

Projektbericht:

Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online-Quizsystems

Kurs: ISEF01 – Projekt Software Engineering

Datum: 22.12.2022

Tutor: [REDACTED]

Gruppe	Nachname	Vorname	Matrikelnummer	Studiengang	Seitenbereich (von – bis)
Projektleitung	[REDACTED]	Florian	9200145	Informatik B.Sc.	6-17
Mitglied 2	Veigel	Manuel	9160309	Wirtschaftsinformatik B.Sc.	29-38
Mitglied 3	[REDACTED]	Markus	32008638	Wirtschaftsinformatik B.Sc.	17-29

Inhaltsverzeichnis

TABELLENVERZEICHNIS	4
ABBILDUNGSVERZEICHNIS	4
ABKÜRZUNGSVERZEICHNIS	5
1. EINLEITUNG.....	6
1.1. PROJEKTZIEL	6
1.2. PROJEKTIDEE/ PROJEKTVISION.....	6
1.3. PROJEKTUMFELD	6
1.4. PROJEKTPLANUNG UND -ORGANISATION	6
1.5. METHODIK	8
2. PROJEKTKONFIGURATION (MS1).....	9
2.1. ANFORDERUNGEN.....	9
2.2. LÖSUNGSANSATZ	10
2.3. ANNAHMEN UND BESCHRÄNKUNGEN	10
2.4. PROJEKTSTRUKTUR UND ABLAUF	11
2.5. RISIKOMANAGEMENT	13
2.6. ROLLEN UND VERANTWORTUNGEN	14
2.7. KOMMUNIKATIONSMANAGEMENT	14
2.8. PERSONALMANAGEMENT	14
3. PROJEKTVIDEO (MS2)	15
4. DOKUMENTATIONSKONZEPT (MS3)	16
4.1. BESCHREIBUNG DER PROJEKTDOKUMENTE.....	16
4.1.1. <i>Product-Backlog/ Anforderungsliste.....</i>	<i>16</i>
4.1.2. <i>Risikoliste</i>	<i>17</i>
4.1.3. <i>Risiko-Werte-Matrix.....</i>	<i>19</i>
4.1.4. <i>Testprotokoll.....</i>	<i>20</i>
4.1.5. <i>Ergebnisdokumentation.....</i>	<i>21</i>
4.1.6. <i>Benutzerhandbuch.....</i>	<i>21</i>
4.2. IMPLEMENTIERUNGSPROZESS.....	21
4.2.1. <i>Vorgehensmodell</i>	<i>21</i>
4.2.2. <i>Rollenverteilung</i>	<i>22</i>
4.3. IMPLEMENTIERUNG.....	22
4.3.1. <i>Architektur</i>	<i>22</i>
4.3.2. <i>Prozessbeschreibung und Modelle.....</i>	<i>23</i>
4.3.3. <i>Programmiersprachen</i>	<i>23</i>
4.3.4. <i>eingesetzte Werkzeuge.....</i>	<i>24</i>

4.4.	QUALITÄTSMANAGEMENT	25
4.4.1.	Qualitätsziele	25
4.4.2.	Qualitätsplanung	26
4.4.3.	Konstruktives Qualitätsmanagement	26
4.4.4.	IT- Sicherheit	27
4.4.5.	Statische Qualitätssicherung	27
4.4.6.	Review-Verfahren	27
4.4.7.	Testfallerstellung	28
5.	ENTWICKLUNG UND TECHNISCHE UMSETZUNG DER QUIZ-APP (MS4)	29
5.1.	VORBEREITUNG	30
5.2.	ENTWICKLUNG (FRONTEND)	30
5.2.1.	Startseite	30
5.2.2.	Hauptseite (Dashboard)	31
5.2.3.	Kursseite	31
5.2.4.	Quizseite	32
5.2.5.	Sonstige Seiten	32
5.3.	ENTWICKLUNG (BACKEND)	32
5.3.1.	Berechtigungsstufen	32
5.3.2.	Benutzerverwaltung mit Authentication	33
5.3.3.	Datenverwaltung mit Cloud Firestore	33
5.3.4.	Datenbanksicherheit	34
5.4.	TESTEN	34
5.5.	BEREITSTELLUNG	35
5.5.1.	Web-App Hosting	35
5.5.2.	Benutzervideo	35
6.	SCREENCAST (MS5)	35
7.	SCHLUSSTEIL: AUSWERTUNG UND REFLEXION DER PROJEKTERGEBNISSE	37
8.	LITERATURVERZEICHNIS	39
9.	VERZEICHNIS DER ANHÄNGE	42

Tabellenverzeichnis

Tabelle 1: Funktionale und nicht-funktionale Anforderungen	9
Tabelle 2: Teilaufgaben und Arbeitspakete	11
Tabelle 3: Urlaubs-/Abwesenheitsplanung	15
Tabelle 4: Risikoliste	18
Tabelle 5: Risiko-Wert-Matrix	20
Tabelle 6: Tools zur statischen Codeanalyse	27

Abbildungsverzeichnis

Abbildung 1: Cloud Firestore	34
------------------------------------	----

Abkürzungsverzeichnis

App	<i>Applikation</i>
CDN	<i>Content Delivery Network</i>
FAQ	<i>Frequently asked questions</i>
GUI	<i>Graphical user interface</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
HTTPS	<i>Hypertext Transfer Protocol Secure</i>
NoSQL	<i>Not only SQL</i>
SDK	<i>Software development kit</i>
SPA.....	<i>Single-page application</i>
SQL.....	<i>Structured Query Language</i>
SSL	<i>Secure Sockets Layer</i>

1. Einleitung

Im folgenden Einleitungsteil dieses Projektberichts, sollen zunächst das Projektziel und die Projektidee vorgestellt werden. Danach wird das Projektumfeld, die Projektplanung und Projektorganisation sowie die Methodik beschrieben.

1.1. Projektziel

Ziel des Projekts ist die *Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online-Quizsystems*.

1.2. Projektidee/ Projektvision

Die im Rahmen des Projekts entstehende Quiz-App soll eine Webanwendung sein, die von den Studierenden über einen Webbrowser bedient und genutzt werden kann. Die App soll den Studierenden bei der Festigung von Lerninhalten und zur Vorbereitung auf die Klausur helfen. Im Fokus der Anwendung soll stehen, dass die Studierenden gemeinsam die Fragen inkl. der zugehörigen Antworten in einen Fragenkatalog einpflegen können. Aufbauend auf diesem Fragenkatalog gibt es zwei verschiedene Quiz-Spielmodi, mit denen die Studierenden ihr Wissen festigen können.

1.3. Projektumfeld

Das Projekt wird in einer Gruppenarbeit im Rahmen des Kurses ISEF01 – Projekt Software Engineering an der IU-Fernstudium realisiert. Die Projektgruppe besteht aus den drei Studierenden Florian [REDACTED] Manuel Veigel und Markus [REDACTED]. Betreut wird das Projekt von dem Tutor des Kurses, Herrn [REDACTED]. Der Austausch der Projektdokumente mit dem Tutor ist über die Redmine Plattform erfolgt. Über diese Plattform hat die Projektgruppe auch das Feedback zu den einzelnen Ergebnissen der Meilensteine vom Tutor erhalten.

Für das Projektthema standen die folgenden drei Themen zur Auswahl:

- Konzeption, Bewertung und Umsetzung von Systemen zur Unterstützung von Teams und Tutoren bei der Bearbeitung der Software Engineering-Fallstudien
- Analyse, Konzeption und prototypische Umsetzung eines Korrekturmanagementsystems
- Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online Quizsystems

1.4. Projektplanung und -organisation

Nachdem sich die Projektgruppe über das Kursforum gefunden hat, wurde zunächst das Projektthema gewählt. Hierbei hat sich das Projektteam für das Thema *Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online-Quizsystems* entschieden. Dieses Thema hat bei allen drei Projektmitgliedern das größte Interesse geweckt. Zum einen war es besonders interessant selbst eine prototypische Umsetzung des Systems machen zu können und

somit die eigenen Vorstellungen in so ein System einfließen lassen zu können. Zum anderen konnten sich alle drei Projektmitglieder am besten mit diesem Thema identifizieren, da es im Studierenden-Alltag sehr nützlich wäre. Alles in allem hat dieses Thema bei der Projektgruppe das größte Interesse geweckt.

Anschließend wurde die grundlegende Projektorganisation festgelegt. In diesem Rahmen hat das Projektteam zunächst die Projektleitung festgelegt. Diese Rolle hat Florian [REDACTED] übernommen. Grundlage dieser Entscheidung war, dass er im beruflichen Umfeld schon erste Erfahrungen im Projektmanagement sammeln konnte. Als Projektleiter hat Florian u.a. die Kommunikation mit dem Tutor und die grundlegende Organisation des Projekts übernommen.

Darüber hinaus wurde im Rahmen der Projektplanung auch die Art und Weise der internen Kommunikation festgelegt. Da eine persönliche Zusammenarbeit aufgrund des Fernstudiums nicht möglich gewesen ist, hat sich die Gruppe dafür entschieden, als Hauptkommunikationskanal Microsoft Teams zu verwenden. Für die Bearbeitung des Projekts wurde daher ein neues Team erstellt. Über das Team konnte die gesamte Kommunikation und alle Dokumente an einem zentralen Ort gebündelt werden. Es wurde sowohl zur schriftlichen Abstimmung als auch zur Organisation und zum Abhalten von Meetings genutzt. Darüber hinaus diente es als zentraler Dateispeicher für die Projektdokumente. Hierdurch hatten alle Teammitglieder immer Zugriff auf die aktuellen Versionen der Projektdokumente und auch eine gemeinsame, parallele Bearbeitung der Dokumente war möglich. Zusätzlich wurden die dort gespeicherten Dokumente automatisch versioniert, sodass auch auf frühere Versionen der Dokumente zurückgegriffen werden konnte.

Nachdem das Thema, die Projektleitung und die Kommunikation festgelegt wurden, hat der Projektleiter die Projektgruppe beim Tutor angemeldet. Dieser hat das Projektthema bestätigt und ein entsprechendes Projekt in der Redmine Plattform angelegt. Über diese ist im weiteren Projektverlauf vor allem der Dokumentenaustausch mit dem Tutor erfolgt.

Nachdem der Tutor das Projektthema bestätigt hat, hat das Projektteam im Rahmen der Projektplanung einen Zeitplan/ Meilensteinplan für den Projektverlauf erstellt. Basis dieses Zeitplans waren die für das Projekt vorgegebenen folgenden Meilensteine:

- MS1: Projektkonfiguration abgeschlossen und bereitgestellt
- MS2: Projektvideo bereitgestellt
- MS3: Dokumentationskonzept bereitgestellt
- MS4: Zentrale Ergebnisartefakte bereitgestellt
- MS5: Ergebnispräsentation als Screencast bereitgestellt
- MS6: Finale Ergebnisse, Dokumentation und Projektbericht bereitgestellt

Ausgehend vom Projektstart Mitte August 2022 hat die Projektgruppe zunächst einen Zeitrahmen für das gesamte Projekt und einen geplanten Endtermin festgelegt. Als Zeitrahmen wurden vier Monate gewählt. Damit hat sich für das Projekt ein geplantes Enddatum am 15.12.2022 ergeben.

Anschließend wurden die o.g. Meilensteine auf diesen Zeitraum aufgeteilt. Eine detaillierte Zeitplanung inkl. einer Aufwandsschätzung hat dann im Meilenstein 1 bei der Projektkonfiguration stattgefunden.

1.5. Methodik

Die Auswahl der Methodik im Rahmen des Projektmanagements, muss aufgrund der Tatsache das Projekte komplexe Vorhaben sind, ausgiebig geplant werden. Dabei spielen Aspekte wie die Anzahl unterschiedlicher Komponenten, Art der Vernetzung im System sowie in ihrer Umwelt und die Veränderung des Systems die sich im Laufe der Zeit ergibt, eine wichtige Rolle (Patzak & Rattay, 2018, S.215). Dabei wird die Gesamtheit des Projekts in einzelne Schritte unterteilt. Wobei es keine optimale Lösung für die Gliederungstiefe und Gliederungslogik gibt (Patzak & Rattay, 2018, S. 216). Das Projektteam hat einen Projektstruktur verfolgt, bei der die Gesamtaufgabe „*Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online-Quizsystems*“ in einzelne Teilaufgaben (Arbeitspakete) heruntergebrochen wurde (Patzak & Rattay, 2018, S. 217). Diesbezüglich wurden in Teams regelmäßig Meetings zur übersichtlichen Darstellung des Projektinhaltsdurchgeführt. Ebenso fand eine Terminplanung der einzelnen Meilensteine statt. Dadurch war es dem Projektteam möglich alle erforderlichen Aufgaben für das Projekt zu sammeln, deren Beziehungen zueinander zu analysieren, auszuführende Arbeitspakete zusammenzusetzen, weitere Ergänzungen vorzunehmen sowie die Streichung von Doppeltätigkeiten (Patzak & Rattay, 2018, S. 218-219)

Für das Projekt wurden folgende Methoden des Projektmanagements angewendet:

- Meilensteinplanung
- Anforderungsanalyse (funktionale und nicht-funktionale Anforderungen)
- Risikoanalyse (Risiko-Werte-Matrix, Risikoliste)
- Product-Backlog
- Kanban-Board

2. Projektkonfiguration (MS1)

Der folgende Hauptteil des Projektberichts, beschreibt die Projektführung. Die Gliederung orientiert sich dabei an den Meilensteinen des Projekts.

