



KINOBUCHUNGSSYSTEM

LEISTUNGSBEURTEILUNG M326

CHRISTOPHER O'CONNOR

TECHNISCHE BERUFSSCHULE ZÜRICH, AP17D



INHALTSVERZEICHNIS

PROJEKTAUFTRAG	2
AUSGANGSLAGE	2
ZIELSETZUNG	2
RAHMENBEDINGUNGEN	2
IDENTIFIZIEREN VON KLASSEN	4
CRC-KARTEN	5
SZENARIOS (USE CASE)	7
PROTOKOLL DER DURCHGEFÜHRTEN SZENARIOS.....	7
KLASSENENTWURF	7
DESIGN PATTERN	8
FACADE	8
MVC	8
SEQUENZDIAGRAMME	9
ANFRAGE FREIE PLÄTZE IN DER VORSTELLUNG XYZ.....	9
PLÄTZE RESERVIEREN	9
DATENBANK ERM	10
GUI – ERGONOMIE	10
TESTUNG	11
TESTFÄLLE	11
ANHANG	13
SOZIALKOMPETENZ	13
BENUTZERFEEDBACK.....	13
PROTOKOLL	14

PROJEKTAUFTRAG

<i>Auftraggeber</i>	<i>Marcel Bernet</i>
<i>Auftragnehmer</i>	<i>Christopher O'Connor</i>

AUSGANGSLAGE

Das Kinobuchungssystem sollte Platzreservierungen für mehrere Kinosäle verwalten. Jeder Kinosaal hat Plätze, die in Reihen angeordnet sind. Kinobesucher können Plätze reservieren und bekommen eine Reihen- und eine Platznummer zugewiesen. Sie können nach nebeneinanderliegenden Plätzen fragen.

Jede Platzreservierung gilt für eine bestimmte Vorstellung (also einen bestimmten Film zu einem bestimmten Zeitpunkt). Vorstellungen finden zu festgelegten Zeitpunkten an festgelegten Tagen statt und sind einem Kinosaal zugeteilt, in dem sie gezeigt werden. Das System speichert die Telefonnummer des Kinobesuchers.

ZIELSETZUNG

Als Lernprodukt wird ein vollständiges Projekt erwartet. Dabei ist das bei den Designübungen erarbeitete Kinobuchungssystem (KBS) zu implementieren.

Von diesem KBS müssen im Rahmen der Analyse und Design die zentralen Aufgaben mit Use-Cases dargestellt und davon Szenarios erarbeitet und verfeinert werden. Dabei soll das CRC-Verfahren für die Klassenfindung und deren Beziehungen untereinander angewendet werden.

Es soll ein lauffähiges, getestetes Programm mit einem ergonomisch gestalteten GUI erstellt werden.

RAHMENBEDINGUNGEN

Die zugrundeliegende Architektur und deren Umsetzung mit Entwurfsmuster muss durch entsprechende UML-Diagramme dargestellt und beschrieben werden.

Projektart:	Software-Lösung Desktop Applikation	
Projektkunde(n):	Offene Kundschaft	
Projektdauer:	Geplanter Beginn:	22.02.19
	Geplantes Ende:	10.05.19
Projektteilziele und -ergebnisse:	Teilziele:	Ergebnisse:
	Datenbank Design und Erstellung	<ul style="list-style-type: none">Datenbank ist bereit für Laufzeit, DB-Server installiert
	Dateisystem	<ul style="list-style-type: none">Installieren und Aufbauen des Dateisystems auf Dateiserver. Bereit für Laufzeit
	Entwicklung des serverbasierten Services	<ul style="list-style-type: none">Verknüpfung zu Datenbank und Dateisystem
	User Interface	<ul style="list-style-type: none">Design und Verknüpfung zu internen Logik fertig gestellt
Meilensteine:	Meilensteine:	Datum:
	Projektauftrag	22.02.2019
	Serverteil fertig gestellt (Datenbank, Service, Dateisystem)	29.03.2019
	Clientteil (User Interface)	05.04.2019
	Testung	03.05.2019
Projektorganisation:	Kernteam:	
	<ul style="list-style-type: none">Christopher O'Connor Sonstige Projektbeteiligte:	
	<ul style="list-style-type: none">Web- und Server-Hoster	
Projektressourcen:	Ressourcen:	Menge:
	Personal	1
	Entwicklungsumgebung	1
	-Programmierung	1
	-Design	1
Projektrisiken und -unsicherheiten:	Technische Umsetzungsprobleme, Know-How	

