit einem Formular NonEosce – nicht E-OSCE Posten (z.B. PC) Break - Pausenposten								
form	<i>Datenhaltung:</i> Allgemeine Daten eines Prüfungsformular. <i>Datenfelder:</i> <table> <tr> <td>formID</td><td>Primärschlüssel</td></tr> <tr> <td>title</td><td>Name des Formular</td></tr> <tr> <td>majorversion</td><td>Hauptversion des Formular</td></tr> <tr> <td>minorversion</td><td>Entsteht durch das Kopieren einer Majorversion.</td></tr> </table> <i>Beziehung:</i> Ein Formular besteht aus mindestens einem Bereich.	formID	Primärschlüssel	title	Name des Formular	majorversion	Hauptversion des Formular	minorversion	Entsteht durch das Kopieren einer Majorversion.
formID	Primärschlüssel								
title	Name des Formular								
majorversion	Hauptversion des Formular								
minorversion	Entsteht durch das Kopieren einer Majorversion.								
topic	<i>Datenhaltung:</i> Allgemeine Daten eines Bereich innerhalb eines Prüfungsformular. <i>Datenfelder:</i> <table> <tr> <td>topicID</td><td>Primärschlüssel</td></tr> <tr> <td>formID</td><td>Fremdschlüssel</td></tr> <tr> <td>title</td><td>Titel des Bereich</td></tr> <tr> <td>introduction</td><td>Einführungstext zum Bereich</td></tr> </table>	topicID	Primärschlüssel	formID	Fremdschlüssel	title	Titel des Bereich	introduction	Einführungstext zum Bereich
topicID	Primärschlüssel								
formID	Fremdschlüssel								
title	Titel des Bereich								
introduction	Einführungstext zum Bereich								

	sequenceNumber Position innerhalb des Formulars. <i>Beziehung:</i> Ein Topic beinhaltet mindestens eine Frage.
question	<i>Datenhaltung:</i> Allgemeine Daten einer Frage innerhalb eines Bereichs. <i>Datenfelder:</i> questionID Primärschlüssel topicID Fremdschlüssel optionsetID Fremdschlüssel problem Fragestellung instruction Anweisung zur Frage sequenceNumber Position innerhalb des Formulars. <i>Beziehung:</i> Einer Frage können sowohl Kriterien als auch ein Set von Optionen zugewiesen werden.
criteria	<i>Datenhaltung:</i> Kriterium einer Frage. <i>Datenfelder:</i> criterialID Primärschlüssel questionID Fremdschlüssel value Kriterium sequenceNumber Position innerhalb Kriterienliste der Frage. <i>Beziehung:</i> Keine
optionset	<i>Datenhaltung:</i> Optionen, welche in einem Set zusammengefasst werden und somit für weitere Fragen wiederverwendet werden können. <i>Datenfelder:</i> optionsetID Primärschlüssel <i>Beziehung:</i>

	Zuweisungstabelle, welche mehrere Optionen zu einem Set abbildet.												
option	<i>Datenhaltung:</i> Eine Option einer Frage. <i>Datenfelder:</i> <table> <tr> <td>optionID</td><td>Primärschlüssel</td></tr> <tr> <td>optionsetID</td><td>Fremdschlüssel</td></tr> <tr> <td>value</td><td>Option</td></tr> <tr> <td>sequenceNumber</td><td>Position innerhalb des Set</td></tr> </table> <i>Beziehung:</i> Keine	optionID	Primärschlüssel	optionsetID	Fremdschlüssel	value	Option	sequenceNumber	Position innerhalb des Set				
optionID	Primärschlüssel												
optionsetID	Fremdschlüssel												
value	Option												
sequenceNumber	Position innerhalb des Set												
person	<i>Datenhaltung:</i> Personalien der Kandidaten und Examinatoren. <i>Datenfelder:</i> <table> <tr> <td>personID</td><td>Primärschlüssel</td></tr> <tr> <td>type</td><td>Art der Person examiner – Examiner candidate - Kandidat</td></tr> <tr> <td>title</td><td>Anrede (nur falls type examiner)</td></tr> <tr> <td>surname</td><td>Nachname</td></tr> <tr> <td>firstname</td><td>Vorname</td></tr> <tr> <td>studentnumber</td><td>Studentennummer (nur falls type candidate)</td></tr> </table> <i>Beziehung:</i> Sowohl der Kandidat als auch der Examiner werden einer Auditstation zugewiesen. Der Examiner betreut diese Auditstation und für den Kandidaten stellt dies den Startposten der Prüfung dar.	personID	Primärschlüssel	type	Art der Person examiner – Examiner candidate - Kandidat	title	Anrede (nur falls type examiner)	surname	Nachname	firstname	Vorname	studentnumber	Studentennummer (nur falls type candidate)
personID	Primärschlüssel												
type	Art der Person examiner – Examiner candidate - Kandidat												
title	Anrede (nur falls type examiner)												
surname	Nachname												
firstname	Vorname												
studentnumber	Studentennummer (nur falls type candidate)												
device	<i>Datenhaltung:</i> Gerät, welches einem Experten oder Prüfungsverantwortlichen zugewiesen wird. <i>Datenfelder:</i> <table> <tr> <td>deviceId</td><td>Primärschlüssel</td></tr> </table>	deviceId	Primärschlüssel										
deviceId	Primärschlüssel												

	person Name der Person keycode Gerätekenung <i>Beziehung:</i> Keine, da auch nicht im System erfasste Personen ein Gerät erhalten können.
eosce_user	<i>Datenhaltung:</i> Benutzer, um sich am E-OSCE Admin anzumelden. <i>Datenfelder:</i> userID Primärschlüssel username Benutzername passwort Passwort <i>Beziehung:</i> Keine

Tabelle 13 - Datenbanktabellen

6.8.3 Import/Export

6.8.3.1 Kandidaten-Import

Der Kandidaten-Import mittels eines CSV-Files realisiert, welches folgendes Format haben muss:

Studentennummer;Vorname;Nachname

6.8.3.2 Examinatoren-Import

Wie die Kandidaten werden auch die Examinatoren über ein CSV-File importiert, welches folgendes Format haben muss:

Anrede;Vorname;Nachname

6.8.3.3 Prüfungsexport

Die abgeschlossene Prüfung wird als Zip-Datei vom Prüfungsserver importiert und in der DB abgespeichert. Solange die Prüfung auf dem Administrationsserver nicht gelöscht wird, kann sie heruntergeladen werden.

7 Tests

Rev.	Datum	Wer	Änderung
0.1	09.05.2011	Matthias Good	Dokument erstellt
0.2	10.05.2011	Matthias Good	Tests hinzugefügt
0.3	08.06.2011	Matthias Good	Review

7.1.1 Zweck

Dieses Dokument beschreibt das Testverfahren für die aufgetretenen Performance-Probleme für die Bachelorarbeit "Administration von E-OSCE Prüfungen".

7.1.2 Gültigkeitsbereich

Dieses Dokument gilt als Grundlage für die ganze Bachelorarbeit und hat deshalb Gültigkeit über die gesamte Arbeitsdauer.

7.1.3 Übersicht

Zu Beginn wird der Testaufbau, resp. die Testdurchführung genauer erläutert und danach folgen die einzelnen Testreihen. Zum Schluss gibt es noch einen Gesamtüberblick über alle Tests.

7.2 Integrations-/Performancetest Sprint 2 (und 3)

Nachdem dem Abschluss aller Tasks des zweiten Sprints, führte das Team einen Integrationstest mit dem bereits im ersten Sprint erstellten System durch. Natürlich wurde bereits während der Implementation getestet, dass sich die neuen Funktionen in das bestehende System integrieren lassen, jedoch sollte hier nun ein erstes Mal der „Real“-Fall getestet werden. Das Ziel dieses Integrations-/Performanztests ist es, die optimale Abstimmung zwischen den im ersten Sprint entwickelten Datenfunktionen (CRUD Kandidaten/Posten) und der Berechnung der optimalen Anzahl der Parcours (inkl. Darstellung dieser Tabelle) zu finden.

7.2.1 Testaufbau /-durchführung

Bei jedem Testdurchgang wurde folgendermassen vorgegangen:

1. Anzahl Posten erstellen
2. Anzahl Kandidaten hinzufügen
3. Neue Prüfung erstellen
4. Posten der Prüfung hinzufügen
5. Kandidaten der Prüfung hinzufügen
6. Optimale Anzahl an Parcours berechnen lassen

Die ersten fünf Punkte wurden bereits im ersten Sprint realisiert und der letzte Punkt im zweiten (resp. dritten) Sprint abgeschlossen.

Pro Testreihe wurden jeweils mehrere Tests mit einer unterschiedlichen Anzahl an Posten und Kandidaten durchgeführt. Eine Testreihe sollte folgende Resultate liefern:

- Welche Funktionen enthalten eventuell noch Fehler?
- Wie performant sind die einzelnen Funktionen?

- Wie performant ist das komplette Projekt?

Alle getesteten Funktionen stammen aus der Berechnung der optimalen Anzahl an Parcours. Das folgende Diagramm dient als kleiner Überblick über diese Funktionen (und ist keinesfalls einem richtigen Sequenzdiagramm gleichzusetzen):

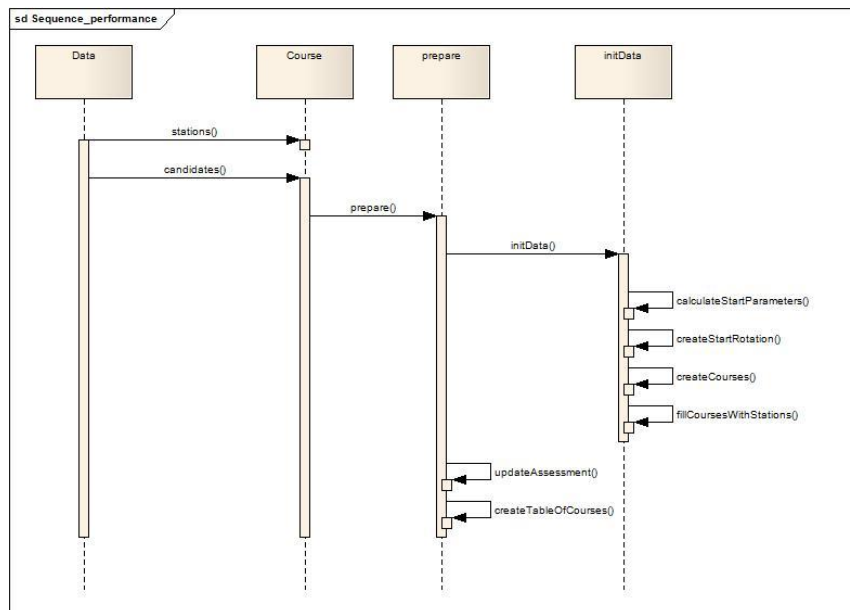


Abbildung 21 - Überblick Berechnung Parcours

7.2.2 Testreihe 1

Die erste Testreihe wurde zuerst mit fünf Posten und zehn Kandidaten durchgeführt, das Ergebnis war nicht gerade überzeugend: nach dem Drücken des „Weiter“-Button zur Berechnung der optimalen Anzahl an Parcours dauerte es knapp fünf Sekunden bis die Parcours auf dem nächsten View in der Tabelle angezeigt wurden. Bei der nächsten Durchführung wurde gleich auf zehn Posten und 150 Kandidaten erhöht und das Resultat war ernüchternd, nach ca. zehn Minuten wurde der Versuch erfolglos abgebrochen.

Das Problem wurde nach kurzer Analyse gefunden: die „prepare()“-Methode wurde fälschlicherweise mehrmals aufgerufen und erzeugte somit jede Menge Daten im Hintergrund.

7.2.3 Testreihe 2

Das Resultat der zweiten Testreihe war schon um einiges besser, aber immer noch weit weg von einer akzeptablen Performanz.

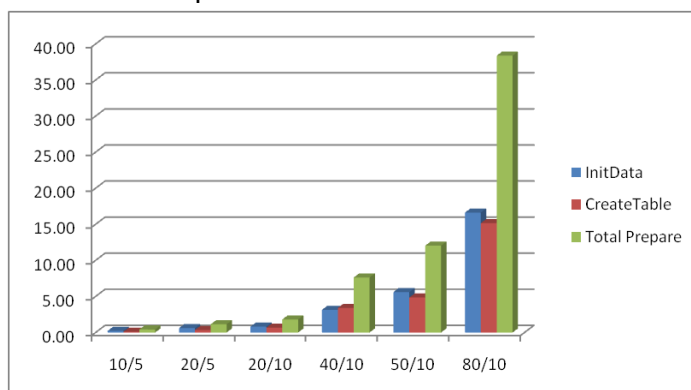


Abbildung 22 - Testreihe 2

Bei 80 Kandidaten und zehn Posten benötigt die komplette „prepare()“-Methode noch knapp 40 Sekunden, davon ca. 30 Sekunden für das Initialisieren der Daten und Erstellen der Tabelle. Somit bleibt immer noch ein Rest von ca. zehn Sekunden und das Problem war schnell gefunden: die Aktualisierung der Prüfung (updateAssessment()).

7.2.4 Testreihe 3a

Aufgrund der letzten Testserie wurde als Erstes die „updateAssessment()“-Funktion untersucht. Diese setzt sich aus drei DB-Abfragen zusammen: merge, flush und get.

Hier war nun gut zu sehen, dass das Updaten der Prüfung selbst fast keine Zeit beansprucht, jedoch das Holen der aktuellen Prüfung aus der DB exponentiell zu den Kandidaten zeitlich ansteigt. Dieses Phänomen war zu diesem Zeitpunkt noch unerklärlich, sodass an der aktuellen Implementation im Moment nichts geändert wurde.

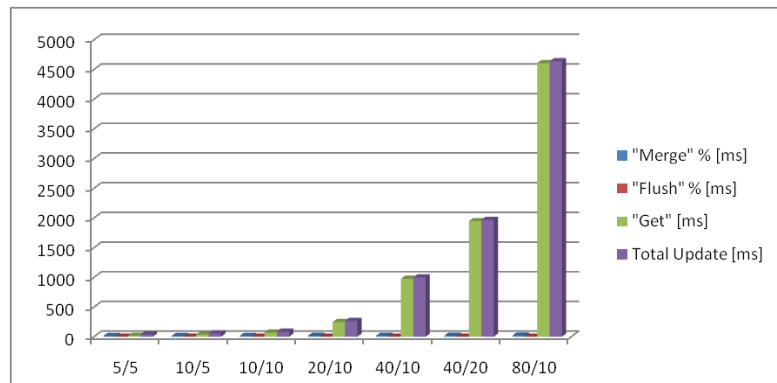


Abbildung 23 - Testreihe 3a

7.2.5 Testreihe 3b

Als Nächstes wurde der Aufbau der Parcours-Tabelle unter die Lupe genommen. Dort fielen die vielen DB-Zugriffe auf, welche sogleich auf ein Minimum reduziert wurden.

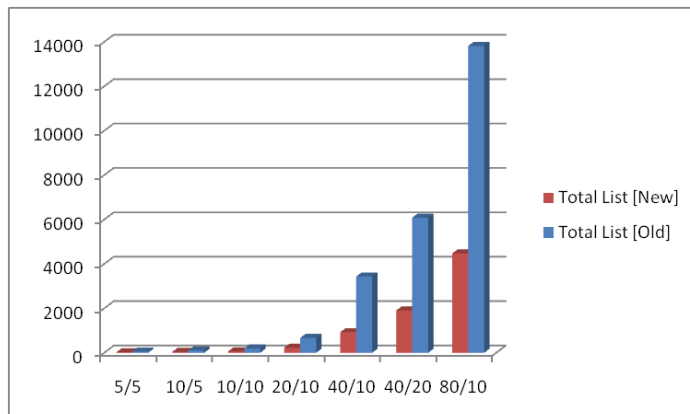


Abbildung 24 - Testreihe 3b

Die Performanz dieser Funktion konnte durch diese Änderungen enorm gesteigert werden, sodass diese nun in ca. einem Drittel der Zeit abläuft. Mit knapp über vier Sekunden für 80 Kandidaten liegt sie aber immer noch klar über der Schmerzengrenze.