Im Rahmen des ersten Meilensteins wurde die Projektkonfiguration erstellt. Hierbei hat das Projektteam zunächst die Projektvision aus [Kapitel 1.2](#) definiert. Die Projektvision ist die Grundlage aller weiteren Planungen und beschreibt den gewünschten Zielzustand. Anschließend wurde das Projektteam vorgestellt. Die Vorstellung beinhaltet den Namen der Teammitglieder, ein Foto, eine kurze Profilbeschreibung und eine Übersicht über die jeweiligen Skills. Die Vorstellung des Projektteams war vor allem für das Projektteam selbst sehr hilfreich. Da alle Teammitglieder im Fernstudium sind und sich persönlich nicht kennen, hat die Vorstellung des Projektteams vor allem dabei geholfen, die anderen Mitglieder besser kennenzulernen und mehr über deren Skills zu erfahren. Nach der Vorstellung des Projektteams wurde der Projektauftrag definiert. Der Projektauftrag ist vor allem deshalb wichtig, weil er die Grundlage für die Projektplanung ist. „Er ist Orientierungspunkt für die Ausrichtung der Aktivitäten im Projekt, für die Bewertung des Fortschritts und für die Priorisierung der Aufgaben während der Projektdurchführung – er hat eine Leuchtturmfunktion“ (Javorsky, 2019, S. 2). Damit der Projektauftrag auch in diesem Projekt als „Leuchtturm“ für das weitere Projekt dienen kann, wurden zunächst die Anforderungen definiert.

2.1. Anforderungen

Das Projektteam hat die folgenden funktionalen und nicht funktionalen Anforderungen definiert:

Tabelle 1: Funktionale und nicht-funktionale Anforderungen

Anforderungen	
Funktionale Anforderungen	Nicht-funktionale Anforderungen
• Fragenkatalog, zu dem Studierende Frage inkl. Antworten hinzufügen können • Zwei verschiedene Spielmodi, die auf dem Fragenkatalog basieren: Einen kooperativen Modus und einen Spielmodus, in dem die Studierenden gegeneinander antreten können • Benutzerkonten für die Studierenden mit Benutzername und Passwort • Verschiedene Bereiche für einzelne Kurse	• Das System soll als Webanwendung entwickelt werden • Das Front-End soll auf dem JavaScript-Framework Vue.js basieren • Die Daten sollen in einem Datenbanksystem persistent gespeichert werden • Die Anwendung soll auf einer Cloud Computing Plattform (z.B. Amazon Web Services) gehostet werden.

Quelle: Eigene Darstellung

Die nicht-funktionalen Anforderungen beschreiben, wie unsere App die Leistungen erbringen soll (Trempp, 2022, S. 104). Zusätzlich dazu beschreiben die funktionalen Anforderungen „ein Verhalten oder Ergebnis des Systems“ (Trempp, 2022, S. 120). Mit den oben genannten Anordnungen möchte das Projektteam also zum einen das gewünschte spätere Verhalten als auch die Art und Weise wie die Leistung erbracht wird definieren. Diese Anforderungen dienen daher als Grundlage für die spätere technische Umsetzung der App und wurden im weiteren Projektverlauf stetig reflektiert und um weitere Anforderungen ergänzt.

2.2. Lösungsansatz

Zusätzlich zu den Anforderungen hat das Projektteam auch den Lösungsansatz festgelegt. Hierbei ist die Entscheidung für die Realisierung des Front-Ends auf das JavaScript Framework Vue.js gefallen. Die Entscheidung ist aus mehreren Gründen auf Vue.js gefallen. Zunächst hat das Team aus der eigenen Erfahrung eine Vorauswahl erstellt mit Frameworks, die für das Front-End der Webanwendung in Frage kommen. Zu dieser Vorauswahl gehörten die JavaScript Frameworks Vue.js, Angular und React. Diese Auswahl wurde dann anhand von Fachliteratur und Internetrecherche noch einmal darauf hin überprüft, ob die drei Frameworks potenziell geeignet sind und ob es evtl. andere Frameworks gibt, die besser geeignet sind. Die Recherche hat ergeben, dass alle drei der genannten Frameworks gut für das Projekt geeignet sind und bei Entwicklern zu den beliebtesten Frameworks für Webanwendungen gehören (Stoyko, 2022). Das Projektteam hat sich dann für Vue.js entschieden, da die Teammitglieder mit diesem Framework bereits die meisten Erfahrungen sammeln konnten. Außerdem kann es im Vergleich mit den anderen Frameworks schnell erlernt werden (Digital Guide Ionos, 2019). Dies war besonders für die Mitglieder mit weniger Erfahrung wichtig, um das Projekt im geplanten Zeitraum realisieren zu können. Als Backend soll ein Datenbanksystem zum Einsatz kommen, in dem der Fragenkatalog und die Benutzerkonten persistent gespeichert werden. Ob ein relationales Datenbanksystem oder ein sogenanntes No-SQL Datenbanksystem zum Einsatz kommt, hat das Projektteam zu diesem Zeitpunkt noch nicht festgelegt. Die Entscheidung für ein Datenbanksystem ist auch von den im nächsten Abschnitt genannten Beschränkungen abhängig.

2.3. Annahmen und Beschränkungen

Das Projektteam hat bereits während der Projektkonfiguration versucht mögliche Beschränkungen, die im Projektverlauf Einfluss auf die Ergebnisse nehmen könnten, zu identifizieren. Hierbei ist das Projektteam vor allem zu dem Entschluss gekommen, dass die Leistung der App bzw. der gesamten technischen Infrastruktur beschränkt sein wird und nicht beliebig skaliert werden kann. Dies ist darauf zurückzuführen, dass das Projektteam nur auf das kostenlose Kontingent der Cloud Plattformen zurückgreifen möchte. Dementsprechend wird die Leistung der App auf, die im kostenlosen Kontingent enthaltenen Leistungen beschränkt sein. Die Entscheidung für das kostenlose Kontingent beschränkt ebenfalls die Auswahl des Datenbanksystems.

2.4. Projektstruktur und Ablauf

Ein weiterer wichtiger Aspekt der Projektkonfiguration ist die Beschreibung der Projektstruktur und des Projektablaufs. Ausgehend von den bereits festgelegten Meilensteinen hat das Projektteam zunächst die notwendigen Teilaufgaben und Arbeitspakete definiert. Dabei wurde zunächst aus den Meilensteinen die notwendigen Teilaufgaben abgeleitet. Anschließend wurden diese Teilaufgaben weiter verfeinert, indem zu jeder Teilaufgabe Arbeitspakete definiert wurden. Ein Arbeitspaket ist „eine in sich geschlossene Aufgabe, die im Projektstrukturplan nicht weiter untergliedert wird“ (Timinger, 2017, S. 80). Durch diese Arbeitspakete wurde von dem Projektteam definiert, was konkret im Projektverlauf erledigt werden muss (Timinger, 2017, S. 80). Für diese Arbeitspakete, konnte dann der Aufwand geschätzt werden. Aus der Summe der geschätzten Aufwände für die einzelnen Arbeitspakete, hat sich dann der geschätzte Aufwand für die jeweilige Teilaufgabe ergeben. Die Ergebnisse werden in der folgenden Tabelle zusammengefasst. Fett gedruckt sind die definierten Teilaufgaben, darunter folgen die zugehörigen Arbeitspakete:

Tabelle 2: Teilaufgaben und Arbeitspakete

Teilaufgaben und Arbeitspakete			
Name	Geschätzter Aufwand in Tagen	Start	Ende
Projektkonfiguration erstellen	15	01.09.2022	15.09.2022
Projektkonfiguration erstellen	12	01.09.2022	12.09.2022
Projektkonfiguration prüfen und freigeben	2	13.09.2022	14.09.2022
Projektkonfiguration abgeschlossen und bereitgestellt (MS 1)	1	15.09.2022	15.09.2022
Projektvideo	10	16.09.2022	25.09.2022
Inhalt definieren	2	16.09.2022	17.09.2022
Video aufzeichnen	5	18.09.2022	22.09.2022
Video schneiden	2	23.09.2022	24.09.2022

Projektvideo bereitgestellt (MS 2)	1	25.09.2022	25.09.2022
Dokumentationskonzept erstellen	20	26.09.2022	15.10.2022
Beschreibung der Projektdokumente	4	26.09.2022	29.09.2022
Implementierungsprozess beschreiben	4	30.09.2022	03.10.2022
Implementierung beschreiben	5	04.10.2022	08.10.2022
Qualitätsmanagement beschreiben	4	09.10.2022	12.10.2022
Review	2	13.10.2022	14.10.2022
Dokumentationskonzept bereitgestellt (MS 3)	1	15.10.2022	15.10.2022
Zentrale Ergebnisartefakte erstellen	31	16.10.2022	15.11.2022
Infrastruktur bereitstellen	3	16.10.2022	18.10.2022
Front-End implementieren	18	19.10.2022	05.11.2022
Back-End implementieren	18	19.10.2022	05.11.2022
Systemtests/Reviews durchführen	5	06.11.2022	10.11.2022
Ergebnisdokumentation erstellen	4	11.11.2022	14.11.2022
Zentrale Ergebnisartefakte bereitgestellt (MS 4)	1	15.11.2022	15.11.2022

Ergebnispräsentation als Screenshot	9	16.11.2022	25.11.2022
Inhalte definieren	2	16.11.2022	17.11.2022
Screenshot aufzeichnen	4	18.11.2022	21.11.2022
Nachbearbeitung/ Review	3	22.11.2022	24.11.2022
Ergebnispräsentation als Screenshot bereitgestellt (MS 5)	1	25.11.2022	25.11.2022
Finale Ergebnisse, Dokumentation und Projektbericht	19	26.11.2022	15.12.2022
Projektbericht erstellen	18	26.11.2022	14.12.2022
Finale Ergebnisse, Dokumentation und Projektbericht bereitgestellt (MS 6)	1	15.12.2022	15.12.2022

Quelle: Eigene Darstellung

Die obenstehende Tabelle wurde anschließend in die Projektmanagement Software „Project Libre“ übertragen (s. Anhang 1: Teilaufgaben und Arbeitspakete), um mithilfe der Software eine detailliertere Planung der Projektstruktur und des Projektablaufs vornehmen zu können. Nachdem alle Arbeitspakete inkl. der Aufwandsschätzung in die Software eingepflegt wurden, konnte mithilfe der Software automatisch ein Gantt-Diagramm erzeugt werden, das den geplanten Projektablauf und die Projektstruktur darstellt. Das Gantt-Diagramm inkl. der Teilaufgaben und Arbeitspakete ist im Anhang 2: Gantt-Diagramm zu finden. Dieses diente als Grundlage für den weiteren Projektverlauf. Nach der Erstellung des Gantt-Diagramms hat das Projektteam mit einer ersten Aktivität im Bereich Risikomanagement begonnen.

2.5. Risikomanagement

Nach der Erstellung des Gantt-Diagramms hat das Projektteam mit einer ersten Aktivität im Bereich Risikomanagement begonnen, da es für gutes Risikomanagement entscheidend ist frühzeitig und proaktiv einzugreifen (Tiemeyer, 2014, S. 508). Auf Grundlage des Gantt-Diagramms konnte ein erstes Risikomanagement im Bereich Zeitplanung erfolgen. Aus dem Gantt Diagramm hat sich ergeben, dass Verzögerungen bei der Bearbeitung der einzelnen Teilaufgaben und Arbeitspakete das

geplante Projektende am 15.12.2022 gefährden könnten. Daher hat sich das Projektteam dazu entschieden, einen Zeitpuffer von zwei Wochen einzuplanen, sodass das Projektende bis zum 31.12.2022 verschoben werden kann. Damit wollte die Projektgruppe zum einen zu großem Zeitdruck vorbeugen. Zum anderen sollte damit gewährleistet werden, dass das Projektziel auch bei einem temporären Ausfall von einem oder mehreren Projektmitgliedern (z.B. aufgrund von Krankheit) noch erreicht werden kann.

2.6. Rollen und Verantwortungen

Neben den bereits genannten Punkten hat das Projektteam im Rahmen der Projektkonfiguration die Rollen und Verantwortungen der einzelnen Teammitglieder bestimmt. Hierbei wurde zunächst aufgegriffen, dass Florian [REDACTED] die Rolle des Projektleiters einnimmt und damit zuständig ist für das Übernehmen der internen und externen Kommunikation. Darüber hinaus wurde festgelegt, dass alle Teammitglieder in der Verantwortung sind, die Ergebnisse der anderen Mitglieder im Rahmen eines Reviews zu überprüfen und somit eine Qualitätssicherung zu implementieren. Diese Entscheidung wurde daher getroffen, da es sich um ein kleines Projektteam handelt und somit alle Mitglieder in allen Bereichen tätig sein müssen, um den Projekterfolg zu gewährleisten.

2.7. Kommunikationsmanagement

Im Bereich des Kommunikationsmanagements wurden die in [Abschnitt 1.4.](#) bereits festgelegten Punkte erneut aufgegriffen, hinterfragt und verfeinert. Die Hauptkommunikationsplattform für das Projektteam sollte weiterhin Microsoft Teams bleiben. Ergänzend wurde festgelegt, dass einmal wöchentlich ein Regelmeeting stattfinden soll. Dieses Meeting dient unter anderem dazu, die aktuellen Ergebnisse auszutauschen, offene Fragen zu klären und das weitere Vorgehen für die nächste Woche abzustimmen. Außerdem sollte es dazu dienen den aktuellen Projektverlauf gemeinsam zu reflektieren und mit dem geplanten Verlauf zu vergleichen. Dadurch sollte gewährleistet werden, dass das Projektteam immer weiß, ob es mit der Bearbeitung der einzelnen Arbeitspakete im Zeitplan liegt. Außerdem sollte es so möglich werden, Verzögerungen und Probleme rechtzeitig zu erkennen und gemeinsam Gegenmaßnahmen einleiten zu können.

