

Team Green

Softwaresystem zur Verwaltung eines Zulassungsverfahrens

WS 20/21 Softwaretechnik Hausaufgabe 1

Teammitglieder	Tätigkeiten
Aleksandar Garkov	Vorgehensweise (siehe Anhang), User Stories ("Bewerber*innen")
Christopher Goldberg	Datenlieferung für Meilensteine und Prozessplan, Qualitätssicherung für gesamtes Dokument
Christian Schmitz	Prozessplan, Users Stories (Teile)
Steffen Weißhaar	Zeitplan, Meilensteine, User Stories ("Zulassungsausschuss")
Maximilian Müller	Risikobewertung, User Stories ("Unterstützer*innen"), Prozessdiagramm
Felix Zornow	Gantt Diagramm, User Stories ("Bewerber*innen")

Zeitplan

Eine sorgfältige Planung ist entscheidend für die Software- und Produktentwicklung. In dieser Phase werden der Projektumfang, die Ressourcenverfügbarkeit und die Zeitplanung thematisiert. Um den Fortschritt des Projektes zu verfolgen, werden drei Meilensteine festgesetzt: der erste Meilenstein umfasst die Planungsphase, der zweite Meilenstein die Umsetzung und der dritte Meilenstein die Reflexion.

- 1. Meilenstein (9.11. bis 16.11.): Planungsdokumente**
 - a. Erstellen der User Stories
 - b. Schätzen der User Stories
 - c. Priorisieren der User Stories
 - d. Abhängigkeiten bestimmen
 - e. Aufplanen der User Stories
- 2. Meilenstein (17.11. bis 25.11.): Code Artefakte**
 - a. Infrastruktur einrichten (Build Automatisierung + Datenbank)
 - b. Interfaces definieren
 - c. Umsetzen der geplanten User Stories nach Priorisierung
 - d. Fehler/ Bugs dokumentieren
 - e. Qualitätssicherung/ Testing (FindBugs Integration)
 - f. Dokumentation erstellen (Handbuch)
- 3. Meilenstein (26.11. bis 30.11.): Reflexion**
 - a. Was hat gut funktioniert?
 - b. Fehler aufarbeiten
 - c. Gegenüberstellung Plan vs. Umsetzung
 - d. Learnings für das nächste Projekt

Ziel des ersten Meilensteins ist es den Projektumfang mittels User Stories (US) zu erfassen und eine Aufwandschätzung vorzunehmen. User Stories ermöglichen es uns die Anforderungen aus der Perspektive der Nutzergruppen zu betrachten und so ein differenziertes Gesamtbild zu erhalten. Eingeteilt sind die User Stories in die Perspektive der Bewerber*innen, Unterstützer*innen und des Zulassungsausschusses. Aufgrund des kurzen Zeitrahmens ist außerdem davon auszugehen, dass man einen Fokus auf die essentiellen Funktionalitäten setzen muss, daher ist eine Priorisierung nötig. Zur Priorisierung benutzen wir MoSCoW. Nach MoSCoW kann man vier Einschätzungen vornehmen: M - Must, S - Should, C - Could und W - Won't. Um den Aufwand der User Stories zu schätzen, benutzen wir eine virtuelle Form des Planning Pokers. Die Aufwände ergeben sich daher als unabhängige Schätzung der einzelnen Teammitglieder. Wie die einzelnen User Stories bewertet sind, kann den Tabellen A, B und C im Anhang entnommen werden.

Im zweiten Meilenstein findet die Umsetzung der User Stories statt. Die im Planning Poker bestimmten Aufwände geben einen Maßstab und keinen tatsächlichen Zeitaufwand an. Zusammen mit Erfahrungswerten, kann man den geschätzten Aufwand in Punkten einem Zeitaufwand zuordnen. Da wir auf diese Erfahrung noch nicht zurückgreifen können, lässt sich der Zeitaufwand nur grob abschätzen. Für die Planung gehen wir davon aus, dass zwei Personenstunden Arbeit drei Punkten (P) aus der Planning Poker Schätzung entsprechen. Mit zunehmender Erfahrung wird sich das Verhältnis genauer schätzen lassen. Da der

insgesamt geschätzte Aufwand die Ressourcen übersteigt, wird sich an der vorgenommenen Priorisierung orientiert. Die für den zweiten Meilenstein geplanten sind in Tabelle 1 dargestellt (Insgesamt 48 P). Für die Tätigkeiten neben der Umsetzung der User Stories schätzen wir insgesamt 26 Punkte Aufwand. Die Zusammensetzung des Aufwands im Verhältnis zur Kapazität ist Tabelle 1 zu entnehmen. Der Aufwand für den zweiten Meilenstein ergibt sich als Summe aus dem Aufwand zur Umsetzung der User Stories und den Nebentätigkeiten.

