

Fernfachhochschule Schweiz

BSc Informatik  
PA\_5 - Projektarbeit

---

# Studienplaner

Plane dein Studium selbst - behalte den Überblick

---

*Modul:* PA\_5, Projektarbeit, BE-Sa-1

*Dozenten:* Winiger Alexander und Kuepfer Severin

*Semester:* HS21/22

*Studenten:* Christoph Grossmann, Marco Ris, Peter Spielhofer

*Erstellungsdatum:* 23. August 21

**Änderungsverzeichnis**

Version	Datum	Änderung	Author
0.1	<b>21.08.2021</b>	<b>Initialisierung (Struktur, Formatvorlagen, Kopf- und Fusszeile)</b>	<b>Christoph Grossman</b>
0.2	<b>21.08.2021 -</b>	<b>Projektphase – Fortwährende Überarbeitung der Dokumentation</b>	<b>Projektteam</b>
1.0	<b>18.12.2021</b>	<b>Finalisierung</b>	<b>Projektteam</b>

## Inhaltsverzeichnis

<b>1 Management Summary .....</b>	8
<b>2 Einleitung .....</b>	9
<b>3 Projektauftrag .....</b>	9
3.1 Ausgangslage ( <i>Ist-Situation</i> ) .....	9
3.2 Projektidee und Zielpublikum .....	10
3.3 Projektziele .....	11
3.4 Mindestanforderung .....	12
3.5 Grobanforderungen .....	12
3.6 Abgrenzung .....	13
3.7 Rahmenbedingungen / Restriktionen / Technische Ressourcen .....	14
3.8 Eingesetzte Ressourcen .....	14
3.9 Projektzeitraum .....	15
3.10 Kostenaspekt .....	15
3.11 Grundlagen .....	16
3.12 Projektphasen .....	16
3.13 Projektmitglieder .....	16
3.14 Rollenverteilung .....	17
3.15 Risiken .....	19
<b>4 Projektplanung .....</b>	22
4.1 Vorgehensmodell .....	23
4.2 Kommunikationsplan .....	24
4.3 Workshopplan .....	25
4.4 Begründeter Lösungsansatz .....	26
4.5 Ressourceneinsatzplan .....	30
4.6 Projektstrukturplan .....	31
4.7 Aufwandschätzverfahren .....	32
4.8 RACI .....	32
4.9 Arbeitspakete Initialisierung .....	34
4.10 Arbeitspakete Vorstudie .....	37
4.11 Arbeitspakete Konzeption .....	41
4.12 Arbeitspakete Realisierung .....	48
4.13 Arbeitspakete Abschluss .....	53
4.14 Meilensteine .....	56
4.15 Ganttdiagramm .....	57

<b>4.16 Projektnetzplan .....</b>	<b>59</b>
<b>5 Projektüberwachung .....</b>	<b>60</b>
5.1 Abweichungen analysieren und korrigieren .....	60
5.2 Kontrollprotokoll Arbeitspakete .....	60
5.3 Kontrollprotokoll User Stories .....	62
5.4 Messbare Lieferobjekte .....	62
<b>6 Analysedokumente .....</b>	<b>63</b>
6.1 Systemidee .....	63
6.2 Use Case .....	64
6.3 Mockup und Aufbau der Webseite .....	67
6.4 User-Stories .....	80
6.5 Fachklassenmodell (Domänenmodell) .....	92
<b>7 Softwareentwurfsdokumente .....</b>	<b>93</b>
7.1 Installation und Hardware .....	93
7.2 Kontextabgrenzung .....	93
7.3 Bausteinsichten der Softwarearchitektur .....	93
7.4 Ebene 2 – White Box Ansicht .....	94
7.5 Datenmodell .....	99
7.6 Klassenmodelle .....	101
7.7 Schnittstellenbeschreibung .....	117
7.8 Dynamische Modelle .....	125
7.9 Softwarearchitektur Frontend .....	131
<b>8 Testkonzept .....</b>	<b>134</b>
8.1 Testziele .....	135
8.2 Teststrategie und Teststufen .....	136
8.3 Testabdeckung .....	136
8.4 Testrahmen .....	136
8.5 Testumgebung .....	138
8.6 Testinfrastruktur .....	138
8.7 Testorganisation .....	138
8.8 Testfallbeschreibungen .....	138
8.9 Testplan .....	157
8.10 Testprotokoll .....	157
8.11 Testabdeckung - Report .....	159



<b>9    Entwicklungsdokumente .....</b>	161
9.1 <i>Entwicklungsprozess .....</i>	161
9.2 <i>Technologien und Tools .....</i>	163
9.3 <i>Entwicklungsrichtlinien.....</i>	164
<b>10    Inbetriebnahme .....</b>	165
10.1 <i>Lieferobjekte .....</i>	165
10.2 <i>Installationsanleitung.....</i>	165
10.3 <i>Benutzerhandbuch .....</i>	167
<b>11    Abschlussbericht .....</b>	170
11.1 <i>Aufgetretene Probleme .....</i>	170
11.2 <i>Auswertung Planung .....</i>	170
11.3 <i>Lessons Learned .....</i>	171
<b>12    Anhang .....</b>	172
12.1 <i>Bekannte Fehler.....</i>	172
12.2 <i>Literaturverzeichnis .....</i>	172
12.3 <i>Abbildungsverzeichnis .....</i>	174
12.4 <i>Tabellenverzeichnis.....</i>	175
<b>13    Eigenständigkeitserklärung .....</b>	180

## GLOSSAR

Tabelle 1: Glossarverzeichnis. Quelle: Autor

Nr.	Begriff	Beschreibung
1	Pair Programming	Paarprogrammierung: Erarbeitung eines Softwarecodes durch zwei Programmierer. Hierbei ist eine strikte Rollenverteilung vorgesehen (Ehrlich, 2019).
2	GIT	Software zur Versionsverwaltung von Softwarequellcodes (Stückler, 2020).
3	Repository	Verzeichnis/Projekt des Softwarequellcodes (Stückler, 2020).
4	Branch	Version der Software. Der Master-Branch ist die Hauptversion der Software. Auf Basis des Master-Branch können «Abspaltungen» mittels neuer Branches erfolgen, um neue Versionen dieser Software zu erstellen (Stückler, 2020).
5	Issue	Ein Element in GitLab welches zur Planung und Verwaltung von Anforderungen, Fehlern (Bugs), Refactoring und Architektur-Anpassungen verwendet wird.
6	Responsive	Bedeutet, dass sich die Oberfläche (UI) an das Endgerät, primär der Auflösung, anpasst.
7	HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard
8	UI/GUI	User Interface/Graphical User Interface, Benutzeroberfläche
9	API	Application Programming Interface, ist eine Schnittstelle zwischen Server und Client.
10	UML	Unified Modelling Language ist eine grafische modellierungsform in der Softwareentwicklung.
11	DOM	Document Object Model, bezeichnet das HTML-Grundgerüst.
12	HTML	Hyper Text Markup Language, bezeichnet die Auszeichnungssprache für den Browser.
13	IDE	Integrated Development Environment, ist eine Software für die Erstellung von Anwendungen (Red Hat Limited, 2019)
14	UUID	128 Bit grosse, standardisierte Identifikationsnummer ( $2^{128}$ ).
15	Join (Datenbank)	Join = Verbund. Im Umfeld von Datenbanken wird dieser Begriff verwendet um das Verbinden von zwei (oder mehreren) Tabellen, welche miteinander in Relation (Beziehung) zu beschreiben.
16	Normalisierung (Datenbank)	Bei relationalen Datenbanken werden die Daten oftmals soweit zerlegt (normalisiert), dass Informationen nicht mehrfach gespeichert werden müssen.
17	Proprietär (Lizenz)	Proprietäre (Software, Systeme usw.) sind in deren Nutzung resp. in der Wieder- und Weiterverwendung eingeschränkt. So dürfen proprietäre Komponenten nicht durch den Entwickler angepasst und weiterverwendet werden.
18	ISTQB	International Software Testing Qualifications Board ist eine international anerkannte Organisation, die für Software Qualität Zertifizierungen ausstellt.

<b>19</b>	Mock	Ein Mock-Objekt ist ein Platzhalter in der Softwareentwicklung. Es ist eine Attrappe für noch nicht realisierte Objekte, die frühzeitige Modultests ermöglicht. Mittels Mocken werden im Testfall Abhängigkeiten vorgetäuscht, deren eigentliche Implementierung wir bei der Testausführung selbst kontrollieren können (Robert, 2013).
<b>20</b>	Mockup	Ein Mockup ist eine Skizze eines Benutzer- oder einer Bedienoberfläche. Sie hat aber nichts mit einem Mock (Position 19 im Glossar) zu tun. Sie muss nicht zu 100% umgesetzt werden aber dient lediglich, wie die Benutzeroberfläche aussehen kann.
<b>21</b>	POC	Proof of concept / Konzeptioneller Beweis eine Vermutung oder einer Theorie. Mittels POC wird eine Theorie (beispielsweise eine Implementierungsidee) mittels tatsächlicher Implementation auf einer Testplattform geprüft und getestet.
<b>22</b>	ECTS	European Credits Transfer System, Punktesystem von Europäischen Hochschulen

#### TERMINOLOGIE

Die nachfolgende Dokumentation verwendet den Begriff «Nutzer», «Nutzende», «Anwendende» als Synonym für die Person, welche die Software benutzt. Damit ist der/die Studierende Person adressiert.

Module des BSc Informatik-Studiengangs werden zwischen Vertiefungs-, Wahl- und Pflicht-Module unterschieden. In diesem Dokument werden dies einfacheheitshalber vermehrt als «Module» zusammengefasst.

Es werden keine spezifischen Geschlechter angesprochen in diesem Dokument. Mit einer Bezeichnung ob männlich oder weiblich sind aus Sicht der Autoren immer alle Geschlechter gleichermassen angesprochen.



## 1 Management Summary

An der Fernfachhochschule Schweiz dürfen die Studierenden des Studiengangs Bsc Informatik jedes Jahr, ab dem 6. Semester, eine Modulplanung vornehmen. Eingeplant werden, müssen jeweils ein Vertiefungs- und ein Wahlmodul oder zwei Wahlmodule. Diese Module ermöglichen verschiedene Relevanzpunkte in verschiedenen Vertiefungsrichtungen. Bis zum Ende des Studiums müssen die Studierenden in einer Vertiefungsrichtung mindestens 8 Relevanzpunkte erreichen.

Die Planung findet auf Basis von verschiedenen Eingabe- und Übersichtsmasken statt. Um diese Aufgabe zu erleichtern, hat das Projektteam eine Lösung, in Form einer Software, für Studierende der HF Passarelle, erarbeitet. Die Software erlaubt es, die Modulplanung mittels verschiedener Auswahloptionen grafisch auf einer Weboberfläche zu tätigen. Die Software wurde im Rahmen des Moduls "Projektarbeit PA\_5" innerhalb von rund 450 Personenstunden erarbeitet.

Die Software ermöglicht eine visuelle Planung, die einige Validierungen bereits selbstständig übernimmt. Beim Design wurde darauf geachtet, dass weitere Validierungsmöglichkeiten eingebaut oder erweitert werden können. Dadurch bietet das entwickelte System eine gute Lösung zur betrachteten Problemstellung der Modulplanung von Studierenden für den Studiengang Bsc Informatik an der FFHS.

Das Projektteam erachtet aufgrund der Erreichung der definierten Ziele das Projekt als erfolgreich.



## 2 Einleitung

Das Studium zum BSc Informatik an der Fernfachhochschule Schweiz, umfasst 180 Leistungspunkte (ECTS-Credits). In den ersten fünf Semestern werden Grundlagen in der Informatik vermitteln, die nachfolgenden vier Semester dienen der individuellen Vertiefung nach Wahl (FFHS, BSc Informatik, kein Datum). Die Studierenden müssen auf Basis einer Auswahlliste von Modulen die Wahl- und Vertiefungs-Module planen. Nebst dieser Wahl- und Vertiefungs-Module werden Pflicht-Module absolviert. Es werden einige Bedingungen und Voraussetzungen an die Planung der Wahl- und Vertiefungs-Module gestellt. So müssen die Studierenden sich für eine Vertiefungsrichtung entscheiden und in dieser Spezialisierung mindestens 8 Relevanzpunkte erarbeiten. Die Anzahl zu erhaltenen Relevanzpunkte unterscheiden sich je nach Modul; so ermöglicht nicht jedes Modul den Erhalt von Relevanzpunkten oder/und die Punkte können in verschiedenen Vertiefungsrichtungen erlangt werden. Ein weiterer Aspekt, der für die Studierenden zu beachten ist, dass die Module einem Zeitplan unterliegen: die Module stehen an verschiedenen Standorten zu verschiedenen Zeitpunkten zur Verfügung.

## 3 Projektauftrag

### 3.1 Ausgangslage (Ist-Situation)

Die aus der Einleitung beschriebene Planung der Wahl- und Vertiefungsmodule findet aktuell (Stand: August 2021) auf der Lernplattform der FFHS, Moodle, statt. Die Planung ist Aufgabe des Studierenden und muss jedes Jahr wiederholt durchgeführt werden. Zur Planung stehen diverse Ansichten/Listen zur Verfügung:

- Übersicht der Relevanzpunkte (welches Modul ergibt wie viele Relevanzpunkte in welcher Vertiefung)
- Beschreibung der Vertiefungs- und Wahlmodule
- Zeitplan der Vertiefungs- und Wahlmodule (welches Modul wird wann an welchem Standort durchgeführt)

Die Planung wird durch die diversen Ansichten/Listen erheblich erschwert und ist aus Sicht des Projektteams nicht komfortabel. Des Weiteren ist es überaus umständlich einzusehen, welche Module wann eingeplant wurden – eine Übersicht der bereits getätigten Planung existiert unseres Wissens in Moodle nicht. Bei Abklärungen mit unserem Studiengangsleiter Herrn Oliver Itting stellt sich heraus, dass für Studierende ein Curriculum erstellt wird. In diesem Curriculum sind dann die jeweiligen geplanten Module ersichtlich. Diese Auflistung, im Excel-Format, bietet eine erheblich bessere Übersicht zur Planung des Studiums, ist jedoch auch nicht vollständig. So fehlen in diesem Curriculum die Informationen zu den Relevanzpunkten und zu möglichen Konflikten. Abhängigkeiten und mögliche Durchführungstermine sind im Curriculum nicht ersichtlich.

Falls Studierende das Studium mit weniger als 4 oder 5 Module durchführen möchten, so führt dies zu einem verlängerten Studium. Diese Option wird in der Modulplanung (Moodle) nicht angezeigt und benötigt eine Rücksprache mit Herrn Oliver Itting um dies im Curriculum einzutragen.

Das Projektteam hat sich deshalb die Aufgabe gestellt, die Studiums Planung für Studierende der Fernfachhochschule Schweiz in Form einer Applikation neu zu gestalten. Da die Planung dieser Module auch das Projektteam betrifft, besteht ein hoher Bedarf und eine grosse Motivation sich dieser Herausforderung zu stellen und einen alternativen Lösungsansatz zur IST-Situation aufzuzeigen und in Form einer Applikation umzusetzen.

## 3.2 Projektidee und Zielpublikum

Auf Basis der Problemstellungen aus der Ausgangslage,

- Planung der Wahl- und Vertiefungsmodule auf Basis von mehreren Ansichten/Listen ist unübersichtlich und aufwendig.
- Übersicht der geplanten Module fehlt (bereits geplante Module muss jeder Studierende für sich in einer separaten Liste führen).
- Die Modulplanung, in Moodle, zeigt nicht alle möglichen Optionen (Studiums Verlängerung) an.

strebt das Projektteam eine Modulplanung für die BSc Informatik Studierenden, angelehnt an das Curriculum der FFHS, mittels einer eigens entwickelten Software an. Der Studienplaner wird von Studierenden der FFHS, welche das verkürzte Studium über die HF Passerelle absolvieren, erstellt. Recherchen, wie andere Bachelor Lehrgänge geplant werden, sind aus Zeitgründen nicht möglich. Der Studienplaner richtet sich deshalb an BSc Informatik Studenten, welche das Studium in einer verkürzten Dauer (HF Passerelle (FFHS, Passerelle Informatik, kein Datum)) absolvieren. Andere Studienmodelle (Start im Erstsemester, Praxisintegriertes Bachelor-Studium usw.) werden jedoch im Software-Design als mögliche Ausbaumöglichkeiten berücksichtigt.

Individueller Studienplan - BSc Informatik							Jahrg. 2018	
		Module Grundstudium				Spezialbelegung	Erworben	Zu belegen
Semester					Pauschale Leistungsübernahme			
1	HS 2018 / 19						73 ECTS	0 ECTS
2	FS 2019						0 ECTS	0 ECTS
3	HS 2019 / 20						0 ECTS	0 ECTS
4	FS 2020						0 ECTS	0 ECTS
5	HS 2020 / 21	DMathLS 5 ECTS	EDS 5 ECTS	ISich 5 ECTS	D&A 5 ECTS	SHWB1 1 ECTS		21 ECTS
6	FS 2021	MITSec 5 ECTS	ESA 5 ECTS	AnPy 5 ECTS	WS 1 PV	SHWB2 1 ECTS		16 ECTS
<b>Summe Grundstudium</b>							<b>110 ECTS</b>	<b>0 ECTS</b>

Abbildung 1: Ausschnitt des Curriculums der FFHS. Quelle: Autor

### 3.3 Projektziele

Nachfolgend beschreibt das Projektteam, gemäss (Hruschka, 2014, S. 33), die wichtigsten 4 Ziele mittels PAM (Purpose, Advantage, Measure) (Hruschka, 2014, S. 32) und ordnet diese nach Muss/Kann ein:

Tabelle 2: Projektziele. Quelle: Autor

ID	Ziel	Muss/Kann
Z1	<p><b>P:</b> Das System muss dem Nutzer (den Studierenden) die Möglichkeit bieten eine Modulplanung über alle zu absolvierenden Semester auf Basis der gewählten Vertiefungsrichtung zu erstellen und dabei die anrechenbaren Relevanz- und ECTS-Punkte anzeigen.</p> <p><b>A:</b> Dadurch erhalten die Studierenden eine Übersicht zur Planung der Module, welche alle Informationen zentral darstellt und einen Überblick über das Studium bietet. Das Zusammensuchen der Informationen über mehrere Listen/Ansichten ist nicht mehr notwendig.</p> <p><b>M:</b> Das Ziel ist erfüllt, wenn die Modulplanung über eine Software-Oberfläche (UI) zur Verfügung gestellt wird und folgende Möglichkeiten bietet:</p> <ul style="list-style-type: none"> <li>- Auswahl der Vertiefungsrichtung.</li> <li>- Module (welche für diese Vertiefungsrichtung möglich sind) können in die Maske eingetragen und wieder entfernt werden.</li> <li>- Die Anzahl Relevanz- und ECTS-Punkte werden auf Basis der Planung errechnet und angezeigt.</li> </ul>	Muss
Z2	<p><b>P:</b> Das System muss bereits eingeplante Module nicht mehr zur Auswahl anbieten können.</p> <p><b>A:</b> Durch die Validierung des Systems wird der Nutzer geführt und Fehlerquellen werden vermieden.</p> <p><b>M:</b> Das Ziel ist erfüllt, wenn die Modulplanung folgende Eigenschaften bietet:</p> <ul style="list-style-type: none"> <li>- Bereits geplante Module können nicht nochmals eingeplant werden.</li> </ul>	Muss
Z3	<p><b>P:</b> Das System kann die Möglichkeit bieten, ein zusätzliches Semester hinzuzufügen, falls der/die Studierende weniger Module pro Semester besuchen und somit das Studium verlängern möchte. Durch das Hinzufügen eines weiteren Semesters soll der Nutzer die Möglichkeit erhalten, Pflichtmodule in einem anderen Semester einzuplanen.</p> <p><b>A:</b> Durch Erfüllung dieses Ziels ist ein gröserer Individualisierungsgrad für den Anwender der Software möglich.</p> <p><b>M:</b> Das Ziel ist erfüllt, wenn es in der Planung die Möglichkeit bietet ein weiteres Semester hinzuzufügen und weitere Module eingeplant werden können resp. bereits eingeplante Module verschoben werden können.</p>	Kann
Z4	<p><b>P:</b> Die Studienplanung ist auf Basis des Studienmodells (HF Paserelle, Start im Erstsemester, Praxisintegriertes Bachelor-Studium usw.) möglich. Dabei wird die jeweilige Semester-Anzahl berücksichtigt, die Pflichtmodule eingetragen und die Planung der Wahl- und Vertiefungsmodule auf Basis des Studienmodells möglich.</p> <p><b>A:</b> Durch die Erfüllung dieses Ziels kann der Studienplaner von allen Bsc Informatik Studierenden verwendet werden.</p> <p><b>M:</b> Die Zielerreichung wird an folgenden Eigenschaften gemessen:</p> <ul style="list-style-type: none"> <li>- Das Studienmodell ist auswählbar.</li> <li>- Die Anzahl Semester wird auf Basis des Studienmodells errechnet und dem Nutzer zur Planung angezeigt.</li> <li>- Die individuellen Pflicht-Module werden, abhängig vom Studienmodell, bereits in der Planung angezeigt.</li> <li>- Die Planung kann gemäss Z1 durchgeführt werden.</li> </ul>	Kann

### 3.4 Mindestanforderung

Die Mindestanforderungen sind die Projektziele, welche als MUSS eingestuft wurden.

Die Software für die Modulplanung soll bereits zum Einstieg verständlich gehalten werden. Dies setzt voraus, dass die Oberfläche einfach, sowie übersichtlich gestaltet ist und Effekte (Drag- und Drop, Slide-Ins usw.) nur gezielt einsetzt. Die Oberfläche soll nicht mit vielen Eingabe- und Auswahl-Möglichkeiten überladen sein. Der Fokus der Software liegt auf der Planung der Module und Berechnung der ECTS- und Relevanz-Punkte auf Basis der ausgewählten Vertiefungsrichtung. Weitere Aspekte wie die Validierung (Modul ist bereits eingeplant, Modul steht für das Semester nicht zur Verfügung usw.) sind zwar für den Nutzenden komfortabel und erhöhen den Mehrwert der Software, gehören jedoch nicht zum Haupt-Fokus der zu erarbeitenden Lösung und werden deshalb als Kann-Ziele definiert. Da das Projektbudget sehr begrenzt und der zur Verfügung stehende Entwicklungszeitraum sehr klein ist, werden solche Nice-To-Have-Features nur bei genügend freien Ressourcen umgesetzt.

### 3.5 Grobanforderungen

Tabelle 3: Grobanforderungen, Quelle: Autor

Titel	Beschreibung
<b>Schichtenarchitektur</b>	<ul style="list-style-type: none"> <li>- Die Software soll nach der 3-Schichten-Architektur entworfen werden: Benutzerschnittstelle (Client, UI), Backend (Server) und Datenbank.</li> </ul>
<b>Benutzeroberfläche</b>	<ul style="list-style-type: none"> <li>- Die Software soll über den Browser aufrufbar sein, damit keine Installation für Endanwender notwendig ist und von jedem Gerät (mit Browser) aufgerufen werden kann.</li> <li>- Die Oberfläche soll entworfen werden, dass diese auf iOS- und Android-Geräten angezeigt werden kann.</li> <li>- Die Benutzeroberfläche kann gemäss genütztem Framework entworfen werden.</li> <li>- Die Benutzeroberfläche erlaubt es, geänderte Daten ohne Neu laden des UI zu aktualisieren.</li> </ul>
<b>Datenhaltung</b>	<ul style="list-style-type: none"> <li>- Die Daten können in einer Datenbank persistiert werden.</li> <li>- Dabei ist die Datenbank, gemäss Schichtenarchitektur abstrahiert als eine eigene Schicht aufgebaut.</li> </ul>
<b>Wartbarkeit und Ausbaufähig</b>	<ul style="list-style-type: none"> <li>- Wartbarkeit und Ausbaufähigkeit durch Modularen Aufbau ermöglichen</li> <li>- Das Hinzufügen weiterer Sprachen für die Benutzeroberfläche muss in der Software unterstützt/berücksichtigt werden.</li> </ul>

### 3.6 Abgrenzung

Nachfolgende Abgrenzungen zeigen auf, was in diesem Projekt nicht umgesetzt werden kann:

*Tabelle 4: Abgrenzungen. Quelle: Autor*

Titel	Beschreibung
<b>Plattform</b>	Die Software wird für den Chrome Browser (jegliche Plattform) entworfen. Jegliche anderen Browser können aus Zeitgründen nicht vollständig geprüft werden.
<b>Authentifizierungsmöglichkeiten für Nutzer</b>	Auf Grund des geringen Projektbudgets (Zeit) wird keine Authentifizierungsmöglichkeit (Login) für Nutzer angeboten. Eine Anbindung an den Switch AAI Dienst (Switch, 2021), welcher von der FFHS eingesetzt wird, kann nicht betrachtet werden.
<b>Integration in das FFHS System</b>	Die Software wird als eigenständige Lösung entworfen. Eine direkte Integration in das FFHS Systemumfeld (Beispielsweise in Moodle oder in SharePoint) ist im Rahmen des Projektes nicht vorgesehen.
<b>Sprachunterstützung</b>	Die Standard-Sprache der Benutzeroberfläche ist Deutsch. Jegliche anderen Sprachen werden aus Zeitgründen nicht angeboten.
<b>Datenumfang</b>	Es wird nur der BSc Informatik Studiengang, im Rahmen der Möglichkeiten und Zugänglichkeit der Daten, abgebildet. Die Abbildung aller möglichen Studiengänge, aller existierenden Module und die vollständige Korrektheit aller Daten ist kein Projektziel.



## 3.7 Rahmenbedingungen / Restriktionen / Technische Ressourcen

### Organisatorische Rahmenbedingungen:

- Projektstart: 21.08.2021
- Projektende: 23.12.2021
- Jedes Projektmitglied muss während der Projektarbeit jede definierte Rolle einnehmen.
- Pro Projektmitglied sind 150-Arbeitsstunden vorgesehen.

Der Projektzeitraum bemisst sich auf 17 Wochen und 5 Tage. Dies bedeutet einen Aufwand von 8.4h pro Woche/pro Teammitglied (150/17.71).

### Formalitäten:

- Das Projekt muss schriftlich (Druckversion + Datenträger per A-Post) bis spätestens am 23.12.2021 anhand Poststempel eingereicht werden.
- Es finden 2 Projektstatus-Meetings statt.

### Folgende technischen Vorgaben gelten gemäss Vorgabe der FFHS für das Projekt:

- Java muss als Haupt-Programmiersprache eingesetzt werden. (51%)
- Es dürfen weitere Programmiersprachen verwendet werden. (49%)
- Der Java-Quellcode ist mittels JavaDoc zu dokumentieren.
- Das Objektorientierte-Programmierparadigma (Richtlinien sind definiert).
- Es sind JUnit-Tests zu erstellen und mittels Testprotokollen zu dokumentieren (Umfang, Vollständigkeit ist nicht definiert).

Die vollständige Auflistung der Vorgaben ist auf der Projektseite zu finden (FFHS, Vorgaben zur Projektarbeit, 2021).

Für die Projektarbeit ist die Nutzung eines MacOS-, Linux- oder Windows-Betriebssystem erforderlich. Die Hardware (Desktop-PC oder Laptop) sollte genügend performant ausgelegt sein, um den Betrieb von mehreren Entwicklungstools (IDEs), mehrere Browser-Instanzen und Betrieb einer lokalen Datenbankinstanz gleichzeitig zu ermöglichen. Es ist schwierig hierzu konkrete Anforderungen zu definieren; Hardware ab ca. 2019 sollte jedoch diesen Anforderungen gerecht werden. Des Weiteren muss die Installation der notwendigen Frameworks/SDKs möglich sein. Dies bedeutet, dass das Projektmitglied Administratoren-Berechtigung auf dem Arbeitsgerät besitzt.

Für die Entwicklung der Software werden verschiedene SDKs und Frameworks verwendet. Da zur Java-Version keine Vorgabe existiert, wird die Software auf Basis einer, durch das Projektteam definierten, Java-Version entwickelt. Dabei kann keine Abwärts-/Aufwärts-Kompatibilität zu anderen Java-Versionen garantiert werden.

## 3.8 Eingesetzte Ressourcen

Nachfolgende Tools und Software werden für die Erstellung und Modellierung der Software sowie Dokumentation des Projekts verwendet:

- Discord
- Miro-Board
- Gitlab
- Microsoft Software (Office365, Teams)
- IntelliJ IDEA (IDE für Java-Entwicklung [Backend])
- Visual Studio Code (IDE für Frontend-Entwicklung)
- Chrome Webbrowser (Testen und Debuggen des UI)
- Enterprise Architect (Erstellung von UML, Use Case usw.)
- Draw.io (Sonstige Grafiken)

## 3.9 Projektzeitraum

Das Projekt findet nach Vorgaben der FFHS statt.

Zeitlicher Ablauf:

- |                          |            |
|--------------------------|------------|
| 1. Kick-off-Meeting      | 21.08.2021 |
| 2. Status-Meeting 1      | 18.09.2021 |
| 3. Status-Meeting 2      | 13.11.2021 |
| 4. Versand Dokumentation | 23.12.2021 |
| 5. Schlussprüfung        | 13.01.2021 |

## 3.10 Kostenaspekt

Die Gesamtkosten, des Projekts, belaufen sich auf 19'402.60 CHF. Die einzelnen Kosten werden in den nachfolgenden Kapiteln aufgeführt.

Da der Personalaufwand, auch Ressourcen darstellen – da die investierte Zeit anderweitig genutzt werden könnte – werden diese auch aufgeführt (Dr. Angermeier, 2014)

### 3.10.1 Personalressourcen

Der Zeitliche Aufwand wird auf 5 ECTS (European Credit Transfer System) pro Projektmitglied geschätzt. Dies entspricht grob 150 Stunden Arbeitsaufwand pro Projektmitglied, womit der Gesamtaufwand mit 450 Stunden budgetiert wird.

Für die Berechnung der Personalkosten wird der Bruttolohn von 7'281 CHF (für die Entwicklung) und 7'660 CHF (für die Projektarbeit) als Berechnungsgrundlage verwendet. Die Datengrundlage basiert auf dem Lohnrechner der Schweizerischen Eidgenossenschaft (Admin, kein Datum) mit folgenden Eingaben: Bern (Region), Dienstleistungen der Informationstechnologie (Branche), Akademische u. vergleichbare Fachkräfte (Berufsgruppe) ohne Kaderfunktion bei 42 Sollstunden pro Woche. Das Alter wurde gemittelt (32 Jahre) und 0 Dienstjahre verwendet. Dies ergibt einen Stundenansatz von 43.30 CHF für die Softwareentwicklung (7'281/168) und 45.50 CHF für die Projektarbeit (7'660/168). In nachfolgender Tabelle werden die gesamten Personalkosten für dieses Projekt berechnet:

Tabelle 5: Projektkosten - Personalressourcen. Quelle: Autor

Tätigkeit	Stunden	Stundenansatz	Total
<b>Projektarbeit</b>	300	45.50	12'650 CHF
<b>Software-Design und -Entwicklung</b>	150	43.30	6'495 CHF
		<b>Total</b>	<b>19'145 CHF</b>

### 3.11 Grundlagen

Als Basis wird das FFHS Curriculum verwendet und alle zur Verfügung stehenden Informationen zu den angebotenen Modulen der Passerelle Informatik. Zur Softwarelösung liegen keine uns bekannten Vorarbeiten vor. Erst mit dem Projektstart wurden die ersten Arbeiten ausgeführt.

### 3.12 Projektphasen



Abbildung 2: Projektphasen, Quelle: Autor

**Initialisierung:** Projektziele definieren und Grobplanung der nötigen Meilensteine, die zur Erfüllung dieser Ziele nötig sind.

**Vorstudie:** Aufstellung der Projektorganisation, Projektauftrag definieren und Erstellen eines Grobplan zur Umsetzung.

**Konzeption:** Fertigstellung des Konzepts, Erarbeitung von Lösungsansätzen sowie Bewertung. Lösungen detaillierter ausarbeiten.

**Realisierung:** Umsetzung Projektziele gemäss Planung, Controlling und Abweichungen dokumentieren.

**Abschluss:** Abschlussbericht erstellen und Dokumentation ergänzen/abschliessen. Beurteilung und Lessons Learned dokumentieren.

### 3.13 Projektmitglieder

Nachfolgende Personen sind am Projekt beteiligt:

Tabelle 6: Projektmitglieder, Quelle: Autor

Vorname Nachname	Studiengang	Beruflicher Background
<b>Christoph Grossmann</b>	BSc Informatik – HF Passerelle	Seit 2019 als Application Manager (Requirements Engineering und Support). HF Studium Informatik 2016 - 2019.
<b>Marco Ris</b>	BSc Informatik – HF Passerelle	2016 – 2018 Junior Webdeveloper, seit 2018 Softwareentwickler (Fullstack, PHP, MySQL, JavaScript) in KMU-Bereich tätig. HF Studium Informatik 2016 - 2019
<b>Peter Spielhofer</b>	BSc Informatik – HF Passerelle	Seit 2017 als Softwareentwickler (Fullstack, C#, Angular) in KMU-Bereich tätig. HF Studium Richtung Softwareentwicklung 2017 - 2020.

### 3.14 Rollenverteilung

Eingeplante Rollenverteilung durch den ganzen Projektverlauf:

Tabelle 7: Rollenverteilung, Quelle: Autor

Name	Rolle	Zeitraum	Projektpphase
<b>Christoph Grossmann</b>	Projektleiter	30.08. - 26.09.21	Initialisierung / Vorstudie
<b>Marco Ris</b>	Requirements Engineer	30.08. - 26.09.21	Initialisierung / Vorstudie
<b>Peter Spielhofer</b>	Requirements Engineer	30.08. - 26.09.21	Initialisierung / Vorstudie
<b>Christoph Grossmann</b>	Requirements Engineer / Softwarearchitekt	27.09. - 25.10.21	Konzeption
<b>Marco Ris</b>	Projektleiter	27.09. - 25.10.21	Konzeption
<b>Peter Spielhofer</b>	Softwarearchitekt / Software-Engineer	27.09. - 25.10.21	Konzeption
<b>Christoph Grossmann</b>	Projektleiter	26.10. - 21.11.21	Konzeption / Realisierung
<b>Marco Ris</b>	Softwarearchitekt / Software-Engineer	26.10. - 21.11.21	Konzeption / Realisierung
<b>Peter Spielhofer</b>	Software-Engineer / Tester	26.10. - 21.11.21	Konzeption / Realisierung
<b>Christoph Grossmann</b>	Software-Engineer / Tester	22.11. - 23.12.21	Realisierung / Abschluss
<b>Marco Ris</b>	Software-Engineer / Tester	22.11. - 23.12.21	Realisierung / Abschluss
<b>Peter Spielhofer</b>	Projektleiter	22.11. - 23.12.21	Realisierung / Abschluss

### 3.14.1 Projektorganisation

Projektleiter, Requirement Engineer, Softwarearchitekt, Software Engineer und Testmanager werden gemäss alternierend durch das Projektteam abgedeckt. Analogieverfahren. Bei diesem Verfahren wird auf bereits gemachte Erfahrungen zurückgegriffen. Vergleich der Vorgangsdauer (Ist-Werte) aus früheren Projekten mit ähnlichen Problemstellungen.

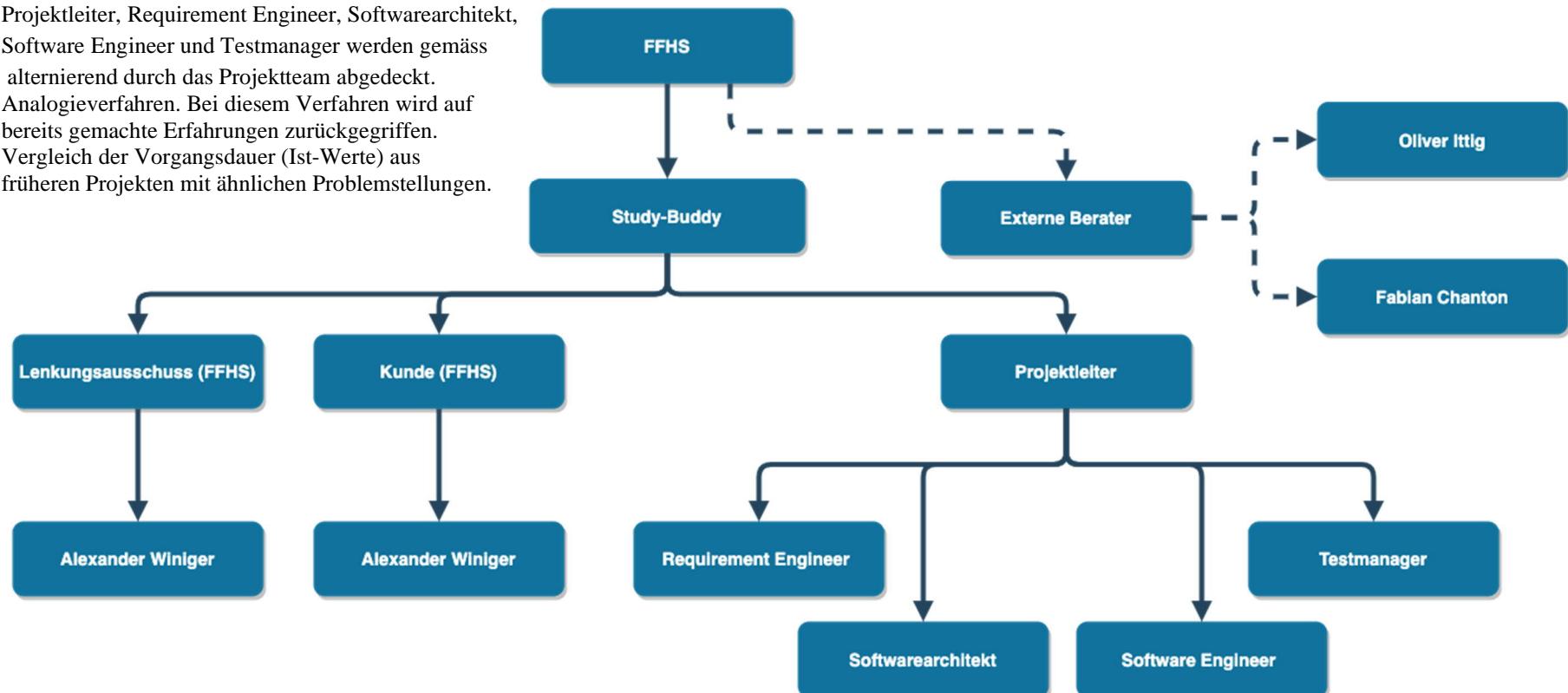


Abbildung 3: Projektorganisation. Quelle: Autor

## 3.15 Risiken

EW = Eintrittswahrscheinlichkeit (gering, mittel, gross) - AW = Auswirkungen (gering, mittel, gross)

Tabelle 8: Projektrisiken, Quelle: Autor

Nr.	Risiko	EW	AW	Beschreibung und Massnahme
<b>R1</b>	Unvorhergesehene Absenzen (Krankheit usw.)	Mittel	Mittel	Früher Start des Projekts, damit genügend zeitlicher Spielraum vorhanden ist. Pendenzen und Aufgaben nach Möglichkeit verteilen und/oder neu priorisieren.
<b>R2</b>	Nachlassendes Interesse der Projektmitglieder in das Projekt	Gering	Gross	Dieses Risiko muss beobachtet werden. Wenn ein nachlassendes Interesse festgestellt wird, müssen die Gründe und entsprechenden Massnahmen im persönlichen Gespräch eruiert werden.
<b>R3</b>	Technisches Risiko, dass durch die gewählten Technologien die Anforderungen nicht umsetzbar sind.	Mittel	Mittel	Allenfalls müssten Neuspezifizierungen im WIE (wie wird die Anforderung umgesetzt) durchgeführt werden (und nicht im WAS). Die Ziele müssen so definiert werden, dass diese mit den eingesetzten Technologien umsetzbar sind.
<b>R4</b>	Technisches Risiko, dass die Technologien während der Projektzeit Änderungen unterzogen wird.	Gross	Gering	Development tools, Programmiersprachen, Frameworks usw. werden ständig weiterentwickelt. Die eingesetzten Technologien erlauben es, die Version «einzufrieren» resp. fix zu definieren.
<b>R5</b>	Schleichende Erweiterung des Projektumfangs durch neue Anforderungen und Ideen	Gross	Gering	Oftmals tauchen während der Projektarbeit neue Ideen und Anforderungen auf. Dies ist ein normaler Vorgang, die Softwareentwicklung ein kreativer Prozess ist; Ideen entwickeln sich während der Umsetzung. Bei Eintritt: Neue Anforderungen abwägen (Umsetzungsmöglichkeit, Aufwand usw.), gegebenenfalls den Anforderungskatalog überarbeiten und den Auftraggeber informieren.
<b>R6</b>	Kommunikation: zu wenig oder zu viel Kommunikation	Mittel	Gering	Um das Mass der geeigneten Kommunikation (Mittel, Frequenz usw.) zu bestimmen, wird ein Kommunikationsplan erstellen. Mittels dieses Plans kann das Risiko minimiert werden. Das Risiko muss ständig überwacht werden (Ist die Kommunikation genügend? Sind die eingesetzten Mittel gut gewählt?) und allenfalls der Plan angepasst werden.
<b>R7</b>	Ungleiche Verteilung der Technischen und Projektorganisatorischen Fähigkeiten	Gross	Mittel	Eine Rahmenbedingung dieser Arbeit ist es, dass alle Projektmitglieder jede definierte Rolle einnehmen muss. Die verschiedenen beruflichen und schulischen Hintergründe der Projektmitglieder führt dazu, dass das Know-how in verschiedenen Disziplinen nicht umfassend vorhanden ist. Um dieses Risiko zu minimieren, werden zu einigen Themen Workshops gemacht. Themen sind: Umgang mit der Entwicklungsumgebung, Erstellen eines Unit-Tests usw. Die Workshop-Themen sind im Workshop-Plan aufgeführt.
<b>R8</b>	Realisierung dauert länger als geplant.	Mittel	Gross	Um dieses Risiko zu minimieren, wird ein Projektpuffer eingeplant. Mittels dieses Puffers kann auf eine länger dauernde Realisierungs-Phase reagiert werden.

### 3.15.1 Aktualisierung Risiken

Zum 2. Status-Meeting vom 13.11.2021 wurde eine Neubewertung der Risiken vorgenommen.

Tabelle 9: Projektrisiken: Quelle: Autor

Nr.	Risiko	EW	AW	Beschreibung und Massnahme
<b>R1</b>	Unvorhergesehene Absenzen (Krankheit usw.)	Mittel	Mittel	Früher Start des Projekts, damit genügend zeitlicher Spielraum vorhanden ist. Pendenzen und Aufgaben nach Möglichkeit verteilen oder/und neu priorisieren.
<b>R2</b>	Nachlassendes Interesse der Projektmitglieder in das Projekt	Gering	Gross	Dieses Risiko sollte beobachtet werden. Wenn ein nachlassendes Interesse festgestellt wird, müssen die Gründe und entsprechenden Massnahmen im persönlichen Gespräch eruiert werden.  Durch die regelmässigen Teammeetings und gelegentlichen «Teamallässe» konnte die Motivation gehalten werden. Persönliche Angelegenheiten wurden durch die anderen Projektmitglieder angefangen.
<b>R3</b>	Technisches Risiko, dass durch die gewählten Technologien die Anforderungen nicht umsetzbar sind.	Mittel	Mittel	Allenfalls müssten Neuspezifizierungen im WIE (wie wird die Anforderung umgesetzt) durchgeführt werden (und nicht im WAS). Die Ziele müssen so definiert werden, dass diese mit den eingesetzten Technologien umsetzbar sind.
<b>R4</b>	Technisches Risiko, dass die Technologien während der Projektzeit Änderungen unterzogen wird.	Gross	Gering	Entwicklungstools, Programmiersprachen, Frameworks usw. werden ständig weiterentwickelt. Die eingesetzten Technologien erlauben es, die Version «einzufrieren» resp. fix zu definieren.
<b>R5</b>	Schleichende Erweiterung des Projektumfangs durch neue Anforderungen und Ideen	Gering	Gering	Oftmals tauchen während der Projektarbeit neue Ideen und Anforderungen auf. Dies ist ein normaler Vorgang, da der Mensch niemals alle Kenntnisse über den vollständigen Umfang an Anforderungen zum Projektstart hat. Des Weiteren ist der Akt der Softwareentwicklung ein kreativer Prozess; Ideen entwickeln sich während der Umsetzung.  Bei Eintritt: Neue Anforderungen abwägen (Umsetzungsmöglichkeit, Aufwand usw.), gegebenenfalls den Anforderungskatalog überarbeiten und den Auftraggeber informieren.  Es wurden Use Cases definiert und User Stories abgeleitet. Es wurden keine weiteren Anforderungen aufgenommen die wie erwartet von aussen noch zugetragen werden.
<b>R6</b>	Kommunikation: zu wenig oder zu viel Kommunikation	Gering	Gering	Da die Datenstruktur, der einzelnen Planungen ohne festes Schema sind (jede Planung kann prinzipiell anders aussehen) und keine komplexe Abfragemöglichkeiten und Beziehungen zwischen den Daten benötigt werden.  Datenbankmodell und Schnittstellenbeschreibung wurde so erstellt, dass eine klare Definition vorhanden ist welche Daten, wo hinterlegt und zur Verfügung stehen werden müssen.
<b>R7</b>	Ungleiche Verteilung der Technischen und Projektorganisatorischen Fähigkeiten	Mittel	Mittel	Eine Rahmenbedingung dieser Arbeit ist es, dass alle Projektmitglieder jede definierte Rolle einnehmen muss. Die verschiedenen beruflichen und schulischen Hintergründe der Projektmitglieder führt dazu, dass das Know-How in verschiedenen Disziplinen nicht umfassend vorhanden ist.  Um dieses Risiko zu minimieren, werden zu einigen Themen Workshops gemacht. Themen sind: Umgang mit der Entwicklungsumgebung, Erstellen eines Unit-Tests usw. Die Workshop-Themen sind im Workshop-Plan aufgeführt.

<b>R8</b>	Realisierung dauert länger als geplant.	Gering	Gross	Um dieses Risiko zu minimieren, wird ein Projektpuffer eingeplant. Mittels dieses Puffers kann auf eine länger dauernde Realisierungs-Phase reagiert werden. <b>Es wurde eine Woche als Projektpuffer eingeplant.</b> <b>Geplanter Fertigstellungstermin ist der 18.12.</b>
-----------	---	--------	-------	---

### 3.15.2 Risikomatrix

Typische Projektrisiken wie zu optimistische Terminplanung, Probleme durch Einsatz neuer Technologien, falsche Priorisierungen usw. können mittels sorgfältiger Projekt- und Produktplanung entgegengewirkt werden und werden deshalb nicht explizit aufgeführt, obwohl diese in jedem Projekt vorhanden sind.

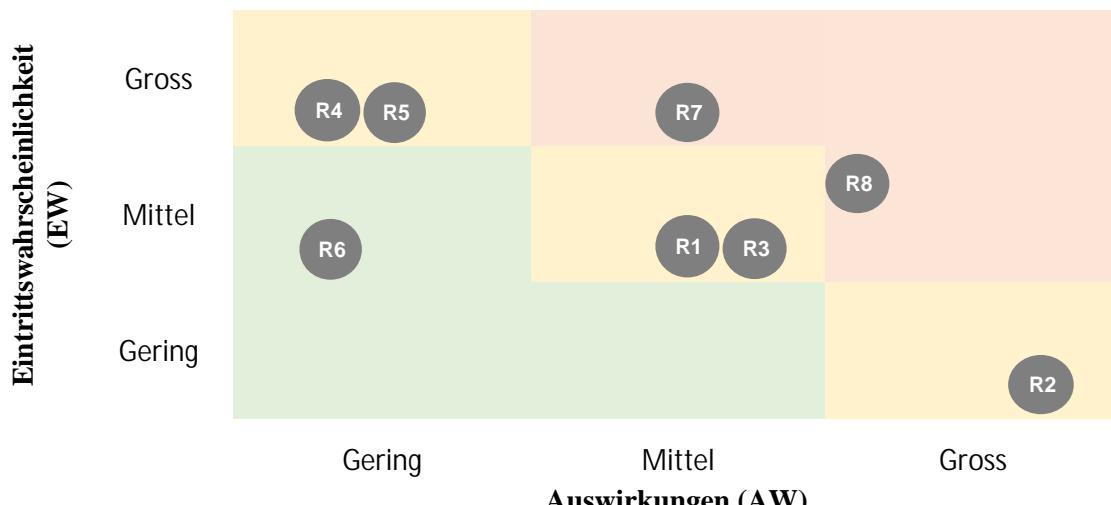


Abbildung 4: Risikomatrix, Quelle: Autor

### 3.15.3 Aktualisierung Risikomatrix

Zum 2. Status-Meeting vom 13.11.2021 wurde eine Neubewertung der Risiken vorgenommen.

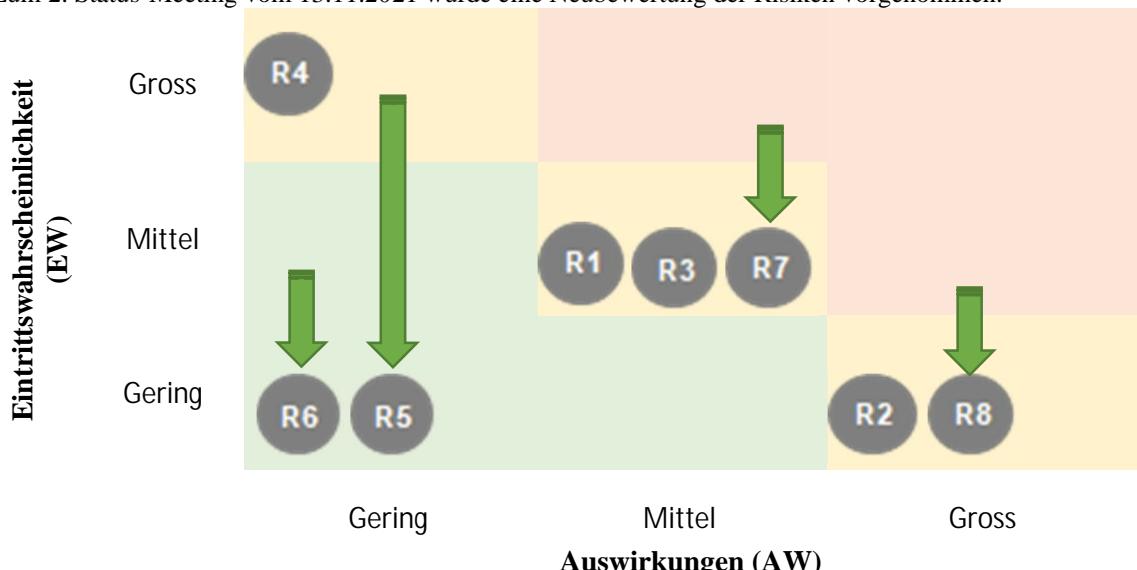


Abbildung 5: Aktualisierte Risikomatrix, Quelle: Autor

### 3.15.4 Problembewertung und -lösung

Die Projektrisiken, welche sich im roten Bereich befinden, müssen bis zum Konzeptions-Prozess mittels notwendiger Workshops in den gelben, bzw. grünen Bereich verschoben werden. Die Projektrisiken, welche sich im gelben Bereich befinden, können bis zum Konzeptions-Prozess bestehen, oder in den grünen Bereich verschieben.

Die Risikomatrix zeigt, dass die Risiken entsprechend angegangen werden müssen und dass sich das Projekt nach einer weiteren Risikoanalyse ab dem Konzeptions-Prozess möglicherweise realisieren lässt.

## 4 Projektplanung

Vom Groben ins Detail:



Abbildung 6: Projektplanung - Ablauf. Quelle: Autor

#### Vorgehen bei der Grobplanung:

- Projektphasen und Meilensteine festlegen,
- Entscheide festlegen, die bei den Meilensteinen zu treffen sind,
- Arbeitspakete definieren und abgrenzen: Arbeitspaketstruktur festlegen und die Verantwortung zuteilen.
- Das Resultat wird im Meilensteinplan und im Projektstrukturplan dokumentiert.

#### Vorgehen bei der Konzeption:

- Detailziele bzw. Ziele für Teilprojekte und -systeme festlegen und Ziele bei Bedarf überarbeiten
- Lösungsvarianten entwickeln und auf die Zielkonformität prüfen
- Lösungsvarianten bewerten und eine Variante vom Projektteam auswählen lassen
- Detaillierte Lösung der gewählten Variante ausarbeiten
- Mittel- und Ressourcenbedarf überprüfen und anpassen

#### Vorgehen bei der Detailplanung:

- Projektstrukturplan aktualisieren
- Die Tätigkeiten festlegen, die innerhalb der Arbeitspakete auszuführen sind
- Abhängigkeit der Tätigkeiten voneinander abklären
- Fähigkeiten und Erfahrungen bestimmen, die notwendig sind für die effiziente Abwicklung der Tätigkeiten, Verfügbarkeit der benötigten Spezialisten klären
- Zeitaufwand und Durchlaufzeit für Ressourenzuteilung schätzen
- Terminierung durchführen, falls nötig die Durchlaufzeit optimieren
- Verfügbarkeit der Ressourcen verbindlich absprechen
- Ressourcenkonflikte erkennen und verbindlich lösen
- Allfällige terminliche Konflikte lösen
- Die Planungsresultate allen Beteiligten und Betroffenen kommunizieren

## 4.1 Vorgehensmodell

In der Entwicklung von Software existieren einige reale Probleme, welche dazu führen, dass die Wahl des Projekt-Vorgehensmodell massgeblich zum Projekterfolg beisteuert. In der Praxis ist es oftmals unmöglich alle Anforderungen bereits zum Projektanfang vollständig festzuhalten und zu spezifizieren. Durch eine rein sequenzielle Vorgehensweise werden Design-Fehler und fehlende oder falsche Anforderungsspezifikationen oftmals erst zu spät entdeckt, womit der Projekterfolg gefährdet oder die Einhaltung des Liefertermins verhindert wird (Rau, 2007).

Auf Grund dieser Problemstellung wird das Wasserfall Vorgehensmodell gewählt wobei die Entwicklungsphase iterativ durchlaufen wird. Dies bedeutet, dass in kleinen Teilschritten kontinuierliche Verbesserungen erarbeitet werden. Konkret bedeutet dies, dass das Endprodukt (der Studienplaner) zu Projektbeginn mit allen geforderten Dokumenten (UML-Diagramme uvm.) beschrieben wird. Jedoch ist diese Beschreibung im Detailierungsgrad grob gehalten, da noch nicht alle Details zur konkreten Umsetzung bekannt sind. Jede Iteration in der Entwicklungsphase führt zu zunehmender Verbesserung und Verfeinerung der Software.

Um den Leistungsumfang festzulegen, müssen die Projektgrenzen, sowie die Projektergebnisse und Lieferobjekte definiert werden. Da ein Projekt die Anforderungen stetig ändern können, ist es wichtig diese Änderungen in einem Prozesse abzuhandeln und alle Änderungen resp. Anfragen zu dokumentieren, damit diese nachvollziehbar sind. Wie die Strategie festgelegt wird, hängt von den Rahmenbedingungen ab. Diese betreffen den vereinbarten Leistungsumfang, den Endtermin, den Kostenrahmen und die verfügbaren Ressourcen. Welche Strategie gewählt werden soll, hängt von der Fragestellung „Welche Planungsgröße ist fix und welche Ziele müssen unbedingt erreicht werden?“ ab.

Wir haben uns für die Termintreue Planung entschieden, da der Projektzeitraum vorgegeben, das Resultat jedoch abhängig von der zur Verfügung stehenden Zeit ist. Den Entwicklungsprozess werden wir im iterativen Verfahren durchführen und hierbei so viel wie möglich.