2.8. Personalmanagement

Um den geplanten Projektverlauf und das geplante Enddatum nicht zu gefährden, wurde vom Projektteam auch ein Personalmanagement vorgenommen. Hierbei war es vor allem wichtig, die Verfügbarkeiten der Projektmitglieder für den Projektverlauf im Überblick zu haben. Daher wurde zunächst die folgende Urlaubs-/ Abwesenheitsplanung für den Projektzeitraum erstellt.

Tabelle 3: Urlaubs-/Abwesenheitsplanung

	Urlaubs-/Abwesenheitsplanung				
Name	August	September	Oktober	November	Dezember
Florian [REDACTED]	27.08.-31.08.	01.09.-11.09.	-	-	-
Manuel Veigel	-	16.09.-30.09.	-	-	-
Markus [REDACTED]	-	-	-	-	-

Quelle: Eigene Darstellung

Auf Grundlage der oben dargestellten Urlaubs-/ Abwesenheitsplanung wurde zunächst der geplante Projektverlauf daraufhin überprüft, ob der geplante Bearbeitungszeitraum trotz der jeweiligen Abwesenheiten eingehalten werden kann. Dabei ist das Projektteam zu dem Ergebnis gekommen, dass die geplanten Bearbeitungszeiten trotz temporärer Abwesenheiten grundsätzlich eingehalten werden können. Zusätzlich wurde festgelegt, dass bei der Abwesenheit eines Teammitglieds, die anderen Mitglieder, die ausstehenden Tätigkeit übernehmen. Damit dies möglich ist, soll in dem jeweiligen Wochenmeeting vor der geplanten Abwesenheit eine Übergabe stattfinden. Diese dient dazu, den aktuellen Bearbeitungsstand mitzuteilen und die offenen Aufgaben auf die verbleibenden Teammitglieder aufzuteilen.

3. Projektvideo (MS2)

Im Meilenstein 2 (Projektvideo) wurde ein ca. 5-minütiges Video erstellt, indem das Projekt und die Lösungsidee vorgestellt werden. Der Clip diente dazu, dass Projekt und die Lösungsidee kurz vorzustellen. Auf Basis der Projektkonfiguration hat sich das Projektteam ein Konzept überlegt, dass die Inhalte des Videos definiert. Das Konzept wurde in den wöchentlichen Teams-Meetings gemeinsam vom Projektteam erarbeitet und abgestimmt. So wurde festgelegt, welche Inhalte in dem Video präsentiert werden sollen. Die festgelegten Inhalte wurden dann auf PowerPoint Folien übersichtlich zusammengestellt. Anschließend wurden die Inhalte als Video aufgenommen. Das Video wurde mithilfe des kostenlosen Tools "Vimeo" erstellt.

Die folgenden Inhalte wurden im Video geschildert:

- Vorstellung des Projektteams
- Warum ein kooperatives und kollaboratives Quizsystem?
- Welche Vorteile hat das Online-Quizsystem?
- Wie funktioniert das Quizsystem?
- Wie sind die Fragen aufgebaut?
- Wie können Studierende kollaborativ und kooperativ mitwirken?
- Wie ist die technische Umsetzung?

Das fertige Video wurde abschließend in der Redmine-Plattform hochgeladen, um es dem Tutor zur Verfügung zu stellen.

4. Dokumentationskonzept (MS3)

Im dritten Meilenstein wurde das Dokumentationskonzept in Form eines Word Dokuments bereitgestellt.

4.1. Beschreibung der Projektdokumente

Hierbei wurde zu Beginn ein Product Backlog mit den umzusetzenden Anforderungen und Funktionen erarbeitet. Teil einer jeden Software Architektur Analyse ist es kurzfristige und langfristige Anforderungen auszubalancieren. Das Product Backlog stellt ein zentrales Prinzip in der agilen Softwareentwicklung dar, um unnötige Vorleistungen zu erkennen und signifikante architektonische Anforderungen rechtzeitig zu berücksichtigen, sowie um Fehlentscheidungen zu vermeiden. In diesem Zusammenhang hat sich das Projektteam mithilfe von User Stories mögliche Anforderungen überlegt, um wichtige Informationen darüber zu gewinnen, wie die App genutzt werden könnte (Galster et al., 2019, S. 88-89). Daher ist es dem Projektteam möglich zu bestimmen, welche Anforderungen, neben den bereits vorgegebenen, für die Anwender relevant sein könnten und welche nicht (Galster et al., 2019, S.91).

4.1.1. Product-Backlog/ Anforderungsliste

Bei der Erstellung des Product Backlog hat das Projektteam die Anforderungen aus der Projektkonfiguration aufgegriffen und ergänzt. Dabei ist die folgende Anforderungsliste entstanden, die gleichzeitig auch als Product Backlog dient:

- Login-Seite und Benutzerkonten für die Studierenden mit Benutzername und Passwort
- Studenten können sich registrieren
- Studenten können ihr Benutzerprofil einsehen und bearbeiten
- Zwei verschiedene Spielmodi: Einen Kooperativen Modus und einen Spielmodus, in dem die Studierenden gegeneinander antreten können
- Datenbanksystem zur Persistenten Speicherung implementieren
- Studenten können ihr Benutzerprofil einsehen und bearbeiten
- Möglichkeit weitere Kurse hinzuzufügen für den Tutor/ Admin
- Kurse als Favoriten speichern
- Fragenpool für jeden Kurs
- Möglichkeit für die Studierenden weitere Fragen einzureichen
- Möglichkeit für den Tutor fragen freizugeben oder abzulehnen
- Verschiedene Berechtigungsstufen für Studenten, Tutoren und Administratoren
- App soll an das Design der IU angepasst sein
- FAQ-Seite mit den wichtigsten Fragen und Antworten

- Die Studierenden sollen die Möglichkeit haben eigene Einstellungen vornehmen zu können, um den Schwierigkeitsgrad des Fragekatalogs zu bestimmen.

Das Product Backlog bildet die Grundlage für das Kanban Board des Projekts, welches das Team anstelle eines Sprint Backlogs als Vorgehensmodell ausgewählt hatte. Die Gründe dafür waren, dass ein Sprint iterativ (wiederholt) aufeinanderfolgend durchgeführt wird. In täglichen Meetings wird der aktuelle Stand der Arbeiten und Hindernisse im Team besprochen (Dreus et al., 2021, S. 93). Dies war aufgrund der Tatsache, dass sich die Teammitglieder noch beruflichen oder anderen Verpflichtungen widmen mussten, zeitlich nicht möglich und aus Sicht der Projektmitglieder für den Projektumfang nicht notwendig.

4.1.2. Risikoliste

Um das Risikomanagement, welches bereits bei der Projektkonfiguration begonnen wurde, weiter fortzusetzen hat das Projektteam eine Risikoliste erstellt. Diese dient dem Team dazu, um die bekannten Risiken des Projekts besser verwalten zu können. Sie dient der Projektleitung und dem Projektteam dazu, qualitativ bekannte Risiken zu beurteilen und ggf. Gegenmaßnahmen einleiten zu können, die das Eintreten von Fehlern verhindern oder zumindest den Schaden verringern (Kuster et al., 2022, S.95). Die Risikoliste wurde mit den zu diesem Zeitpunkt bekannten Risiken erstellt. Da sich die Risiken aber im Laufe des Projektes ändern können, fanden im weiteren Verlauf des Projektes regelmäßig Risikoanalysen statt. Hierfür wurden die wöchentlichen Meetings genutzt.

Die Risikoliste enthält eine ID und eine Beschreibung des Risikos. Darüber hinaus werden die Auswirkungen bei Eintritt des Risikos kurz beschrieben. Außerdem wird das Risiko hinsichtlich der Eintrittswahrscheinlichkeit und der möglichen Schadenshöhe kurz bewertet und mögliche Gegenmaßnahmen beschrieben. Bei der Bewertung der Eintrittswahrscheinlichkeit wurde eine Einstufung von 1 (geringe Eintrittswahrscheinlichkeit) bis 10 (hohe Eintrittswahrscheinlichkeit) vorgenommen. Bei der Bewertung der Auswirkungen wurde zwischen den drei Stufen gering, mittel und hoch unterschieden.

Aufgrund des kleinen Projektteams mit drei Mitgliedern wurde auf eine Beschreibung von Zuständigkeiten verzichtet. Im Falle von notwendigen Gegenmaßnahmen wird die Zuständigkeit kurzfristig und nach Verfügbarkeit festgelegt.

Tabelle 4: Risikoliste

Risikoliste				
ID	Beschreibung	Auswirkungen	Bewertung des Risikos	Gegenmaßnahmen
1	Ausfall eines Teammitglieds z.B. aufgrund von Krankheit	- Einhaltung des Zeitplans gefährdet - Verbleibende Teammitglieder müssen die Aufgaben übernehmen	- Eintrittswahrscheinlichkeit: 8 - Auswirkung: Hoch	- Der Ausfall eines Teammitglieds z.B. aufgrund von Krankheit kann nicht verhindert werden - Verminderung des Schadens durch regelmäßigen Austausch, gleichmäßige Verteilung der Aufgaben
2	Meilenstein kann nicht gemäß Zeitplan erreicht werden	- Weniger Bearbeitungszeit für die folgenden Meilensteinen - Abschluss des Projektes zum geplanten Termin gefährdet	- Eintrittswahrscheinlichkeit: 5 - Auswirkung: Mittel	- Zeitpuffer für das Projekt - andere Meilensteine können schneller abgeschlossen werden
3	Funktionen lassen sich nicht wie geplant realisieren	- Implementierung der jeweiligen Funktion gefährdet	- Eintrittswahrscheinlichkeit: 6 - Auswirkung: mittel bis hoch	Je nach Wichtigkeit der Funktion: - Alternative Realisierungsmöglichkeiten suchen - Streichen der geplanten Funktion
4	Änderung des kostenlosen Kontingents beim	- Anwendung kann nicht wie geplant gehostet werden	- Eintrittswahrscheinlichkeit: 3	- Angebot und Ankündigen des gewählten Anbieters regelmäßig kontrollieren

	gewählten Cloud Provider		- Auswirkung: niedrig	- möglicherweise Entstehende Kosten berechnen - zu alternativem Anbieter wechseln, falls zu Kosten zu hoch
5	Datenverlust bei bereits erzielten Projektergebnissen z.B. durch versehentliches Löschen oder Überschreiben bei paralleler Bearbeitung	- Bisher erstellte Projektergebnisse müssen erneut erstellt werden - Zeitverzug	- Eintrittswahrscheinlichkeit: 7 - Auswirkung: hoch	- Versionskontrolle mit git
6	Unstimmigkeiten/ Meinungsverschiedenheiten innerhalb des Teams	- Umsetzung von Anforderungen gefährdet - Projekt wird verzögert - Team löst sich auf	- Eintrittswahrscheinlichkeit: 2 - Auswirkung: hoch	- Regelmäßiger Austausch und Abstimmung in Meetings - Konstruktive Kritik

Quelle: Eigene Darstellung

4.1.3. Risiko-Werte-Matrix

Um die Realisierung der Anforderungen besser planen und eine Priorisierung der anstehenden Aufgaben vornehmen zu können, wurden die Anforderungen in eine Risiko-Werte-Matrix eingetragen (Hoffmann, W., 2017, S.36). Als erstes umzusetzen sind dabei die Anforderungen mit hohem Risiko aber auch hohem Wert für das Projekt. Als zweites werden die Anforderungen umgesetzt, die zwar einen hohen Wert aber eher ein geringes Risiko haben. Zuletzt sollen die Anforderungen umgesetzt werden, mit einem geringen Risiko für das Projekt aber auch einem relativ geringen Nutzen.

Tabelle 5: Risiko-Wert-Matrix

Vermeiden	Als erstes umsetzen
Die Studierenden sollen die Möglichkeit haben eigene Einstellungen vornehmen zu können, um den Schwierigkeitsgrad des Fragekatalogs zu bestimmen.	- Zwei verschiedene Spielmodi: Einen Kooperativen Modus und einen Spielmodus, in dem die Studierenden gegeneinander antreten können - Datenbanksystem zur persistenten Speicherung implementieren - Kursauswahl - Fragenpool für jeden Kurs - Verschiedene Berechtigungsstufen für Student, Tutor und Administrator
Zuletzt umsetzen	Als zweites umsetzen
- Fragenkatalog, zu dem Studierende Fragen inkl. Antworten hinzufügen können - Verschiedene Bereiche für einzelne Kurse - App soll an das Design der IU angepasst sein - Studenten können ihr Benutzerprofil einsehen und bearbeiten - Kurse als Favoriten speichern - FAQ-Seite mit den wichtigsten Fragen und Antworten	- Login-Seite und Benutzerkonten für die Studierenden mit Benutzername und Passwort - Studenten können sich registrieren - Möglichkeit weitere Kurse hinzuzufügen für den Tutor/ Admin - Möglichkeit für die Studierenden weitere Fragen einzureichen - Möglichkeit für den Tutor fragen freizugeben oder abzulehnen

Quelle: Eigene Darstellung

4.1.4. Testprotokoll

Im Rahmen der Erstellung eines Testkonzepts hat sich das Projektteam dazu entschieden, ein Testprotokoll zu entwerfen. Mit den Tests wurde die Erfüllung der umzusetzenden Anforderungen und Funktionen aus der Anforderungsliste überprüft. Nach Abschluss der Tests wurden alle gefunden Fehler im Testprotokoll erfasst (Kuster et al., 2022, S.179).

Das ausgefüllte Testprotokoll wurde dem Anhang 6: Testprotokoll hinzugefügt.