Tabelle 1: Aufwand im 2. Meilenstein

User Stories mit (Priorisierung) - Aufwand in Punkten	Tätigkeiten neben der Umsetzung der US - Aufwand in Punkten	Kapazität im 2. Meilenstein
US 1.2 (M) - 13 US 1.3 (M) - 5 US 1.5 (M) - 3 US 2.5 (M) - 8 US 3.2 (M) - 13 US 3.4 (M) - 3 US 1.7 (S) - 1 US 1.8 (S) - 2	Infrastruktur einrichten - 3 Interfaces definieren - 2 Fehler dokumentieren - 2 Qualitätssicherung - 13 Dokumentation - 3	6 h pro Woche und Person 6 Personen $6 * 8 = 48 \text{ h}$ 1 h ~ 1,5 P (Schätzung) 48 h ~ 72 P
= 48 P	= 26 P	Gesamt 74 P

Im Prozessdiagramm (Abbildung 1) und im Gantt Diagramm (Abbildung 2) werden die Abhängigkeiten, die voraussichtliche Abfolge der User Stories und an welchen User Stories parallel gearbeitet werden kann dargestellt. Ziel ist es von Beginn an Tätigkeiten zu parallelisieren. Grundsätzlich lassen sich die Tätigkeiten dem Front- und Backend zuordnen. Aufgaben im Frontend werden hauptsächlich von Felix Z., Alex G. und Max M. übernommen. Das Backend übernehmen Christopher G., Chris S. und Steffen W. Da User Stories in der Regel sowohl das Front- als auch Backend betreffen, werden diese nicht kategorisiert.