7.2.6 Testreihe 4

In einem nächsten Schritt waren die einzelnen DB-Zugriffe an der Reihe. Diese wurden optimiert, wo es möglich war (z.B. Ersetzen der find()-Methode des Entity-Managers durch eigene „select“-Abfrage) und das Resultat war nochmals ein Performance-Boost.

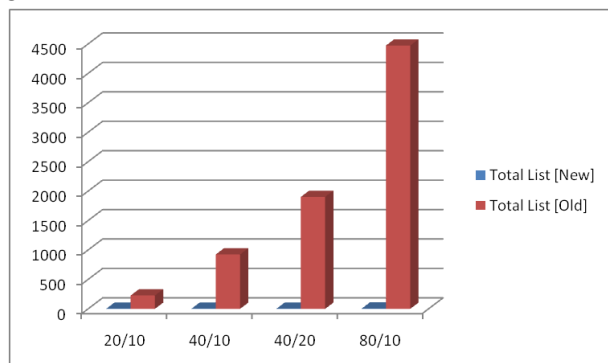


Abbildung 25 - Testreihe 4 List

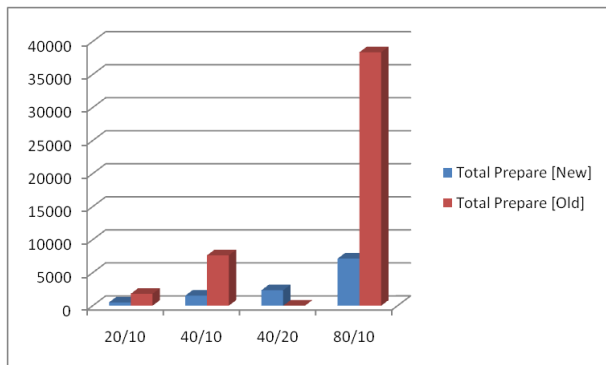


Abbildung 26 - Testreihe 4 Prepare

Lediglich das Aktualisieren der Prüfung dauerte nun noch verhältnismässig lange.

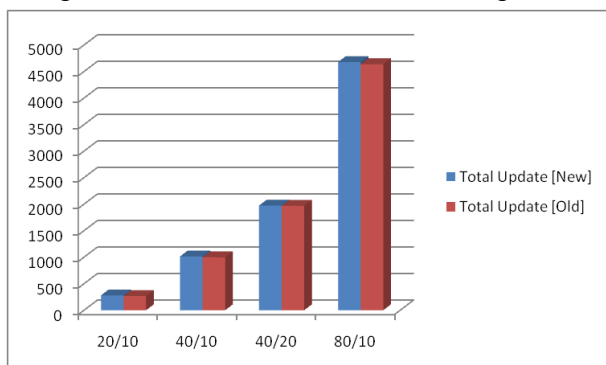


Abbildung 27 - Testreihe 4 Update

7.2.7 Testreihe 5

Ein bisschen mehr als fünf Sekunden waren zwar schon einmal eine klare Verbesserung, aber so viel Wartezeit wäre zukünftigen Benutzern nicht zumutbar.

Die Datenbank-Zugriffe wurden ein weiteres Mal untersucht und nach längerer Analyse-Zeit wurden zwei Optimierungen gefunden, welche sich extrem auf die Performance auswirkten:

- Das Setzen der Fetching Strategie auf „select“. Hibernate unterstützt mehrere Strategien um die von Hibernate generierten „select-Statements“ zu optimieren. Die gewählte „select“-Strategie bewirkt ein „Lazy loading“ aller Collections und Entities.
- Das Setzen der Batch-Size. Diese gibt die Anzahl Collections oder Entities an, welche zusammen in einem Select-Statement ausgeführt werden sollen (ansonsten wird für jedes Entity ein eigene select-Abfrage generiert)

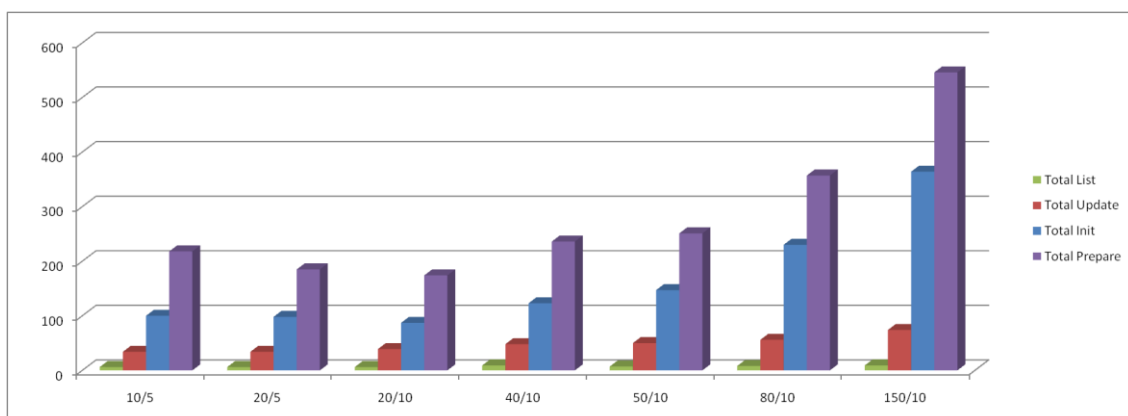


Abbildung 28 - Testreihe 5

Das Ergebnis ist nun endlich zufriedenstellend. Für einen realen Datensatz (150 Kandidaten / zehn Posts) benötigt die Berechnung der optimalen Anzahl an Parcours nun nur noch ca. eine halbe Sekunde.

7.3 Gesamtüberblick

Der Mehraufwand hat sich auf jeden Fall gelohnt, die Performance-Steigerung ist wie unten ersichtlich enorm.

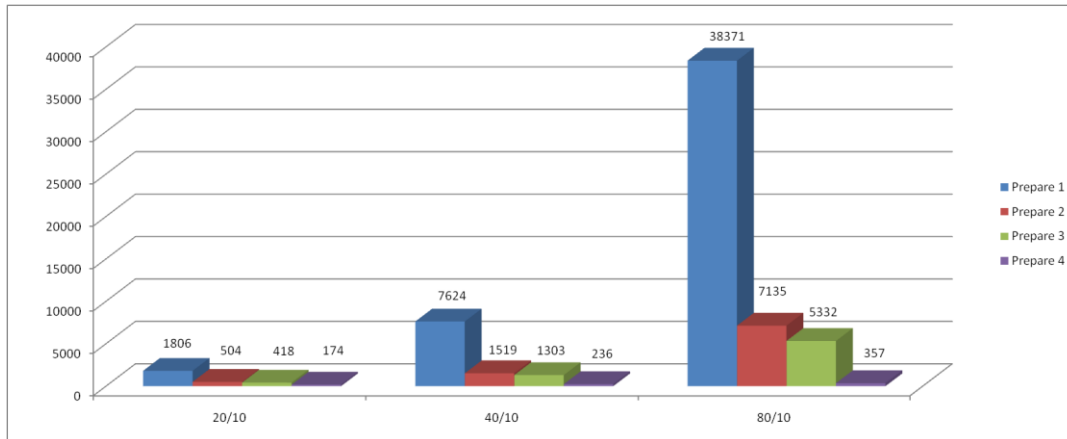


Abbildung 29 - Gesamtüberblick Tests

Folgende Optimierungen haben zu diesem Resultat geführt:

- Minimieren der Datenbank-Zugriffe in den einzelnen Funktionen
- Optimieren des restlichen Funktionscode (Refactoring)
- Verwenden von vorhandenen Datenbank-Optimierungen (fetch und batch-size)

8 Projektmanagement

Rev.	Datum	Wer	Änderung
0.1	09.03.2011	Matthias Good	Dokument erstellt
0.2	11.03.2011	Matthias Good	Alle Themen bearbeitet
0.3	13.06.2011	Matthias Good	Review

8.1.1 Zweck

Dieses Dokument beschreibt das Projektmanagement für die Bachelorarbeit "Administration von E-OSCE Prüfungen".

8.1.2 Gültigkeitsbereich

Dieses Dokument gilt als Grundlage für die ganze Bachelorarbeit und hat deshalb Gültigkeit über die gesamte Arbeitsdauer.

8.1.3 Übersicht

Das Dokument Projektmanagement gibt einen Überblick über die Organisation des Projektes, die Managementabläufe und die Infrastruktur. Ausserdem werden die Qualitätsmassnahmen aufgelistet sowie die Risikomanagement aufgezeigt.

8.2 Projektorganisation

8.2.1 Projektmitarbeiter

Das Team besteht aus drei einander gleichgestellten Mitgliedern, Prof. Hans Rudin fungiert dabei als Projektbetreuer.

Philipp Marugg	Sven Lenz	Matthias Good
		

Tabelle 14 - Projektmitarbeiter

Das Team hat sich folgende Zeiten für das Projekt reserviert:

Dienstag:	08:10 – 16.50
Mittwoch:	08:10 – 16.50
Freitag:	08.10 – 15.10

8.2.2 Externe Schnittstellen

Die Bachelorarbeit wird von Prof. Hans Rudin (IFS Rapperswil) betreut. Als weitere Betreuer auf Seiten des IFS stehen Mischa Trecco und Kevin Gaunt zur Verfügung.

8.3 Managementabläufe

8.3.1 Projektkostenvoranschlag

Der Bachelorarbeit stehen pro Teammitglied ca. 22.5h/Woche zur Verfügung. Dies wird aus folgender Rechnung ersichtlich:

$12 \text{ ECTS} \times 30\text{h} / 16 = 22.5 \text{ Stunden/Woche}$

Der Projektstart wurde auf den 21. Februar 2011 festgelegt und der Abgabetermin auf den 17. Juni 2011. Sollten während der Durchführungen unerwartete Probleme auftreten, kann die Arbeitszeit pro Woche und Teammitglied angehoben werden. Zusätzlich kann die Ferienwoche während des Semesters dazu genutzt werden, um eventuelle Probleme zu beheben.

8.3.2 Projektplan

Der geplante Ablauf des Projekts ist in der Excel-Datei „Projektplan.xls“ angehängt [ANH05].

8.3.3 Vorgehen nach Scrum & RUP

Das Team entscheidet sich wie bereits bei der Studienarbeit für ein gemischtes Vorgehensmodell.

Bei der Implementation wird das Vorgehensmodell *Scrum* [11] verwendet. In der Anforderungsanalyse werden die erstellten User Stories im Product Backlog [ANH06] aufgenommen, dabei agiert das komplette Team sowohl als Product Owner, als auch als Scrum Master. Vor jedem Sprint findet ein Scrum Meeting statt (Zusammensetzung aus Planning Meeting und Sprint Review Meeting), in welchem einerseits die erreichten Ziele und allenfalls die aufgetretenen Probleme festgehalten werden und andererseits über die Inhalte des nächsten Sprints entschieden werden. Dabei werden die für den nächsten Sprint geplanten User Stories zum entsprechenden Sprint-Backlog [ANH07-11] hinzugefügt. Als Fortschrittskontrolle pro Sprint wird ein Burndown-Chart [ANH07-11] verwendet, welches den Planungs- und Realfall aufzeigt. Es sind insgesamt fünf Sprints geplant, welche jeweils zwei Wochen dauern.

Bei der Dokumentation wird nach *RUP* (Rational Unified Process) [12] vorgegangen. In der Inception-Phase wird die wesentliche Funktionalität des Projektes definiert, der Projektplan evaluiert, sowie die Risiken identifiziert. In der darauffolgenden Elaboration-Phase wird die Anforderungsanalyse erstellt. In der Construction-Phase, welche zeitgleich mit den fünf Sprints abläuft, wird zu Beginn die Software Architektur dokumentiert und gegen Ende eine Benutzerdokumentation erstellt. Ausserdem werden hier der technische Bericht und das Management Summary geschrieben. Die letzte Phase stellt die Transition dar, in welcher die Zeitauswertung und die persönlichen Erfahrungsberichte erstellt werden. Zudem werden alle Dokumente einem Review unterzogen, bevor sie am Abgabetermin dem Betreuer übergeben werden.

8.3.4 Besprechungen

Das Team trifft sich einmal in der Woche mit dem Betreuer zu einem Meeting, bei dem der aktuelle Projektstand und allfällige Probleme besprochen werden. Die Besprechung findet jeweils jeden Mittwochnachmittag statt, sofern dafür Bedarf besteht. Die beiden weiteren Betreuer, Mischa Trecco und Kevin Gaunt nehmen an diesen Besprechungen ebenfalls teil.

8.4 Infrastruktur

8.4.1 Räumlichkeiten

Die wöchentlichen Besprechungen finden jeweils in einem Sitzungszimmer der HSR statt. Das Team arbeitet entweder im vorgesehenen Bachelorarbeitszimmer 1.262 oder von Zuhause aus (dies ist von der Arbeitssituation abhängig).

8.4.2 Hardware

- HSR-Arbeitsrechner im Bachelorarbeitszimmer 1.262 der HSR
- Private Computer der Teammitglieder
- Versionsverwaltungsserver der HSR

8.4.3 Software

Betriebssysteme	Microsoft Windows 7 Enterprise
Programmiersprache	Java 6
Entwicklungsumgebung	Eclipse EE Helios
Versionsverwaltung	SVN, Subclipse 1.6.2
Datenbank	PostgreSQL 9.0
Server	JBoss 6.0
Dokumentation	Microsoft Office 2007 Atlassian Confluence 3.3.3

Tabelle 15 - Software

8.4.4 Versionsverwaltung

Als Versionsverwaltung für den Code wird der SVN-Server der HSR verwendet. Das Team verwendet dazu das Eclipse-Plugin Subclipse [13].

8.4.5 Kommunikation

Das Team verwendet für den internen Austausch von Dokumenten das Programm „Dropbox“ [14]. Für den Austausch mit den Betreuern und auch zur Unterstützung von Besprechungen wird mit einem Wiki (Atlassian Confluence [15]) gearbeitet.

8.5 Qualitätsmassnahmen

8.5.1 Dokumente

Das Team sorgt dafür, dass die Dokumente immer auf dem aktuellen Stand sind, somit kann eine hohe Qualität erreicht werden. Zusätzlich werden die Dokumente von einem weiteren Teammitglied sowie einem Betreuer gegengelesen und gegebenenfalls korrigiert.

8.5.2 Sitzungen

Für die wöchentlichen Sitzungen mit dem Betreuer wird jeweils im Vorfeld eine Traktandenliste mit den zu besprechenden Punkten erstellt. Während den Sitzungen protokolliert ein Teammitglied das Besprochene und stellt das daraus resultierende Sitzungsprotokoll auf dem Wiki zur Einsicht zur Verfügung. Somit können mögliche Missverständnisse schnell aus dem Weg geräumt werden.

8.5.3 Heuristische Evaluation

Für die Beurteilung der Benutzeroberfläche (Gebrauchstauglichkeit) werden in der Analyse-Phase Heuristische Evaluationen mit Experten durchgeführt. Anhand des daraus gewonnenen Feedbacks, können Usability-Probleme schon früh während des Projektes erkannt und behoben werden.