Tabelle 10: Planungsstrategie. Quelle: Autor

Planungsstrategie	Planungsgröße			Fokus der Anpassung
	Ergebnisse	Ressourcen	Endtermin	
<b>Kapazitätstreue Strategie</b> <i>Realisiere den gesamten Projektumfang so schnell wie möglich</i>	Fix	Fix	Variabel	Terminziele Anpassen
<b>Puffertreue Strategie</b> <i>Realisiere den gesamten Projektumfang mit so wenig Ressourcen wie möglich</i>	Fix	Variabel	Fix	Fehlende Ressourcen ergänzen/Zu viel Ressourcen zurückgeben
<b>Termintreue Planung</b> <i>Realisiere so viel wie möglich zum fixen Endtermin</i>	Variabel	Fix	Fix	Leistungsziele anpassen

## 4.2 Kommunikationsplan

PL = Projektleiter - PM = Projektmitglied

Tabelle 11: Kommunikationsplan, Quelle: Autor

Datum	Titel	Was	Medium	Frequenz
<b>21.08.21</b>	Kick off-Meeting für alle PM	- Ausgangssituation/Hintergründe - Gruppenfindung - Auftragsdefinition - Apero und gegenseitiges Kennenlernen	Persönlich	einmalig
<b>23.08.21</b>	Projektinitialisierungs-Meeting für PM	- Projektorganisation und -rollen - Kommunikationsstrukturen - Besprechung weiteres Vorgehen - Aufgabenverteilung	Remote (Teams)	einmalig
	Kaffee-Klatsch aller PM	- Fragen und Probleme welche nicht, bis Statusmeeting warten können - Kaffee-Klatsch (Allgemeines)	Discord	Täglich
<b>Mo 18:00 Uhr</b>	Statusmeeting für alle PM	- Berichte über aktuelle Arbeitspaketfortschritt - Erhebung laufender Aktivitäten und Check offener AP - Erkennen kritischer Planabweichungen - Ausblick und Klärungen offener Fragen	Remote (Teams, Miro)	Wöchentlich
<b>18.09.21, 13.11.21</b>	Projektstatusbericht an Alexander Winiger und Severin Kuepfer	- Verdichtete Informationen über Projektstand - Entscheidungen zu Themen, welche von der PL nicht allein getroffen werden können - Risiken und deren Massnahmen - Budget und Ressourcen-Abweichungen	Remote (Teams, PowerPoint)	Fixe Termine
<b>23.12.21</b>	Späteste Abgabe der Dokumentation per Postweg	- Gebundene Dokumentation - USB-Stick: <ul style="list-style-type: none"><li>○ Quellcode</li><li>○ Dokumentation digital</li><li>○ Lauffähiges System</li><li>○ Reports</li></ul>	Post	Fixer Termin
<b>13.01.22</b>	Schlussprüfungs-Bericht für Alexander Winiger und Serverin Kuepfer	- Projektziele und erreichte Ergebnisse - Lieferergebnisse aufzeigen - Ausblick und Klärungen offener Fragen und Aufgaben	Persönlich (PowerPoint)	Fixer Termin
<b>13.01.22</b>	Teamauflösung und Lessons Learned für PM	<ul style="list-style-type: none"><li>• Rückblick</li><li>• Danksagung an alle Beteiligten und Vorstellung des erzielten Erfolgs</li><li>• Teamauflösung</li><li>• Ausblick und Klärungen offener Fragen</li></ul>	Persönlich	Einmalig

## 4.3 Workshopplan

Wie im Risiko R7 beschrieben, ist ein grosser Unterschied in den verschiedenen Disziplinen vorhanden, die in dieser Projektarbeit behandelt werden müssen. Es ist primär die technische Disziplin, die den Projekterfolg gefährdet. Die Workshops werden im definierten Zeitraum im Rahmen der wöchentlichen Montagssitzung durchgeführt. Die Workshops werden von Peter Spielhofer geleitet und vorbereitet.

Tabelle 12: Workshop-Plan, Quelle: Autor

Thema	Beschreibung	Zeitraum	Wer
<b>Entwicklungs-umgebung</b>	<p>Voraussetzung: IDEs, Frameworks und SDK sind gemäss Vorgabe installiert!</p> <p>Inhalt:</p> <ul style="list-style-type: none"> <li>• Welche Einstellungen sind zu tätigen?</li> <li>• Wie werden die Artifakte gestartet?</li> <li>• Welche Plugins/Addons werden benötigt?</li> <li>• Wie ist die Entwicklungsumgebung einzurichten?</li> <li>• GIT Einrichten. Arbeiten mit Branches</li> <li>• JavaDoc Einführung und Erklärung</li> <li>• Vorbereitungen für den Docker-Betrieb der Datenbank</li> </ul> <p>Das notwendige Wissen wird von Peter Spielhofer vermittelt. Die aufzuwendende Zeit ist nicht Teil des Projektbudgets – die Lerninhalte werden in der Freizeit vermittelt.</p>	02.11.21	Alle
<b>Testing</b>	<p>Was wird getestet und wie werden Tests erstellt? Was ist ein guter/messbarer Test? Was sind Mocks und wie werden diese eingesetzt? Wie werden Tests ausgeführt?</p> <p>Das notwendige Wissen wird von Peter Spielhofer vermittelt. Die aufzuwendende Zeit ist nicht Teil des Projektbudgets – die Lerninhalte werden in der Freizeit vermittelt.</p>	02.11.21	Alle

## 4.4 Begründeter Lösungsansatz

Nachfolgendes Kapitel behandelt die Lösungsvariante. Dabei wurde nach dem Prinzip des Problemlösungszyklus vorgegangen: Zielsuche, Lösungssuche und anschliessender Variantenentscheid. Für den Variantenentscheid werden Muss- und Wunschkriterien erarbeitet. Die untersuchten Lösungsvarianten müssen die Musskriterien erfüllen. Danach wird auf Basis der Wunschkriterien eine Nutzwertanalyse durchgeführt, um die optimale Lösungsvariante zu eruieren.

### 4.4.1 Musskriterien

Basierend auf der Grobanforderung werden nachfolgende Musskriterien, welche für die Evaluierung der zu verwenden Lösungsvariante sind, abstrahiert:

*Tabelle 13: Musskriterien, Quelle: Autor*

Nummer	Bezeichnung
<b>MK1</b>	Das UI muss getrennt (abstrahiert) von den anderen Schichten umgesetzt werden können.
<b>MK2</b>	Das UI ist im Browser aufrufbar.
<b>MK3</b>	Das UI kann auf mobilen Endgeräten (iOS, Android) aufgerufen werden.
<b>MK4</b>	Die Technologie ermöglicht die Erstellung von Standard-Eingabe Elementen (Textfeld, Button usw.).
<b>MK5</b>	Das UI erlaubt es Visualisierung und Logik zu trennen.
<b>MK6</b>	Verschiedene Datenbankmodelle (dokumentorientiert, relational usw.) werden unterstützt.
<b>MK7</b>	Die Datenbank lässt sich losgelöst von den anderen Schichten implementieren.

### 4.4.2 Wunschkriterien

Basierend auf der Grobanforderung werden folgen nachfolgende Wunschkriterien, welche für die Evaluierung der zu verwenden Lösungsvariante sind, abstrahiert:

*Tabelle 14: Wunschkriterien, Quelle: Autor*

Nummer	Bezeichnung
<b>WK1</b>	Die eingesetzte Technologie ist im Projektteam bekannt und wurde bereits beruflich oder im schulischen Umfeld verwendet.
<b>WK2</b>	Die Lösungsvariante trägt dazu bei, die Entwicklungszeit zu mindern.
<b>WK3</b>	Die Technologie basiert auf aktuellen Standards und wird weitgehend verwendet.
<b>WK4</b>	Die Technologie ist vom Komplexitätsgrad innerhalb 4-5 Wochen lernbar.



#### 4.4.3 Lösungsansätze

Nachfolgende Lösungsansätze müssen die Musskriterien erfüllen.

Durch die Vorgabe der FFHS, dass Java als Hauptentwicklungssprache eingesetzt werden soll, wäre eine Umsetzung als reine Java-Applikation möglich. Eine Umsetzung, durch den Einsatz von verschiedenen Technologien (Technologie-Mix) wäre auch möglich. Hierbei muss die Vorgabe (Java) berücksichtigt werden. Diese Überlegungen führen zu nachfolgenden **Umsetzungsmöglichkeiten**:

- JavaFX-Applikation
- Android-Applikation
- Webapplikation mit Backend (Java) und Frontend (Angular, Vue.js, React usw.)
- Spring mit Template-Engine (Thymeleaf, Freemarker usw.)
- Java/Jakarta EE mit JSP

Bei der JavaFX-Applikation treten in Hinsicht auf die Bedienoberfläche einige Einschränkungen auf. Mittels jpro laufen JavaFX-Anwendung auch im Browser (Sandec GmbH, kein Datum). Des Weiteren existieren Portierungen, um JavaFX-Anwendung auf Android- und iOS-Endgeräten lauffähig zu machen (gluonhq, 2021).

Die Umsetzung als Android-Applikation ist möglich. Im Prinzip existiert die Schichtenarchitektur auch im Android-System. Jedoch kann das UI nicht durch ein externes System ersetzt werden. Android-Anwendungen können mittels Emulatoren auf der iOS-Plattform ausgeführt werden. Auch können mittels Simulatoren Android Apps im Browser ausgeführt werden (Geekbullet Admin, 2021).

Die Ausführung von Android-Applikationen und JavaFX-Anwendungen auf Mobilen Endgeräten und Browser ist nativ nicht möglich, jedoch mittels Portierungen und Emulatoren zu bewerkstelligen. Diese Anforderungen können mit anderen Technologien ohne Hilfsmittel abgedeckt werden. Zudem sind solche Portierungen und Emulatoren immer Kompromisse; die Stabilität und Funktionalität der Software wird durch deren Einsatz dieser Hilfsmittel nicht positiv unterstützt.

Somit scheiden einige Lösungsvarianten aus, da diese die Musskriterien nicht erfüllen. Es existieren noch nachfolgende Lösungsvarianten, welche in der Nutzwertanalyse weiter behandelt werden können:

- Java/Jakarta EE mit JSP
- Webapplikation mit Backend (Java) und Frontend (Angular, Vue.js, React usw.)
- Spring mit Template-Engine (Thymeleaf, Freemarker usw.)

#### 4.4.4 Variantenentscheid

Im nachfolgenden Kapitel werden die Lösungsansätze mit den Wunschkriterien gegenübergestellt.

##### Präferenzmatrix

- 0: Der Zeilenwert ist weniger wichtig.
- 1: Der Zeilenwert ist gleich wichtig.
- 2: Der Zeilenwert ist wichtiger.

Das Vorgehen orientiert sich dabei an das Beispiel (Andrea, 2016): Ist der Zeilenwert weniger wichtig, gleich wichtig oder wichtiger als der Spaltenwert? Daraus ergibt sich die Punkteverteilung:

Tabelle 15: Präferenzmatrix. Quelle: Autor

Wunschkriterium	WK1	WK2	WK3	WK4	Summe	Faktor
<b>WK1</b>		2	2	1	5	42
<b>WK2</b>	0		1	0	1	8.25
<b>WK3</b>	0	1		1	2	16.5
<b>WK4</b>	1	2	1		4	33.25
				<b>Summe</b>	12	100

##### Nutzwertanalyse

- 0: Erfüllt das Kriterium nicht.
- 1: Erfüllt das Kriterium teilweise (oder mit Zusatzmittel).
- 2: Erfüllt das Kriterium.
- 3: Übertrifft das Kriterium.

JSP: Java/Jakarta EE mit JSP

WEB: Webapplikation mit Backend (Java) und Frontend (Angular, Vue.js, React usw.)

SPR: Spring mit Template-Engine (Thymeleaf, Freemarker usw.)

Tabelle 16: Nutzwertanalyse. Quelle: Autor

		Lösungsvarianten					
Kriterium	Faktor	JSP		WEB		SPR	
		Bewertung	Total	Bewertung	Total	Bewertung	Total
WK1	0.42	1	0.42	2 <sup>1</sup>	0.84	1	0.42
WK2	0.0825	0	0	1	0.0825	1	0.0825
WK3	0.165	0	0	2	0.33	2	0.33
WK4	0.3325	0	0	0	0	1	0.3325
	1		0.42		1.2525		1.165

<sup>1</sup> Marco Ris hat Erfahrungen mit HTML, JavaScript und CSS/SCSS, Christoph Grossmann und Peter Spielhofer haben Erfahrungen mit Angular, HTML, CSS

Die Nutzwertanalyse wurde auf Basis von Erfahrungen im Umgang mit den Technologien erarbeitet. Das Resultat deckt sich mit den Erwartungen des Projektteams.

Der Variantenentscheid hat ergeben, dass die Software als Backend (Java) und Frontend (Angular) – Lösung umgesetzt wird. Java SE steht unter der Oracle Technology Network License (Oracle, Oracle Technology Network License Agreement for Oracle Java SE, 2019) und Angular stellt das Framework unter der MIT-Lizenz zur Verfügung (Angular, 2021). Diese Lizenzen erfüllen die Lizenz-Bedingungen gemäss Grobanforderung (Kapitel 3.5).

#### 4.4.5 Persistierung der Daten

Für die Persistierung der Daten stehen diverse Datenbank-Technologien zur Verfügung welche prinzipiell zwischen SQL und NoSQL eingeordnet werden können. Die Wahl der geeigneten Datenbank-Technologie hängt von einigen Kriterien ab (Vettor, 2021):

*Tabelle 17: Fragestellung für die Wahl der Datenbanktechnologie. Quelle: Autor in Anlehnung an (Vettor, 2021)*

Fragestellung	SQL	NoSQL
<b>Werden die Daten mehr gelesen oder geschrieben?</b>	Vorteilhaft wenn Daten stark strukturiert sind und referentielle Integrität gefordert wird.	Optimiert für dynamische Daten, welche sich häufig ändern.
<b>Bestehen Relationen zwischen den Daten?</b>	Joins durch Tabellen für normalisierte Datenmodelle.	Relationen werden über denormalisierte (Nicht-Normalisierte) Datenmodelle ermöglicht.
<b>Werden komplexe Abfragen (Joins) eingesetzt?</b>	Unterstützung von komplexen Datenabrufen und Berichten.	Fokus auf einfache Datenabrufe ohne Joins.
<b>Werden vordefinierte Schemata benötigt?</b>	Datenstruktur und Datentyp sind in der Tabelle vorgegeben. Anpassungen sind möglich, jedoch muss die gesamte Tabelle angepasst werden.	Es können flexible Schemata hinzugefügt werden. Verschieden strukturierte Daten können dabei miteinander gemischt werden.

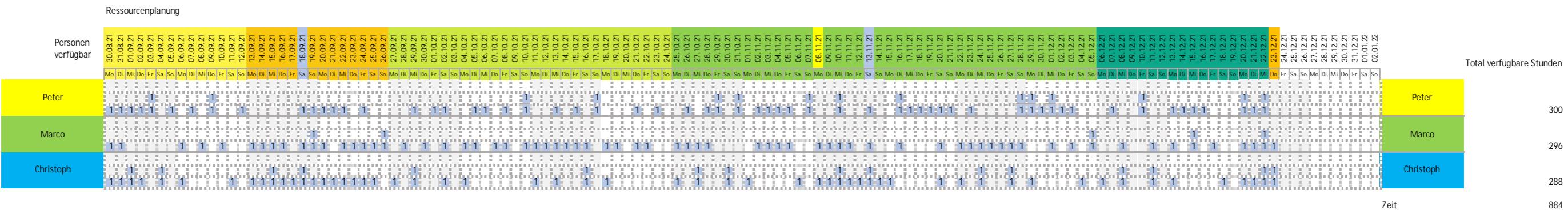
Auf Basis dieser Fragestellungen wäre der Einsatz einer NoSQL-Datenbanktechnologie für die Persistierung der einzelnen Planungen vorteilhaft, da die Datenstruktur, der einzelnen Planungen ohne festes Schema sind (jede Planung kann prinzipiell anders aussehen) und keine komplexe Abfragemöglichkeiten und Beziehungen zwischen den Daten benötigt werden. Jedoch existieren einige Daten, bei welchen eine Beziehung zwischen den Informationen Sinn machen. Beispielsweise könnten Modul-Informationen (Modul-Name, Semester-Verfügbarkeit, ECTS-Punkte, Relevanz-Punkte usw.) normalisiert und in einzelne Entitäten zerlegt werden. Dies würde die Speicherung von Duplikaten vermeiden.

Die Wahl der geeigneten Datenbank-Technologie fällt auf MySQL. Diese Wahl begründet sich wie folgt:

- Im Projektteam sind ausführlichere Erfahrungen zu MySQL vorhanden.
- MySQL bietet Möglichkeiten beider Datenbank-Welten: Relational wie auch NoSQL (Oracle, Guide to MySQL and NoSQL - Delivering the Best of Both Worlds, 2021).
- MySQL lässt sich lokal betreiben, so ist die Entwicklung dieses Systems nicht an ein Cloud-Produkt gebunden.
- MySQL bietet Tools wie MySQL Workbench oder HeidiSQL, welche dem Projektteam bekannt und erprobt sind.

## 4.5 Ressourceneinsatzplan

Zunächst werden die Verfügbarkeiten der Projektmitglieder ermittelt, anhand eines Ressourcenkapazitätsplan. Die Beteiligten tragen ihre dem Projekt zur Verfügung stehende Zeit in einen Zeitstrahl. Die blauen Einträge repräsentieren eine Verfügbarkeit von bis zu 4 Stunden. Diese Zeit soll nicht zwingend ausgelastet werden, dient aber als Richtwert.



## 4.6 Projektstrukturplan

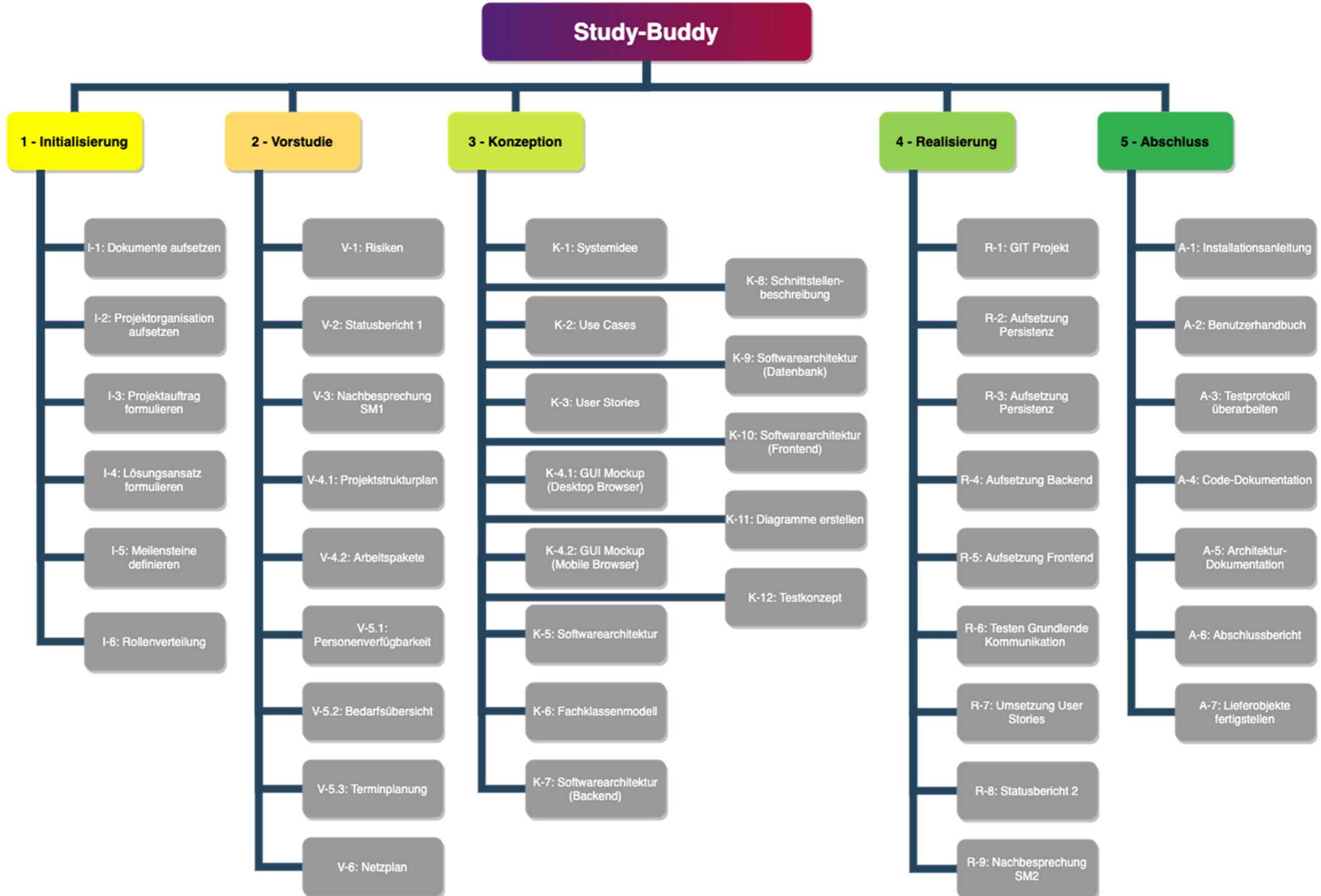


Abbildung 10: Projektstrukturplan. Quelle: Autor



## 4.7 Aufwandschätzverfahren

Zur Aufwandschätzung wird das Bottom-Up Prinzip angewandt. Nach der groben Planung des Projektumfangs werden die Arbeitspakete durch das Projektteam geschätzt. Die Schätzung der aufzuwendenden Zeiten der einzelnen Arbeitspakete haben wir mit der Drei-Zeiten-Schätzung erstellt. Hierbei werden immer ganze Zahlen verwendet und auch wieder auf solche aufgerundet.

Miller hat hierzu folgende Definition formuliert (Madauss, 2016):

- Optimistische Zeit (to): Schätzung der kürzest möglichen Zeit, die zur Durchführung einer bestimmten Tätigkeit benötigt wird. Eine Zeit, die nur dann erzielt wird, wenn man außergewöhnliches Glück hat und alles auf Anhieb funktioniert.
- Wahrscheinliche Zeit (tw): Schätzung der Zeit, die normalerweise zur Durchführung einer ähnlichen Tätigkeit benötigt wird. Eine Zeit, die bei mehrmaliger, voneinander abhängiger Wiederholung derselben Tätigkeit immer wieder benötigt wird.
- Pessimistische Zeit (tp): Schätzung der längsten Zeit, die für die Durchführung einer Tätigkeit benötigt wird. Eine Zeit, die nur bei außergewöhnlichen Problemen benötigt wird. Sie sollte die Möglichkeit anfänglicher Fehler oder gar neuen Beginnens einschließen.

Die Ermittlung der mittleren Zeit (tm) kann entsprechend der Beta-Verteilung nachfolgender Formel erfolgen:

$$tm = (to + 4 tw + tp)/6$$

Zahlen wurden gemäss errechnetem Wert auf ganze Zahlen aufgerundet.

## 4.8 RACI

Um die entsprechenden Verantwortlichen zu wählen, haben wir uns die RACI-Matrix zuhilfe genommen. RACI ist eine Matrix, welche die Kernverantwortlichkeiten der verschiedenen beteiligten Rollen an der Fertigstellung von Aufgaben oder Arbeitsergebnissen in einem Projekt oder Prozess beschreibt. Sie ist besonders hilfreich bei der Klarstellung von Rollen und Verantwortlichkeiten. RACI ist eine Abkürzung der vier üblichen Kernverantwortlichkeiten: (ISTQB, 2011)

- **Responsible** (Durchführungsverantwortlich)
- **Accountable** (Ergebnisverantwortlich)
- **Consulted** (Mitwirkend)
- **Informed** (Informiert)

Tabelle 11: RACI-Matrix der Arbeitspakete. Quelle: Autor

Arbeitspaket/ Mitglied	Christoph Grossmann	Marco Ris	Peter Spielhofer	Projekt Mitglieder	Auftraggeber
I-1	R	C	C	A	I
I-2	C	R	C	A	I
I-3	C	C	R	A	I
I-4	C	C	R	A	I
I-5	C	C	R	A	I
I-6	C	R	C	A	I
V-1	C	R	C	A	I
V-2	R	C	C	A	I
V-3	R	C	C	A	I
V-4.1	R	C	C	A	I
V-4.2	C	C	R	A	I
V-5.1	R	C	C	A	I
V-5.2	R	C	C	A	I
V-5.3	R	C	C	A	I
V-6	C	R	C	A	I
K-1	R	C	C	A	I
K-2	C	C	R	A	I
K-3	C	R	C	A	I
K-4.1	C	R	C	A	I
K-4.2	C	C	R	A	I
K-5	C	C	R	A	I
K-6	C	C	R	A	I
K-7	C	C	R	A	I
K-8	C	C	R	A	I
K-9	R	C	C	A	I
K-10	C	R	C	A	I
K-11	C	R	C	A	I
K-12	C	R	C	A	I
R-1	R	C	C	A	I
R-2	R	C	C	A	I
R-3	C	C	R	A	I
R-4	C	R	C	A	I
R-5	R	C	C	A	I
R-6	R	C	C	A	I
R-7	C	C	R	A	I
R-8	R	C	C	A	I
R-9	R	C	C	A	I
A-1	C	C	R	A	I
A-2	R	C	C	A	I
A-3	C	C	R	A	I
A-4	C	R	C	A	I
A-5	R	C	C	A	I
A-6	R	C	C	A	I
A-7	C	C	R	A	I

## 4.9 Arbeitspakete Initialisierung

Tabelle 18: Arbeitspaket I-1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-1
AP-Titel	Dokumente aufsetzen		
AP-Start	30.08.2021	AP-Ende	31.08.2021
SOLL-Zeit	6	IST-Zeit	4
<b>Beschreibung</b>			
Dokumente müssen aufgesetzt werden und Entscheidungen getroffen werden, welche Dateiformate sowie Programme verwendet werden.			
<b>Voraussetzungen zum Start</b>			
Kick-Off			
<b>Erwartete Resultate</b>			
<ul style="list-style-type: none"> <li>• Dokumentation</li> <li>• Terminplanung Gantt</li> <li>• Ressourcenplanung</li> <li>• Sitzungsprotokolle</li> <li>• Statusmeeting</li> </ul>			
Dokumente sind erstellt und gelten als Vorlagen			
<b>Bemerkung</b>			
O365 wird verwendet, wenn immer möglich, um Kollaboration zu ermöglichen.			
Verantwortlich	CHG	Mitwirkend	PS, MR

Tabelle 19: Arbeitspaket I-2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-2
AP-Titel	Projektorganisation aufstellen		
AP-Start	30.08.2021	AP-Ende	02.09.2021
SOLL-Zeit	15	IST-Zeit	15
<b>Beschreibung</b>			
<ul style="list-style-type: none"> <li>• Vorgehensmodell</li> <li>• Phasenmodell</li> <li>• Rollen</li> <li>• Ressourceneinsatzplan</li> </ul>			
<b>Voraussetzungen zum Start</b>			
I-1			
<b>Erwartete Resultate</b>			
<ul style="list-style-type: none"> <li>• Vorgehensmodell ist definiert in der Dokumentation</li> <li>• Phasenmodell ist verständlich illustriert</li> <li>• Rollen wurden definiert und in Projektorganisation festgehalten</li> <li>• Ressourceneinsatzplan ist bereit, um von den beteiligten ausgefüllt zu werden</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	PS, CHG

Tabelle 20: Arbeitspaket I-3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-3
AP-Titel	Projektauftrag formulieren		
AP-Start	30.08.2021	AP-Ende	02.09.2021
SOLL-Zeit	20	IST-Zeit	25
Beschreibung			
	<ul style="list-style-type: none"> <li>• Ausgangslage</li> <li>• Projektidee</li> <li>• Projektziele</li> <li>• Mindestanforderungen</li> <li>• Grobanforderungen</li> <li>• Abgrenzungen</li> <li>• Kosten</li> </ul>		
Voraussetzungen zum Start			
I-1			
Erwartete Resultate			
Dokumentation ergänzt mit:			
	<ul style="list-style-type: none"> <li>• Formulierte Ausgangslage</li> <li>• Projektidee niedergeschrieben</li> <li>• Definierte Mindestanforderungen</li> <li>• Definierte Grobanforderungen</li> <li>• Erste Abgrenzungen festgelegt</li> <li>• Erwartete Kosten analysiert</li> </ul>		
Bemerkung			
Verantwortlich	PS	Mitwirkend	CHG, MR

Tabelle 21: Arbeitspaket I-4. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-4
AP-Titel	Lösungsansatz formulieren		
AP-Start	02.09.2021	AP-Ende	06.09.2021
SOLL-Zeit	30	IST-Zeit	20
Beschreibung			
Anhand der Projektziele deren Mindest- und Grobanforderungen sollen Lösungsansätze ausgearbeitet werden und zu einem Variantenentscheid gebracht werden.			
Voraussetzungen zum Start			
I-3			
Erwartete Resultate			
Dokumentation ergänzt mit:			
	<ul style="list-style-type: none"> <li>• Muss-/ Wunschkriterien</li> <li>• Ausgearbeitete Lösungsansätze</li> <li>• Begründeter Variantenentscheid</li> </ul>		
Bemerkung			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 22: Arbeitspaket I-5. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-5
AP-Titel	Meilensteine definieren		
AP-Start	07.09.2021	AP-Ende	09.09.2021
SOLL-Zeit	8	IST-Zeit	5
<b>Beschreibung</b>			
Grobe Meilensteine aufgrund Phasenplanung und Lösungsansatz definieren.			
<b>Voraussetzungen zum Start</b>			
I-4			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
<ul style="list-style-type: none"> <li>• Meilensteinplanung</li> <li>• Meilenstein Ergebnisse</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	CHG, MR

Tabelle 23: Arbeitspaket I-6. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Initialisierung	AP-Nr.	I-6
AP-Titel	Rollenverteilung		
AP-Start	10.09.2021	AP-Ende	11.09.2021
SOLL-Zeit	3	IST-Zeit	3
<b>Beschreibung</b>			
Eine rotierende Rollenverteilung soll erstellt werden die garantiert, dass alle Projektmitglieder alle Rollen mindestens einmal inne haben werden.			
<b>Voraussetzungen zum Start</b>			
I-2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Rotierende Rollenverteilung</li> </ul>			
<b>Bemerkung</b>			
Wurde in der wöchentlichen Sitzung gemeinsam abgestimmt.			
Verantwortlich	MR	Mitwirkend	CHG, PS

## 4.10 Arbeitspakete Vorstudie

Tabelle 24: Arbeitspaket V-1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-1
AP-Titel	Risiken		
AP-Start	13.09.2021	AP-Ende	15.09.2021
SOLL-Zeit	8	IST-Zeit	12
<b>Beschreibung</b> Projektrisiken ausfindig machen und mit Eintrittswahrscheinlichkeit und Auswirkungsniveau Kategorisieren. Allenfalls Massnahmen definieren.			
<b>Voraussetzungen zum Start</b>			
<b>Erwartete Resultate</b> Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Risikodarstellung</li> <li>• Risikomatrix</li> </ul>			
<b>Bemerkung</b> Zur Minderung der «roten» Risiken werden Workshops organisiert.			
Verantwortlich	MR	Mitwirkend	PS, CHG

Tabelle 25: Arbeitspaket V-2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-2
AP-Titel	Statusbericht 1		
AP-Start	16.09.2021	AP-Ende	18.09.2021
SOLL-Zeit	4	IST-Zeit	6
<b>Beschreibung</b> Inhalt des Statusberichts Folgende Beurteilungsaspekte müssen bezüglich Einhaltung/Abweichung (Soll-/ Ist-Vergleich) im Projektstatus-Bericht aufgenommen werden:			
<ul style="list-style-type: none"> <li>• Kurzbeschreibung Gesamtstatus inkl. Bewertung (kritisch, teilweise kritisch, planmäßig) und allfällige Massnahmen</li> <li>• Status Termine inkl. Bewertung / allfällige Massnahmen</li> <li>• Status Lieferobjekte inkl. Bewertung / allfällige Massnahmen</li> <li>• Status Qualität inkl. Bewertung / allfällige Massnahmen</li> <li>• Andere Probleme / Risiken und allfällige Massnahmen</li> <li>• Nächste Schritte / Änderungsanträge</li> <li>• Aktualisierte Rollenorganisation</li> <li>• Nur 1. Statusmeeting: Nennung des gewählten Vorgehensmodell der Softwareentwicklung</li> <li>• Varianten</li> </ul>			
<b>Voraussetzungen zum Start</b>			
<b>Erwartete Resultate</b> Eigenständiges Dokument:			
<ul style="list-style-type: none"> <li>• Schriftlich erstellter Statusbericht</li> </ul>			
<b>Bemerkung</b> Statusbericht muss abgegeben werden nach Status-Meeting.			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 26: Arbeitspaket V-3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-3
AP-Titel	Nachbesprechung SM1		
AP-Start	18.09.2021	AP-Ende	19.09.2021
SOLL-Zeit	2	IST-Zeit	2
<b>Beschreibung</b>			
Massnahmen definieren aus Status-Meeting.			
<b>Voraussetzungen zum Start</b>			
V-2			
<b>Erwartete Resultate</b>			
<ul style="list-style-type: none"> <li>• Massnahmen eingeleitet gemäss Status-Meeting</li> </ul>			
<b>Bemerkung</b>			
Nachbesprechung dokumentieren mit gelerntem und niederschreiben. Dieses Dokument muss ebenfalls abgegeben werden zu Händen der Dozenten.			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 27: Arbeitspaket V-4.1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-4.1
AP-Titel	Projektstrukturplan		
AP-Start	19.09.2021	AP-Ende	22.09.2021
SOLL-Zeit	10	IST-Zeit	6
<b>Beschreibung</b>			
Anhand der Projektziele werden Projektstrukturplan und Netzplan erstellt.			
<b>Voraussetzungen zum Start</b>			
V-3			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
<ul style="list-style-type: none"> <li>• Projektstrukturplan</li> <li>• Netzplan</li> </ul>			
<b>Bemerkung</b>			
Als Massnahme aus dem Statusmeeting wurde dieses Arbeitspaket aufgeteilt, um einen höheren Detailierungsgrad zu erhalten. Es werden Arbeitspakte erstellt und anschliessend in einem Netzplan dargestellt.			
Verantwortlich	CHG	Mitwirkend	PS

Tabelle 28: Arbeitspaket V-4.2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-4.2
AP-Titel	Arbeitspakete		
AP-Start	19.09.2021	AP-Ende	22.09.2021
SOLL-Zeit	14	IST-Zeit	22
<b>Beschreibung</b>			
Tätigkeiten in planbare Arbeitspakete formulieren.			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• AP-Kennung</li> <li>• Zeitraum</li> <li>• Soll-/Ist-Zeit</li> <li>• Abhängigkeit</li> <li>• Erwartete Resultate</li> <li>• Verantwortlichkeit</li> </ul>			
<b>Voraussetzungen zum Start</b>			
V-3			
<b>Erwartete Resultate</b>			
Definierte Arbeitspakete mit geschätzter Soll-Zeit und festgelegter Verantwortlichkeit.			
<b>Bemerkung</b>			
Detaillierte Arbeitspakete wurde als Massnahme aus dem 1. Status-Meeting erstellt.			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 29: Arbeitspaket V-5.1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-5.1
AP-Titel	Personenverfügbarkeit		
AP-Start	22.09.2021	AP-Ende	22.09.2021
SOLL-Zeit	4	IST-Zeit	2
<b>Beschreibung</b>			
Die Personenverfügbarkeit wird unter den Projektmitglieder erhoben. Die verfügbare Zeit einer Ressource soll aufgezeigt werden.			
<b>Voraussetzungen zum Start</b>			
V-3			
<b>Erwartete Resultate</b>			
Personenverfügbarkeit in einer Grafik dargestellt die ersichtlich macht, wann die einzelnen Projektmitglieder verfügbar sind und mögliche Kapazitätsgrenzen eintreffen.			
<b>Bemerkung</b>			
Wurde als Massnahme aus dem 1. Status-Meeting erstellt.			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 30: Arbeitspaket V-5.2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-5.2
AP-Titel	Bedarfsermittlung		
AP-Start	22.09.2021	AP-Ende	22.09.2021
SOLL-Zeit	5	IST-Zeit	3
<b>Beschreibung</b>			
Arbeitspakete werden dem Zeitstrahl gegenübergestellt und Kapazitätsbedarf ermittelt.			
<b>Voraussetzungen zum Start</b>			
V-4.2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
• Kapazitätsbedarf			
<b>Bemerkung</b>			
Wurde als Massnahme aus dem 1. Status-Meeting erstellt.			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 31: Arbeitspaket V-5.3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-5.3
AP-Titel	Terminplanung		
AP-Start	23.09.2021	AP-Ende	24.09.2021
SOLL-Zeit	5	IST-Zeit	4
<b>Beschreibung</b>			
Die Personenverfügbarkeit wird erhoben und Arbeitspakete gemäss Kapazität eingeplant.			
<b>Voraussetzungen zum Start</b>			
V-5.1, V-5.2			
<b>Erwartete Resultate</b>			
Arbeitspakete dargestellt in terminlicher Abfolge, abgestimmt mit der Bedarfsermittlung und Ressourcenverfügbarkeit.			
<b>Bemerkung</b>			
Wurde als Massnahme aus dem 1. Status-Meeting erstellt.			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 32: Arbeitspaket V-6. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Vorstudie	AP-Nr.	V-6
AP-Titel	Netzplan		
AP-Start	24.09.2021	AP-Ende	26.09.2021
SOLL-Zeit	15	IST-Zeit	14
<b>Beschreibung</b>			
Anhand der Arbeitspakete und Terminplanung ist ein Netzplan zu erstellen, der Abhängigkeiten und kritischen Pfad aufzeigt			
<b>Voraussetzungen zum Start</b>			
V-5.3			
<b>Erwartete Resultate</b>			
Arbeitspakete wurden mit Terminplanung in einen Netzplan vereint und ein kritischer Pfad dargestellt.			
<b>Bemerkung</b>			
Wurde als Massnahme aus dem 1. Status-Meeting erstellt.			
Verantwortlich	MR	Mitwirkend	CHG, PS

## 4.11 Arbeitspakete Konzeption

Tabelle 33: Arbeitspaket K-1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-1
AP-Titel	Systemidee		
AP-Start	27.09.2021	AP-Ende	27.09.2021
SOLL-Zeit	4	IST-Zeit	5
Beschreibung	Anhand der Projektidee und des Lösungsansatzes wird eine Systemidee erstellt auf Basis welcher, anschliessend Use Cases erstellt werden.		
Voraussetzungen zum Start			
Erwartete Resultate			
Dokumentation ergänzen mit:	<ul style="list-style-type: none"> <li>• Systemidee</li> </ul>		
Bemerkung			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 34: Arbeitspaket K-2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-2
AP-Titel	Use Cases		
AP-Start	29.09.2021	AP-Ende	29.09.2021
SOLL-Zeit	3	IST-Zeit	4
Beschreibung	Beschreibung der Use Cases anhand der Systemidee.		
Voraussetzungen zum Start			
K-1			
Erwartete Resultate			
Dokumentation ergänzt mit:	<ul style="list-style-type: none"> <li>• Sammlung von primären Use Cases aus der Systemidee abstrahiert</li> </ul>		
Bemerkung	Es können im Laufe der Konzeption und Realisierung weitere kleinere Use Cases dazu kommen. Dies ist jetzt noch nicht abschätzbar.		
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 35: Arbeitspaket K-3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-3
AP-Titel	User Stories		
AP-Start	29.09.2021	AP-Ende	03.10.2021
SOLL-Zeit	16	IST-Zeit	15
<b>Beschreibung</b>			
Beschreibung der User Stories gemäss Use Case. Die User Stories werden gemeinsamen definiert			
<b>Voraussetzungen zum Start</b>			
K-2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
• Sammlung von primären User Stories aus den Use Cases abstrahiert			
<b>Bemerkung</b>			
Es können im Laufe der Konzeption und Realisierung weitere kleinere Use Stories dazu kommen. Dies ist jetzt noch nicht abschätzbar.			
Verantwortlich	MR	Mitwirkend	CHG, PS

Tabelle 36: Arbeitspaket K-4.1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-4.1
AP-Titel	GUI Mockup (Desktop Browser)		
AP-Start	02.10.2021	AP-Ende	02.10.2021
SOLL-Zeit	4	IST-Zeit	3
<b>Beschreibung</b>			
Anhand der Use Cases wird ein Mockup erstellt für das GUI auf dem Desktop. Um die der Programmierung eine grobe Richtlinie zu erhalten, werden Mockups verwendet, um ein erstes Bild in den Köpfen der Projektmitarbeiter zu erstellen. Auf den Mockups sollen der Navigationsverlauf und die einzelnen Szenen ersichtlich sein. In den Szenen werden die Eingabe-/ und Ausgabefelder sowie Hinwestexte, Titel und grobe Richtlinien für die Darstellung (Farbe, Schriftgrösse) definiert.			
<b>Voraussetzungen zum Start</b>			
K-2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
• Mockup Desktop Browser			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	CHG, PS

Tabelle 37: Arbeitspaket K-4.2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-4.2
AP-Titel	GUI Mockup (Mobile Browser)		
AP-Start	02.10.2021	AP-Ende	02.10.2021
SOLL-Zeit	4	IST-Zeit	2
<b>Beschreibung</b>			
Anhand der Use Cases wird ein Mockup erstellt für das GUI auf dem Mobile Browser. Um die der Programmierung eine grobe Richtlinie zu erhalten, werden Mockups verwendet, um ein erstes Bild in den Köpfen der Projektmitarbeiter zu erstellen. Die Mockups für Mobile Geräte orientieren sich an den Mockups für Desktop-Geräte. In den Mockups für Mobile Geräte werden jedoch auf die spezifischen Eigenschaften kleiner Bildschirme eingegangen: Was kann evtl. ausgeblendet werden? Wie werden grosse Elemente dargestellt (umbrechen, anders darstellen usw.).			
<b>Voraussetzungen zum Start</b>			
K-2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Mockup Mobile Browser</li> </ul>			
<b>Bemerkung</b>			
Verschiedene Displaygrößen werden nicht miteinbezogen.			
Verantwortlich	PS	Mitwirkend	CHG, MR

Tabelle 38: Arbeitspaket K-5. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-5
AP-Titel	Softwarearchitektur (Backend, Frontend und Datenbank)		
AP-Start	05.10.2021	AP-Ende	06.10.2021
SOLL-Zeit	3	IST-Zeit	2
<b>Beschreibung</b>			
Bausteinsicht der Elemente: Backend, Frontend und Datenbank.			
<b>Voraussetzungen zum Start</b>			
K-4.1, K-4.2			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Bausteinsicht (Sichten der Architektur)</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	CHG, MR

Tabelle 39: Arbeitspaket K-6. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	K-6
AP-Titel	Fachklassenmodell		
AP-Start	05.10.2021	AP-Ende	06.10.2021
SOLL-Zeit	4	IST-Zeit	4
<b>Beschreibung</b>			
Fachklassen beschreiben die wichtigsten Klassen, die durch Use Cases identifiziert werden konnten.			
<b>Voraussetzungen zum Start</b>			
K-5			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
• Fachklassenmodelle			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 40: Arbeitspaket K-7. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-7
AP-Titel	Softwarearchitektur (Backend): Java Klassenmodelle		
AP-Start	08.10.2021	AP-Ende	10.10.2021
SOLL-Zeit	8	IST-Zeit	10
<b>Beschreibung</b>			
Java Klassenmodelle des Backends erstellen.			
<b>Voraussetzungen zum Start</b>			
K-6			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
• Klassenmodelle			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 41: Arbeitspaket K-8. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	K-8
AP-Titel	Schnittstellenbeschreibung		
AP-Start	08.10.2021	AP-Ende	10.10.2021
SOLL-Zeit	10	IST-Zeit	11
<b>Beschreibung</b>			
Schnittstellen-Konzept. Im Schnittstellen-Konzept wird die API-Design:			
<ul style="list-style-type: none"> <li>• Schnittstellen: Frontend → Backend, Backend → Datenbank</li> <li>• Methoden</li> <li>• Benötigter Eingabewert (Parameter, Daten-Objekt/JSON)</li> <li>• Erwarteter Rückgabewert (http-Response, Daten)</li> </ul>			
Die definitive API-Dokumentation wird anschliessend (beim Abschluss) mittels SWAGGER erstellen.			
<b>Voraussetzungen zum Start</b>			
K-7			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Schnittstellenbeschreibung</li> <li>• API-Dokumentation</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 42: Arbeitspaket K-9. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-9
AP-Titel	Softwarearchitektur (Datenbank): Datenmodell		
AP-Start	11.10.2021	AP-Ende	13.10.2021
SOLL-Zeit	8	IST-Zeit	10
<b>Beschreibung</b>			
Ausarbeitung der benötigten Datenmodelle und Dokumentation jener. Erstellung eines ERD und Beschreibung der Attribute.			
<b>Voraussetzungen zum Start</b>			
K-8			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
<ul style="list-style-type: none"> <li>• Datenmodelle</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 43: Arbeitspaket K-10. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-10
AP-Titel	Softwarearchitektur (Frontend)		
AP-Start	11.10.2021	AP-Ende	13.10.2021
SOLL-Zeit	4	IST-Zeit	3
<b>Beschreibung</b>			
Darstellung Aufbau des Frontend (Komponenten, Klassen, Abhängigkeiten).			
<b>Voraussetzungen zum Start</b>			
K-8			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
• Softwarearchitektur Frontend			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	PS, CHG

Tabelle 44: Arbeitspaket K-11. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-11
AP-Titel	Diagramme erstellen		
AP-Start	15.10.2021	AP-Ende	17.10.2021
SOLL-Zeit	8	IST-Zeit	8
<b>Beschreibung</b>			
Dynamisches Modell: Zustands-, Sequenz- oder Aktivitätsdiagramm welches einen weiteren Ablauf komplexer Abläufe dokumentiert. Hier muss dann zum Schluss unbedingt rein, welchen Ablauf wir beschreiben. Zum jetzigen Zeitpunkt weiss ich es noch nicht.			
<b>Voraussetzungen zum Start</b>			
K-9, K-10			
<b>Erwartete Resultate</b>			
Dokumentation ergänzen mit:			
• Dynamische Modelle			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	CHG, PS

Tabelle 45: Arbeitspaket K-12. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Konzeption	AP-Nr.	K-12
AP-Titel	Testkonzept		
AP-Start	18.10.2021	AP-Ende	23.10.2021
SOLL-Zeit	20	IST-Zeit	24
<b>Beschreibung</b>			
Im Testkonzept werden Testziele, Testobjekte, Testarten, Testinfrastruktur und Testorganisation. Eine Vorgabe für Testfallbeschreibungen und die generelle Testplanung muss erstellt werden.			
<b>Voraussetzungen zum Start</b>			
Alle Vorhergehenden Ergebnisse (APs)			
<b>Erwartete Resultate</b>			
Dokumentation ergänzt mit:			
<ul style="list-style-type: none"> <li>• Testkonzept <ul style="list-style-type: none"> <li>◦ Testziele</li> <li>◦ Testobjekte</li> <li>◦ Testarten</li> <li>◦ Testumgebung</li> <li>◦ Testinfrastruktur</li> <li>◦ Testorganisation</li> <li>◦ Testfallbeschreibung</li> </ul> </li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	PS, CHG

## 4.12 Arbeitspakte Realisierung

Tabelle 46: Arbeitspaket R-1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-1
AP-Titel	GIT Projekt		
AP-Start	25.10.2021	AP-Ende	26.10.2021
SOLL-Zeit	1	IST-Zeit	1
Beschreibung			
GIT-Projekt initialisieren	<ul style="list-style-type: none"> <li>• GIT-Projekt erstellen (Name definieren)</li> <li>• Berechtigungen definieren</li> <li>• Zugriff sicherstellen</li> </ul>		
Voraussetzungen zum Start			
Erwartete Resultate	Verwendbares GIT für das ganze Projektteam		
Bemerkung			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 47: Arbeitspaket R-2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-2
AP-Titel	Aufsetzung Persistenz I		
AP-Start	25.11.2021	AP-Ende	26.10.2021
SOLL-Zeit	2	IST-Zeit	2
Beschreibung			
Datenbanksystem installieren	<ul style="list-style-type: none"> <li>• Docker installieren</li> <li>• Dockerfile erstellen</li> <li>• Docker Container (Mysql)</li> </ul>		
Voraussetzungen zum Start			
Erwartete Resultate	Erstelltes Dockerfile		
Bemerkung			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 48: Arbeitspaket R-3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-3
AP-Titel	Aufsetzung Persistenz II		
AP-Start	26.10.2021	AP-Ende	27.10.2021
SOLL-Zeit	8	IST-Zeit	10
<b>Beschreibung</b>			
Datenbanksystem initialisieren			
<ul style="list-style-type: none"> <li>• Schema gemäss Datenmodell umsetzen</li> <li>• Initialer Datenstamm erstellen (CSV-Datei welche beim Starten von Docker eingelesen wird)</li> <li>• Schema und Datenstamm in Docker integrieren</li> </ul>			
<b>Voraussetzungen zum Start</b>			
R-2			
<b>Erwartete Resultate</b>			
Datenstamm und Schema wird bei der Erstellung des Docker eingebunden.			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 49: Arbeitspaket R-4. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-4
AP-Titel	Aufsetzung Backend		
AP-Start	28.10.2021	AP-Ende	29.10.2021
SOLL-Zeit	6	IST-Zeit	6
<b>Beschreibung</b>			
Spring installieren			
<ul style="list-style-type: none"> <li>• Mit GIT-Projekt verknüpfen</li> <li>• Maven Projekt erstellen</li> <li>• POM.XML definieren (Abhängigkeiten/Zusatzpakete konfigurieren, GroupID und Artifact-ID definieren)</li> <li>• Paket-Struktur (Main und Test) umsetzen</li> <li>• Application.properties anpassen           <ul style="list-style-type: none"> <li>◦ Logging, Datenbank-Zugriff</li> </ul> </li> <li>• Projektstart testen (Projekt ausführen)</li> </ul>			
<b>Voraussetzungen zum Start</b>			
R-1			
<b>Erwartete Resultate</b>			
Ausführbare Grundapplikation erstellt			
<b>Bemerkung</b>			
Verantwortlich	MR	Mitwirkend	PS, CHG

Tabelle 50: Arbeitspaket R-5. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-5
AP-Titel	Aufsetzung Frontend		
AP-Start	30.10.2021	AP-Ende	31.10.2021
SOLL-Zeit	4	IST-Zeit	4
<b>Beschreibung</b>			
Frontend Projekt initialisieren			
<ul style="list-style-type: none"> <li>• Mit GIT-Projekt verknüpfen</li> <li>• Installation gemäss bootstrap.io</li> <li>• Bootstrap (ngx-bootstrap) einbinden</li> <li>• Projektstart testen</li> <li>• Grundlegende Komponentenstruktur abbilden gemäss Klassendiagramm           <ul style="list-style-type: none"> <li>◦ Ordner für Komponenten erstellen</li> </ul> </li> </ul>			
<b>Voraussetzungen zum Start</b>			
R-1			
<b>Erwartete Resultate</b>			
<ul style="list-style-type: none"> <li>• Ausführbare Grundapplikation erstellt</li> <li>• Komponentenstruktur erstellt</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 51: Arbeitspaket R-6. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-6
AP-Titel	Testen Grundlende Kommunikation Backend-Frontend-Datenbank		
AP-Start	01.11.2021	AP-Ende	02.11.2021
SOLL-Zeit	4	IST-Zeit	5
<b>Beschreibung</b>			
<ul style="list-style-type: none"> <li>• Datenbankverbindung in Spring konfigurieren</li> <li>• Einen Webcontroller im Backend anlegen welche Daten von der Datenbank ausliest und zur Verfügung stellt</li> <li>• Einen Test-Service auf dem Frontend anlegen, welcher mit dem Webcontroller im Backend kommuniziert</li> <li>• Verbindungs- resp. Datenabruft von Frontend -&gt; Backend -&gt; Datenbank testen</li> <li>• Rückbau der Test-Implementierungen</li> </ul>			
<b>Voraussetzungen zum Start</b>			
R-2, R-3, R-4			
<b>Erwartete Resultate</b>			
Durchgehend sichergestellter Datenfluss Frontend -> Backend -> Datenbank			
<b>Bemerkung</b>			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 52: Arbeitspaket R-7. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-7
AP-Titel	Umsetzung User Stories		
AP-Start	03.11.2021	AP-Ende	05.12.2021
SOLL-Zeit	100	IST-Zeit	118
<b>Beschreibung</b>			
Die Soll-Zeit basiert auf der Summe aller Arbeitspakete.			
Umsetzung der User Stories gemäss TDD-Prinzip:			
<ol style="list-style-type: none"> <li>1. Erstellen des Testfalls im Backend (je nach Komplexität: 0.5-1h)</li> <li>2. Implementation gemäss Testfall (Test auf grün/funktionierend)</li> <li>3. Java-Doc ergänzen</li> <li>4. Abgabe und Überprüfung des Testfalls und der Implementation nach 4-Augen-Prinzip (0.5h)</li> <li>5. Dokumentation des Testfalls im Testprotokoll (0.5h)</li> </ol>			
<b>Voraussetzungen zum Start</b>			
R-6			
<b>Erwartete Resultate</b>			
Iterative Abarbeitung der User Stories:			
<ul style="list-style-type: none"> <li>• Dokumentation ergänzt mit den abgearbeiteten User Stories</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 53: Arbeitspaket R-8. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-8
AP-Titel	Statusbericht 2		
AP-Start	12.11.2021	AP-Ende	13.11.2021
SOLL-Zeit	6	IST-Zeit	8
<b>Beschreibung</b>			
Inhalt des Statusberichts			
Folgende Beurteilungsaspekte müssen bezüglich Einhaltung/Abweichung (Soll-/ Ist-Vergleich) im Projektstatus-Bericht aufgenommen werden:			
<ul style="list-style-type: none"> <li>• Kurzbeschreibung Gesamtstatus inkl. Bewertung (kritisch, teilweise kritisch, planmäßig) und allfällige Massnahmen</li> <li>• Status Termine inkl. Bewertung / allfällige Massnahmen</li> <li>• Status Lieferobjekte inkl. Bewertung / allfällige Massnahmen</li> <li>• Status Qualität inkl. Bewertung / allfällige Massnahmen</li> <li>• Andere Probleme / Risiken und allfällige Massnahmen</li> <li>• Nächste Schritte / Änderungsanträge</li> <li>• Aktualisierte Rollenorganisation</li> </ul>			
<b>Voraussetzungen zum Start</b>			
<b>Erwartete Resultate</b>			
Eigenständiges Dokument:			
<ul style="list-style-type: none"> <li>• Schriftlich erstellter Statusbericht</li> </ul>			
<b>Bemerkung</b>			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 54: Arbeitspaket R-9. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Realisierung	AP-Nr.	R-9
AP-Titel	Nachbesprechung SM2		
AP-Start	15.11.2021	AP-Ende	15.11.2021
SOLL-Zeit	2	IST-Zeit	2
Beschreibung			
Massnahmen definieren aus Status-Meeting.			
Voraussetzungen zum Start			
R-8			
Erwartete Resultate			
Bemerkung			
Nachbesprechung dokumentieren mit gelerntem und niederschreiben. Dieses Dokument muss ebenfalls abgegeben werden zu Händen der Dozenten.			
Nachbesprechung wurde erst am Montag nach dem eigentlichen Status-Meeting durchgeführt, aufgrund des Umzugstermins eines Projektmitgliedes. Die Abgabe der Dokumente an die Dozenten erfolgte per Mail.			
Verantwortlich	CHG	Mitwirkend	MR, PS

## 4.13 Arbeitspakete Abschluss

Tabelle 55: Arbeitspaket A-1. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-1
AP-Titel	Installationsanleitung		
AP-Start	07.12.2021	AP-Ende	09.12.2021
SOLL-Zeit	4	IST-Zeit	6
<b>Beschreibung</b> Die Installationsanleitung enthält alle Informationen, um die Applikation auf einem neuen Device einzurichten und starten zu können. Systemanforderungen und Vorbereitungen können entnommen werden.			
<b>Voraussetzungen zum Start</b>			
<b>Erwartete Resultate</b> Eigenständiges Dokument, welches eine Schritt-für-Schritt-Anleitung beinhaltet, um die Applikation auszuführen.			
<b>Bemerkung</b>			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 56: Arbeitspaket A-2. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-2
AP-Titel	Benutzerhandbuch		
AP-Start	08.12.2021	AP-Ende	08.12.2021
SOLL-Zeit	4	IST-Zeit	4
<b>Beschreibung</b> Das Benutzerhandbuch enthält alle Informationen, um die Applikation bedienen zu können. Es werden Funktionen erklärt, auch wenn diese den Entwicklern als intuitiv erscheinen mögen.			
<b>Voraussetzungen zum Start</b>			
<b>Erwartete Resultate</b> Eigenständiges Dokument, welches die Funktionen der Applikation erklärt und durch die Applikation leitet.			
<b>Bemerkung</b>			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 57: Arbeitspaket A-3. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-3
AP-Titel	Testprotokoll überarbeiten		
AP-Start	10.12.2021	AP-Ende	10.12.2021
SOLL-Zeit	6	IST-Zeit	1
Beschreibung	Alle Testfälle ausführen und deren Erfüllung prüfen. Testprotokoll bearbeiten und eventuelle Abweichungen zum IST-Stand nachtragen.		
Voraussetzungen zum Start			
Erwartete Resultate			
Finalisiertes Testprotokoll			
Bemerkung			
Verantwortlich	PS	Mitwirkend	MR, CHG