4.1.5. Ergebnisdokumentation

Neben dem Testprotokoll wurde im Rahmen der Ergebnisdokumentation ein Screencast aufgenommen, indem das Projektteam die Ergebnisse des Projektes präsentiert und dokumentiert. Laut MacLeod et al. (2017, S.1) sind im Laufe der Zeit Screencasts eine „beliebte Alternative zur textbasierten Dokumentation geworden.“

4.1.6. Benutzerhandbuch

Zusätzlich wurde ein Konzept für die Benutzerschulung erstellt, in denen die Anwender mit den Funktionen der Web-Anwendung vertraut gemacht werden. Dabei wurden auf klassische handgeschriebene Benutzerdokumentationen bzw. Bedienanweisungen verzichtet. Stattdessen sollte ein Benutzervideo für die Quiz-App erstellt werden, in denen den Anwendern Schritt für Schritt der Umgang mit der App erklärt wird (Küster et al., 2022, S. 214-215). Dort werden konkret die einzelnen Funktionen, Möglichkeiten und Inhalte sowohl für die Studierenden und Tutoren verdeutlicht, sodass jedem User ein einfacher Einstieg in die App ermöglicht werden kann. Zusätzlich sollte eine FAQ-Seite mit den wichtigsten Fragen und Antworten bereitgestellt werden.

4.2. Implementierungsprozess

Im Rahmen des Implementierungsprozesses wurden die einzelnen Schritte für eine problemlose Projektdurchführung definiert (Broy & Kuhrmann, 2021, S.428). Als wichtige Bestandteile in dieser Phase ist das Vorgehensmodell und die Rollenverteilung zu nennen.

4.2.1. Vorgehensmodell

Bei einem Vorgehensmodell handelt es sich um eine Art Bindeglied zwischen den Aufgaben der Projektorganisation und des Managements. Ebenso werden technische und methodische Aufgaben im Bereich der System- und Softwareentwicklung erfasst, um eine strukturierte Vorgehensweise und Zuordnung von spezifischen Aufgaben zu gewährleisten. Dadurch ist für alle Teammitgliedern, das Projekt greifbar und die Grundlage für die Projektplanung und Projektdurchführung können geschaffen werden. (Broy & Kuhrmann, 2021, S. 83-83).

Beim Vorgehensmodell hat sich das Projektteam für das agile Kanban Modell entschieden. Es stammt aus dem Produktionssteuerungsprozess der Toyota Motor Corporation und findet in der Softwareentwicklung zur Reduktion von Verschwendungen Anwendung. Sie ist eine einfache Methode zur Organisation und Verbesserung der Arbeitsprozesse. Scrum gibt dabei den Rahmen vor und hilft dabei die passenden Aufgaben nach dem [Pull-Prinzip] auszuwählen (Broy & Kuhrmann, 2021, S. 112-113). Das Projekt ist mit dem geplanten Umfang, zeitlichen Verlauf (Abschluss bis 15.12.2022) und den drei Teammitgliedern eher klein, weshalb andere agile Methoden wie z.B. ein Sprint zu aufwändig wären. Ein Sprint kennzeichnet eine Iteration einer Time Box laut Scrum für bis zu 30 Kalendertage pro Sprint vor. Zusätzlich müsste für jeden Tag ein Daily Scrum durchgeführt

werden, welcher aufgrund von Verpflichtungen bezogen auf die berufliche Tätigkeit oder anderen Prüfungsleistungen nicht umsetzbar wäre (Broy & Kuhrmann, 2021, S. 106).

4.2.2. Rollenverteilung

Während des Implementierungsprozesses hat das Projektteam auf eine konkrete Rollenverteilung verzichtet. Der Projektleiter des Teams, Florian [REDACTED] war verantwortlich für die interne und externe Kommunikation mit dem Tutor und das Hochladen der fertiggestellten Meilensteine. Im Rahmen des Projekts war jedes Teammitglied verantwortlich, die Ergebnisse der anderen Mitglieder in Form von Reviews zu untersuchen.

Im Verlauf des Entwicklungsprozesses hatte jedes Teammitglied die Möglichkeit selbständig offene Anforderungen und Funktionen aus dem Backlog auszuwählen und zu bearbeiten. Alle verbliebenen Aufgaben wurden in einer Art Burndown-Chart dargestellt, wo die noch zu erledigenden Aufgaben visualisiert wurden (Broy & Kuhrmann, 2021, S.106). Bei Schwierigkeiten und Hindernissen, koordinierte der Projektleiter den internen Austausch.

4.3. Implementierung

Im Anschluss an alle Entwurfsaufgaben wurde in Abhängigkeit vom ausgewählten Vorgehensmodell mit der Implementierung begonnen. In dieser Phase geht es darum einen Systementwurf gemäß der erarbeiteten Schnittstellenspezifikation zu realisieren, indem Komponenten und Module ausprogrammiert werden. Bestandteile der Implementierung sind die Entwicklung der Architektur, Beschreibung des Prozessablaufs, Modelle, Programmiersprachen und die eingesetzten Werkzeuge, die den Entwicklungsprozess effizienter gestalten (Broy & Kuhrmann, 2021, S. 427).

4.3.1. Architektur

Entwickelt wurde die Anwendung nach dem MVVM Architektur Modell. Die MVVM-Architektur trennt das GUI-Design vom Backend der Anwendung und kann ideal bei der Entwicklung von Clientanwendungen eingesetzt werden. Das Modell stellt dabei das Domänenmodell dar und die View die Benutzeroberfläche der Anwendung mit der, der Anwender interagiert. Das View-Modell enthält die Logik von View. Beide kommunizieren miteinander. Dadurch ist die gemeinsame Nutzung von berechneten bzw. resultierenden Daten für die Ansicht möglich (Setia, 2020). Dieses lässt sich mit dem für die Implementierung gewählten Framework (Vue.js) sehr gut umsetzen. Bei Visual Studio Code handelt es sich um eine leistungsfähige Code-fokussierte Entwicklungsumgebung, welche den kompletten Lebenszyklus der Anwendungsentwicklung mit einem integrierten Debugger (Versionskontroll-Engine Git) unterstützt. Es kann mit einzelnen Codedateien, Ordnern oder losen Dateien gearbeitet werden mit dem Ziel, Web-, Mobil- und Cloud-Plattformen zu schreiben. Diese Entwicklung gewährleistet große Flexibilität und Unabhängigkeit und unterstützt die zur Erstellung der App benötigten Sprachen JavaScript, HTML und CSS (Cascading Style Sheets) (Sole, A. D., 2021, S.1-3).

Die Daten werden persistent in der Google Firebase Datenbank *Cloud Firestore* gespeichert. Bei dieser handelt es sich um eine NoSQL Datenbank, die über eine API von der Anwendung angesprochen wird und für die Entwicklung von Mobil-, Web- und Serviceanwendungen eingesetzt wird (Vijayan, 2018).

4.3.2. Prozessbeschreibung und Modelle

Um den Überblick über die Prozesse der Anwendung zu behalten, wurde ein Use Case Diagramm erstellt (siehe Anhang 3: Use-Case-Diagramm). Mithilfe eines Anwendungsfalldiagramms in UML konnten die Funktionen identifiziert werden, die für die Quiz-App realisiert werden müssen. Beispielsweise müssen die Studenten in der Lage sein sich in die Anwendung einloggen zu können. Anhand des Use Case Diagramms war es dem gesamten Projektteam möglich die Anforderung und notwendigen Funktionen des Systems zu verstehen (Xue-Geng, S. -r. Zou, 2022).

Zusätzlich wurde ein UML-Zustandsdiagramm entwickelt (siehe Anhang 4: UML-Zustandsdiagramm), welches die Prozesse und den Dialogfluss innerhalb der Anwendung visualisiert. In dem Diagramm werden die Übergänge zwischen den einzelnen Objekten abgebildet. Wichtige Elemente sind dabei Zustände und Übergänge, welche mit Rechtecken mit abgerundeten Ecken formatiert und mit dem Namen des Zustands gekennzeichnet werden. Übergänge geben die Zustandsveränderung an und weisen von einem Zustand zu einem anderen. Außerdem besitzt jedes Zustandsdiagramm einen "Anfangszustand, Endzustand, Ereignis, Zustand, Übergangsverhalten, Austrittspunkt, Übergang und Auslöser" (Nyamsi, 2020, S.16-17).

Im Zusammenhang mit dem Design der Datenstruktur wurde ein Entity-Relationship-Diagramm (siehe Anhang 5: Entity-Relationship-Diagramm) erstellt. Das Diagramm stellt die Datenstruktur der Anwendungsdaten dar. Als Darstellungsform wurde die modifizierte Chen-Notation für (min, max) gewählt. In einem Entity-Relationship-Diagramm sind alle Klassen Kandidaten für Entitätstypen, für die es betriebliche Funktionen gibt oder an Relationstypen beteiligt sind (Lübbert & Zeh, 2021, S. 9-11).

Beispiel:

- Ein Kurs beinhaltet mehrere Fragen und eine Frage gehört genau zu einem Kurs.
- Ein Tutor administriert mehrere Kurse und ein Kurs wird genau von einem Tutor administriert.
- Ein Student hat mehrere Kurse und ein Kurs hat mehrere Studenten

4.3.3. Programmiersprachen

Für die Programmierung wurden auf die Sprachen HTML, JavaScript und CSS zurückgegriffen.

Zur Visualisierung der Anwendung wird auf die Auszeichnungssprache HTML (Hypertext Markup Language) zurückgegriffen. Damit kann die Struktur und der Aufbau des Quizsystems gestaltet werden. Sie dient als Auszeichnungssprache zur Verwandlung von Textdokumente in Webseiten und Webapplikationen. Sie bestimmt die Struktur und den Aufbau einer Internetseite. Ihre

Hauptaufgabe besteht darin die semantische Beschreibung des Inhalts wiederzugeben und das Dokument zu strukturieren (Robbins, 2010, S.1).

Für das Layout sowie Elemente zur Positionierung und Animation der Anwendung wird Cascading Stylesheet (CSS) verwendet. Durch Änderungen am Stylesheet lassen sich mit Formatvorlagen, das Aussehen an der Quiz-App gestalten. Die Stärke besteht in der uneingeschränkten Flexibilität, falls das Layout nicht mehr zeitgemäß ist oder neue Strukturen, Änderungen erfordern. Ist das Aussehen, mit CSS einmal festgelegt, bezieht sich das, auf alle dort definierten HTML-Elementen und müssen nicht erneut definiert werden (Attardi, 2020, S.1).

Mit der Programmiersprache JavaScript werden die notwendigen Funktionalitäten und das dynamische Verhalten der App realisiert. Dabei werden die Skripte in Form von Quellcode gespeichert und während der Ausführung in Maschinencode transformiert (Bewersdorff, 2018, S. 43-44).

4.3.4. eingesetzte Werkzeuge

Um den Entwicklungsprozess effizienter zu gestalten, wurde das intuitive Vue-Framework Nuxt.js eingesetzt. Nuxt.js vereinfacht die Projekterstellung und automatisiert beispielsweise den Import von Vue-Komponenten und Plugins.

Für den Aufbau der Benutzeroberfläche wurde Vuetify.js verwendet. Vuetify ist ein UI-Framework, das auf Vue.js aufgebaut ist und dem Material-Design-Standard entspricht. Bei Visual Studio Code handelt es sich um eine leistungsfähige Code-fokussierte Entwicklungsumgebung, welcher den kompletten Lebenszyklus der Anwendungsentwicklung mit einem integrierten Debugger (Versionskontroll-Engine Git) unterstützt. Es kann mit einzelnen Codedateien, Ordnern oder losen Dateien gearbeitet werden mit dem Hintergrund, Web-, Mobil- und Cloud-Plattformen zu schreiben. Diese Entwicklung gewährleistet große Flexibilität und Unabhängigkeit und unterstützt die zur Erstellung der App benötigten Sprachen JavaScript, HTML und CSS (Sole, A., D., 2021, S.1-3).

Für dieses Projekt wurde der Extensions Volar, ESLint und Prettier installiert. Die Gründe hierfür sind unter anderem, dass Volar eine Reihe an Tools bereitstellt, die für die Entwicklung von Vue.js Applikationen essenziell sind (z.B. Syntax Highlighting). ESLint von Microsoft analysiert JavaScript-Code und hilft beim Finden und Beheben von Problemen. Es hilft dabei, den Code konsistent und lesbar zu halten. Prettier stellt sicher, dass der Code einheitlich formatiert ist.

Die Versionsverwaltung des Quellcodes lief über das Open-Source-Tool Git. Das Code-Repository wurde auf GitHub gehostet. Mithilfe der Plattform ist das Hosting von Code und die Versionskontrolle innerhalb der Softwareentwicklung möglich. Sie wurde für das Projekt verwendet, da durch einfache Git-Befehle eine Schnittstelle dargestellt werden kann, welche innerhalb des Terminals angewendet werden kann (Speight, 2021, S.135).

4.4. Qualitätsmanagement

Ein Qualitätsmanagementsystem enthält alle zuständigen Verantwortlichkeiten, Abläufe, Dokumentationen, Anweisungen und Leitlinien, die benötigt werden, um die entsprechenden Qualitätsanforderungen für ein Produkt einzuhalten. Im Rahmen des Qualitätsmanagements hat jedes Projektmitglied überprüft, ob die festgelegten Projektziele / Meilensteine/ Projektergebnisse erreicht wurden. In regelmäßigen Meetings wurden die nächsten Schritte abgesprochen, um eine hohe Prozessqualität zu garantieren und die Aufgaben mit der richtigen Priorisierung zu bearbeiten. Aber auch Qualitätsziele, Verantwortlichkeiten, Maßnahmen zur kontinuierlichen Verbesserung wurden dort festgelegt (Linz, 2017, S. 163-164). Dabei hat sich das Projektteam an das Wirkungsprinzip nach PDCA orientiert.