8.5.4 Tests

8.5.4.1 Unit-Tests

Während der Implementierung werden wo möglich und sinnvoll Unit-Tests durchgeführt. Diese Tests werden mithilfe von „JUnit 4“ [16] erstellt und müssen erfolgreich durchlaufen, bevor der Quellcode eingecheckt werden darf.

8.5.4.2 System-/Performance-Tests

Am Ende jedes Sprints führt das Team System-Tests durch, bei welchen die neu implementierten Anforderungen getestet werden. Dabei wird das Projekt anhand von Testdaten auf Fehler überprüft.

Zudem werden zwischendurch Performance-Tests durchgeführt, wo die Applikation mit ähnlichen Daten wie in der Produktivumgebung gefüllt wird und dann die Performanz gemessen wird.

8.6 Risikomanagement

8.6.1 Risiken

In der folgenden Tabelle sind Risiken aufgelistet, welche während der Projektlaufzeit auftreten können. Zudem sind darin die Auswirkungen der jeweiligen Risiken sowie deren Massnahme beim Eintreten dargestellt.

ID	Risiko	Auswirkungen	Massnahme	Kosten des Schadens [Std]	Wahrscheinlichkeit des Eintreffens	Gewichteter Schaden [Std]	Priorität
1	Ausfall des SVN-Servers	Ein Teil der Arbeit geht verloren	Backup des SVN-Servers und tägliches Backup auf eine externe Festplatte	50	20%	10	Hoch
2	Umsetzung der Paper-Prototype nicht möglich wegen technischen Einschränkungen	Anpassen der Requirements	Anhand des Prototyps mögliche Probleme entfernen	50	20%	10	Mittel
3	Requirements auf Seiten des IML ändern sich	Anforderungen neu anpassen und erarbeiten	Regelmässige Sitzungen / Reserven einbauen	40	20%	8	Hoch

4	Ausfall Arbeitsstation	HW eines Projektmitglieds fällt aus	Mehrere Arbeitsplätze (an der Schule und zuhause)	4	5%	2	Tief
5	Ausfall eines Projektmitglieds	Die Arbeit des Projektmitglieds kann nicht ausgeführt werden	Arbeiten innerhalb des Teams aufteilen	200	5%	10	Tief
6	Fehleinschätzungen des Aufwandes	Es entsteht mehr Arbeitszeit als geplant	Aussortieren gewisser Funktionen in den Sitzungen	30	10%	3	Mittel
7	Probleme mit den verwendeten Technologien	Die Einarbeitung in eine Technologie dauert länger als geplant	Viel Zeit in Einarbeitung einplanen. (mit bekannten Techn. arbeiten)	30	10%	3	Mittel
8	Für das UI ist eine komplizierte Drag & Drop Lösung gewünscht, was in einer Web-Applikation nicht einfach Umzusetzen ist	Weitere Arbeiten verzögern sich.	Prototype des UI erstellen um Möglichkeiten abzuklären.	150	50%	75	Hoch
9	Performanceprobleme bei der Datenbankprogrammierung	Weitere Arbeiten verzögern sich.	Regelmässige Perfor-mancetests am Ende jedes Sprints	30	30%	10	Hoch
Summe				554	140%	121	

Tabelle 16 - Risikoplan

8.6.2 Hinzugefügte Risiken

- *Risiko 8*
Das Risiko wurde in der Woche vier nach dem Fertigstellen der finalen Paperprototypes aufgenommen, da ein komplexes Web-GUI mit Drag & Drop geplant war.
- *Risiko 9*
Das Risiko wurde in der Woche neun hinzugefügt, nachdem Performanceprobleme bei der Erstellung von Prüfungen mit vielen Kandidaten und Posten aufgetreten sind.

8.6.3 Eintretene Risiken und Massnahmen

- *Risiko 3*
Während den wöchentlichen Meetings sind neue Anforderungen seitens des IML und IFS an die E-OSCE Adminkomponente gestellt worden. Als Massnahme wurden die User Stories im Product Backlog durch das Team neu priorisiert und die Priorisierung mit den Betreuern abgesprochen. Dabei wurden die Stories mit den niedrigsten Prioritäten in den letzten Sprint verschoben und als optionale Features definiert. Diese Features sollen nur bei noch vorhandener Zeit implementiert werden.
- *Risiko 8*
Durch das komplexe GUI wurde ein Prototype basierend auf JSF und Primefaces für die Rotationsübersicht entworfen. Da der Prototype die gegebenen Anforderungen nicht genügend abdeckte beziehungsweise auch Probleme mit dem Drag & Drop bei der Erstellung der Formulare aufgetreten sind, musste während den Sprints 2 und 3 zusätzliche Zeit für die Implementation am Wochenende aufgewendet werden.
- *Risiko 9*
Während den Abschlusstests des dritten Sprints wurden massive Performanceprobleme beim Lesen und Speichern von vielen Datensätzen aus der Datenbank festgestellt. Daraufhin wurden mittels eines Codereviews allfällige Schwachstellen eliminiert und die Zugriffe auf die Datenbank optimiert. Für die weiteren Sprints wurde jeweils am Ende ein Perfomancetest durchgeführt. Die Ergebnisse sind in Kapitel 7 dokumentiert.

9 Benutzerdokumentation

9.1 Installation

9.1.1 Systemanforderungen

9.1.1.1 Client

Die Systemanforderungen an den Client beinhalten lediglich einen der folgenden Browser in einer aktuellen Version:

- Firefox
- Google Chrome
- Safari

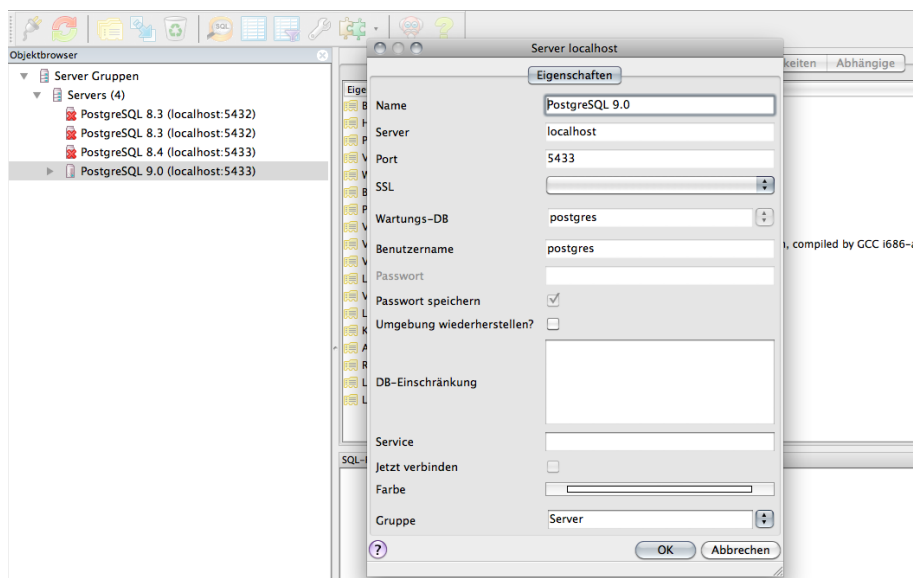
und einen Netzwerkzugang zum Server.

9.1.1.2 Server

Der Server ist unter allen Betriebssystemen lauffähig, welche mindestens das Java Runtime Environment 6 (JRE 6) installiert haben. Zusätzlich zu der JRE 6 werden mindestens 512 MB freier Speicher für den JBoss vorausgesetzt.

9.1.2 Installation der Datenbank

1. Download der PostgreSQL Datenbank von der URL
<http://www.postgresql.org/download/>
2. Installation der PostgreSQL Datenbank nach Herstellerangaben
 - a. Während der Installation wird man aufgefordert, einen Benutzernamen und ein Passwort einzugeben. Diese Informationen sollte notiert werden, da sie bei der Installation der E-OSCE Adminkomponente wieder verwendet werden müssen
3. Nach der erfolgreichen Installation ist der pgAdmin zu starten. Default mässig sind bereits einige Datebankserver installiert. Für die E-OSCE-Adminkomponente wird ein Server in der Version 9.x verwendet. Bei Bedarf müssen in den Einstellungen des Servers noch die Adresse (Server) und der Port geändert werden. Diese Daten



- sollten ebenfalls für die Installation der E-OSCE Adminkomponente notiert werden.
4. Auf dem Server muss jetzt noch eine neue Datenbank erstellt werden.

Der Name kann dabei frei gewählt werden. Der Name der Datenbank wird später bei der Installation der E-OSCE Adminkomponente ebenfalls wiederverwendet. Weitere Projektspezifische Einstellungen sind nicht notwendig. Das Datenbankschema wird vom Server automatisch beim ersten Start auf dem Server generiert.

5. Als letzter Schritt muss der Datenbankserver noch gestartet werden. Das Skript dazu befindet sich im Installationsordner der Datenbank.

9.1.3 Installation der E-OSCE-Adminkomponente

Die Adminkomponente wird als Zip-Datei geliefert [ANH00], welche entpackt werden muss. Die für die Installation relevanten Ordner sind in der Tabelle 17 – Ordnerstruktur beschrieben.

Ordner	Beschreibung
<i>bin</i>	Enthält das Startskript run.bat (Windows) beziehungsweise run.sh (Unix)
<i>certificates</i>	In diesem Ordner müssen vor dem Starten der Adminkomponente das E-OSCE Adminzertifikat sowie der Truststore hinterlegt werden. Diese werden für die Kommunikation mit dem Prüfungsserver verwendet.
<i>conf</i>	Im Ordner conf befindet sich die Datei eosceAdmin.properties. Vor dem Start der Applikation müssen in dieser Datei noch einige Konfigurationen vorgenommen werden. Die Beschreibungen der einzelnen Parameter im File sind in der Tabelle 18 - eosceAdmin.properties zu finden.

<i>server\default\log</i>	In diesem Ordner befinden sich die Log-Dateien des Servers
<i>restliche</i>	Alle anderen Ordner werden vom Server benötigt.

Tabelle 17 – Ordnerstruktur

Nachdem das E-OSCE Adminzertifikat und der Truststore in den entsprechenden Ordner kopiert worden sind, müssen die folgenden Konfigurationen am `eosceAdmin.properties`-File vorgenommen werden:

Konfiguration	Beschreibung
<i>eosce.db.servername</i>	Servename der PostgreSQL Datenbank. Default: localhost
<i>eosce.db.portnumber</i>	Portnummer der PostgreSQL Datenbank. Default: 5432
<i>eosce.db.name</i>	Datenbankname, welcher bei der Installation der Datenbank angegeben worden ist. Default: eosceAdmin
<i>eosce.db.username</i>	Benutzername des technischen Datenbank-Users. Default: postgres
<i>eosce.db.password</i>	Passwort des technischen Datenbank-Users. Default: admin
<i>eosce.assessment.server.ip</i>	IP-Adresse des Prüfungsservers Default: 127.0.0.1
<i>eosce.assessment.server.https.port</i>	HTTPS-Port des Prüfungsservers Default: 443
<i>jboss.web.http.port</i>	http-Port des E-OSCE Adminserver Default: 8080
<i>javax.net.ssl.keyStoreType</i>	Keystore Typ. Default: PKCS12
<i>javax.net.ssl.keyStore</i>	Pfad zum Keystore File. Dieser muss relativ zum bin Verzeichnis angegeben werden. Default: ../certificates/admin_1.p12
<i>javax.net.ssl.keyStorePassword</i>	Passwort des Keystore.
<i>javax.net.ssl.trustStoreType</i>	Truststore Typ. Default: JKS
<i>javax.net.ssl.trustStore</i>	Pfad zum Truststore File. Dieser muss relativ zum bin Verzeichnis angegeben werden. Default: ../certificates/truststore.jks
<i>javax.net.ssl.trustStorePassword</i>	Passwort des Truststore

Tabelle 18 - eosceAdmin.properties

9.2 Erste Schritte

9.2.1 Starten der E-OSCE-Adminkomponente

Ist die Installation und Konfiguration der Datenbank und des E-OSCE Adminserver abgeschlossen, kann der E-OSCE Adminserver über das Skript run.bat (Windows) oder run.sh (Unix) im Ordner bin gestartet werden.

Läuft der Admin-Server, kann in einem Browser die Applikation über die URL

<http://localhost:8080/eosceAdmin/login.jsf>

geöffnet werden.

9.2.2 Login

Beim erstmaligen Login wird man aufgefordert, einen neuen E-OSCE Admin-User zu definieren. Bei späteren Logins muss jeweils mit diesem User eingeloggt werden.

Weiter Admin-User können momentan nur über die Datenbank (Tabelle eosce_user) hinzugefügt werden.

9.2.3 Startseite

Die Startseite stellt die Übersicht über die bereits erstellten Prüfungen dar. Detailliertere Informationen zur Übersicht sind im Abschnitt „Planer“ beschrieben.

9.3 Editor



Abbildung 30 - Editor-Übersicht

9.3.1 Prüfungsbogen

9.3.1.1 Neuer Prüfungsbogen

Durch das Klicken auf den "Neuer Prüfungsbogen"-Button erscheint ein Dialog, in welchem man den Namen des Prüfungsbogens eingeben kann. Die weiteren Schritte um einen Prüfungsbogen zu erstellen werden im Abschnitt "Prüfungseditor" erläutert.



Abbildung 31 - Neuer Prüfungsbogen

9.3.1.2 Prüfungsbogen bearbeiten

Ein bereits erstellter Prüfungsbogen kann im Nachhinein bearbeitet werden, indem man den gewünschten Posten auswählt und dann auf den "Bearbeiten"-Button klickt. Die weiteren Schritte um einen Prüfungsbogen zu erstellen werden im Abschnitt "Prüfungsbogen erstellen" erläutert.

9.3.1.3 Neue Version

Falls man einen bereits erstellten Prüfungsbogen um einige Fragen erweitern will oder sonstige kleine Änderungen vornehmen will, kann man durch das Klicken auf den "Neue Version"-Button eine Kopie des ausgewählten Prüfungsbogens erstellen. Diese Kopie erhält automatisch die nächst höhere Versionsnummer.



Abbildung 32 - Neue Prüfungsbogenversion

Möchte man jedoch eine komplett neue Version des Prüfungsbogens erstellen, kann man über die Aktion „Neuer Prüfungsbogen“ einen neuen Bogen mit demselben Namen erstellen. Dabei wird die Version zum Beispiel von 1.0 auf die Version 2.0 erhöht.

9.3.1.4 Prüfungsbogen löschen

Um einen Prüfungsbogen zu löschen, wählt man diesen aus und drückt auf den "Löschen"-Button.

9.3.1.5 Prüfungsbogen suchen

Die Prüfungsektor-Ansicht bietet ein Suchfeld oberhalb der Prüfungsbogen-Tabelle an, in welchem nach bereits erstellten Prüfungsbogen gesucht werden kann.

9.3.2 Prüfungsektor

Wenn man einen neuen Prüfungsbogen erstellt hat oder einen Prüfungsbogen bearbeitet, gelangt man in den Prüfungsektor, in welchem man den Prüfungsbogen mit Fragen und den dazugehörigen Antworttypen füllen kann.



Abbildung 33 - Leerer Prüfungsbogen

9.3.2.1 Neuer Bereich

Ein Bereich gliedert die einzelnen Fragen in einen gemeinsamen Kontext. Bei einem neu erstellten Prüfungsbogen ist bereits ein leerer Bereich vorbereitet. Ein Bereich besteht aus einem Bereichstitel und optional aus einer Einführung (die Eingabe des Titels und der Einführung müssen durch Klicken auf den "Speichern"-Button gespeichert werden). Die Einführung soll dem Examinator bei der Prüfung einen kurzen Überblick über das folgende Themengebiet geben.