Tabelle 58: Arbeitspaket A-4. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-4
AP-Titel	Code-Dokumentation		
AP-Start	11.12.2021	AP-Ende	13.12.2021
SOLL-Zeit	8	IST-Zeit	7
Beschreibung	Fertigstellung des JavaDoc und der AngularDoc		
Voraussetzungen zum Start			
Erwartete Resultate			
HTML Dokumentationen bereitgestellt für die Abgabe			
Bemerkung			
Verantwortlich	MR	Mitwirkend	CHG, PS

Tabelle 59: Arbeitspaket A-5. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-5
AP-Titel	Architektur-Dokumentation		
AP-Start	11.12.2021	AP-Ende	13.12.2021
SOLL-Zeit	10	IST-Zeit	8
Beschreibung	Überarbeitung der Softwarearchitekturdokumente: IST-Stand abgleichen. Da in der Abgabe der effektive Stand gewünscht/erwartet wird, müssen die Softwarearchitekturdokumente, welche in der Konzeption erstellt wurden, auf den aktuellen Stand angepasst werden.		
Voraussetzungen zum Start			
Erwartete Resultate			
Aktualisierte Softwarearchitektur in der Dokumentation			
Bemerkung			
Verantwortlich	CHG	Mitwirkend	MR, PS

Tabelle 60: Arbeitspaket A-6. Quelle: Autor

Arbeitspaketbeschreibung			
Phase	Abschluss	AP-Nr.	A-6
AP-Titel	Abschlussbericht		
AP-Start	11.12.2021	AP-Ende	13.12.2021
SOLL-Zeit	10	IST-Zeit	12
Beschreibung			
Der Abschlussbericht enthält eine Reflektion der aufgetretenen Probleme, die Planung wird der Ist-Situation gegenübergestellt und eine Auswertung angestellt. Um einen Mehrwert daraus zu ziehen, werden Lessons-Learned formuliert.			
Voraussetzungen zum Start			
Abschluss Realisierung			
Erwartete Resultate			
Dokumentation ergänzen mit:			
<ul style="list-style-type: none"> <li>• Abschlussbericht           <ul style="list-style-type: none"> <li>○ Reflektion Probleme</li> <li>○ Auswertung Planung</li> <li>○ Lessons Learned</li> </ul> </li> </ul>			
Bemerkung			
Verantwortlich	CHG	Mitwirkend	PS, MR

Tabelle 61: Arbeitspaket A-7. Quelle: Autor

Arbeitspaketbeschreibung						
Phase	Abschluss	AP-Nr.	A-7			
AP-Titel	Lieferobjekte fertigstellen					
AP-Start	12.12.2021	AP-Ende	17.12.2021			
SOLL-Zeit	18	IST-Zeit	25			
Beschreibung						
Finale Kontrolle der Lieferobjekte und Fertigstellung der fehlenden Komponenten.						
<ul style="list-style-type: none"> <li>• Projektantrag</li> <li>• Projektstrukturplan</li> <li>• Konzeption</li> <li>• Testkonzept</li> <li>• Testprotokolle</li> <li>• Javadoc</li> <li>• Installationsanleitung</li> <li>• Benutzerhandbuch</li> <li>• USB-Stick mit Quellcode</li> <li>• Dokumentation in Papierform</li> </ul>						
Voraussetzungen zum Start						
Realisierung abgeschlossen						
Erwartete Resultate						
Lieferobjekte sind bereit zur Abgabe						
Bemerkung						
Verantwortlich	PS	Mitwirkend	MR, CHG			

## 4.14 Meilensteine

Tabelle 62: Meilensteine. Quelle: Autor

Nr	Meilensteine	Ergebnisse
0	Kick-Off	<ul style="list-style-type: none"> <li>• Projektgruppe geformt</li> <li>• Projektthema festgelegt</li> </ul>
1	Initialisierung abgeschlossen	<ul style="list-style-type: none"> <li>• Dokumente erstellt</li> <li>• Projektorganisation aufgestellt</li> <li>• Projektauftrag formuliert           <ul style="list-style-type: none"> <li>◦ Ausgangslage</li> <li>◦ Projektidee</li> <li>◦ Projektziele</li> <li>◦ Anforderungen</li> <li>◦ Abgrenzung</li> <li>◦ Projektphasen</li> </ul> </li> <li>• Rollenverteilung</li> <li>• Meilensteine definiert</li> </ul>
2	Status-Meeting 1	<ul style="list-style-type: none"> <li>• Statusbericht 1</li> </ul>
3	Vorstudie abgeschlossen	<ul style="list-style-type: none"> <li>• Risiken</li> <li>• Arbeitspakete</li> <li>• Projektstrukturplan</li> </ul>
4	Konzeption abgeschlossen	<ul style="list-style-type: none"> <li>• Systemidee formuliert</li> <li>• Use Cases beschrieben</li> <li>• UserStory's abgeleitet</li> <li>• Mockups erstellt           <ul style="list-style-type: none"> <li>◦ Browser</li> <li>◦ Mobile</li> </ul> </li> <li>• Testkonzept erstellt</li> </ul>
5	Status-Meeting 2	<ul style="list-style-type: none"> <li>• Statusbericht II</li> </ul>
6	Realisierung abgeschlossen	<ul style="list-style-type: none"> <li>• Fachklassenmodell erstellt</li> <li>• Schnittstellenbeschreibung formuliert</li> <li>• Priorisierte UserStorys abgearbeitet</li> </ul>
7	Abgabe Projektarbeit	<ul style="list-style-type: none"> <li>• Installationsanleitung</li> <li>• Benutzerhandbuch</li> <li>• Testprotokoll</li> <li>• Code-Dokumentation           <ul style="list-style-type: none"> <li>◦ JavaDoc</li> <li>◦ AngularDoc</li> </ul> </li> <li>• Abschlussbericht ergänzt</li> <li>• Fertiggestellte Lieferobjekte</li> </ul>

## 4.15 Ganntdiagramm

Tabelle 63: Gantt Auflistung. Quelle: Autor

AP-Nr.	Schritt	Abhängigkeit	Resource	Start	Arbeitstage	Ende
	1 - Initialisierung			30.08.21	13	12.09.21
I-1	Dokumente aufsetzen		CHG	30.08.21	1	31.08.21
I-2	Projektorganisation aufstellen	Dokumente aufsetzen	CHG	30.08.21	3	02.09.21
I-3	Projektauftrag formulieren	Dokumente aufsetzen	PS	30.08.21	3	02.09.21
I-4	Lösungsansatz formulieren	Projektauftrag formulieren	MR	02.09.21	4	06.09.21
I-5	Meilensteine definieren	Lösungsansatz formulieren	CHG	07.09.21	2	09.09.21
I-6	Rollenverteilung	Projektorganisation aufstellen	CHG	10.09.21	1	11.09.21
	MS - Projekt initialisiert			12.09.21	0	12.09.21
	2 - Vorstudie	1 - Initialisierung		13.09.21	13	26.09.21
V-1	Risiken		MR	13.09.21	2	15.09.21
	MS - Statusbericht 1			16.09.21	0	16.09.21
V-2	Statusbericht 1		CHG	16.09.21	2	18.09.21
V-3	Nachbesprechung SM1	Statusbericht 1	PS, MR, CHG	18.09.21	1	19.09.21
V-4.1	Projektstrukturplan	Nachbesprechung SM1	CHG	19.09.21	3	22.09.21
V-4.2	Arbeitspakete	Nachbesprechung SM1	CHG	19.09.21	3	22.09.21
V-5.1	Personenverfügbarkeit	Nachbesprechung SM1	CHG	22.09.21	1	23.09.21
V-5.2	Bedarfsermittlung	Arbeitspakete	CHG	23.09.21	2	25.09.21
V-5.3	Terminplanung	Projektstrukturplan	CHG	23.09.21	2	25.09.21
V-6	Netzplan	Arbeitspakete	MR	24.09.21	2	26.09.21
	MS - Abschluss Vorstudie			27.09.21	0	27.09.21
	3 - Konzeption	2 - Vorstudie		27.09.21	27	24.10.21
K-1	Systemidee		CHG	27.09.21	1	28.09.21
K-2	Use Cases	Systemidee	CHG	29.09.21	1	30.09.21
K-3	User Stories	Use Cases	CHG	30.09.21	4	04.10.21
K-4.1	GUI Mockup (Desktop Browser)	Use Cases	MR	02.10.21	1	03.10.21
K-4.2	GUI Mockup (Mobile Browser)	Use Cases	PS	02.10.21	1	03.10.21
K-5	Softwarearchitektur (Backend, Frontend GUI Mockup (Desktop Brow:	PS	05.10.21	2	07.10.21	
K-6	Fachklassenmodell	Softwarearchitektur (Backer	PS	05.10.21	2	07.10.21
K-7	Softwarearchitektur (Backend): Java K	Fachklassenmodell	CHG	08.10.21	2	10.10.21
K-8	Schnittstellenbeschreibung	Softwarearchitektur (Backer	PS	08.10.21	2	10.10.21
K-9	Softwarearchitektur (Datenbank): Dat	Schnittstellenbeschreibung	CHG	11.10.21	2	13.10.21
K-10	Softwarearchitektur (Frontend)		MR	11.10.21	2	13.10.21
K-11	Diagramme erstellen	Softwarearchitektur (Fronte	MR	15.10.21	2	17.10.21
K-12	Testkonzept		MR	18.10.21	5	23.10.21
	MS - Software konzeptioniert			31.10.21	0	31.10.21
	4 - Realisierung	3 - Konzeption		25.10.21	41	05.12.21
R-1	GIT Projekt		PS	25.10.21	1	26.10.21
R-2	Aufsetzung Persistenz I		PS	25.10.21	1	26.10.21
R-3	Aufsetzung Persistenz II	Initialisierung Persistenz	PS	26.10.21	1	27.10.21
R-4	Aufsetzung Backend	GIT Projekt	PS	28.10.21	1	29.10.21
R-5	Aufsetzung Frontend	GIT Projekt	PS	30.10.21	1	31.10.21
R-6	Testen Grundlende Kommunikation Backend-Frontend-Datenban	CHG		01.11.21	1	02.11.21
R-7	Umsetzung User Stories	Testen Grundlende Kommur	PS, MR, CHG	03.11.21	32	05.12.21
R-8	Statusbericht 2			12.11.21	2	14.11.21
	MS - Statusbericht 2			13.11.21	0	13.11.21
R-9	Nachbesprechung SM2			15.11.21	1	16.11.21
	MS - Lauffähiges System			06.12.21	0	06.12.21
	5 - Abschluss	4 - Realisierung		06.12.21	12	18.12.21
A-1	Installationsanleitung	Umsetzung User Stories	PS	07.12.21	2	09.12.21
A-2	Benutzerhandbuch	Umsetzung User Stories	CHG	08.12.21	1	09.12.21
A-3	Testprotokoll überarbeiten	Umsetzung User Stories	MR	10.12.21	1	11.12.21
A-4	Code-Dokumentation	Umsetzung User Stories	PS	11.12.21	2	13.12.21
A-5	Architektur-Dokumentation	Umsetzung User Stories	PS	11.12.21	2	13.12.21
A-6	Abschlussbericht	Umsetzung User Stories	CHG	11.12.21	2	13.12.21
A-7	Lieferobjekte fertigstellen	Umsetzung User Stories		12.12.21	1	13.12.21
	Zeitplan-Puffer			18.12.21	5	23.12.21

Das nachfolgende Gantt Diagramm baut auf einer Vorlage von YouTube auf (Tatakaus, 2020).



Abbildung 12: Gantt Diagramm. Quelle: Autor

## 4.16 Projektnetzplan

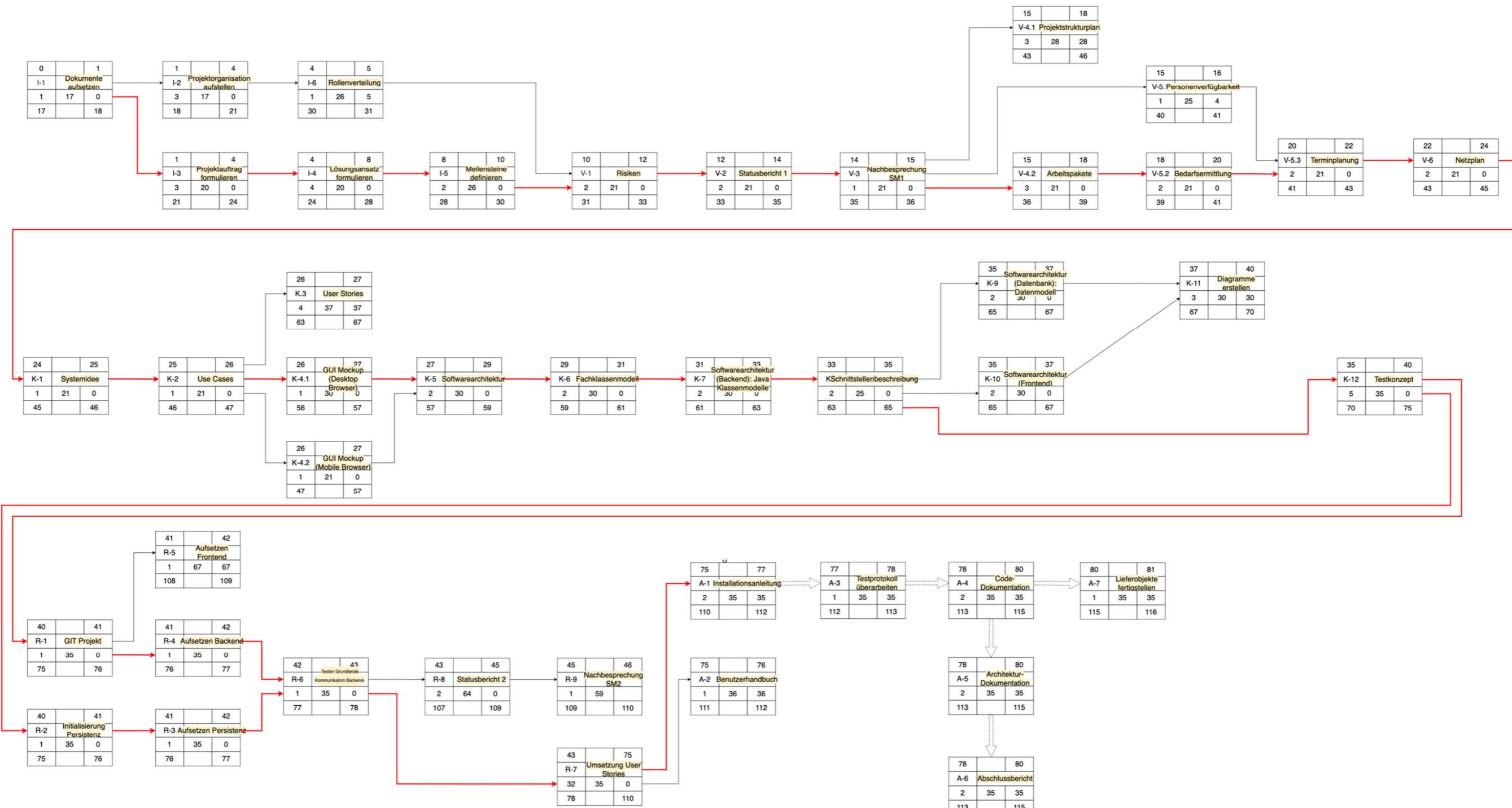


Abbildung 13: Projektnetzplan. Quelle: Autor

## 5 Projektüberwachung

Beim Controlling der Arbeits- und Projektebene geht es primär um die Termine, Kosten und Ergebnisse eines Einzelprojektes. Die Verantwortung liegt hierbei beim PL resp. TPL. Es bildet die Basis für das Controlling auf der Projektportfolio-ebene. Die Controlling-Instrumente auf dieser Ebene unterscheiden sich zwischen inneren Zyklus (wirken sich auf Projektarbeiten aus) wie und äusseren Zyklus (wirken sich auf Projektmanagement aus).

### 5.1 Abweichungen analysieren und korrigieren

Die Analyse und Korrektur der Abweichungsursachen sind essentiell, denn es verringert die Gefahr von Schnellschüssen. Es gibt verschiedene systematische Methoden (Ereignisbaumanalyse, Fehlerbaumanalyse, Ursachen-Wirkungs-Analyse, Ausschlussverfahren) wie auch Ad-Hoc-Methoden (Brainstorming) welche jedoch alle detaillierte Projekt- und Methodenkenntnisse voraussetzen. Nachfolgend werden die Massnahmen bei Abweichungen aufgelistet.

Tabelle 64: Massnahmen bei Abweichungen. Quelle: Autor

Problem	Mögliche Gegenmassnahme	Mögliche Konsequenz
Terminabweichung	<ul style="list-style-type: none"> <li>• Zusätzliche Ressourcen einsetzen</li> <li>• Auslastung bestehender Ressourcen erweitern (Überstunden leisten)</li> <li>• Funktionsumfang reduzieren</li> <li>• Termine verschieben</li> </ul>	<ul style="list-style-type: none"> <li>• Budgetüberschreitung</li> <li>• Abnehmende Qualität</li> <li>• Negative Auswirkung auf Motivation</li> </ul>
Budgetüberschreitung	<ul style="list-style-type: none"> <li>• Aufgaben / Leistungen vereinfachen oder weglassen</li> <li>• Bestehende Lösungen / Komponenten verwenden</li> <li>• Gezielt auf Perfektionismus verzichten</li> </ul>	<ul style="list-style-type: none"> <li>• Terminabweichung oder abnehmende Qualität</li> </ul>
Interpersonelle Konflikte	<ul style="list-style-type: none"> <li>• Moderator einsetzen</li> <li>• Fischgräte Diagramm</li> </ul>	<ul style="list-style-type: none"> <li>• Austritt eines Projektmitglieds</li> </ul>
Georderte Funktionalität nicht erreichbar	<ul style="list-style-type: none"> <li>• Spezialisten hinzuziehen</li> <li>• Alternativen suchen</li> </ul>	<ul style="list-style-type: none"> <li>• Budgetüberschreitung</li> </ul>
Kapazität reicht nicht aus	<ul style="list-style-type: none"> <li>• Prioritätenänderungen von weniger dringlichen Aufgaben</li> <li>• Überstunden leisten</li> <li>• Effizienz erhöhen z.B. durch Motivation der Mitarbeiter</li> </ul>	<ul style="list-style-type: none"> <li>• Budgetüberschreitung</li> </ul>

### 5.2 Kontrollprotokoll Arbeitspakete

Die Arbeitspakete werden von allen Projektmitgliedern umgesetzt, sobald sie Zeit haben. Wer wann Zeit hat ist im Ressourcenplan zu entnehmen.

Der in der unteren Liste als Verantwortlicher definierte ist dafür verantwortlich, dass einer der beiden Kontrolleure diese Arbeitspakete auf die Erledigung kontrollieren und dies mit ihrem Kürzel und Datum versehen.

Tabelle 65: Kontrollprotokoll der Arbeitspakete. Quelle: Autor

Arbeitspaket	Verantwortlicher	Kontrolle	Erledigt Kürzel	SOLL	IST	Datum
I-1	CHG	MR, PS	MR	6	4	30.08.2021
I-2	MR	CHG, PS	PS	15	15	02.09.2021
I-3	PS	CHG, MR	MR	20	25	02.09.2021
I-4	PS	CHG, MR	MR	30	20	06.09.2021
I-5	PS	CHG, MR	MR	8	5	06.09.2021
I-6	MR	CHG, PS	CHG	3	3	11.09.2021
V-1	MR	CHG, PS	CHG	8	12	15.09.2021
V-2	CHG	MR, PS	MR	4	6	16.09.2021
V-3	CHG	MR, PS	PS	2	2	18.09.2021
V-4.1	CHG	MR, PS	MR	10	6	25.09.2021
V-4.2	PS	CHG, MR	MR	14	22	24.09.2021
V-5.1	CHG	MR, PS	PS	4	2	22.09.2021
V-5.2	CHG	MR, PS	MR	5	3	25.09.2021
V-5.3	CHG	MR, PS	MR	5	4	28.09.2021
V-6	MR	CHG, PS	CHG	15	14	27.09.2021
K-1	CHG	MR, PS	MR	4	5	28.09.2021
K-2	PS	CHG, MR	MR	3	4	30.09.2021
K-3	MR	CHG, PS	PS	16	15	02.10.2021
K-4.1	MR	CHG, PS	CHG	4	3	04.10.2021
K-4.2	PS	CHG, MR	PS	4	2	02.10.2021
K-5	PS	CHG, MR	MR	3	2	05.10.2021
K-6	PS	CHG, MR	MR	4	4	07.10.2021
K-7	PS	CHG, MR	MR	8	10	10.10.2021
K-8	PS	CHG, MR	CHG	10	14	11.10.2021
K-9	CHG	MR, PS	PS	8	10	15.10.2021
K-10	MR	CHG, PS	MR	4	3	13.10.2021
K-11	MR	CHG, PS	MR	8	8	16.10.2021
K-12	MR	CHG, PS	CHG	20	24	22.10.2021
R-1	CHG	MR, PS	MR	1	1	25.10.2021
R-2	CHG	MR, PS	PS	2	2	26.10.2021
R-3	PS	CHG, MR	MR	8	10	28.10.2021
R-4	MR	CHG, PS	PS	6	6	29.10.2021
R-5	CHG	MR, PS	PS	4	4	31.10.2021
R-6	CHG	MR, PS	MR	4	5	03.11.2021
R-7	PS	CHG, MR	MR	100	118	05.12.2021
R-8	CHG	MR, PS	CHG	6	8	13.11.2021
R-9	CHG	MR, PS	CHG	2	2	15.11.2021
A-1	PS	CHG, MR	PS	4	6	07.12.2021
A-2	CHG	MR, PS	CHG	4	4	08.12.2021
A-3	PS	CHG, MR	PS	6	1	10.12.2021
A-4	MR	CHG, PS	MR	8	7	13.12.2021
A-5	PS	MR, PS	PS	10	8	13.12.2021
A-6	CHG	MR, PS	CHG	10	12	13.12.2021
A-7	PS	CHG, MR	MR	12	25	17.12.2021
<b>Total</b>				<b>432</b>	<b>466</b>	

## 5.3 Kontrollprotokoll User Stories

Die User-Stories werden von allen Projektmitgliedern umgesetzt, sobald sie Zeit haben. Wer wann Zeit hat ist im Ressourcenplan zu entnehmen.

Der in der unteren Liste als Verantwortlicher definierte ist dafür verantwortlich, dass einer der beiden Kontrolleure diese User-Story auf die Erledigung kontrollieren und dies mit ihrem Kürzel und Datum versehen.

Tabelle 66: Kontrollprotokoll der User Stories. Quelle: Autor

User Story	Verantwortlicher	Kontrolle	SOLL	IST	Erledigt Kürzel	Datum
<b>U-1</b>	PS	MR	8	8	MR	05.11.2021
<b>U-2</b>	MR	PS	8	9	PS	05.11.2021
<b>U-3</b>	CHG	PS	2	2	PS	06.11.2021
<b>U-4</b>	PS	CHG	8	7	CHG	09.11.2021
<b>U-4 – Nachtrag</b>	PS	MR	2	2	MR	01.12.2021
<b>U-5</b>	PS	MR	4	3	MR	10.11.2021
<b>U-6</b>	PS	MR	8	7	MR	14.11.2021
<b>U-7</b>	PS	MR	8	9	MR	21.11.2021
<b>U-8</b>	MR	CHG	4	6	CHG	22.11.2021
<b>U-9</b>	PS	CHG	3	14	CHG	25.11.2021
<b>U-10</b>	MR	CHG	3	4	CHG	25.11.2021
<b>U-11</b>	PS	CHG	4	4	CHG	26.11.2021
<b>U-12</b>	PS	CHG	3	1	CHG	26.11.2021
<b>U-13</b>	PS	CHG	8	13	CHG	20.11.2021
<b>U-14</b>	MR	PS	8	7	PS	01.12.2021
<b>U-15</b>	PS	CHG	15	18	CHG	05.12.2021
<b>U-16</b>	CHG	PS	4	4	PS	01.12.2021
		<b>Total</b>	<b>100</b>	<b>118</b>		

## 5.4 Messbare Lieferobjekte

- Projektantrag
- Projektstrukturplan
- Konzeption
- Testkonzept
- Testprotokolle
- Javadoc
- Installationsanleitung
- Benutzerhandbuch
- USB-Stick mit Quellcode
- Dokumentation in Papierform

## 6 Analysedokumente

Auf Basis der Anforderungen und der Systemidee werden Use Cases erstellt. Auf Basis der Use Cases werden Mockups erstellt. Auf Basis der Use Cases und den Mockups werden messbare Anforderungen in Form von User Stories erstellt. Diese setzt sich aus der Anforderung aus User-Sicht, den Akzeptanzkriterien, der Diskussion und der Priorität zusammen. Die Diskussion dient primär dazu, Gedanken und vorhersehbare Herausforderungen zu notieren, welche bereits bei der Erstellung der Story bekannt sind. Nachfolgende Abbildung visualisiert das Vorgehen.



Abbildung 14: Von den Anforderungen zu User Stories. Quelle: Autor

### 6.1 Systemidee

Die Software soll Studierende der FFHS dabei unterstützen, das Studium optimal zu planen. Im Moment (Stand: Herbst 2021) wird die Modulevaluation von den Studierenden händisch auf Basis von verschiedenen Listen (Liste mit den Relevanzpunkten, Zeitplan der Vertiefungs- und Wahlmodule, Beschreibung der Module) getätig. Der Mehrwert unserer Software soll die einfache Planung und bessere Übersicht der Modulplanung sein. Dazu wird eine Darstellung erarbeitet, in welcher die Module auf Basis von Parametern (Studienort, Vertiefungsrichtung) ermöglicht. Die Darstellung orientiert sich dabei am Curriculum, welches durch den Studiengangleiter BSc Informatik, Oliver Ittig, an die Studierenden versendet wird. Damit die Software allen Studierenden unabhängig der eingesetzten Software/Hardware geräteübergreifend zur Verfügung steht, wird diese als Webanwendung entworfen. Auf Grund der Vorgabe Java als Hauptsprache einzusetzen, wird im Backend Java eingesetzt. Die Kommunikation zwischen der Web-Sicht (Frontend) und Backend erfolgt über definierte Schnittstellen (API) mittels HTTP-Requests.

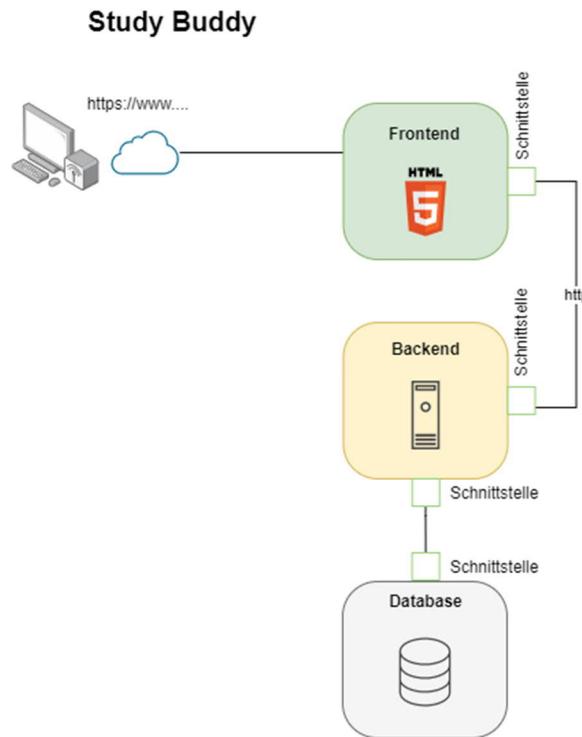


Abbildung 15: Systemidee, Autor: Eigene Abbildung

## 6.2 Use Case

### 6.2.1 Haupt Use Case - Vogelperspektive

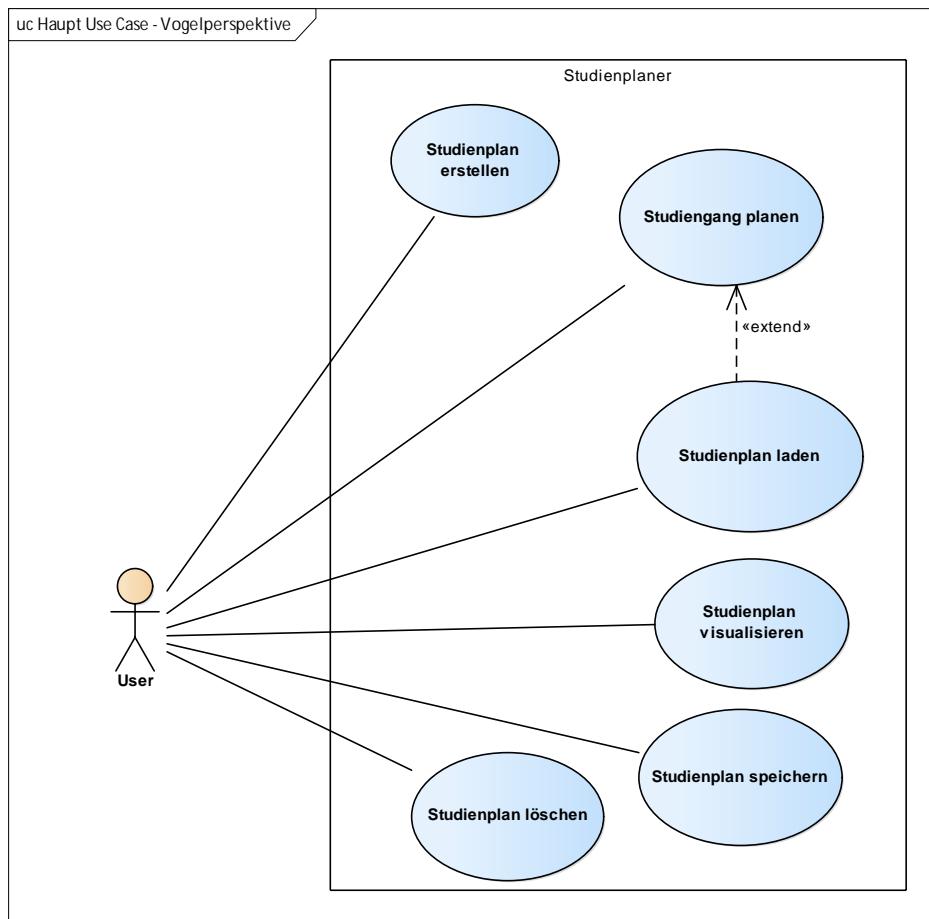


Abbildung 16: Haupt Use Case (Vogelperspektive) welches die Aktivitäten des Studienplaners beinhaltet. Quelle: Autor

Der Use Case «Studiengang editieren» erweitert den Use Case «Studiengang planen» um die Möglichkeit, einen Studiengang mittels einer Kennung (Name des Plans) aufzurufen, um diesen editieren zu können. Folgende Use Cases werden noch in weitere Schichten (Ebenen) tiefer aufgeteilt, da diese noch weitere Anwendungsfälle beinhalten:

- Studiengang planen
- Studienplan laden

Diese Use Cases werden in den nachfolgenden Kapiteln weiter spezifiziert.

## 6.2.2 Studiengang planen

Damit ein User ein Studiengang planen kann, müssen verschiedene Tätigkeiten zur Verfügung stehen:

- Studienplanung anzeigen
- Modul eintragen
- Modul entfernen
- Relevanzpunkte sichten
- ECTS-Punkte sichten
- Modul Informationen sichten

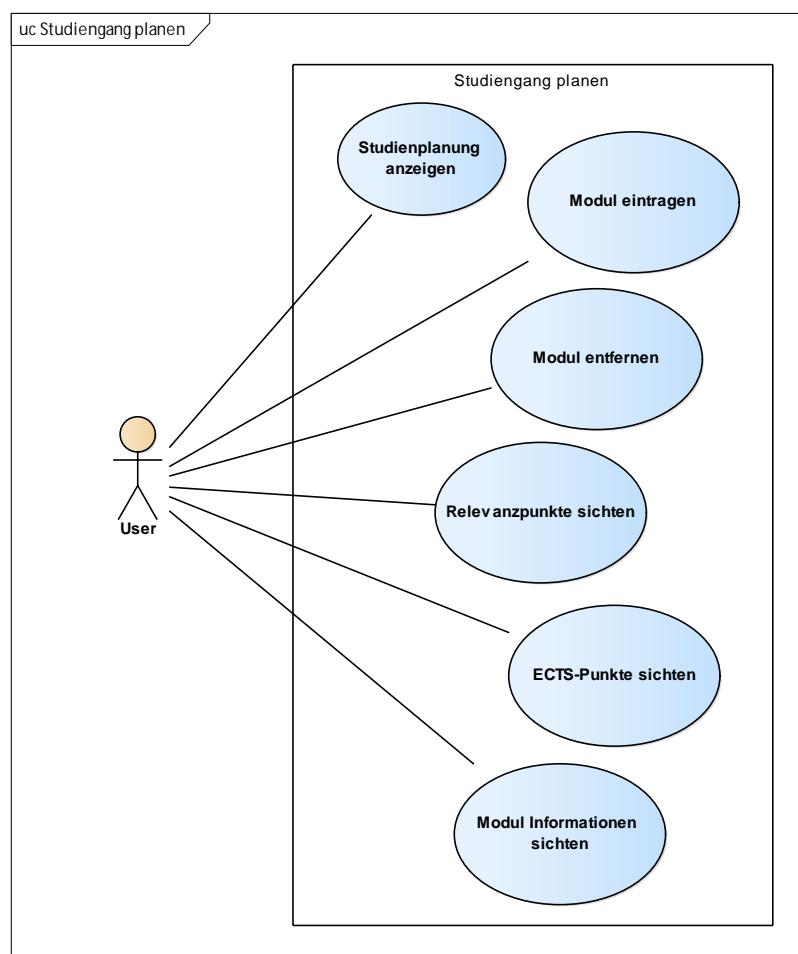


Abbildung 17: Use Case "Studiengang planen". Quelle: Autor

### 6.2.3 Modul Informationen sichten

Der User möchte zu den zur Verfügung stehenden Modulen Informationen sichten können:  
Die Relevanzpunkte pro Vertiefungsrichtung (Module ergeben in verschiedenen Vertiefungsrichtungen unterschiedliche Relevanzpunkte) und wann die Module zur Verfügung stehen (Semesterplan sichten).

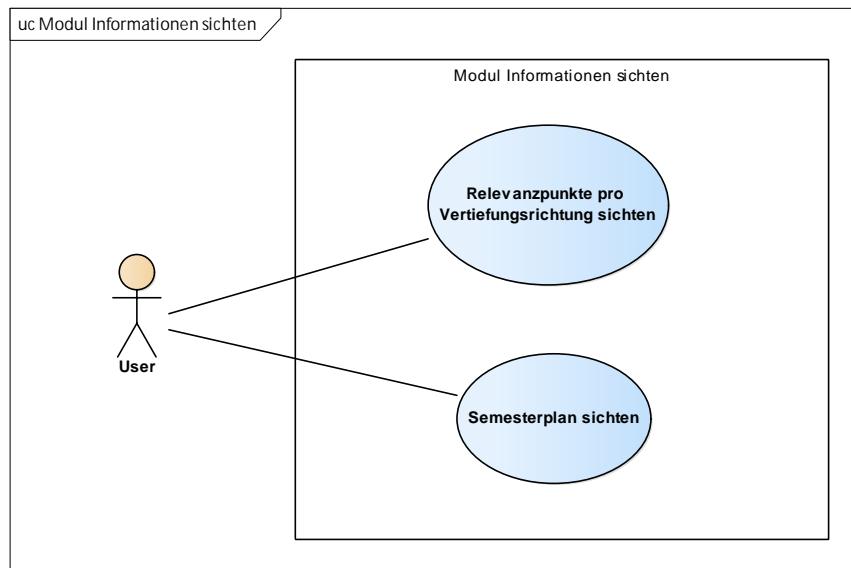


Abbildung 18: Use Case "Modul Informationen sichten". Quelle: Autor

### 6.2.4 Studienplan laden

Der Anwendungsfall "Studienplan laden" besteht aus den zwei Aktionen: "Studienplan mittels Namen laden" und "Studiengang planen". Bei "Studiengang planen" handelt es sich um den Use Case, welcher unter 6.2.2 beschrieben ist.

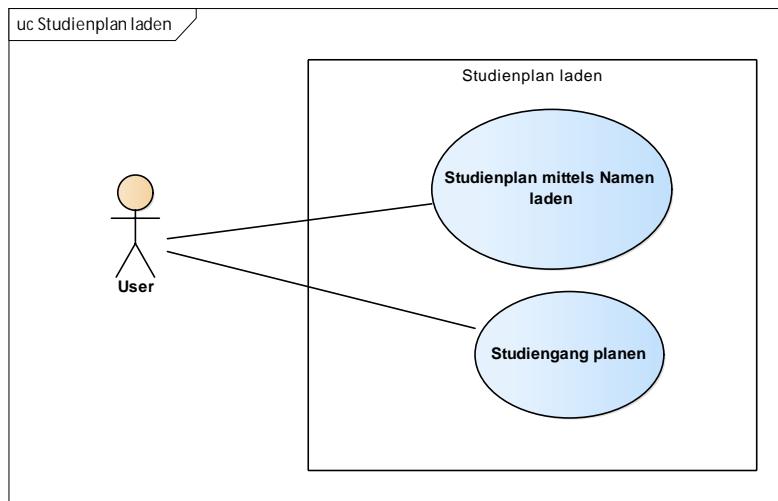


Abbildung 19: Use Case "Studiengang editieren". Quelle: Autor

## 6.2.5 Modul eintragen

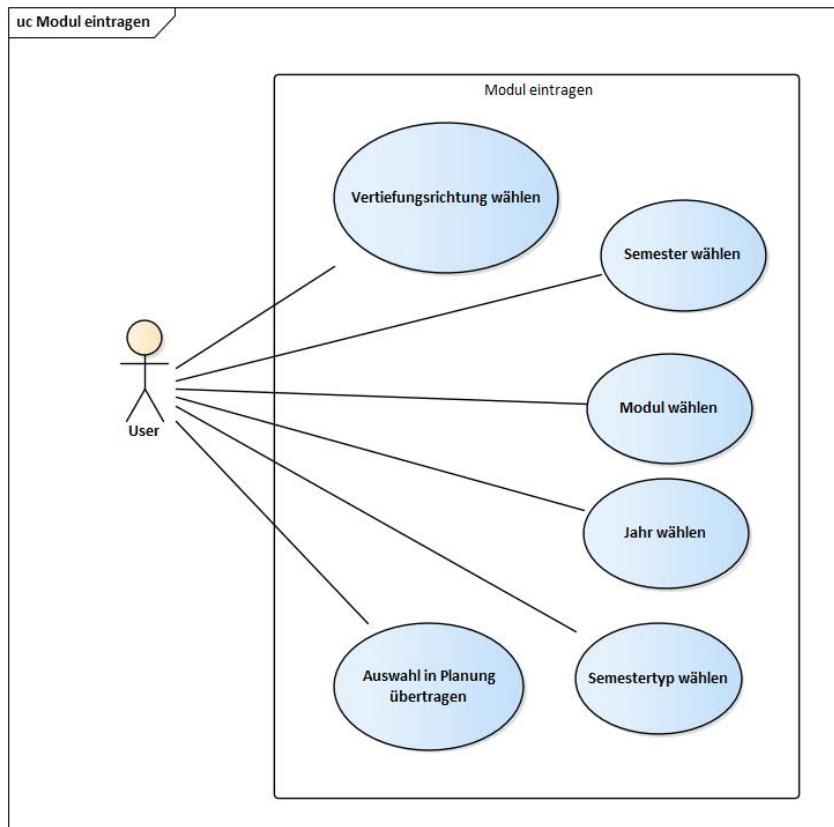


Abbildung 20: Use Case "Modul eintragen". Quelle: Autor

## 6.3 Mockup und Aufbau der Webseite

Das Frontend wird in nachfolgende Seiten aufgeteilt:

Tabelle 67: URLs der Systemlösung (UI). Quelle: Autor

Seite	Pfad
<b>Home</b>	http://url <sup>2</sup>
<b>Studienplan erstellen</b>	http://url/studyplan/add
<b>Studienplan laden</b>	http://url/studyplan/edit?name=studienplanName
<b>Studienplan editieren/löschen</b>	http://url/studyplan/edit?name=studienplanName
<b>Studiengang planen</b>	http://url/studyplan/edit?name=studienplanName
<b>Module hinzufügen</b>	http://url/studyplan/edit?name=studienplanName
<b>Impressum</b>	http://url/impressum

Es gibt also zwei wichtige URL's für die Erstellung und Planung des Studienplans. Die URL <http://url/studyplan/add> führt zur Maske zur Erstellung eines neuen Studienplans. Die URL <http://url/studyplan/edit> führt zur Maske zur Planung, Editierung, Visualisierung, Löschung des Studienplans. Außerdem werden in dieser Maske die Module zum Studienplan hinzugefügt respektive gelöscht.

<sup>2</sup> url ist ein Platzhalter. Im lokalen Betrieb ist die url "localhost:4200"

### 6.3.1 Home

Nachfolgend das Mockup Startseite auf einem Desktop-Browser. Die Startseite enthält die Navigations-Bar, ein Studienplan laden Formular, einen Begrüßungstext und den Footer (Angaben zu den Autoren).

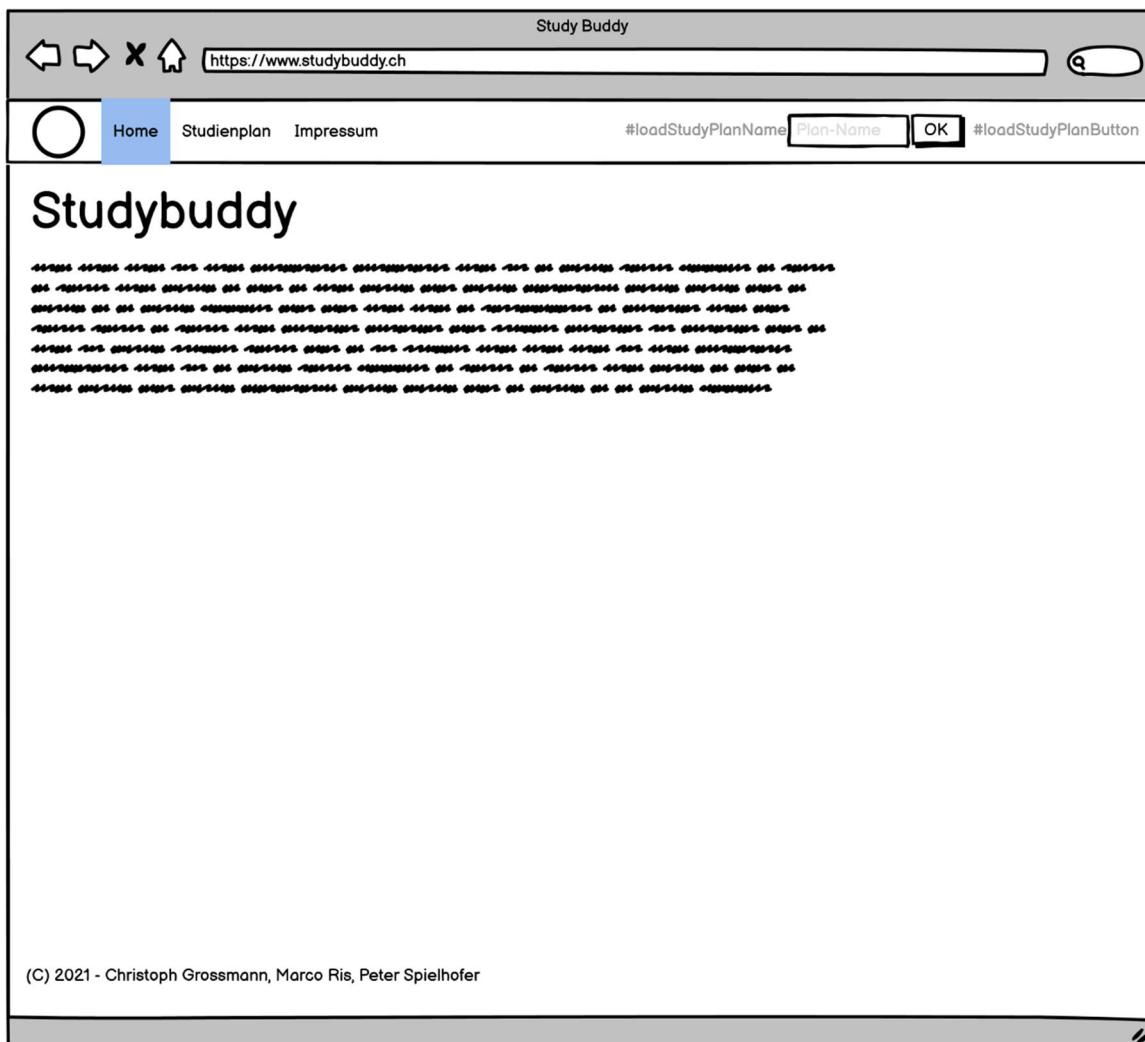


Abbildung 21: Mockup Home (Startseite). Quelle: Autor

Die Darstellung auf Mobil-Geräten soll wie das nachfolgende Mockup dargestellt werden. Die Navigations-Liste wird, wie bei Bootstrap üblich, untereinander, statt nebeneinander dargestellt. Ansonsten enthält es dieselben Elemente wie bei der Darstellung auf dem Desktop.

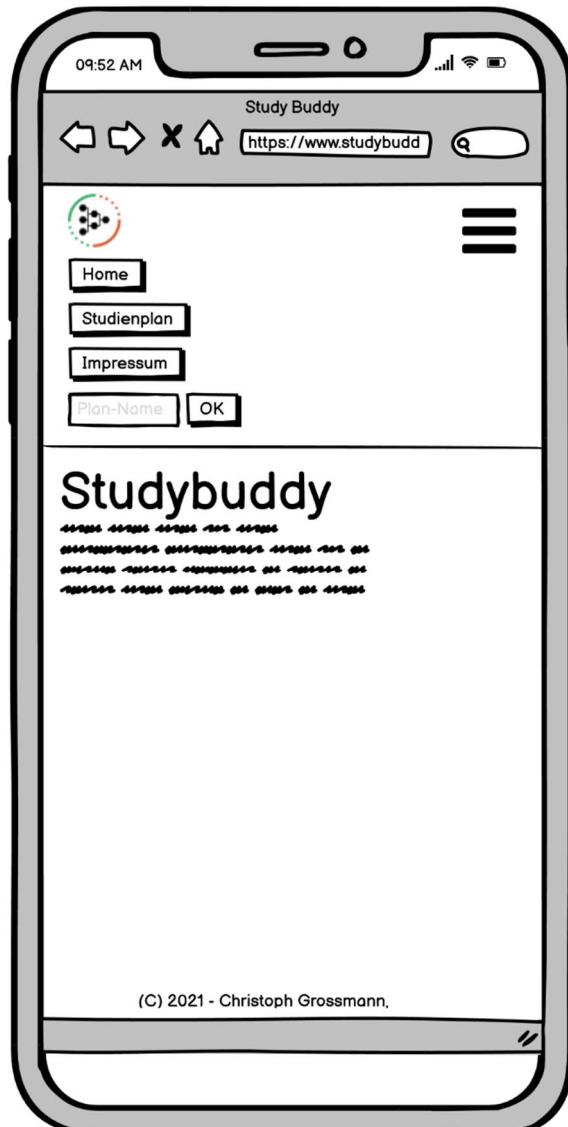


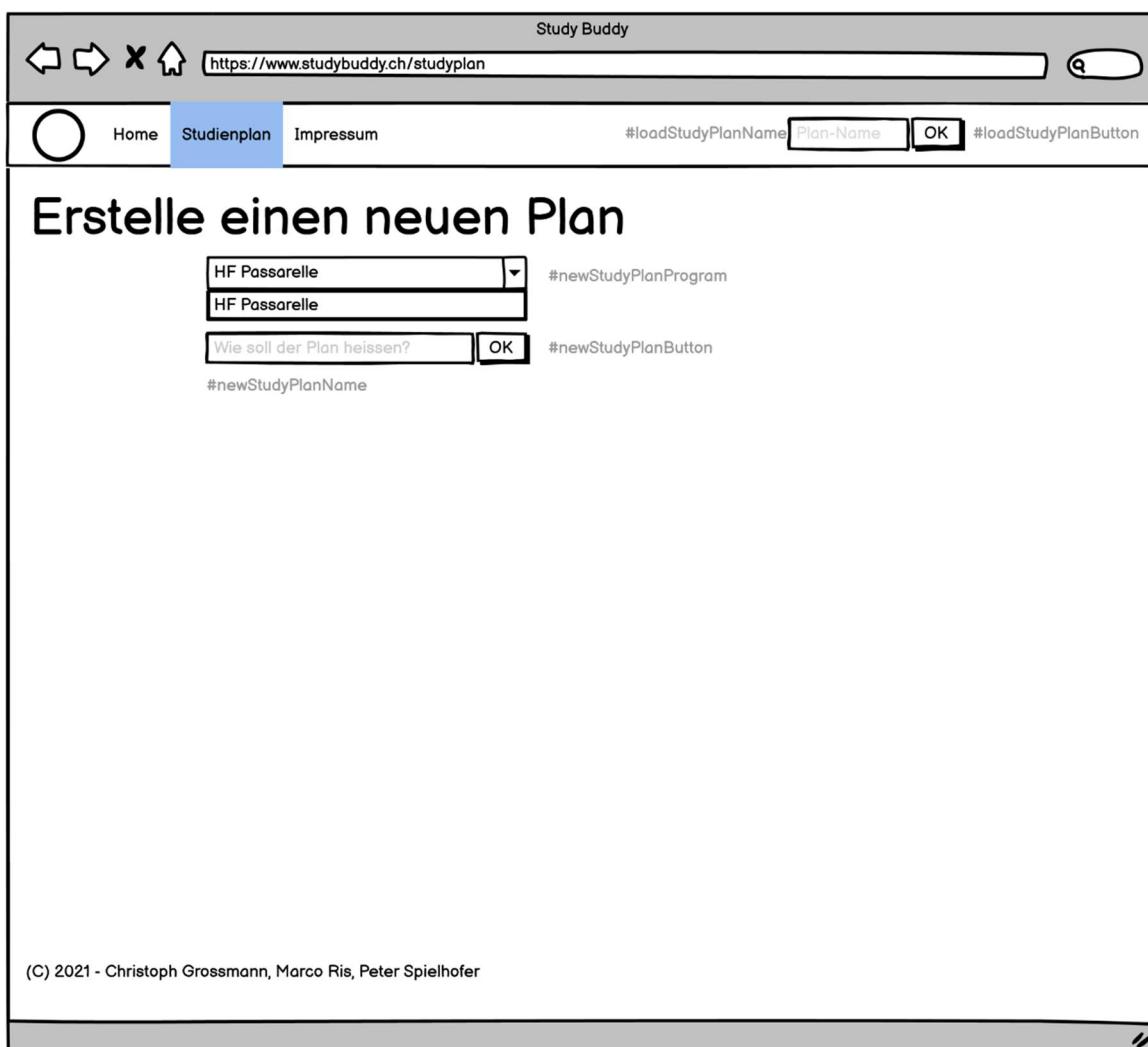
Abbildung 22: Mockup Home (Startseite) auf Mobilgerät. Quelle: Autor

Tabelle 68: Elementbeschreibung und Typ vom Mockup Home. Quelle: Autor

Element	Beschreibung	Typ
keine	Hier wird nur Text angezeigt.	String

### 6.3.2 Studienplan erstellen

Nachfolgend das Mockup für die Erstellung eines neuen Studienplans auf einem Desktop-Browser. Die Seite enthält die Navigations-Bar, ein Formular zur Erstellung des neuen Studienplans und den Footer (Angaben zu den Autoren).



The mockup shows a web browser window titled "Study Buddy". The address bar contains the URL "https://www.studybuddy.ch/studyplan". The navigation bar includes icons for back, forward, stop, and home, along with a search bar. Below the navigation bar is a horizontal menu with three items: "Home" (unselected), "Studienplan" (selected and highlighted in blue), and "Impressum". To the right of the menu are two buttons: "#loadStudyPlanName" and "OK", followed by "#loadStudyPlanButton". The main content area is titled "Erstelle einen neuen Plan". It features a dropdown menu set to "HF Passarelle" with the label "#newStudyPlanProgram" next to it. Below the dropdown is a text input field containing "HF Passarelle" with the label "#newStudyPlanName" next to it. At the bottom of this section is an "OK" button with the label "#newStudyPlanButton" next to it. At the very bottom of the page, within the browser's scrollable area, is a footer with the text "(C) 2021 - Christoph Grossmann, Marco Ris, Peter Spielhofer".

Abbildung 23: Mockup Neuen Studienplan erstellen. Quelle: Autor

Die Darstellung auf Mobil-Geräten soll wie das nachfolgende Mockup dargestellt werden. Die Navigations-Liste wird, wie bei Bootstrap üblich, untereinander, statt nebeneinander dargestellt. Ansonsten enthält es dieselben Elemente wie bei der Darstellung auf dem Desktop.

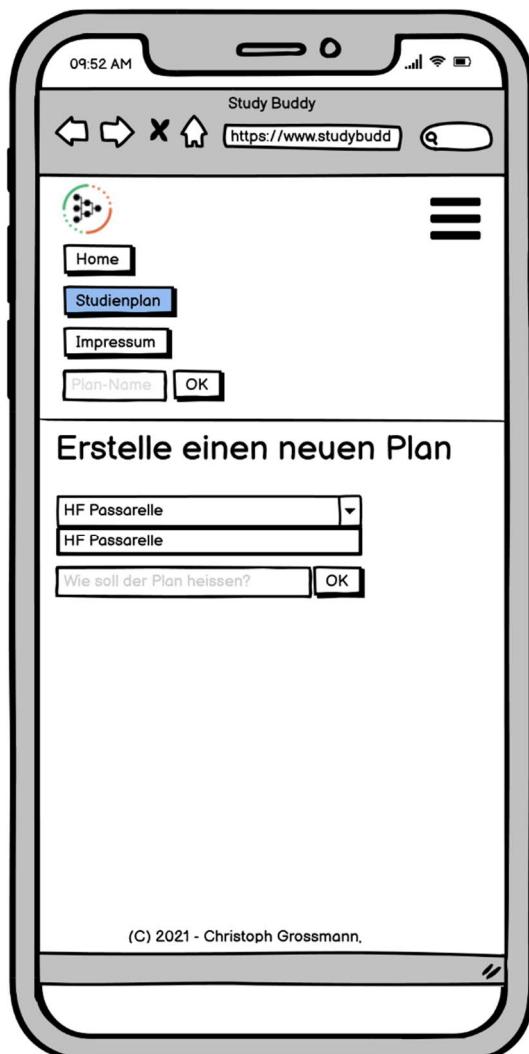


Abbildung 24: Mockup Neuen Studienplan erstellen auf Mobilgerät. Quelle: Autor

Tabelle 69: Elementbeschreibung und Typ vom Mockup Neuen Studienplan erstellen. Quelle: Autor

Element	Beschreibung	Typ
#newStudyPlanProgram	Auswahl des Einstieges (hier HF Passerelle).	Dropdown
#newStudyPlanName	Name des Studienplans. Placeholder: "Wie soll der Plan heißen?".	Input Feld (String)
#newStudyPlanButton	Dieser Button erstellt einen neuen Studienplan. Dazu ruft er eine Methode auf, welche die Daten persistiert.	Button

### 6.3.3 Studienplan laden

Ein bestehender Studienplan soll jederzeit in der Navigations-Leiste geladen werden können. Dazu wird in der Navigationsleiste ein Eingabe-Feld platziert, in welchem der Name des Studienplans eingegeben und mittels OK Button geladen werden kann.