- P=Plan: Das Team hat hinsichtlich der definierten Ziele und Anforderungen, geplant was umgesetzt werden soll.
- D=Do: Das Team hat mit dem Ablauf begonnen.
- C=Check: Das Team hat gemeinsam in den wöchentlichen Meetings überprüft, ob die Ergebnisse der App, den definierten Anforderungen entsprechen.
- A=Act: Das Team hat auf Basis von Abweichungen von den Ergebnissen, Maßnahmen ergriffen, die zur Verbesserung des Prozesses und Optimierung der Produktqualität gemäß den definierten Anforderungen, führen (Linz, 2017, S.165).

4.4.1. Qualitätsziele

Im Rahmen der Qualitätsziele hatte sich das Projektteam mit einer konkreten Definition of Done beschäftigt. In ihr wurde festgelegt, wann eine Anforderung als *erfolgreich umgesetzt* zählt. Fester Bestandteil sind die Qualitätsziele, die das Projektteam zur Erfüllung der Anforderungen benötigt (Broy & Kuhrmann, 2021, S. 283).

Dabei wurden vom Projektteam folgende Qualitätsziele beschlossen:

- Belastbarkeit der Anwendung: Nach der Fertigstellung wird überprüft, wie sich das System verhält, wenn sich mehrere Studenten gleichzeitig einloggen.
- Sicherheit der gespeicherten Daten: Nach einem Absturz der Anwendung sollten alle gespeicherten Daten aus der Datenbank neu geladen werden können.
- Benutzbarkeit: Das System soll ohne großen Aufwand von den Studenten verwendet werden können. Bei registrierten Benutzern sollten es maximal 4 Seiten sein, bevor ein Lern/ Spielmodus gestartet wird. Bei neuen Registrierungen maximal 6 Seiten.
- Änderbarkeit: Studenten müssen in der Lage sein, selbstständig Erweiterungen am Fragenkatalog vornehmen zu können.
- Effizienz: Der Ladeprozess einer Seite darf maximal eine Minute dauern.
- Funktionalität: Die Quiz-App bietet die Möglichkeit aus zwei verschiedenen Spielmodi auszuwählen

Anhand dieser Ziele konnte sichergestellt werden, dass die Anforderungen gemäß den Ansprüchen der Stakeholder umgesetzt werden konnte.

4.4.2. Qualitätsplanung

Bei der Qualitätsplanung wurde festgehalten, dass jedes Projektmitglied für die Ausführung der QS-Aktivitäten zuständig ist. Sobald ein Mitglied eine Aufgabe aus dem Kanban Board beendet hatte, überprüften die übrigen Teilnehmer die Qualität, der Zwischenergebnisse des Projekts. Außerdem wurden nach der Fertigstellung der Web-Anwendung, Anwendungsfallbasierte Testfälle durchgeführt.

4.4.3. Konstruktives Qualitätsmanagement

Das Qualitätsmanagement teilt sich in konstruktives und analytische Qualitätssicherung auf. Hierbei hat das Projektteam bereits vor der Erstellung (a priori) der Software, Maßnahmen bzw. Abgesicherte Vorgehensweisen definiert, um die Qualität des Softwareprodukts zu verbessern und zur Vorbeugung von Fehlern dienen. Die vom Projektteam gewählten Maßnahmen gliedern sich in technische, organisatorische und zwischenmenschliche Maßnahmen (Broy & Kuhrmann, 2021, S. 25-26)

Als technische Maßnahmen sind hierbei beschriebene Entwicklungsumgebung Visual Studio Code inkl. der genannten Extensions Volar, ESLint und Prettier zu nennen. Diese Extensions helfen dem Projektteam dabei die Qualität des erstellten Programmcodes zu erhöhen. So wird durch die Erweiterungen unter anderem eine korrekte Syntax sowie eine einheitliche Formatierung gewährleistet. Durch die einheitliche Formatierung wird der Code für alle Entwickler besser lesbar und erleichtert somit z.B. die Fehlersuche. Extensions bieten den Entwicklern die Möglichkeit innerhalb des Projekts, anhand des Namen die Erweiterung herauszufiltern sowie die Angabe der Version und den Namen des Herausgebers. Außerdem lässt sich die Funktionalität einfach über Visual Studio Code installieren (Speight, 2021, S.14).

Als organisatorische Maßnahmen zählt die Einhaltung des Community Standards, welches für das Schreiben in Vue.js Code eingehalten wurde, damit der Code für den Entwickler besser lesbar und nachvollziehbar ist. Zusätzliche kam das Kanban Template mittels Jira zum Einsatz.

Im Rahmen der zwischenmenschlichen Maßnahmen fanden regelmäßig (meistens einmal in der Woche) Meetings, des Projektteams in Microsoft Teams statt. Sie dienten dazu, die Zusammenarbeit der Teammitglieder, den Austausch und die Kommunikation zu verbessern. Innerhalb des Meetings hat jedes Projektmitglied seine bisherigen Ergebnisse vorgetragen und Feedback in Form von Anregungen, Anerkennung und Verbesserungen erhalten. So konnten auch die Erfahrung, Stärken und Skills der anderen Projektmitgliedern in die eigenen Ergebnisse mit einfließen und die Qualität der einzelnen Ergebnisse erhöht werden.

4.4.4. IT- Sicherheit

Da die Web-Anwendung auf offenen Netzwerken läuft und dadurch angreifbar ist, musste sich das Projektteam mit dem Thema IT-Sicherheit auseinandersetzen (Walter, 2008, S.575). Im Rahmen der IT-Sicherheit hat sich das Projektteam dazu entschlossen, dass die Anwendung nur per HTTPS erreichbar sein soll. "Hierbei wird vom HTTP-Client ein sicherer TSL-Kanal zum Server aufgebaut, über den dann anschließend die HTTP-Daten transferiert werden" (Eckert, 2018, S.778). Dadurch konnte die Kommunikation mit der Anwendung verschlüsselt werden, sodass z.B. die Zugangsdaten nicht im Klartext an die Anwendung übertragen wurden. Außerdem ist die Kommunikation der Anwendung mit der Datenbank per HTTPS verschlüsselt.

Darüber hinaus wurden alle Kennwörter verschlüsselt in der Datenbank gespeichert, sodass diese nicht eingesehen werden können.

4.4.5. Statische Qualitätssicherung

Bei der statischen Qualitätssicherung handelt es sich um analytische Verfahren. Diese stellen Maßnahmen dar, die bei der Überprüfung der Entwicklungsergebnisse unterstützen. In diesem Zusammenhang wurden auf Codeebene die implementierten Bestandteile eines Systems auf Korrektheit überprüft (Broy & Kuhrmann, 2021, S. 26). Hierfür wurden für das Projekt Reviews und eine statische Codeanalyse mithilfe von Extensions Volar, ESLint und Prettier durchgeführt.

Tabelle 6: Tools zur statischen Codeanalyse

Extensions Volar	Syntaxhervorhebung und Refactoring einzelner Dateikomponenten (https://vueschool.io/lessons/volar-the-official-language-feature-extension-for-vs-code)
ESLint	Statische Codeanalyse zur Problemfindung (https://de.eslint.org/)
Prettier	Code-Formatierer, ermöglicht konstanten Stil, Codeanalyse (https://marketplace.visualstudio.com/items?itemName=esbenp.prettier-vscode)

Quelle: Eigene Darstellung

4.4.6. Review-Verfahren

Um die erzeugten Ergebnisse zu überprüfen, fanden nach deren Erstellung Review statt. Hierbei wurde der erstellte Programmcode auf Fehler überprüft. Beim Review Verfahren hat sich das Projektteam für den Walkthrough entschieden, welcher sich aufgrund niedriger Anzahl an Projektteilnehmer für dieses Projekt perfekt umsetzen ließ. Nach Kleuker (2019, S. 22) ist ein Walkthrough ein kürzeres Prüfverfahren. Eine Inspektion als Review Verfahren wäre aufgrund des

relativ kleinen Projektumfangs und aufgrund des kleinen Projektteams zu umfangreich und wurde daher nicht als Review Verfahren gewählt.

Im Rahmen dieses Walkthroughs nahm das Teammitglied, welches für die Erstellung zuständig war, die Rolle des Autors und des Moderators ein. Die anderen beiden Teammitgliedern agierten als Gutachter. Der Ablauf des Walkthroughs ist so geplant, dass sich die Gutachter zunächst vorbereiteten, indem sie den Code individuell überprüften. Nach dieser individuellen Überprüfung wurde der Code gemeinsam durchgegangen und diskutiert. Abschließend wurden die entdeckten Mängel im Rahmen der Nachbereitung beseitigt (Kleuker, 2019, S. 22-23).

Beim Feedback innerhalb des Reviews wurde darauf geachtet, dass das Schema des sogenannten *Feedback Burgers* eingehalten wird. Nach diesem Schema soll es einen positiven Einstieg mit Lob geben. Darauf folgt dann die eigentliche Kritik. Abgeschlossen wurde das Feedback dann wieder positiv mit einem Lob (Broy & Kuhrmann, 2021, S. 520).

Zusätzlich führte das Projektteam eine statische Codeanalyse aus, in der das System nicht ausgeführt wurde. Hierbei handelt es sich um einen White-Box-Test. Hierbei gilt es das System auf bestimmte Fehlertypen oder der Einhaltung von Codierungsrichtlinien zu untersuchen:

- „Fehlende Initialisierung von Variablen
- Fehlende Rückgaben in allen Pfaden einer Funktion
- Nicht-erreichbarer Code
- Fehlende oder falsche Typumwandlungen
- Unsichere Funktionssignaturen
- Mangelnde Einhaltung von Namenskonventionen und von Codierungsrichtlinien“ (Broy & Kuhrmann, 2021, S. 470-471)

4.4.7. Testfallerstellung

Ein Testprozess ist in fünf Schritte aufgeteilt und muss entsprechend organisiert und strukturiert werden (Broy & Kuhrmann, 2021, S. 474).

Geplant ist, dass alle notwendigen Tests nach der Fertigstellung des Prototyps am 10.11. 2022 vollzogen werden. Da es sich um eine Web-Anwendung handelt konnten sich die Projektmitglieder über einen Link Zugriff über die App verschaffen. Im Planungsschritt die Aktivitäten festgelegt, die zu Überprüfung des Systems benötigt werden (Broy & Kuhrmann, 2021, S. 475).

Bei der Testfallerstellung hat sich das Projektteam für eine anwendungsfallbasierte Testfallerstellung entschieden. Die Testfälle wurden aus Use-Cases abgeleitet, um die Anforderungen gründlich zu überprüfen zu können (Witte, 2018, S.93). Die Anzahl der Anwendungsfälle ergibt sich aus der Zahl der möglichen Ausgänge (Nutzungsprofil der Quiz-App). Kurzgefasst: Je mehr logische Pfade, desto mehr Testfälle müssen durchgeführt werden.

UseCase - Volumen = Anzahl UseCases / (Anzahl UseCases-Ausgänge) (Witte, 2018, S. 88).

Zur Testdurchführung traf sich das Projektteam in Microsoft Teams, um gemeinsam folgende Funktionen des Systems zu überprüfen:

- ein Student loggt sich in die Anwendung ein
- Studenten können sich registrieren
- ein Kurs auswählen und hinzufügen
- eine neue Frage für den Fragen-Pool einreichen
- Tutor kann diese für den Fragen-Pool freigegeben oder ablehnen. Anschließend soll diese Frage dann im Fragen-Pool enthalten sein.
- ein Student kann im „Duell-Modus“ spielen kann.
- Studenten können ihr Benutzerprofil einsehen und bearbeiten
- Kurse als Favoriten speichern
- Studierende könne ihre Fragen filtern
- Studierende können Fragen aus dem Fragenpool bewerten und kommentieren
- Der High Score der einzelnen Studenten aktualisiert sich bei jedem Spiel

Dabei wurden alle Anwendungsfallpfade getestet, die durchlaufen werden konnten. Dadurch war es dem Projektteam möglich eine 100% Funktionsüberdeckung zu erreichen. Mit der Testabdeckung konnte das Projektteam überprüfen, welche der Pfade durchlaufen werden konnte und welche nicht (Witte, 2018, S. 93).

Im Evaluationsschritt wurden die Tests ausgewertet und das Testprotokoll aus Tab. 5 ausgefüllt. Alle Testfälle konnten erfolgreich erfüllt werden mit Ausnahme der Registrierung von Studierenden. Da als Voraussetzung für die Anmeldung eine IU-E-Mail-Adresse benötigt wird, müssten sich die Studierenden diese, über den Outlook in myCampus registrieren. Unglücklicherweise wird die Anfrage vom IU-Server blockiert, sodass eine Verifizierung der E-Mail-Adresse nicht möglich war.

5. Entwicklung und technische Umsetzung der Quiz-App (MS4)

Wie viele moderne Webanwendungen wurde die Quiz-App als Single-Page-Applikation entwickelt. SPAs bestehen, im Gegensatz zu klassischen Webseiten, aus nur einem einzigen HTML-Dokument. Die Seiteninhalte werden dynamisch aktualisiert, üblicherweise als Reaktion auf Benutzeraktionen. Dies führt zu deutlich geringeren Wartezeiten und einem verbesserten Nutzererlebnis, da nur Teile der Präsentationslogik nachgeladen werden müssen. Eine weitere wichtige Eigenschaft von SPAs ist die Einstufung als „offline-friendly“:

Durch die vollständige Umsetzung der Präsentationsschicht auf dem Client ist es möglich, mithilfe der Web-Storage-Funktion persistente Zwischenspeicher anzulegen. Für den Fall, dass der Nutzdatenserver nicht erreichbar ist, können Daten aus dem Zwischenspeicher verwendet werden. Die Anwendung kann somit weiterhin auf dem Client betrieben werden, ohne dass eine Verbindung zum Server bestehen muss. (Wikimedia Foundation Inc., 2022)

Zur Realisierung der Offline-Verfügbarkeit von Web-Apps werden in der Regel *Service Worker*¹ zu Hilfe genommen. Da im Rahmen der Entwicklung eines lauffähigen Prototyps auf die Bereitstellung von Offline-Daten verzichtet wurde, soll an dieser Stelle nicht genauer auf den Implementierungsprozess einer Offline-Funktion eingegangen werden.