IDENTIFIZIEREN VON KLASSEN

Substantive: Verben:

<i>Kinobuchungssystem</i>	Verwaltet (Platzreservierung) Speichert (Telefonnummer)
<i>Vorstellung</i>	Verwaltet (Platzreservierung, Film, Zeitpunkt, Kinosaal)
<i>Kinosaal</i>	Speichert (Vorstellung)
<i>Zeitpunkt</i>	Speichert (Datum, Zeit, Ort)
<i>Film</i>	Speichert (Name, Sprache)
<i>Platzreservierung</i>	Verwaltet (Reihe)
<i>Reihe</i>	Verwaltet (Plätze) Speichert (Plätze)
<i>Platz</i>	Speichert (Frei/Belegt) Speichert (Kinobesucher)
<i>Kinobesucher</i>	Speichert (Telefonnummer)
<i>Telefonnummer</i>	Speichert (Nummer)

CRC-KARTEN

CRC steht für Class/Responsibilities/Collaborators. Die Idee besteht darin, für jede Klasse eine Pappkarte zu verwenden.

<i>Klassenname</i>	
<i>Zuständigkeiten</i>	<i>Partnerklassen</i>

<i>Kinobuchungssystem</i>	
<i>Kann Vorstellungen nach Filmtitel und Tag finden</i> <i>Speichert eine Sammlung von Vorstellungen</i> <i>Speichert eine Sammlung von Kinosäale</i> <i>Akzeptiert Platzreservierungen vom Benutzer</i> <i>Sucht nach Reservationen eines Kinobesuchers</i> <i>Kann Plätze durchsuchen nach (Datum, Telefonnummer)</i> <i>Neue Kinosäle Erfassen</i> <i>Neue Vorstellungen Erfassen</i> <i>Speichert Sammlung von Filmen</i>	<i>Vorstellung</i> <i>Sammlung (Vorstellung)</i> <i>Kinosaal</i>

<i>Sammlung (Vorstellung)</i>	
<i>Speichert Vorstellungen</i>	<i>Vorstellung</i>

<i>Vorstellung</i>	
<i>Kann Plätze reservieren</i> <i>Speichert Kinosaal</i> <i>Speichert Reihen</i> <i>Speichert Zeitpunkt</i> <i>Speichert Film</i> <i>Kann alle Reservationen dieser Vorstellung Ausgeben</i>	<i>Platzreservierung</i> <i>Kinosaal</i> <i>Reihe</i> <i>Zeitpunkt</i> <i>Film</i> <i>Kinobesucher</i>

<i>Sammlung (Kinosaal)</i>	
<i>Speichert Kinosäle</i>	<i>Kinosaal</i>

<i>Kinosaal</i>	
<i>Speichert Name</i> <i>Speichert Reihen</i>	<i>Reihe</i>

Zeitpunkt	
Speichert Datum Speichert Zeit	

Sammlung (Filme)	
Speichert Filme	Film

Film	
Speichert Name des Films Speichert Sprache des Films	

Platzreservierung	
Kann Plätze Reservieren Kann Plätze Stornieren Kann freie Plätze nebeneinander finden Kann Plätze durchsuchen nach (Datum, Telefonnummer)	Reihe Platz

Reihe	
Speichert Sammlungen von Plätzen Akzeptiert Reservierungsanfrage für einen Platz Kann Plätze über ihre Nummer finden Kann freie Plätze nebeneinander Finden	Platz