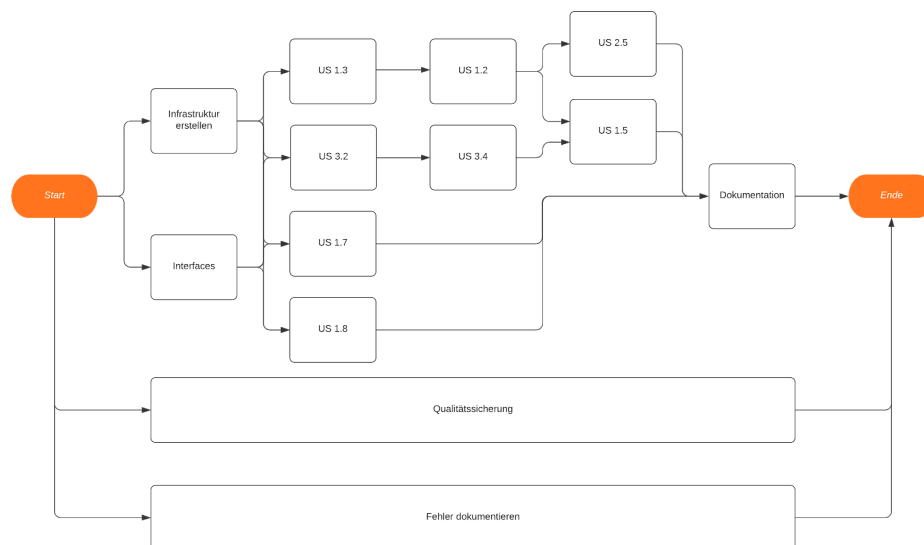


Abbildung 1: Prozessdiagramm

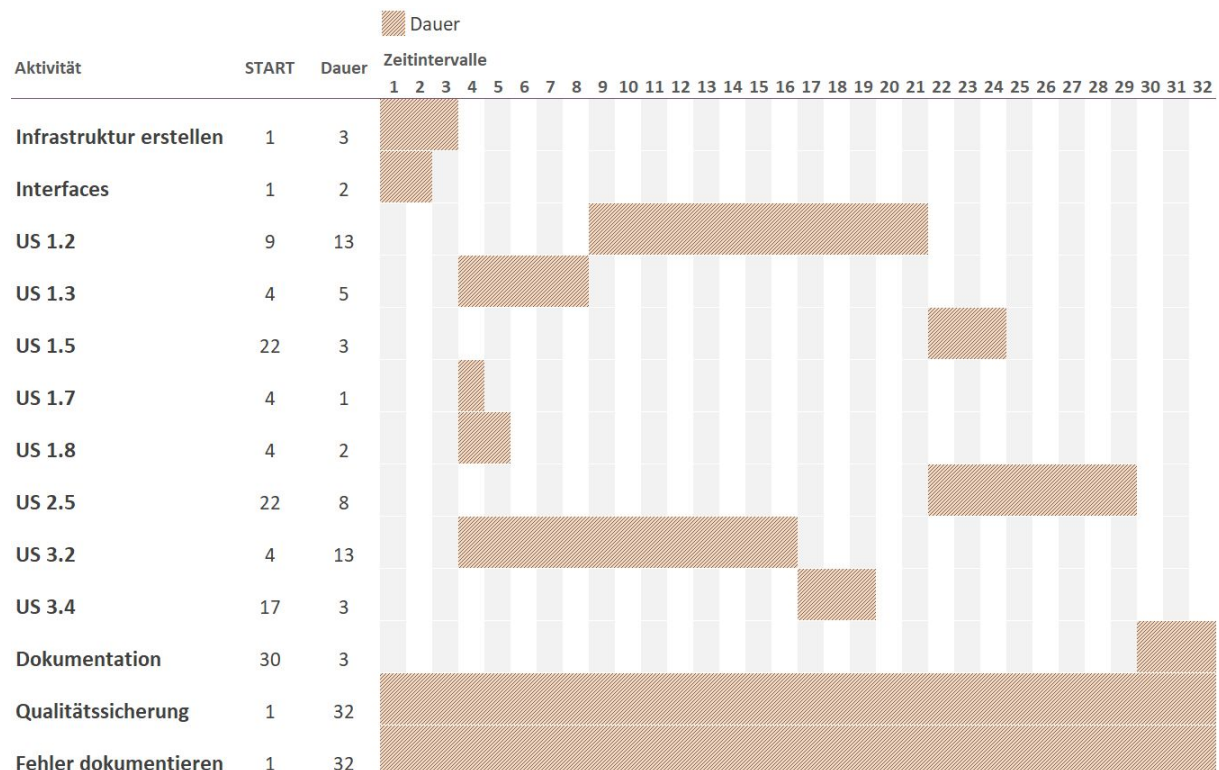


Abbildung 2: Gantt Diagramm

Um einen besseren Überblick über die User Stories zu erhalten, gibt es zu jeder oben beschriebenen Perspektive ein Epic (E), das sich aus kleineren User Stories zusammensetzt (siehe Tabelle 2). Der Aufwand der Epics ergibt sich als Summe der Aufwände der untergeordneten User Stories mit der Priorisierung (M, S, C). Die Priorisierung "W" wird nicht eingerechnet, da die Priorität dieser User Stories zu gering ist.

Tabelle 2: Übersicht der Epics

Nr.	Vorne	Hinten	Aw. in P
E 1	Als Bewerber*in möchte ich meine Bewerbung online über eine Webanwendung (Bewerbungsportal) einreichen können, um am Zulassungsprozess für den Studiengang Informatik teilnehmen zu können. (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert, wenn sich ein Bewerber*innen einloggt, kann er seine Bewerbung online über das Bewerbungsportal einreichen.	31
E 2	Als Unterstützer*in möchte ich Empfehlungsschreiben für Bewerber*innen bequem in einer Webanwendung hochladen können, um diese für einen Studiengang zu empfehlen (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert, wenn sich ein*e Unterstützer*in einloggt, können Empfehlungsschreiben für entsprechende Bewerber*innen hochgeladen werden.	37

E 3	Als Zulassungsausschuss möchte ich über eine Webanwendung alle relevanten Informationen von den Bewerber*innen erfassen, um diese für den Zulassungsprozess auszuwerten. (M)	Angenommen für den Studiengang Informatik haben mehrere Bewerber*innen registriert und ihre Daten sowie Unterlagen eingereicht, wenn sich ein*e Mitarbeiter*in des Zulassungsausschusses einloggt, dann kann er sich einen Überblick verschaffen und die Daten der Bewerber*innen einsehen.	37
-----	--	---	----

Im dritten Meilenstein wird das Vorgehen aus der Umsetzungsphase reflektiert und aufgearbeitet. Es wird betrachtet was gut gelaufen ist und was man das nächste mal besser machen kann. Ziel ist so zu verhindern, dass Fehler wiederholt werden.