5.1. Vorbereitung

Bevor mit der Implementierung eines lauffähigen Prototyps begonnen werden konnte, musste sich das Projektteam mit der Frage nach den zu verwendenden Cloud-Computing Diensten und den passenden Anbietern beschäftigen. Für diese Projektarbeit fanden ausschließlich die kostenlosen Kontingente der diversen Cloud-Computing Anbieter Berücksichtigung. Nach umfassender Recherche und projektinternen Absprachen fiel die Wahl auf Googles Entwicklungs-Plattform *Firebase*. Genauer wurden die in [Kapitel 5.3](#) beschriebenen Firebase-Dienste *Authentication*, *Cloud Firestore* und *App Check*, sowie der in [Kapitel 5.5](#) beschriebene Firebase-Dienst *Hosting* verwendet. Ein Grund für diese Entscheidung war das Firebase JavaScript SDK, durch das die Firebase Backend-Dienste im Client angesprochen werden können. So konnte die gesamte Programmlogik clientseitig implementiert und auf eine typische Einrichtung eines Anwendungsservers verzichtet werden. Hierdurch konnte der Entwicklungsprozess vereinfacht und wichtige Projektressourcen eingespart werden.

5.2. Entwicklung (Frontend)

In den nachfolgenden Unterkapiteln werden die einzelnen Oberflächen (Seiten) der Quiz-App sowie deren Funktionsweise beschrieben.

5.2.1. Startseite

Ein Anmeldeportal dient als Startseite der Quiz-App. Hier können sich die Benutzer registrieren, anmelden und die Zurücksetzung ihres Benutzerpassworts beantragen. Darüber hinaus befindet sich im Footer ein Link zur Hilfeseite, auf der neue Benutzer Informationen über den Anmeldeprozess finden können.

Für die Registrierung bei der Quiz-App wird eine gültige E-Mail-Adresse der IU benötigt. Folgende Endungen werden akzeptiert:

- iubh-fernstudium.de
- iubh.de
- iu.org

¹ „**Service Worker** bezeichnet eine moderne Browsertechnologie, die mittels JavaScript einen Proxy zwischen dem Webbrowser und dem Server bereitstellt ... Der bekannteste Einsatzzweck ist es, bestimmte Seiten auch offline verfügbar zu machen. Daneben bieten Service Worker auch die Grundlage für Push-Benachrichtigungen, progressive Web-Apps und weitere Verwendungen“ (Wikimedia Foundation Inc., 2021).

Letztgenannte Endung ist dem Lehrpersonal und Mitarbeitern der IU vorbehalten, weshalb Benutzern mit dieser E-Mail-Endung automatisch die Berechtigungsstufe Tutor zugeteilt wird. Die einzelnen Berechtigungsstufen werden in [Kapitel 5.3.1](#) detailliert. Nach erfolgter Registrierung muss die gewählte E-Mail-Adresse über einen Link in einer automatisch verschickten Bestätigungsmail verifiziert werden. Erst nach der Verifizierung können die Benutzer auf die meisten Funktionen der App zugreifen. Um modernen Sicherheitsstandards gerecht zu werden, erfordert das Benutzerkennwort eine Mindestlänge von acht Zeichen sowie das Vorhandensein eines Klein- oder Großbuchstaben (A-Z) und einer Zahl (0-9).

5.2.2. Hauptseite (Dashboard)

Bereits angemeldete Benutzer werden beim Aufruf der Quiz-App direkt zur Hauptseite, dem Dashboard, weitergeleitet. Eine neuerliche Anmeldung bei jedem Seitenaufruf ist nicht erforderlich. Vorausgesetzt, der Benutzer hat die Verwendung von essenziellen und funktionalen Cookies in den Browsereinstellungen zugelassen. In den meisten Internet-Browsers sind Cookies standardmäßig aktiviert.

Auf dem Dashboard werden die verfügbaren Kurse im Reiter „Alle Kurse“ aufgelistet. Den Benutzern steht es frei, Kurse als Favoriten zu markieren. Diese Kurse sind anschließend im Reiter „Meine Kurse“ zu finden. Nach einem Klick auf den gewählten Kurs werden die Benutzer zur entsprechenden Kursseite weitergeleitet.

5.2.3. Kursseite

Jede Kursseite ist in drei feste Bereiche unterteilt. Zunächst ein allgemeiner Bereich, in dem die Bezeichnung des Kurses, die Kursnummer, sowie Name und Kontaktinformationen des zuständigen Tutors zu finden sind.

Im Bereich „Challenge“ finden die Studenten eine Übersicht der begonnen und zuletzt beendeten Quizduelle gegen ihre Kommilitonen. Im Abschnitt „Persönliche Statistik“ werden die Anzahl der beendeten Quizduelle und der beantworteten Fragen, Siege, Niederlagen und weitere Informationen angezeigt. Studenten können des Weiteren neue Quizfragen erstellen, die anschließend vom zuständigen Tutor bestätigt werden müssen. Durch einen Klick auf den prominent platzierten Spielebutton wird eine neue Quizrunde gestartet. Hierbei wird zunächst geprüft, ob der Student zuvor ein Spiel unterbrochen hat. Ist dies der Fall, wird das Spiel fortgesetzt. Andernfalls wird in der Datenbank nach einem offenen Spiel gesucht, dem der Student als zweiter Spieler beitreten kann. Wurde kein offenes Spiel gefunden, wird ein neues erstellt. Anschließend wird zur Spieleseite gewechselt, die im nächsten Unterkapitel vorgestellt wird.

Der Bereich „Community Fragen“ beinhaltet eine Liste von offenen Fragen und Aufgabenstellungen samt Musterlösungen. Der Fragenkatalog kann von allen Kursteilnehmern gepflegt und erweitert werden. Im Gegensatz zu Quizfragen werden offene Fragen nach der Erstellung direkt in den Fragenkatalog aufgenommen und müssen nicht zuvor durch den Tutor bestätigt werden. Jede Frage

bzw. Aufgabenstellung kann von den anderen Kursteilnehmern hinsichtlich Nützlichkeit und Schwierigkeit bewertet werden. Die Liste der Fragen kann nach diversen Merkmalen, z.B. dem Erstellungsdatum oder dem passenden Kapitel im Studienskript, sortiert werden. Ein Suchfeld erleichtert es den Studenten, eine bestimmte Aufgabenstellung zu finden.

Ein vierter Bereich, welcher sich unter dem allgemeinen Bereich befindet, ist nur für Tutoren und Administratoren sichtbar. Hier werden die von den Studenten neu erstellten Quizfragen in einem Slider angezeigt und können bestätigt oder abgelehnt werden.

5.2.4. Quizseite

Quizduelle finden nicht in Echtzeit statt. Spieler haben die Möglichkeit die Runde zu unterbrechen und zu einem späteren Zeitpunkt fortzusetzen. Ein weiteres Spiel kann erst dann begonnen werden, wenn der Spieler die Runde beendet, d.h. alle fünf Fragen beantwortet, hat.

Die Oberfläche setzt sich zusammen aus der Quizfrage, vier Schaltflächen mit den Antwortmöglichkeiten, einem Timer, den Namen der Spieler und einem Indikator der bisher richtig bzw. falsch beantworteten Fragen. Die Zeit zur Beantwortung einer Frage beträgt stets 20 Sekunden. Wurde die Frage nicht rechtzeitig beantwortet, wird sie automatisch als falsch markiert.

5.2.5. Sonstige Seiten

Zu den *Benutzereinstellungen* gelangen die Benutzer durch einen Klick auf das Profilbild in der Navigationsleiste. Hier können der Anzeigename und das Profilbild geändert werden. Zusätzlich werden nutzerspezifische Informationen wie E-Mail-Adresse, Datum der Registrierung und Zeitpunkt der letzten Anmeldung angezeigt. Am Seitenende befindet sich ein Logout-Button.

Die *Hilfeseite* ist über die Navigationsleiste sowie den Footer zugänglich. Diese Oberfläche beinhaltet die wichtigsten Fragen und Antworten rund um die Benutzung der Quiz-App. Ein Suchfeld erleichtert es den Benutzern, die gesuchten Informationen zu finden.

Darüber hinaus befindet sich in der Navigationsleiste eine Schaltfläche mit der Aufschrift *Community*. Über diesen Link würden die Studierenden in der produktionsreifen App zu einem externen Forum gelangen, in dem sie sich über Kursinhalte und die Web-App austauschen könnten. Anstelle eines Forums wäre auch eine Integration mit den Course Feeds an der IU denkbar.

5.3. Entwicklung (Backend)

5.3.1. Berechtigungsstufen

Zur Steuerung der Zugriffsrechte wurden verschiedene Berechtigungsstufen für Studenten, Tutoren und Administratoren erstellt. Die Berechtigungsstufe entscheidet darüber, welche Inhalte den Benutzern in der Web-App präsentiert werden. Beispielsweise können Tutoren die von Studenten erstellten Quizfragen einsehen und bestätigen oder ablehnen. Administratoren können darüber hinaus neue Kurse erstellen und bestehende Kurse löschen. Bei der Registrierung erhalten Benutzer

mit der Adressendung „@iu.org“ automatisch Tutorenrechte erteilt. Die Vergabe von Administratorenrechten kann ausschließlich in der Firebase Web-Konsole erfolgen.

5.3.2. Benutzerverwaltung mit Authentication

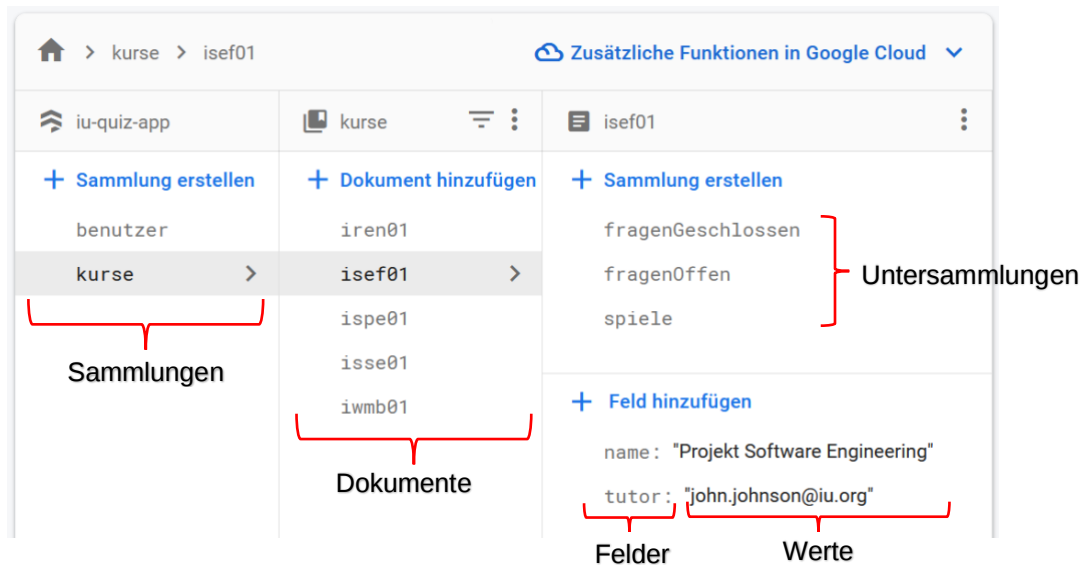
Die Erstellung, Verwaltung und Authentifizierung von Benutzern läuft innerhalb der Firebase-Plattform über den Dienst *Authentication*, der sich eng in die anderen Firebase-Dienste integrieren lässt. Der Dienst übernimmt auch den Versand von E-Mails, beispielsweise nach erfolgter Registrierung oder nach der Beantragung einer Passwortrücksetzung.

Um einen Benutzer bei der App anzumelden, werden zunächst die Authentifizierungsdaten vom Benutzer abgefragt (Google LLC, 2022a). Im Fall dieser Quiz-App sind das die E-Mail-Adresse und das Kennwort des Benutzers. Anschließend werden diese Anmeldeinformationen an das Firebase Authentication SDK übergeben und verschlüsselt an die Firebase-Server übermittelt. Die Firebase Backend-Dienste überprüfen dann die Anmeldeinformationen und senden eine Antwort zurück an den Client. Nach einer erfolgreichen Anmeldung kann auf die grundlegenden Profilinformationen des Benutzers zugegriffen und der Zugriff des Benutzers auf Daten gesteuert werden, die in anderen Firebase-Produkten gespeichert sind.

5.3.3. Datenverwaltung mit Cloud Firestore

Cloud Firestore ist eine in der Cloud gehostete NoSQL-Datenbank, auf die Apple-, Android- und Web-Apps direkt über native SDKs zugreifen können (Google LLC, 2022b). Gemäß dem NoSQL-Datenmodell von Cloud Firestore werden Daten in Dokumenten (Documents) gespeichert, die Felder enthalten, denen Werte zugeordnet sind. Diese Dokumente werden in Sammlungen (Collections) gespeichert, die verwendet werden können, um Daten zu organisieren und Abfragen zu erstellen. Dokumente unterstützen verschiedenste Datentypen, von einfachen Zeichenfolgen und Zahlen bis hin zu komplexen, verschachtelten Objekten. Es können auch Untersammlungen innerhalb von Dokumenten erstellt und hierarchische Datenstrukturen aufgebaut werden, die sich skalieren lassen, wenn die Datenbank wächst.