Platz	
Akzeptiert Reservierungen Speichert Reservierungszustand (Frei/Reserviert) Akzeptiert Stornierungen Speichert Reservierender Kinobesucher Zustand der anliegenden Plätze können angefragt werden	Kinobesucher

Kinobesucher	
Speichert Telefonnummer Speichert Name	Telefonnummer

Telefonnummer	
Speichert Telefonnummer	

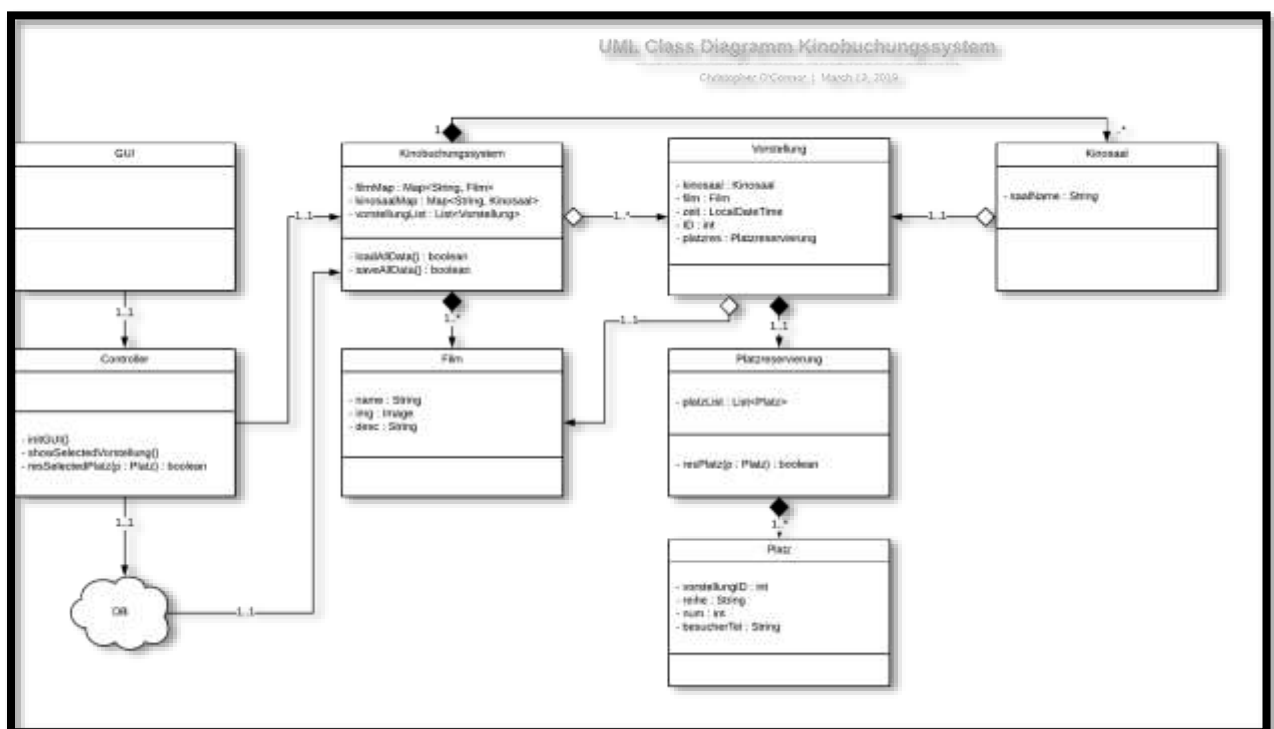
SZENARIOS (USE CASE)

PROTOKOLL DER DURCHFÜHRTEN SZENARIOS

1. Ein Kunde möchte fünf Plätze nebeneinander reservieren.
2. Ein Kunde ruft an und teilt mit, dass er die Platznummern, die er gestern reservieren liess, vergessen hat.
3. Ein Kunde ruft an und möchte seine Reservierung stornieren.
4. Eine Kundin, die bereits reserviert hat, ruft an. Sie möchte wissen, ob sie einen weiteren Platz direkt neben ihren bereits reservierten Plätzen bekommen kann.
5. Eine Vorstellung wird abgesagt. Das Kino möchte alle Kunden anrufen, die für diese Vorstellung reserviert haben.
6. Das System soll für ein neues Kino eingerichtet werden.
7. Ein neuer Film kommt in die Kinos.