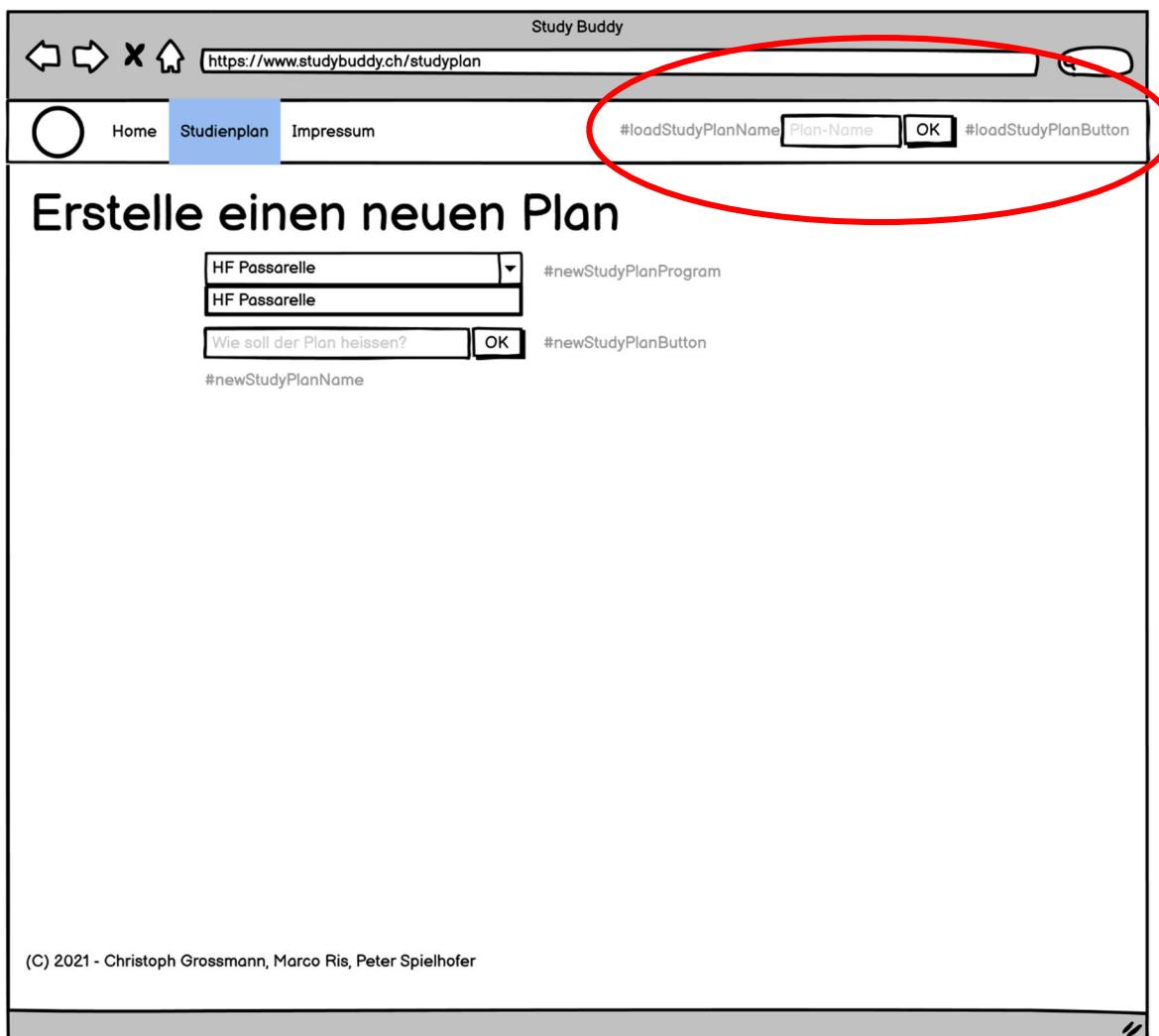


Abbildung 25: Mockup bestehenden Studienplan laden. Quelle: Autor

Auf den Mobilgeräten soll derselbe Mechanismus, zum Laden eines bestehenden Studienplans erfolgen. Das Eingabefeld, für den Namens des zu ladenden Studienplans, wird auch in der Navigationsleiste positioniert.

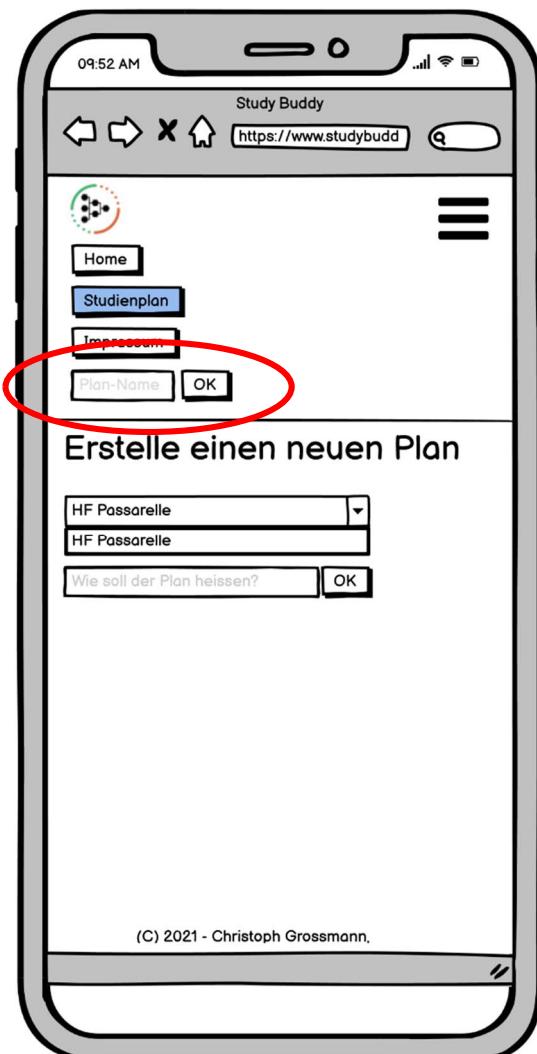


Abbildung 26: Elementbeschreibung und Typ vom Mockup Studienplan laden auf Mobilgeräten. Quelle: Autor

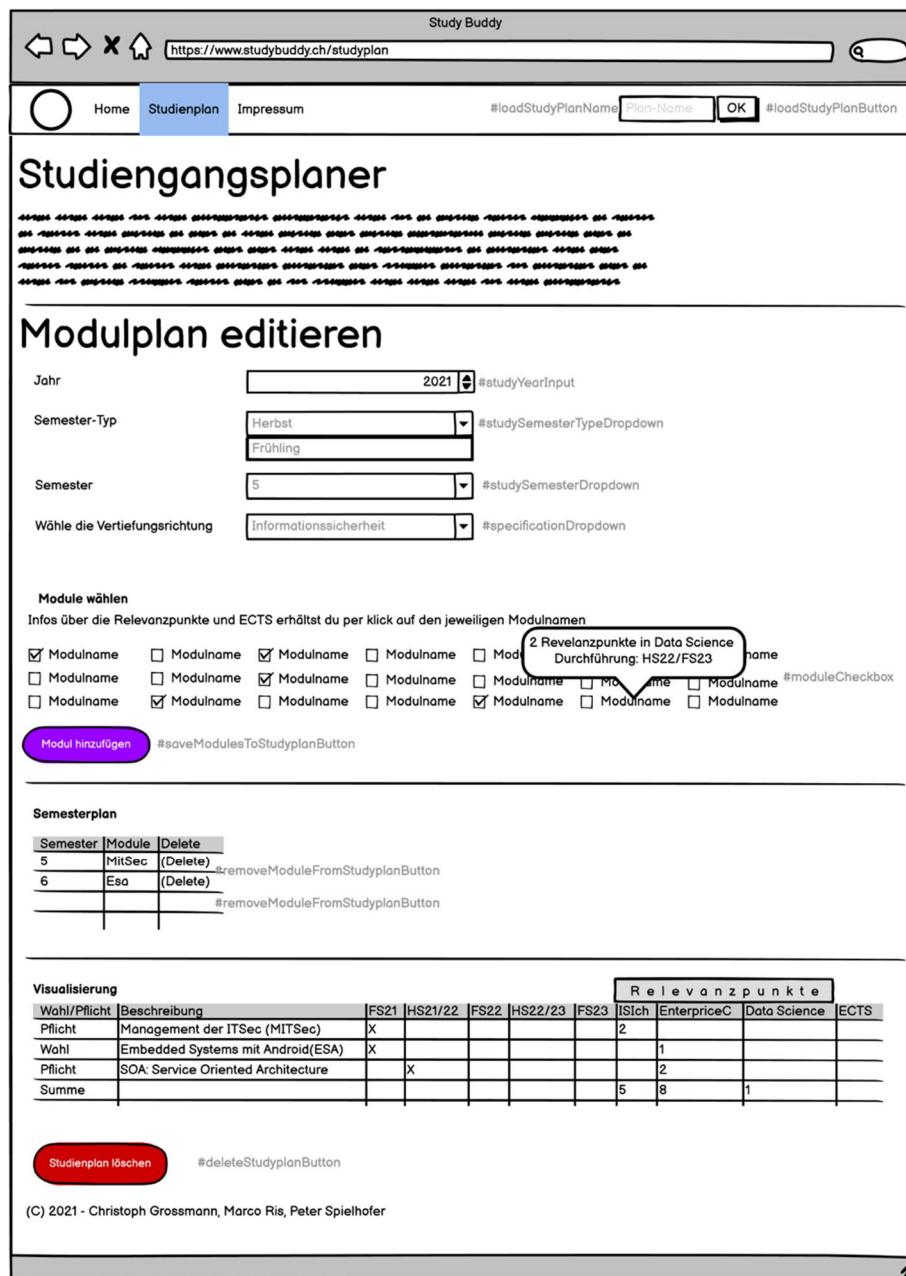
Tabelle 70: Elementbeschreibung und Typ vom Mockup Studienplan laden. Quelle: Autor

Element	Beschreibung	Typ
#loadStudyPlanName	Hier wird der Plan-Name eingegeben, welcher geladen werden soll. Placeholder: "Plan-Name".	Input Feld (String)
#loadStudyPlanButton	Dieser Button ruft die Methode auf, in der der Studienplan-Name im System gesucht wird und den entsprechenden Studienplan lädt, oder eine Fehlermeldung ausgibt, falls es den Studienplan nicht gibt.	Button

### 6.3.4 Studiengang planen, Module hinzufügen, Studienplan visualisieren, Studienplan löschen

Das nachfolgende Mockup zeigt alle Aktionen, welche auf einem Studienplan getätigten werden können:

- Studiengang planen (Sprich: Modul zum Studienplan hinzufügen und Modul entfernen)
- Relevanz- und ECTS-Punkte sichten (Studienplan visualisieren)
- Studienplan löschen



The screenshot shows the 'Modulplan editieren' (Module Plan Edit) section of the Study Buddy application. It includes fields for 'Jahr' (Year) set to 2021, 'Semester-Typ' (Semester Type) dropdown with 'Herbst' selected, 'Semester' dropdown with '5' selected, and 'Wähle die Vertiefungsrichtung' (Choose Specialization) dropdown with 'Informationssicherheit' selected. Below these are sections for 'Module wählen' (Select Modules) and 'Semesterplan' (Semester Plan). The 'Module wählen' section contains a table of modules with checkboxes for selecting specific ones. The 'Semesterplan' section shows a table of modules assigned to semester 5. At the bottom, there is a 'Visualisierung' (Visualization) table showing module details and their relevance points across different semesters. A red button labeled 'Studiengang löschen' (Delete Study Plan) is visible at the bottom left.

Wahl/Pflicht	Beschreibung	Relevanzpunkte								ECTS
		FS21	HS21/22	FS22	HS22/23	FS23	ISICh	EnterpriseC	Data Science	
Pflicht	Management der ITSec (MITSec)	X				2				
Wahl	Embedded Systems mit Android(ESA)	X				1				
Pflicht	SOA: Service Oriented Architecture		X			2				
Summe						5	8	1		

Abbildung 27: Studiengang planen. Quelle: Autor

Da auf den mobilen Geräten weniger Platz in der Breite zur Verfügung stehen, müssen einige Elemente (Tabellen) anders dargestellt werden. Es ist zu prüfen, ob dies mittels CSS oder speziellen HTML-Elementen umgesetzt werden kann. Die geplante Visualisierung auf mobilen Geräten sollte wie folgt aussehen (es sind mehrere Mockups da die Darstellung ansonsten zu klein ist):

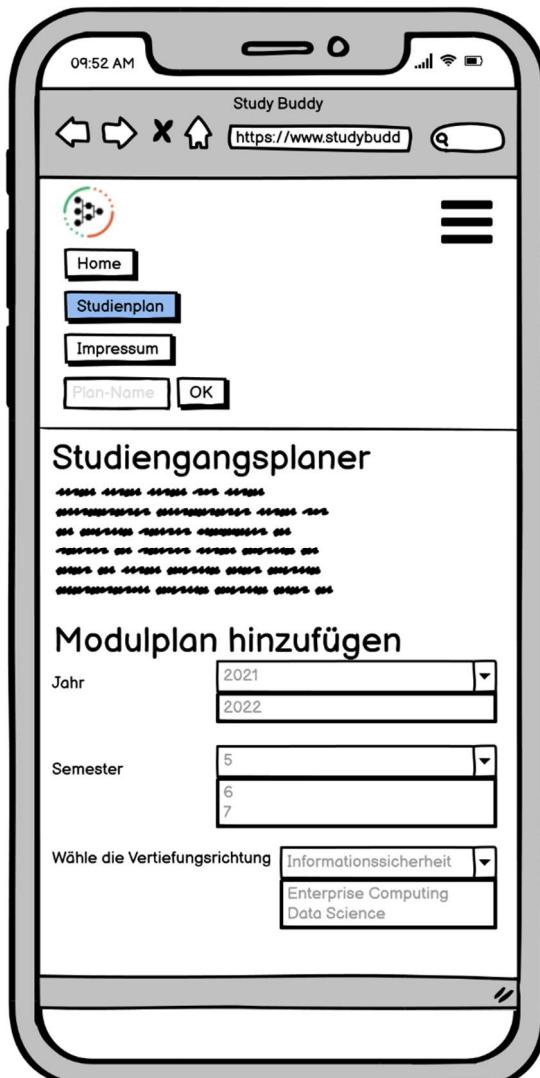


Abbildung 28: Studiengang planen (1) auf Mobilgeräten. Quelle: Autor

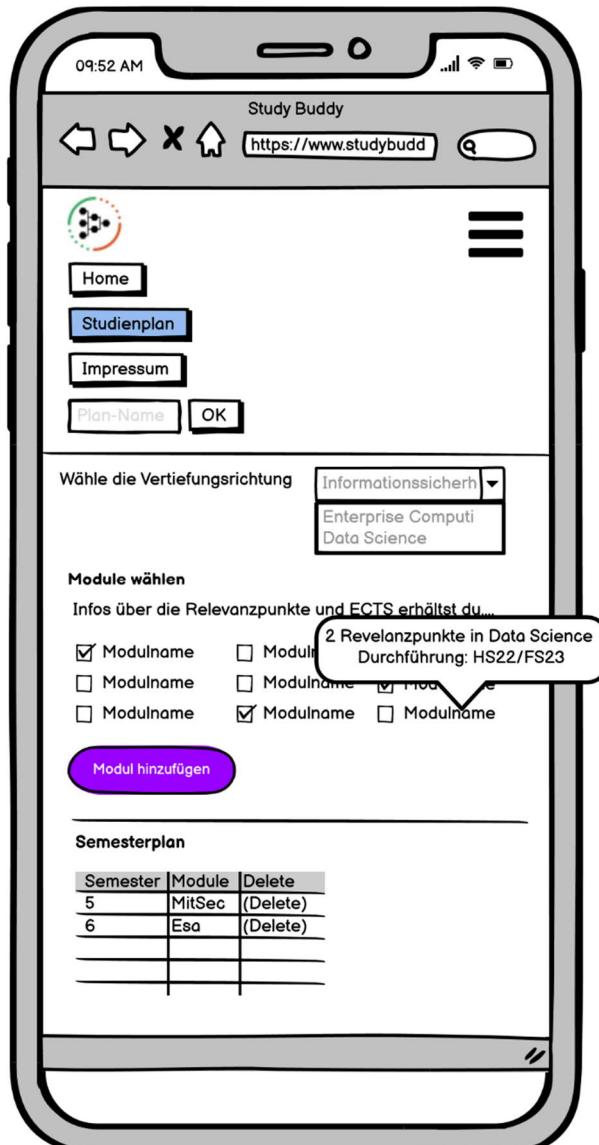


Abbildung 29: Studiengang planen (2) auf Mobilgeräten. Quelle: Autor

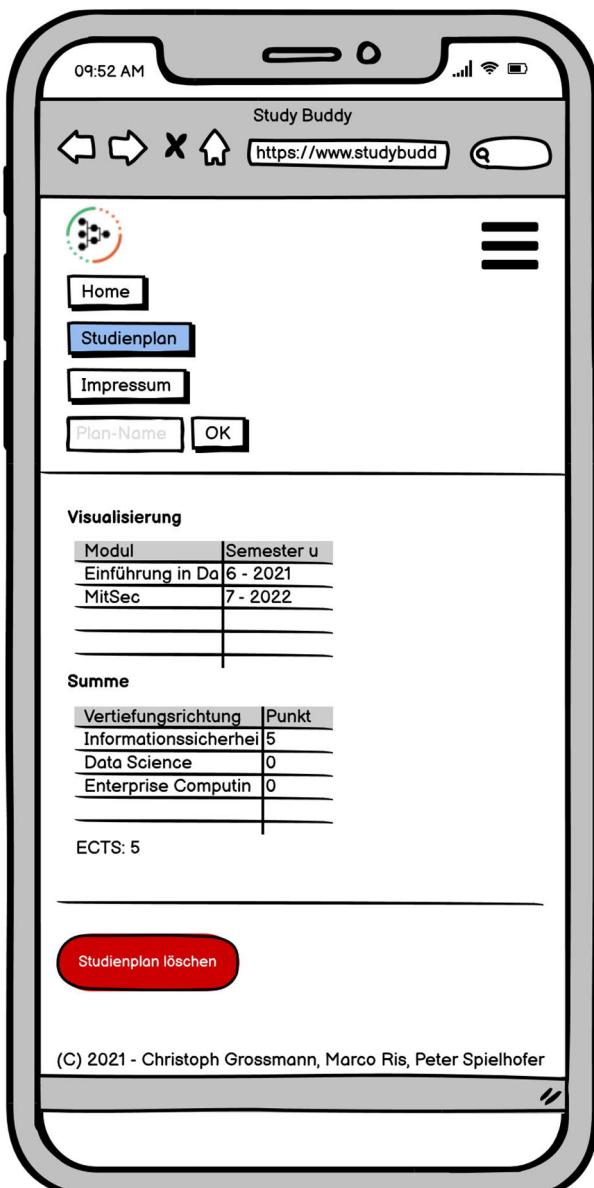


Abbildung 30: Studiengang planen (3) auf Mobilgeräten. Quelle: Autor

Tabelle 71: Elementbeschreibung und Typ vom Mockup Module hinzufügen. Quelle: Autor

Element	Beschreibung	Typ
#studyYearInput	Mittels Eingabe in ein Textfeld kann das Studienjahr gewählt werden.	Input
#studySemesterTypeDropdown	Mittels dieses Dropdowns kann der Semester-Typ gewählt werden (Herbst oder Frühling).	Dropdown
#studySemesterDropdown	Mittels Dropdown kann das Semester gewählt werden.	Dropdown
#specificationDropdown	Mittels Dropdown kann die Viertiefungsrichtung gewählt werden.	Dropdown
#moduleCheckbox	Mittels Checkboxen werden die Module für das Semester ausgewählt.	Checkbox
#saveModulesToStudyplanButton	Dieser Button persistiert den Studienplan mit den Modulen.	Button
#removeModuleFromStudyPlanButton	Mittels Button kann das Modul aus der Studienplanung entfernt werden.	Button
#deleteStudyplanButton	Dieser Button fragt den Benutzer mittels einem PopUp ob der Studienplan gelöscht werden soll. Mit Klick auf OK/Ja wird der Studienplan gelöscht.	Button

### 6.3.5 Impressum

Die Impressum-Seite enthält Informationen zu den Autoren des Systems. Für die Darstellung auf mobilen Geräten sind keine Schwierigkeiten zu erwarten.

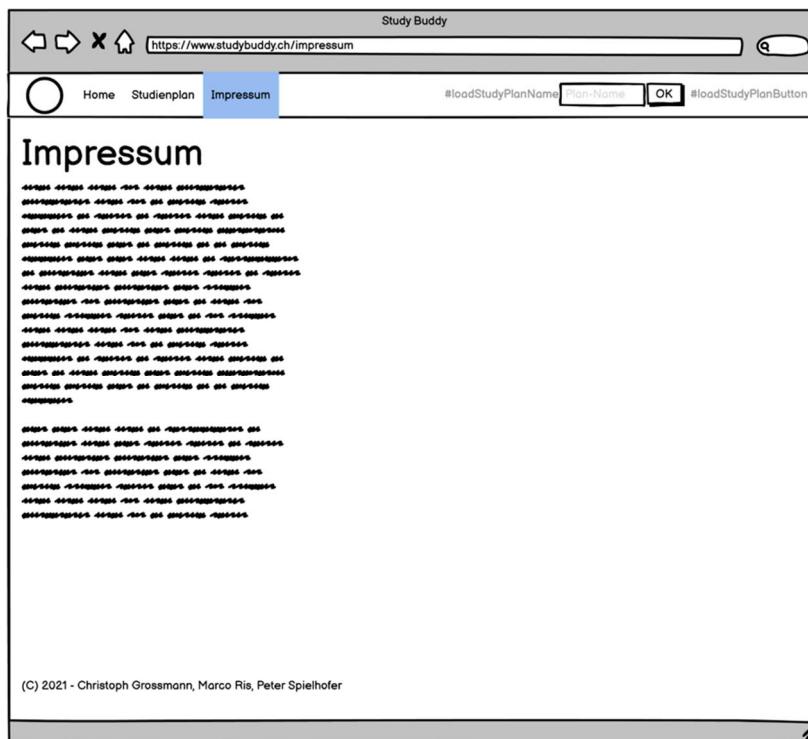


Abbildung 31: Mockup Impressum. Quelle: Autor

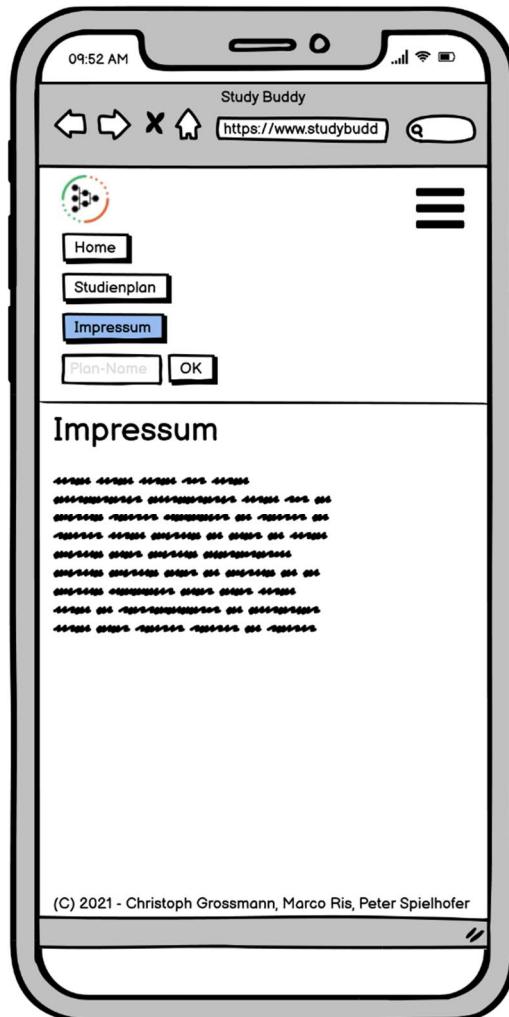


Abbildung 32: Mockup Impressum auf Mobilgeräten. Quelle: Autor

Tabelle 72: Elementbeschreibung und Typ vom Mockup Impressum. Quelle: Autor

Element	Beschreibung	Typ
<b>keine</b>	Hier wird nur Text angezeigt	String

## 6.4 User-Stories

### 6.4.1 Konfliktbewältigung

User-Stories, welche mit anderen User-Stories, im Konflikt stehen benötigen einer speziellen Behandlung. Solche Konflikte werden im Projektteam untersucht und die Konsequenzen besprochen. Mögliche Konsequenzen:

- User-Stories, A oder User-Stories, B wird nicht umgesetzt (Konfliktvermeidung). Zwingend erforderlich ist hierbei eine Begründung zur Nichtumsetzung.
- User-Stories, A oder User-Stories, B wird so verändert, damit beide User-Stories, umgesetzt werden können (Konfliktbewältigung).
- User-Stories, A und User-Stories, B werden gemäss Definition umgesetzt und der daraus entstandene Konflikt in einem weiteren Schritt mittels einer neuen User-Stories, C behandelt (Konfliktakzeptanz welche später behandelt wird).

### 6.4.2 Priorisierung der User-Stories

Erläuterungen zu den Prioritäten: Die User-Stories, werden unterschiedlich Priorisiert (ähnlich wie Muss-, Kann-Ziele). Jede(r) Anforderung usw. wird gewürdigt und als User-Stories, erfasst. Die Umsetzung dieser User-Stories, hängt jedoch von verschiedenen Faktoren (ist genügend Zeit vorhanden? wie wichtig ist die User-Story,?) ab. Nachfolgende Auflistung erklärt ergänzend die Idee, die hinter einer Priorisierung steht:

Tabelle 73: Erklärung zu den Prioritäten. Quelle: Autor

Bezeichnung	Erklärung	Farocode
<b>Priorität 1</b>	Zu dieser Gruppe gehören alle Anforderungen, welche essenziell für die Software sind.  Eine Nichterfüllung oder Nichtumsetzung: <ul style="list-style-type: none"> <li>• macht die Software unvollständig (gemäss Zielsetzung)</li> <li>• führt zu Fehlern in der Software</li> <li>• gefährdet den Projekterfolg</li> <li>• führt dazu, dass andere User Stories mit derselben Priorität nicht umgesetzt werden können.</li> </ul>	
<b>Priorität 2</b>	Alle Anforderungen, welche folgende Kriterien erfüllen: <ul style="list-style-type: none"> <li>• Ergänzt die Software um Features, welche die Akzeptanz massgeblich steigert</li> <li>• Eine Nichtumsetzung einer solchen Anforderung trägt nicht negativ zum Gesamtbild oder Stabilität der Software bei.</li> </ul>	
<b>Priorität 3</b>	Diese Priorität kann genutzt werden, um Anforderungen welche «nice to have» zu klassifizieren. Die Umsetzung ebensolcher Anforderung kann erfolgen, falls alle anderen Anforderungen der Priorität 1 und Priorität 2 erfolgt ist.	

### 6.4.3 Auflistung der User-Stories

Tabelle 74: User Story #1 - Studienplan erstellen. Quelle: Autor

Nummer	U-1		
<b>Titel</b>	Studienplan erstellen		
<b>Referenz Use Case</b>	Haupt Use Case – Vogelperspektive		
<b>Referenz Mockup</b>	Studienplan erstellen		
<b>User Story</b>	Als User möchte ich einen neuen Studienplan erstellen können.		
<b>Diskussion</b>	<ul style="list-style-type: none"> <li>Welche Komponente/Ansicht sind erforderlich? Es wird eine Komponente "studyPlanStart.component" benötigt (html- und scss-Datei).</li> <li>Welche Services sind dazu notwendig? Es wird eine Service-Klasse (Schnittstelle zum Backend) gemäss Schnittstellen-Definition benötigt: StudyPlanService.ts</li> <li>Welche Daten sind für die Erstellung des Studienplans notwendig? Programm (HF Passarelle) und Name muss gewählt werden. Der Name muss eindeutig sein – dies muss bei der Eingabe geprüft werden.</li> <li>Gibt es Restriktionen bezüglich des Namens? Der Studienplannname darf maximal 255 Zeichen lang sein und sollte nur Zeichen A-Z, a-Z und 0-9 enthalten. Dies sollte bereits auf dem UI geprüft werden. Input Pattern: [A-Za-z0-9]{255}</li> </ul>		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>Der Benutzer kann einen neuen Studienplan erstellen.</li> <li>Der Studienplan wird auf Basis der Auswahl: "Programm" und "Name" erstellt.</li> <li>Nach dem Bestätigen des neuen Studienplans mittels OK Button "#newStudyPlanButton" wird der User automatisch auf die Planung des Studienplans (<a href="http://url/studienplan/add">http://url/studienplan/add</a>) weitergeleitet (welche aber noch zu erstellen ist).</li> </ul>		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	MR
<b>SOLL</b>	8	<b>IST</b>	8

Tabelle 75: User Story #2 - Studienplanung anzeigen. Quelle: Autor

<b>Nummer</b>	<b>U-2</b>		
<b>Titel</b>	Studienplanung anzeigen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studienplanung anzeigen		
<b>User Story</b>	Als User möchte ich eine Maske/Ansicht, auf welcher ich die Studiengangs Planung vornehmen kann.		
<b>Diskussion</b>	<ul style="list-style-type: none"> <li>• Welche Komponente/Ansicht sind erforderlich? Es wird eine Komponente "studyPlan.component" benötigt (html- und scss-Datei). Es wird auch Service-Klasse (Schnittstelle zu Backend) gemäss Schnittstellen-Definition benötigt.</li> <li>• Welche Services sind dazu notwendig? Es wird eine Service-Klasse (Schnittstelle zu Backend) gemäss Schnittstellen-Definition benötigt: StudyPlanService.ts</li> <li>• Welche Daten sind für diese Ansicht notwendig? Die ID-welche mittels URL Parameter übermittelt wird, wird genutzt um die Planung anzuzeigen. Im NgOnInit der Komponente soll der Studienplan mittels ID geladen werden.</li> </ul>		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>• Ansicht/Komponente für die Anzeige der Studienplanung ist vorhanden.</li> <li>• Mittels URL (<a href="http://url/studienplan/edit">http://url/studienplan/edit</a>) kann die Ansicht angezeigt werden.</li> <li>• Die Ansicht beinhaltet die Elemente gemäss Mockup.</li> <li>• Die Elemente sind mit Dummy-Daten vorgefüllt (Platzhalter-Daten).</li> <li>• Die Ansicht beinhaltet noch keine Logik, da diese mit nachfolgenden User Stories erarbeitet wird.</li> </ul>		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	MR	<b>Mitwirkend</b>	PS
<b>SOLL</b>	8	<b>IST</b>	9

Tabelle 76: User Story #3 - Jahr wählen. Quelle: Autor

<b>Nummer</b>	<b>U-3</b>		
<b>Titel</b>	Jahr wählen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich ein Jahr, zu welchem ich das Modul hinzufügen möchte, auswählen können.		
<b>Diskussion</b>	Welches Jahr soll zur Auswahl gestellt werden? Das aktuelle Jahr? Als Vorauswahl sollte das aktuelle Jahr des Users – welches über den Client-Browser ermittelt werden kann – zur Verfügung gestellt werden.		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>Jahr ist im Auswahlfeld #studyYearDropdown wählbar. Das Jahr wird 4-stellig dargestellt.</li> <li>Der Initial-Wert ist das aktuelle Jahr des Client-Browsers.</li> </ul>		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	CHG	Mitwirkend	PS
<b>SOLL</b>	<b>2</b>	<b>IST</b>	2

Tabelle 77: User Story #4 - Vertiefungsrichtung wählen. Quelle: Autor

<b>Nummer</b>	<b>U-4</b>		
<b>Titel</b>	Vertiefungsrichtung wählen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich für die Studiengangs-Planung die Vertiefungsrichtung wählen können. Dies soll mir helfen, nur noch die Module anzuzeigen, welche für diese Vertiefungsrichtung relevant sind. Dabei interessieren mich die Module, für die ich für die gewählte Vertiefungsrichtung Relevanzpunkte erhalte.		
<b>Diskussion</b>	Wie wirkt sich der Filter auf dem UI aus? Mittels des Filters wird die Auflistung der Module geändert. In der Auflistung sollen dann nur noch die Module angezeigt werden, welche für die Vertiefungsrichtung relevant sind. Nachtrag: Es hat sich gezeigt, dass auch die Basis-Module angezeigt werden sollen, da diese zu jedem Zeitpunkt eingeplant werden können → Zusätzliche User-Story erstellen		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>Filter für Vertiefungsrichtung (#specificationDropdown) ist vorhanden.</li> <li>Der Filter wirkt sich auf die Modulauflistung aus: Es werden nur noch die Module angezeigt, die für diese Vertiefungsrichtung relevant sind (für die gewählte Vertiefungsrichtung werden Relevanzpunkte vergeben).</li> </ul>		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	CHG	Mitwirkend	MR
<b>SOLL</b>	8	<b>IST</b>	9

Tabelle 78: User Story #4 - Nachbesserung. Quelle: Autor

Nummer	U-4: Nachbesserung		
<b>Titel</b>	Vertiefungsrichtung wählen – Basis-Module immer anzeigen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich die Basis-Module jederzeit einplanen können. Da die Basis-Module je nach Studiengang unabhängig von der gewählten Vertiefungsrichtung eingeplant werden können.		
<b>Diskussion</b>	<p>Was ist genau zu tun?            Es hat sich gezeigt, dass auch die Basis-Module angezeigt werden sollen, da diese zu jedem Zeitpunkt eingeplant werden können (es gibt zu den Basis-Modulen keinen fixen Zeitplan). Die Testfälle zu U4 müssen auch angepasst werden.</p>		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>Filter für Vertiefungsrichtung (#specificationDropdown) ist vorhanden.</li> <li>Der Filter wirkt sich auf die Modulaufstellung aus: Es werden die Module gemäss Vertiefungsrichtung UND die Basis-Module angezeigt.</li> </ul>		
<b>Priorität</b>	Priorität 2		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	MR
<b>SOLL</b>	2	<b>IST</b>	2

Tabelle 79: User Story #5 - Semester und Semestertyp wählen. Quelle: Autor

Nummer	U-5		
<b>Titel</b>	Semester und Semestertyp wählen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich für die Studiengangs Planung ein Semester und den Semestertyp wählen können.		
<b>Diskussion</b>	<p>Sind irgendwelche Vorgaben einzuhalten oder beginnt jede Studienplanung beim Semester 1? Wie wirkt sich die Semesterwahl auf die Einplanung des Modules aus?</p> <p>Da wir den Studienplaner auf das Programm "HF Passarelle" ausrichten, sollten wir berücksichtigen, dass das Semester (gemäss Webseite FFHS) bei Semester 5 beginnt und das Studium im Standard-Fall bei Semester 9 endet. Die Semesterwahl wirkt sich wie folgt aus: Das Modul, das der User wählt, soll in dem gewählten Semester eingetragen werden. Außerdem sollen als Semestertyp nur die beiden Semester "Herbst" und "Frühling" zur Verfügung stehen.</p>		
<b>Akzeptanzkriterien</b>	<ul style="list-style-type: none"> <li>Die User Story ist erfüllt, wenn der User ein Semester für das ausgewählte Modul aus dem Dropdown #studySemesterDropdown gewählt werden kann.</li> <li>Für das Programm "HF Passarelle" stehen nur die Semester 5-9 zur Verfügung, da diese die Standard-Vorgabe für dieses Programm sind.</li> </ul>		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	MR
<b>SOLL</b>	4	<b>IST</b>	3

Tabelle 80: User Story #6 - Modul wählen. Quelle: Autor

Nummer	U-6		
Titel	Modul wählen		
Referenz zu Use Case	Studiengang planen		
Referenz zu Mockup	Studiengang planen		
User Story	Als User möchte ich für die Studiengangs Planung ein Modul wählen um dieses in den Studienplan eintragen können.		
Diskussion	<p>Mit welcher Technik sollen Module ausgewählt werden können?</p> <ul style="list-style-type: none"> <li>• Auswahl durch Klick</li> <li>• Auswahl durch Drag&amp;Drop</li> <li>• Auswahl durch Checkbox</li> </ul> <p>Die Übertragung der Module mittels Drag&amp;Drop wäre sicherlich "state of the art" und entspricht unseren Vorstellungen. Es sollte mittels POC geprüft werden, ob diese einfach umzusetzen ist. Der POC sollte jedoch nach 4h abgebrochen werden.</p> <p>Ergebnis aus POC: Es gibt einige Zusatzpakete, um Drag&amp;Drop Events mit Angular umzusetzen. Es wurde das Paket ng2-dragula geprüft, welches im Prinzip die Umsetzung ermöglicht. Jedoch treten einige Problemstellungen in der Umsetzung auf. Die Unterstützung für Mobil-Geräte ist nicht ideal (Für mobile Geräte ist Drag&amp;Drop schwierig umzusetzen). Außerdem stürzte die Applikation immer wieder ab. Der gemeinsame Entscheid läuft deshalb auf Umsetzung mittels Klick-Event.</p>		
Akzeptanzkriterien	<ul style="list-style-type: none"> <li>• Die User Story ist erfüllt, wenn der User ein Modul aus dem Bereich #moduleCheckbox wählen kann. Ein gewähltes Modul wird farblich hervorgehoben, so dass es für den User ersichtlich ist, dass das Modul als Auswahl angenommen wurde.</li> <li>• Es kann jeweils nur ein Modul in der Kombination "Semester", "Jahr", "Vertiefungsrichtung" gewählt werden.</li> </ul>		
Priorität	Priorität 1		
Verantwortlich	MR	Mitwirkend	PS
SOLL	4 + 4 für POC (DragDrop)	IST	7

Tabelle 81: User Story #7 - Auswahl in Planung übertragen. Quelle: Autor

Nummer	U-7		
<b>Titel</b>	Auswahl in Planung übertragen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich die getätigte Auswahl: Jahr, Semester, Vertiefungsrichtung und Modul in die Planung übertragen können.		
<b>Diskussion</b>	<p>In Kombination mit dem POC aus U-6 wird folgendes umgesetzt: Mit einem Klick auf das Modul wird die Auswahl (Jahr, Semester, Vertiefungsrichtung und Modul in die Planung übertragen). Die Akzeptanzkriterien aus U-6 sind für die Erfüllung dieser User Story zu berücksichtigen.</p> <p><b>Das Modul sollte nur einmal eingetragen werden können. Eine Mitteilung wird angezeigt, falls versucht wird, das Modul mehrfach einzutragen.</b></p>		
<b>Akzeptanzkriterien</b>	Die User Story gilt als erfüllt, wenn <b>durch einen Klick auf den Button #addModuleToStudyplanButton</b> die getätigte Auswahl in die Planung übertragen wird. Die Auswahl wird dabei im Backendpersistiert. Das Modul wird nur einmal eingetragen. Ein erneutes Eintragen des Modules ist nicht möglich.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	Mitwirkend	MR
<b>SOLL</b>	8	IST	9

Tabelle 82: User Story #8 - Modul entfernen. Quelle: Autor

Nummer	U-8		
<b>Titel</b>	Modul entfernen		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich ein eingeplantes Modul auch wieder entfernen können.		
<b>Diskussion</b>	<p>Gibt es etwas zu validieren oder können Probleme auftreten?</p> <p><b>Es sollten keine Probleme auftreten. Es sollte jedoch validiert werden, dass der Studienplan existiert.</b></p>		
<b>Akzeptanzkriterien</b>	Modul ist aus der Planung mittels Button <b>#removeModuleFromStudyplanButton</b> entfernt werden. Das Modul wird aus der Planung entfernt (Datenbank) und im UI in der Planung nicht mehr angezeigt.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	MR	Mitwirkend	CHG
<b>SOLL</b>	4	IST	6

Tabelle 83: User Story #9 - Relevanzpunkte sichten. Quelle: Autor

Nummer	U-9		
<b>Titel</b>	Relevanzpunkte sichten		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich die erreichten Relevanzpunkte der bereits geplanten Module sichten können.		
<b>Diskussion</b>	<p>Wie soll das am besten umgesetzt werden: Sollen die Relevanzpunkte auf dem UI ausgerechnet werden? Und wie kann sichergestellt werden, dass die Relevanzpunkte nach einer Anpassung am Studienplan (z.B. neues Modul hinzufügen oder Modul entfernen) aktuell und valid ist?</p> <p>Die Berechnung der Relevanzpunkte sollte auf dem UI ausgeführt werden. Es besteht zwar die Möglichkeit, die Relevanzpunkte mittels API-Endpoint vom Backend abzuholen aber diese Variante führt zu mehr Traffic.</p> <p>Die Berechnung der Relevanzpunkte konnte nicht auf dem UI durchgeführt werden. Die Umsetzung war zu kompliziert und die Aktualisierung war in einigen Fällen erst durch Neu laden (F5) möglich. Es wurde deshalb doch die Variante mittels API-Endpoints umgesetzt. Die dazu notwendigen Testfälle wurden erstellt und dokumentiert.</p>		
<b>Akzeptanzkriterien</b>	In der Visualisierung werden die Relevanzpunkte je Vertiefungsrichtung aufgeführt. Die Relevanzpunkte sind auch nach Anpassung an der Studienplanung immer aktuell.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	CHG
<b>SOLL</b>	3	<b>IST</b>	14

Tabelle 84: User Story #10 - ECTS-Punkte sichten. Quelle: Autor

Nummer	U-10		
<b>Titel</b>	ECTS-Punkte sichten		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich sehen, wie viele ECTS-Punkte ich mit den eingeplanten Modulen erhalte.		
<b>Diskussion</b>	<p>Wie soll das am besten umgesetzt werden: Sollen die Relevanzpunkte auf dem UI ausgerechnet werden? Und wie kann sichergestellt werden, dass die Relevanzpunkte nach einer Anpassung am Studienplan (z.B. neues Modul hinzufügen oder Modul entfernen) aktuell und valid ist?</p> <p>Die Umsetzung des U-10 hat gezeigt dass die Relevanzpunkte mittels API-Endpoint auf dem Backend umgesetzt werden soll.</p>		
<b>Akzeptanzkriterien</b>	In einer Auflistung werden die ECTS-Punkte je Vertiefungsrichtung aufgeführt. Die Relevanzpunkte sind auch nach Anpassung an der Studienplanung immer aktuell.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	MR	<b>Mitwirkend</b>	PS
<b>SOLL</b>	3	<b>IST</b>	4

Tabelle 85: User Story #11 - Relevanzpunkte pro Vertiefungsrichtung sichten. Quelle: Autor

<b>Nummer</b>	<b>U-11</b>		
<b>Titel</b>	Relevanzpunkte pro Vertiefungsrichtung sichten		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich sehen, wie viele Relevanzpunkte ein Modul in welcher Vertiefungsrichtungen ergibt.		
<b>Diskussion</b>	<p>Wie sollen diese Informationen dargestellt werden?</p> <p>Eine Möglichkeit wäre eine Pop-Up über den aufgelisteten Modulen:</p> 		
<b>Akzeptanzkriterien</b>	Informationen zu den Relevanzpunkten pro Vertiefungsrichtung sind abrufbar.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	MR	<b>Mitwirkend</b>	PS
<b>SOLL</b>	4	<b>IST</b>	4

Tabelle 86: User Story #12 - Semesterplan sichten. Quelle: Autor

<b>Nummer</b>	<b>U-12</b>		
<b>Titel</b>	Semesterplan sichten		
<b>Referenz zu Use Case</b>	Studiengang planen		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich sehen, wann welches Modul zu welchem Semester zur Auswahl steht.		
<b>Diskussion</b>	<p>Wie sollen die Modul Informationen dargestellt werden?</p> <p>Eine Möglichkeit wäre eine Pop-Up über den aufgelisteten Modulen:</p>  <p>Auf Basis der Erfahrungen der U-11 kann dies einfach mittels NGX-Bootstrap umgesetzt werden. Die Formatierung sollte so gemacht werden, dass bei Modulen ohne Information, kein Text angezeigt wird.</p>		
<b>Akzeptanzkriterien</b>	Informationen zu den Semester-Verfügbarkeiten pro Modul sind abrufbar.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	CHG
<b>SOLL</b>	3	<b>IST</b>	1

Tabelle 87: User Story #13 - Studienplan speichern. Quelle: Autor

<b>Nummer</b>	<b>U-13</b>		
<b>Titel</b>	Studienplan speichern		
<b>Referenz zu Use Case</b>	Haupt Use Case - Vogelperspektive		
<b>Referenz zu Mockup</b>	Studiengang planen		
<b>User Story</b>	Als User möchte ich die Planung, welche ich in der Software erstellt habe, sichern können. Die Sicherung ermöglicht mir eine erneute Bearbeitung der Planung.		
<b>Diskussion</b>	<p>Wie wird die Sicherung ausgeführt? Sichern wir es Dateibasiert (quasi ein Export) oder in der Software (Datenbank)? Wie kann eine Versionierung (Software kann sich weiterentwickeln) gewährleistet? Muss die Sicherung Rückwärtskompatibel sein (D.h. wenn ein Studienplan zu einem Zeitpunkt AB erstellt wird und sich die Spezifikation zu CD ändert, kann der Studienplan AB wieder geöffnet werden)?</p> <p>Bei der Erstellung einer neuen Planung wird ein neues Objekt angelegt, welches auf der Datenbank persistiert wird. Der erneute Aufruf des Plans kann mittels Namen ermöglicht werden. Eine Rückwärtskompatibilität kann auf Grund des engen Zeitrahmens des Projektes nicht gewährleistet werden. Die Daten, welche in den vorhergehenden User Stories nur auf dem Frontend gespeichert wurden sollen nun auf dem Backend persistiert werden.</p>		
<b>Akzeptanzkriterien</b>	Die User Story ist fertiggestellt, wenn die Planung gesichert werden kann. Der User muss für die Sicherung keine spezifische Aktion unternehmen; jede Aktion (hinzufügen/löschen eines Moduls) wird automatischpersistiert.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	CHG
<b>SOLL</b>	8	<b>IST</b>	10

Tabelle 88: User Story #14 - Studienplan mittels Namen laden. Quelle: Autor

<b>Nummer</b>	<b>U-14</b>		
<b>Titel</b>	Studienplan mittels Namen laden		
<b>Referenz zu Use Case</b>	Haupt Use Case - Vogelperspektive		
<b>Referenz zu Mockup</b>	Studienplan laden		
<b>User Story</b>	Als User möchte ich mein Studienplan auch wieder laden können. Dies möchte ich mittels Eingabe des Studienplan-Namens ausführen können.		
<b>Diskussion</b>	<p>Was soll passieren, wenn der Studienplan nicht existiert?</p> <p>Wir sollten nach dem Klick auf den Button #loadStudyPlanButton verifizieren, dass der Plan existiert. Wenn der Plan nicht existiert, soll eine Meldung ("Der Studienplan existiert nicht") angezeigt werden.</p>		
<b>Akzeptanzkriterien</b>	Studienplan lässt sich mittels Namen aufrufen. Dabei wird nach dem Klicken auf den Button #loadStudyPlanButton der User auf die Seite <a href="http://url/studienpan/edit?name=">http://url/studienpan/edit?name=</a> weitergeleitet. Falls der Name falsch ist (der Studienplan existiert nicht), so wird dem User eine Meldung angezeigt, dass die Eingabe falsch ist.		
<b>Priorität</b>	Priorität 1		
<b>Verantwortlich</b>	MR	<b>Mitwirkend</b>	PS
<b>SOLL</b>	8	<b>IST</b>	7

Tabelle 89: User Story #15 - Studienplan visualisieren. Quelle: Autor

Nummer	U-15																																																																																																																																																																																			
<b>Titel</b>	Studienplan visualisieren																																																																																																																																																																																			
<b>Referenz zu Use Case</b>	Haupt Use Case - Vogelperspektive																																																																																																																																																																																			
<b>Referenz zu Mockup</b>	Studienplan visualisieren																																																																																																																																																																																			
<b>User Story</b>	Als User möchte ich die Planung, welche ich in der Software erstellt habe, visuell darstellen können. In dieser Visualisierung möchte ich eine grafische Übersicht über die geplanten Module inkl. der erhaltenen Relevanz- und ECTS-Punkte sichten können. Des Weiteren muss ersichtlich sein in welchem Semester ich welches Modul eingeplant habe.																																																																																																																																																																																			
<b>Diskussion</b>	<p>Wie wird die Visualisierung umgesetzt? Tabellarisch oder mit einem Zusatz-Tool?</p> <p>Eine Beispielhafte Umsetzung könnte wie folgt aussehen (gemäss Mockup):</p> <table border="1"> <thead> <tr> <th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th><th>I</th><th>J</th><th>K</th></tr> <tr> <th></th><th>Beschreibung</th><th>FS21</th><th>HS21/22</th><th>FS22</th><th>HS22/23</th><th>FS23</th><th>Isech</th><th>Relevanzpunkte</th><th>Enterprise</th><th>Data Science</th><th>Ects</th></tr> </thead> <tbody> <tr> <td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>2</td><td>Wahl/Pflicht</td><td></td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td><td></td><td></td></tr> <tr> <td>3</td><td>Pflicht</td><td>Beschreibung</td><td>FS21</td><td>HS21/22</td><td>FS22</td><td>HS22/23</td><td>FS23</td><td>Isech</td><td></td><td></td><td></td></tr> <tr> <td>4</td><td></td><td>Management der IT-Sec (MITSec)</td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td>5</td></tr> <tr> <td>5</td><td></td><td>Embedded Systems mit Android (ESA)</td><td></td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td><td>5</td></tr> <tr> <td>6</td><td></td><td>SOA: Service Oriented Architecture</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>5</td></tr> <tr> <td>7</td><td></td><td>Programmier Paradigmen (ProPa)</td><td></td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td><td>5</td></tr> <tr> <td>8</td><td></td><td>Verteilte Systeme und Anwendungen (VSA)</td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>2</td><td>5</td></tr> <tr> <td>9</td><td></td><td>CICo: Cloud Computing</td><td></td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td><td>5</td></tr> <tr> <td>10</td><td></td><td>Internetsicherheit (INSich)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>5</td></tr> <tr> <td>11</td><td></td><td>DevOps</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>12</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>Summe</td><td>5</td><td>8</td><td>1</td></tr> <tr> <td>13</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>40</td></tr> </tbody> </table> <p>Die Visualisierung können wir mittels Tabellen ermöglichen. Der Einsatz eines Zusatz-Tools ist weniger sinnvoll, da eine Tabellarische Ansicht unserem Mockup entspricht.</p>	A	B	C	D	E	F	G	H	I	J	K		Beschreibung	FS21	HS21/22	FS22	HS22/23	FS23	Isech	Relevanzpunkte	Enterprise	Data Science	Ects	1												2	Wahl/Pflicht							2				3	Pflicht	Beschreibung	FS21	HS21/22	FS22	HS22/23	FS23	Isech				4		Management der IT-Sec (MITSec)							1		5	5		Embedded Systems mit Android (ESA)							2		5	6		SOA: Service Oriented Architecture									5	7		Programmier Paradigmen (ProPa)							2		5	8		Verteilte Systeme und Anwendungen (VSA)							1	2	5	9		CICo: Cloud Computing							2		5	10		Internetsicherheit (INSich)								1	5	11		DevOps										12								Summe	5	8	1	13											40
A	B	C	D	E	F	G	H	I	J	K																																																																																																																																																																										
	Beschreibung	FS21	HS21/22	FS22	HS22/23	FS23	Isech	Relevanzpunkte	Enterprise	Data Science	Ects																																																																																																																																																																									
1																																																																																																																																																																																				
2	Wahl/Pflicht							2																																																																																																																																																																												
3	Pflicht	Beschreibung	FS21	HS21/22	FS22	HS22/23	FS23	Isech																																																																																																																																																																												
4		Management der IT-Sec (MITSec)							1		5																																																																																																																																																																									
5		Embedded Systems mit Android (ESA)							2		5																																																																																																																																																																									
6		SOA: Service Oriented Architecture									5																																																																																																																																																																									
7		Programmier Paradigmen (ProPa)							2		5																																																																																																																																																																									
8		Verteilte Systeme und Anwendungen (VSA)							1	2	5																																																																																																																																																																									
9		CICo: Cloud Computing							2		5																																																																																																																																																																									
10		Internetsicherheit (INSich)								1	5																																																																																																																																																																									
11		DevOps																																																																																																																																																																																		
12								Summe	5	8	1																																																																																																																																																																									
13											40																																																																																																																																																																									
<b>Akzeptanzkriterien</b>	Die User Story ist erfüllt, wenn die Studienplanung visualisiert ist und folgende Informationen angezeigt werden:																																																																																																																																																																																			
	<ul style="list-style-type: none"> <li>• Geplante Module pro Semester</li> <li>• Relevanzpunkte pro Modul pro Vertiefung</li> <li>• ECTS Punkte</li> </ul> <p>In der Visualisierung werden nur die eingeplanten Module aufgeführt.</p>																																																																																																																																																																																			
<b>Priorität</b>	Priorität 2																																																																																																																																																																																			
<b>Verantwortlich</b>	PS	<b>Mitwirkend</b>	MR																																																																																																																																																																																	
<b>SOLL</b>	15	<b>IST</b>	24																																																																																																																																																																																	

Tabelle 90: User Story #16 - Studienplan löschen. Quelle: Autor

<b>Nummer</b>	<b>U-16</b>		
<b>Titel</b>	Studienplan löschen		
<b>Referenz zu Use Case</b>	Haupt Use Case - Vogelperspektive		
<b>User Story</b>	Als User möchte ich die Planung, welche ich in der Software erstellt habe, löschen können. Nicht mehr verwendete Planungen sollen nach der Löschung nicht mehr im System vorhanden sein.		
<b>Diskussion</b>	Wie wird die Löschung ausgeführt? Kann durch den Namen eine Eindeutige Zuweisung stattfinden? Was passiert nach der Löschung?  Löschen kann mittels ID durchgeführt werden. Nach der Bestätigung und Löschung wird der User auf die Hauptseite weitergeleitet.		
<b>Akzeptanzkriterien</b>	Die User Story ist fertiggestellt, wenn eine Planung erfolgreich aus dem System entfernt werden kann. Die Löschung wird mittels Button #deleteStudyplanButton durch den User initiiert. Nach der Löschung wird der User auf die Hauptseite weitergeleitet. Der Studienplan kann nicht mehr mittels des Namens aufgerufen werden (verifiziert die korrekte Löschung).		
<b>Priorität</b>	Priorität 3		
<b>Verantwortlich</b>	CHG	<b>Mitwirkend</b>	PS
<b>SOLL</b>	10	<b>IST</b>	13

## 6.5 Fachklassenmodell (Domänenmodell)

Basis für das nachfolgende, nach UML Notation, Fachklassenmodell ist der Anwendungsfalls «Module planen», welche die wichtigsten Fachklassen beinhaltet.