Um einen weiteren Bereich hinzuzufügen, klickt man auf das Feld "<<Klicken für neuen Bereich>>" (2)

9.3.2.2 Bereich bearbeiten

Indem man mit der Maus über den gewünschten Bereich fährt, erscheinen der "Bearbeiten"- und der "Löschen"-Button. Durch Klicken auf den "Bearbeiten"-Button kann man die Anga-

ben zu einem Bereich bearbeiten (Fragen lassen sich auf diese Weise nicht bearbeiten, dazu muss derjenige "Bearbeiten"-Button geklickt werden, welcher sich bei der Frage befindet).

9.3.2.3 Bereich löschen

Um einen Bereich zu löschen fährt man wie beim Bearbeiten mit der Maus über den gewünschten Bereich und klickt dann auf den "Löschen"-Button. Wenn ein Bereich gelöscht wird, werden auch alle dazugehörigen Fragen gelöscht.

9.3.2.4 Neue Frage

Um eine neue Frage zu erstellen, klickt man auf das Feld <<Klicken für neue Frage>> (1). Als Erstes gibt man die gewünschte Frage in das erste Feld ein. Danach kann man optional noch eine Anweisung für den Examinator eingeben.

Zu jeder Frage gibt es mehrere Antwortmöglichkeiten, welche man als Antworttyp speichern kann. Beim ersten Starten der E-OSCE-Adminkomponente ist die Liste der Antworttypen noch leer. Um einen neuen Antworttyp zu erstellen, klickt man auf den "+"-Button neben dem Dropdown-Menü.

Abbildung 34 - Neuer Antworttyp

Es erscheint ein leeres Feld, in welches man die erste Antwortoption eingeben kann. Für jede weitere Antwortmöglichkeit klickt man auf das Feld "Klicken für neue Option". Falls die Frage Kriterien enthalten soll, kann man diese durch Klicken auf das Feld "Klicken für neues Kriterium" hinzufügen.

Abbildung 35 - Neues Kriterium

Wenn alle Eingaben getätigt worden sind, kann die Frage durch Klicken auf den "Speichern"-Button gespeichert werden. Dieser Button erscheint, wenn man sich mit der Maus über dem Frage-Bereich befindet. Durch das Speichern der Frage wird auch der Antworttyp gespeichert und kann nun zur Verwendung aus der Liste ausgewählt werden und muss nicht mehr jedes Mal manuell erstellt werden.

9.3.2.5 Frage bearbeiten

Durch das Klicken auf den "Bearbeiten"-Button bei der gewünschten Frage, kann man die gemachten Angaben bearbeiten.

9.3.2.6 Frage löschen

Um eine Frage zu löschen, fährt man mit der Maus über die gewünschte Frage und es erscheint ein "Löschen"-Button. Durch Klicken auf diesen Button wird die Frage gelöscht, der dazugehörige Antworttyp bleibt jedoch bestehen.

9.3.2.7 Drag & Drop

9.3.2.7.1 Bereiche umsortieren

Wenn man mehrere Bereiche erstellt hat, kann man diese per Drag&Drop neu sortieren. Die Fragen, welche zu einem Bereich gehören, werden dabei zusammen mit dem Bereich ver-

schoben.

Bsp. Kopfweh-Bereich mit Hals-Nase Ohren-Bereich vertauschen

Prüfungsbogen HNO v.2.0

Kopfweh
Bereichstitel: Kopfweh
Einführung:

Frage 1: Richtig diagnostiziert?
Option: Ja / Nein
<< Klicken für neue Frage >>

Hals Nase Ohren
Bereichstitel: Hals Nase Ohren
Einführung:

Frage 1: Hat sich der Kandidat vorgestellt?
Option: Ja / Nein
<< Klicken für neue Frage >>

<< Klicken für neuen Bereich >>

Fertig

Abbildung 36 - Drag&Drop Bereich 01

Prüfungsbogen HNO v.2.0

Kopfweh
Bereichstitel: Kopfweh
Einführung:

Frage 1: Richtig diagnostiziert?
Option: Ja / Nein
<< Klicken für neue Frage >>

Hals Nase Ohren
Bereichstitel: Hals Nase Ohren
Einführung:

Frage 1: Hat sich der Kandidat vorgestellt?
Option: Ja / Nein
<< Klicken für neue Frage >>

Fertig

Abbildung 37 - Drag&Drop Bereich 02

Prüfungsbogen HNO v.2.0

Hals Nase Ohren
Bereichstitel: Hals Nase Ohren
Einführung:

Frage 1: Hat sich der Kandidat vorgestellt?
Option: Ja / Nein
<< Klicken für neue Frage >>

Kopfweh
Bereichstitel: Kopfweh
Einführung:

Frage 1: Richtig diagnostiziert?
Option: Ja / Nein
<< Klicken für neue Frage >>

<< Klicken für neuen Bereich >>

Fertig

Abbildung 38 - Drag&Drop Bereich 03

9.3.2.7.2 Fragen umsordieren

Per Drag&Drop kann eine Frage sowohl innerhalb des Bereiches verschoben werden, als auch einem anderen Bereich zugewiesen werden.

9.4 Basisdaten

Die folgenden Abschnitte beschreiben das Hinzufügen, Bearbeiten und Löschen der prüfungsrelevanten Daten (Posten, Kandidaten, Examinatoren und Geräte). Die folgende Abbildung zeigt, wie man auf die verschiedenen Daten zugreifen kann.

9.4.1 Posten



Abkürzung	Beschreibung	Prüfungsbogen
HNO1	HNO	HNO 1.0
HNO2	HNO	HNO 1.1
HNO3	HNO	HNO 2.0
KPW1	Kopfweh	KPW 1.0
KPW2	Kopfweh	KPW 2.0

Abbildung 39 - Posten-Übersicht

9.4.1.1 Posten hinzufügen

Durch das Klicken auf den "Neuer Posten"-Button erscheint ein Dialog, in welchem man die Abkürzung und die Beschreibung des Postens eingeben kann. Zudem kann man per "autocomplete" einen bereits erstellten Prüfungsbogen (siehe Editor) dem Posten hinzufügen.



Neuen Posten ✕
Abkürzung:
Beschreibung:
Prüfungsbogen:

Abbildung 40 - Neuer Posten

9.4.1.2 Posten bearbeiten

Ein bereits erstellter Posten kann im Nachhinein bearbeitet werden, indem man den gewünschten Posten auswählt und dann auf den "Bearbeiten"-Button klickt. Es erscheint ein Dialog mit den gleichen Angaben wie bei "Posten hinzufügen", wo die gewünschten Änderungen vorgenommen werden können.

9.4.1.3 Posten löschen

Um einen Posten zu löschen, wählt man diesen aus und drückt auf den "Löschen"-Button. Dabei wird lediglich der Posten gelöscht und nicht der Prüfungsbogen, welcher dem Posten hinzugefügt worden ist.

9.4.1.4 Posten suchen

Die Posten-Ansicht bietet ein Suchfeld oberhalb der Posten-Tabelle an, in welchem nach bereits erstellten Posten gesucht werden kann.

9.4.2 Examinatoren

Examinatoren



Suche:

Name	Vorname	Anrede
Cox	Perry	Dr.
Feuerstein	Fred	Dr.
Kelso	Bob	Dr.
Man	Bat	Dr.
Parker	Peter	Prof.
Sommer	Alfred	Dr.
Stinson	Barney	Dr.

[Löschen](#) [Bearbeiten](#) [Neuer Examinator](#)

Abbildung 41 - Examinatoren-Übersicht

9.4.2.1 Examinator hinzufügen

Durch das Klicken auf den "Neuer Examinator"-Button erscheint ein Dialog, in welchem man den Namen und Vornamen, sowie die Anrede des Examinators eingeben kann.

Neuer Examinator ✕

Name:

Vorname:

Anrede:

[Abbrechen](#) [Absenden](#)

Abbildung 42 - Neuer Examinator

9.4.2.2 Examinatorenliste importieren

Neben dem Hinzufügen einzelner Examinatoren, besteht auch die Möglichkeit, eine ganze Liste von Examinatoren zu importieren.

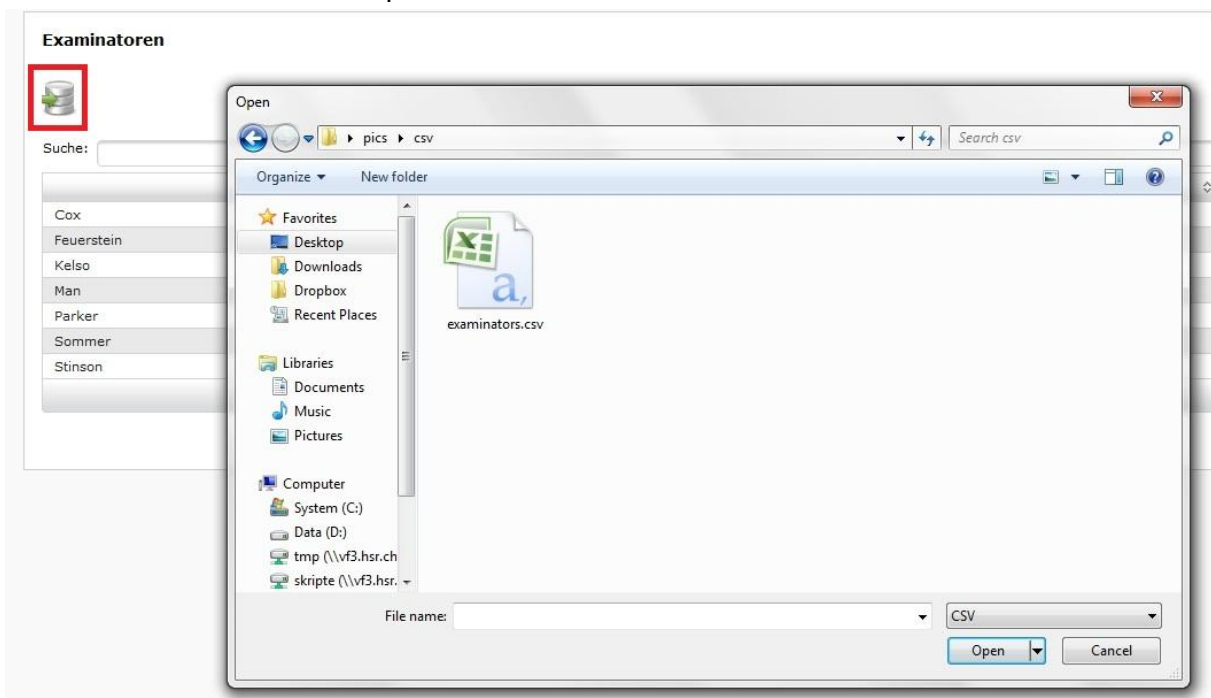


Abbildung 43 - Examinatorenliste importieren

Das CSV-File für die Examinatorenliste muss dabei folgende Struktur aufweisen:

	A	B	C
1	Dr.	Feuerstein	Fred
2	Dr.	Stinson	Barney
3	Dr.	Kelso	Bob
4	Dr.	Cox	Perry
5	Dr.	Sommer	Alfred

Anrede
Nachname
Vorname

Abbildung 44 - Struktur Examinatorenliste

9.4.2.3 Examinator bearbeiten

Ein bereits erfasster Examinator kann im Nachhinein bearbeitet werden, indem man den gewünschten Examinator auswählt und dann auf den "Bearbeiten"-Button klickt. Es erscheint ein Dialog mit den gleichen Angaben wie bei "Examinator hinzufügen", wo die gewünschten Änderungen vorgenommen werden können.

9.4.2.4 Examinator löschen

Um einen Examinatoren zu löschen, wählt man diesen aus und drückt auf den "Löschen"-Button.

Achtung: Falls der Examinator noch einer Prüfung zugeteilt ist, kann dieser nicht gelöscht werden.

9.4.2.5 Examinator suchen

Die Examinator-Ansicht bietet ein Suchfeld oberhalb der Examinator-Tabelle an, in welchem nach bereits erfassten Examinatoren gesucht werden kann.

9.4.3 Kandidaten

Kandidaten



Suche:

Name	Vorname	Studentennummer
Gruber	Patrick	131
Müller	Fritz	132
Muster	Peter	124
Pfister	Sascha	112

Abbildung 45 - Kandidaten-Übersicht

9.4.3.1 Kandidat hinzufügen

Durch das Klicken auf den "Neuer Kandidat"-Button erscheint ein Dialog, in welchem man den Namen und Vornamen, sowie die Studentennummer des Kandidaten eingeben kann.

Neuer Kandidat

Name:

Vorname:

Studentennummer:

Abbrechen

Absenden

Abbildung 46 - Neuer Kandidat

9.4.3.2 Kandidatenliste importieren

Neben dem Hinzufügen einzelner Kandidaten, besteht auch die Möglichkeit, eine ganze Liste von Kandidaten zu importieren.

Kandidaten

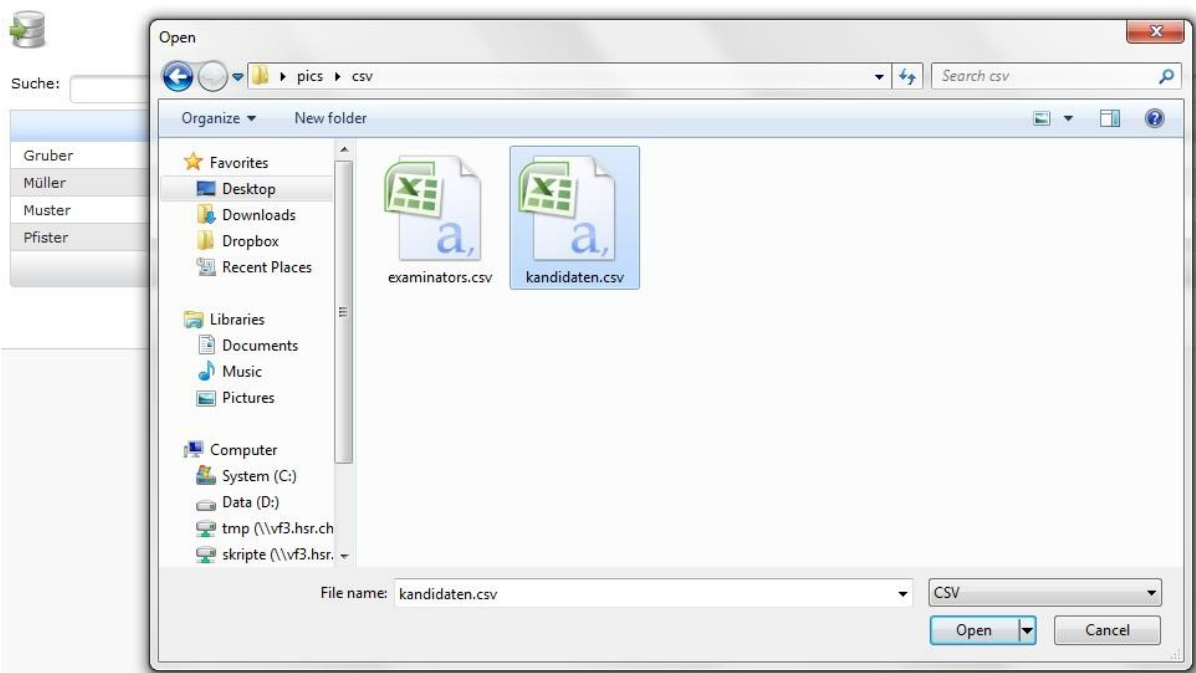


Abbildung 47 - Kandidatenliste importieren

Das CSV -File für die Kandidatenliste muss dabei folgende Struktur aufweisen:

	A	B	C	
1	16040	Abplanalp	Dany	
2	16240	Ajradini	Almir	
3	16359	ALJILJI	Habib	
4	16132	Artho	Pascal Roman	
5	12536	Aschmann	Gustav	
6	15797	Baier	Donat	
7	15946	Bammatter	Adrian	
8	16172	Bapst	Daniel	

Studentennr.