KLASSENENTWURF

Der grobe Klassenentwurf wurde aus den erarbeiteten CRC-Karten erstellt. Diese sind grösstenteils aus der Auftragsdokumentation erarbeitet durch vordefinierte Use-Cases. Durch bereits existierende Klassen in der Standard Java Library konnte Bspw. Die CRC-Karte «Zeitpunkt» durch Java.DateTime ersetzt werden.



DESIGN PATTERN

Nachfolgende Designpatterns werden in diesem Projekt realisiert:

Facade

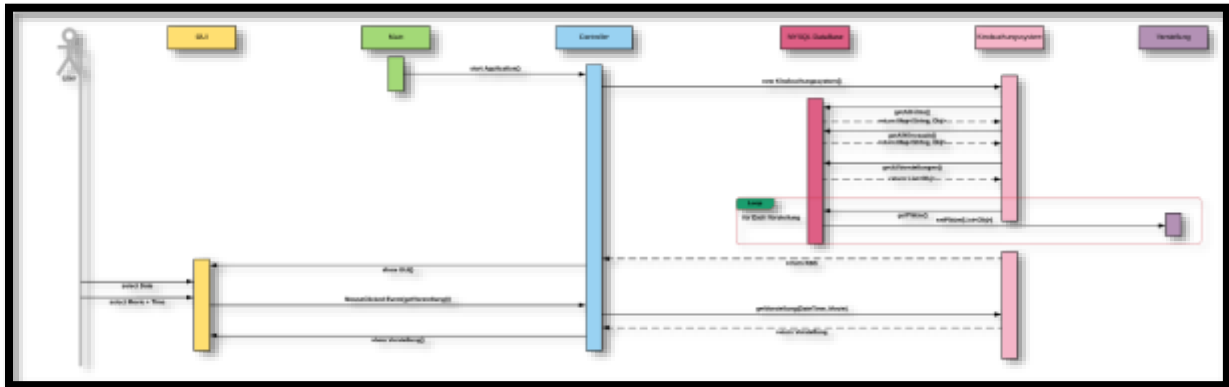
Verbirgt die Komplexität des Systems und bietet dem Client eine Schnittstelle, über die der Client auf das System zugreifen kann. Dieses Pattern umfasst eine einzige Klasse, die vereinfachte Methoden bereitstellt, die vom Client benötigt werden, und Aufrufe von Methoden an vorhandene Systemklassen delegiert.

MVC

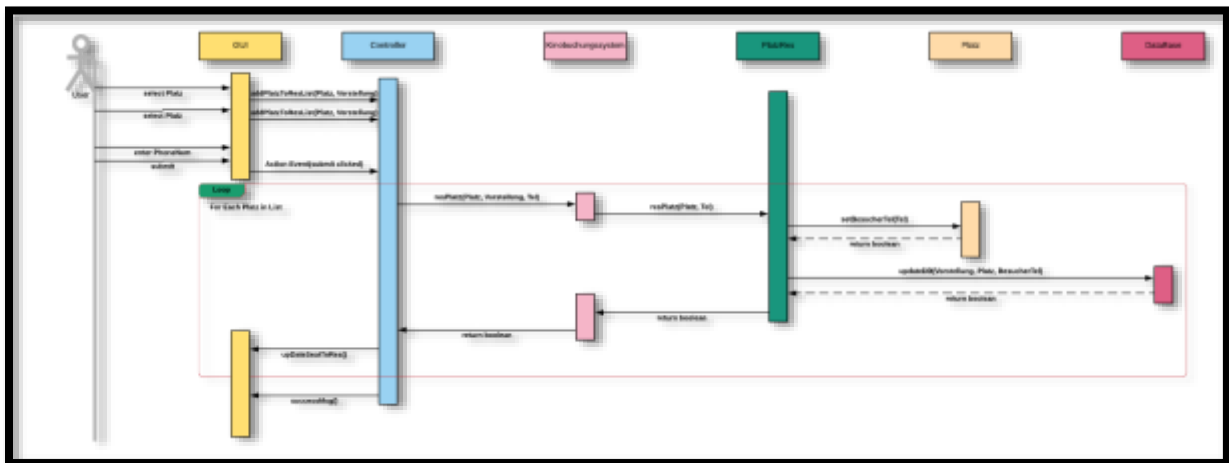
Steht für Model-View-Controller. Dieses Pattern wird verwendet, um die Bedenken der Anwendung zu trennen.