Abbildung 1: Cloud Firestore



Quelle: Übernommen aus Cloud Firestore Web-Konsole

Die Datenabfrage in Cloud Firestore ist ausdrucksstark, effizient und flexibel (Google LLC, 2022b). Es können flache Abfragen erstellt werden, um Daten auf Dokumentebene abzurufen, ohne die gesamte Sammlung oder verschachtelte Untersammlungen abrufen zu müssen. Abfragen können Sortierungen, Filterungen und Begrenzungen hinzugefügt werden, um die zurückgegebenen Ergebnisse einschränken und paginieren zu können.

Der Zugriff auf Daten in Cloud Firestore kann mit dem in [Unterkapitel 5.3.2](#) beschriebenen Dienst Firebase Authentication sowie Cloud Firestore-Sicherheitsregeln gesteuert und beschränkt werden.

5.3.4. Datenbanksicherheit

Um die in der Datenbank gespeicherten Daten vor Missbrauch zu schützen, wurde der Dienst *Firebase App Check* verwendet. App Check verhindert den Zugriff nicht autorisierter Clients auf die genannten Backend-Ressourcen (Google LLC, 2022c). Mit App Check verwenden Geräte, auf denen die App ausgeführt wird, einen App- oder Gerätenachweisanbieter, der eines oder beide der folgenden Gegebenheiten bestätigt:

- Anfragen stammen von der authentischen App
- Anfragen stammen von einem authentischen, nicht manipulierten Gerät

Diese Bescheinigung wird jeder Anfrage angehängt, die die Web-App an die Firebase APIs sendet. Anfragen von Clients ohne gültige Bescheinigung werden abgelehnt, ebenso wie alle Anfragen, die von einer App oder Plattform stammen, die nicht manuell autorisiert wurde.

5.4. Testen

Nach Abschluss sämtlicher Tests hat das Projektteam das in der Planungsphase erstellte Testprotokoll ausgefüllt, das im Anhang 6: Testprotokoll zu finden ist.

5.5. Bereitstellung

5.5.1. Web-App Hosting

Der Zugriff auf die Web-App per Internet-Browser wurde durch den Dienst *Firebase Hosting* ermöglicht. Firebase Hosting bietet schnelles und sicheres Hosting für Web-Apps, sowie statische und dynamische Inhalte und Microservices (Google LLC., 2022d). Alle Inhalte werden auf Knotenpunkten eines globalen CDN² zwischengespeichert, wodurch eine hohe Verfügbarkeit und geringe Zugriffszeiten sichergestellt werden können. Jedes Firebase-Projekt erhält kostenlose Subdomains in den Domains *web.app* und *firebaseapp.com*. Darüber hinaus werden für die zur Verfügung gestellten Domains automatisch SSL-Zertifikate ausgestellt, damit alle Inhalte sicher bereitgestellt werden können.

Mithilfe des Firebase CLI können Web-App-Updates durch die Ausführung eines einzigen Konsolenbefehls hochgeladen und automatisch veröffentlicht werden. In der Firebase Web-Konsole können wiederum Rollback-Aktionen durchgeführt werden, um schnell und einfach zu einer früheren App-Version zurückkehren zu können. Das Firebase CLI stellt weitere nützliche Entwicklungstools bereit, die jedoch im Rahmen dieses Projekts keine Anwendung gefunden haben.

5.5.2. Benutzervideo

Wie im Dokumentationskonzept, [Kapitel 4.1.6](#), erläutert, hat das Projektteam bewusst auf die Erstellung eines klassischen, textbasierten Benutzerhandbuchs verzichtet. Stattdessen wurde ein How-to-Video erstellt, in dem den Benutzern die Funktionen der Quiz-App vorgestellt und erklärt werden. Zusätzlich wurde die in [Unterkapitel 5.2.4](#) erwähnte FAQ-Seite implementiert. Usability-Tests³ haben gezeigt, dass insbesondere neue Benutzer lieber Anleitungsvideos ansehen und Textanleitungen überspringen, während fortgeschrittene Benutzer es bevorzugen, im Text nach Antworten auf ihre spezifischen Fragen zu suchen (Johnson, 2011).

6. Screencast (MS5)

Im letzten Meilenstein (MS5) dem Screencast wurden, abschließend alle erreichten Ergebnisse der Meilensteine 1 – 5 in einer 20-minütigen Präsentation vorgestellt und festgehalten. Der Inhalt umfasst den Soll-Ist-Vergleich, inwieweit die Anforderungen und Aufgaben an der Quiz-App gelöst wurden, Welche Ziele erreicht wurden und welche nicht sowie die Effizienz bei der Durchführung des Projekts (Esch et al., 2016, S.42). Nach MacLeod et al. (2017, S.1-2) können Screencasts dazu verwendet werden, wichtige Informationen zur Anleitung eines Systems zu präsentieren, welche sich

² "Ein Content Delivery Network (CDN) ist eine geografisch verteilte Gruppe von Servern, die zusammenarbeiten, um Internetinhalte schnell bereitzustellen" (Cloudflare, Inc., 2022).

³ "Usability testing refers to evaluating a product or service by testing it with representative users ... The goal is to identify any usability problems, collect qualitative and quantitative data and determine the participant's satisfaction with the product" (Usability.gov, 2013).

nicht beschreiben lassen, indem der Entwickler seinen Bildschirm teilt, während er die Funktionen der Software vorführt.

In diesem Zusammenhang wurde in den wöchentlichen Meetings des Projektteams ein Konzept entwickelt. Mithilfe von Microsoft PowerPoint wurden Folien erstellt, die die Inhalte übersichtlich zusammenstellen. Jedes Teammitglied bekam Bereiche zugeteilt, die im Video präsentiert wurden. Die Inhalte der einzelnen Folien wurden dank der integrierten Aufzeichnungsfunktion von PowerPoint aufgenommen. Anschließend wurde das Video aus den einzelnen Clips mithilfe des kostenlosen Tools Vimeo geschnitten und erstellt.

Die Folien umfassen die folgenden Inhalte:

- Vorstellung der Meilensteinplanung und Projektverlauf
- Vorstellung der Ergebnisse zu den einzelnen Meilensteinen:
- MS1: Projektkonfiguration
 - Vorstellung des Teams
 - Kommunikation
- MS2: Projektvideo bereitgestellt
- MS3: Dokumentationskonzept bereitgestellt
 - Entity-Relationship-Diagramm
 - Benutzung der Anwendung
- MS4: Zentrale Ergebnisartefakte
- Präsentation der IU-Quiz-App
 - Ergebnisse

Neben der Präsentation der einzelnen Meilensteine und deren Verlauf, wurde auch die Anwendung inklusive aller Funktionen aus Sicht der Studierenden und des Tutors vorgeführt. Hierzu wurde die kostenlose Version der Software Screencast-O-Matic verwendet. Der fertige Screencast wurde dem Tutor im Anschluss in der Redmine-Plattform bereitgestellt.

7. Schlussteil: Auswertung und Reflexion der Projektergebnisse

Am Ende des Projektes konnte ein Prototyp eines kooperativen und kollaborativen Online-Quizsystems erstellt werden. Der Quellcode ist in Anhang 7: Link zum Github Repository verlinkt und die Anwendung ist im Anhang 8: Adresse der Webanwendung verlinkt. Das System kann Studierende bei der Festigung der Lerninhalte und Vorbereitung auf die Klausur unterstützen kann. Es ist als Web-Anwendung programmiert, sodass es von überall über einen Browser bedient werden kann. In der App gibt es zwei verschiedene Spielmodi. Im Challenge-Modus können die Studierenden gegeneinander antreten und so ihr Wissen auf die Probe stellen. Grundlage dieses Modus ist ein Fragenpool mit Fragen zu den Kursinhalten. Zu den Fragen werden immer vier Antwortmöglichkeiten vorgegeben von denen eine richtig ist. Für die Pflege des Fragenkatalog wurde vom Entwicklerteam überlegt, dass die Studierenden selbst Fragen erstellen können und einreichen können. Diese können dann vom Tutor geprüft und freigegeben werden. Auf diese Art kommen immer wieder neue Fragen in den Fragenkatalog und die Fragen sind immer auf einem aktuellen Stand. Im Kooperativen haben die Studierenden die Möglichkeit offene Fragen zu stellen, mit denen die anderen Studierenden ihr Wissen überprüfen können. Neben den beiden Spielmodi wird die Anwendung durch eine Loginseite abgesichert. Benutzerkonten können nur mit IU-Mailadresse erstellt werden. Nach dem Login finden die Studierenden eine Kursauswahl, bei der Kurse auch als Favoriten gespeichert werden können. Die Daten der Anwendung werden persistent in der Firebase-Datenbank gespeichert. Als Feinschliff wurde die App an das IU-Design angepasst und eine FAQ-Seite mit den wichtigsten Fragen und Antworten erstellt. Es lässt sich daher festhalten, dass das Ergebnis das Projektziel *Konzeption und prototypische Umsetzung eines kooperativen und kollaborativen Online-Quizsystems* erfüllt. Wie im Testprotokoll in Anhang 6: Testprotokoll zu sehen ist, konnten grundsätzlich alle Anforderungen, die zu Beginn des Projekts definiert wurden, umgesetzt werden. Im Projektverlauf ist das Projektteam auf einige Herausforderungen gestoßen. Diese haben sich auf technische aber auch auf organisatorische Punkte bezogen. Durch einen guten und regelmäßigen Austausch, vor allem in den wöchentlichen Meetings, konnte das Projektteam aber alle Herausforderungen meistern.

Aus dem Projektverlauf konnte das Projektteam einige Erkenntnisse gewinnen, die auch für zukünftige Projekte sehr wichtig sein werden. Es hat sich gezeigt, dass vor allem die Meilensteine 1 und 3 (Projektkonfiguration und Dokumentationskonzept) für das Projekt und den Projektverlauf sehr wichtig waren. Durch die in diesen Meilensteinen erarbeiteten Inhalte, hat das Projekt einen sehr konkreten Rahmen und Ablauf bekommen. Dadurch wurde der Implementierungsprozess für das Projektteam sehr erleichtert, weil das Projektteam genau wusste, welche Anforderungen umzusetzen sind, wer welche Rolle bei der Umsetzung hat und wie die Implementierungen getestet werden können. Eine weitere wichtige Erkenntnis aus dem Projektverlauf ist, dass der Ausfall von Teammitgliedern einen großen Einfluss auf den Projektverlauf haben kann und dass dieser Fall im Rahmen der Projektplanung und des Risikomanagements auf jeden Fall beachtet werden muss, damit das Projektziel nicht gefährdet ist. In diesem Projekt haben vor allem der im Rahmen des

Risikomanagements eingeplante Zeitpuffer, der regelmäßige Austausch und die gute Teamarbeit dazu beigetragen, dass krankheitsbedingte Ausfälle gut aufgefangen werden konnten und das Projektziel nicht gefährdet wurde. Dem Projektteam war es daher möglich, trotz eines straffen Zeitplans alle Termine im Großen und Ganzen einzuhalten. Zwar ist es vorgekommen, dass einzelne Meilensteine ein paar Tage verspätet abgeschlossen wurden, dafür konnten andere Meilensteine früher abgeschlossen oder auf den Zeitpuffer zurückgegriffen werden. Zu großen Verzögerungen ist es bei der Bearbeitung des Projekts daher nicht gekommen. Außerdem kann das Projektteam aus dem Projekt den Schluss ziehen, dass der Erfolg eines Projektes auch sehr stark vom Projektteam und einem regelmäßigen Austausch innerhalb des Projektteams abhängig ist. Das eher kleine Projektteam konnte trotz seiner geringen Anzahl an beteiligten Personen, alle Aufgaben gut meistern. Dies lag unter anderem daran, dass das Projektteam hat auf eine klare Aufgaben- und Rollenverteilung verzichtet hat. Jedes Teammitglied konnte somit gleichberechtigt am Projekt mitarbeiten und sich einbringen. Alle Zwischenergebnisse wurden in den wöchentlichen Meetings besprochen und notwendige Verbesserungen und Optimierungen durchgeführt. Diese Art der Projektarbeit und Kommunikation hat sehr gut funktioniert. Jedes Mitglied konnte seine Kreativität einbringen, wodurch deutlich mehr Ideen in das Projekt eingeflossen und somit das Ergebnis verbessert wurde. So entstanden aus individuellen Ideen, Lösungen und Optimierungen. Zusätzlich konnte ein hohes Maß an Qualität sichergestellt werden, da jeder die Ergebnisse des anderen begutachtete. Gerade wegen der lockeren Rollenverteilung, war die Zusammenarbeit harmonisch und es konnte respektvoll miteinander kommuniziert werden. Unstimmigkeiten wurden direkt angesprochen und geklärt.

Da es sich bei der App noch um eine prototypische Umsetzung handelt, müssten vor einer wirklichen Realisierung noch einige Punkte beachtet werden. So ist die Leistung der App aktuell auf das kostenlose Kontingent des Cloudanbieters beschränkt. Um eine gute Performance der App auch bei vielen gleichzeitigen Nutzern gewährleisten zu können, müsste die App über dieses kostenlose Kontingent hinaus skaliert werden. Außerdem müssen rechtliche Rahmenbedingungen beachtet werden, für den Fall, dass sich die IU Internationale Hochschule dazu entscheidet, das Quiz-System einzusetzen. Dazu gehören Names- und Markenrecht, Impressumspflicht, Nutzungsbedingungen inklusive Leistungsangebot und datenschutzrechtliche Fragen, AGBs, Datenschutzerklärung und Cookie-Banner.