Nachname

Vorname

Abbildung 48 - Struktur Kandidatenliste

9.4.3.3 Kandidat bearbeiten

Ein bereits erfasster Kandidat kann im Nachhinein bearbeitet werden, indem man den gewünschten Kandidaten auswählt und dann auf den "Bearbeiten"-Button klickt. Es erscheint ein Dialog mit den gleichen Angaben wie bei "Kandidat hinzufügen", wo die gewünschten Änderungen vorgenommen werden können.

9.4.3.4 Kandidat löschen

Um einen Kandidaten zu löschen, wählt man diesen aus und drückt auf den "Löschen"-Button.

Achtung: Falls der Kandidat noch einer Prüfung zugeteilt ist, kann er nicht gelöscht werden.

9.4.3.5 Kandidat suchen

Die Kandidaten-Ansicht bietet ein Suchfeld oberhalb der Kandidaten-Tabelle an, in welchem nach bereits erfassten Kandidaten gesucht werden kann. Dabei kann man nicht nur nach dem Namen des Kandidaten suchen, sondern auch nach dessen Studentenummer.

9.4.4 Infrastruktur

Infrastruktur

Suche:

Geräte-ID	Keycode	Person
IPad01	1234	Müller Peter
IPad02	1432	Muster Max

[Löschen](#) [Bearbeiten](#) [Neues Gerät](#)

Abbildung 49 - Infrastruktur-Übersicht

9.4.4.1 Gerät hinzufügen

Durch das Klicken auf den "Neues Gerät"-Button erscheint ein Dialog, in welchem man die ID und den Keycode des Gerätes eingeben kann. Optional ist das Feld "Person", in welches man den Namen des Examinators eingeben kann, welcher dieses Gerät erhalten soll.

Neues Gerät ✕

Geräte-ID:

Keycode:

Person:

[Abbrechen](#) [Absenden](#)

Abbildung 50 - Neues Gerät

9.4.4.2 Gerät bearbeiten

Ein bereits erfasstes Gerät kann im Nachhinein bearbeitet werden, indem man das gewünschte Gerät auswählt und dann auf den "Bearbeiten"-Button klickt. Es erscheint ein Dialog mit den gleichen Angaben wie bei "Gerät hinzufügen", wo die gewünschten Änderungen vorgenommen werden können.

9.4.4.3 Gerät löschen

Um ein Gerät zu löschen, wählt man dieses aus und drückt auf den "Löschen"-Button.

9.4.4.4 Gerät suchen

Die Infrastruktur-Ansicht bietet ein Suchfeld oberhalb der Geräte-Tabelle an, in welchem nach bereits erfassten Geräten gesucht werden kann.

9.5 Planer

Prüfungsplaner

Suche:

Prüfungen		
Frühling 2013	Rotationen: 1 Parcours: 1 Posten: 5 Kandidaten: 4	Die Prüfung ist unvollständig • Es ist nicht jedem Posten ein Examinator zugewiesen Zum Fehler
Herbst 2012	Rotationen: 0 Parcours: 0 Posten: 0 Kandidaten: 0	Hinweis: Es sind keine Posten zugeteilt Weiter
Sommer 2011	Rotationen: 2 Parcours: 2 Posten: 2 Kandidaten: 2	Die Prüfung kann exportiert werden exportieren

[Löschen](#) [Bearbeiten](#) [Neue Prüfung](#)

Abbildung 51 - Planer-Übersicht

9.5.1 Prüfungsplaner

9.5.1.1 Neue Prüfung

Durch das Klicken auf den "Neue Prüfung"-Button wird der Prüfungsplaner-Wizard gestartet, welcher den Benutzer durch die Prüfungsplanung führt. Dieser Wizard wird in den folgenden Abschnitten genauer erläutert.

9.5.1.2 Prüfung bearbeiten

Eine bereits erfasste Prüfung kann im Nachhinein bearbeitet werden, indem man die gewünschte Prüfung auswählt und dann auf den "Bearbeiten"-Button klickt. Der Benutzer gelangt dann an den Anfang des Wizards, wo die eingegebenen Daten bearbeitet werden können.

9.5.1.3 Prüfung löschen

Um eine Prüfung zu löschen, wählt man diese aus und drückt auf den "Löschen"-Button. Das Löschen der Prüfung führt dazu, dass die Beziehungen der Posten, Examinatoren und Kandidaten zur Prüfung ebenfalls gelöscht werden, nicht aber die Posten, Examinatoren und Kandidaten selbst.

9.5.1.4 Prüfung suchen

Die Prüfungsplaner-Ansicht bietet ein Suchfeld oberhalb der Prüfungen-Tabelle an, in welchem nach bereits erstellten Prüfungen gesucht werden kann.

9.5.2 Prüfungswizard



Abbildung 52 - Wizard

Der Prüfungswizard führt den Benutzer durch die komplette Planung einer Prüfung. Dabei hat der Benutzer zwei Möglichkeiten, sich durch diesen Wizard zu klicken:

1. durch das jeweilige Klicken auf den "Weiter"-Button am unteren rechten Rand
2. durch das Klicken auf den nächsten Schritt im Wizard selbst (sofern dieser nicht deaktiviert ist)

Um zum vorherigen Schritt zurückzugelangen, klickt man entweder auf den "Zurück"-Button am unteren linken Rand oder direkt auf den gewünschten Schritt direkt im Wizard.

Will man den Wizard vorzeitig verlassen, befindet sich auf der rechten Seite der Button „Fertig“ welcher den Wizard abschliesst, den momentanen Prüfungsstatus validiert und die Übersicht der Prüfungen anzeigt.

In den folgenden Abschnitten werden die einzelnen Schritte genauer erläutert.

9.5.2.1 Informationen

Geben Sie einen Prüfungsnamen an und legen Sie das Startdatum der Prüfung fest.

- Bewertungsdauer = Bewertungszeit eines Kandidaten
- Pausendauer = Wechselzeit zwischen den Posten

Prüfungsname:

Start:

Bewertungsdauer (in s):

Pausendauer (in s):

Abbildung 53 - Informationen

Im ersten Schritt wird der Benutzer dazu aufgefordert, den Prüfungsnamen sowie den Start der Prüfung, d.h. der Prüfungstag und die Zeit einzugeben. Die eingegebene Zeit entspricht dem Starttermin der ersten Rotation der Prüfung.
Ebenfalls sollen hier die Posten- und Pausendauer (in Sekunden) angegeben werden.

9.5.2.2 Posten

Abbildung 54 - Posten

Auf der linken Seite befindet sich eine Liste mit allen verfügbaren Posten, d.h. alle Posten, welche bisher erfasst wurden, sind in dieser Liste vertreten. Der Benutzer kann einen solchen Posten mittels Drag&Drop, Doppelklick oder durch betätigen des "Hinzufügen"-Buttons zu der Prüfung hinzufügen (in die Liste der hinzugefügten Posten). Es können auch mehrere Posten angeklickt werden und zur Prüfung hinzugefügt werden. Per "Alle Hinzufügen"-Button werden alle verfügbaren Posten in die Liste der hinzugefügten Posten verschoben. Das Entfernen eines oder mehrerer Posten funktioniert auf die gleiche Weise in die gegengesetzte Richtung.

Im Suchfeld kann nach einem verfügbaren Posten gesucht werden.

Zusätzlich ist es hier noch einmal möglich, einen neuen Posten zu den verfügbaren Posten hinzuzufügen. Durch Klicken auf den "Neuer Posten"-Button erscheint derselbe Dialog, wie beim Erfassen eines neuen Postens bei den Basisdaten.

[Neuer Posten]

Hinweis: Es muss mind. ein Posten zur Prüfung hinzugefügt werden, damit man zum nächsten Schritt gelangen kann.

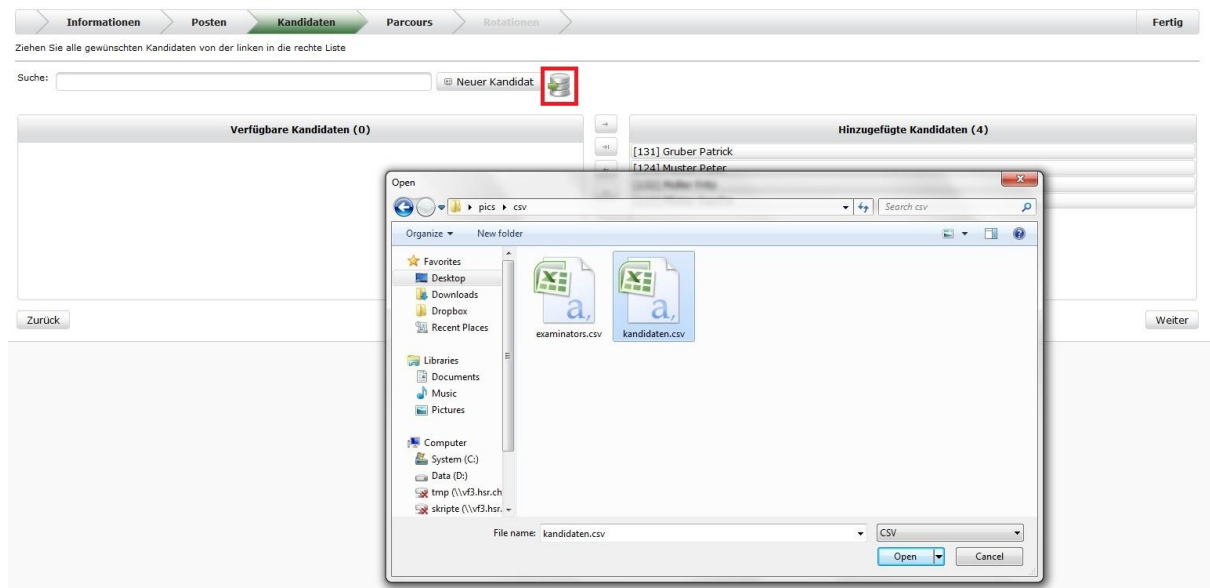
9.5.2.3 Kandidaten

Abbildung 55 - Kandidaten

Die Kandidaten-Seite ist genau gleich aufgebaut wie im Schritt davor die Posten-Seite. Auf der linken Seite befinden sich die verfügbaren Kandidaten, welche sich wiederum per Drag&Drop, Doppelklick oder Betätigen des "Hinzufügen"-Buttons zur Prüfung hinzufügen lassen. Ebenso kann man mehrere oder alle Kandidaten in die Liste der hinzugefügten Kandidaten verschieben.

Auch das Suchen eines verfügbaren Kandidaten und das Hinzufügen eines neuen Kandidaten sind hier möglich.

Ein Unterschied zum vorherigen Schritt stellt das "Import"-Icon dar. Durch Klicken auf dieses Icon kann man eine vorgefertigte Liste (siehe auch Kapitel 9.4.3.2) von Kandidaten direkt einer Prüfung zuweisen. Falls diese Liste Kandidaten enthält, welche sich bereits in der "verfügbare Kandidaten"-Liste befinden, werden diese automatisch aus dieser Liste gelöscht und in die Liste der hinzugefügten Kandidaten verschoben.



Hinweis: Es muss mind. ein Kandidat zur Prüfung hinzugefügt werden, damit man zum nächsten Schritt gelangen kann.

9.5.2.4 Parcours



Abbildung 56 - Parcours

Die Parcours-Seite gibt dem Benutzer eine Übersicht über alle Parcours und die dazugehörigen Posten. Die minimale Anzahl der Parcours wird anhand der hinzugefügten Posten und Kandidaten berechnet und die Farbe nach dem Zufallsprinzip hinzugefügt. Jeder Parcours wird jeweils mit allen hinzugefügten Posten gefüllt. Die Anzeige der Parcours ist hier statisch, d.h. es können weder die Parcours selbst noch die zugehörigen Posten in der Reihenfolge verändert werden (dies ist erst beim nächsten Schritt "Rotationen" möglich).

9.5.2.4.1 Parcours hinzufügen (1)

Um einen weiteren Parcours hinzuzufügen, klickt man auf den "Parcours hinzufügen"-Button und es wird ein neuer Parcours erstellt, welcher mit den gleichen Posten gefüllt wird, wie sie bei allen anderen Parcours vorkommen.

9.5.2.4.2 Parcoursfarbe ändern (2)

Die Farbe eines Parcours lässt sich durch einen Klick auf das Farbkästchen ändern. Dadurch öffnet sich eine Farbpalette, woraus man die gewünschte Farbe auswählen kann.

9.5.2.4.3 Parcours löschen (3)

Um einen Parcours zu löschen, klickt man auf das "Löschen"-Symbol neben dem gewünschten Parcours. Dabei wird lediglich der Parcours selbst, nicht aber die Posten oder Kandidaten gelöscht. Nach dem Löschen werden die übrigen Parcours neu nummeriert.

Hinweis: Es können nicht weniger als die minimal berechnete Anzahl an Parcours existieren, so dass der "Löschen"-Button erst auftaucht, wenn es mehr Parcours gibt als das Minimum.

9.5.2.4.4 Pausenposten (4)

Durch das Klicken auf den "Pausenposten hinzufügen"-Button wird jedem Parcours ein Pausenposten hinzugefügt. Es können beliebig viele Pausenposten zu einem Parcours hinzugefügt werden.

Um einen Pausenposten zu löschen, klickt man auf das "Löschen"-Symbol neben einem der Pausenposten. Dabei spielt es keine Rolle, von welchem Parcours der Pausenposten gelöscht wird, da derjenige Pausenposten aus allen vorhandenen Parcours entfernt wird.

9.5.2.4.5 Non-E-OSCE-Posten (5)

Nebst den E-OSCE-Posten (hier jeweils nur Posten genannt) gibt es auch noch spezielle Non-E-OSCE-Posten. Diese können hier manuell genau gleich wie Pausenposten zu den Parcours hinzugefügt werden. Durch Klicken auf den "Non-E-OSCE-Posten hinzufügen"-Button öffnet sich ein Dialog, in welchen man den Namen dieses Postens eingeben kann. Es können beliebig viele Non-E-OSCE-Posten zu einem Parcours hinzugefügt werden.

Um einen Non-E-OSCE-Posten zu löschen, geht man genau gleich vor wie beim Löschen eines Pausenpostens. Durch Klicken auf eines der "Löschen"-Symbole wird derjenige Non-E-OSCE-Posten aus allen Parcours entfernt.

9.5.2.4.6 Anzahl Räume (6)

Das optionale Feld "Anzahl Räume" erlaubt die Eingabe der vorhandene Räume für die Prüfung. Anhand dieser Zahl wird die minimale Anzahl an Rotationen berechnet. Falls das Feld leer gelassen wird, wird ein Plan mit nur einer Rotation generiert.

Die Anzahl Räume kann im Nachhinein noch geändert werden, damit sich die Anzahl an Rotationen jedoch verändert, muss der Plan von neuem generiert werden (siehe nächster Abschnitt).