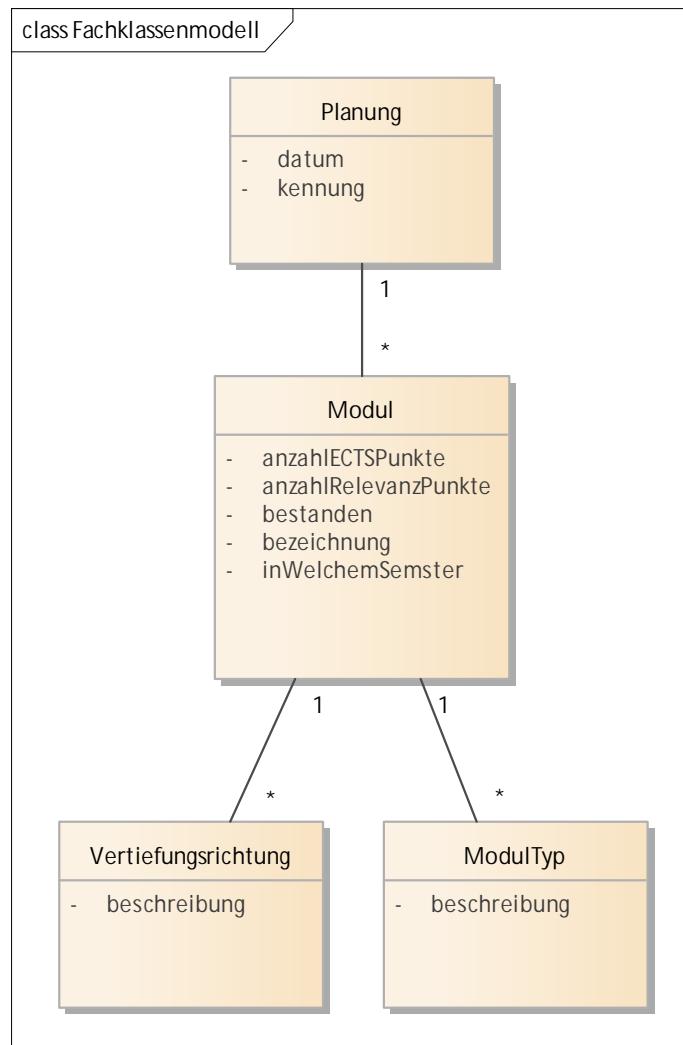


Abbildung 33: Fachklassenmodell. Quelle: Autor

Das Fachklassenmodell stellt die wichtigsten Klassen dar, die durch das Use Case identifiziert werden konnten. Die Planung beinhaltet die Module mit den Informationen zu der Anzahl Relevanz-Punkte, in welchem Semester das Modul stattfindet, die Bezeichnung des Moduls und ob das Modul bestanden ist. Das Modul enthält weitergehend die Information zu welchen Vertiefungsrichtungen (Data Science usw.) es gehört und zu welchen Modul-Typ (Wahlmodul, Vertiefungsmodul) es zuzuordnen ist.

## 7 Softwareentwurfsdokumente

### 7.1 Installation und Hardware

Das gesamte System wird lokal betrieben. Da das Projektbudget begrenzt ist, kann das System nicht in (einer kostenpflichtigen) der Cloud betrieben werden

### 7.2 Kontextabgrenzung

Die Kontextabgrenzung ist in der Ebene 1 der Bausteinsicht enthalten. Das System ist ein für sich geschlossene Lösung und hat keine Schnittstellen gegen aussen (zu anderen Systemen).

### 7.3 Bausteinsichten der Softwarearchitektur

Die nachfolgende Sicht zeigt das System des Studienplaners mit dessen Komponenten auf. Die Darstellung und Dokumentation orientiert sich auf Basis Bausteinsicht nach Arc42 - Dokchess Beispiel (Dr. Starke, 2021).

Die Bausteinsicht bildet die gewünschte Struktur des Systems als System-Komponenten ab. Die Sicht macht Strukturen und Zusammenhänge zwischen den Bausteinen der Architektur explizit. Die Bausteinsicht beantwortet die Frage, aus welchen Komponenten/Subsystemen das System besteht und wie diese in Abhängigkeit zueinander stehen (Starke, 2009).

#### 7.3.1 Ebene 1 – Black Box Ansicht

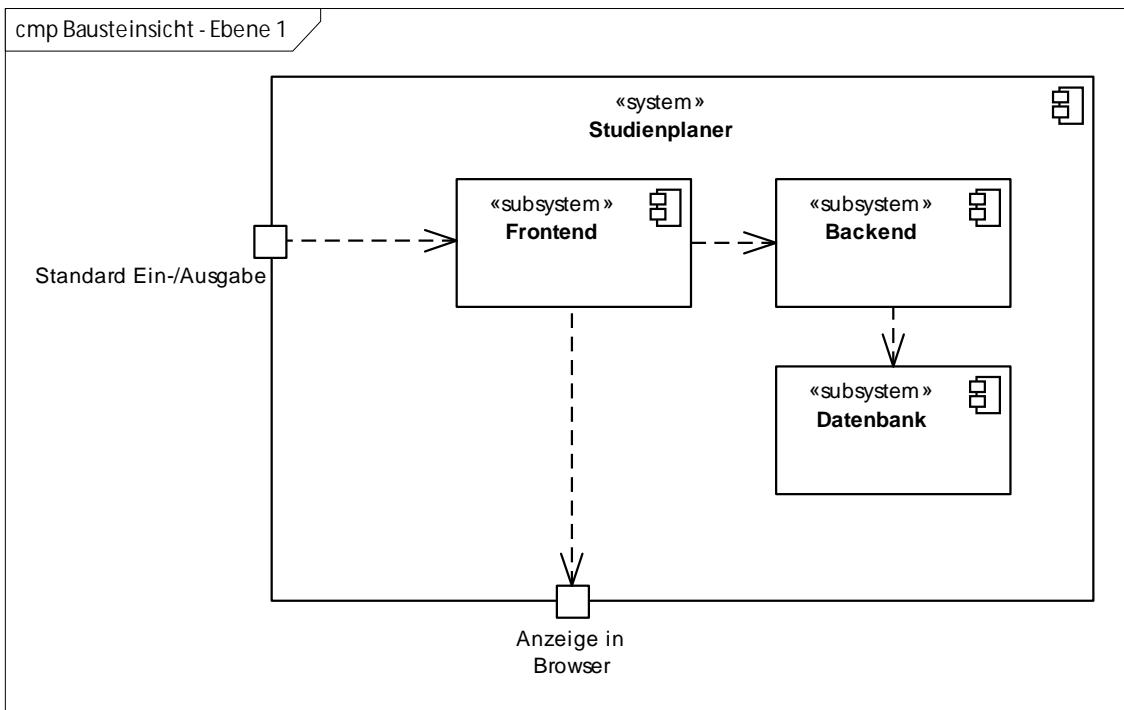


Abbildung 34: Bausteinsicht Ebene 1. Quelle: Autor

Der Studienplaner beinhaltet 3 Subsysteme. Die Pfeile beschreiben die Aufrufe/Rückrufe zwischen den Subsystemen. Diese werden als Schnittstellen bezeichnet und in den nachfolgenden Kapiteln näher beschrieben. Das System "Studienplaner" besteht aus folgenden Subsystemen:

Tabelle 91: Subsysteme Ebene 1. Quelle: Autor

Subsystem	Beschreibung
<b>Frontend</b>	Stellt das grafische Benutzer-Interface für die Planung der Module uvm. für den User zur Verfügung.
<b>Backend</b>	Stellt Schnittstellen für das Subsystem Frontend zur Verfügung und stellt die funktionellen Operationen zur Verfügung.
<b>Datenbank</b>	Persistiert (speichert) die Daten gemäss Datenmodell in die Datenbank.

## 7.4 Ebene 2 – White Box Ansicht

### 7.4.1 Frontend

Dieses Subsystem stellt die Eingabe und Ausgabemöglichkeiten für den User zur Verfügung. Auf diesem Subsystem werden alle Eingaben durch den User entgegengenommen (Input: Browser) und entsprechend verarbeitet. Die Verarbeitung, insbesondere möglicher Validierungsmöglichkeiten, ist auf dieser Ebene nicht genauer spezifiziert. Das Subsystem basiert auf einer Webanwendung (HTML, CSS) welches mittels des Frameworks Angular erstellt und für den Browser kompiliert wird. Das Subsystem stellt seine Funktionalität über den Angular Core Abstraktions-Layer zur Verfügung, welcher nicht näher betrachtet wird (Black Box). Module werden vom Subsystem app.module importiert. Dadurch können eigene Module, welche Komponenten zur Verfügung stellen, genutzt werden. Die Komponenten verwenden Templates, in welchen mittels HTML, CSS und Angular-Direktiven ein Event Binding und Property Binding realisiert wird. Die Komponenten nutzen die Services um mit diesen mit dem Backend (API) zu kommunizieren. Für die Anzeige wird das Dynamische DOM aufbereitet und über den Browser wieder angezeigt (bidirektionale Kommunikation) (Google, Angular, 2021).

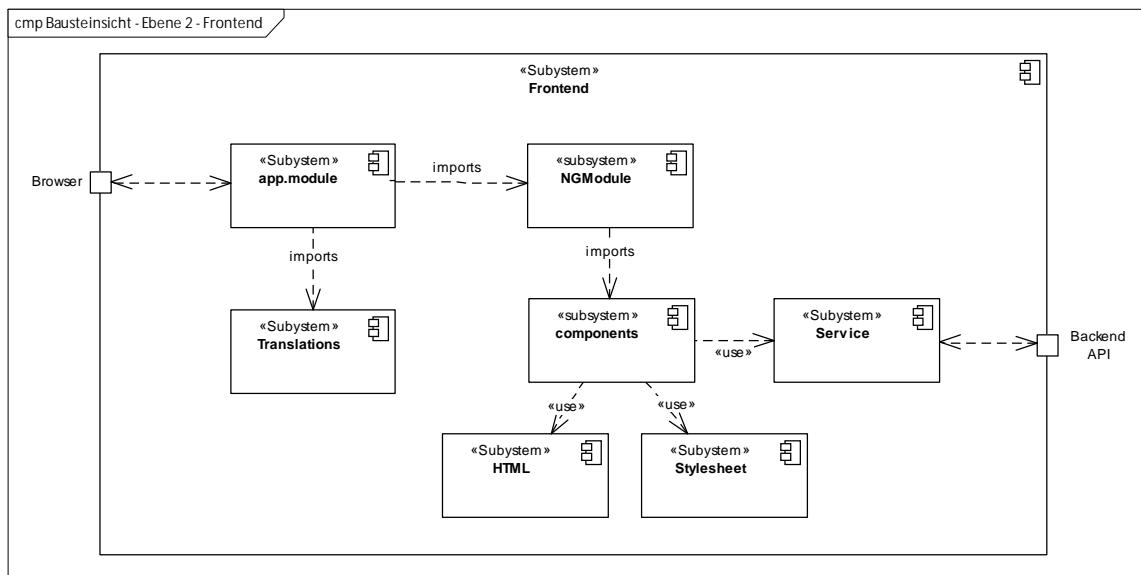


Abbildung 35: Bausteinansicht-Ebene 2 - Frontend. Quelle: Autor

Tabelle 92: Subsysteme Ebene 2 - Frontend. Quelle: Autor

Subsystem	Beschreibung
<b>App.module</b>	Die Registrierung der verschiedenen Module wird in der Datei app.module.ts bereitgestellt.
<b>Translations</b>	Vorbereitung der Möglichkeit die App in verschiedenen Sprachen zu übersetzen durch einfaches Erstellen einer neuen JSON-Datei
<b>NgModule</b>	NgModule werden als eigenständige Teile Der Applikation erstellt und beinhalten eine oder mehrere Components.
<b>Components</b>	Dies sind Bausteine der Module die mit eigenen Stylesheets, Logik und HTML für den Browser kompiliert werden.
<b>Service</b>	Hier wird die Kommunikation von den Components über das API zum Backend bereitgestellt.
<b>HTML</b>	Definition der anzuzeigenden Elemente des Components mittels Property Binding.
<b>Stylesheet</b>	Individuelle CSS-Einstellungen eines jeden Components.

Das Subsystem "Frontend" besitzt nachfolgende Schnittstellen:

Tabelle 93: Schnittstellen Ebene 2 - Frontend. Quelle: Autor

Schnittstelle von	Schnittstelle zu	Beschreibung
<b>Browser</b>	App.Module	(Black Box) Kommunikation mit dem Browser (Client).
<b>App.Module</b>	NgModule	Import-Schnittstelle.
<b>App.Module</b>	Translations	Import-Schnittstelle. Zur Globalen Nutzung der Übersetzungen in Form einer JSON-Datei.
<b>NgModule</b>	Components	Import-Schnittstelle. Importieren der Angular-Komponenten.
<b>Components</b>	HTML	Import-Schnittstelle. Importieren der HTML-Datei.
<b>Components</b>	Stylesheet	Import-Schnittstelle. Importieren der SCSS-Datei.
<b>Components</b>	Service	Verwendung der Service zur Kommunikation mit dem Backend
<b>Service</b>	Backend	Kommunikation mit dem Backend über HTTP.

#### 7.4.2 Backend

Das Subsystem Backend stellt über eine API die Daten für das Frontend zur Verfügung. http-Request werden vom Controller-Subsystem entgegengenommen und an das Service-Subsystem weitergeleitet. Das Service-Subsystem führt die Operation (Verarbeitung der Daten) durch. Das Service-Subsystem kennt das Repository-Subsystem welche für die Persistierung der Daten in die Datenbank zuständig ist.

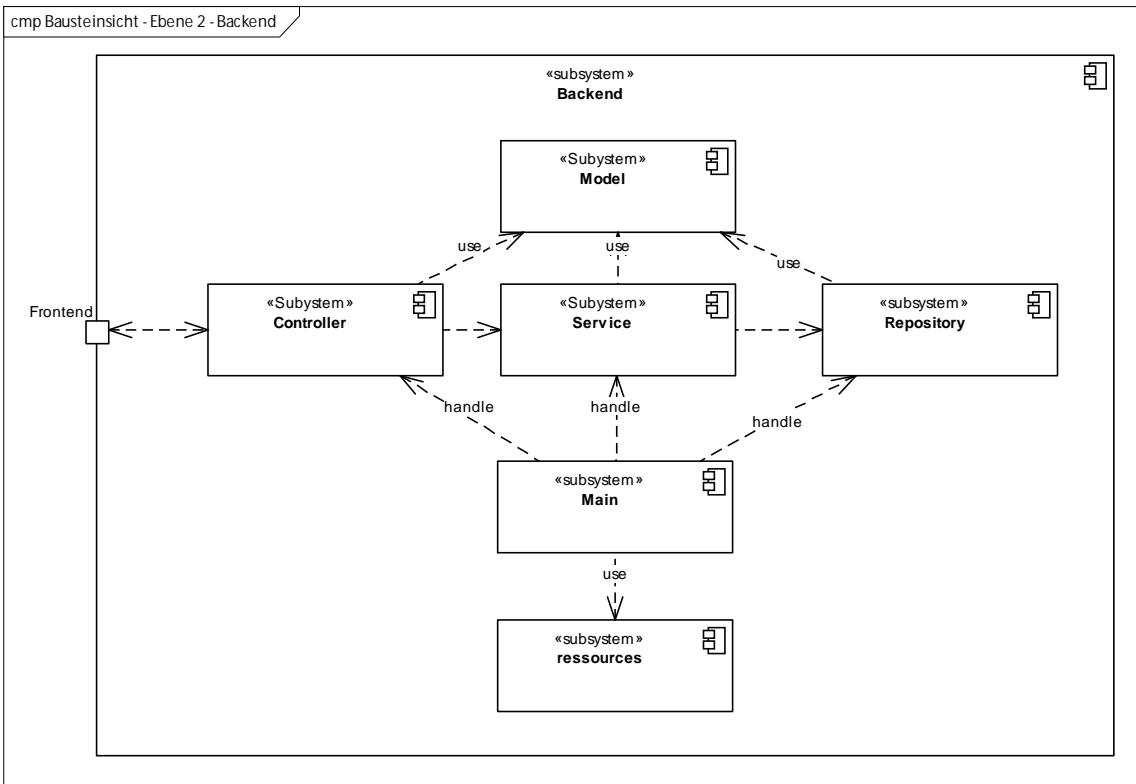


Abbildung 36: Bausteinsicht-Ebene 2 - Backend. Quelle: Autor

Das Subsystem Backend besteht aus nachfolgenden Subsystemen:

Tabelle 94: Subsysteme Ebene 2 - Backend. Quelle: Autor

Subsystem	Beschreibung
<b>Controller</b>	Die Schnittstelle zum Browser. Dieses Subsystem bietet eine API für http-Request an. Über dieses Subsystem werden Anfragen vom Browser entgegengenommen und in das Subsystem "Service" weitergeleitet.
<b>Service</b>	Verarbeitet Aufrufe vom Subsystem Controller. Es sendet und ruft Daten über das Subsystem "Repository". In diesem Subsystem wird die eigentliche Logik des Subsystems "Backend" implementiert (Studienplan erstellen, Studienplan laden usw.).
<b>Repository</b>	Dieses Subsystem verwaltet die Operationen für die Datenbank. Über dieses Subsystem werden somit alle Datenbank-Operationen ausgeführt (Daten laden, Daten speichern usw.).
<b>Model</b>	Dieses Subsystem besteht aus Modell-Klassen, welche für die System-Daten repräsentieren. Es beinhaltet simple POJO-Klassen wie auch komplexe Java-Klassen.
<b>Main</b>	Dieses Subsystem stellt den Einstiegspunkt für die Ausführung der Backend-Anwendung dar. Dieses Subsystem wird von der JVM aufgerufen um das gesamte Subsystem "Backend" zur Verfügung zu stellen.
<b>Ressources</b>	In diesem Subsystem werden Konfigurations-Dateien, Bilder und weitere Ressourcen hinterlegt. Da Spring eingesetzt wird, beinhaltet dieses Subsystem auch die Spring-Konfigurationsdatei "application.properties".

Das Subsystem "Backend" besitzt nachfolgende Schnittstellen:

Tabelle 95: Schnittstellen Ebene 2 - Backend. Quelle: Autor

Schnittstelle von	Schnittstelle zu	Beschreibung
<b>Frontend</b>	<b>Controller</b>	Über diese Schnittstelle wird es ermöglicht http-Requests vom Frontend an das Backend zur Verfügung zu stellen. Die Schnittstelle wird mittels http-Endpoints zur Verfügung gestellt. Auf eine Authentifizierung wird hierbei verzichtet; die Endpoints sind somit public.
<b>Controller</b>	<b>Frontend</b>	Anfragen vom Frontend werden auf dem Controller entgegengenommen und in tieferen Schichten (Service) verarbeitet. Die Antwort (Response) erfolgt wieder zurück ans Frontend mittels http-Response.
<b>Controller</b>	<b>Model</b>	Subsystem Controller verwendet das Subsystem Controller für die Instanzierung der Model-Klassen.
<b>Controller</b>	<b>Service</b>	Das Service-Subsystem wird für operative Tätigkeiten über das Controller-Subsystem verwendet. Das Subsystem Service bietet hierzu Methoden über ein Interface an und liefert Objekte an das Controller-Subsystem zurück.
<b>Service</b>	<b>Model</b>	Das Service-Subsystem verwendet das Subsystem Controller für die Instanzierung der Model-Klassen.
<b>Service</b>	<b>Repository</b>	Das Service-Subsystem greift auf Methoden des Repository-Subsystems zu. Die Methoden werden über ein Interface zur Verfügung gestellt.
<b>Main</b>	<b>Controller</b>	Das Main-Subsystem verwaltet als Startpunkt der Applikation das Controller-Subsystem.
<b>Main</b>	<b>Service</b>	Das Main-Subsystem verwaltet als Startpunkt der Applikation das Service-Subsystem.
<b>Main</b>	<b>Repository</b>	Das Main-Subsystem verwaltet als Startpunkt der Applikation das Repository-Subsystem.
<b>Main</b>	<b>ressources</b>	Das Main-Subsystem verwaltet als Startpunkt der Applikation das Ressourcen-Subsystem.

### 7.4.3 Datenbank

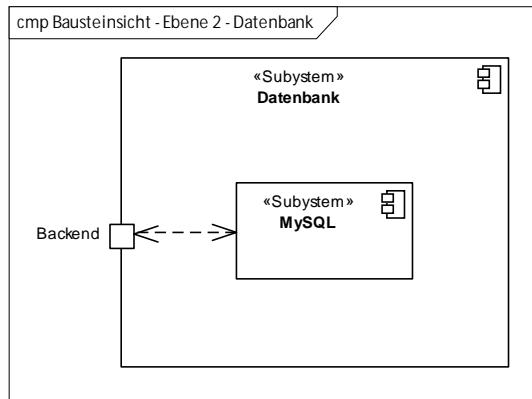


Abbildung 37: Bausteinsicht Ebene 2 – Datenbanken. Quelle: Autor

Das Subsystem "Datenbank" besteht aus einem Subsystem "MySQL" welches nicht näher betrachtet wird. Es wird als Black-Box repräsentiert, da wir uns nicht näher um die genaue Funktionsweise dieses Subsystems kümmern müssen. Die Schnittstelle vom Subsystem "Datenbank" wird über die Client-Verbindung ermöglicht.

Das Subsystem "Datenbank" besitzt nachfolgende Schnittstellen:

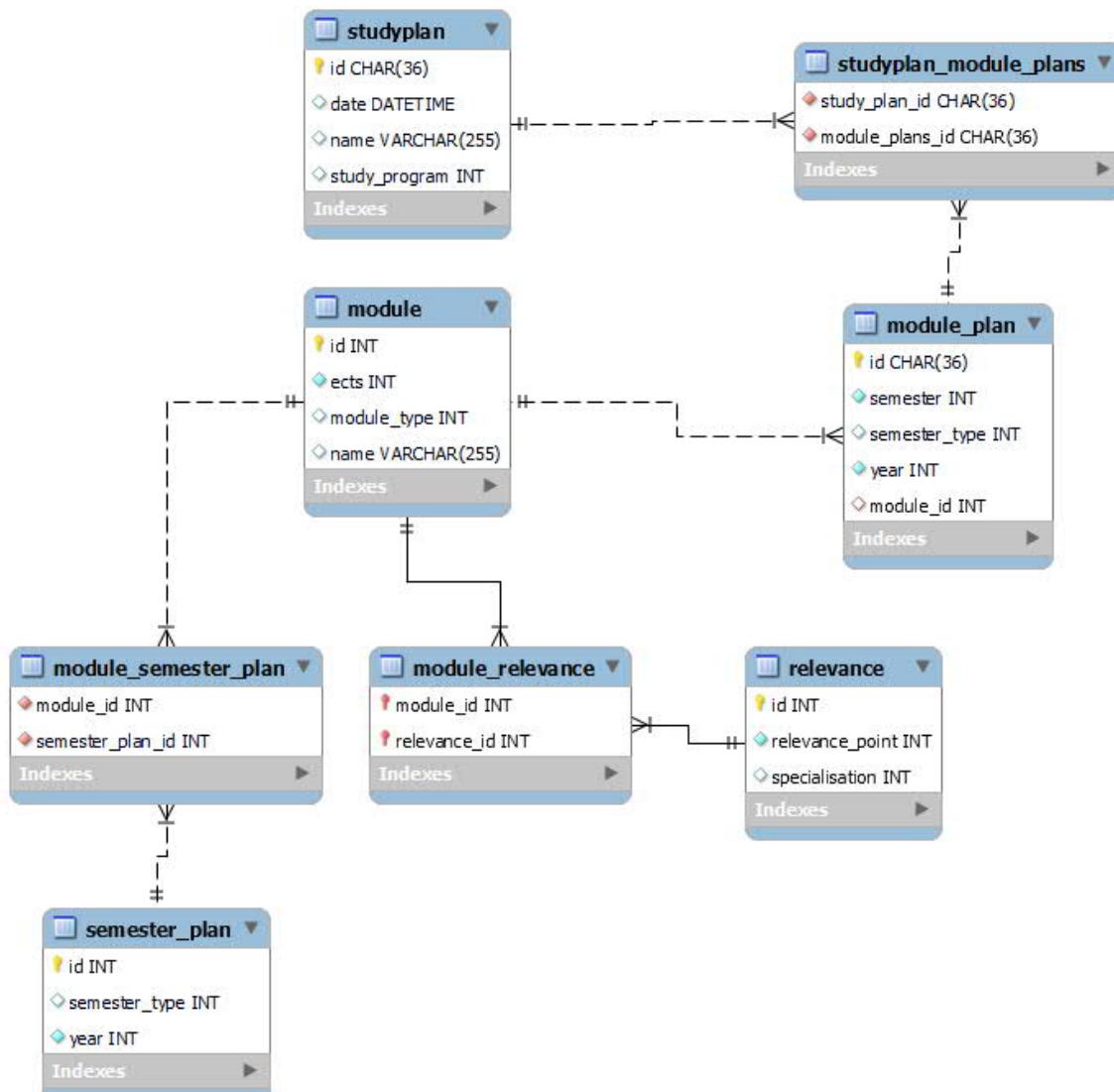
Tabelle 96: Schnittstellen Ebene 2 - Datenbank. Quelle: Autor

Schnittstelle von	Schnittstelle zu	Beschreibung
<b>Backend</b>	<b>MySQL</b>	Fordert/Fragt Daten vom MySQL Subsystem an.
<b>MySQL</b>	<b>Backend</b>	Liefert Daten/Antworten zu Backend.

## 7.5 Datenmodell

Das Datenmodell wurde vorgängig auf einem Notizzettel entworfen und die entsprechenden Klassen in Java erstellt. Da sich das Datenmodell während der gesamten Entwicklung ständig weiterentwickelt hat, wurde nach der Fertigstellung der Finalen Softwareversion (nach Erledigung der letzten User Story) das Datenmodell "Reverse Engineered". D.h. das nachfolgende Datenmodell wurde mittels MySQL-Workbench generiert.

Gesamtübersicht Version 1.0 (29.11.2021)



*Abbildung 38: Datenmodell des Studienplaners. Quelle: Autor*

## 7.5.1 Tabelle studyplan

Tabelle 97: Tabellenattribute Tabelle studyplan. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	id	CHAR	36		Eindeutige Identifikation
	date	DATETIME		x	Das Datum des Studienplans
	name	VARCHAR	255	x	Der Name des Studienplans
	study_program	INT		x	Das Studienprogramm (HF Passarelle usw.)

## 7.5.2 Tabelle studyplan\_module\_plans

Mapping-Tabelle für Relation: "Studienplan hat Modulplan".

Tabelle 98: Tabellenattribute Tabelle studyplan\_module\_plans. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
	study_plan_id	CHAR	36		Referenz zur Tabelle studyplan
	module_plans_id	CHAR	36		Referenz zur Tabelle module_plan

## 7.5.3 Tabelle module\_plan

Tabelle 99: Tabellenattribute Tabelle module\_plan. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	id	CHAR	36		Eindeutige Identifikation
	semester	INT	10		Das Semester des Modulplans
	semester_type	INT	10	x	Der Semestertyp (Herbst usw.)
	year	INT	10		Das Jahr des Modulplans
	module_id	INT	10	x	Fremdschlüssel zur Tabelle module

## 7.5.4 Tabelle module

Tabelle 100: Tabellenattribute Tabelle module. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	id	INT	10		Eindeutige Identifikation
	ects	INT	10		Die ECTS Punkte für das Modul
	module_type	INT	10	x	Der Modultyp (Basis usw.)
	name	VARCHAR	255	x	Modul-Name

## 7.5.5 Tabelle module\_relevance

Mapping-Tabelle für Relation: "Modul hat Relevanzen".

Tabelle 101: Tabellenattribute Tabelle module\_relevance. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	module_id	INT	10		Referenz zur Tabelle module
	relevance_id	INT	10		Referenz zur Tabelle relevance

## 7.5.6 Tabelle relevance

Tabelle 102: Tabellenattribute Tabelle relevance. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	id	INT	10		Eindeutige Identifikation
	relevance_point	INT	10		Die Relevanzpunkte
	specialisation	INT	10	x	Die Spezialisierung

## 7.5.7 Tabelle module\_semester\_plan

Mapping-Tabelle für Relation: "Modul hat Semesterplan".

Tabelle 103: Tabellenattribute Tabelle module\_semester\_plan. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
	module_id	INT	10		Die Referenz zur Tabelle module
	semester_plan_id	INT	10		Die Referenz zur Tabelle semester_plan

## 7.5.8 Tabelle semester\_plan

Tabelle 104: Tabellenattribute Tabelle semester\_plan. Quelle: Autor

PK	Name	Datentyp	Länge	NULL erlaubt	Beschreibung
X	id	INT	10		Eindeutige Identifikation
	semester_type	INT	10	x	Der Semestertyp (Herbst usw.)
	year	INT	10		Das Jahr

## 7.6 Klassenmodelle

Nachfolgend werden die Klassenmodelle der einzelnen Software-Schichten aufgeführt. Die Methoden-Beschreibung erfolgt mittels Java-Doc im Quellcode.

Bei einigen Klassen wird für die Identifikation eine *UUID* verwendet. Der Grund hierfür ist, dass die Wahrscheinlichkeit für ein Duplikat bei einer zufällig erzeugten *UUID* verschwindend gering ist. Bei statischen Dateien, welche nicht durch das Programm erzeugt werden (Beispielsweise die Module und die Semesterplanung) wird der Datentyp *int* für die Identifikation verwendet.

## 7.6.1 Main-Klassen

Package: ch.ffhs.pa5.backend



Abbildung 39: Klassendiagramm der BackendApplication. Quelle: Autor

Tabelle 105: Spezifikation Klasse BackendApplication. Quelle: Autor

Klasse: BackendApplication		
Diese Klasse dient als Einstieg/Startbasis für die Java-Anwendung.		
Methode	Parameter	Beschreibung
main	String[]	Parameter für die Main-Methode

Tabelle 106: Spezifikation Klasse Webconfig. Quelle: Autor

Klasse: WebConfig		
Diese Klasse dient zur Konfiguration der Web-Eigenschaften der Spring-Applikation. Hier wird die Konfiguration des Servlets vorgenommen um Cross-Origin Resource Sharing (CORS) zu ermöglichen		
Methode	Parameter/Rückgabe	Beschreibung
addCorsMappings	CorsRegistry	Einstellungen zum CORS für die gesamte Applikation.
defaultViewResolver		Gibt einen InternalResourceViewResolver zurück. Wird verwendet, um den angegebenen URI in den tatsächlichen URI aufzulösen

## 7.6.2 Model-Klassen

Package: ch.ffhs.pa5.backend.model

Einige Daten werden statisch im Programm als *Enum* hinterlegt. Für die Klassen StudyPlan, ModulePlan und Module wird das Builder-Pattern eingesetzt. Die statischen Builder-Klassen enthalten alle Felder, die in der Klasse selbst vorhanden sind. Da in den Klassen einige Felder *nullable* (als NULL) sein können, müssten zur Instanzierung der Klasse mehrere Konstruktoren erstellt werden. Dies setzen wir mittels Builder-Klassen um: leere Felder können einfach weggelassen werden. Außerdem entfällt das *Überladen* der Konstruktoren, falls die Klassen mit weiteren Felder erweitert werden (Riaan, 2016). Die Builder-Klassen werden in den nachfolgenden Auflistungen nicht näher spezifiziert, da es in unserem Fall die jeweiligen Klassen vollständig abbildet (Builder repräsentiert alle Felder, welche auch in der zu instanzierenden Klasse vorhanden sind). In den Klassen-Diagrammen sind die Builder-Klassen jedoch dargestellt.

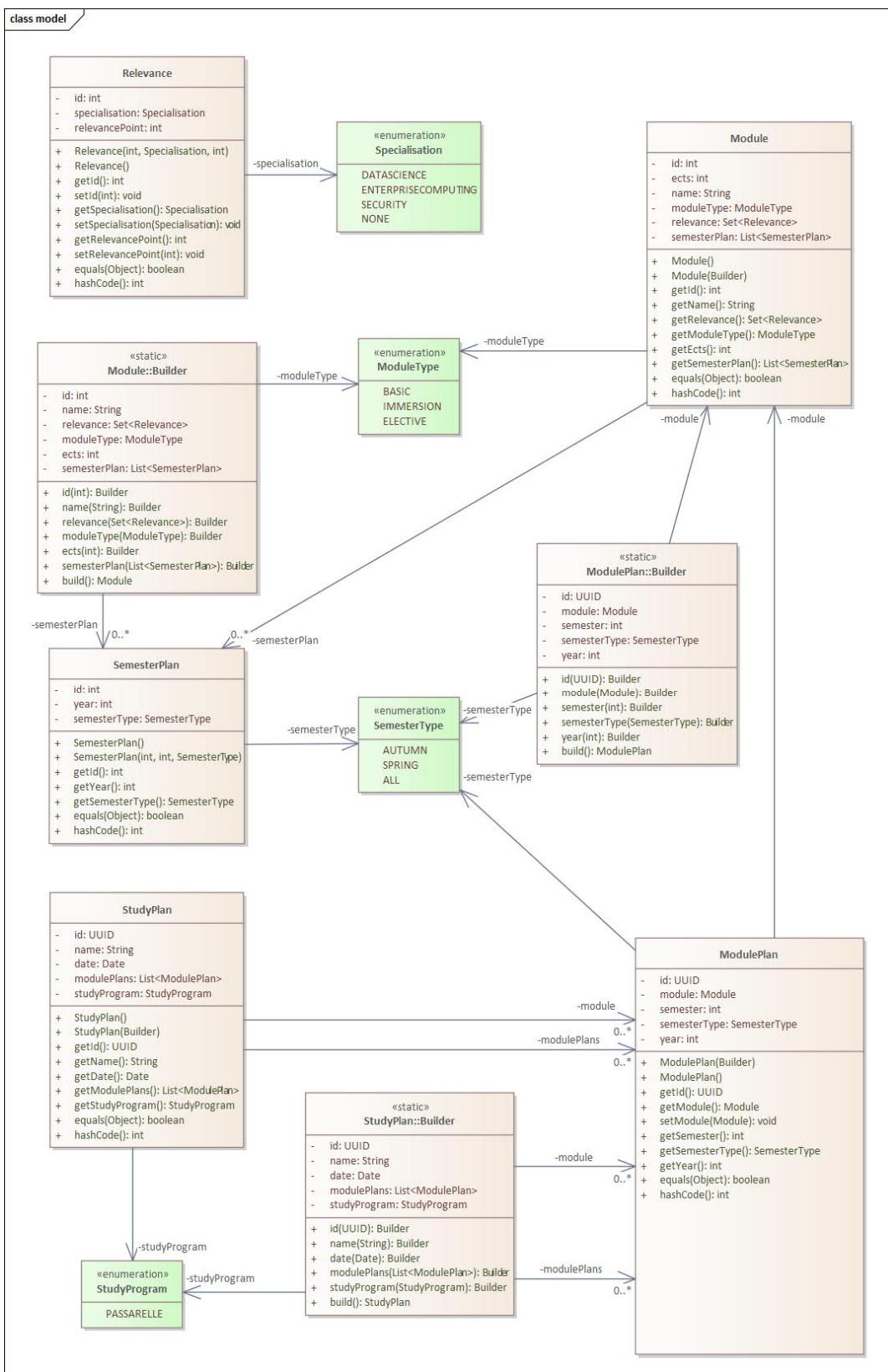


Abbildung 40 40: Klassendiagramm der Model-Klassen. Quelle: Autor

Erklärung zur Abbildung: Die durchzogenen Linien bedeuten eine Assoziation (Beschreibt eine Beziehung zwischen den Klassen). Die Klassen-Attribute werden in roter Farbe dargestellt.

Nachfolgend werden zuerst die Enum-Klassen aufgelistet und danach die weiteren Model-Klassen genauer spezifiziert. Konstruktoren und Getter-/Setter-/Hash-/Equals-Methoden werden nicht aufgeführt, da diese den Standard-Implementationen (erzeugt mittels IntelliJ IDEA) entsprechen.

Tabelle 107: Spezifikation Klasse Specialisation. Quelle: Autor

<b>Klasse: Specialisation</b>		
Die Klasse beinhaltet die Informationen der einzelnen Vertiefungsrichtungen		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
DATASCIENCE	String/int	Wert 0. Entspricht der Vertiefungsrichtung "Data Science".
ENTERPRISECOMPUTING	String/int	Wert 1. Entspricht der Vertiefungsrichtung "Enterprise Computing".
SECURITY	String/int	Wert 2. Entspricht der Vertiefungsrichtung "Informationssicherheit".
NONE	String/int	Wert 3. Entspricht der Vertiefungsrichtung "Keine". Wird lediglich für das UI benötigt.

Tabelle 108: Spezifikation Klasse ModuleType. Quelle: Autor

<b>Klasse: ModuleType</b>		
Die Klasse repräsentiert die Informationen zu den wählbaren Modultypen.		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
BASIS	String/int	Wert 0. Entspricht dem Typ "Basismodul".
IMMERSION	String/int	Wert 1. Entspricht dem Typ "Pflichtmodul".
ELECTIVE	String/int	Wert 2. Entspricht dem Typ "Wahlmodul".

Tabelle 109: Spezifikation Klasse StudyProgram. Quelle: Autor

<b>Klasse: StudyProgram</b>		
Die Klasse repräsentiert das angebotene Studiengangsprogramm. Im Moment wird nur das Modell "HF Passarelle" angeboten.		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
PASSARELLE	String/int	Wert 0. Entspricht dem Wert "HF Passarelle".

Tabelle 110: Spezifikation Klasse SemesterType. Quelle: Autor

<b>Klasse: SemesterType</b>		
Die Klasse stellt die Informationen der möglichen Semester (Herbst, Frühling) zur Verfügung.		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
AUTUMN	String/int	Wert 0. Entspricht dem Wert "Herbst".
SPRING	String/int	Wert 1. Entspricht dem Wert "Frühling".
ALL	String/int	Wert 2. Entspricht dem Wert "Alle". Wird lediglich für das UI benötigt.

Tabelle 111: Spezifikation Klasse Module. Quelle: Autor

<b>Klasse: Module</b>		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
id	int	Eindeutige Identifikation des Objekts
ects	Integer	Die Anzahl ECTS-Punkte für das Modul
name	String	Modul-Name
modulType	ModulType	Der Modultyp (Vertiefungsrichtung)(Enum).
relevance	Set<Relevance>	Eindeutige (einmalig vorkommend) Informationen zu den Relevanzpunkten. Hierzu wird ein Set verwendet, da die Relevance (z.B. 5 Relevanzpunkte in DataScience) pro Modul nur einmalig zugewiesen werden kann/darf.
semesterPlan	List<SemesterPlan>	Eindeutige (einmalig vorkommend) Information über die möglichen Semester (Beispielsweise «Herbstsemester 2021»).

Tabelle 112: Spezifikation Klasse StudyPlan. Quelle: Autor

<b>Klasse: StudyPlan</b>		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
id	UUID	Eindeutige Identifikation des Objekts.
name	String	Systemweiter eindeutiger Name des Studienplans. Wird für den Aufruf (zur Anpassung der Planung) verwendet.
date	Date (java.util.Date)	Datum der Planung. Wird beim Erstellen und Anpassen aktualisiert.
modulePlans	List<ModulePlan>	Die Liste der Modulplanung.
studyProgram	StudyProgram	Das Studienmodell (Enum).

Tabelle 113: Spezifikation Klasse ModulePlan. Quelle: Autor

<b>Klasse: ModulePlan</b>		
<b>Attribut</b>	<b>Datentyp</b>	<b>Beschreibung</b>
id	UUID	Eindeutige Identifikation des Objekts.
module	Module	Das zur Modulplanung dazugehörige Modul.
semester	int	Stellt die Information dar: zu welchem Semester wurde das Modul eingeplant.
semesterType	SemesterType	Der gewählte SemesterTyp (Herbst, Frühling), zu welchem ein Modul eingeplant wurde.
year	int	Das Jahr des eingeplanten Modules.

Tabelle 114: Spezifikation Klasse SemesterPlan. Quelle: Autor

#### Klasse: SemesterPlan

Die Klasse beinhaltet die Informationen zu der Planung der Module. Ein Modul findet nur bei bestimmten Kombinationen (Jahr und Semester) statt. Die Daten werden mittels Spring Boot beim Starten initialisiert (data.sql).

Attribut	Datentyp	Beschreibung
id	int	Eindeutige Identifikation des Objekts
year	int	Jahr (z.B. 2021)
SemesterType	SemesterType	Semester-Typ (z.B. «Herbst»)

Tabelle 115: Spezifikation Klasse Relevance. Quelle: Autor

#### Klasse: Relevance

Die Klasse beinhaltet die Informationen zu Relevanzpunkten (Relevanzpunkte pro Vertiefungsrichtung). Die Daten werden mittels Spring Boot beim Starten initialisiert (data.sql).

Attribut	Datentyp	Beschreibung
id	int	Eindeutige Identifikation des Objekts
Specialisation	Specialisation	Die Spezialisierung. Wird benötigt für die Zuordnung: «Anzahl Relevanzpunkte in der Spezialisierung xyz»
relevancePoint	Integer	Die Anzahl Relevanzpunkte.

### 7.6.3 Repository-Klassen

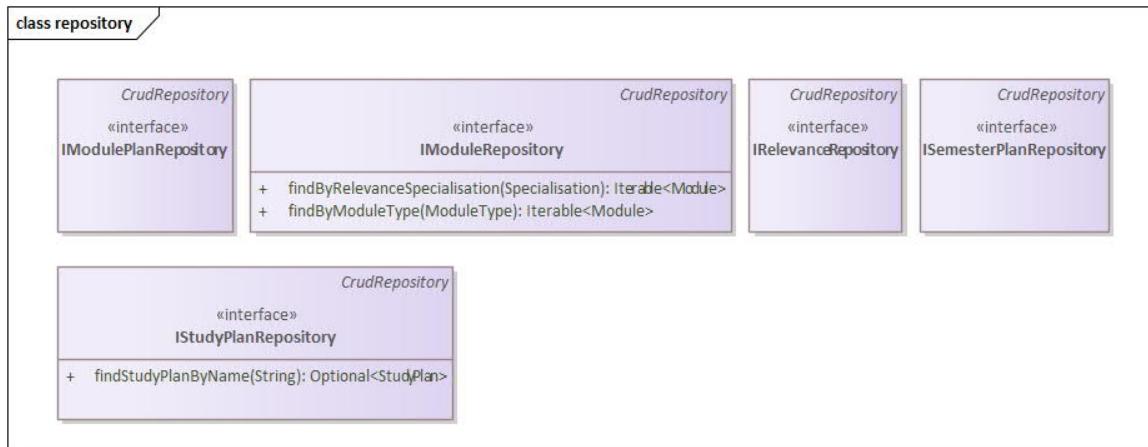


Abbildung 41: Klassendiagramm der Repository-Klassen. Quelle: Autor

## 7.6.4 Service-Klassen

**Package:** ch.ffhs.pa5.backend.service

Die Service-Klassen implementieren die eigentliche Logik des Backend-Systems. Die Aufrufe übers API werden an die Service-Klasse weitergeleitet, welche die Berechnung/Ausführung der Aufgabe übernimmt.

Die Service-Klassen wiederum benötigen (Abhängigkeit) Repository-Klassen zur Speicherung, Löschung und Abfrage der Daten aus der Datenbank-Schicht. Damit diese *Helperklassen* nicht selbst erzeugt werden müssen (*new ModuleRepository()*) und damit in den Tests keine *Repository Fake-Helperklassen* implementiert werden müssen, verwenden wir Spring für die Instanziierung dieser Klassen (auch DI – Dependency Injection genannt), wodurch diese Abhängigkeiten von einem DI-Container beim Start der Spring-Applikation injiziert werden. In Spring kann die Dependency-Injection über mehrere Wege umgesetzt werden: das erforderliche Objekt als Argument im Konstruktor definieren, Mithilfe der Setter-Methode oder die Instanziierung über ein Klassenattribut (Koller, 2021). Wir haben uns Instanziierung über den Konstruktor entschieden. Die Vorteile (die Abhängigkeiten sind gegen aussen verborgen und dass die Übersicht der Abhängigkeiten gewährleistet ist) haben wir für unser Projekt als wichtig erachtet.

Die Klassenattribute, welche für DI benötigt werden, werden in den nachfolgenden Auflistungen aufgeführt. Dabei handelt es sich bei dieser Schicht (Service-Schicht) um Abhängigkeiten zur Repository-Schicht um Datenbank-Operationen ausführen zu können.

Konstruktoren und Getter-/Setter-/Hash-/Equals-Methoden werden in den nachfolgenden Auflistungen nicht aufgeführt, da diese den Standard-Implementierungen entsprechen und nur die Felder für die DI setzen.

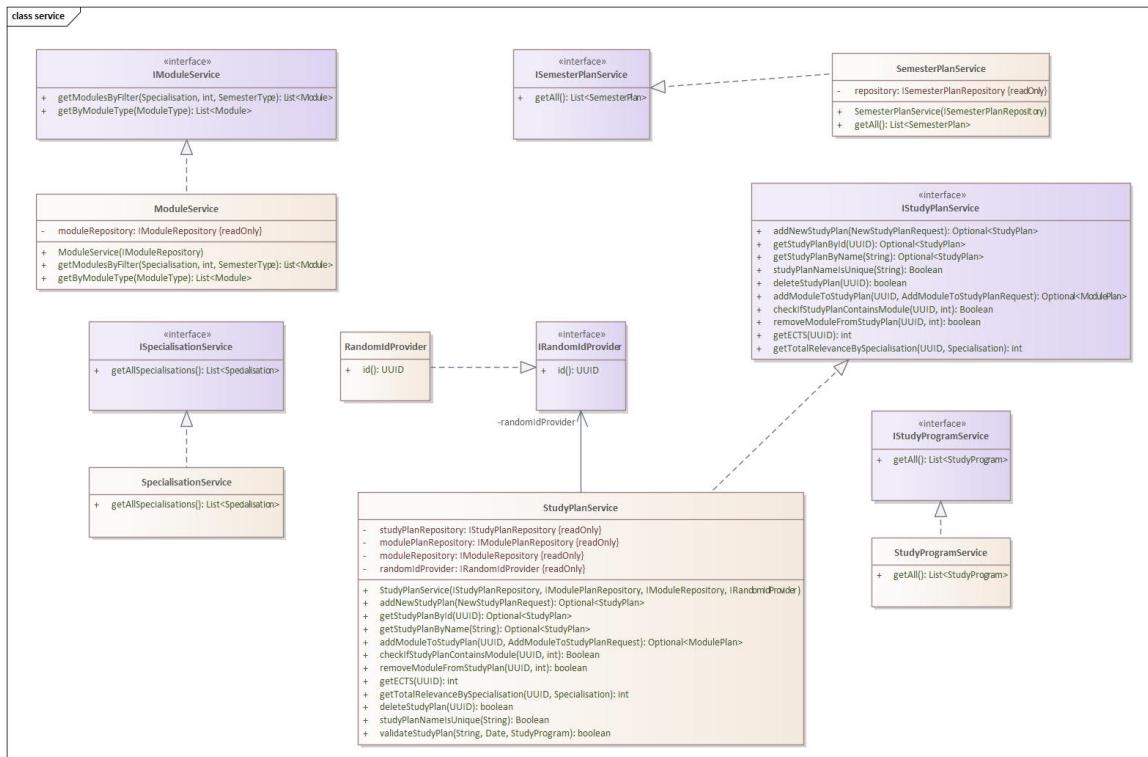


Abbildung 42: Klassendiagramm der Service-Klassen. Quelle: Autor

Erklärung zur Abbildung: Die gestrichelten Linien bedeuten eine Relation (x verwendet y). Die Klassen-Attribute werden in roter Farbe dargestellt.

Tabelle 116: Spezifikation Klasse ModuleService. Quelle: Autor

#### Klasse: ModuleService

Diese Service-Klasse ModulService dient der Abfrage der Module. Die Modul-Daten werden mittels data.sql importiert. Es stehen nur Abfragen zur Verfügung, da die Spezialisierungen nicht geändert werden können.

Attribut	Datentyp	Beschreibung
iModuleRepository	IModuleRepository	Feld für DI des Modul-Repositories

Tabelle 117: Spezifikation Methode getModulesByFilter. Quelle: Autor

#### Methode: getModulesByFilter

Diese Methode liefert auf Basis der Filter (Parameter) eine Liste von Modulen. Die Liste beinhaltet in jedem Fall Werte, da unabhängig von den Filtern die Basis-Module angefügt werden. Die Basis-Module müssen unabhängig von den Filtern zur Verfügung stehen, da diese in jedem Semester und jedem Jahr eingeplant werden können.

Parameter	Specialisation specialisation int year Semestertype semesterType
Rückgabewert	List<Module>

Tabelle 118: Spezifikation Methode getModuleType. Quelle: Autor

#### Methode: getByModuleType

Liefert eine Liste von Modulen basierend auf dem Modul-Typ. Falls zum gewählten Modul-Typ keine Module existieren, wird eine leere Liste zurückgegeben. Modul-Typen sind in einem ENUM hinterlegt.

Parameter	ModuleType moduleType
Rückgabewert	List<Module>

Tabelle 119: Spezifikation Klasse RandomIdProvider. Quelle: Autor

#### Klasse: RandomIdProvider

Diese Hilfs-Klasse dient der Erzeugung einer eindeutigen Identifikations-Nummer. Die Klasse wurde erstellt um innerhalb der Test-Klassen die Identifikations-Nummer vorgeben (Mocken) zu können.

Attribut	Datentyp	Beschreibung
Id	UUID	Eindeutige Identifikationsnummer als UUID.

Tabelle 120: Spezifikation Klasse SpecialisationService. Quelle: Autor

#### Klasse: SpecialisationService

Diese Service-Klasse SpecialisationService dient der Abfrage der Spezialisierungen. Die Spezialisierungen sind im System einprogrammiert (ENUM) und können nicht geändert werden. Es stehen nur Abfragen zur Verfügung, da die Spezialisierungen nicht geändert werden können.

Tabelle 121: Spezifikation Methode getAllSpecialisations. Quelle: Autor

#### Methode: getAllSpecialisations

Diese Methode liefert alle Spezialisierungen des Systems zurück. Diese Methode liefert in jedem Fall eine nicht leere Liste zurück. Die Spezialisierungen sind in einem ENUM hinterlegt.

Parameter	-
Rückgabewert	List<Spezialisation>

Tabelle 122: Spezifikation Klasse StudyPlanService. Quelle: Autor

#### Klasse: StudyPlanService

Diese Service-Klasse StudyPlanService dient der Erstellung, Abfrage, Anpassung und Löschung des Studienplans. Die Eindeutige Identifikation des Studienplans findet über die Identifikationsnummer (UUID) statt. Der Studienplan wird auf dem UI erstellt und verwaltet.

Attribut	Datentyp	Beschreibung
iStudyPlanRepository	IStudyPlanRepository	Feld für DI des StudyPlan-Repositories
iModulePlanRepository	IModulePlanRepository	Feld für DI des ModulPlan-Repositories
iModuleRepository	IModuleRepository	Feld für DI des Modul-Repositories
iRandomIdProvier	IRandomIdProvider	Feld für DI des UUID-Generators

Tabelle 123: Spezifikation Methode addNewStudyPlan. Quelle: Autor

#### Methode: addNewStudyPlan

Erstellt einen neuen Studienplan auf Basis der Werte, welche über UI eingegeben wurden. Es wird automatisch eine ID erstellt, welche für die Weiterverwendung benötigt wird. Die Werte, welche auf dem UI eingegeben wurden, werden hier geprüft. Falls die eingegebenen Daten nicht gültig sind, wird eine leere Option zurückgegeben.

Parameter	NewStudyPlanRequest studyPlanRequest
Rückgabewert	Optional<StudyPlan>

Tabelle 124: Spezifikation Methode getStudyPlanId. Quelle: Autor

#### Methode: getStudyPlanById

Liefert den gewünschten Studienplan auf Basis der Identifikations-Nummer. Das Ergebnis kann leer sein (leere Option), falls der Studienplan nicht existiert.

Parameter	UUID studyPlanId
Rückgabewert	Optional<StudyPlan>

Tabelle 125: Spezifikation Methode getStudyPlanByName. Quelle: Autor

#### Methode: getStudyPlanByName

Liefert den gewünschten Studienplan auf Basis des Namens. Das Ergebnis kann leer sein (leere Option), falls der Studienplan nicht existiert.

Parameter	String studyPlanName
Rückgabewert	Optional<StudyPlan>

Tabelle 126: Spezifikation Methode addModuleToStudyPlan. Quelle: Autor

#### Methode: addModuleToStudyPlan

Fügt dem Modulplan eines Studienplans ein Modul hinzu. Das Ergebnis kann leer sein (leere Option), falls der Studienplan oder das Modul nicht existiert.

Parameter	UUID studyPlanId AddModuleToStudyPlanRequest modulePlanningRequest
Rückgabewert	Optional<StudyPlan>

Tabelle 127: Spezifikation Methode checkIfStudyPlanContainsModule. Quelle: Autor

<b>Methode: checkIfStudyPlanContainsModule</b>	
Prüft anhand der Identifikations-Nummern vom Studienplan und des Moduls ob dieses bereits vorhanden ist.	
<b>Parameter</b>	UUID studyPlanId int moduleId
<b>Rückgabewert</b>	Boolean

Tabelle 128: Spezifikation Methode removeModuleFromStudyPlan. Quelle: Autor

<b>Methode: removeModuleFromStudyPlan</b>	
Entfernt ein Modul aus dem Modulplan eines Studienplans. Kann FALSE sein, wenn der Studienplan oder das Modul im Modulplan nicht gefunden wird.	
<b>Parameter</b>	UUID studyPlanId int moduleId
<b>Rückgabewert</b>	Boolean

Tabelle 129: Spezifikation Methode getECTS. Quelle: Autor

<b>Methode: getECTS</b>	
Liefert das Total als Zahlenwerts der ECTS-Punkte im Studienplan.	
<b>Parameter</b>	UUID studyPlanId
<b>Rückgabewert</b>	int

Tabelle 130: Spezifikation Methode getTotalRelevanceBySpecialisation. Quelle: Autor

<b>Methode: getTotalRelevanceBySpecialisation</b>	
Liefert das Total als Zahlenwerts der Referenzpunkte im Studienplan pro Spezialisierung.	
<b>Parameter</b>	UUID studyPlanId Specialisation specialisation
<b>Rückgabewert</b>	int

Tabelle 131: Spezifikation Klasse StudyPlanService. Quelle: Autor

<b>Klasse: StudyPlanService</b>	
Dieser Service-Klasse dient der Abfrage aller vorhandenen Studienprogramme. Im Moment wird nur das Modell "HF Passarelle" angeboten. Die Spezialisierungen sind im System einprogrammiert (ENUM) und können nicht geändert werden. Es stehen nur Abfragen zur Verfügung, da die Spezialisierungen nicht geändert werden können.	

Tabelle 132: Spezifikation Methode getAll. Quelle: Autor

<b>Methode: getAll</b>	
Liefert das Total als Zahlenwerts der Referenzpunkte im Studienplan pro Spezialisierung.	
<b>Parameter</b>	-
<b>Rückgabewert</b>	List<StudyProgram>

## 7.6.5 Controller-Klassen

Die Controller-Klassen repräsentieren die API-Schicht gegen aussen. Sie bieten den Konsumenten (UI) die notwendigen Schnittstellen, zur Umsetzung der User-Stories und Use-Cases an.

Die API-Endpunkte wurden als Rest-Schnittstellen umgesetzt. Dabei wurden folgende http-Methoden für die Anfragen verwendet:

Tabelle 133: Spring Controller HTTP-Methoden und Annotation. Quelle: Autor

http-Methode	Spring-Annotation	Beschreibung/CRUD-Operation
<b>POST</b>	@PostMapping	Ressourcen erstellen. Mittels der Post-Methoden können Ressourcen (Studienpläne, Module usw.) angelegt/erstellt werden. Die CRUD-Operation die in der Service-Schicht ausgeführt wird, ist dabei die repository.save()-Methode.
<b>GET</b>	@GetMapping	Ressource abfragen. Mittels der Get-Methode können Ressourcen (Studienpläne, Module usw.) abgefragt werden. Die CRUD-Operation die in der Service-Schicht ausgeführt wird, ist dabei die repository.find()-Methode.
<b>DELETE</b>	@DeleteMapping	Ressource löschen. Mittels der Delete-Methode können Ressourcen (Studienpläne, Module usw.) gelöscht werden. Die CRUD-Operation die in der Service-Schicht ausgeführt wird, ist dabei die repository.delete()-Methode.