Risikobewertung

Damit bei einem Projekt solcher Größe keine unerwarteten Probleme auftreten, die das Voranschreiten oder die Fertigstellung behindern, ist es wichtig sich eine Übersicht über die möglichen Risiken zu verschaffen, abzuwägen wie wahrscheinlich und gravierend diese sind und wie sie sich vermeiden oder zumindest die Eintrittswahrscheinlichkeit verringern lässt. Dafür haben wir uns zu einer tabellarischen Darstellung dieser Risiken entschieden. Gesammelt wurden die folgenden Risiken anhand vorheriger Erfahrungen aus anderen Projekten, bereits in Meetings genannten Bedenken und allgemeinen Risiken, denen man in der Softwareentwicklung ausgesetzt ist.

Die Gewichtung der einzelnen Risiken setzt sich zusammen aus einer geschätzten Wahrscheinlichkeit, dass das entsprechende Ereignis tatsächlich eintritt, sowie eines vermuteten Ausmaßes des Schadens in solch einem Fall.

Um das Risiko möglichst gering zu halten, wurden verschiedene, vor allem präventive, Maßnahmen definiert. So soll die Eintrittswahrscheinlichkeit weiter minimiert werden. Die Umsetzung dieser Maßnahmen ist entscheidend zur Bewältigung unseres Projektes ohne dabei einen allzu großen Schaden erleiden zu müssen.

Tabelle 3: Darstellung der Risiken

Nr.	Risiko	W	A	G	Maßnahmen
1.	Zu wenig Zeit eingeplant	3	3	9	<ul style="list-style-type: none"> • Zeiten mit Puffer bzw. großzügiger planen • Erstmal nur auf wichtigste Funktionen fokussieren und unwichtigere Funktionen erst danach einbauen
2.	Nicht alle Funktionen werden eingebaut	2	3	6	<ul style="list-style-type: none"> • Klare Übersicht behalten über Struktur des Programms, damit keine Funktionen vergessen werden

					<ul style="list-style-type: none"> Keine Zeit mit Überoptimierung verschwenden, es soll nur ein Prototyp werden also muss es nicht perfekt sein
3.	Fähigkeiten der Teammitglieder sind nicht ausreichend	2	2	4	<ul style="list-style-type: none"> Benötigte Fähigkeiten klar definieren Fähigkeiten der Teammitglieder klar benennen Teammitglieder ihren Fähigkeiten entsprechend Aufgaben zuweisen Fehlende Fähigkeiten aneignen
4.	Missverständnisse/ Fehlkommunikation zwischen den Teammitgliedern durch die besondere Situation (Covid-19)	1	2	2	<ul style="list-style-type: none"> Regelmäßige Meetings Klare Kommunikation Channel nutzen, die wir definiert haben (WhatsApp-gruppe, Discord, Google Drive, Github)
5.	Das Endprodukt funktioniert nicht wie erwartet	1	1	1	<ul style="list-style-type: none"> Im vorhinein Struktur des Programms definieren Ausreichend testen Nur Prototyp, Produkt muss nicht perfekt sein

Legende:

W - Wahrscheinlichkeit, dass das Ereignis eintritt. Skala von 1 (niedrig) bis 3 (hoch)

A - Ausmaß des Ereignisses. Skala von 1 (niedrig) bis 3 (hoch)

*G - Gewichtung des Ereignisses. Berechnet aus Wahrscheinlichkeit * Ausmaß*

Prozessplan

Als primärer Kommunikationskanal für das Projekt wurde im initialen Meeting Discord festgelegt. Ähnlich wie Slack können Textinformationen sowie Dateien in themenbasierten Kanälen ausgetauscht und gesammelt werden. Discord bietet gegenüber Slack unter anderem den Vorteil, dass auch reine Sprachkanäle angelegt werden können. Zudem ist der Großteil der Gruppenmitglieder mit Discord vertraut. Da es sich um ein Hausaufgaben-Projekt handelt, ergaben sich keine gravierenden Datenschutzbedenken für die Nutzung eines Cloud-Dienstes. Nachfragen und andere kleinere bzw. "nicht offizielle" Abstimmungen finden auch über eine Whatsapp-Gruppe statt.