9.5.2.4.7 Plan generieren (7)

Beim Klicken auf den "Plan generieren"-Button wird ein komplett neuer Plan zusammengestellt. Im Detail heisst das, dass zuerst die Anzahl an Rotationen erstellt wird, welche auf Grund der Anzahl Räume errechnet wurde. Danach werden alle Parcours gleichmässig auf diese Rotationen aufgeteilt. Zusätzlich werden alle Kandidaten, welche der Prüfung hinzugefügt worden sind, nach dem Zufallsprinzip einem Posten als Startkandidat eingetragen.

Achtung:

Der Button "Plan generieren" sollte nur beim ersten Durchlauf einer Prüfungsplanung gedrückt werden. Bei nachträglichen Änderungen (z.B. Hinzufügen eines Postens oder Kandidaten) an einer Prüfung besteht die Möglichkeit direkt über den Wizard zum Rotations-Schritt zu gelangen.

Die folgenden beiden Listen zeigen die Unterschiede zwischen den oben genannten Varianten auf:

Variante 1: per Wizard zu den Rotationen:

- vorherige Änderungen gehen nicht verloren, d.h. die Anzahl Rotationen bleibt gleich, genau wie die Aufteilung der Parcours

- Die Reihenfolge der Posten innerhalb eines Parcours bleibt erhalten
- Die Posten behalten dieselben Startkandidaten, Examinatoren und Prüfungsbogen
- Neu hinzugefügte Posten werden jeweils am Ende jedes Parcours angefügt und besitzen weder einen Startkandidaten noch einen Examinatoren (lediglich der Prüfungsbogen ist dem Posten zugewiesen)
- Gelöschte Posten werden aus allen Parcours gelöscht (dies kann dazu führen, dass nicht mehr genügend Posten für alle Kandidaten existieren)
- Neue hinzugefügte Kandidaten stehen in der Rotations-Ansicht zur Verfügung, werden nicht explizit einem Posten ohne Startkandidat hinzugefügt.
- Gelöschte Kandidaten werden von der Prüfung entfernt (dies kann zu einer grossen Anzahl an Posten mit Pausenkandidaten führen).

Variante 2: per „Plan generieren“ zu den Rotationen:

- Alle vorherigen Änderungen gehen verloren, alle Rotationen inkl. Parcours (und Posten) werden dabei gelöscht
- Rotationen werden neu erstellt und die Parcours gleichmässig diesen Rotationen zugeteilt.
- Die Posten der Parcours sind wieder in der ursprünglichen Reihenfolge und besitzen wieder einen neuen Zufalls-Startkandidaten, aber keinen Examinatoren.

9.5.2.5 Rotationen

Abbildung 57 - Rotationen

Die Rotationssicht bietet dem Benutzer eine Übersicht über die komplette Prüfung. Dabei ist die Übersicht in drei Tabs unterteilt:

- Examinatoren-Ansicht: Zeigt eine Übersicht mit den zu den Posten zugeteilten Examinatoren
- Kandidaten-Ansicht: Zeigt eine Übersicht mit den Startkandidaten
- Formular-Ansicht: Zeigt eine Übersicht mit den ausgewählten Prüfungsbogen

9.5.2.5.1 Rotation hinzufügen (1)

Um eine neue Rotation hinzuzufügen, klickt man auf den „Rotation hinzufügen“-Button. Es wird eine leere Rotation erstellt, welche die gleiche Startzeit aufweist, wie bei der Informations-Ansicht (erster Schritt im Wizard) eingegeben wurde.

9.5.2.5.2 Rotation löschen (2)

Sobald eine Rotation keine Parcours mehr enthält, kann diese gelöscht werden (durch Klicken auf das „Löschen“-Symbol).

9.5.2.5.3 Rotationsstartzeit festlegen (3)

Die Rotationsstartzeit kann jederzeit geändert werden. Dazu wählt man aus dem Kalender das gewünschte Datum aus und fügt dahinter die Startzeit ein.

Hinweis: Die Zeitangabe hinter der Startzeit ist die automatisch berechnete Endzeit der Rotation

9.5.2.5.4 Grundlegende Drag&Drop-Funktionen

In der Rotations-Übersicht lässt sich fast jedes Element per Drag&Drop verschieben, dabei gibt es Tab-spezifische Funktionen, welche in den nächsten Abschnitten genauer erläutert werden, und grundlegende Drag&Drop-Funktionen, welche für alle Tabs gültig sind:

- Parcours Drag&Drop:

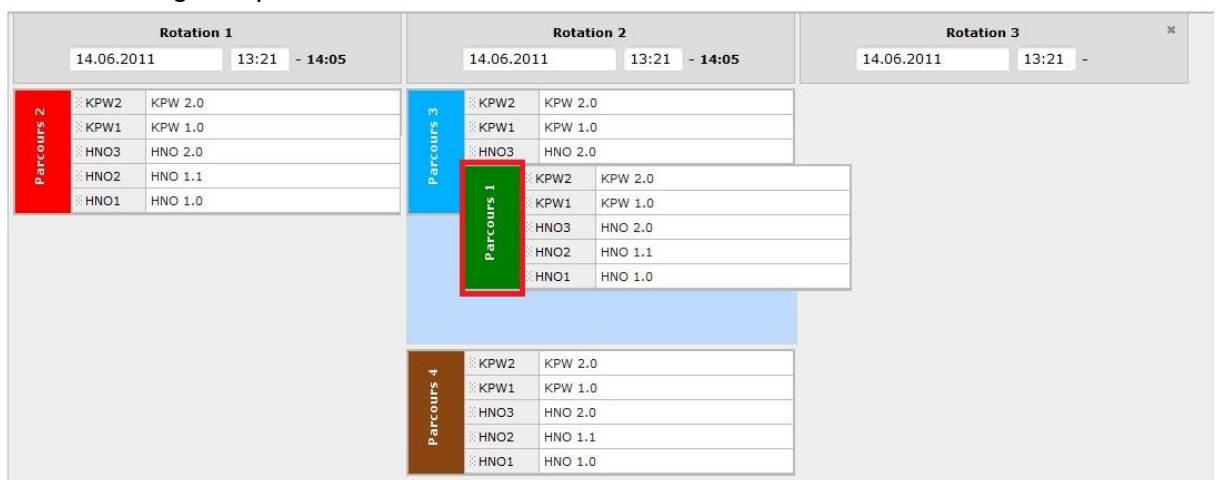


Abbildung 58 - Drag&Drop Parcours

Die Parcours können durch Auswählen (und festhalten der Maustaste, „drag“ genannt) der Farbfläche (markierter roter Bereich) über die ganze Tabelle gezogen werden. Sobald der Parcours über die gewünschte Stelle gezogen wurde (diese wird durch ein blaues Kästchen angezeigt), kann der Parcours losgelassen werden. Ein Parcours kann sowohl innerhalb einer Rotation verschoben werden, als auch einer anderen Rotation zugewiesen werden. Dabei werden die Posten, welcher zu diesem Parcours gehören jeweils mitverschoben.

- Posten Drag&Drop:



Abbildung 59 - Drag&Drop Posten

- Ein einzelner Posten kann innerhalb des Parcours verschoben werden, sodass eine neue Reihenfolge entsteht. Wird ein Posten ausgewählt kann dieser an die Position eines anderen Postens desselben Parcours gezogen werden. Durch Drag&Drop wird hier ein Austausch der beiden beteiligten Posten vorgenommen. Dabei werden die Daten, welche zum Posten gehören (Startkandidat, Examiner und Prüfungsbogen) ebenfalls mit dem Posten vertauscht.

9.5.2.5.5 Examinatoren-Ansicht

The screenshot shows the 'Examinatoren' tab in the software interface. It displays three rotations for the date 14.06.2011, each from 13:21 to 14:05. Each rotation contains four parcours (1, 2, 3, 4). Each parcours has five exam positions (KPW2, KPW1, HNO3, HNO2, HNO1). All positions currently show 'Kein Examiner ausgewählt' (No examiner selected).

Abbildung 60 - Examinatoren-Ansicht

Zu Beginn (nachdem der Plan generiert wurde) sind alle Einträge hier noch leer, resp. ist kein Examiner ausgewählt. Dieser kann nun für jeden Posten manuell ausgewählt werden. Break- und Non-E-OSCE-Posten benötigen keinen Examinatoren. Durch Klicken auf das leere Feld öffnet sich ein Dropdown-Menü, welches alle erfassten Examinatoren beinhaltet. Daraus kann der gewünschte Examiner ausgewählt werden.

This screenshot shows the same interface as Abbildung 60, but with a dropdown menu open for the HNO3 position in Parcours 1 of Rotation 1. The dropdown lists several examiners, including Cox Perry, Dr., Feuerstein Fred, Prof., Kelso Bob, Dr., and others. The 'Auswählen...' option is highlighted at the top of the list.

Abbildung 61 - Dropdown Examinatoren

Hinweis: Sobald ein Examiner einem Posten zugewiesen worden ist, wird er nicht mehr in den Dropdown-Listen der Posten, welche zur gleichen Rotation gehören, angezeigt.

Die Examinatoren-Ansicht bietet noch spezielle Drag&Drop-Funktionen an. Zum einen können durch Ziehen des gewünschten Examinators an die Position eines anderen Examinators desselben Parcours die beiden Examinatoren miteinander vertauscht werden. Dabei werden die Posten nicht mit vertauscht, d.h. die beiden Examinatoren werden einem anderen Posten zugewiesen.

Hinweis: Der Examiner kann nicht einem Posten eines anderen Parcours in der gleichen Rotation zugewiesen werden.

Rotation 1		
14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Man Bat Dr.
	KPW1	Kelso Bob Dr.
	HNO3	Cox Perry Dr.
	HNO2	Sommer Alfred Dr.
	HNO1	Stinson Barney Dr.

Abbildung 62 - Examinatoren_Exchange_2

Parcours 1	KPW2	Man Bat Dr.
	KPW1	Cox Perry Dr.
	HNO3	Kelso Bob Dr.
	HNO2	Sommer Alfred Dr.
	HNO1	Stinson Barney Dr.

Abbildung 63 - Examinatoren_Exchange_1

Wird ein Examiner aber an die Position eines anderen Examinators ausserhalb der Rotation gezogen, gibt es zwei verschiedene Varianten:

1. Der Examiner wird an eine Position gezogen, wo bereits ein anderer Examiner ausgewählt ist:

Rotation 1			Rotation 2		
14.06.2011 12:00 - 12:44			14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Man Bat Dr.	Parcours 3	KPW2	daLenz Sven Chief
	KPW1	Kelso Bob Dr.		KPW1	Stinson Barney Dr.
	HNO3	Cox Perry Dr.		HNO3	Cox Perry Dr.
	HNO2	Sommer Alfred Dr.		HNO2	Kelso Bob Dr.
	HNO1	Stinson Barney Dr.		HNO1	Sommer Alfred Dr.

Abbildung 64 - Variante01_1

Parcours 1	KPW2	Man Bat Dr.	Parcours 3	KPW2	daLenz Sven Chief
	KPW1	Kelso Bob Dr.		KPW1	Stinson Barney Dr.
	HNO3	Cox Perry Dr.		HNO3	Cox Perry Dr.
	HNO2	Kelso Bob Dr.		HNO2	Sommer Alfred Dr.
	HNO1	Stinson Barney Dr.		HNO1	Sommer Alfred Dr.

Abbildung 65 - Variante 01_2

Die beiden Examinatoren werden wie zuvor ausgetauscht, die Posten bleiben an ihrem ursprünglichen Ort.

Achtung: Examinatoren können auf diese Weise mehrfach in einer Rotation vorkommen, was zu einem Fehler führt.

2. Der Examiner wird an eine Position gezogen, wo noch kein Examiner ausgewählt ist:

Rotation 1			Rotation 2		
14.06.2011 12:00 - 12:44			14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Man Bat Dr.	Parcours 3	KPW2	Kein Examiner ausgewählt
	KPW1	Kelso Bob Dr.		KPW1	Kein Examiner ausgewählt
	HNO3	Cox Perry Dr.		HNO3	Kein Examiner ausgewählt
	HNO2	Sommer Alfred Dr.		HNO2	Kein Examiner ausgewählt
	HNO1	Stinson Barney Dr.		HNO1	Sommer Alfred Dr.

Abbildung 66 - Variante 02_1

Parcours 1	KPW2	Man Bat Dr.
	KPW1	Kelso Bob Dr.
	HNO3	Cox Perry Dr.
	HNO2	Sommer Alfred Dr.
	HNO1	Stinson Barney Dr.
Parcours 3	KPW2	Kein Examiner ausgewählt
	KPW1	Kein Examiner ausgewählt
	HNO3	Cox Perry Dr.
	HNO2	Kein Examiner ausgewählt
	HNO1	Sommer Alfred Dr.

Abbildung 67 - Variante02_2

Der Examiner wird an die leere Position kopiert, sodass beide Posten den gleichen Examiner enthalten.

9.5.2.5.6 Kandidaten-Ansicht

Rotation 1 14.06.2011 12:00 - 12:44			Rotation 2 14.06.2011 12:00 - 12:44			Rotation 3 14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Gruber Patrick	Parcours 3	KPW2	BREAK Kandidat	Parcours 4	KPW2	BREAK Kandidat
	KPW1	Müller Fritz		KPW1	BREAK Kandidat		KPW1	BREAK Kandidat
	HNO3	Muster Peter		HNO3	BREAK Kandidat		HNO3	BREAK Kandidat
	HNO2	Pfister Sascha		HNO2	BREAK Kandidat		HNO2	BREAK Kandidat
	HNO1	BREAK Kandidat		HNO1	BREAK Kandidat		HNO1	BREAK Kandidat
Parcours 2	KPW2	BREAK Kandidat						
	KPW1	BREAK Kandidat						
	HNO3	BREAK Kandidat						
	HNO2	BREAK Kandidat						
	HNO1	BREAK Kandidat						

Abbildung 68 - Kandidaten-Ansicht

Die Posten erhalten nach der Plangenerierung automatisch einen zufälligen Startkandidaten.

Hinweis: Falls es mehr Posten als Kandidaten gibt, werden die restlichen Posten mit Pausenkandidaten („BREAK Kandidat“) gefüllt.

Durch Klicken auf einen Kandidaten öffnet sich eine Dropdown-Liste mit den noch nicht zugeordneten Kandidaten (zuoberst in der Liste befindet sich immer der Pausenkandidat), woraus der gewünschte Kandidat ausgewählt werden kann.

Rotation 1 14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Gruber Patrick
	KPW1	Müller Fritz
	HNO3	Muster Peter
	HNO2	BREAK Kandidat
	HNO1	BREAK Kandidat
Parcours 2	KPW2	Artho Pascal Roman
	KPW1	Aschmann Gustav
	HNO3	BEQIRAJ Faik
	HNO2	BERISHA Asdren
	HNO1	Baier Donat

Abbildung 69 - Dropdown Kandidaten

Durch das Ziehen eines Kandidaten an die gewünschte Position werden die beiden beteiligten Kandidaten miteinander vertauscht (dies gilt auch, wenn einer der beiden Kandidaten ein „Break“-Kandidat, also ein Pausenkandidat ist).