Die Controller benötigen, wie die Service-Klassen Abhängigkeiten, um die Ausführung der Aufgabe in die nächst *tiefere* Schicht weiterzuleiten (Service-Klassen). Die Abhängigkeiten werden im Konstruktor als DI (Dependency Injection) automatisch von Spring initialisiert. Nachfolgende Abbildung stellt die Controller-Klassen grafisch als Klassendiagramm dar:

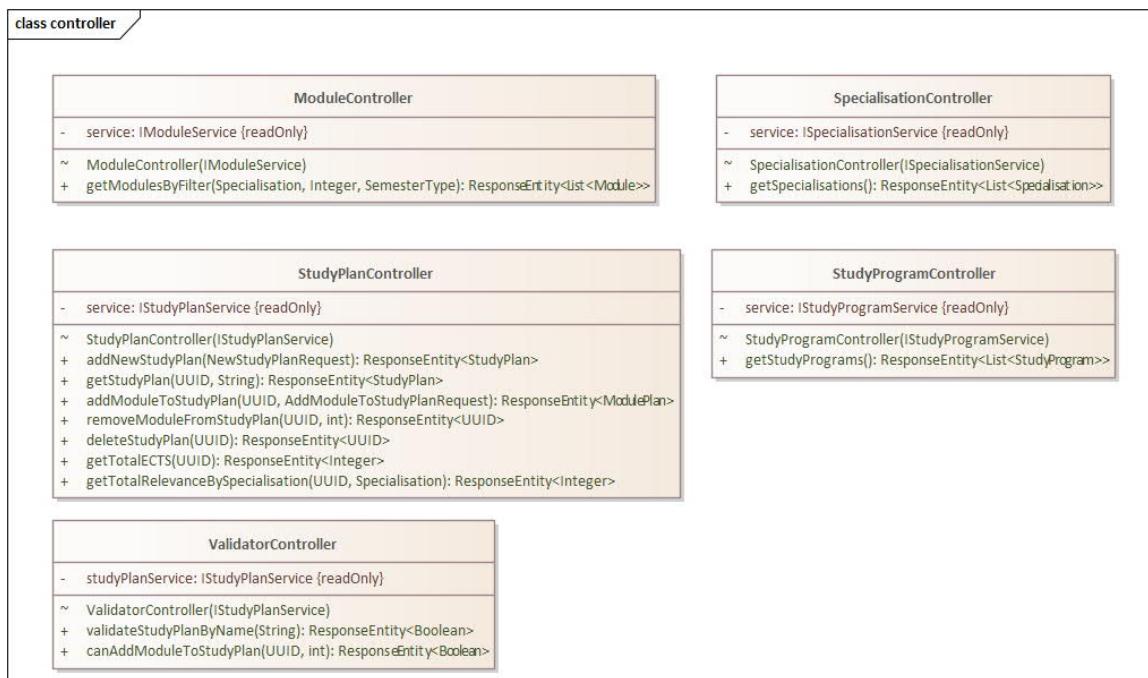


Abbildung 43: Klassendiagramm der Controller-Klassen

Erklärung zur Abbildung: Klassen-Attribute werden in roter Farbe dargestellt.

Tabelle 134: Spezifikation Klasse ModuleController. Quelle: Autor

#### Klasse: ModuleController

Der ModuleController stellt die API zur Verfügung, welche vom Frontend HTTP-Requests entgegennimmt und diese an den jeweiligen Service weiterleitet. Ein Modul kann belegt werden, um sich in einem Thema weiterzubilden und Referenzpunkte sowie ECTS-Punkte zu erhalten.

Attribut	Datentyp	Beschreibung
Service	IModuleService	Feld welcher den Zugriff auf den Service zur Verwaltung der Modul-Daten auf Service-Schicht ermöglicht.

Tabelle 135: Spezifikation Methode getModulesByFilter. Quelle: Autor

#### Methode: getModulesByFilter

Dieser Endpunkt liefert eine Liste von Modulen auf Basis der definierten Filter.

Parameter	Specialisation specialisation int year Semestertype semesterType
Rückgabewert	List<Module>

Tabelle 136: Spezifikation Klasse StudyProgramController. Quelle: Autor

#### Klasse: StudyProgramController

Der StudyProgramController stellt die API zur Verfügung, welche vom Frontend HTTP-Requests entgegen nimmt und diese an den jeweiligen Service weiterleitet. Ein StudyProgram gibt die Art des Studienganges an.

Attribut	Datentyp	Beschreibung
Service	IStudyProgramService	Feld welcher den Zugriff auf den Service zur Verwaltung der StudyProgram-Daten auf Service-Schicht ermöglicht.

Tabelle 137: Spezifikation Methode getStudyPrograms. Quelle: Autor

#### Methode: getStudyPrograms

Liefert alle Studienprogramme. Im Moment steht nur die Passarelle HF zur Verfügung.

Parameter	-
Rückgabewert	List<StudyProgram>

Tabelle 138: Spezifikation Klasse SpecialisationController. Quelle: Autor

#### Klasse: SpecialisationController

Der SpecialisationController stellt die API zur Verfügung, welche vom Frontend HTTP-Requests entgegennimmt und diese an den jeweiligen Service weiterleitet. Eine Spezialisierung gibt an, für welche Fachrichtung Referenzpunkte erzielt werden müssen um, entsprechende Spezialisierung zu erfüllen.

Attribut	Datentyp	Beschreibung
Service	ISpecialisationService	Feld welcher den Zugriff auf den Service zur Verwaltung der Specialisation-Daten auf Service-Schicht ermöglicht.

Tabelle 139: Spezifikation Methode getSpecialisations. Quelle: Autor

#### Methode: getSpecialisations

Liefert eine Liste aller Spezialisierungen.

Parameter	-
Rückgabewert	List<Specialisation>

Tabelle 140: Spezifikation Klasse ValidatorController. Quelle: Autor

#### Klasse: ValidatorController

ValidatorController prüft, ob der gewünschte Name des Studienplans schon vorhanden/vergeben ist und ein bestimmtes Modul bereits einem Studienplan zugeordnet ist.

Attribut	Datentyp	Beschreibung
Service	IStudyPlanService	Feld welcher den Zugriff auf den Service zur Verwaltung der StudyPlan-Daten auf Service-Schicht ermöglicht.

Tabelle 141: Spezifikation Methode validateStudyPlanByName. Quelle: Autor

#### Methode: validateStudyPlanByName

Liefert die Antwort, ob der Namen eines Studienplans gültig (noch nicht verwendet) ist.

Parameter	String studyPlanName
Rückgabewert	Boolean

Tabelle 142: Spezifikation Methode canAddModuleToStudyPlan. Quelle: Autor

#### Methode: canAddModuleToStudyPlan

Liefert die Antwort, ob ein Modul (Parameter: moduleId) bereits im Studienplan vorhanden ist (Parameter: studyPlanId).

Parameter	UUID studyPlanId int moduleId
Rückgabewert	Boolean

Tabelle 143: Spezifikation Klasse StudyPlanController. Quelle: Autor

#### Klasse: StudyPlanController

Der StudyPlanController stellt die API zur Verfügung, welche vom Frontend HTTP-Requests entgegennimmt und diese an den jeweiligen Service weiterleitet. Ein StudyPlan enthält alle Informationen die, einem Studienplan beinhalten, wie z.B. Module, Spezialisierung. usw.

Attribut	Datentyp	Beschreibung
Service	IStudyPlanService	Feld welcher den Zugriff auf den Service zur Verwaltung der StudyPlan-Daten auf Service-Schicht ermöglicht.

Tabelle 144: Spezifikation Methode addNewStudyPlan. Quelle: Autor

#### Methode: addNewStudyPlan

Erstellt einen neuen Studienplan anhand eines StudyPlanRequest.

Parameter	NewStudyPlanRequest studyPlanRequest
Rückgabewert	StudyPlan

Tabelle 145: Spezifikation Methode getStudyPlan. Quelle: Autor

#### Methode: getStudyPlan

Liefert den Studienplan aufgrund des mitgegebenen Parameters. Es kann entweder eine UUID oder der Name mitgegeben werden.

Parameter	UUID moduleId oder String name
Rückgabewert	StudyPlan

Tabelle 146: Spezifikation Methode addModuleToStudyPlan. Quelle: Autor

<b>Methode: addModuleToStudyPlan</b>	
Fügt dem Studienplan (Parameter: studyPlanId) ein Modul hinzu.	
<b>Parameter</b>	UUID moduleId AddModuleToStudyPlanRequest module
<b>Rückgabewert</b>	ModulePlan

Tabelle 147: Spezifikation Methode removeModuleFromStudyPlan. Quelle: Autor

<b>Methode: removeModuleFromStudyPlan</b>	
Entfernt das ausgewählte Modul (Parameter: moduleId) vom Studienplan (Parameter: studyPlanId).	
<b>Parameter</b>	UUID studyPlanId int moduleId
<b>Rückgabewert</b>	ModulePlan

Tabelle 148: Spezifikation Methode deleteStudyPlan. Quelle: Autor

<b>Methode: deleteStudyPlan</b>	
Entfernt das ausgewählte Modul (Parameter: moduleId) vom Studienplan (Parameter: studyPlanId).	
<b>Parameter</b>	UUID studyPlanId
<b>Rückgabewert</b>	UUID

Tabelle 149: Spezifikation Methode getTotalECTS. Quelle: Autor

<b>Methode: getTotalECTS</b>	
Liefert die totale Anzahl an ECTS Punkten zu einem StudienPlan (Parameter: studyPlanId)	
<b>Parameter</b>	UUID studyPlanId
<b>Rückgabewert</b>	Integer

Tabelle 150: Spezifikation Methode getTotalRelevanceBySpecialisation. Quelle: Autor

<b>Methode: getTotalRelevanceBySpecialisation</b>	
Liefert die totale Anzahl an Referenzpunkte pro Spezialisierung (Parameter: specialisation) zu einem StudienPlan (Parameter: studyPlanId).	
<b>Parameter</b>	UUID studyPlanId Specialisation specialisation
<b>Rückgabewert</b>	Integer

## 7.6.6 Controller-Model-Klassen

Package: ch.ffhs.pa5.backend.model

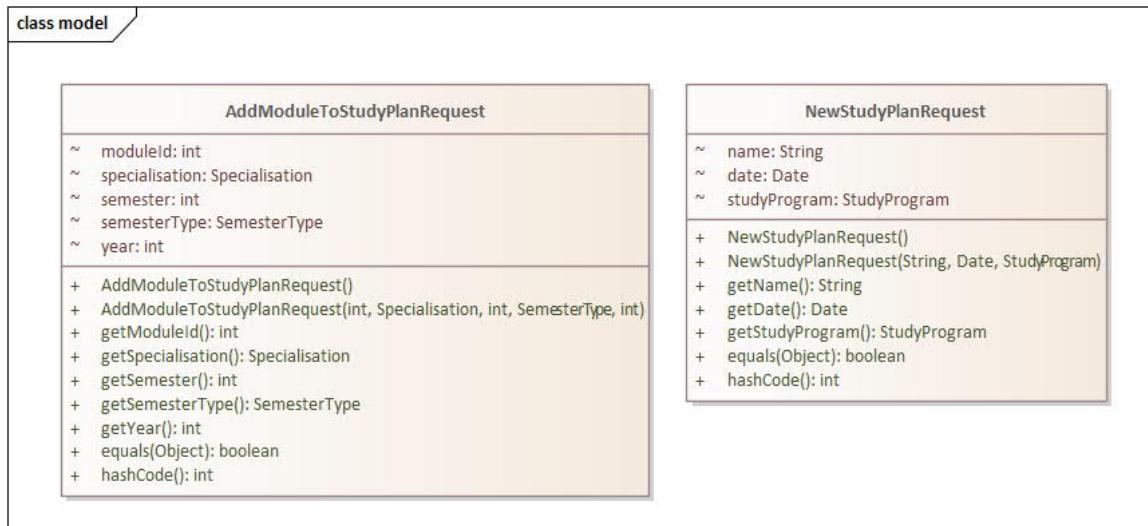


Abbildung 44: Klassendiagramm der Model-Klassen (innerhalb der Controller-Schicht). Quelle: Autor

Tabelle 151: Spezifikation Klasse AddModuleToStudyPlanRequest. Quelle: Autor

### Klasse: AddModuleToStudyPlanRequest

Diese Klasse dient als Hilfsklasse zum Hinzufügen eines Modules zu einem Studienplan.

Attribut	Datentyp	Beschreibung
moduleId	Int	Die ID des hinzuzufügenden Modules.
specialisation	Specialisation	Die Specialisation, zu dem das Modul zugeordnet wird.
semester	int	Das Semester, zu dem das Modul zugeordnet wird.
semesterType	SemesterType	Der SemesterType, zu dem das Modul zugeordnet wird.
year	int	Das Jahr, zu dem das Modul zugeordnet wird.

Tabelle 152: Spezifikation Klasse NewStudyPlanRequest. Quelle: Autor

### Klasse: NewStudyPlanRequest

Diese Klasse dient als Hilfsklasse für die Anfrage zur Erstellung eines neuen Studienplans.

Attribut	Datentyp	Beschreibung
name	String	Der Name des neuen Studienplans.
date	Date	Das Erstellungsdatum des neuen Studienplans.
studyProgram	StudyProgram	Das Studienprogramm des neuen Studienplans.

## 7.6.7 Exceptions

**Package:** ch.ffhs.pa5.backend.controller.exception

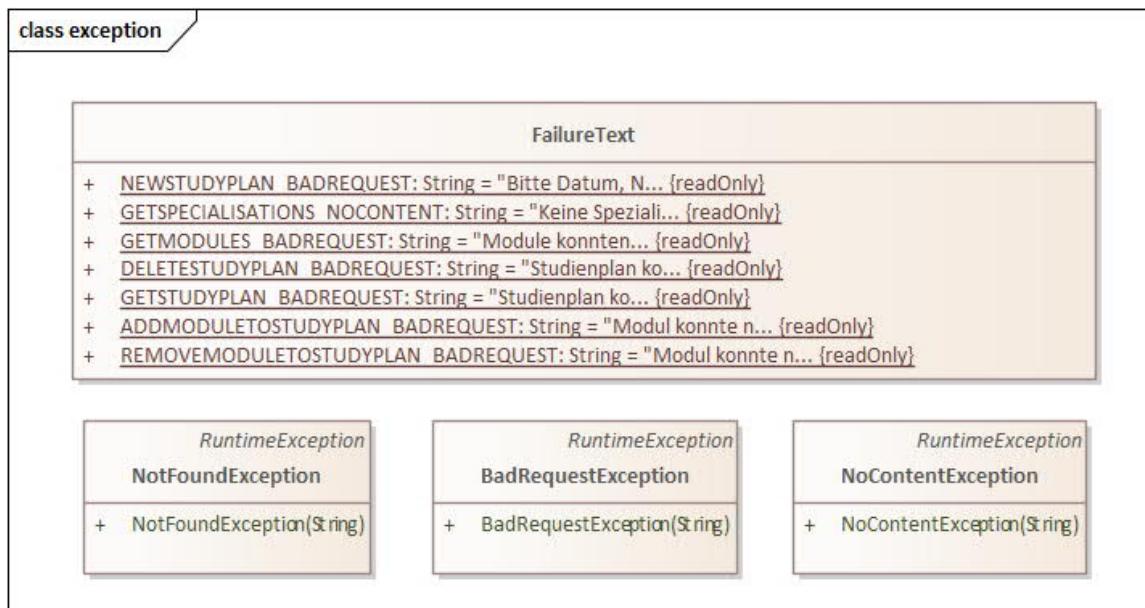


Abbildung 45: Klassendiagramm der Exception-Klassen. Quelle: Autor

Tabelle 153: Spezifikation Klasse FailureText. Quelle: Autor

### Klasse: FailureText

Diese Klasse dient als Hilfsklasse für die Anfrage zur Erstellung eines neuen Studienplans.

Attribut	Parameter	Beschreibung
FEHLERTEXTE	String	Diverse FehlerTexte für http-Fehler (204 und 400)

Tabelle 154: Spezifikation Exception-Klassen. Quelle: Autor

### Klasse: NotFoundException, BadRequestException und NoContentException

Diese Klassen dienen für die Fehlerausgabe auf den Controller-Klassen zur Rückgabe eines definierten Fehlertextes

## 7.7 Schnittstellenbeschreibung

In der Schnittstellenbeschreibung werden gemäss Aufgabenstellung (FFHS, Vorgaben zur Projektarbeit, 2021) pro Anwendungsfall die die ein- und ausgehenden Daten, Objekte und Ereignisse aufgelistet.

- Studiengang planen
  - Modul entfernen
  - Geplante Relevanzpunkte sichten
  - Geplante ECTS-Punkte sichten
  - Modul Informationen sichten
    - Relevanzpunkte pro Vertiefung sichten
    - Semesterplan sichten
- Studienplan laden
  - Studienplan mittels Namen laden
  - Studiengang planen
- Studienplan visualisieren
- Studienplan speichern
- Studienplan löschen

Die vollständige API-Dokumentation ist auch als HTML-Datei im Abgabeordner im Verzeichnis "API-Dokumentation" verfügbar.

### 7.7.1 Studienplan erstellen

Tabelle 155: API - ValidatorController. Quelle: Autor

API	GET <code>/api/v1/validator/validateStudyPlan?studyPlanName={studyPlanName}</code>
Eingabewert	String
Rückgabewert	Boolean
Fehlerfall	False
Beschreibung	Prüfen, ob der gewünschte Name des Studienplans schon vorhanden/vergeben ist.

Tabelle 156: API - StudyPlanController. Quelle: Autor

API	POST api/v1/studyPlans/
Eingabewert	NewStudyPlanRequest
Rückgabewert	StudyPlan
	<pre>{   "name": "string",   "date": "2021-12-10T08:46:05.207Z",   "studyProgram": "PASSARELLE" }</pre>
Fehlerfall	HTTP 400 mit Fehlerbeschreibung: " Bitte Datum, Name und Programm prüfen. Studienplan konnte nicht erstellt werden".
Beschreibung	Erstellung eines neuen Studienplans

Tabelle 157: API - StudyProgramController. Quelle: Autor

API	POST api/v1/studyPrograms/
Rückgabewert	List<StudyProgram>
	<pre>[   "PASSARELLE" ]</pre>
Fehlerfall	Leere Liste
Beschreibung	Liste aller vorhanden Studienprogramme erhalten

## 7.7.2 Studienplanung anzeigen

Tabelle 158: API - StudyPlanController. Quelle: Autor

API	GET api/v1/studyPlans?studyPlanId={studyPlanId}
<b>Eingabewert</b>	UUID studyPlanId
<b>Rückgabewert</b>	<p>StudyPlan</p> <pre>{     "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",     "name": "string",     "date": "2021-12-10T08:34:08.539Z",     "modul ePl ans": [         {             "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",             "modul e": {                 "id": 0,                 "ects": 0,                 "name": "string",                 "modul eType": "BASIC",                 "rel evance": [                     {                         "id": 0,                         "speci al i sati on": "DATASCI ENCE",                         "rel evancePoi nt": 0                     }                 ],                 "semesterPlan": [                     {                         "id": 0,                         "year": 0,                         "semesterType": "AUTUMN"                     }                 ]             },             "semester": 0,             "semesterType": "AUTUMN",             "year": 0         }     ],     "studyProgram": "PASSARELLE" }</pre>
<b>Fehlerfall</b>	HTTP 204 mit Fehlerbeschreibung: "Studienplan konnte nicht gefunden werden".
<b>Beschreibung</b>	Abfragen des Studienplans mittels UUID.

Tabelle 159: API - StudyPlanController. Quelle: Autor

<b>API</b>	GET api/v1/studyPlans?studyPlanName={studyPlanName}
<b>Eingabewert</b>	String studyPlanName
<b>Rückgabewert</b>	<pre>{     "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",     "name": "string",     "date": "2021-12-10T08:34:08.539Z",     "modul ePlans": [         {             "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",             "module": {                 "id": 0,                 "ects": 0,                 "name": "string",                 "moduleType": "BASIC",                 "relevance": [                     {                         "id": 0,                         "specialisation": "DATASCIENCE",                         "relevancePoint": 0                     }                 ],                 "semesterPlan": [                     {                         "id": 0,                         "year": 0,                         "semesterType": "AUTUMN"                     }                 ],                 "semester": 0,                 "semesterType": "AUTUMN",                 "year": 0             }         ],         "studyProgram": "PASSARELLE"     ] }</pre>
<b>Fehlerfall</b>	HTTP 204 mit Fehlerbeschreibung: "Studienplan konnte nicht gefunden werden".
<b>Beschreibung</b>	Abfragen des Studienplans mittels des Namens.

Tabelle 160: API - SpecialisationController. Quelle: Autor

<b>API</b>	GET api/v1/specialisations
<b>Rückgabewert</b>	List<Specialisation>
	<pre>[     "DATASCIENCE" ]</pre>
<b>Fehlerfall</b>	HTTP 204 mit Fehlerbeschreibung: "Keine Spezialisierungen vorhanden".
<b>Beschreibung</b>	Damit der User eine Vertiefungsrichtung auswählen kann, werden auf dem Frontend alle verfügbaren Vertiefungsrichtungen angezeigt. Diese werden über diese Schnittstelle abgerufen.

Tabelle 161: API - SpecialisationController. Quelle: Autor

<b>API</b>	<b>GET</b> <code>api/v1/modules?specialisation={specialisation}&amp;year={year}&amp;semestertype={semesterType}</code>
<b>Eingabewert</b>	Specialisation specialization int year SemesterType semesterType
<b>Rückgabewert</b>	List<Module> <pre>[   {     "id": 0,     "ects": 0,     "name": "string",     "moduleType": "BASIC",     "relevance": [       {         "id": 0,         "specialization": "DATASCIENCE",         "relevancePoint": 0       }     ],     "semesterPlan": [       {         "id": 0,         "year": 0,         "semesterType": "AUTUMN"       }     ]   } ]</pre>
<b>Fehlerfall</b>	HTTP 204 mit Fehlerbeschreibung: "Keine Spezialisierungen vorhanden".
<b>Beschreibung</b>	Diese Schnittstelle dient dem User zur Anzeige der Module der gewählten Vertiefungsrichtung. Damit alle Module angezeigt werden können (wenn keine Vertiefungsrichtung gewählt wird), steht die Konstante "NONE" zur Verfügung.

### 7.7.3 Modul eintragen

Tabelle 162: API - StudyPlanController. Quelle: Autor

API	<b>POST api/v1/studyPlans/{studyPlanId}/modules</b>
<b>Eingabewert</b>	<pre>UUID studyPlanId AddModuleToStudyPlanRequest module {     "moduleId": 0,     "specialisation": "DATASCIENCE",     "semester": 0,     "semesterType": "AUTUMN",     "year": 0 }</pre>
<b>Rückgabewert</b>	<pre>ModulePlan {     "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",     "module": {         "id": 0,         "ects": 0,         "name": "string",         "moduleType": "BASIC",         "relevance": [             {                 "id": 0,                 "specialisation": "DATASCIENCE",                 "relevancePoint": 0             }         ],         "semesterPlan": [             {                 "id": 0,                 "year": 0,                 "semesterType": "AUTUMN"             }         ],         "semester": 0,         "semesterType": "AUTUMN",         "year": 0     } }</pre>
<b>Fehlerfall</b>	HTTP 400 mit Fehlerbeschreibung: "Modul konnte nicht hinzugefügt werden".
<b>Beschreibung</b>	Modul zu Studienplanung hinzufügen.

Tabelle 163: API - ValidatorController. Quelle: Autor

API	<b>GET /api/v1/validator/canAdd/studyPlan/{studyPlanId}/module/{moduleId}</b>
<b>Eingabewert</b>	UUID studyPlanId int moduleId
<b>Rückgabewert</b>	Boolean
<b>Fehlerfall</b>	-
<b>Beschreibung</b>	Liefert die Antwort, ob ein Modul (Parameter: moduleId) bereits im Studienplan vorhanden ist (Parameter: studyPlanId).

## 7.7.4 Modul entfernen

Tabelle 164: API - StudyPlanController. Quelle: Autor

API	<b>DELETE api/v1/studyPlans/{studyPlanId}/module/{moduleId}</b>
<b>Eingabewert</b>	UUID studyPlanId int moduleId
<b>Rückgabewert</b>	UUID
<b>Fehlerfall</b>	HTTP 400 mit Fehlerbeschreibung: "Studienplan konnte nicht angepasst werden".
<b>Beschreibung</b>	Modulplanung updaten.

## 7.7.5 Studienplan mittels Namen laden

Tabelle 165: API - StudyPlanController. Quelle: Autor

API	<b>GET api/v1/studyPlans?studyPlanName={studyPlanName}</b>
<b>Eingabewert</b>	String studyPlanName
<b>Rückgabewert</b>	StudyPlan <pre>{     "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",     "name": "string",     "date": "2021-12-11T08:49:13.147Z",     "modules": [         {             "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",             "module": {                 "id": 0,                 "ects": 0,                 "name": "string",                 "moduleType": "BASIC",                 "relevance": [                     {                         "id": 0,                         "specialization": "DATASCIENCE",                         "relevancePoint": 0                     }                 ],                 "semesterPlan": [                     {                         "id": 0,                         "year": 0,                         "semesterType": "AUTUMN"                     }                 ]             },             "semester": 0,             "semesterType": "AUTUMN",             "year": 0         }     ],     "studyProgram": "PASSARELLE" }</pre>
<b>Fehlerfall</b>	HTTP 204 mit Fehlerbeschreibung: "Studienplan konnte nicht gefunden werden".
<b>Beschreibung</b>	Abfragen des Studienplans mittels des Namens.

## 7.7.6 Studienplan löschen

Tabelle 166: API - StudyPlanController. Quelle: Autor

API	<b>DELETE api/v1/studyPlans/{studyPlanId}</b>
<b>Eingabewert</b>	UUID studyPlanId
<b>Rückgabewert</b>	UUID studyPlanId
<b>Fehlerfall</b>	HTTP 400 mit Fehlerbeschreibung: "Studienplan konnte nicht gelöscht werden".
<b>Beschreibung</b>	Studienplan löschen.

## 7.7.7 Studienplan visualisieren

Tabelle 167: API - StudyPlanController. Quelle: Autor

API	<b>GET api/v1/studyPlan/{studyPlanId}/modules/ects</b>
<b>Eingabewert</b>	UUID studyPlanId
<b>Rückgabewert</b>	Integer
<b>Fehlerfall</b>	-
<b>Beschreibung</b>	Liefert die totale Anzahl an ECTS Punkten zu einem StudienPlan (Parameter: studyPlanId)

Tabelle 168: API - StudyPlanController. Quelle: Autor

API	<b>GET api/v1/studyPlan/{studyPlanId}/specialisations/{specialization}</b>
<b>Eingabewert</b>	UUID studyPlanId
<b>Rückgabewert</b>	Integer
<b>Fehlerfall</b>	-
<b>Beschreibung</b>	Liefert die totale Anzahl an Referenzpunkte pro Spezialisierung (Parameter: specialization) zu einem StudienPlan (Parameter: studyplanId)

## 7.8 Dynamische Modelle

Zustandsdiagramm: Studienplan

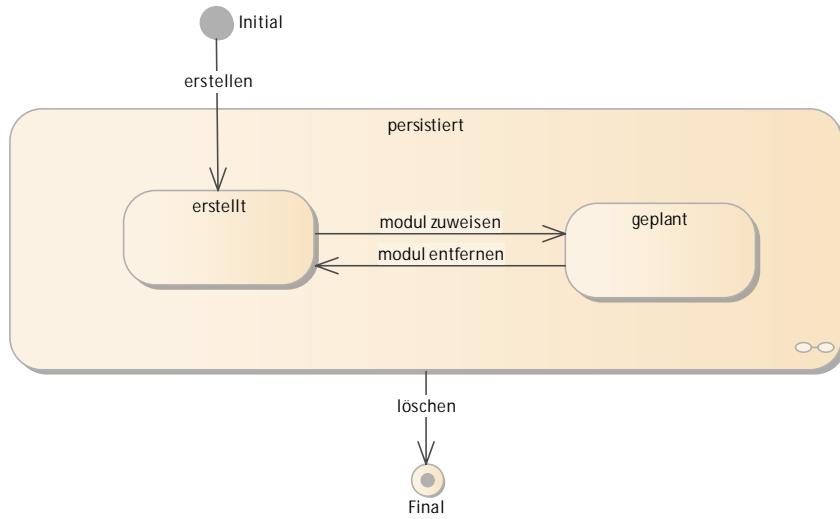


Abbildung 46: Zustandsdiagramm - Studienplan. Quelle: Autor

Ein Studienplan wird erstellt und kann anschliessend mit Modulen befüllt werden. Es gibt zwei Zustände für einen Studienplan. Entweder er beinhaltet bereits Module oder ist neu erstellt worden und enthält noch keine Module.

## 7.8.1 Aktivitätsdiagramm: Studienplan

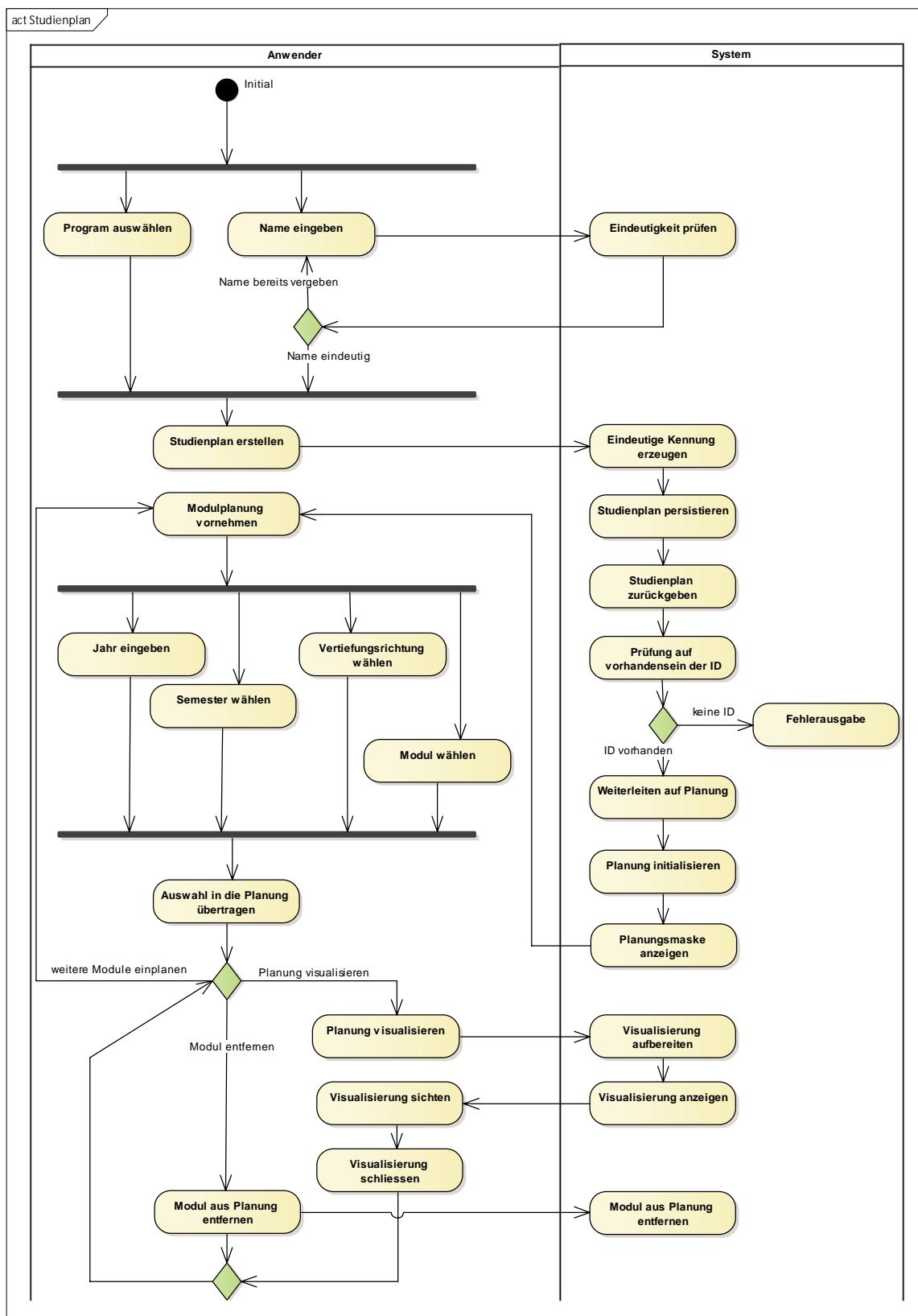


Abbildung 47: Aktivitätsdiagramm - Studienplan. Quelle: Autor

## 7.8.2 Sequenzdiagramme

Für einige Use Cases wird ein Sequenzdiagramm erstellt, welche die Interaktionen zwischen den Gruppen von Objekten sowie die Reihenfolge darstellt.

### 7.8.2.1 Studienplan erstellen

Bevor ein User einen neuen Studienplan erstellen kann, wird die Maske mit den Studienprogrammen aufbereitet:

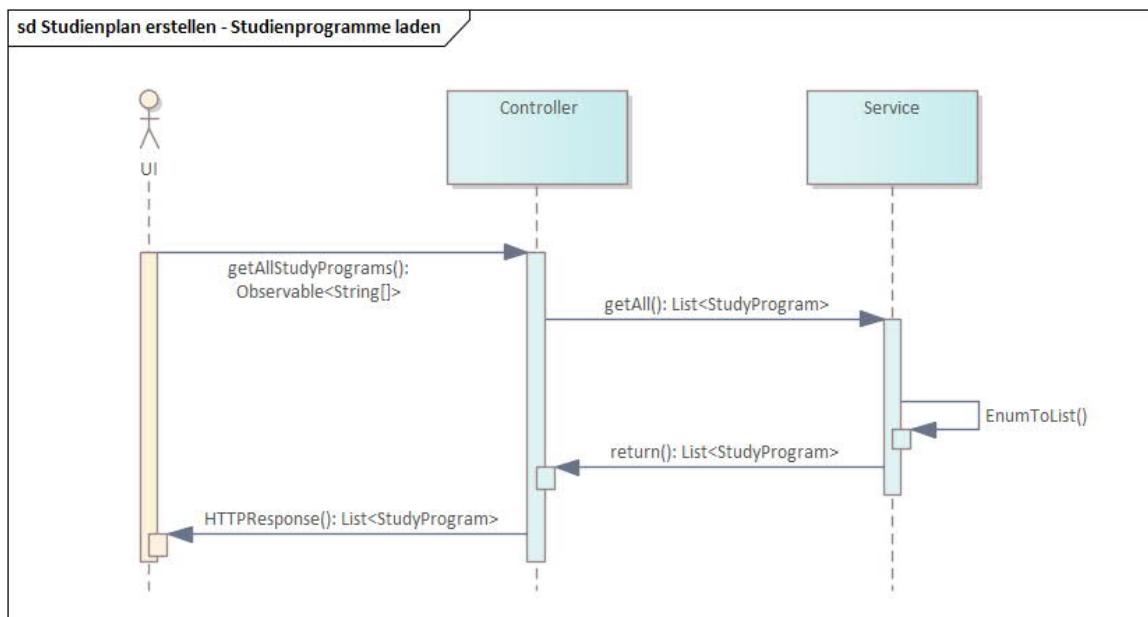


Abbildung 48: Sequenzdiagramm - Studienplan erstellen- Studienprogramme laden. Quelle: Autor

Bei der Eingabe des Studienplan-Namens wird überprüft, ob der Name einzigartig (im System noch nicht vorkommt) ist:

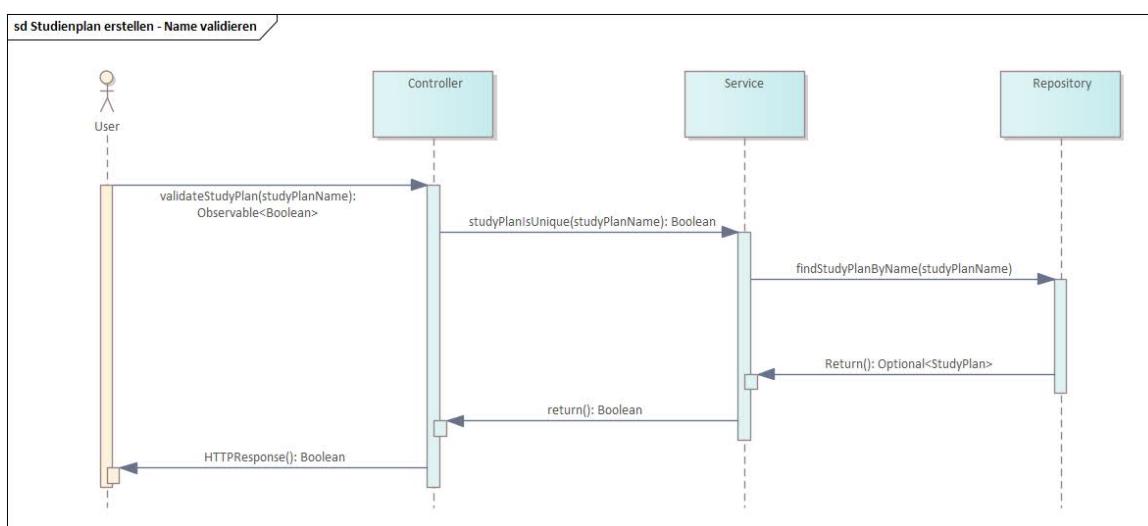


Abbildung 49: Sequenzdiagramm - Studienplan erstellen- Name validieren. Quelle: Autor

Wenn der User den Request für einen neuen Studienplan absetzt:

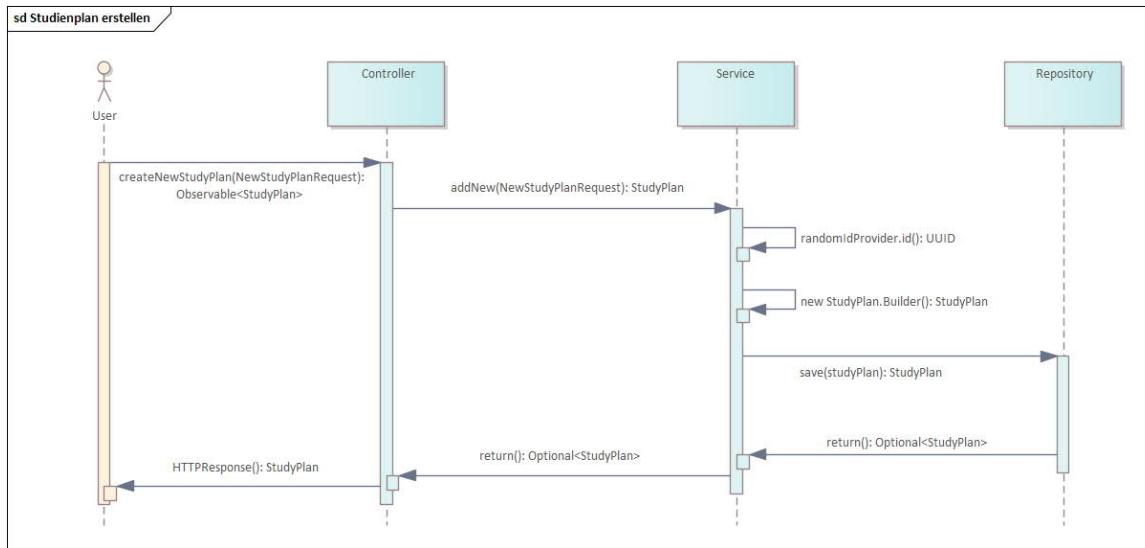


Abbildung 50: Sequenzdiagramm - Studienplan erstellen. Quelle: Autor

### 7.8.2.2 Studienplanung anzeigen

In nachfolgendem Sequenzdiagramm wird aufgezeigt, welche Abläufe stattfinden, um eine Studienplanung anzuzeigen. Dies beinhaltet mehrere Anwendungsfälle: Studienplanung anzeigen und Studienplan visualisieren. Damit eine Studienplanung angezeigt werden kann, wird eine Angular Komponente benötigt. Auf diese Komponente wird der User, nach der Erstellung oder nach dem Laden eines Studienplans mittels Namen, auf die weitergeleitet. In der Komponente wird die ID oder der Name ausgelesen und einige initial-Werte gesetzt (Jahr, Semester). Danach wird der Studienplan geladen (getStudyPlanById() oder getStudyPlanByName()) und die Informationen zu den Anzahl ECTS und Relevanz-Punkten vom Backend abgefragt. Der komplette Ablauf ist im nachfolgenden Sequenz-Diagramm ersichtlich:

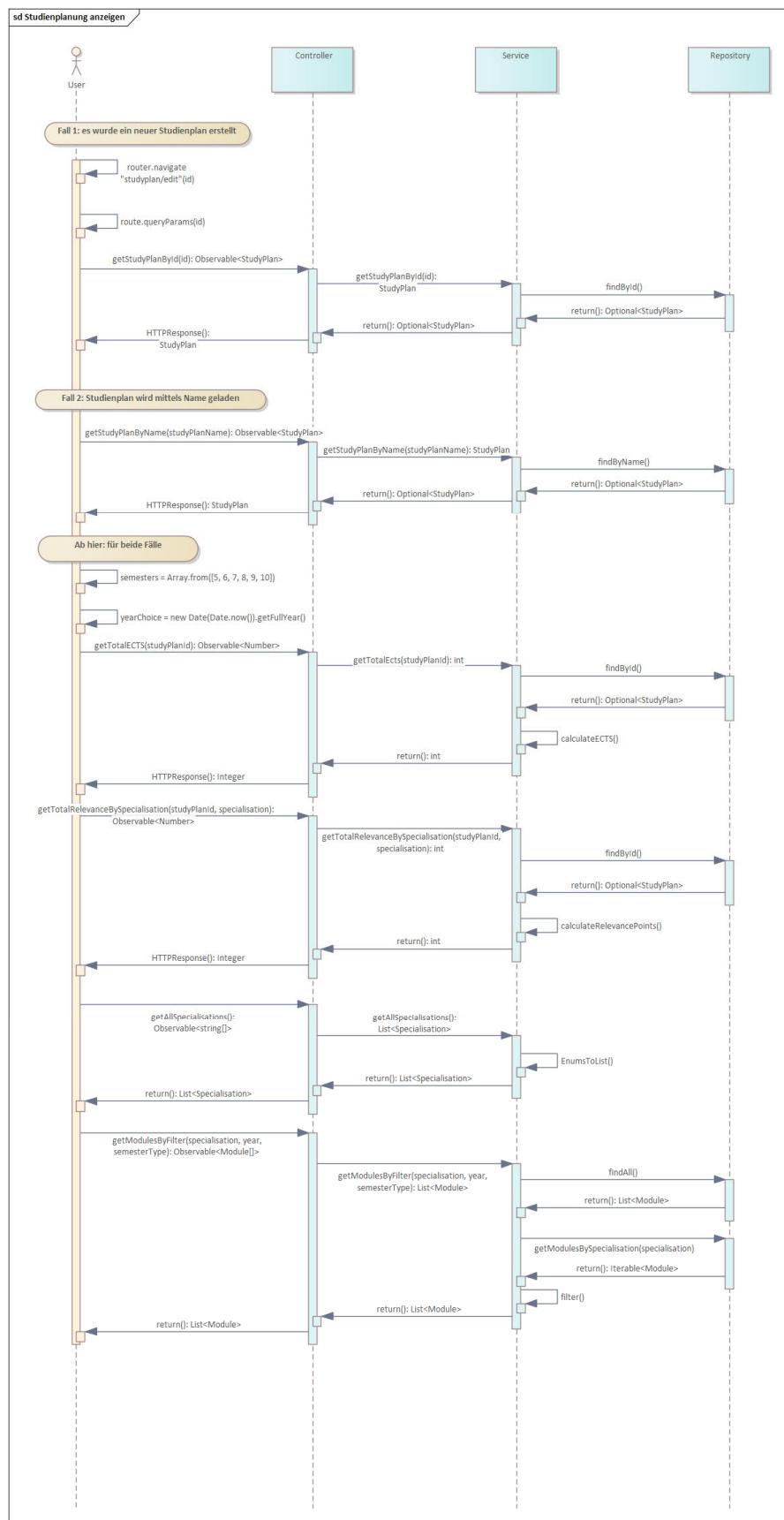


Abbildung 51: Sequenzdiagramm - Studienplanung anzeigen. Quelle: Autor

23.12.2021

### 7.8.2.3 Modul eintragen

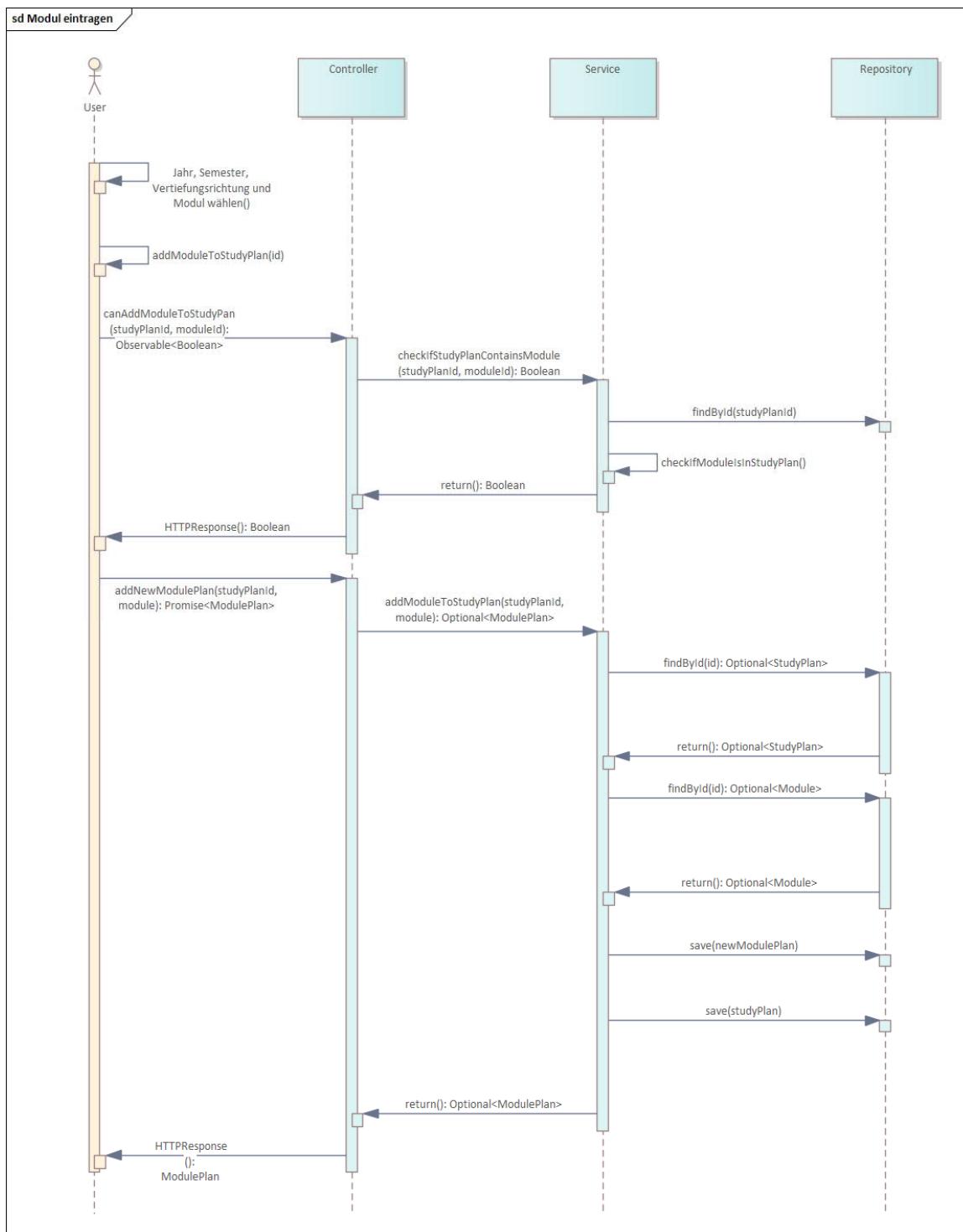


Abbildung 52: Sequenzdiagramm – Modul eintragen. Quelle: Autor

### 7.8.2.4 Studienplan laden

Für den Anwendungsfall: Studienplan mittels Namen laden

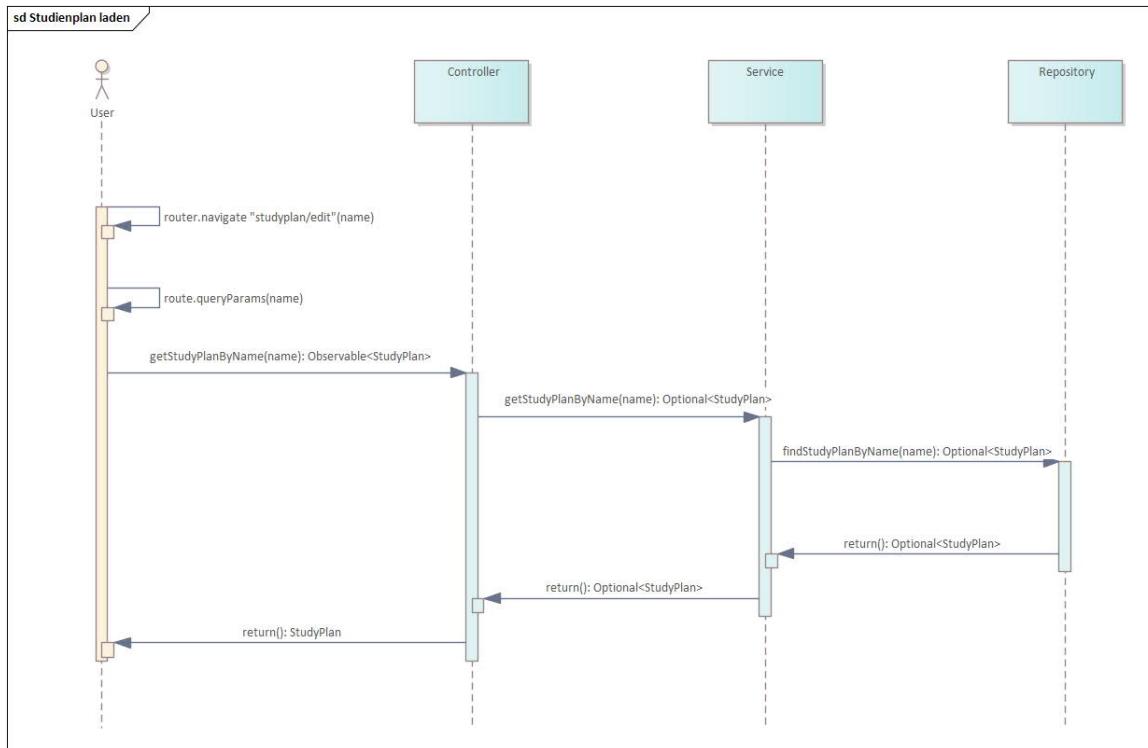


Abbildung 53: Sequenzdiagramm - Studienplan laden. Quelle: Autor

## 7.9 Softwarearchitektur Frontend

Eine komplette Dokumentation des Frontends ist in der Abgabe im Verzeichnis "Frontend Dokumentation" zu finden. In den nächsten Kapiteln werden die wichtigsten Komponenten näher spezifiziert:

### 7.9.1 Komponenten-Übersicht

Das Frontend besteht aus nachfolgenden Komponenten:

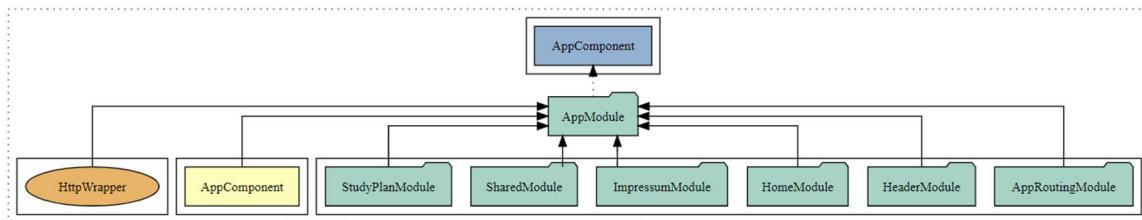


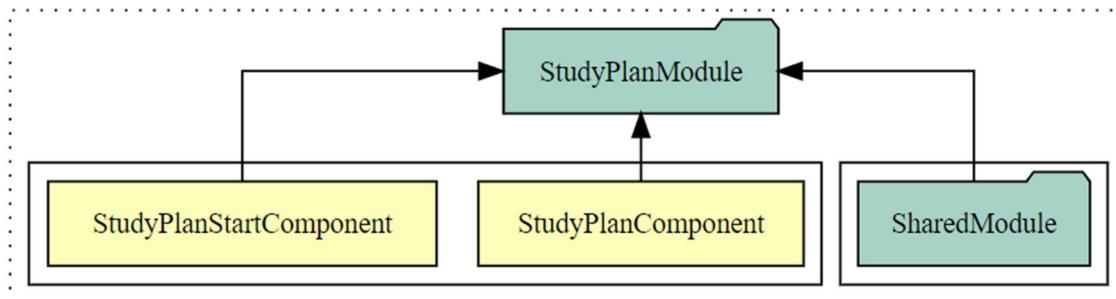
Abbildung 54: Frontend Übersicht der Komponenten. Quelle: Autor

Die einzelnen Module importieren die jeweiligen Komponenten, Dienste und Datentypen, welche über das Modul zur Verfügung gestellt werden. Die Komponenten werden der Angular-Anwendung über den Export zur Verfügung gestellt.

Tabelle 169: Hauptübersicht der Komponenten. Quelle: Autor

Komponente	Beschreibung
<b>http Wrapper</b>	Wrapper um den http-Client. Wird verwendet, um die Ziel-URL global konfigurieren zu können. Des weiteren könnten hier Token usw. im Header mitgegeben werden (wurde jedoch nicht implementiert).
<b>StudyPlanModule</b>	Modul der StudyPlanKomponenten. Hier werden die Komponenten, die über dieses Modul zur Verfügung gestellt werden, definiert.
<b>SharedModule</b>	Die SharedModule ermöglicht es, Dienste und Daten-Typen Systemweit zu nutzen. Beispielsweise werden über die SharedModule die Dienste zur API-Kommunikation zur Verfügung gestellt.
<b>ImpressumModule</b>	Modul der ImpressumKomponente. Hier werden die Komponenten, die über dieses Modul zur Verfügung gestellt werden, definiert.
<b>HomeModule</b>	Modul der HomeKomponente. Hier werden die Komponenten, die über dieses Modul zur Verfügung gestellt werden, definiert.
<b>HeaderModule</b>	Modul der HeaderModule. Hier werden die Komponenten, die über dieses Modul zur Verfügung gestellt werden, definiert.
<b>AppRoutingModule</b>	Modul der AppRoutingModule. Hier werden die Routen, die Systemweit zur Verfügung stehen, definiert.
<b> AppModule</b>	Module-Definition zum Haupteinstiegs-Punkt der Angular-Applikation.
<b> AppComponent</b>	Haupteinstiegs-Punkt der Angular-Applikation.

## 7.9.2 StudyPlanModule



## 7.9.3 SharedModule

Die SharedModule stellt Dienste und Datentypen innerhalb der gesamten Angular-Applikation zur Verfügung. Die SharedModule werden in jedem anderen Modul eingebunden (importiert) um die zur Verfügung gestellten Objekte nutzen zu können. Die Dienste, die über die SharedModule zur Verfügung gestellt werden, sind:

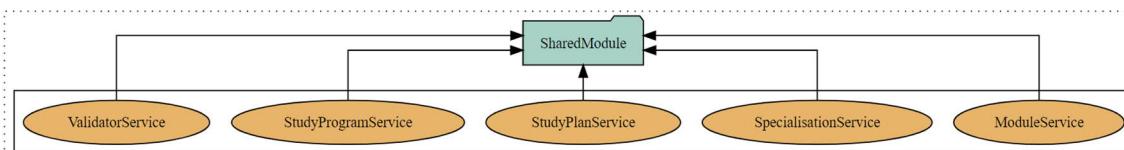
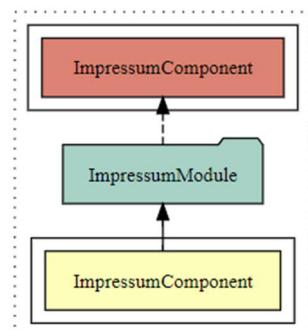


Abbildung 55: Übersicht der SharedModule. Quelle: Autor

Tabelle 170: SharedModule. Quelle: Autor

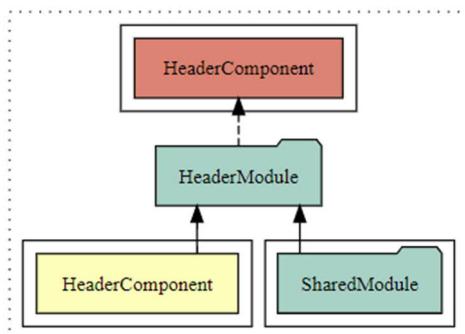
Service	Bechreibung
<b>ValidatorService</b>	Kommunikation mit dem ValidatorController
<b>StudyProgramService</b>	Kommunikation mit dem StudyProgramController
<b>StudyPlanService</b>	Kommunikation mit dem StudyPlanController
<b>SpecialisationService</b>	Kommunikation mit dem SpecialisationController
<b>ModuleService</b>	Imports/Exports und Deklarationen der Services und zur Verfügung gestellten Datentypen.