- Modell
 - Repräsentiert ein Objekt welches Daten beinhaltet. Es kann auch eine Logik zum Aktualisieren des Controllers aufweisen, wenn sich seine Daten ändern.
- View
 - Stellt die Visualisierung der Daten dar, die das Modell enthält.
- Controller
 - Wirkt auf das Modell und Ansicht. Sie steuert den Datenfluss in das Modellobjekt und aktualisiert die Ansicht, wenn sich Daten ändern.

ANFRAGE FREIE PLÄTZE IN DER VORSTELLUNG XYZ

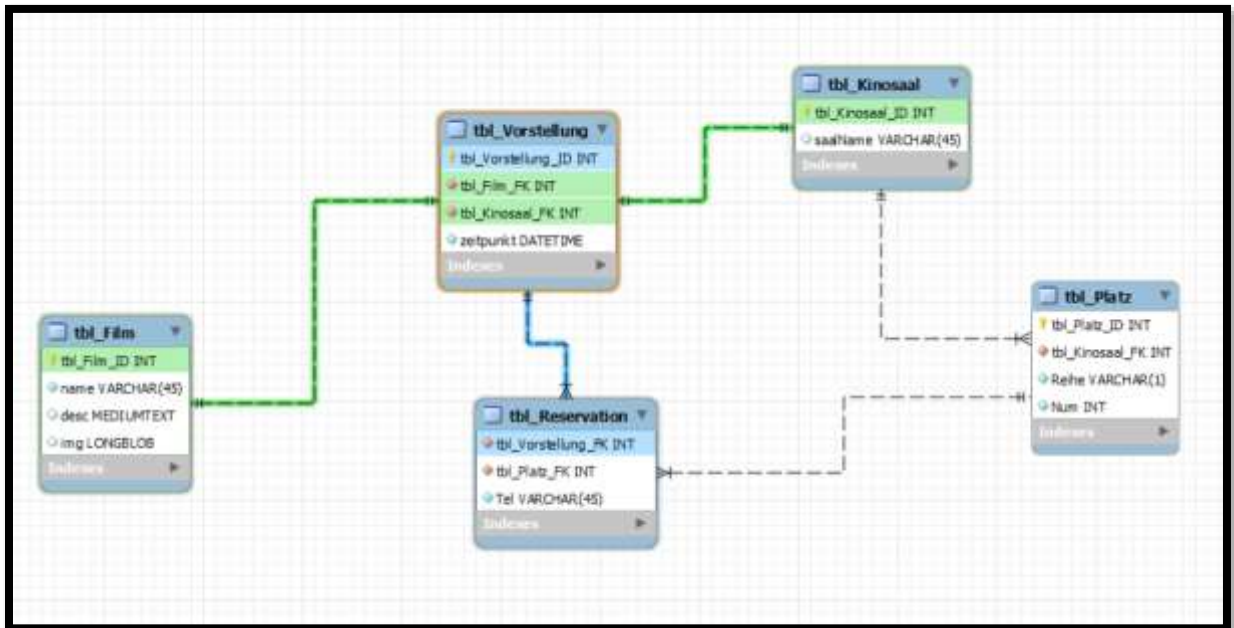


PLÄTZE RESERVIEREN



DATENBANK ERM

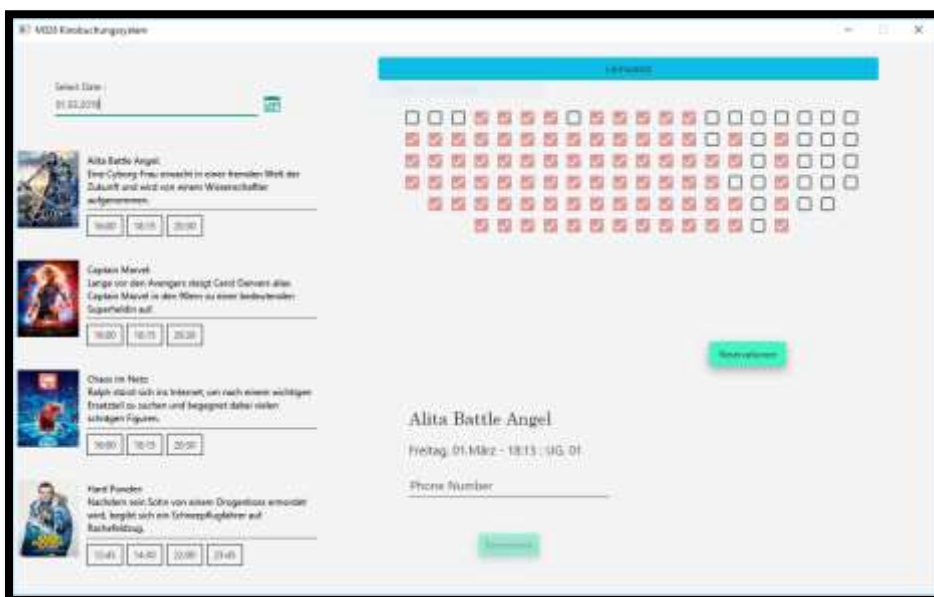
Screenshot vom Datenbank ERM erstellt mit MySQL-Workbench. Das ERM wurde aus dem Klassendiagramm abgeleitet.



GUI – ERGONOMIE

Bei der Software Ergonomie wird der Schwerpunkt auf eine intuitiv, leicht bedienbare Software fallen in dessen Kern es um den Arbeitsauftrag geht und nicht auf zusätzliche «Neat Futures» gesetzt wird.

Um die Klassische Web Ergonomie nach zu bilden wird es wie es bereits Standard geworden ist oben links ein Auswahlfeld geben und am linken Rand das Selektionsmenü. Im Center können dann die Einstellungen vorgenommen werden.



TESTUNG

Für die Testung der Applikation wurden im Verlauf der Realisierung JUnit Test Cases hinzugefügt damit die Funktionalität getestet und während der Weiterentwicklung gewährleistet werden kann.