Als Jour Fixe wurde in der ersten Sitzung Montag 17:30 Uhr, nach der Vorlesung, festgelegt. Sofern weitere Sitzungen zur Abstimmung notwendig sind, was insbesondere zu Beginn zur Planung des Projektes erforderlich war, wird jeweils am Ende einer Sitzung ein weiterer Termin avisiert. Sofern bereits bekannt, wird die Agenda für die nächste Teamsitzung bereits angelegt. Weitere Tagesordnungspunkte können jederzeit zwischen den Sitzungen hinzugefügt werden. Zusätzlich wurde festgelegt, dass Sitzungen nur in Ausnahmefällen länger als zwei Stunden dauern sollten.

Dokumente werden über Google Docs erstellt, verwaltet und ausgetauscht. Kollaboratives arbeiten ist somit plattformunabhängig möglich. Zudem sind alle Teammitglieder mit Google Docs vertraut.

Die Koordination der Entwicklung des Programmiercodes wird größtenteils über die Projektmanagement Features von Github erfolgen, insbesondere über Issues und Labels (Board). Änderungen am Code erfolgen planmäßig über individuelle Feature-Branche und entsprechende Merge-Requests, die von anderen Teammitgliedern vor dem Merge jeweils gesichtet und bestätigt werden müssen. Die IDE ist jedem freigestellt selber zu wählen. Innerhalb des Teams wird VS Code, IntelliJ und Eclipse genutzt.

Insgesamt wird versucht, die jeweiligen Stärken und Interessen der Teammitglieder zu berücksichtigen. Die Aufgabenverteilung erfolgt jeweils innerhalb der Teamsitzungen. Die Ergebnisse werden gemeinsam in den Sitzungen besprochen. Individuelle Abstimmungen sind jedoch jederzeit möglich und gewünscht, insbesondere bei Aufgaben, die mehrere Teammitglieder gemeinschaftlich bearbeiten. Die Koordination erfolgt innerhalb dieser Taskforces individuell.

Die Sitzungsprotokolle werden jeweils von einem anderen Teammitglied geführt und sind jederzeit für alle Teammitglieder im Google Docs Projektverzeichnis einsehbar. Die Besprechung wird gemeinschaftlich mit allen Teammitgliedern vorangetrieben, die Moderation (Koordination) erfolgt durch den Protokollführer. Die Überwachung obliegt jeweils einem anderen Teammitglied. Die Prüfung findet gemeinschaftlich statt. Jedes Teammitglied übernimmt jeweils einen definierten Teil der zu prüfenden Dokumente respektive Aufgaben.

Anhang

User Stories

Tabelle A: User Stories aus Sicht der Bewerber*innen

Nr.	Vorne	Hinten	Aw
E 1	Als Bewerber*in möchte ich meine Bewerbung online über eine Webanwendung (Bewerbungsportal) einreichen können, um am Zulassungsprozess für den Studiengang Informatik teilnehmen zu können. (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert, wenn sich ein*e Bewerber*in einloggt, kann er*sie seine Bewerbung online über das Bewerbungsportal einreichen.	31
1.2	Als Bewerber*in möchte ich, dass meine Daten und Dateien in einer Datenbank gespeichert werden, damit diese später vom Zulassungsausschuss geprüft werden können. (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert, wenn ein*e Bewerber*in am Ende des Prozesses seine Daten und Dateien bestätigt, dann werden sie in der Datenbank abgespeichert.	13
1.3	Als Bewerber*in möchte ich ein Formular (im Bewerbungsportal), um relevante Informationen über meine Bewerbung geben zu können. (M)	Angenommen ein*e Bewerber*in hat sich im Bewerbungsportal eingeloggt, wenn er relevante Informationen über seine Bewerbung geben möchte, dann könnte er ein Formular ausfüllen.	5
1.4	Als Bewerber*in möchte ich einen Button, um Empfehlungsschreiben von Dritten hochladen zu können. (W)	Angenommen ein*e Unterstützer*in kann nicht gewährt werden, wenn ein Bewerber*innen ein Empfehlungsschreiben von Dritten hochladen will, dann kann er es mit einem Button abgeben.	5
1.5	Als Bewerber*in möchte ich, nachdem die Bewerbung bearbeitet wurde, der Oberfläche entnehmen können, ob ich zugelassen bin oder nicht. (M)	Angenommen die Ergebnisse sollten schon veröffentlicht werden, wenn ein*e Bewerber*in das Bewerbungsportal öffnet, dann sieht er*sie, ob er zugelassen ist oder nicht.	3
1.6	Als Bewerber*in möchte ich den Bearbeitungsstatus verfolgen, um mehr Kontrolle darüber zu haben. (S)	Angenommen ein*e Bewerber*in hat seine*ihre Bewerbung abgeschickt, wenn er*sie wissen möchte wie weit die Zulassung mit der Bearbeitung der Bewerbung ist, dann kann er*sie nachgucken wie weit seine*ihre Bewerbung bearbeitet wurde	3