## 7.9.4 Sonstige Komponenten



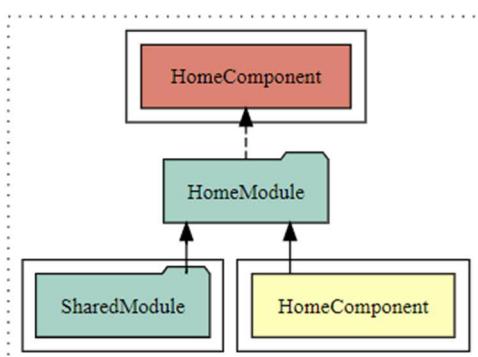
Stellt die Impressum-Komponente über das ImpressumModule zur Verfügung.

Abbildung 56: Impressum Component. Quelle: Autor



Stellt die Header-Komponente (Navigation) über das HeaderModule zur Verfügung.

Abbildung 57: Header Component. Quelle: Autor



Stellt die Home-Komponente (Startseite) über das HomeModule zur Verfügung.

Abbildung 58: Home Component. Quelle: Autor

## 7.9.5 Routing auf Komponenten

URL-Aufrufe im Browser werden mittels des AppRoutingModule-Moduls auf die jeweilig zuständige Angular-Komponenten gematcht. Nachfolgende Abbildung zeigt das Routing der Endpunkte (URLs) auf die entsprechende Komponente:

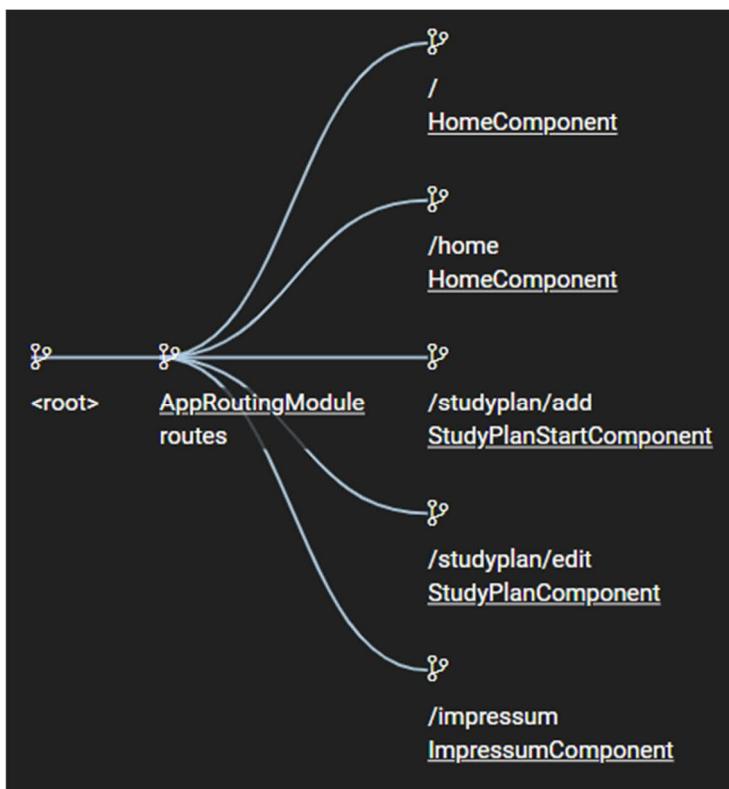


Abbildung 59: Frontend Routing auf Komponenten. Quelle: Autor

## 8 Testkonzept

Das Testkonzept beschreibt die Testziele, Testobjekte, Testarten, Testinfrastruktur sowie die Testorganisation. Es umfasst ebenfalls die Testplanung und die Testfallbeschreibungen. Für jeden Testfall wird eine detaillierte Testfallbeschreibung erstellt. Diese stellt die Spezifikation des Tests dar. Die Testplanung legt den logischen und zeitlichen Ablauf der Tests fest. Das Testkonzept bildet die Grundlage, auf der die Testorganisation und die Testinfrastruktur bereitgestellt und die Tests durchgeführt werden. Es wird bei neuen Erkenntnissen stets nachgeführt. Als Testkonzept wird hier das Testkonzept von HERMESS, welches erprobt und standardisiert ist, verwendet. (Eidgenossenschaft, 2021).

Das Backend wird mit Unit-Tests getestet. Das Frontend wird manuell getestet.

Wir wenden das «Unit Testing and the Arrange, Act and Assert (AAA) Pattern» (Gomes, 2017) an, welches wie folgt in drei Teile aufgebaut ist:

- Arrange: Hier werden die Vorbereitungen für den Testfall getroffen (Objekte initialisieren, Mockups definieren)
- Act: Hier wird die Methode/Endpoint aufgerufen
- Assert: Hier wird das Resultat verglichen und geprüft

Ein Codebeispiel kann wie folgt aussehen:

```
// arrange
var studyPlanName = "test";

given(service.studyPlanNameIsUnique(studyPlanName)).willReturn(true);

// act
var result = mvc.perform(
    get("/api/v1/validator/validStudyPlan")
        .param("studyPlanName", studyPlanName)
        .contentType(MediaType.APPLICATION_JSON));

// assert
result
    .andExpect(status().isOk())
    .andExpect(content().string("true"));
```

## 8.1 Testziele

Globale messbare Testziele über alle Testfälle hinweg:

Nr.	Beschreibung	Messgrösse	Priorität*
1	Schnittstellentest erfolgreich (Schichtenarchitektur)	Datenaustausch korrekt	M/1
2	Daten können in die Datenbank persistiert werden (Datenhaltung)	Datenpersistierung korrekt	M/1
3	Programm Studienplanung funktional	Ein Studienplan kann gespeichert werden	M/1
4	Programm Studienplanung funktional	Ein Studienplan kann angezeigt werden	M/1
5	Programm Studienplanung funktional	Ein Studienplan kann bearbeitet werden	M/1
6	Programm Studienplanung funktional	Ein Studienplan kann gelöscht werden	M/1
7	Testabdeckung Klassen, Methoden und Codezeilen	>= 85%	M/1

\* Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief

Tabelle 171: Übergeordnete Testziele. Quelle: Autor

## 8.2 Teststrategie und Teststufen

Es werden alle Schichten des Backend (Controller, Service und Repository) getestet. In den Schichten Controller und Service werden die aufzurufenden Abhängigkeiten (Services und Repository) mittels "gemockt" gefaketed.

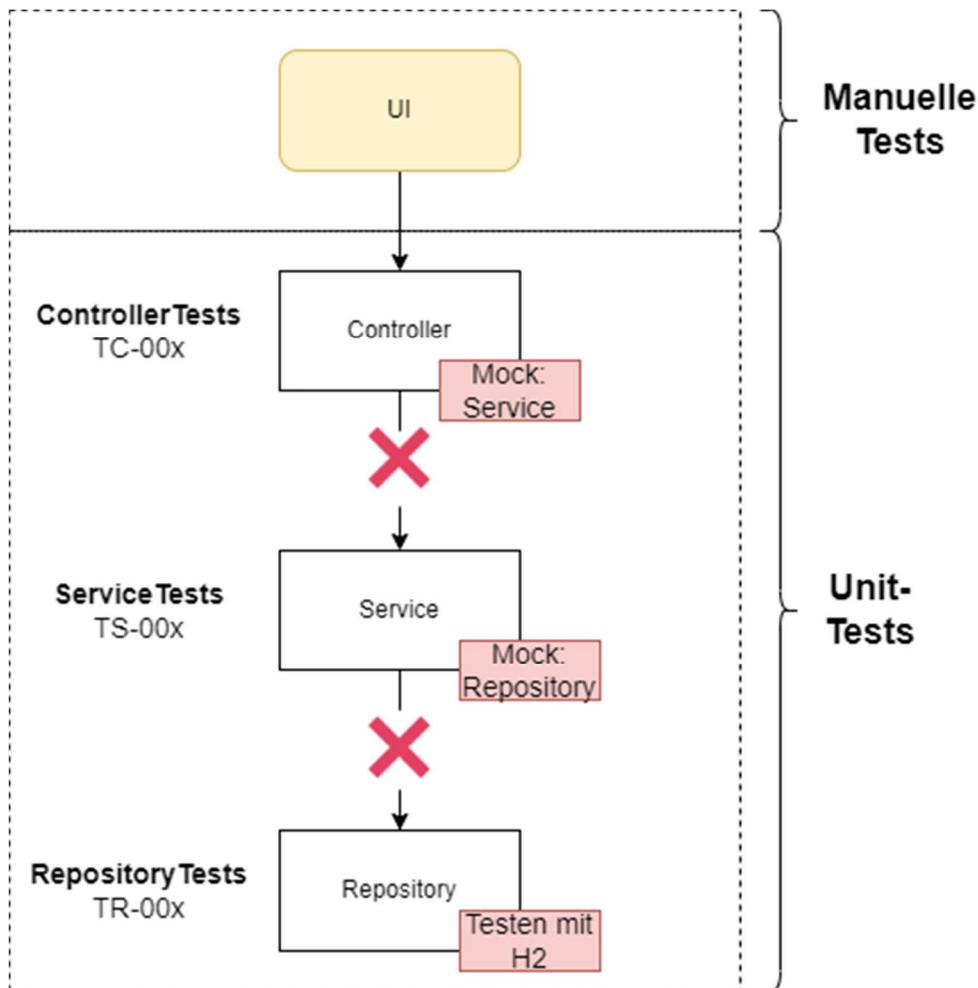


Abbildung 60: Teststrategie. Quelle: Autor

## 8.3 Testabdeckung

Für die Testabdeckung wurde ein Report in der Entwicklungsumgebung (IntelliJ IDEA), welcher als HTML-Datei exportiert werden kann, generiert. Es wird eine Testabdeckung der Klassen, Methoden und Code-Linien von über 85% angestrebt. Die Reports sind im Unterkapitel des Testkonzeptes unter Reports zu entnehmen.

## 8.4 Testrahmen

### 8.4.1 Testvoraussetzungen

Dies sind die Testvoraussetzungen, welche für die Testfälle und -ausführung relevant sind. Folgende Testvoraussetzungen werden jeweils in den Testfällen unter dem Punkt Testvoraussetzung angegeben und werden aus Redundanzgründen nur hier aufgeführt:

- Testziele definiert
- IDE für JUnit-Tests eingerichtet
- Testplan ist erstellt
- Testprotokoll vorbereitet
- Testteam definiert
- Start und Abbruchbedingungen definiert

Die Projektmitglieder verfügen bereits über einiges Vorwissen zum Thema Softwaretests. Die Testfälle werden unter Anleitung und im gemeinsamen Pair-Programming mit Peter Spielhofer durchgeführt um Wissenslücken zu schliessen und eine gemeinsame Verständnisbasis für die Testfälle zu gewinnen. Folgende Kenntnisse sind vorhanden:

Tester	Vorkenntnisse	Kürzel
<b>Christoph Grossmann</b>	Schulisch und Whitebox-Tests	CHG
<b>Marco Ris</b>	FFHS Modul SWQ (2021)	MR
<b>Peter Spielhofer</b>	TDD in C# (seit 2017)	PS

Tabelle 172: Tester und Vorkenntnisse. Quelle: Autor

#### 8.4.2 Mängelklassifizierung

Die festgestellten Mängel, bzw. die nicht erfüllten Anforderungen (Erwartungen), werden in Klassen von 1 bis 4 eingestuft. Die Klasse 0 findet nur dann Verwendung, wenn ein einwandfreies Ergebnis gesondert ausgewiesen werden kann:

Nr.	Mängelklassen	Beschreibung
0	mängelfrei	Einwandfrei und anforderungsgerecht
1	belangloser Mangel	Verwendung möglich, Brauchbarkeit ist vorhanden, Mangel sollte dennoch nicht vorkommen
2	leichter Mangel	Verwendung möglich, Brauchbarkeit ist nur wenig beeinträchtigt
3	schwerer Mangel	Verwendung ist noch möglich, Brauchbarkeit ist stark verringert
4	kritischer Mangel	Unbrauchbar; Wesentliche Funktionalität ist nicht gegeben; Betrieb ist nicht verantwortbar (z.B. sicherheitsspezifisch)

Tabelle 173: Mängelklassen. Quelle: Autor

Die Klassifizierung spiegelt die Folgenschwere und den Aufwand zur Behebung der möglich feststellbaren Mängel. Die Zuordnung der festgestellten Mängel zu einer Mängelklasse gibt grob auch die Priorität vor, in welcher Reihenfolge die Behebung der Mängel angegangen werden soll.

Wird eine Mängelklasse zwischen 1-3 erreicht, kann das System/Produkt unter Vorbehalt abgenommen werden. Zur Behebung der Mängel sind jedoch Massnahmen zu definieren. Eine Nachprüfung ist zwingend.

Werden hingegen Mängel der Klasse 4 festgestellt, kann das System/Produkt nicht abgenommen werden und der Auftragnehmer muss umgehend Massnahmen treffen, um diese Mängel zu beheben. Der Auftragnehmer hat zudem die erneute Abnahme zu veranlassen und hat den Auftraggeber zu informieren.



### 8.4.3 Start- und Abbruchbedingungen

Die Startbedingung wird in den Testvoraussetzungen definiert. Die Abbruchbedingung kann das Budget des Testens sein oder aber die Ressource Zeit. Da wir hier kein Budget für das Projekt haben, macht es keinen Sinn die Abbruchbedingung auf das Budget zu setzen. Wir nehmen als Abbruchbedingung die Ressource Zeit, bis wann entwickelt, getestet und debuggt wird. Dies wird genauer im Arbeitspaket R-7 definiert.

## 8.5 Testumgebung

Als Testumgebung nehmen wir das Framework JUnit in der Version 5, welches sehr bekannt ist und sich in vielen Programmen etabliert hat. (Team, 2021)

## 8.6 Testinfrastruktur

### 8.6.1 Testsystem

Das Testsystem, hier JUnit5 wird durch das Packetmanagement maven dem Projekt hinzugefügt. Die Tests werden nach dem Testplan durchgeführt, debuggt und im Testprotokoll festgehalten.

### 8.6.2 Testdaten

Die Testdaten bestehen jeweils aus einem Positiven und einem Negativen case, welche jeweils ein Resultat liefern und geprüft/verglichen werden.

### 8.6.3 Testhilfsmittel

Als Testhilfsmittel verwenden wir JUnit welches von der IDE gestartet werden kann. Die Resultate (reports) werden in der Ausgabe des IDE angezeigt und können im Anhang neben dem Testprotokoll entnommen werden.

## 8.7 Testorganisation

Als Testorganisation werden hier die Projektinternen Mitglieder, die in den Testvoraussetzungen aufgelistet sind, genannt.

Marco Ris hat das Wahlfach der FFHS Software Qualität (SWQ) besucht und ist für das Testkonzept, die Durchführung und Protokollierung verantwortlich. Testdurchführer sind Christoph Grossmann und Peter Spielhofer. Die Testdurchführung ist in der Testplanung zu entnehmen.

## 8.8 Testfallbeschreibungen

Die Bezeichnungen der folgenden Testfälle beziehen sich auf die Testebene:

- TC sind die Testfälle für die Ebene Controller
- TS sind die Testfälle für die Ebene Service und
- TR sind die Testfälle für die Ebene Repository.

Die Testfälle werden gruppiert nach User-Story aufgelistet.

## 8.8.1 Studienplan erstellen

### Controller-Tests

Tabelle 174: Testfall TC-001. Quelle: Autor

ID / Bezeichnung	TC-001
Referenz zu User-Story	U-1
Testebene	Controller - ValidatorController
Methoden Name	validateStudyPlanByName
Test-Klasse	ValidatorControllerTest.java
Beschreibung	Name des Studienplans validieren.
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Name existiert noch nicht (TRUE)
Negative Case 1	Name existiert bereits (FALSE)
Negative Case 1	Name ist leer (FALSE)
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: TRUE</li> <li>• Negative Case 1: FALSE</li> <li>• Negative Case 2: FALSE</li> </ul>

Tabelle 175: Testfall TC-002. Quelle: Autor

ID / Bezeichnung	TC-002
Referenz zu User-Story	U-1
Testebene	Controller - StudyPlanController
Methoden Name	addNewStudyPlan
Test-Klasse	StudyPlanControllerTest.java
Beschreibung	Neuen Studienplan erstellen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	NewStudyPlanRequest ist valide und ein Optional von StudyPlan wird zurückgegeben
Negative Case	NewStudyPlanRequest ist invalid und ein HTTP 400 wird zurückgegeben.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: HTTP 400</li> </ul>

Tabelle 176: Testfall TC-003. Quelle: Autor

ID / Bezeichnung	TC-003
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Controller – StudyProgramController
<b>Methoden Name</b>	getStudyPrograms
<b>Test-Kasse</b>	StudyProgramController.java
<b>Beschreibung</b>	Studienprogramme erhalten
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Alle Studienprogramme von Endpunkt erhalten. Die Studienprogramme werden in der Service-Klasse auf Basis eines ENUMs aufbereitet. Der Endpunkt liefert in jedem Fall ein positives Resultat, resp. eine Liste mit Studienprogrammen.
<b>Negative Case</b>	-
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;StudyProgram&gt;</li> <li>• Negative Case: -</li> </ul>

### Service-Tests

Tabelle 177: Testfall TS-001. Quelle: Autor

ID / Bezeichnung	TS-001
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Service - StudyPlanService
<b>Methoden Name</b>	studyPlanNameIsUnique
<b>Test-Klasse</b>	StudyPlanServiceTest.java
<b>Beschreibung</b>	Prüfen ob schon ein Studienplan mit demselben Namen existiert oder nicht.
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Studienplan mit demselben Namen existiert noch nicht. Somit wird TRUE zurückgegeben.
<b>Negative Case 1</b>	Studienplan mit demselben Namen existiert bereits. Somit wird FALSE zurückgegeben.
<b>Negative Case 2</b>	Ein weiterer Fehler kann auftreten, wenn die Datenbank nicht erreichbar ist. In der Realität würde JPA einen DataException werfen, was nicht getestet werden kann. Der Test wird simuliert, in dem der Datenbankaufruf eine Exception wirft. Es wird somit ein NullPointerException geworfen, was zu einem FALSE führt.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: True</li> <li>• Negative Case 1: False</li> <li>• Negative Case 2: False</li> </ul>

Tabelle 178: Testfall TS-002. Quelle: Autor

ID / Bezeichnung	TS-002
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Service - StudyPlanService
<b>Methoden Name</b>	validateStudyPlan
<b>Test-Klasse</b>	StudyPlanServiceTest.java
<b>Beschreibung</b>	StudyPlan validieren (sind alle relevanten Informationen auf dem Studienplan vorhanden)
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	StudyPlan ist valide. Valide bedeutet: <code>name != null</code> <code>name &lt;= 255</code> <code>date != null</code> <code>studyProgram != null</code>
<b>Negative Case</b>	Hier reicht ein Testfall, da alle oben aufgeführten Bedingungen erfüllt sein müssen, damit die Validierung TRUE zurückgibt. Es wird mittels <code>name &gt; 255</code> getestet
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>Positive Case: True</li> <li>Negative Case: False</li> </ul>

Tabelle 179: Testfall TS-003. Quelle: Autor

ID / Bezeichnung	TS-003
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Service - StudyPlanService
<b>Methoden Name</b>	addNewStudyPlan
<b>Test-Klasse</b>	StudyPlanServiceTest.java
<b>Beschreibung</b>	Neuen Studienplan erstellen
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	<ul style="list-style-type: none"> <li>Erstellung eines neuen Studienplans (NewStudyPlanRequest)</li> <li>Service zur Erstellung dieses Studienplans wird aufgerufen</li> <li>Service gibt StudyPlan zurück</li> <li>UUID ist vorhanden</li> </ul>
<b>Negative Case 1</b>	Ein Fehler, welcher auftreten könnte: StudyPlan wird nicht gespeichert (in der Realität würde JPA einen DataException werfen, was nicht getestet werden kann). Es wird somit ein NullPointerException geworfen, was zu einem leeren Optional führt.
<b>Negative Case 2</b>	Wenn der Request für einen neuen Studienpan unvollständig ist (Beispielsweise der Name ist leer) darf dieser nicht verarbeitet werden. Rückgabe, leeres Optional.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>Positive Case: Optional&lt;StudyPlan&gt;</li> <li>Negative Case 1: Optional.empty()</li> <li>Negative Case 2: Optional.empty()</li> </ul>

Tabelle 180: Testfall TS-004. Quelle: Autor

ID / Bezeichnung	TS-004
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Service - StudyProgramService
<b>Methoden Name</b>	getAll
<b>Test-Klasse</b>	StudyProgramService.java
<b>Beschreibung</b>	Alle Studienprogramme (von Enum) als Liste zurückgeben.
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Studienprogramme werden von der Enum-Klasse in eine Liste umgewandelt und zurückgegeben.
<b>Negative Case</b>	Es existiert kein negativ Case, da die Werte fix im Programm einprogrammiert sind (als ENUM).
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;StudyProgram&gt;</li> <li>• Negative Case: -</li> </ul>

## Repository-Tests

Tabelle 181: Testfall TR-001. Quelle: Autor

ID / Bezeichnung	TR-001
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Repository - IStudyPlanRepository
<b>Methoden Name</b>	findStudyPlanByName
<b>Test-Klasse</b>	StudyPlanRepositoryTest.java
<b>Beschreibung</b>	Prüfen ob ein Studienplan mit demselben Namen schon existiert
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Wenn ein bereits existierender Studienplan mittels Namens gesucht wird, wird der Studienplan zurückgegeben.
<b>Negative Case</b>	Im Fehlerfall wirft JPA eine Exception (DataException), jedoch werden die Exceptions auf Service-Ebene behandelt. Deshalb wird kein negative Case getestet
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: Optional&lt;Studyplan&gt;</li> <li>• Negative Case: Exception</li> </ul>

Tabelle 182: Testfall TR-002. Quelle: Autor

ID / Bezeichnung	TR-002
<b>Referenz zu User-Story</b>	U-1
<b>Testebene</b>	Repository - IStudyPlanRepository
<b>Methoden Name</b>	save
<b>Test-Klasse</b>	StudyPlanRepositoryTest.java
<b>Beschreibung</b>	Neuen Studienplan erstellen
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Wenn ein neuer Studienplan gespeichert wird, wird das gespeicherte Objekt zurückgegeben.
<b>Negative Case</b>	Im Fehlerfall wirft JPA eine Exception (DataException), jedoch werden die Exceptions auf Service-Ebene behandelt. Deshalb wird kein negative Case getestet
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: Exception</li> </ul>

## 8.8.2 Studienplanung anzeigen

### Controller-Tests

Tabelle 183: Testfall TC-004. Quelle: Autor

ID / Bezeichnung	TC-004
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Controller - StudyPlanController
<b>Methoden Name</b>	getStudyPlan
<b>Test-Klasse</b>	StudyPlanControllerTest.java
<b>Beschreibung</b>	Hole Studienplan mittels UUID.
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Ein Objekt vom Typ StudyPlan wird zurückgegeben, wenn der Studienplan gefunden werden kann.
<b>Negative Case</b>	Ein HTTP 400 wird zurückgegeben, falls der Service den Studienplan nicht gefunden hat.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: HTTP 400</li> </ul>

Tabelle 184: Testfall TC-005. Quelle: Autor

ID / Bezeichnung	TC-005
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Controller - SpecialisationController
<b>Methoden Name</b>	getSpecialisations
<b>Test-Klasse</b>	SpecialisationControllerTest.java
<b>Beschreibung</b>	Alle Vertiefungsrichtungen zurückgeben
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case Beschreibung</b>	Wenn der Service die Spezialisierungen serialisieren (Liste) kann, wird eine Liste von Spezialisierungen zurückgegeben.
<b>Negative Case</b>	Wenn der Service die Spezialisierungen nicht serialisieren (Liste) kann und die Liste leer ist, wird ein HTTP 204 zurückgegeben.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;Specialisation&gt;</li> <li>• Negative Case: HTTP 204</li> </ul>

Tabelle 185: Testfall TC-006. Quelle: Autor

ID / Bezeichnung	TC-006
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Controller - ModuleController
<b>Methoden Name</b>	getModulesByFilter
<b>Test-Klasse</b>	ModuleControllerTest.java
<b>Beschreibung</b>	Alle Module auf Basis der angegebenen Filter (inklusive Basis-Module) zurückgeben
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case Beschreibung</b>	Ein Liste mit Module auf Basis der angegebenen Filter (inklusive Basis-Module) wird zurückgegeben.
<b>Negative Case</b>	Es gibt keinen negativen Case, da in jedem Fall Module zurückgegeben werden. Sind alle Filter leer, so werden alle Module zurückgegeben. Werden auf Grund des Filters keine Module gefunden, werden zumindest die Basis-Module zurückgegeben.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;Module&gt;</li> <li>• Negative Case: -</li> </ul>

## Service-Tests

Tabelle 186: Testfall TS-005. Quelle: Autor

ID / Bezeichnung	TS-005
Referenz zu User-Story	U-2
Testebene	Service - StudyPlanService
Methoden Name	getStudyPlanById
Test-Klasse	StudyPlanServiceTest.java
Beschreibung	Gebe den gefundenen Studienplan anhand der ID zurück.
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall wird ein Optional vom Typ StudyPlan zurückgegeben, falls ein Studienplan mit derselben ID existiert.
Negative Case	Im negativen Fall wird ein leeres Optional zurückgegeben, falls der Studienplan nicht existiert.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Optional&lt;Studyplan&gt;</li> <li>• Negative Case: Optional.empty()</li> </ul>

Tabelle 187: Testfall TS-006. Quelle: Autor

ID / Bezeichnung	TS-006
Referenz zu User-Story	U-2
Testebene	Service - SpecialisationService
Methoden Name	getAllSpecialisations
Test-Klasse	StudyPlanServiceTest.java
Beschreibung	Alle Vertiefungsrichtungen zurückgeben
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case Beschreibung	Ein ArrayList<Specialisation> wird zurückgegeben.
Negative Case	Es gibt keinen negative Case. Das Serialisieren von Enums als Liste ist eine Java Implementation, die nur bei einem fehlerhaften Enum fehlschlagen kann. Dies kann nicht mit einem Test geprüft werden. Dieser Fall kann auf dem Controller abgefangen werden.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: ArrayList&lt;Specialisation&gt;</li> <li>• Negative Case: -</li> </ul>

Tabelle 188: Testfall TS-007. Quelle: Autor

ID / Bezeichnung	TS-007
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Service - ModuleService
<b>Methoden Name</b>	getModulesByFilter
<b>Test-Klasse</b>	ModuleServiceTest.java
<b>Beschreibung</b>	Alle Module der angegebenen Filter (inklusive Basis-Module) zurückgeben
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case 1</b>	Ein Liste mit Module auf Basis der angegebenen Filter (inklusive Basis-Module) wird zurückgegeben.
<b>Positive Case 2</b>	Ein Liste mit Module auf Basis der angegebenen Filter (nur Basis-Module) wird zurückgegeben.
<b>Negative Case</b>	Es werden in jedem Fall Module zurückgegeben werden. Sind alle Filter leer, so werden alle Module zurückgegeben. Werden auf Grund des Filters keine Module gefunden, werden zumindest die Basis-Module zurückgegeben. Der negative Case ist im Positive Case 2 abgedeckt.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case 1: List&lt;Module&gt;</li> <li>• Positive Case 2: List&lt;Module&gt;</li> <li>• Negative Case: -</li> </ul>

Tabelle 189: Testfall TS-008. Quelle: Autor

ID / Bezeichnung	TS-008
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Service - ModuleService
<b>Methoden Name</b>	findByModuleType
<b>Test-Klasse</b>	ModuleServiceTest.java
<b>Beschreibung</b>	Alle Module des Modultypen zurückgeben
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Ein Liste mit Module zum angegebenen Typ wird zurückgegeben.
<b>Negative Case</b>	Eine leere Liste wird zurückgegeben, wenn die Liste der Module mit dem angegebenen Type leer ist.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;Module&gt;</li> <li>• Negative Case: Leere List&lt;Module&gt;</li> </ul>

## Repository-Tests

Tabelle 190: Testfall TR-003. Quelle: Autor

ID / Bezeichnung	TR-003
Referenz zu User-Story	U-2
Testebene	Repository - IStudyPlanRepository
Methoden Name	findById
Test-Klasse	StudyPlanRepositoryTest.java
Beschreibung	Studienplan mittels UUID abrufen.
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall wird ein Optional vom Typ StudyPlan zurückgegeben, falls ein Studienplan mit derselben ID existiert.
Negative Case	Im Fehlerfall wirft JPA eine Exception (DataException), jedoch werden die Exceptions auf Service-Ebene behandelt. Deshalb wird kein negative Case getestet
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Optional&lt;Studyplan&gt;</li> <li>• Negative Case: Exception</li> </ul>

Tabelle 191: Testfall TR-004. Quelle: Autor

ID / Bezeichnung	TR-004
Referenz zu User-Story	U-2
Testebene	Repository - ModuleRepository
Methoden Name	findAll
Test-Klasse	ModuleRepositoryTest.java
Beschreibung	Alle Module abrufen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Wenn Module existieren, wird eine Liste mit Modulen zurückgegeben.
Negative Case	Wenn keine Module existieren, wird eine leere Liste mit Module zurückgegeben.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;Module&gt;</li> <li>• Negative Case: Leere List&lt;Module&gt;</li> </ul>

Tabelle 192: Testfall TR-005. Quelle: Autor

ID / Bezeichnung	TR-005
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Repository - ModuleRepository
<b>Methoden Name</b>	findByRelevanceSpecialisation
<b>Test-Klasse</b>	ModuleRepositoryTest.java
<b>Beschreibung</b>	Nach Relation von Relevanz und Vertiefung suchen
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Wenn das Modul mit der gesuchten Spezialisierung existiert, wird ein Iterable<Module> zurückgegeben.
<b>Negative Case</b>	Wenn das Modul mit der gesuchten Spezialisierung existiert, wird ein leeres Iterable<Module> zurückgegeben.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: Iterable&lt;Module&gt;</li> <li>• Negative Case: Leeres Iterable&lt;Module&gt;</li> </ul>

Tabelle 193: Testfall TR-006. Quelle: Autor

ID / Bezeichnung	TR-006
<b>Referenz zu User-Story</b>	U-2
<b>Testebene</b>	Repository - ModuleRepository
<b>Methoden Name</b>	findByModuleType
<b>Test-Klasse</b>	ModuleRepositoryTest.java
<b>Beschreibung</b>	Alle Module des Modultypen finden
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	Wenn das Modul mit dem gesuchten Modultyps existiert, wird ein List<Module> zurückgegeben.
<b>Negative Case</b>	Wenn das Modul mit dem gesuchten Modultyps nicht existiert, wird ein leere List<Module> zurückgegeben.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: List&lt;Module&gt;</li> <li>• Negative Case: Leere Liste vom Typ List&lt;Module&gt;</li> </ul>

### 8.8.3 Modul eintragen

#### Controller-Tests

Tabelle 194: Testfall TC-007. Quelle: Autor

<b>ID / Bezeichnung</b>	<b>TC-007</b>
<b>Referenz zu User-Story</b>	U-7
<b>Testebene</b>	Controller - StudyPlanController
<b>Methoden Name</b>	addModuleToStudyPlan
<b>Beschreibung</b>	Module dem Studienplan hinzufügen
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	ModulePlan wird zurückgegeben.
<b>Negative Case</b>	Ein HTTP 400 wird zurückgegeben, falls der ausgerufene Service das Modul nicht hinzufügen konnte.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: HTTP 400</li> </ul>

Tabelle 195: Testfall TC-008. Quelle: Autor

<b>ID / Bezeichnung</b>	<b>TC-008</b>
<b>Referenz zu User-Story</b>	U-7
<b>Testebene</b>	Controller - ValidatorController
<b>Methoden Name</b>	canAddModuleToStudyPlan
<b>Beschreibung</b>	Validieren ob Module dem Studienplan hinzugefügt werden kann
<b>Testvoraussetzung</b>	In Kapitel 8.6.1 definiert
<b>Positive Case</b>	True wird zurückgegeben, falls die Abfrage gültig ist (Studyplan-ID und Module-ID ist nicht 0) und das Modul eingetragen werden kann (Modul ist nicht im StudyPlan).
<b>Negative Case</b>	False wird zurückgegeben, falls die Studyplan-ID oder Module-ID 0 ist oder nicht eingetragen werden kann.
<b>Erwartetes Ergebnis</b>	<ul style="list-style-type: none"> <li>• Positive Case: True</li> <li>• Negative Case: False</li> </ul>

## Service-Tests

Tabelle 196: Testfall TS-009. Quelle: Autor

ID / Bezeichnung	TS-009
Referenz zu User-Story	U-7
Testebene	Service - StudyPlanService
Methoden Name	addModuleToStudyPlan
Beschreibung	Module dem Studienplan hinzufügen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Ein Optional<ModulePlan> wird zurückgegeben
Negative Case	Im negativen Fall wird ein leeres Optional zurückgegeben, falls der Studienplan oder das Modul nicht existiert.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Optional&lt; ModulePlan &gt;</li> <li>• Negative Case: Optional&lt;empty&gt;</li> </ul>

## Service Tests

Tabelle 197: Testfall TS-010. Quelle: Autor

ID / Bezeichnung	TS-010
Referenz zu User-Story	U-7
Testebene	Service - StudyPlanService
Methoden Name	checkIfStudyPlanContainsModule
Beschreibung	Validieren ob Module dem Studienplan hinzugefügt werden kann
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Prüfen, ob das Modul bereits im Studienplan vorhanden ist. Wenn ja gebe True zurück.
Negative Case	Falls das Modul nicht im Studienplan vorhanden ist, wird False zurückgegeben.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: True</li> <li>• Negative Case: False</li> </ul>

## Repository-Tests

Tabelle 198: Testfall TR-007. Quelle: Autor

ID / Bezeichnung	TR-007
Referenz zu User-Story	U-7
Testebene	Repository - StudyPlanRepository
Methoden Name	save
Beschreibung	Aktualisiert den Studienplan
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Case wird der Studienplan zurückgegeben. Es wird geprüft ob der zurückgegebene Studienplan die gespeicherten Änderungen beinhaltet.
Negative Case	Im negativen wird auch der StudyPlan zurückgegeben. Es wird geprüft ob der zurückgegebene Studienplan die gespeicherten Änderungen beinhaltet. Dieser Fall kann nicht getestet werden.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: StudyPlan</li> </ul>

## 8.8.4 Modul entfernen

### Controller-Tests

Tabelle 199: Testfall TC-009. Quelle: Autor

ID / Bezeichnung	TC-009
Referenz zu User-Story	U-8
Testebene	Controller - StudyPlanController
Methoden Name	removeModuleFromStudyPlan
Beschreibung	Modul aus Studienplan löschen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Ein HTTP 200 und die ID des Studienplan wird zurückgegeben.
Negative Case	Ein HTTP 400 wird zurückgegeben.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: HTTP 200</li> <li>• Negative Case: HTTP 400</li> </ul>

## Service-Tests

Tabelle 200: Testfall TS-011. Quelle: Autor

ID / Bezeichnung	TS-011
Referenz zu User-Story	U-8
Testebene	Service - StudyPlanService
Methoden Name	removeModuleFromStudyPlan
Beschreibung	Modul aus Studienplan löschen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Studienplan existiert. Modul entfernen OK. Gibt TRUE zurück.
Negative Case 1	Studienplan existiert nicht. Modul kann nicht entfernt werden. Gibt FALSE zurück.
Negative Case 2	Modul ist auf dem Studienplan nicht vorhanden. Modul kann nicht entfernt werden. Gibt FALSE zurück.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: True</li> <li>• Negative Case 1: False</li> <li>• Negative Case 2: False</li> </ul>

## 8.8.5 Studienplan mittels Namen laden

### Controller-Tests

Tabelle 201: Testfall TC-010. Quelle: Autor

ID / Bezeichnung	TC-010
Referenz zu User-Story	U-14
Testebene	Controller - StudyPlanController
Methoden Name	getStudyPlan
Beschreibung	Hole den Studienplan über den Namen.
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Ein Optional von StudyPlan wird zurückgegeben, falls der Studienplan existiert.
Negative Case	Ein HTTP 400 wird zurückgegeben, falls der Studienplan nicht existiert.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: StudyPlan</li> <li>• Negative Case: HTTP 400</li> </ul>

## Service-Tests

Tabelle 202: Testfall TS-012. Quelle: Autor

ID / Bezeichnung	TS-012
Referenz zu User-Story	U-14
Testebene	Service - StudyPlanService
Methoden Name	getStudyPlanByName
Beschreibung	Gibt den gefundenen Studienplan anhand des Names zurück.
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Ein Optional von StudyPlan wird zurückgegeben
Negative Case	Ein leeres Option wird zurückgegeben
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Optional&lt;Studyplan&gt;</li> <li>• Negative Case: Optional&lt;empty&gt;</li> </ul>

## 8.8.6 Studienplan löschen

### Controller-Tests

Tabelle 203: Testfall TC-011. Quelle: Autor

ID / Bezeichnung	TC-011
Referenz zu User-Story	U-16
Testebene	Controller - StudyPlanController
Methoden Name	deleteStudyPlan
Beschreibung	Studienplan löschen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Bei Löscherfolgt wird ein HTTP 200 wird zurückgegeben.
Negative Case	Ein HTTP 400 wird zurückgegeben, wenn der Service den Studienplan nicht löschen konnte.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: HTTP 200</li> <li>• Negative Case: HTTP 400</li> </ul>

## Service-Tests

Tabelle 204: Testfall TS-013. Quelle: Autor

ID / Bezeichnung	TS-013
Referenz zu User-Story	U-16
Testebene	Service - StudyPlanService
Methoden Name	deleteStudyPlan
Beschreibung	Studienplan löschen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Studienplan konnte gelöscht werden. Es wird TRUE zurückgegeben.
Negative Case 1	Studienplan konnte nicht gelöscht werden da dieser nicht gefunden wurde. Es wird FALSE zurückgegeben.
Negative Case 2	Studienplan konnte nicht gelöscht werden, da er nach dem Löschen noch vorhanden ist. Es wird FALSE zurückgegeben.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: True</li> <li>• Negative Case 1: False</li> <li>• Negative Case 2: False</li> </ul>

## Repository-Tests

Tabelle 205: Testfall TR-008. Quelle: Autor

ID / Bezeichnung	TR-008
Referenz zu User-Story	U-16
Testebene	Repository - StudyPlanRepository
Methoden Name	delete
Beschreibung	Studienplan löschen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Der delete-Befehl gibt kein Resultat zurück (Void). Es wird mittels abfrage, ob der Studienplan noch vorhanden ist (findById) überprüft ob der Studienplan gelöscht wurde (auf Service-Ebene).
Negative Case	Der delete-Befehl gibt kein Resultat zurück (Void). Es wird mittels abfrage, ob der Studienplan noch vorhanden ist (findById) überprüft ob der Studienplan gelöscht wurde (auf Service-Ebene).
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Void</li> <li>• Negative Case: Void</li> </ul>

## 8.8.7 Studienplan visualisieren

### Controller-Tests

Tabelle 206: Testfall TC-012. Quelle: Autor

ID / Bezeichnung	TC-012
Referenz zu User-Story	U-10
Testebene	Controller - StudyPlanController
Methoden Name	getTotalECTS
Beschreibung	Totalisierte ECTS-Punkte zu einem Studienplan abrufen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall werden die ECTS-Punkte als Integer zurückgegeben.
Negative Case	Es wird auch im Negativen Fall ein Integer zurückgegeben. Jedoch hat der Integer den Wert 0.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Integer</li> <li>• Negative Case: Integer</li> </ul>

Tabelle 207: Testfall TC-013. Quelle: Autor

ID / Bezeichnung	TC-013
Referenz zu User-Story	U-11
Testebene	Controller - StudyPlanController
Methoden Name	getTotalRelevanceBySpecialisation
Beschreibung	Totalisierte Relevanz-Punkte zu einem Studienplan abrufen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall werden die Relevanzpunkte als Integer zurückgegeben.
Negative Case	Es wird auch im Negativen Fall ein Integer zurückgegeben. Jedoch hat der Integer den Wert 0.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Integer</li> <li>• Negative Case: Integer</li> </ul>

## Service-Tests

Tabelle 208: Testfall TS-014. Quelle: Autor

ID / Bezeichnung	TS-014
Referenz zu User-Story	U-10
Testebene	Service - StudyPlanService
Methoden Name	getECTS
Beschreibung	Totalisierte ECTS-Punkte zu einem Studienplan abrufen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall werden die ECTS-Punkte als Integer zurückgegeben.
Negative Case	Es wird auch im Negativen Fall ein Integer zurückgegeben. Jedoch hat der Integer den Wert 0.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Integer</li> <li>• Negative Case: Integer</li> </ul>

Tabelle 209: Testfall TS-015 Quelle: Autor

ID / Bezeichnung	TS-015
Referenz zu User-Story	U-11
Testebene	Service - StudyPlanService
Methoden Name	getTotalRelevanceBySpecialisation
Beschreibung	Totalisierte ECTS-Punkte zu einem Studienplan abrufen
Testvoraussetzung	In Kapitel 8.6.1 definiert
Positive Case	Im Positiven Fall werden die Relevanzpunkte als Integer zurückgegeben.
Negative Case	Es wird auch im Negativen Fall ein Integer zurückgegeben. Jedoch hat der Integer den Wert 0.
Erwartetes Ergebnis	<ul style="list-style-type: none"> <li>• Positive Case: Integer</li> <li>• Negative Case: Integer</li> </ul>

## 8.9 Testplan

Tabelle 210: Testplan. Quelle: Autor

Nr.	Aktivität	Verantwortlich	Mitarbeiter	End-Termin
1	Testfälle im Rahmen der User Stories erstellen und ausführen. Testfälle müssen durchlaufen.	MR	PS, CHG	03.12.2021
3	Testprotokoll und Report erstellen	MR	PS	05.12.2021

## 8.10 Testprotokoll

Das Testprotokoll hält die Testergebnisse fest. Die Testergebnisse werden gemäss den im Testkonzept definierten Mängelklassen bewertet. Der vollständige Bericht ist in der Abgabe im Verzeichnis "Testprotokoll" zu finden.

Tabelle 211: Testprotokoll. Quelle: Autor

Nr.	ID	Bemerkung	Testdatum	Tester	FK*
1	TC-001	Keine	07.11.2021	MR	0
2	TC-002	Keine	07.11.2021	MR	0
3	TS-001	Keine	07.11.2021	MR	0
4	TS-002	Keine	07.11.2021	MR	0
5	TS-003	Keine	07.11.2021	MR	0
6	TR-001	Keine	07.11.2021	MR	0
7	TR-002	Keine	07.11.2021	MR	0
8	TC-003	Keine	07.11.2021	MR	0
9	TC-004	Keine	08.11.2021	MR	0
10	TC-005	Keine	08.11.2021	MR	0
11	TS-006	Keine	08.11.2021	MR	0
12	TS-004	Keine	07.11.2021	MR	0
13	TS-005	Keine	08.11.2021	MR	0
14	TR-003	Keine	08.11.2021	MR	0
15	TR-004	Keine	08.11.2021	MR	0
16	TR-005	Keine	08.11.2021	MR	0
17	TC-006	Keine	08.11.2021	MR	0
18	TS-007	Keine	08.11.2021	MR	0

<b>19</b>	TR-006	Keine	08.11.2021	MR	0
<b>20</b>	TC-007	Keine	13.11.2021	MR	0
<b>21</b>	TS-008	Keine	08.11.2021	MR	0
<b>22</b>	TR-007	Keine	13.11.2021	MR	0
<b>23</b>	TR-008	Keine	20.11.2021	MR	0
<b>24</b>	TR-009	Keine	20.11.2021	MR	0
<b>25</b>	TR-010	Keine	20.11.2021	MR	0
<b>26</b>	TC-008	Keine	13.11.2021	MR	0
<b>27</b>	TS-009	Keine	13.11.2021	MR	0
<b>28</b>	TR-011	Keine	20.11.2021	MR	0
<b>29</b>	TR-012	Keine	20.11.2021	MR	0
<b>30</b>	TC-009	Keine	13.11.2021	MR	0
<b>31</b>	TS-010	Keine	13.11.2021	MR	0
<b>32</b>	TR-013	Keine	20.11.2021	MR	0
<b>33</b>	TC-010	Keine	29.11.2021	MR	0
<b>34</b>	TC-011	Keine	29.11.2021	MR	0
<b>35</b>	TS-014	Keine	05.12.2021	MR	0
<b>36</b>	TS-015	Keine	05.12.2021	MR	0

**Legende:** FK = Mängelklasse (Testergebnis)

## 8.11 Testabdeckung - Report

Für das Projekt StudyBuddy haben wir eine Testabdeckung von über 96% der Klassen, 89% der Methoden und 87% der Code-Linien erreicht. Der vollständige Bericht ist in der Abgabe im Verzeichnis "Testabdeckung" zu finden.

*Tabelle 212: Testreport - Übersicht aller Testfälle. Quelle: Autor*

Package	Klasse, %	Methoden, %	Code -Linien, %
Alle Klassen	96.6% (28/29)	89.3% (125/140)	87.5% (349/399)

### Details nach Package

*Tabelle 213: Testreport - Details nach Package. Quelle: Autor*

Package	Klasse, %	Methoden, %	Code-Linien, %
<b>ch.ffhs.pa5.backend</b>	100% (2/2)	80% (4/5)	88.9% (8/9)
<b>ch.ffhs.pa5.backend.controller</b>	100% (5/5)	100% (17/17)	98.2% (54/55)
<b>ch.ffhs.pa5.backend.controller.exception</b>	66.7% (2/3)	66.7% (2/3)	66.7% (2/3)
<b>ch.ffhs.pa5.backend.controller.model</b>	100% (2/2)	87.5% (14/16)	77.3% (34/44)
<b>ch.ffhs.pa5.backend.model</b>	100% (12/12)	86.8% (66/76)	82.4% (136/165)
<b>ch.ffhs.pa5.backend.service</b>	100% (5/5)	95.7% (22/23)	93.5% (115/123)

## Details nach Klasse

Tabelle 214: Testprotokoll- Details nach Klasse. Quelle: Autor

Class	Klasse, %	Methoden, %	Code-Linien, %
<b>BackendApplication</b>	100% (1/1)	50% (1/2)	50% (1/2)
<b>WebConfig</b>	100% (1/1)	100% (3/3)	100% (7/7)
<b>ModuleController</b>	100% (1/1)	100% (2/2)	100% (4/4)
<b>SpecialisationController</b>	100% (1/1)	100% (2/2)	100% (6/6)
<b>StudyPlanController</b>	100% (1/1)	100% (8/8)	97% (32/33)
<b>StudyProgramController</b>	100% (1/1)	100% (2/2)	100% (4/4)
<b>ValidatorController</b>	100% (1/1)	100% (3/3)	100% (8/8)
<b>BadRequestException</b>	100% (1/1)	100% (1/1)	100% (1/1)
<b>FailureText</b>	0% (0/1)	0% (0/1)	0% (0/1)
<b>NoContentException</b>	100% (1/1)	100% (1/1)	100% (1/1)
<b>AddModuleToStudyPlanRequest</b>	100% (1/1)	88.9% (8/9)	76.9% (20/26)
<b>NewStudyPlanRequest</b>	100% (1/1)	85.7% (6/7)	77.8% (14/18)
<b>Module</b>	100% (2/2)	94.4% (17/18)	82.5% (33/40)
<b>ModulePlan</b>	100% (2/2)	88.2% (15/17)	94.9% (37/39)
<b>ModuleType</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>Relevance</b>	100% (1/1)	60% (6/10)	71.4% (15/21)
<b>SemesterPlan</b>	100% (1/1)	71.4% (5/7)	61.1% (11/18)
<b>SemesterType</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>Specialisation</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>StudyPlan</b>	100% (2/2)	93.8% (15/16)	82.1% (32/39)
<b>StudyProgram</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>ModuleService</b>	100% (1/1)	80% (4/5)	72.4% (21/29)
<b>RandomIdProvider</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>SpecialisationService</b>	100% (1/1)	100% (2/2)	100% (2/2)
<b>StudyPlanService</b>	100% (1/1)	100% (12/12)	100% (88/88)
<b>StudyProgramService</b>	100% (1/1)	100% (2/2)	100% (2/2)

Bei den rot-markierten Zellen handelt es sich entweder um Getter-/Setter-/Hash- oder Equals-Methoden oder Konstruktoren, die nicht getestet werden. Beispielsweise ist die Testabdeckung in der ModuleService-Klasse vollständig, jedoch werden einige schliessende-Klammer als ungetestet interpretiert.



```

68 } }
69 /**
70 */
71 * Liefert eine Liste von Modulen basierend auf dem Modul-Typ.
72 * Falls zum gewählten Modul-Typ keine Module existieren, wird eine leere Liste zurückgegeben.
73 * Modul-Typen sind in einem ENUM hinterlegt.
74 *
75 * @param moduleType Module zum gewählten Modul-Typ
76 * @return Liste von Modulen
77 */
78 @Override
79 public List<Module> getByModuleType(ModuleType moduleType) {
80     var result :Iterable<Module> = this.iModuleRepository.findByModuleType(moduleType);
81     return StreamSupport.stream(result.spliterator(), parallel: false)
82         .collect(Collectors.toList());
83 }
84 }
85

```

Abbildung 61: Fehlerhafte Interpretation der Testabdeckung. Quelle: Autor

## 9 Entwicklungs dokumente

### 9.1 Entwicklungsprozess

Ausgangslage, für die Entwicklung der Software, bilden die Issues, welche nach deren Erstellung im Team besprochen, mit einem Fertigstellungs-Datum versehen und einem Projektmitglied zur Erledigung zugewiesen werden. Jedes Team-Mitglied erarbeitet entweder selbstständig oder mit einem anderen Team-Mitglied in Form der Paarprogrammierung die Umsetzung der zugewiesenen Issues. Generell gilt der Grundsatz, dass sich das Projektteam gegenseitig bei fachlichen und sonstigen Fragen unterstützt.

Die Umsetzungsqualität wird auf Basis des Entwicklungsprozesses gewährleistet, welcher folgende Vorgaben beinhaltet:

- Versionskontrolle mittels GIT: Jede Änderung am Quellcode muss mittels GIT eingereicht werden. Es ist nicht erlaubt, Quellcode direkt im Repository zu ändern.
  - Ausnahmen bilden Änderungen an der readme.md, pom.xml oder sonstigen Architektur-Dateien (application.properties und sonstige Dateien im resources-Verzeichnis).
- Issues können mittels eigener Branches umgesetzt werden. Hierbei können mehrere Issues als «Feature» zusammengesetzt werden.
  - Die Benennung des Branch ist passend (Name ist selbstsprechend) zu wählen.
  - Die Benennung des Branch erfolgt nach dem Camel Case Schema.
  - Fehler/Bugs können direkt auf dem Master-Branch behoben werden
- Die Fertigstellung eines Issues oder einem Feature erfolgt nach dem 4-Augen-Prinzip

Nachfolgende Visualisierung stellt den Entwicklungsprozess als Ablauf nochmals genauer dar:

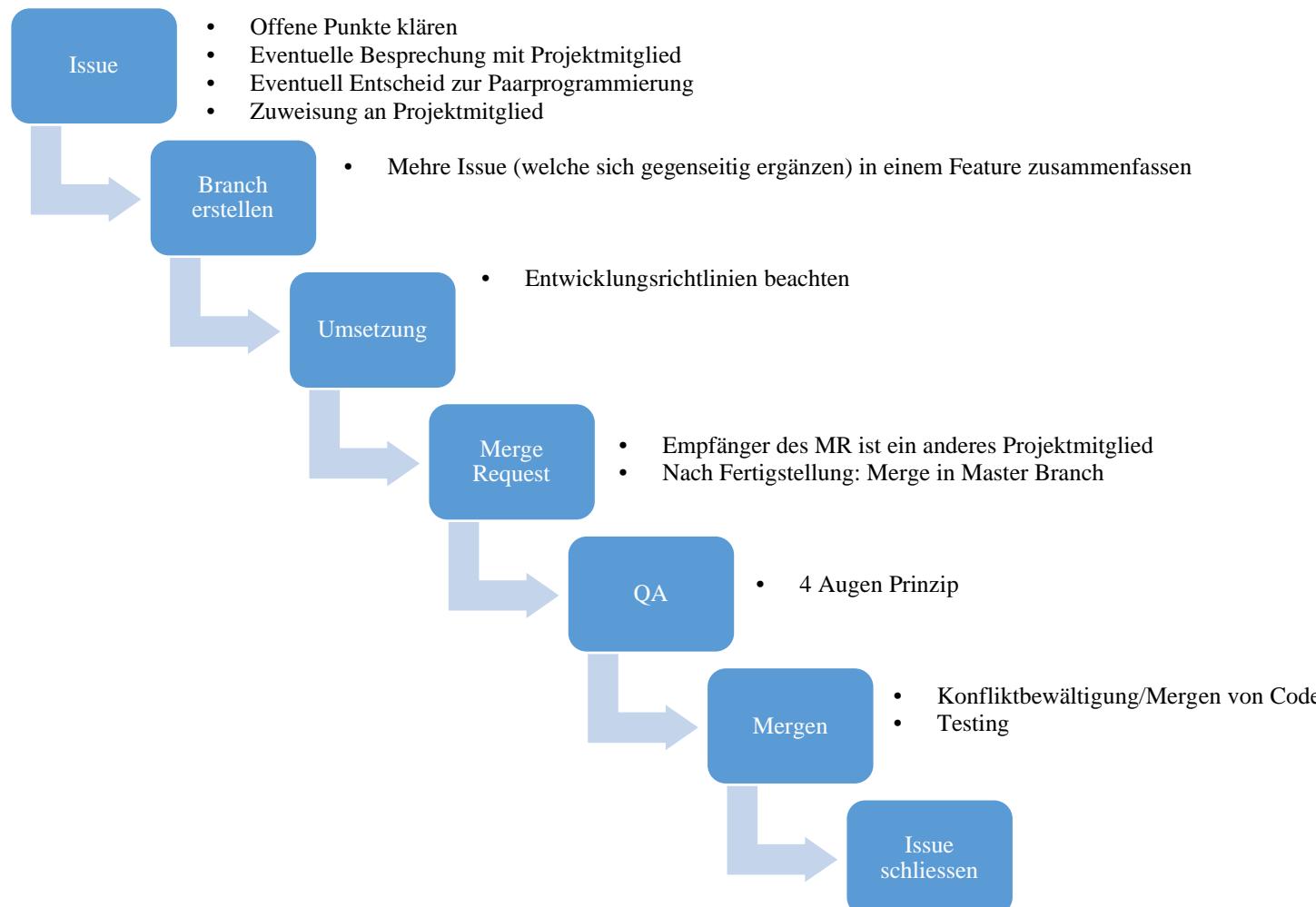


Abbildung 62: Visualisierter Entwicklungsprozess. Quelle: Autor

## 9.2 Technologien und Tools

Die Software wird als Webanwendung umgesetzt. Für die Backend- und Frontend-Entwicklung werden verschiedene Programmiersprachen und Frameworks eingesetzt. Folgende Auflistung stellt dar, welche Technologien eingesetzt werden:

*Tabelle 215: Technologien und Tools. Quelle: Autor*

Bereich	Technologie	Erklärung und Begründung
<b>Backend</b>	Java 16	Die aktuelle Java Version (Release 2021) bietet einige neue Features, welche bereits von anderen Sprachen (Kotlin, C#) bekannt sind und auf dessen Nutzen nicht verzichtet werden sollte (z.B. records, switch-expression) (Oracle, Oracle, 2021).
	Spring Framework	Erleichtert den Umgang mit Dependency Injection, Datenbank-Anbindung (JPA-Data), Controller (API) uvm. (Spring, 2021).
	Maven	Das Erstellen einer ausführbaren JAR-/WAR-Datei wird durch das komfortable Abhängigkeits- und Build-Management unterstützt (Tozzi, 2019).
	JUnit 5	JUnit 5 besteht aus drei verschiedenen Modulen (Platform, Jupiter, Vintage) und wird in diesem Projekt für die Unit-Tests eingesetzt. JUnit 5 nutzt Java 8 Funktionen, dadurch können die Tests performanter und effizienter erstellt und ausgeführt werden (Bechtold, 2021).
<b>Frontend</b>	Angular	Angular wird als Frontend Entwicklungsumgebung eingesetzt. Dadurch ist die Webanwendung skalierbar, reaktiv und durch die gute Tool-Unterstützung optimal umsetzbar (Google, Angular, 2021).
	HTML 5	Der aktuelle Standard für das Webdesign. Durch die Abwärtskompatibilität wird gewährleistet, dass auch ältere Browser-Versionen unterstützt werden (Albers-Zoller, 201).
	SCSS	Die Stylesheets (CSS) werden mittels SCSS erstellt. SCSS unterstützt den Entwickler bei der Erstellung von CSS-Dateien. SCSS wird durch einen Compiler ins gebräuchliche CSS kompiliert (Spinnler, kein Datum).