TESTFÄLLE

ID	Scenario	Schritte	Erwartetes Resultat	Erhaltenes Resultat	Bestanden
01	Applikation starten + Daten aus DB laden	1. Applikation starten	GUI wird angezeigt und es gibt keine Error Meldungen	Wie erwartet	Bestanden
02	Zeige Filme vom 01.03.2019	1. Datum in Datumsfeld auswählen	Liste aller Filme am gewählten Datum werden angezeigt mit der Korrekten Vorstellungswelt	Wie erwartet	Bestanden
03	Zeige Filme vom 02.03.2019 (Keine Vorstellungen an diesem Tag)	1. Datum in Datumsfeld auswählen	Liste aller Filme erscheint Leer mit dem Verweis dass heute keine Vorstellungen stattfinden.	Wie erwartet	Bestanden
04	Vorstellung Auswählen und Saal anzeigen	1. Datum in Datumsfeld auswählen 2. Vorstellung auswählen	Vorstellungssaal wird angezeigt mit den bereits reservierten Plätzen.	Wie erwartet	Bestanden
05	Sitzplatz Reservieren	1. Vorstellung auswählen 2. Sitzplatz wählen 3. Telefonnummer eintragen 4. Reservieren klicken	Bestätigungsmeldung über die erhaltene Reservation. Platz wird reserviert und als vergeben angezeigt.	Wie erwartet	Bestanden
06	Sitzplatz Reservieren eines nicht verfügbaren Platzes	1. Vorstellung auswählen 2. Sitzplatz wählen 3. Telefonnummer eintragen 4. Reservieren klicken	Nicht verfügbarer Sitz ist nicht wählbar	Wie erwartet	Bestanden
07	Sitzplatz Reservieren mit Falscher oder Ohne deklarierten Telefonnummer	1. Vorstellung auswählen 2. Sitzplatz wählen 3. Telefonnummer falsch eintragen 4. Reservieren klicken	Button: Reservieren Nicht anklickbar mit ungültiger Telefonnummer	Wie erwartet	Bestanden

08	Mehrere Sitzplätze Reservieren	1. Vorstellung auswählen 2. n Sitzplätze auswählen 3. Telefonnummer eintragen 4. Reservieren klicken	Bestätigungsmeldung über die erhaltene Reservation. Plätze werden reserviert und als vergeben angezeigt.	Wie erwartet	Bestanden
09	Platzreservation stornieren	1. Vorstellung auswählen 2. Reservationen klicken 3. Reservation -> Delete klicken 4. Apply klicken	Ausgewählte Reservation wird entfernt und der Sitz wird freigegeben	Wie erwartet	Bestanden
10	Platzreservation Stornierung abrechen	1. Vorstellung auswählen 2. Reservationen klicken 3. Reservation -> Delete klicken 4. Cancel klicken	Ausgewählter Sitz wird nicht wie freigegeben	Wie erwartet	Bestanden
11	Platzreservation stornieren von mehreren Plätzen	1. Vorstellung auswählen 2. Reservationen klicken 3. Reservationen -> Delete klicken 4. Apply klicken	Alle Ausgewählten Reservationen werden entfernt und die Sitze werden freigegeben	Wie erwartet	Bestanden

Durch das ausgiebige Testen während der Realisierung konnten bereits während dessen Bugs behoben werden. Im Nachhinein mussten deswegen keine Änderungen vorgenommen werden.

ANHANG

SOZIALKOMPETENZ

Um meine Sozialkompetenz nachweisen zu können bei dieser Einzelarbeit möchte ich gerne festhalten, dass ich mich während der Idee Findung und erarbeiten der CRC-Karten sowie bei Unklarheiten, mich ausgiebig mit Herrn Savas austauschte. Wir haben ausserdem unsere individuellen Lösungen verglichen, hinterfragt und angepasst nach dem jeweiligen Feedback.

BENUTZERFEEDBACK

Das GUI ist sehr schlicht und auf das wesentliche gehalten und erfüllt den Arbeitsauftrag. Die Bedienung ist sehr intuitiv und ist nicht unnötig komplex. Auch die Anordnung der einzelnen Elemente ist meinem Empfinden nach sehr gut gelungen. Sie weicht nicht vom «Mainstream» ab, dadurch findet man sich sehr schnell auch ohne eine Erklärung zurecht. «Mir gefällt's».

PROTOKOLL

Tag 1, 22.02.19

Einführung in das Modul 326 – Objektorientierte Analyse & Design

In den Auftrag eingesehen und Teams gebildet sowie kleine Aufträge erledigt.

Tag 2, 01.03.19

Projektauftrag angenommen. Ausgangslage und Zielsetzung wurde angeschaut sowie dokumentiert. «Identifizierung von Klassen finden» Auftrag bearbeitet und die «Verb-Substantiv Methode» angewendet.