1.7	Als Bewerber*in möchte ich eine Kontakt-Zeile haben, damit ich bei einem Problem mit der Bewerbung jemanden anrufen oder per E-Mail kontaktieren kann. (S)	Angenommen, ein Fehler ist bei der Bewerbung aufgetreten, wenn der*die Bewerber*in dann noch Fragen zum weiteren Verlauf hat, dann kann er*sie eine E-mail schreiben oder einen Anruf machen um sich zu informieren	1
1.8	Als Bewerber*in möchte ich ein FAQ-Feature haben, um die häufigsten Fragen anschauen zu können. (S)	Angenommen der*die Bewerber*in hat noch Fragen, wenn er*sie diese beantwortet haben will, dann kann er*sie zunächst im FAQ-Feature nachschauen ob seine*ihre Frage dort beantwortet ist	2
1.9	Als Bewerber*in möchte ich ein "Über uns"-Feature, damit ich etwas mehr über die Organisation lesen kann, der ich meine Daten anvertraue. (C) (läuft über die Uni-Webseite)	Angenommen der*die Bewerber*in will mehr über die Institution wissen, wenn er*sie recherchieren will, dann wird er*sie auf die Uni-Website weitergeleitet	2
1.10	Als Bewerber*in möchte ich allgemeine Information über das Bewerben sehen, damit ich sicher sein kann, dass ich keine Fristen verpasse. (S)	Angenommen es stehen bestimmte Termine oder Fristen an, wenn der*die Bewerber*in diese einhalten muss, dann werden ihm*ihr allgemeine Informationen darüber direkt beim login angezeigt	2

Tabelle B: Aus Sicht der Unterstützer*innen

Nr.	Vorne	Hinten	Aw.
E 2	Als Unterstützer*in möchte ich Empfehlungsschreiben für Bewerber*innen bequem in einer Web anwendung hochladen können, um diese für einen Studiengang zu empfehlen (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert, wenn sich ein*e Unterstützer*in einloggt, können Empfehlungsschreiben für entsprechende Bewerber*innen hochgeladen werden.	37
2.2	Als Unterstützer*in möchte ich ein Empfehlungsschreiben nur einmal hochladen müssen, um mir unnötige Arbeit zu sparen. (W) (Anm.: System nur für einen Studiengang geplant, nicht für mehrere; Abstimmung mit Auftraggeber nötig)	Angenommen ein*e Bewerber*in bewirbt sich für mehrere Studiengänge, wenn ein*e Unterstützer*in ein Empfehlungsschreiben hochladen will dann muss dies nur einmal gemacht werden.	8
2.3	Als Unterstützer*in möchte ich wissen, ob meine Dokumente erfolgreich hochgeladen wurden, um Fehler zu vermeiden. (S)	Angenommen ein*e Unterstützer*in möchte ein Dokument hochladen, wenn es hochgeladen ist, dann kann man erkennen ob es erfolgreich hochgeladen wurde.	3