Rotation 1			Rotation 2		
14.06.2011 12:00 - 12:44			14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	Gruber Patrick	Parcours 3	KPW2	Aschmann Gustav
	KPW1	Müller Fritz		KPW1	BEQIRAJ Faik
	HNO3	Muster Peter		HNO3	BERISHA Asdren
	HNO2	Barandun Curdin		HNO2	Böhm Matthias
	HNO1	Bapst Daniel		HNO1	Bissegger Katja

Abbildung 70 - Kandidaten Exchange_1

Parcours 1	KPW2	Gruber Patrick	Parcours 3	KPW2	Aschmann Gustav
	KPW1	Müller Fritz		KPW1	BEQIRAJ Faik
	HNO3	BERISHA Asdren		HNO3	Muster Peter
	HNO2	Barandun Curdin		HNO2	Böhm Matthias
	HNO1	Bapst Daniel		HNO1	Bissegger Katja

Abbildung 71 - Kandidaten Exchange_2

Achtung: Es müssen alle Kandidaten einem Posten zugewiesen werden, welche der Prüfung hinzugefügt worden sind.

9.5.2.5.7 Formular-Ansicht

Rotation hinzufügen

Rotation 1			Rotation 2			Rotation 3		
14.06.2011 12:00 - 12:44			14.06.2011 12:00 - 12:44			14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	KPW 2.0	Parcours 3	KPW2	KPW 2.0	Parcours 4	KPW2	KPW 2.0
	KPW1	KPW 1.0		KPW1	KPW 1.0		KPW1	KPW 1.0
	HNO3	HNO 2.0		HNO3	HNO 2.0		HNO3	HNO 2.0
	HNO2	HNO 1.1		HNO2	HNO 1.1		HNO2	HNO 1.1
	HNO1	HNO 1.0		HNO1	HNO 1.0		HNO1	HNO 1.0
Parcours 2	KPW2	KPW 2.0						
	KPW1	KPW 1.0						
	HNO3	HNO 2.0						
	HNO2	HNO 1.1						
	HNO1	HNO 1.0						

Fertig

Abbildung 72 - Formulare-Ansicht

Den Posten wird jeweils derjenige Prüfungsbogen zugeteilt, welcher beim Erfassen des Postens ausgewählt wird. Um einen anderen Prüfungsbogen auszuwählen, klickt man auf den bestehenden Prüfungsbogen und wählt aus der Dropdown-Liste den gewünschten Prüfungsbogen aus.

Rotation 1		
14.06.2011 12:00 - 12:44		
Parcours 1	KPW2	KPW 2.0
	KPW1	KPW 1.0
	HNO3	HNO 2.0
	HNO2	HNO 1.1
	HNO1	HNO 1.0
Parcours 2	KPW2	KPW 2.0
	KPW1	KPW 1.0
	HNO3	HNO 2.0
	HNO2	HNO 1.1
	HNO1	HNO 1.0

Abbildung 73 - Dropdown Formulare

Die Formulare lassen sich nicht per Drag&Drop verschieben.

9.5.2.5.8 Prüfung abschliessen

Mit einem Klick auf den „Fertig“-Button in der Rotations-Ansicht kann die komplette Prüfung abgeschlossen werden.

Hinweis: Die Prüfung kann zu jeder Zeit per Klick auf den „Fertig“-Button im Wizard abgeschlossen werden. Dabei wird der aktuelle Stand gespeichert.

9.5.3 Prüfungsstatus

Auf der Startseite (Prüfungsplaner-Ansicht) erhält man einen Überblick über alle erfassten Prüfungen. Neben einigen Angaben zu den Daten der Prüfung (Anz. Rotationen, Parcours, Posten, Kandidaten) wird auch der Status einer Prüfung angezeigt. Die verschiedenen Status-Meldungen werden in der folgenden Tabelle aufgelistet.

Status	Bedeutung
Ungültig: (Die Prüfung ist unvollständig)	Der Status ist rot und es werden alle Fehler in einer Liste dargestellt. Die Prüfung kann so nicht den Examinatoren bereitgestellt werden. Durch Klicken auf den „Zum Fehler“-Button gelangt man direkt zum ausgewählten Fehler. Die Prüfung ist unvollständig - Die Durchführungszeiten der Rotationen überlappen sich. Zum Fehler - Es ist nicht jedem Posten ein Examinator zugewiesen. Zum Fehler
Gültig: (Die Prüfung ist vollständig)	Die Prüfung ist komplett ausgefüllt und enthält keine erkennbaren Fehler mehr. Durch einen Klick auf den „Exportieren“-Button wird die Prüfung den Examinatoren (resp. den Geräten der Examinatoren) bereitgestellt. Die Prüfung kann exportiert werden. exportieren
Bereitgestellt: (Die Prüfung ist bereit)	Die Prüfung wurde erfolgreich den Examinatoren bereitgestellt. Änderungen an der Prüfung haben keinen Einfluss mehr. Die Prüfung ist für die Durchführung bereitgestellt.
Laufend: (Die Prüfung läuft)	Sobald die Prüfung beginnt, wird automatisch dieser Status gesetzt. Die Prüfung wird zur Zeit durchgeführt.
Importieren: (Die Prüfung ist beendet)	Sobald die Prüfung zu Ende ist, kann der Benutzer durch Klicken auf den „Importieren“-Button die Prüfung in die Adminkomponente importieren. Die Prüfung kann importiert werden. importieren
Herunterladen: (Die Prüfung kann heruntergeladen werden)	Sobald die Prüfung importiert wurde, kann diese lokal auf dem Computer als Zip-Datei abgespeichert werden. Die Prüfungsergebnisse können heruntergeladen werden. herunterladen

Tabelle 19- Prüfungsstatus

9.6 Monitor

Der Monitor lässt sich mithilfe des „zum Monitoring“ Buttons auf der Loginseite öffnen.

Abbildung 74 – Monitor öffnen

9.6.1 Prüfung überwachen

Abbildung 75 – Übersicht Monitor

Sobald Fehler (rot) oder Probleme (gelb) auftreten erscheinen entsprechende Meldungen. Rote Fehler werden zusätzlich durch ein akustisches Alarmsignal gemeldet.

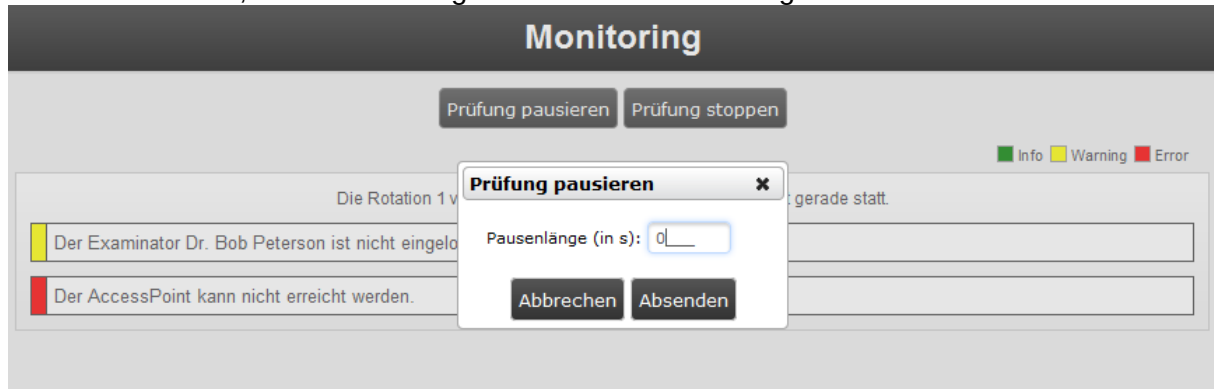
Die meisten Fehler verschwinden automatisch, sobald das entsprechende Problem gelöst wurde. Die übrigen Fehler können mittels „Fehler beheben?“- Button entfernt werden.

Abbildung 76 – Fehler beheben

9.6.2 Prüfung pausieren

Um eine Prüfung zu pausieren, muss der „Prüfung pausieren“ Button betätigt werden. Daraufhin kann die Länge der Pause in Sekunden eingegeben werden. Die Prüfung pausiert, sobald die aktuelle Bewertungszeit abgelaufen ist. Nachdem die eingegebene Pausen-

zeit verstrichen ist, wird die Prüfung automatisch wieder fortgesetzt.



9.6.3 Prüfung stoppen

Um eine Prüfung komplett zu stoppen, bietet der Monitor den Button „Prüfung stoppen“.

Achtung: Ist eine Prüfung gestoppt, kann sie nicht fortgesetzt werden.

9.6.4 Monitor auf dem Smartphone



Abbildung 77 - Monitor Smartphone [17]

Der Monitor kann auch im Browser eines Smartphones ausgeführt werden. Dabei wird die Seite automatisch dem Display des verwendeten Gerätes angepasst.

10 Zeitauswertung

Für die erfolgreiche Durchführung der Bachelorarbeit werden pro Teammitglied 12 ECTS-Punkte gutgeschrieben. Da 1 ECTS-Punkt ca. 30 Stunden Aufwand entspricht sollte jeder im Team mindestens 360 Stunden für die Arbeit aufwenden (insgesamt 1080 Stunden für alle drei Mitglieder). Das folgende Diagramm zeigt den Gesamtaufwand, wie er gefordert wurde (Total Vorgabe), wie das Team den Aufwand geschätzt hat (Total Soll) und wie gross der Aufwand tatsächlich war (Total Ist).

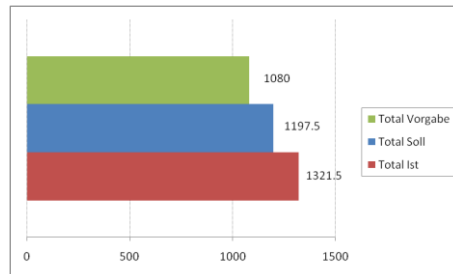


Abbildung 78 - Gesamtüberblick Zeitauswertung

Der Unterschied zwischen dem Soll- und dem Ist-Aufwand ist im nächsten Diagramm zu sehen. Daraus ist zu lesen, dass die Aufwandschätzung, obschon sie schon einiges höher ausgefallen ist als die Vorgabe, bei einigen Tasks zu niedrig ist.

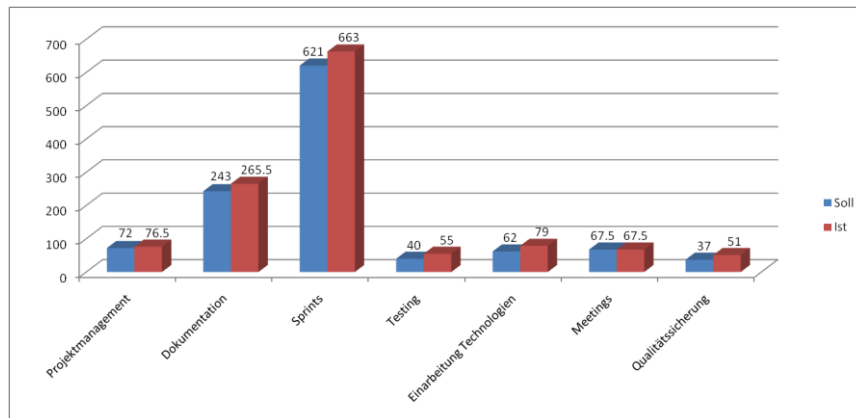


Abbildung 79 - Aufwand Ist-Soll

Das abschliessende Diagramm zeigt die Anteile der verschiedenen Tasks an den Ist-Aufwand. Knapp mehr als die Hälfte der Zeit hat die Implementation während den fünf Sprints eingenommen. Etwa ein Fünftel der Projektzeit wurde in die Dokumentation investiert, die restlichen Tasks nehmen den letzten Viertel ein.

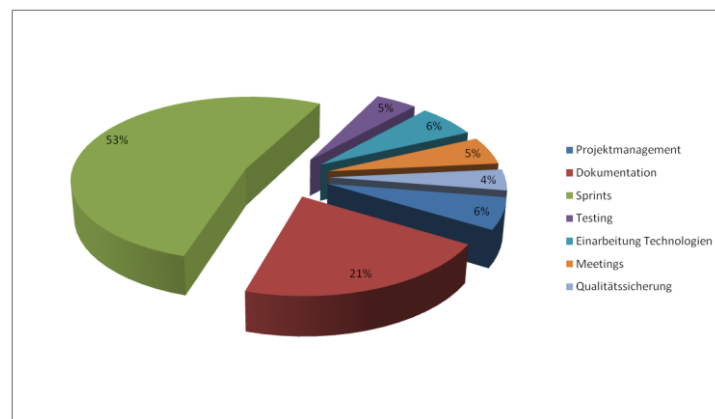


Abbildung 80 - Ist-Aufwand

11 Erfahrungsberichte

11.1 Matthias Good

11.1.1 Allgemein

Das Thema E-OSCE-Prüfungen hat von Anfang an mein Interesse geweckt. Der Ansatz, eine Prüfung mit einem komplett papierlosen Prozess durchzuführen, erschien mir äusserst innovativ. Anhand der anfänglichen Requirement-Liste, welche wir vom Institut für Medizin erhalten hatten, waren dem Projekt einige Vorgaben gesetzt, doch viele Entscheide (z.B. die verwendete Technologie) konnten durch das Team mitbestimmt werden.

11.1.2 Ablauf des Projektes

Als Erstes musste sich das Team für eine von mehreren möglichen Architekturen und Technologien entscheiden, wir einigten uns rasch auf eine Java Web-Applikation Lösung. Mich motivierte dabei vor allem die Implementation der Business-Logik mittels EJB's, da ich dies während der bisherigen Studienzeit immer als grosse Herausforderung angesehen habe.

Danach erstellten wir die Anforderungsspezifikation, in welcher auch die ersten Paperprototypes kreiert wurden. Diese entsprachen nicht ganz den Vorstellungen des Betreuerteams, sodass sie nochmals angepasst wurden. Mit der zweiten Version dieser Prototypes wurde dann eine heuristische Evaluation mit zwei Experten durchgeführt. Somit verfügten wir früh über einen Architektur-Prototypen, von welchem wir wussten, dass er allen gewünschten Anforderungen gewachsen war.

Die nachfolgende Implementierung forderte uns einiges ab. Sowohl das User Interface, welches komplizierte Interaktionen (wie z.B. Drag & Drop) im Browser unterstützen musste, als auch die Datenbank-Programmierung, welche erst durch intensive Tests genügend performant gemacht wurde, stellten uns vor schwierige Aufgaben. Doch mit grossem Einsatz jedes einzelnen Teammitglieds konnten alle Anforderungen, welche an das Projekt gestellt wurden, umgesetzt werden.

11.1.3 Teamarbeit

Wie bereits bei der Studienarbeit war die Arbeit mit Sven und Philipp sehr angenehm. Bei Fragen oder Problemen wurde einander kurz ausgeholfen, sodass das Projekt im Grossen und Ganzen planmässig durchgeführt werden konnte.

11.1.4 Fazit

Ich empfand die Bachelorarbeit als einiges intensiver als noch die Studienarbeit und ich war stets motiviert am Ende eine lauffähige Applikation abliefern zu können, welche eines Tages an der Universität Bern zum Einsatz kommen wird.

Durch das Betreuerteam erhielten wir jeweils sofortiges Feedback auf Demonstrationen des aktuellen Projektstandes, was uns für den weiteren Verlauf des Projektes weiterhalf.

Die 16 Wochen intensive Arbeit haben sich für mich sehr gelohnt, da ich dabei einiges im Bezug auf Projektplanung und -umsetzung dazu lernen konnte.

11.1 Sven Lenz

11.1.1 Allgemein

Das Projekt „Administration von E-OSCE Prüfungen“ war anfangs der Bachelorarbeit noch ein unbeschriebenes Blatt. Ausser dem Anforderungskatalog, den wir zu Beginn erhalten haben, hatten wir freie Wahl der einzusetzenden Technologie und dem Vorgehen zur Lösung des Projektes. In der Softwareentwicklung ist dies für mich eine der besten jedoch auch anspruchsvollsten Gelegenheiten, da von Beginn des Projektes an auf einen sauberen Aufbau der Architektur geachtet werden muss.