Diese Methode war sehr neu für mich. Durch den erhaltenen Fliesstext war diese Aufgabe schnell erledigt. In Zukunft werde ich diese Methode sicherlich weiterhin anwenden.

Tag 3, 08.03.19

Aus der «Verb-Substantiv Methode» wurden CRC-Karten abgeleitet und erstellt.

Auftrag zu den Szenarios «Use-Cases» wurde ausgeführt und protokolliert.

CRC-Karten waren für mich absolutes Neuland. Ich konnte mich schnell an die Richtlinien halten und konnte den Auftrag erfolgreich umsetzen.

Die vorgegebenen Szenarios waren sehr leicht verständlich für mich und sehr ausgiebig beschrieben. Während dieser Arbeit konnte ich mir ein genaueres Bild vom gewünschten KBS verschaffen. Die UseCases werden auch in Zukunft angewendet.

Tag 4, 15.03.19

UML-Klassendiagramm erstellt aus den vorhandenen CRC-Karten. Passende Designpatterns wurden gesucht und dokumentiert.

ERM – Datenbankmodell erstellt nach Bedürfnissen meiner Applikation.

Klassendiagramme waren für mich nichts Neues. Mir war nicht klar wie simpel Designpatterns sein können und dass ich einige wie z.B. das MVC bereits angewendet habe. Durch die verschiedenen vielfältigen Designpatterns «durch zu wühlen» und mich genauer zu den einzelnen einzulesen hat mir sehr viel Spass gemacht. Das Umdenken vom Klassendiagramm zu einem ERM war bei dieser Aufgabe eigentlich kein grosses Problem und konnte für die definierten Anforderungen umgesetzt werden.

Tag 5, 22.03.19

Dokumentation auf den aktuellsten Stand bringen und überarbeiten für die Abgabe.

Es wurde nicht ganz klar kommuniziert was von dieser Dokumentation erwartet wird. Dadurch war der Auftrag eher offen gestaltet. Das hat mir persönlich zugesagt und ich konnte mich «Freier» bewegen. Durch einen kleinen Einschub von Erwartungen über diese Design Dokumentation der Lehrperson wurden Unklarheiten spezifiziert. Dadurch musste ich einzelne Anpassungen und Ausformulierungen Nachtragen.

Tag 6, 29.03.19

Am heutigen Tag durfte ich mich endlich der Realisierung widmen. Durch das Vorwissen welches ich mir in anderen Projekten aneignen konnte mit JavaFx, musste ich nicht grössere Recherchen durchführen und konnte gleich mit dem Umsetzen starten. Beginnend mit dem MockUp konnte ich schnell voranschreiten und die ersten Design Ideen welche in den letzten Wochen durch meinen Kopf gingen umsetzen. Ich habe mich vorwiegend mit dem Anordnen von der Ordner Struktur gewidmet und die Schnittstellen definiert (GUI -> Controller -> Backend -> DB).

Tag 7, 05.04.19

Durch die von mir selbst definierten Schnittstellen habe ich heute das Backend erstellt. Das Hinzufügen von Testdaten in der DB erwies sich als sehr nützlich während den kleineren Tests von einzelnen Methoden mit Konsolenausgaben. Letztlich habe ich noch das GUI grafisch gefertigt.

Tag 8, 12.04.19

Heute habe ich alle Schnittstellen miteinander verbunden und mögliche Testscenarios getestet. Nach einzelnen Verbesserungen habe ich dem Projekt noch den Letzten Schliff verpasst und die Dokumentation vervollständigt.

Tag 9, 10.05.19: Fazit

Zum heutigen Tag gibt es nicht mehr sehr viel beizutragen, denn heute ist der Abgabe Termin. Ich empfinde meine Arbeit als sehr gelungen und freue mich auf ähnliche arbeiten im Geschäftsalltag.

Viele Teile dieses Moduls kannte ich bereits, dennoch konnte ich viel dazu lernen und mitnehmen.