Die Wahl der Entwicklungstools ist jedem Projektemitglied der persönlichen Präferenz überlassen. Eine gemeinsame Basis von Entwicklungstools erleichtert jedoch die Hilfestellung bei allfälligen Problemen. Der Einsatz, gemäss Kapitel 3.8 wird deshalb empfohlen.



## 9.3 Entwicklungsrichtlinien

Die Qualität der Software beinhaltet auch die einheitliche und korrekte Umsetzung in der jeweiligen Entwicklungssprache. Deshalb werden folgende Richtlinien für den Java-Quellcode festgelegt:

- Camel Case Namenskonvention (berechneRelevanzPunkte(), variableA).
- Organisation in Pakete/Schichten.
- Arbeiten mit Interfaces - Zwecks Erweiterbarkeit.
- Nicht mehr veränderbare Variablen werden als final deklariert.
- Alle Java-Klassen und öffentlichen Methoden mit Javadoc dokumentieren:
  - erlaubte und nicht erlaubte Inputparameter
  - Varianten der Ausnahmebehandlung und die zu erwartenden Rückgabewerte
  - Dokumentation jeweils vor einer offiziellen Projektabgabe mit  
mvn javadoc:javadoc  
erzeugen ./backend/src/target/site/apidocs/index.html

Für das Frontend gelten andere Richtlinien, da sich die sprachliche Syntax von Typescript zu Java unterscheidet:

- Verschachtelte HTML-Tags mittels Tabulatoren einrücken.
- Für Sonderzeichen (Umlaute, Dollarzeichen usw.) in HTML die entsprechenden HTML-Unicodes verwenden.
- Organisation des Codes in Komponenten (jede Komponente ist ein eigener Ordner).
- Globale Stylesheets. Abweichungen zum globalen Stylesheet werden in den jeweiligen Spezifizierungs-Dateien (komponente1.scss) umgesetzt.
- Typsicherheit gewährleisten: Keine Any-Klassen verwenden
- Falls ein Lifecycle-Management der Klassen benötigt wird, wird dies mittels Implementierung der OnInit-, OnDestroy-Klassen umgesetzt.
- Subscriptions müssen im Lifecycle behandelt werden (im OnDestroy werden die Subscriptions wieder aufgelöst).

Ergänzend gelten nachfolgende Richtlinien:

- Sinnvolle und sprechende Benennung der Variablen/Klassen/Pakete usw.
- Konsolen-Ausgaben (zwecks Fehlerbehandlung o.Ä.) und nicht mehr benötigte Kommentare müssen vor dem Merge Request wieder entfernt werden.
- Sichtbarkeiten von Methoden/Variablen konsequent nutzen (Private, Public usw.)
- Vernünftige Größen der Klassen und Methoden:
  - Trennung der Zuständigkeiten (eine Klasse resp. Methode hat genau eine Aufgabe/einen Zweck).
  - Koppelung vermeiden (Klassen sollten so wenig voneinander wissen wie möglich).
- Statische Methoden sind zu vermeiden.
  - Ausnahmen bilden Utility-Klassen (Helper-Klassen).
- Behandlung der Ausnahmefälle (Exceptions)

## 10 Inbetriebnahme

### 10.1 Lieferobjekte

Auf dem gelieferten USB-Stick ist folgende Ordner-Struktur ersichtlich:

Tabelle 216: Lieferobjekte - Verzeichnisstruktur. Quelle: Autor

Verzeichnis	Beschreibung
\Quellcode	Quellcode
\Quellcode\Backend	Backend (Java)
\Quellcode\Frontend	Frontend (Angular)
\JavaDoc	Java Dokumentation als JavaDoc HTML-Dateien
\Testabdeckung	Testabdeckung als HTML-Dateien
\Testprotokoll	Testprotokoll als HTML-Dateien
\API-Dokumentation	API-Dokumentation als HTML-Dateien
\Ressourcenplanung	Die Ressourcenplanung als Excel-Datei
\Statusmeeting	Bericht zu Statusmeeting
\Frontend-Dokumentation	Dokumentation des Frontend

### 10.2 Installationsanleitung

**Wichtig:** Das System verwendet die Ports 3306 (MySQL), 8080 (Backend) und 4200 (Frontend). Diese sind unbedingt freizuhalten, sonst kann das System nicht gestartet werden. Unter Windows kann mit dem Befehl `netstat` geprüft werden, ob die Ports frei sind:

```
netstat -aon | find "4200"
```

und mit `taskkill` kann ein entsprechender Prozess beendet werden, welcher den Port besetzt:

```
taskkill /f /pid <PID>
```

#### 10.2.1 Variante 1: Docker starten

Die einfache Variante erfordert als Vorbereitung die Installation von Docker auf dem Hostsystem:

1. Docker Desktop installieren: <https://www.docker.com/products/docker-desktop>
2. Docker ausführen, um weiterzufahren

Der mitgelieferte Datenträger enthält ein Verzeichnis namens «studybuddy\_docker»

3. Das Verzeichnis studybuddy\_docker samt Inhalt auf dem Hostsystem platzieren
4. Mit der Kommandozeile in das gerade platzierte Verzeichnis navigieren
5. Folgenden Befehl bei vorhandener Internetverbindung ausführen:

```
docker compose up
```

Docker wird nun zwei Container aus dem DockerHub herunterladen und starten. Sollte dies erfolgreich durchgeführt worden sein, kann mittels Browser auf «localhost:8080» navigiert werden, um die Applikation zu benutzen.



## 10.2.2 Variante 2: Projekt manuell starten

Als Fallback-Methode bietet sich jedoch nachfolgende Start-Methode an, welche sich bewährt hat (auch wenn diese ein wenig aufwendiger ist):

Folgende Voraussetzungen sind für das Starten des Systems zu erfüllen:

1. Docker Desktop installieren: <https://www.docker.com/products/docker-desktop>
2. NodeJS installieren (Version 16.13.0): <https://nodejs.org/en/blog/release/v16.13.0/>
  - a. Bei der Installation können weitere Abhängigkeiten wie Chocolatey installiert werden. Es ist von Vorteil, diese auch zu installieren
3. Java SE (Version 16.0.2): <https://www.oracle.com/java/technologies/javase/jdk16-archive-downloads.html>

Startvorgang der einzelnen Komponenten:

1. Die Docker-Desktop Anwendung muss im Hintergrund laufen
2. MySQL starten  
Im Projektverzeichnis (im Hauptverzeichnis des Projekts) folgenden Befehl in der CMD ausführen

```
docker compose up db
```

3. Backend starten  
Im Projektverzeichnis (im Hauptverzeichnis des Projekts) folgenden Befehl in der CMD ausführen

```
java -jar backend-1.0.0.jar
```

4. Frontend starten  
Im Frontend-Verzeichnis (Verzeichnis: Frontend) folgenden Befehl in der CMD ausführen

```
npm install -g @angular/cli@12.2.0
npm install
ng serve
```

5. Browser starten  
In der Adresse folgendes eingeben: **http://localhost:4200**

### Getestet mit

Angular CLI: 12.2.0

NPM: 6.14.5

Node version: v16.13.1

Java Version: build 16.0.2+7-67

## 10.3 Benutzerhandbuch

### 10.3.1 Menüleiste



Abbildung 63: Benutzerhandbuch - Menüleiste. Quelle: Autor

1. «Study-Buddy» als Home-Verlinkung
2. «Home» führt auf die Home-Site zurück
3. «Studienplan» leitet zur Erstellung eines neuen Studienplans
4. «Impressum» wird verwendet, um Infos über die Ersteller und Betreiber zu hinterlegen
5. «Suchfunktion» kann durch die Eingabe des Namens oder der ID eines zuvor erstellen Studienplans, dieser wieder gefunden werden

### 10.3.2 Home

Die Home-Site wird verwendet, um eine kurze Einführung zu geben worum es sich handelt und wie das Tool zu handhaben ist. Wichtige Informationen zu Einschränkungen für den Benutzer können einfach publiziert werden.

Dient generell als Einstieg der Applikation.

### 10.3.3 Studienplan erstellen



Abbildung 64: Benutzerhandbuch - Studienplan erstellen. Quelle: Autor

1. Bestimmt durch Auswahl aus Drop-Down die Art des Studiengangs
  - Momentan wird lediglich der Studiengang HF Passerelle BSc Informatik unterstützt
2. Eingabe der gewünschten Bezeichnung des Studienplans
3. Durch Bestätigung mit «OK» wird ein neuer Studienplan angelegt
  - Falls ein Studienplan mit dem gleichen Namen existiert, wird der OK Button deaktiviert.

## 10.3.4 Studienplan editieren

### Intro:

**Deine Auswahl**

Auf Basis deiner gewählten Angaben werden die dir zur Verfügung stehenden Module angeboten.  
Der Name, deines Studienplans, ist dein persönlicher Zugangscode mit welchem du auf den Plan zugreifen kannst. Merke in dir also gut!

Studienprogramm: HF Passarelle  
Studienplan Name: CHG\_2022-2024

1 Studienplan Identifikation: 7ebe0d04-83c5-42ad-98fe-9464e588cd16

Abbildung 65: Benutzerhandbuch - "Deine Auswahl". Quelle: Autor

- Eigenständig generierte Studienplan-ID

### Modul zu Studienplan hinzufügen:

**Modul dem Studienplan hinzufügen**

1 Jahr: 2022  
2 Semester-Typ: Wähle den Semester-Typ (Herbst/Frühling)  
3 Wähle die Vertiefungsrichtung: Keine  
4 Semester: Wähle das Semester  
5 Module:

Einführung in Data Science, Informations-Sicherheit, Diskrete Mathematik und lineare Systeme, Datenstrukturen & Algorithmen, Wahrscheinlichkeitsrechnung und Statistik, Analysis mit Python, Projektarbeit, Lineare Algebra, Seminararbeit, Innovation & Technologie, Bachelor-Thesis, MaLe: Machine Learning, IR: Information Retrieval, BDN: Big Data NoSQL, NA: Netzwerk Analyse, VSA: Verteilte Systeme und Anwendungen, JEE: Java Enterprise Edition, UDB: Unternehmensdatenbanken, SOA: Service Oriented Architecture, INSich: Internet Sicherheit, RISich: Reaktive Informationssicherheit, Krypt: Kryptographie, M-ITSec: Management der Informationssicherheit, GameDev: Game Development, MK: Mobilkommunikation, SWQ: Software-Qualität, C++, FWebT: Fortgeschritten Web-Technologien, ES&HH: Embedded Systems und Hardware Hacking, DWBI: Data Warehousing und Business Intelligence, FAC: Formale Sprachen, Automaten, Compilerbau, DevOps: Development Operations, CICo: Cloud Computing, DeepL: Deep Learning, RBD: Realtime Big Data, ESA: Embedded Systems mit Android, BLK: Blockchain

Legende: Basismodul, Wahlmodul, Vertiefungsmodul

Abbildung 66: Benutzerhandbuch - "Modul hinzufügen". Quelle: Autor

- Durchführungsstart des Moduls
- Wahl des Semesters
- Angestrebte Vertiefungsrichtung
- Semester des eigenen Studienganges
- Anhand der Angaben gefilterte Module

Module können mit Linksklick dem Studienplan hinzugefügt werden. Anschliessend erscheinen sie in der Liste, sowie der Visualisierung.

### Studienplan:

Studienplan			1
Semester	Module		Löschen
8	CICo: Cloud Computing		<button>Löschen</button>
8	INSich: Internet Sicherheit		<button>Löschen</button>

Abbildung 67: Benutzerhandbuch- Studienplan "Modulliste". Quelle: Autor

1. Module können direkt aus dem Studienplan gelöscht werden

### Visualisierung:

Visualisieren							
Modul	Jahr	Frühling	Herbst	ISec	Enterpr.C	DataSc	ECTS
CICo: Cloud Computing	2022	<b>Wahlmodul</b>		1	2	1	5
INSich: Internet Sicherheit	2022	<b>Vertiefungsmodul</b>		2			5
Summe				3	2	1	10

Abbildung 68: Benutzerhandbuch - "Visualisierung". Quelle: Autor

Auflistung der Module mit den entsprechenden Daten zu Relevanz- und ECTS-Punkten

### Studienplan löschen:

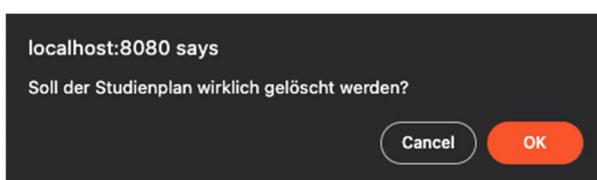
Unter der Visualisierung besteht die Möglichkeit den gerade geöffneten Studienplan dauerhaft zu löschen.

**Studienplanung Löschen**

Hiermit wird der Studienplan dauerhaft gelöscht. Dies kann nicht rückgängig gemacht werden!

Löschen

Bei der Betätigung von «Löschen» wird via Alert eine Bestätigung des Benutzers verlangt. Im Falle einer Bestätigung mit «OK» wird der Studienplan gelöscht.





## 11 Abschlussbericht

### 11.1 Aufgetretene Probleme

**Technologisch:** drei verschiedene Systeme lokal Betreiben resp. als Docker zur Verfügung stellen.

**Layout Desktop und Mobile Browser:** Es mussten wiedererwartet Anpassungen vorgenommen werden, um die Ansicht für den mobilen Client nutzbar zu machen.

**Ressourcenengpass:** Durch Mehraufwand aus dem 1.Status-Meeting mussten weitere Massnahmen getroffen werden, um die Planung zu schärfen und belastbar zu machen.

**Motivationshänger:** Durch die temporäre Mehrbelastung und den anhaltenden Druck rund um das Modul PA5, musste darauf geachtet werden, dass die Motivation der Projektteilnehmer nicht zu sehr abnimmt. Hierzu wurden im uns möglichen Rahmen Ausflüge geplant, um Themen ausserhalb des Projektes zu besprechen.

**Einsatz unbekannter Komponente:** Für verschiedene Komponente der Umsetzung musste zuerst ein Proof of Concept erstellt werden, welches nicht in jedem Fall die gewünschten Ergebnisse erbringen konnte. Beispiel: Die Funktion Drag&Drop musste so fallen gelassen werden.

**Einhaltung regelmässiger Termine:** Bereits bei drei Projektmitgliedern werden regelmässige Termine komplizierter einzuhalten. Einige der wöchentlichen Sitzungen mussten verschoben oder im schlimmsten Fall sogar gestrichen werden.

**Wahl der Tools:** Einzelne der den Teilnehmern bereits bekannten Tools konnten nicht OS übergreifend genutzt werden.

### 11.2 Auswertung Planung

Mit anfänglichen Schwierigkeiten eine belastbare Planung zu gewährleisten konnte ab dem ersten Status-Meeting die Planung so weit angepasst werden, dass bestimmte Aussagen getroffen werden können zu jeweiligen Arbeitspaketen.

Durch die Methode der 3 Schätzungen konnten recht akkurate Zeitschätzungen erstellt werden, die mit vorschreitendem Projekt verfeinert werden konnten. Für weitere Projekte muss der Ressourcenplanung von Beginn an den ihr zustehenden Wert beigemessen werden. Für den Verlauf eines Projektes und das Wohlbefinden der beteiligten Projektmitglieder, muss stets ein Auge auf die individuelle Auslastung geworfen werden. Ein essenzieller Punkt in der gegenseitigen Abstimmung und Kontrolle kommen die wöchentlichen Projektmeetings zur Geltung.

Die Planung konnte nach der grossen Überarbeitung lediglich mit kleineren Anpassungen während der wöchentlichen Sitzung wie angedacht durchgeführt werden. Abweichungen zu geschätzten Zeiten traten durchgehend in kleiner werdendem Rahmen ein. Es bestand zu keinem Zeitpunkt das Bedürfnis einer Sofortmassnahme.



## 11.3 Lessons Learned

**Christoph:**

Eine anfängliche Ressourcenplanung macht Sinn bei einem so umfangreichen Projekt. Generell habe ich den Aufwand des Projektmanagement in diesem Rahmen unterschätzt. Die Definition von Arbeitspaketen und deren Einteilung sowie Kontrolle der erledigten Arbeiten kostet viel Zeit. Der Aufwand diese Techniken für den Rahmen der Schule zu erarbeiten sind ebenfalls hoch. Die Erfahrungen in der Praxis zeigen in der Softwareentwicklung den agilen Ansatz da besser auf unvorhergesehenes reagiert werden kann.

Das Zusammenspiel der verschiedenen Persönlichkeiten im Pair-Programming hat mir grosse Freude bereitet. Ich würde diese Form jederzeit wieder praktizieren in einem nächsten Projekt. Durch meine eher begrenzten Programmierfähigkeiten konnte ich so von den Gedankengängen der Geübteren lernen und eigene Fortschritte erzielen.

Das eingesetzte Tool zur kreativen Zusammenarbeit, das Miro-Board, eignet sich nur teilweise für diese Art der Projektführung. Die flexible Darstellung ist für ein agiles Vorgehen ausgelegt und benötigt einiges an Pflege, um aktuell gehalten zu werden. Wurden die Sitzungen protokolliert oder Diagramme erstellt, können sie nur auf Umwegen mit ausreichender Qualität exportiert werden.

Die Drei-Zeiten-Methode zur Aufwandschätzung eignet sich gut für grössere Arbeitspakete mit Aufwand von 10h+. Bei kleineren Aufwandschätzungen ergibt sich in fast keinem Falls eine Abweichung zum ursprünglich geschätzten Wert. Je grösser der Spielraum zwischen schlechtestem Fall und bestem Fall desto realistischer wird die Schätzung.

**Marco:**

Am Anfang hatte ich grosse Mühe die Rolle des Projektleiters zu übernehmen, weil Christoph als erster startete und den Job von Anfang an gut gemacht hat. Der Workflow schwächte kurz ab, bis ich meine Rolle als Projektleiter dann wahrnehmen konnte.

In meinem Berufsumfeld benutzen wir leider kein Projektmanagement und ich habe gelernt, dass eine sorgfältige Planung sehr wichtig für den Projekterfolg ist.

Ich durfte das erste Mal beim Pair-Programming dabei sein und konnte dadurch lernen wie man Testfälle in JAVA mit JUnit schreibt und wie die Software-Architektur mit dem Frontend und Backend funktioniert.

Zum Starten unserer Anwendung fanden wir, dass es ein Start-Script geben sollte, welches Docker-Container erstellt und der Anwender dann ohne Probleme den Browser mit unserer App aufrufen kann. Es hat sich jedoch herausgestellt, dass es doch nicht so einfach ist, damit es auf verschiedenen Systemen funktioniert. Geplant waren 12 Stunden und schlussendlich haben wir über 25 Stunden mit probieren und scheitern vergeudet. Genau für solche Misserfolge haben wir einen Puffer in der Ressourcenplanung mit eingeplant. Das Arbeitsverhältnis empfand ich als angenehm und alle waren aktiv mit Eigeninitiativen dabei.

**Peter:**

Die wichtigste Erkenntnis, die ich aus diesem Projekt ziehe, ist die vorgängige und vor allem umfassende Planung der Tätigkeiten. Aus meinem beruflichen Umfeld bin ich es gewohnt, Aufgaben mit wenig Informationen iterativ und agil anzugehen. Dieses Vorgehen war in diesem Projekt nicht möglich. Die Gründe dafür waren die definierten Ressourcen (Zeit) und die definierten Vorgaben (Projektmanagement). Wir sind mit wenig vorgängiger Planung im ersten Statusmeeting komplett auf Eis gelaufen; was uns aber im Rückblick guttat. Wir haben auf Basis des Feedbacks aus dem ersten Statusmeeting viele Personenstunden und Aufwand in die Planung gesteckt, womit das Projekt aus meiner Sicht zu einem vollen Erfolg wurde. Weiteres, aus der Berufswelt bekannte Vorgehen, konnte in diesem Projekt nicht beibehalten werden, womit ich mich am Anfang schwertat. Ich bin es nicht gewohnt, umfassende Requirements- und sonstige Software-Analysen in Form von UML-Diagrammen anzufertigen. Die Projektarbeit hat mir insofern einen richtigen Know-How-Schub in Sachen Projektmanagement und Software-Engineering erbracht.

Die Stimmung und der Umgang im Team empfand ich zu jederzeit als sehr kollegschaftlich, rücksichtsvoll und unterstützend. Es war das perfekte Team für eine solche Arbeit und ich würde es sofort wieder mit denselben Personen durchführen.

## 12 Anhang

### 12.1 Bekannte Fehler

Beschreibung	Fehler-Grad
<p>Unter Umständen werden Umlaute im UI nicht korrekt dargestellt. Beispielsweise wird das Modul "Einführung in Data Science" auf Marco's Computer wie folgt dargestellt:</p> 	Nicht schwerwiegend.
<p>Bei der Erstellung eines neuen Studienplans wird für einen kurzen Moment die 404 Ansicht aufgeblendet. Bis der Studienplan erstellt und wieder von der Datenbank geholt werden konnte.</p> 	Nicht schwerwiegend.

### 12.2 Literaturverzeichnis

- Admin. (kein Datum). *Statistischer Lohnrechner 2018*. Von Schweizer Eidgenossenschaft:  
<https://www.gate.bfs.admin.ch/salarium/public/index.html#/start> abgerufen
- Albers-Zoller, J. (15. Januar 201). *selfhtml*. Von HTML/Tutorials/HTML5:  
<https://wiki.selfhtml.org/wiki/HTML/Tutorials/HTML5> abgerufen
- Andrea. (25. August 2016). *Paarweise Vergleich*. Von projekte-leicht-gemacht: <https://projekte-leicht-gemacht.de/blog/pm-methoden-erklaert/paarweiser-vergleich/> abgerufen
- Angular. (2021). *The MIT License*. Von Angular: <https://angular.io/license> abgerufen
- Bechtold, S. (12. April 2021). *Junit*. Von JUnit 5 User Guide: <https://junit.org/junit5/docs/current/user-guide/> abgerufen
- Dr. Angermeier, G. (14. Mai 2014). *Bestandteile eines Projektauftrags - mit Checkliste*. Von Projektmagazin: <https://www.projektmagazin.de/projektauftrag> abgerufen
- Dr. Starke, G. (2021). *Arc42*. Von BAUSTEINSICHT: [https://www.dokchess.de/05\\_bausteinsicht/](https://www.dokchess.de/05_bausteinsicht/) abgerufen
- Ehrlich, L. (14. November 2019). *Pair Programming: Agile Softwareentwicklung*. Von real.digital:  
<https://t2informatik.de/wissen-kompakt/pair-programming/> abgerufen
- Eidgenossenschaft, S. (2021). *HERMES online*. Von HERMES online: <https://www.hermes.admin.ch/> abgerufen
- FFHS. (2021). *Vorgaben zur Projektarbeit*. Von FFHS:  
[https://moodle.ffhs.ch/pluginfile.php/4163578/mod\\_resource/content/1/PA\\_Arbeitsmappe\\_HS20-21\\_V1-0.pdf](https://moodle.ffhs.ch/pluginfile.php/4163578/mod_resource/content/1/PA_Arbeitsmappe_HS20-21_V1-0.pdf) abgerufen
- FFHS. (kein Datum). *BSc Informatik*. Von Fernfachhochschule Schweiz:  
<https://www.ffhs.ch/de/bachelor/informatik#aufbau> abgerufen
- FFHS. (kein Datum). *Passerelle Informatik*. Von Fernfachhochschule Schweiz:  
<https://www.ffhs.ch/de/bachelor/bsc-informatik/passerelle> abgerufen
- Geekbullet Admin. (25. April 2021). *Run Android Apps in Chrome Browser – Step by Step Guide*. Von Geekbullet: <https://geekbullet.com/run-android-apps-in-chrome-browser/> abgerufen
- gluonhq. (2021). *JavaFXPorts*. Von gluonhq: <https://gluonhq.com/products/mobile/javafxports/> abgerufen
- Gomes, P. (09 2017). *Medium*. Von Medium: <https://medium.com/@pjbgf/title-testing-code-ocd-and-the-aaa-pattern-df453975ab80> abgerufen
- Google. (2021). *Angular*. Von Introduction to the Angular Docs: <https://angular.io/docs> abgerufen

- Google. (2021). *Angular*. Von Introduction to Angular concepts: <https://angular.io/guide/architecture> abgerufen
- Grossmann, C. (31. August 2021). *PA5\_Studienplaner*. Von Git FFHS: [https://git.ffhs.ch/christoph.grossmann/pa5\\_studienplaner](https://git.ffhs.ch/christoph.grossmann/pa5_studienplaner) abgerufen
- Hruschka, P. (2014). *Business Analysis und Requirements Engineering*. München: Hanser.
- ISTQB. (2011). *ISTQB*. Von International Software Testing Qualifications Board: <https://glossary.istqb.org/de/term/raci-matrix-1> abgerufen
- Koller, D. (31. März 2021). *Dependency-Injektion in Spring Boot*. Von Dev Insider: <https://www.dev-insider.de/dependency-injektion-in-spring-boot-a-1005234/> abgerufen
- Kuhrmann, M. (2018). *Enzyklopädie der Wirtschaftsinformatik*. Von Rational Unified Process (RUP): [https://www.enzyklopädie-der-wirtschaftsinformatik.de/lexikon/is-management/Systementwicklung/Vorgehensmodell/Rational-Unified-Process-\(RUP\)](https://www.enzyklopädie-der-wirtschaftsinformatik.de/lexikon/is-management/Systementwicklung/Vorgehensmodell/Rational-Unified-Process-(RUP)) abgerufen
- Luber, S. (12. Juni 2017). *Was ist NoSQL?*. Von BigData Insider: <https://www.bigdata-insider.de/was-ist-nosql-a-615718/> abgerufen
- Madauss, B.-J. (2016). *Projektmanagement*. Springer.
- Oracle. (10. April 2019). *Oracle Technology Network License Agreement for Oracle Java SE*. Von Oracle: <https://www.oracle.com/downloads/licenses/javase-license1.html> abgerufen
- Oracle. (2021). *Guide to MySQL and NoSQL - Delivering the Best of Both Worlds*. Von MySQL: <https://www.mysql.com/why-mysql/white-papers/guide-to-mysql-and-nosql-delivering-the-best-of-both-worlds/> abgerufen
- Oracle. (2021). *Oracle*. Von JDK Release Notes: <https://www.oracle.com/java/technologies/javase/jdk-relnotes-index.html> abgerufen
- Rau, K.-H. (2007). *Objektorientierte Systementwicklung*. Wiesbaden: Vieweg Verlag.
- Red Hat Limited. (8. Januar 2019). *Red Hat Limited*. Von Was ist eine IDE?: <https://www.redhat.com/de/topics/middleware/what-is-ide> abgerufen
- repdex. (21. Januar 2021). *Android Emulators for iOS*. Von repdex: <https://repdex.online/top-7-best-android-emulators-for-ios/> abgerufen
- Riaan, N. (13. Dezember 2016). *Design Patterns: The Builder Pattern*. Von Dzone: <https://dzone.com/articles/design-patterns-the-builder-pattern> abgerufen
- Robert, C. (1. November 2013). *Unit Tests mit Mockito*. Von entwickler.de: <https://entwickler.de/software-architektur/cocktail-gefallig/> abgerufen
- Sandec GmbH. (kein Datum). *Java in the Browser*. Von JPRO: <https://www.jpro.one/> abgerufen
- Schlisch, M. (2019). Softwaretesten nach ISTQB für dummies. In M. Schlisch, *Softwaretesten nach ISTQB für dummies* (S. 360). John Wiley & Sons.
- solid IT gmbh. (2021). *DB-Engines Ranking of Document Stores*. Von DB Engines: <https://db-engines.com/en/ranking/document+store> abgerufen
- Spinnler, V. (kein Datum). *nextgeneration*. Von SCSS / SASS: <https://nextgeneration.mysign.ch/de/themen/wissen/frontend-entwicklung/scss-sass.html> abgerufen
- Spring. (2021). *Spring*. Von Why Spring?: <https://spring.io/why-spring> abgerufen
- Starke, G. (2009). *Effektive Software Architekturen*. München: Hanser.
- Stückler, M. (24. Februar 2020). *t3n*. Von Was ist eigentlich Github?: <https://t3n.de/news/eigentlich-github-472886/> abgerufen
- Switch. (2021). *SWITCHaai*. Von Switch: <https://www.switch.ch/aai/> abgerufen
- Tatakaus. (25. Juni 2020). *Excel Gantt Diagramm erstellen 3.0 - mit Abhängigkeiten, Meilensteinen & Plan Soll Ist [Vorlage]*. Von Youtube: <https://www.youtube.com/watch?v=uODpxKrib1M> abgerufen
- Team, T. J. (2021). *JUnit 5*. Von JUnit 5: <https://junit.org/junit5> abgerufen
- Tozzi, C. (01. Januar 2019). *ComputerWeekly*. Von Was Apache Maven als Java Build Tool einzigartig macht: <https://www.computerweekly.com/de/tipp/Was-Apache-Maven-als-Java-Build-Tool-einzigartig-macht#:~:text=Maven%20bietet%20automatisiertes%20Abh%C3%A4ngigkeitsmanagement%3A%20Maven,l%C3%A4dt%20sie%20bei%20Bedarf%20herunter.> abgerufen
- Vettor, R. (19. Januar 2021). *Vergleich der relationalen und NoSQL-Daten*. Von Microsoft: <https://docs.microsoft.com/de-de/dotnet/architecture/cloud-native/relational-vs-nosql-data> abgerufen

## 12.3 Abbildungsverzeichnis

Abbildung 1: Ausschnitt des Curriculums der FFHS. Quelle: Autor .....	10
Abbildung 2: Projektphasen, Quelle: Autor .....	16
Abbildung 3: Projektorganisation. Quelle: Autor .....	18
Abbildung 4: Risikomatrix, Quelle: Autor .....	21
Abbildung 5: Aktualisierte Risikomatrix, Quelle: Autor .....	21
Abbildung 6: Projektplanung - Ablauf. Quelle: Autor .....	22
Abbildung 7: Ressourceneinsatzplan - Ausschnitt 1. Quelle: Autor .....	30
Abbildung 8: Ressourceneinsatzplan - Ausschnitt 2. Quelle: Autor .....	30
Abbildung 9: Ressourceneinsatzplan - Ausschnitt 3. Quelle: Autor .....	30
Abbildung 10: Projektstrukturplan. Quelle: Autor .....	31
Tabelle 11: RACI-Matrix der Arbeitspakete. Quelle: Autor .....	33
Abbildung 12: Gantt Diagramm. Quelle: Autor .....	58
Abbildung 13: Projektnetzplan. Quelle: Autor .....	59
Abbildung 14: Von den Anforderungen zu User Stories. Quelle: Autor .....	63
Abbildung 15: Systemidee, Autor: Eigene Abbildung.....	63
Abbildung 16: Haupt Use Case (Vogelperspektive) welches die Aktivitäten des Studienplaners beinhaltet. Quelle: Autor .....	64
Abbildung 17: Use Case "Studiengang planen". Quelle: Autor .....	65
Abbildung 18: Use Case "Modul Informationen sichten". Quelle: Autor .....	66
Abbildung 19: Use Case "Studiengang editieren". Quelle: Autor .....	66
Abbildung 20: Use Case "Modul eintragen". Quelle: Autor .....	67
Abbildung 21: Mockup Home (Startseite). Quelle: Autor .....	68
Abbildung 22: Mockup Home (Startseite) auf Mobilgerät. Quelle: Autor .....	69
Abbildung 23: Mockup Neuen Studienplan erstellen. Quelle: Autor .....	70
Abbildung 24: Mockup Neuen Studienplan erstellen auf Mobilgerät. Quelle: Autor .....	71
Abbildung 25: Mockup bestehenden Studienplan laden. Quelle: Autor .....	72
Abbildung 26: Elementbeschreibung und Typ vom Mockup Studienplan laden auf Mobilgeräten. Quelle: Autor .....	73
Abbildung 27: Studiengang planen. Quelle: Autor .....	74
Abbildung 28: Studiengang planen (1) auf Mobilgeräten. Quelle: Autor .....	75
Abbildung 29: Studiengang planen (2) auf Mobilgeräten. Quelle: Autor .....	76
Abbildung 30: Studiengang planen (3) auf Mobilgeräten. Quelle: Autor .....	77
Abbildung 31: Mockup Impressum. Quelle: Autor .....	78
Abbildung 32: Mockup Impressum auf Mobilgeräten. Quelle: Autor .....	79
Abbildung 33: Fachklassenmodell. Quelle: Autor .....	92
Abbildung 34: Bausteinsicht Ebene 1. Quelle: Autor .....	93
Abbildung 35: Bausteinansicht-Ebene 2 - Frontend. Quelle: Autor .....	94
Abbildung 36: Bausteinansicht-Ebene 2 - Backend. Quelle: Autor .....	96
Abbildung 37: Bausteinansicht Ebene 2 – Datenbanken. Quelle: Autor .....	98
Abbildung 38: Datenmodell des Studienplaners. Quelle: Autor .....	99
Abbildung 39: Klassendiagramm der BackendApplication. Quelle: Autor .....	102
Abbildung 40: Klassendiagramm der Model-Klassen. Quelle: Autor .....	103
Abbildung 41: Klassendiagramm der Repository-Klassen. Quelle: Autor .....	106
Abbildung 42: Klassendiagramm der Service-Klassen. Quelle: Autor .....	107
Abbildung 43: Klassendiagramm der Controller-Klassen .....	111
Abbildung 44: Klassendiagramm der Model-Klassen (innerhalb der Controller-Schicht). Quelle: Autor .....	115
Abbildung 45: Klassendiagramm der Exception-Klassen. Quelle: Autor .....	116
Abbildung 46: Zustandsdiagramm - Studienplan. Quelle: Autor .....	125
Abbildung 47: Aktivitätsdiagramm - Studienplan. Quelle: Autor .....	126
Abbildung 48: Sequenzdiagramm - Studienplan erstellen- Studienprogramme laden. Quelle: Autor .....	127
Abbildung 49: Sequenzdiagramm - Studienplan erstellen- Name validieren. Quelle: Autor .....	127
Abbildung 50: Sequenzdiagramm - Studienplan erstellen. Quelle: Autor .....	128

Abbildung 51: Sequenzdiagramm - Studienplanung anzeigen. Quelle: Autor .....	129
Abbildung 52: Sequenzdiagramm – Modul eintragen. Quelle: Autor .....	130
Abbildung 53: Sequenzdiagramm - Studienplan laden. Quelle: Autor .....	131
Abbildung 54: Frontend Übersicht der Komponenten. Quelle: Autor.....	131
Abbildung 55: Übersicht der SharedModule. Quelle: Autor .....	132
Abbildung 56: Impressum Component. Quelle: Autor .....	133
Abbildung 57: Header Component. Quelle: Autor .....	133
Abbildung 58: Home Component. Quelle: Autor.....	133
Abbildung 59: Frontend Routing auf Komponenten. Quelle: Autor.....	134
Abbildung 60: Teststrategie. Quelle: Autor .....	136
Abbildung 61: Fehlerhafte Interpretation der Testabdeckung. Quelle: Autor .....	160
Abbildung 62: Visualisierter Entwicklungsprozess. Quelle: Autor .....	162
Abbildung 63: Benutzerhandbuch - Menüleiste. Quelle: Autor .....	167
Abbildung 64: Benutzerhandbuch - Studienplan erstellen. Quelle: Autor .....	167
Abbildung 65: Benutzerhandbuch - "Deine Auswahl". Quelle: Autor .....	168
Abbildung 66: Benutzerhandbuch - "Modul hinzufügen". Quelle: Autor .....	168
Abbildung 67: Benutzerhandbuch- Studienplan "Modulliste". Quelle: Autor.....	169
Abbildung 68: Benutzerhandbuch - "Visualisierung". Quelle: Autor.....	169

## 12.4 Tabellenverzeichnis

Tabelle 1: Glossarverzeichnis. Quelle: Autor .....	6
Tabelle 2: Projektziele. Quelle: Autor .....	11
Tabelle 3: Grobanforderungen, Quelle: Autor .....	12
Tabelle 4: Abgrenzungen. Quelle: Autor .....	13
Tabelle 5: Projektressourcen - Personalressourcen. Quelle: Autor .....	15
Tabelle 6: Projektmitglieder, Quelle: Autor .....	16
Tabelle 7: Rollenverteilung, Quelle: Autor .....	17
Tabelle 8: Projektrisiken, Quelle: Autor .....	19
Tabelle 9: Projektrisiken: Quelle: Autor .....	20
Tabelle 10: Planungsstrategie. Quelle: Autor .....	23
Tabelle 62: Kommunikationsplan, Quelle: Autor.....	24
Tabelle 63: Workshop-Plan, Quelle: Autor.....	25
Tabelle 11: Musskriterien, Quelle: Autor .....	26
Tabelle 12: Wunschkriterien, Quelle: Autor .....	26
Tabelle 13: Präferenzmatrix. Quelle: Autor.....	28
Tabelle 14: Nutzwertanalyse. Quelle: Autor .....	28
Tabelle 15: Fragestellung für die Wahl der Datenbanktechnologie. Quelle: Autor in Anlehnung an (Vettor, 2021) .....	29
Tabelle 16: Arbeitspaket I-1. Quelle: Autor .....	34
Tabelle 17: Arbeitspaket I-2. Quelle: Autor.....	34
Tabelle 18: Arbeitspaket I-3. Quelle: Autor .....	35
Tabelle 19: Arbeitspaket I-4. Quelle: Autor .....	35
Tabelle 20: Arbeitspaket I-5. Quelle: Autor .....	36
Tabelle 21: Arbeitspaket I-6. Quelle: Autor .....	36
Tabelle 22: Arbeitspaket V-1. Quelle: Autor .....	37
Tabelle 23: Arbeitspaket V-2. Quelle: Autor .....	37
Tabelle 24: Arbeitspaket V-3. Quelle: Autor .....	38
Tabelle 25: Arbeitspaket V-4.1. Quelle: Autor .....	38
Tabelle 26: Arbeitspaket V-4.2. Quelle: Autor .....	39
Tabelle 27: Arbeitspaket V-5.1. Quelle: Autor .....	39
Tabelle 28: Arbeitspaket V-5.2. Quelle: Autor .....	40
Tabelle 29: Arbeitspaket V-5.3. Quelle: Autor .....	40
Tabelle 30: Arbeitspaket V-6. Quelle: Autor .....	40
Tabelle 31: Arbeitspaket K-1. Quelle: Autor .....	41
Tabelle 32: Arbeitspaket K-2. Quelle: Autor .....	41

Tabelle 33: Arbeitspaket K-3. Quelle: Autor .....	42
Tabelle 34: Arbeitspaket K-4.1. Quelle: Autor .....	42
Tabelle 35: Arbeitspaket K-4.2. Quelle: Autor .....	43
Tabelle 36: Arbeitspaket K-5. Quelle: Autor .....	43
Tabelle 37: Arbeitspaket K-6. Quelle: Autor .....	44
Tabelle 38: Arbeitspaket K-7. Quelle: Autor .....	44
Tabelle 39: Arbeitspaket K-8. Quelle: Autor .....	45
Tabelle 40: Arbeitspaket K-9. Quelle: Autor .....	45
Tabelle 41: Arbeitspaket K-10. Quelle: Autor .....	46
Tabelle 42: Arbeitspaket K-11. Quelle: Autor .....	46
Tabelle 43: Arbeitspaket K-12. Quelle: Autor .....	47
Tabelle 44: Arbeitspaket R-1. Quelle: Autor .....	48
Tabelle 45: Arbeitspaket R-2. Quelle: Autor .....	48
Tabelle 46: Arbeitspaket R-3. Quelle: Autor .....	49
Tabelle 47: Arbeitspaket R-4. Quelle: Autor .....	49
Tabelle 48: Arbeitspaket R-5. Quelle: Autor .....	50
Tabelle 49: Arbeitspaket R-6. Quelle: Autor .....	50
Tabelle 50: Arbeitspaket R-7. Quelle: Autor .....	51
Tabelle 51: Arbeitspaket R-8. Quelle: Autor .....	51
Tabelle 52: Arbeitspaket R-9. Quelle: Autor .....	52
Tabelle 53: Arbeitspaket A-1. Quelle: Autor .....	53
Tabelle 54: Arbeitspaket A-2. Quelle: Autor .....	53
Tabelle 55: Arbeitspaket A-3. Quelle: Autor .....	54
Tabelle 56: Arbeitspaket A-4. Quelle: Autor .....	54
Tabelle 57: Arbeitspaket A-5. Quelle: Autor .....	54
Tabelle 58: Arbeitspaket A-6. Quelle: Autor .....	55
Tabelle 59: Arbeitspaket A-7. Quelle: Autor .....	55
Tabelle 60: Meilensteile. Quelle: Autor .....	56
Tabelle 61: Gantt Auflistung. Quelle: Autor .....	57
Tabelle 64: Massnahmen bei Abweichungen. Quelle: Autor .....	60
Tabelle 65: Kontrollprotokoll der Arbeitspakete. Quelle: Autor .....	61
Tabelle 66: Kontrollprotokoll der User Stories. Quelle: Autor .....	62
Tabelle 67: URLs der Systemlösung (UI). Quelle: Autor .....	67
Tabelle 68: Elementbeschreibung und Typ vom Mockup Home. Quelle: Autor .....	69
Tabelle 69: Elementbeschreibung und Typ vom Mockup Neuen Studienplan erstellen. Quelle: Autor .....	71
Tabelle 71: Elementbeschreibung und Typ vom Mockup Studienplan laden. Quelle: Autor .....	73
Tabelle 72: Elementbeschreibung und Typ vom Mockup Module hinzufügen. Quelle: Autor .....	78
Tabelle 73: Elementbeschreibung und Typ vom Mockup Impressum. Quelle: Autor .....	79
Tabelle 74: Erklärung zu den Prioritäten. Quelle: Autor .....	80
Tabelle 75: User Story #1 - Studienplan erstellen. Quelle: Autor .....	81
Tabelle 76: User Story #2 - Studienplanung anzeigen. Quelle: Autor .....	82
Tabelle 77: User Story #3 - Jahr wählen. Quelle: Autor .....	83
Tabelle 78: User Story #4 - Vertiefungsrichtung wählen. Quelle: Autor .....	83
Tabelle 79: User Story #4 - Nachbesserung. Quelle: Autor .....	84
Tabelle 80: User Story #5 - Semester und Semestertyp wählen. Quelle: Autor .....	84
Tabelle 81: User Story #6 - Modul wählen. Quelle: Autor .....	85
Tabelle 82: User Story #7 - Auswahl in Planung übertragen. Quelle: Autor .....	86
Tabelle 83: User Story #8 - Modul entfernen. Quelle: Autor .....	86
Tabelle 84: User Story #9 - Relevanzpunkte sichten. Quelle: Autor .....	87
Tabelle 85: User Story #10 - ECTS-Punkte sichten. Quelle: Autor .....	87
Tabelle 86: User Story #11 - Relevanzpunkte pro Vertiefungsrichtung sichten. Quelle: Autor .....	88
Tabelle 87: User Story #12 - Semesterplan sichten. Quelle: Autor .....	88
Tabelle 88: User Story #13 - Studienplan speichern. Quelle: Autor .....	89
Tabelle 89: User Story #14 - Studienplan mittels Namen laden. Quelle: Autor .....	89
Tabelle 90: User Story #15 - Studienplan visualisieren. Quelle: Autor .....	90

Tabelle 91: User Story #16 - Studienplan löschen. Quelle: Autor .....	91
Tabelle 92: Subsysteme Ebene 1. Quelle: Autor .....	94
Tabelle 93: Subsysteme Ebene 2 - Frontend. Quelle: Autor.....	95
Tabelle 94: Schnittstellen Ebene 2 - Frontend. Quelle: Autor.....	95
Tabelle 95: Subsysteme Ebene 2 - Backend. Quelle: Autor .....	96
Tabelle 96: Schnittstellen Ebene 2 - Backend. Quelle: Autor .....	97
Tabelle 97: Schnittstellen Ebene 2 - Datenbank. Quelle: Autor .....	98
Tabelle 98: Tabellenattribute Tabelle studyplan. Quelle: Autor .....	100
Tabelle 99: Tabellenattribute Tabelle studyplan_module_plans. Quelle: Autor .....	100
Tabelle 100: Tabellenattribute Tabelle module_plan. Quelle: Autor .....	100
Tabelle 101: Tabellenattribute Tabelle module. Quelle: Autor.....	100
Tabelle 102: Tabellenattribute Tabelle module_relevance. Quelle: Autor .....	100
Tabelle 103: Tabellenattribute Tabelle relevance. Quelle: Autor .....	101
Tabelle 104: Tabellenattribute Tabelle module_semester_plan. Quelle: Autor .....	101
Tabelle 105: Tabellenattribute Tabelle semester_plan. Quelle: Autor.....	101
Tabelle 106: Spezifikation Klasse BackendApplication. Quelle: Autor .....	102
Tabelle 107: Spezifikation Klasse Webconfig. Quelle: Autor .....	102
Tabelle 108: Spezifikation Klasse Specialisation. Quelle: Autor .....	104
Tabelle 109: Spezifikation Klasse ModuleType. Quelle: Autor .....	104
Tabelle 110: Spezifikation Klasse StudyProgram. Quelle: Autor .....	104
Tabelle 111: Spezifikation Klasse SemesterType. Quelle: Autor .....	104
Tabelle 112: Spezifikation Klasse Module. Quelle: Autor.....	105
Tabelle 113: Spezifikation Klasse StudyPlan. Quelle: Autor.....	105
Tabelle 114: Spezifikation Klasse ModulePlan. Quelle: Autor .....	105
Tabelle 115: Spezifikation Klasse SemesterPlan. Quelle: Autor .....	106
Tabelle 116: Spezifikation Klasse Relevance. Quelle: Autor .....	106
Tabelle 117: Spezifikation Klasse ModuleService. Quelle: Autor.....	108
Tabelle 118: Spezifikation Methode getModulesByFilter. Quelle: Autor .....	108
Tabelle 119: Spezifikation Methode getModuleType. Quelle: Autor .....	108
Tabelle 120: Spezifikation Klasse RandomIdProvider. Quelle: Autor.....	108
Tabelle 121: Spezifikation Klasse SpecialisationService. Quelle: Autor .....	108
Tabelle 122: Spezifikation Methode getAllSpecialisations. Quelle: Autor .....	108
Tabelle 123: Spezifikation Klasse StudyPlanService. Quelle: Autor.....	109
Tabelle 124: Spezifikation Methode addNewStudyPlan. Quelle: Autor.....	109
Tabelle 125: Spezifikation Methode getStudyPlanId. Quelle: Autor .....	109
Tabelle 126: Spezifikation Methode getStudyPlanByName. Quelle: Autor .....	109
Tabelle 127: Spezifikation Methode addModuleToStudyPlan. Quelle: Autor .....	109
Tabelle 128: Spezifikation Methode checkIfStudyPlanContainsModule. Quelle: Autor .....	110
Tabelle 129: Spezifikation Methode removeModuleFromStudyPlan. Quelle: Autor.....	110
Tabelle 130: Spezifikation Methode getECTS. Quelle: Autor .....	110
Tabelle 131: Spezifikation Methode getTotalRelevanceBySpecialisation. Quelle: Autor .....	110
Tabelle 132: Spezifikation Klasse StudyPlanService. Quelle: Autor.....	110
Tabelle 133: Spezifikation Methode getAll. Quelle: Autor .....	110
Tabelle 134: Spring Controller HTTP-Methoden und Annotation. Quelle: Autor .....	111
Tabelle 135: Spezifikation Klasse ModuleController. Quelle: Autor.....	112
Tabelle 136: Spezifikation Methode getModulesByFilter. Quelle: Autor .....	112
Tabelle 137: Spezifikation Klasse StudyProgramController. Quelle: Autor .....	112
Tabelle 138: Spezifikation Methode getStudyPrograms. Quelle: Autor .....	112
Tabelle 139: Spezifikation Klasse SpecialisationController. Quelle: Autor .....	112
Tabelle 140: Spezifikation Methode getSpecialisations. Quelle: Autor .....	112
Tabelle 141: Spezifikation Klasse ValidatorController. Quelle: Autor .....	113
Tabelle 142: Spezifikation Methode validateStudyPlanByName. Quelle: Autor .....	113
Tabelle 143: Spezifikation Methode canAddModuleToStudyPlan. Quelle: Autor .....	113
Tabelle 144: Spezifikation Klasse StudyPlanController. Quelle: Autor.....	113
Tabelle 145: Spezifikation Methode addNewStudyPlan. Quelle: Autor.....	113
Tabelle 146: Spezifikation Methode getStudyPlan. Quelle: Autor.....	113

Tabelle 147: Spezifikation Methode addModuleToStudyPlan. Quelle: Autor .....	114
Tabelle 148: Spezifikation Methode removeModuleFromStudyPlan. Quelle: Autor .....	114
Tabelle 149: Spezifikation Methode deleteStudyPlan. Quelle: Autor .....	114
Tabelle 150: Spezifikation Methode getTotalECTS. Quelle: Autor .....	114
Tabelle 151: Spezifikation Methode getTotalRelevanceBySpecialisation. Quelle: Autor .....	114
Tabelle 152: Spezifikation Klasse AddModuleToStudyPlanRequest. Quelle: Autor .....	115
Tabelle 153: Spezifikation Klasse NewStudyPlanRequest. Quelle: Autor .....	115
Tabelle 154: Spezifikation Klasse FailureText. Quelle: Autor .....	116
Tabelle 155: Spezifikation Exception-Klassen. Quelle: Autor .....	116
Tabelle 156: API - ValidatorController. Quelle: Autor .....	117
Tabelle 157: API - StudyPlanController. Quelle: Autor .....	118
Tabelle 158: API - StudyProgramController. Quelle: Autor .....	118
Tabelle 159: API - StudyPlanController. Quelle: Autor .....	119
Tabelle 160: API - StudyPlanController. Quelle: Autor .....	120
Tabelle 161: API - SpecialisationController. Quelle: Autor .....	120
Tabelle 162: API - SpecialisationController. Quelle: Autor .....	121
Tabelle 163: API - StudyPlanController. Quelle: Autor .....	122
Tabelle 164: API - ValidatorController. Quelle: Autor .....	122
Tabelle 165: API - StudyPlanController. Quelle: Autor .....	123
Tabelle 166: API - StudyPlanController. Quelle: Autor .....	123
Tabelle 167: API - StudyPlanController. Quelle: Autor .....	124
Tabelle 168: API - StudyPlanController. Quelle: Autor .....	124
Tabelle 169: API - StudyPlanController. Quelle: Autor .....	124
Tabelle 170: Hauptübersicht der Komponenten. Quelle: Autor .....	132
Tabelle 171: SharedModule. Quelle: Autor .....	133
Tabelle 172: Übergeordnete Testziele. Quelle: Autor .....	135
Tabelle 173: Tester und Vorkenntnisse. Quelle: Autor .....	137
Tabelle 174: Mängelklassen. Quelle: Autor .....	137
Tabelle 175: Testfall TC-001. Quelle: Autor .....	139
Tabelle 176: Testfall TC-002. Quelle: Autor .....	139
Tabelle 177: Testfall TC-003. Quelle: Autor .....	140
Tabelle 178: Testfall TS-001. Quelle: Autor .....	140
Tabelle 179: Testfall TS-002. Quelle: Autor .....	141
Tabelle 180: Testfall TS-003. Quelle: Autor .....	141
Tabelle 181: Testfall TS-004. Quelle: Autor .....	142
Tabelle 182: Testfall TR-001. Quelle: Autor .....	142
Tabelle 183: Testfall TR-002. Quelle: Autor .....	143
Tabelle 184: Testfall TC-004. Quelle: Autor .....	143
Tabelle 185: Testfall TC-005. Quelle: Autor .....	144
Tabelle 186: Testfall TC-006. Quelle: Autor .....	144
Tabelle 187: Testfall TS-005. Quelle: Autor .....	145
Tabelle 188: Testfall TS-006. Quelle: Autor .....	145
Tabelle 189: Testfall TS-007. Quelle: Autor .....	146
Tabelle 190: Testfall TS-008. Quelle: Autor .....	146
Tabelle 191: Testfall TR-003. Quelle: Autor .....	147
Tabelle 192: Testfall TR-004. Quelle: Autor .....	147
Tabelle 193: Testfall TR-005. Quelle: Autor .....	148
Tabelle 194: Testfall TR-006. Quelle: Autor .....	148
Tabelle 195: Testfall TC-007. Quelle: Autor .....	149
Tabelle 196: Testfall TC-008. Quelle: Autor .....	149
Tabelle 197: Testfall TS-009. Quelle: Autor .....	150
Tabelle 198: Testfall TS-010. Quelle: Autor .....	150
Tabelle 199: Testfall TR-007. Quelle: Autor .....	151
Tabelle 200: Testfall TC-009. Quelle: Autor .....	151
Tabelle 201: Testfall TS-011. Quelle: Autor .....	152
Tabelle 202: Testfall TC-010. Quelle: Autor .....	152

Tabelle 203: Testfall TS-012. Quelle: Autor.....	153
Tabelle 204: Testfall TC-011. Quelle: Autor .....	153
Tabelle 205: Testfall TS-013. Quelle: Autor.....	154
Tabelle 206: Testfall TR-008. Quelle: Autor .....	154
Tabelle 207: Testfall TC-012. Quelle: Autor .....	155
Tabelle 208: Testfall TC-013. Quelle: Autor .....	155
Tabelle 209: Testfall TS-014. Quelle: Autor.....	156
Tabelle 210: Testfall TS-015 Quelle: Autor.....	156
Tabelle 211: Testplan. Quelle: Autor .....	157
Tabelle 212: Testprotokoll. Quelle: Autor .....	157
Tabelle 213: Testreport - Übersicht aller Testfälle. Quelle: Autor .....	159
Tabelle 214: Testreport - Details nach Package. Quelle: Autor .....	159
Tabelle 215: Testprotokoll- Details nach Klasse. Quelle: Autor .....	160
Tabelle 216: Technologien und Tools. Quelle: Autor .....	163
Tabelle 217: Lieferobjekte - Verzeichnisstruktur. Quelle: Autor.....	165



## 13 Eigenständigkeitserklärung

"Ich erkläre hiermit,

- dass ich die vorliegende Arbeit selbstständig, ohne fremde Hilfe und ohne Verwendung anderer als der angegebenen Hilfsmittel verfasst habe;
- dass ich sämtliche verwendeten Quellen erwähnt und zitiert habe;
- dass ich sämtliche immateriellen Rechte an von mir allfällig verwendeten Materialien wie Bilder oder Grafiken erworben habe, diese frei zur Nutzung verfügbar sind, oder dass diese Materialien von mir selbst erstellt wurden;
- dass das Thema, die Arbeit oder Teile davon nicht Gegenstand eines Leistungsnachweises einer anderen Veranstaltung oder Kurses waren, sofern dies nicht ausdrücklich mit dem Referenten /der Referentin im Voraus vereinbart wurde und in der Arbeit ausgewiesen wird;
- dass ich mir bewusst bin, dass meine Arbeit elektronisch auf Plagiate überprüft werden kann und ich hiermit der Fernfachhochschule Bern, laut Prüfungsordnung, das Urheberrecht soweit einräume, wie es für die Verwaltungshandlungen notwendig ist;
- dass ich mir bewusst bin, dass ein Verstoss gegen diese Eigenständigkeitserklärung sowie insbesondere die Inanspruchnahme eines Ghostwriter-Service verfolgt und dass daraus disziplinarische wie auch strafrechtliche Folgen resultieren können, welche zum Ausschluss von der Schule resp. zur Titelaberkennung führen können.“

Bern, den 23. Dezember 2021

Christoph Grossmann

Marco Ris

Peter Spielhofer