2.4	Als Unterstützer*in möchte ich einsehen können, welche Empfehlungen ich gegeben habe, um niemanden zu vergessen oder doppelt zu empfehlen. (C)	Angenommen ein*e Unterstützer*in hat bereits eine oder mehrere Empfehlungen hochgeladen, wenn er*sie sich einloggt, dann gibt es eine Übersicht aller Empfehlungen der*des Unterstützer*in	8
2.5	Als Unterstützer*in möchte ich einfach den*die richtige*n Bewerber*in finden können, um die richtige Empfehlung für den*die richtige*n Bewerber*in einreichen zu können. (M)	Angenommen ein*e Unterstützer*in möchte eine Empfehlung geben, wenn er*sie sich auf der Plattform befindet, dann kann man den*die richtige*n Bewerber*in z.B. mithilfe einer ID finden.	8
2.6	Als Unterstützer*in möchte ich nachvollziehen können in welchem Prozessschritt die Bewerber*innen sind, um einen strukturierten Überblick zu behalten (abgegeben, in bearbeitung, angenommen/abgelehnt) (DSGVO?). (W)	Angenommen ein*e Unterstützer*in hat sich in das Zulassungsportal eingeloggt, wenn er*sie auf die Ansicht "Alle Bewerber" wechselt, dann werden ihm*ihr die Status der Bewerber*innen in einer Übersicht angezeigt.	3
2.7	Als Unterstützer*in möchte ich Empfehlungen, die ich gegeben habe, filtern können, um mir einen besseren Überblick verschaffen zu können. (C)	Angenommen ein*e Unterstützer*in hat Empfehlungen hochgeladen, wenn die Übersicht aller Bewerber*innen geladen ist kann diese nach Kriterien gefiltert werden.	5
2.8	Als Unterstützer*in möchte ich einen Account anlegen können, um meine Daten zu speichern.(S)	Angenommen ein*e Unterstützer*in möchte Empfehlungen abgeben, dann muss sich ein*e Unterstützer*in vorher einen Account anlegen, um abgegebene Empfehlungen einfach administrieren zu können.	13

Tabelle C: User Stories aus Sicht des Zulassungsausschusses

Nr.	Vorne	Hinten	Aw.
E 3	Als Zulassungsausschuss möchte ich über eine Webanwendung alle relevanten Informationen von den Bewerber*innen erfassen, um diese für den Zulassungsprozess auszuwerten. (M)	Angenommen für den Studiengang Informatik haben sich mehrere Bewerber*innen registriert und ihre Daten sowie Unterlagen eingereicht, wenn sich ein*e Mitarbeiter*in des Zulassungsausschusses einloggt, dann kann er*sie sich einen Überblick verschaffen und die Daten der Bewerber*innen einsehen.	37

3.2	Als Zulassungsausschuss möchte ich, dass sich die Bewerber*innen selbstständig registrieren können, um die Informationen einer Person zuordnen zu können. (M)	Angenommen es beginnt die Bewerbungsphase für den Studiengang Informatik, dann können Bewerber*innen, die sich auf den Studiengang Informatik bewerben möchten und sich auf der Startseite des Bewerbungsportals wiederfinden, sich selbstständig mit einer E-Mail registrieren und sich zu einem späteren Zeitpunkt erneut einloggen.	13
3.3	Als Zulassungsausschuss möchte ich mir die Bewerber*innen in einer Übersicht anzeigen lassen, um mir einen Überblick zu verschaffen. (S)	Angenommen ein*e Mitarbeiter*in hat sich in das Zulassungsportal eingeloggt, wenn er*sie auf die Ansicht "Alle Bewerber" wechselt, dann werden ihm*ihr die Bewerber*innen in einer Übersicht angezeigt.	8
3.4	Als Zulassungsausschuss möchte ich den Status eines Bewerbers oder einer Bewerberin über die Webanwendung ändern können, um den*die Bewerber*in über das Ergebnis des Zulassungsausschusses zu informieren und den Bewerbungsprozess abzuschließen. (M)	Angenommen die Daten der Bewerber*innen wurden ausgewertet, wenn die Entscheidung für eine*n Bewerber*in getroffen wurde, dann kann ein*e Mitarbeiter*innen den aktuellen Status des*der Bewerbers*Bewerberin über die Webanwendung ändern.	3
3.5	Als Zulassungsausschuss möchte ich, dass die Nutzer*innen über die rechtlichen Rahmenbedingungen informiert werden, um die DSGVO sowie weitere rechtliche Vorgaben zu erfüllen. (W) (Anm.: Es handelt sich ein nachgelagertes System, die Nutzer*innen sind bereits informiert.)	Angenommen ein*e Bewerber*in registriert sich, dann muss er*sie über seine*ihre Rechte informiert werden.	3
3.6	Als Zulassungsausschuss möchte ich FAQ-Fragen selber administrieren können, um den Aufwand bei Änderungen zu verringern. (W)	Angenommen, das Mitglied des Zulassungsausschusses kann sich einloggen, dann muss es die FAQ-Fragen verwalten können.	20
3.7	Als Zulassungsausschuss möchte ich Informationen über einzelne Bewerber*innen abrufen können, um mir ein genaues Bild zu verschaffen. (S)	Angenommen, das Mitglied des Zulassungsausschusses kann sich einloggen, dann muss es die Informationen über alle Bewerber*innen abrufen können und auch auf die Informationen einzelner Bewerber*innen Zugreifen	8