11.1.2 Ablauf des Projektes

Zu Beginn des Projektes war dem Bachelorarbeitsteam überlassen, eine geeignete Architektur und Technologie für die Administrationskomponente zu wählen. Nach der Evaluation der verschiedenen Möglichkeiten haben wir uns für eine Enterprise Applikation im Java Umfeld entschieden. Sowohl durch meine Arbeit bei der Finnova als auch die Studienarbeit an der HSR im vergangenen Jahr konnte ich bereits einiges im Bereich der Entwicklung eines Applikationsserver erlernen. Jedoch hat sich mir noch nie die Gelegenheit geboten, eine Enterprise Applikation von Grund auf zu gestalten.

Nach dem Projektsetup, welches doch noch einige Schwierigkeiten aufwies, da niemand von uns die dazu notwendige Erfahrung besass, ging es darum, ein geeignetes User Interface in einem Webbrowser darzustellen. Einen ersten Entwurf des UI mussten wir leider wieder verwerfen, da die gewünschten Arbeiten zwar erledigt werden konnten, jedoch den optimalen Benutzungskomfort nicht abdeckten. In einem zweiten Entwurf zusammen mit dem Betreuungsteam entwickelten wir dann ein anspruchsvolles UI, welches es nun in die Tat umzusetzen galt. Auch hier standen wir wieder vor Problemen, da ein anspruchsvolles und komplexes User Interface für einen Webbrowser zu entwickeln doch nicht so einfach war, wie wir es uns Anfangs vorgestellt haben. Schlussendlich haben wir meiner Meinung nach jedoch das maximal machbare in der Webentwicklung zustande gebracht und das UI ohne grössere Abstriche implementieren können.

Durch diese zeitintensiven Herausforderungen hatten wir in dieser Bachelorarbeit leider zu wenig Zeit, die Dokumentation noch detaillierter zu gestalten. Gerne hätte ich zum Beispiel noch eine ausführliche JSF-Komponentenstudie geschrieben, um die Erfahrungen in dieser Bachelorarbeit zu dokumentieren, damit zukünftige Weiterentwicklungen nicht in denselben Problemen stecken bleiben.

11.1.3 Teamarbeit

Die Zusammenarbeit mit Matthias und Philipp hat von Anfang an reibungslos funktioniert. Jeder hat seine Stärken in einem anderen Gebiet und so ergänzten wir uns optimal bei der Arbeit. Dies erleichterte uns eine klare Aufgabenverteilung um die Bachelorarbeit schnell und effektiv umzusetzen.

11.1.4 Fazit

Mit dem Resultat unserer Bachelorarbeit bin ich sehr zufrieden. Innert 16 Wochen haben wir alle Phasen einer Softwareentwicklung erfolgreich absolviert und können dem Auftraggeber eine lauffähige und auf die Arbeit massgeschneiderte Applikation liefern. Mich würde es freuen zu hören, dass diese Adminkomponente in Zukunft bei der Planung von E-OSCE Prüfungen eingesetzt wird.

11.1 Philipp Marugg

11.1.1 Allgemein

Am Ende des Herbstsemesters 2010 wurde uns in einem Meeting mit Prof. Hans Rudin und Kevin Gaunt das E-OSCE Projekt zum ersten Mal vorgestellt. Daher hatte ich von Anfang an eine gute Vorstellung vom genauen Umfang unserer Arbeit.

11.1.2 Ablauf des Projektes

Im ersten Schritt mussten wir uns für eine Technologie entscheiden. Wir entschieden uns für eine Java Enterprise Lösung mit einem Webbrowser basierten User-Interface. Obwohl ich persönlich eine Silverlight-Lösung bevorzugt hätte, empfand ich unsere Wahl als eine sehr interessante Variante. Ich konnte zum ersten Mal das gelernte Wissen aus dem Modul „Enterprise-Computing“ richtig einsetzen.

Ein weiterer spannender Teil der Arbeit waren das Thema Drag & Drop. Da es für uns sehr schwierig war den Zeitaufwand für dieses Feature abzuschätzen, entwickelten wir schnellstmöglich einen Prototyp der Rotationsübersicht. Dieser stimmte uns zuversichtlich und reduzierte das Risiko einer Fehleinschätzung des zukünftigen Arbeitsaufwandes.

Im Grossen und Ganzen verlief die Arbeit in den 16 Wochen relativ gut und ohne grössere Unannehmlichkeiten, einzig und allein das von uns eingesetzte JSF-Framework Primefaces war teilweise zum Haare raufen, da mehrere Komponenten Bugs aufwiesen.

11.1.3 Teamarbeit

Die Zusammenarbeit im Team funktionierte sehr gut. Die verschiedenen Teilaufgaben wurden vor jedem Sprint auf die verschiedenen Teammitglieder aufgeteilt und bei Problemen unterstützen wir uns stets gegenseitig um möglichst schnell eine Lösung zu finden.

11.1.4 Fazit

Mit dem Resultat unserer Bachelorarbeit bin ich sehr zufrieden. Sowohl das Erfassen von Prüfungsbogen, sowie die Prüfungsplanung empfinde ich als einfach und komfortabel zu bedienen. Zusätzlich konnte ich mir neues Know-How im Java-EE-Server Umfeld aneignen.

12 Glossar

Begriff	Definition
CRUD	Create, Read, Update, Delete
CSV	Comma-Separated Values
EIS	Enterprise Information System
E-OSCE Adminserver	Administrationserver, Adminkomponente, E-OSCE Adminkomponente
Growl	Angelehnt an das Benachrichtigungssystem von Mac OS X.
IFS	Institut für Software
IML	Institut für Medizinische Lehre
OSCE	Objective Structured Clinical Exams
UI	User Interface
SVN	Subversion

Tabelle 20 - Glossar

13 Literaturverzeichnis

13.1 Kapitel 2 (Management Summary)

- [00] weiterführende Informationen zum Projekt E-OSCE, <http://www.e-osce.ch/>, letzter Zugriff am 15.06.2011

13.2 Kapitel 5 (Analyse)

- [01] Persona1, http://www.krankenhaus-sulzbach-presse.de/website/img/pix/chefaerzte/urologie/chefarzt_dr_gib_thumb.jpg, letzter Zugriff am 31.05.2011
- [02] Persona2, http://www.klinikum.uni-heidelberg.de/fileadmin/pressestelle/images/pm148_08.jpg, letzter Zugriff am 31.05.2011

13.3 Kapitel 6 (Software Architektur)

- [03] jQuery, <http://jquery.com>, letzter Zugriff am 11.06.2011
- [04] JSF, <http://javaserverfaces.java.net>, letzter Zugriff am 11.06.2011
- [05] Primefaces, <http://www.primefaces.org>, letzter Zugriff am 11.06.2011
- [06] EJB 3.0, <http://www.jboss.org/ejb3>, letzter Zugriff am 11.06.2011
- [07] JAXB 2.0, <http://jaxb.java.net>, letzter Zugriff am 11.06.2011
- [08] JBoss 6.0, <http://www.jboss.org/jbossas/downloads.html>, letzter Zugriff am 11.05.2011
- [09] PostgreSQL 9.0 , <http://www.postgresql.org/download/>, letzter Zugriff am 11.05.2011
- [10] Eclipse IDE, <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/heliossr2>, letzter Zugriff am 11.05.2011

13.4 Kapitel 8 (Projektmanagement)

- [11] Scrum, <http://de.wikipedia.org/wiki/Scrum>, letzter Zugriff am 13.06.2011
- [12] RUP, http://de.wikipedia.org/wiki/Rational_Unified_Process, letzter Zugriff am 13.06.2011
- [13] Subclipse (Eclipse Plugin), <http://subclipse.tigris.org/>, letzter Zugriff am 13.06.2011
- [14] Dropbox, <http://www.dropbox.com/>, letzter Zugriff am 13.06.2011
- [15] Atlassian Confluence, <http://www.atlassian.com/software/confluence/>, letzter Zugriff am 13.06.2011
- [16] JUnit 4.0, <http://www.junit.org/>, letzter Zugriff am 13.06.2011

13.1 Kapitel 9 (Benutzerdokumentation)

- [17] iPhone, <http://www.apple.com/iphone/ios4/>, letzter Zugriff am 15.06.2011

14 Anhangsverzeichnis

Die folgenden Dokumente befinden sich im Anhang

Referenz	Titel	Dokumentname
[ANH00]	eosceAdmin Applikation	00_eosceAdmin.zip
[ANH01]	eosceAdmin Sourcecode	01_eosceAdminSource.zip
[ANH02]	Paperprototypen in der Version 1 und 2	02_paperprototypes.zip
[ANH03]	Poster	03_Poster.ppt
[ANH04]	Protokolle und Traktanden	04_Protokolle_Traktanden.docx
[ANH05]	Projektplan	05_Projektplan.xlsx
[ANH06]	Product Backlog	06_productBacklog.xlsx
[ANH07]	Sprint Backlog 1	07_sprintBacklog01.xlsx
[ANH08]	Sprint Backlog 2	08_sprintBacklog02.xlsx
[ANH09]	Sprint Backlog 3	09_sprintBacklog03.xlsx
[ANH10]	Sprint Backlog 4	10_sprintBacklog04.xlsx
[ANH11]	Sprint Backlog 5	11_sprintBacklog05.xlsx
[ANH12]	Velocity	12_velocity.xlsx
[ANH13]	Testskript heuristische Evaluation	13_Testskript_heuristische_Evaluation.docx
[ANH14]	Heuristische Evaluation Proband 1	14_heuristische_evaluation1.m4v
[ANH15]	Heuristische Evaluation Proband 2	15_heuristische_evaluation2.m4v
[ANH16]	Aufgabenstellung Bachelorarbeit „E-OSCE Adminserver“	16_eOsceAdminServer_Aufgabenstellung.doc
[ANH17]	Vereinbarung über Urheber- und Nutzungsrechte	17_Vereinbarung.pdf
[ANH18]	Klassendiagramm	18_ClassDiagram.bmp

Tabelle 21 - Anhangsverzeichnis

15 Tabellenverzeichnis

Tabelle 1 - Use Cases	19
Tabelle 2 - User Stories	23
Tabelle 3 - Domain Model	26
Tabelle 4 - Paperprototypes	33
Tabelle 5 - Technologien	40
Tabelle 6 - Namenskonventionen	42
Tabelle 7 - Applikationen	44
Tabelle 8 - eosceAdminApp Klassen	50
Tabelle 9 - eosceAdminDaoInterface Klassen	51
Tabelle 10 - eosceAdminDao Klassen	52
Tabelle 11 - eosceAdminDomain Klassen	53
Tabelle 12 - eosceAdminTest Klassen	54
Tabelle 13 - Datenbanktabellen	65
Tabelle 14 - Projektmitarbeiter	73
Tabelle 15 - Software	75
Tabelle 16 - Risikoplan	78
Tabelle 17 – Ordnerstruktur	82
Tabelle 18 - eosceAdmin.properties	82
Tabelle 19- Prüfungsstatus	102
Tabelle 20 - Glossar	109
Tabelle 21 - Anhangsverzeichnis	111

16 Abbildungsverzeichnis

Abbildung 1 - OSCE Prozess	7
Abbildung 2 - E-OSCE Prozess	7
Abbildung 3 - Gesamtüberblick E-OSCE	8
Abbildung 4 - Übersicht über komplette Prüfung	9
Abbildung 5 - Persona 1 [01]	15
Abbildung 6 - Persona 2 [02]	16
Abbildung 7 - Use Case Model	18
Abbildung 8 - DomainModel	25
Abbildung 9 - Überblick der verwendeten Komponenten	38
Abbildung 10 - technische Architektur	40
Abbildung 11 – Deployment Diagramm	43
Abbildung 12 - Übersicht App, DaoInterface und Dao	45
Abbildung 13 - Klassendiagramm Domain (eosceAdminDomain)	46
Abbildung 14- Klassendiagramm Test (eosceAdminTest)	47
Abbildung 15 - Sequenzdiagramm CRUD	55
Abbildung 16 - Sequenzdiagramm Plan generieren	56
Abbildung 17 - Sequenzdiagramm Export	57
Abbildung 18 - Sequenzdiagramm Monitor	58
Abbildung 19 - Datenmodell	59
Abbildung 20 - Klassendiagramm Schema (eosceAdminSchema)	66
Abbildung 21 - Überblick Berechnung Parcours	69
Abbildung 22 - Testreihe 2	69
Abbildung 25 - Testreihe 4 List	70
Abbildung 23 - Testreihe 3a	70
Abbildung 24 - Testreihe 3b	70
Abbildung 26 - Testreihe 4 Prepare	71
Abbildung 27 - Testreihe 4 Update	71
Abbildung 28 - Testreihe 5	71
Abbildung 29 - Gesamtüberblick Tests	72
Abbildung 30 - Editor-Übersicht	83
Abbildung 31 - Neuer Prüfungsbogen	83
Abbildung 32 - Neue Prüfungsbogenversion	84
Abbildung 33 - Leerer Prüfungsbogen	84
Abbildung 34 - Neuer Antworttyp	85
Abbildung 35 - Neues Kriterium	85
Abbildung 36 - Drag&Drop Bereich 01	86
Abbildung 37 - Drag&Drop Bereich 02	86
Abbildung 38 - Drag&Drop Bereich 03	86
Abbildung 39 - Posten-Übersicht	87
Abbildung 40 - Neuer Posten	87
Abbildung 41 - Examinatoren-Übersicht	88
Abbildung 42 - Neuer Examinator	88
Abbildung 43 - Examinatorenliste importieren	88

Abbildung 44 - Struktur Examinatorenliste	89
Abbildung 45 - Kandidaten-Übersicht	89
Abbildung 46 - Neuer Kandidat	89
Abbildung 47 - Kandidatenliste importieren	90
Abbildung 48 - Struktur Kandidatenliste	90
Abbildung 49 - Infrastruktur-Übersicht	91
Abbildung 50 - Neues Gerät	91
Abbildung 51 - Planer-Übersicht	91
Abbildung 52 - Wizard	92
Abbildung 53 - Informationen	92
Abbildung 54 - Posten	93
Abbildung 55 - Kandidaten	93
Abbildung 56 - Parcours	94
Abbildung 57 - Rotationen	96
Abbildung 58 - Drag&Drop Parcours	97
Abbildung 59 - Drag&Drop Posten	97
Abbildung 60 - Examinatoren-Ansicht	98
Abbildung 61 - Dropdown Examinatoren	98
Abbildung 62 - Examinatoren_Exchange_2	99
Abbildung 63 - Examinatoren_Exachange_1	99
Abbildung 64 - Variante01_1	99
Abbildung 65 - Variante 01_2	99
Abbildung 66 - Variante 02_1	99
Abbildung 67 - Variante02_2	100
Abbildung 68 - Kandidaten-Ansicht	100
Abbildung 69 - Dropdown Kandidaten	100
Abbildung 70 - Kandidaten Exchange_1	101
Abbildung 71 - Kandidaten Exchange_2	101
Abbildung 72 - Formulare-Ansicht	101
Abbildung 73 - Dropdown Formulare	101
Abbildung 74 – Monitor öffnen	103
Abbildung 75 – Übersicht Monitor	103
Abbildung 76 – Fehler beheben	103
Abbildung 77 - Monitor Smartphone [17]	104
Abbildung 78 - Gesamtüberblick Zeitauswertung	105
Abbildung 79 - Aufwand Ist-Soll	105
Abbildung 80 - Ist-Aufwand	105