		können.	
3.8	Als Zulassungsausschuss möchte ich die Übersicht der Bewerber*innen, filtern können, um mir einen besseren Überblick zu verschaffen. (C)	Angenommen, das Mitglied des Zulassungsausschusses hat die Informationen über die Bewerber*innen abgerufen, dann muss es diese filtern können.	5

Vorgehensweise

Schritt 1: Planning (Planen)

Eine sorgfältige Planung ist die erste und eine der Kernphasen der Software- oder Produktentwicklung, da sie die Festlegung des Projektumfangs, die Festlegung von Mitteln und Wegen für das neue System zur Erreichung der strategischen Geschäftsziele, der Ressourcenverfügbarkeit, der Kostenprobleme und des Zeitrahmens und Bestimmen von Lösungen voraussetzt.

Schritt 2: Analysis (Analyse)

Die Machbarkeitsanalyse zeigt alle technischen und wirtschaftlichen Aspekte an, die sich auf den Entwicklungsprozess der Anwendung auswirken: Zeit, Ressourcen und Aufgaben sowie Schätzungen der Beteiligung der Teammitglieder helfen bei der Berechnung des ROI (Return Of Investment) und der Bestimmung der Projektkosten und des Projektgewinns. Die Anforderungsanalyse hilft auch dabei, die Risiken gleich zu Beginn zu identifizieren, sodass Strategien zur Risikominderung von Anfang an ausgearbeitet werden können. Eine klar strukturierte Dokumentation sorgt für eine bessere Zusammenarbeit und ein besseres Verständnis sowohl für das Entwicklungsteam als auch für die Kunden.

Schritt 3: Design (Entwurf)

Während der Entwurfsphase wird die detaillierte Softwarearchitektur erstellt, die den spezifischen Projektanforderungen entspricht. Dies ist eine Art visuelle Modellierung von der Funktionalität der Lösung bis hin zur Definition der grundlegenden Hardware- / Softwarekomponenten, Softwaretools für die zukünftige Entwicklung, Strukturierungsfähigkeiten und Prozesse zur Realisierung der Geschäftsanforderungen und -ziele der vorgeschlagenen Lösung.

Schritt 4: Development (Entwicklung)

In der Entwicklungsphase geht es darum, Code zu schreiben und die Konstruktionsdokumentation innerhalb des Softwareentwicklungsprozesses in die eigentliche Software umzuwandeln. Das Software-Engineering-Team muss sicherstellen, dass der Code den Softwareanforderungsspezifikationen und den Anforderungen der Stakeholder entspricht usw. Der Release-Zyklus für die Softwareentwicklung reicht vom Alpha-, Beta- und Release-Kandidaten bis zum eigentlichen Produktionsaufbau. Sobald die vollständige Architektur (DB, API usw.) und die geplante Funktionalität der Lösung erstellt sind, beginnt die Testphase.

Schritt 5: Testing

Das QA-Team (Quality Assurance) führt eine Reihe von Tests durch, darunter Funktionstests, Systemintegration und Interoperabilität sowie Benutzerakzeptanztests usw., um sicherzustellen, dass der Code sauber ist und die Geschäftsziele der Lösung erreicht werden.

Schritt 6: Implementation

Die neu erstellte und getestete Anwendung wird in die Produktion verschoben, einschließlich der Übertragung von Daten und Komponenten, während in den nächsten Versionen nur die spezifischen Änderungen bereitgestellt werden. Jetzt können System-Analysten und Endbenutzer die fertige Anwendung tatsächlich sehen und ausprobieren.

Schritt 7: Maintenance (Wartung)

Die letzte Phase des Softwareentwicklung- Lebenszyklus umfasst Wartung und regelmäßige Updates. Die Phase wird mit größter Aufmerksamkeit behandelt, da das Produkt während der Phase poliert, verbessert und, entsprechend den tatsächlichen Rückmeldungen zu seiner Leistung, verfeinert wird. Neue Funktionen werden hinzugefügt, oder zusätzliche Benutzeranforderungen erfüllt.