8. Literaturverzeichnis

- Attardi, J. (2020). *Modern CSS. Master the Key Concepts of CSS for Modern Web Development*. Billerica, MA, USA: Apress.
- Bewersdorff, J. (2018). *Objektorientierte Programmierung mit JavaScript. Direktstart für Einsteiger*. Wiesbaden: Springer Vieweg.
- Broy, M., & Kuhrmann, M. (2021). *Vorgehensmodelle in der Softwareentwicklung*. Springer: Berlin, Heidelberg.
- Cloudflare, Inc. (2022). *Was ist ein CDN?* Von <https://www.cloudflare.com/de-de/learning/cdn/what-is-a-cdn/> abgerufen
- Digital Guide Ionos. (10. Mai 2019). Von <https://www.ionos.com/digitalguide/websites/web-development/popular-javascript-frameworks-and-libraries/> abgerufen
- Drews, G., Hillebrand, N., Kärner, M., Peipe, S., & Rohrschneider, U. (2021). *Praxishandbuch Projektmanagement - inkl. Arbeitshilfen online*. Freiburg: Haufe.
- Eckert, C. (2018). *IT-Sicherheit: Konzepte – Verfahren – Protokolle*. Oldenburg: De Gruyter.
- Galster, M., Gilson, F., & Georis, F. (2019). *Welche Qualitätsmerkmale finden wir in Product Backlogs? Eine Perspektive des maschinellen Lernens*. Switzerland: Springer International Publishing.
- Google LLC. (2022a). *Firebase Authentication*. Von <https://firebase.google.com/docs/auth> abgerufen
- Google LLC. (2022b). *Cloud Firestore*. Von <https://firebase.google.com/docs/firestore> abgerufen
- Google LLC. (2022c). *Firebase App Check*. Von <https://firebase.google.com/docs/app-check> abgerufen
- Google LLC. (2022d). *Firebase Hosting*. Von <https://firebase.google.com/docs/hosting> abgerufen
- Hoffmann, W. (2017). *Risikomanagement : Kurzanleitung Heft 4*. Berlin/ Heidelberg: Springer.
- Javorsky, A. (09. Januar 2019). Den Projektauftrag klären. 4 Kernfragen, die ein Projektauftrag beantworten muss. *Projekt Magazin*(01), 1-17.
- Johnson, T. (2011). *A Few Notes from Usability Testing: Video Tutorials Get Watched, Text Gets Skipped*. Von <https://idratherbewriting.com/2011/07/22/a-few-notes-from-usability-testing-video-tutorials-get-watched-text-gets-skipped/> abgerufen
- Kuster, J., Bachmann, C., Hubmann, M., R., L., & Schneider, P. (2022). *Handbuch Projektmanagement : Agil – Klassisch – Hybrid: (5. Aufl.)*. Deutschland: Springer Gabler.

- Lübbert, C., & Zeh, T. (2022). Skizze eines Verfahrens zur Erstellung von Ontologien mittels Formaler Begriffsanalyse. In *Informatik Spektrum* 45 (S. 3–12). Springer.
- Linz, T. (2017). *Testen in Scrum-Projekten. Leitfaden für Softwarequalität in der agilen Welt : Aus- und Weiterbildung zum ISTQB® Certified Agile Tester – Foundation Extension*. dpunkt.verlag.
- MacLeod, L., Bergen, A., & Storey, M. (2017). Dokumentieren und Teilen von Softwarewissen mithilfe von Screencasts. *Empir Software Eng* 22, 1478–1507.
- Nyamsi, E. A. (2020). *IT-Lösungen auf Basis von SysML und UML Anwendungsentwicklung mit Eclipse UML Designer und Eclipse Papyrus*. Wiesbaden: Springer Vieweg.
- Patzak, G. &. (2018). *Projektmanagement: Projekte, Projektportfolios, Programme und projektorientierte Unternehmen: Vol. 7., aktualisierte Auflage*. Linde Verlag.
- Robbins, J. N. (2010). *HTML & XHTML kurz & gut*. Köln: O'Reilly Verlag.
- Setia, V. (11. August 2020). *MVVM-Architektur. C#-Ecke*. Von <https://www.c-sharpcorner.com/article/mvvm-architecture/> abgerufen
- Sole, A. D. (2021). *Visual Studio Code Distilled: Evolved Code Editing for Windows, macOS, and Linux*. Cremona, Italy: Apress.
- Speight, A. (2021). *Visual Studio® Code für Python-Programmierer® – Titelseite*. John Wiley & Söhne. Von <https://app.knovel.com/hotlink/pdf/id:kt012XQXG2/visual-studio-code-python/title-page> abgerufen
- Stoyko, T. (4. Februar 2022). *Incora*. Von <https://incora.software/insights/react-vs-angular-vs-vue-the-main-differences-and-use-cases/55> abgerufen
- Tiemeyer, E. (2014). *Handbuch IT-Projektmanagement: Vorgehensmodelle, Managementinstrumente, Good Practices*. München: Carl Hanser Verlag GmbH & Co. KG.
- Timinger, H. (2017). *Modernes Projektmanagement : Mit traditionellem, agilem und hybridem Vorgehen zum Erfolg* (1. Auflage Ausg.). Weinheim: Wiley-VCH.
- Trempe, H. (2022). *Agile objektorientierte Anforderungsanalyse Planen – Ermitteln – Analysieren – Modellieren – Dokumentieren – Prüfen*. Wiesbaden: Springer Vieweg.
- Usability.gov. (2013). *Usability Testing*. Von <https://www.usability.gov/how-to-and-tools/methods/usability-testing.html> abgerufen
- Vijayan, J. (2018). Google Updates Cloud Firestore NoSQL Database Beta for GCP. *EWeek*, S. 6-2.
- Voigt, P. D. (2018). *Qualitätssicherung. Definition: Was ist "Qualitätssicherung"?* . Von <https://wirtschaftslexikon.gabler.de/definition/qualitaetssicherung-44396> abgerufen

- Walter, T. (2008). *Kompendium der Web-Programmierung, Dynamische Websites*. Springer Verlag.
- Wikimedia Foundation Inc. (2021). *Service Worker*. Von https://de.wikipedia.org/wiki/Service_Worker abgerufen
- Wikimedia Foundation Inc. (2022). *Single-Page-Webanwendung*. Von <https://de.wikipedia.org/wiki/Single-Page-Webanwendung> abgerufen
- Witte, F. (2018). *Metriken für das Testreporting Analyse und Reporting für wirkungsvolles Testmanagement*. Wiesbaden: Springer Vieweg.
- Xue-Geng, S. -r., Yao, J. -y., & X. -q. Shi. (2022). "Event-B Modeling for Use Case Diagrams and Application in Immune System Model". *7th International Conference on Computational Intelligence and Applications (ICCIA)*, S. 6-10.

9. Verzeichnis der Anhänge

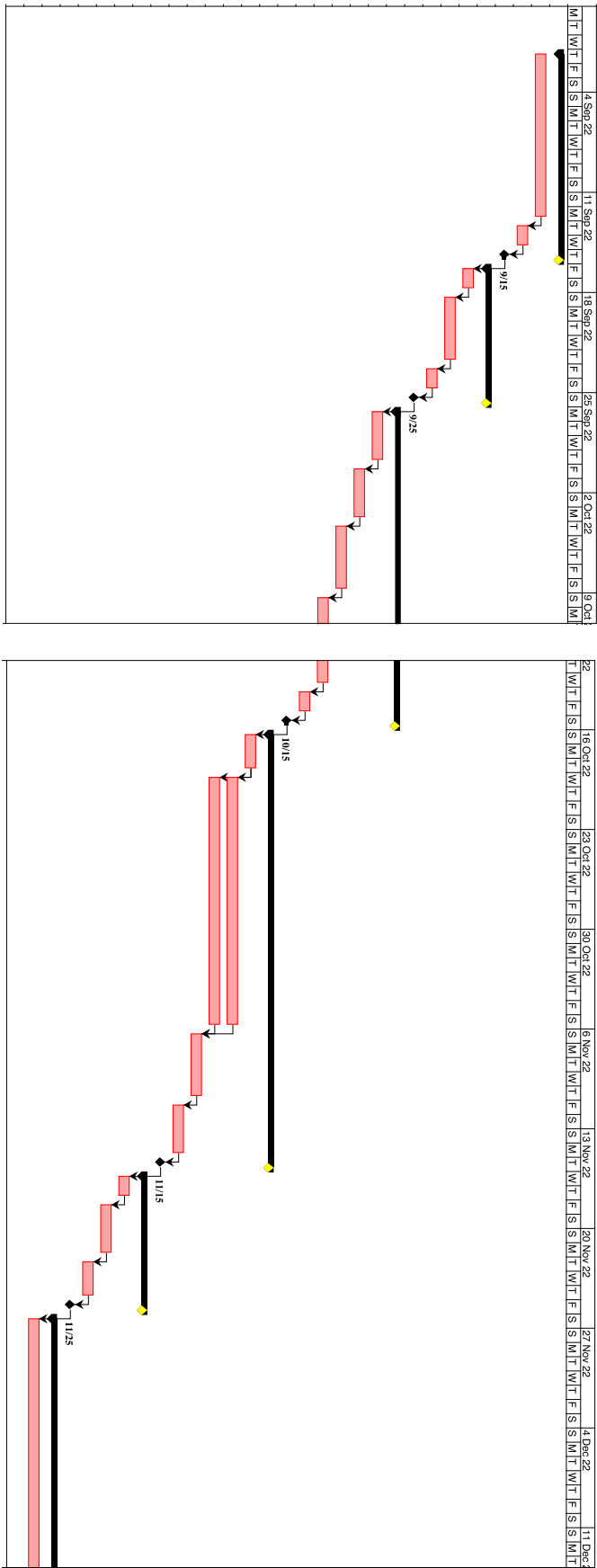
Anhang 1: Teilaufgaben und Arbeitspakete	43
Anhang 2: Gantt-Diagramm.....	44
Anhang 3: Use-Case-Diagramm	45
Anhang 4: UML-Zustandsdiagramm.....	46
Anhang 5: Entity-Relationship-Diagramm	47
Anhang 6: Testprotokoll	48
Anhang 7: Link zum Github Repository	50
Anhang 8: Adresse der Webanwendung.....	50

Anhang 1: Teilaufgaben und Arbeitspakete

		Name	Duration	Critical	Start	Finish
1		Projektkonfiguration erstellen	15 days?	☒	9/1/22 8:00 AM	9/15/22 5:00 PM
2		Projektkonfiguration erstellen	12 days?	☒	9/1/22 8:00 AM	9/12/22 5:00 PM
3		Projektkonfiguration prüfen und freigeben	2 days?	☒	9/13/22 8:00 AM	9/14/22 5:00 PM
4		Projektkonfiguration abgeschlossen und bereit...	1 day?	☒	9/15/22 8:00 AM	9/15/22 5:00 PM
5		Projektvideo	10 days?	☒	9/16/22 8:00 AM	9/25/22 5:00 PM
6		Inhalt definieren	2 days?	☒	9/16/22 8:00 AM	9/17/22 5:00 PM
7		Video aufzeichnen	5 days?	☒	9/18/22 8:00 AM	9/22/22 5:00 PM
8		Video schneiden	2 days?	☒	9/23/22 8:00 AM	9/24/22 5:00 PM
9		Projektvideo bereitgestellt	1 day?	☒	9/25/22 8:00 AM	9/25/22 5:00 PM
10		Dokumentationskonzept erstellen	20 days?	☒	9/26/22 8:00 AM	10/15/22 5:00 PM
11		Beschreibung der Projektdokumente	4 days?	☒	9/26/22 8:00 AM	9/29/22 5:00 PM
12		Implementierungsprozess beschreiben	4 days?	☒	9/30/22 8:00 AM	10/3/22 5:00 PM
13		Implementierung beschreiben	5 days?	☒	10/4/22 8:00 AM	10/8/22 5:00 PM
14		Qualitätsmanagement beschreiben	4 days?	☒	10/9/22 8:00 AM	10/12/22 5:00 PM
15		Review	2 days?	☒	10/13/22 8:00 AM	10/14/22 5:00 PM
16		Dokumentationskonzept bereitgestellt	1 day?	☒	10/15/22 8:00 AM	10/15/22 5:00 PM
17		Zentrale Ergebnisaufnahme erstellen	31 days?	☒	10/16/22 8:00 AM	11/15/22 5:00 PM
18		Infrastruktur bereitstellen	3 days?	☒	10/16/22 8:00 AM	10/18/22 5:00 PM
19		Front-End implementieren	18 days?	☒	10/19/22 8:00 AM	11/5/22 5:00 PM
20		Back-End implementieren	18 days?	☒	10/19/22 8:00 AM	11/5/22 5:00 PM
21		Systemtests/ Review durchführen	5 days?	☒	11/6/22 8:00 AM	11/10/22 5:00 PM
22		Ergebnisdokumentation erstellen	4 days?	☒	11/11/22 8:00 AM	11/14/22 5:00 PM
23		Zentrale Ergebnisaufnahme bereitgestellt	1 day?	☒	11/15/22 8:00 AM	11/15/22 5:00 PM
24		Ergebnispräsentation als Screencast	10 days?	☒	11/16/22 8:00 AM	11/25/22 5:00 PM
25		Inhalte definieren	2 days?	☒	11/16/22 8:00 AM	11/17/22 5:00 PM
26		Screencast aufzeichnen	4 days?	☒	11/18/22 8:00 AM	11/21/22 5:00 PM
27		Nachbearbeitung/ Review	3 days?	☒	11/22/22 8:00 AM	11/24/22 5:00 PM
28		Ergebnispräsentation als Screencast fertigstellen	1 day?	☒	11/25/22 8:00 AM	11/25/22 5:00 PM
29		Finale Ergebnisse, Dokumentation und Pr...	20 days?	☒	11/26/22 8:00 AM	12/15/22 5:00 PM
30		Projektbericht erstellen	19 days?	☒	11/26/22 8:00 AM	12/14/22 5:00 PM
31		Finale Ergebnisse, Dokumentation und Projektb...	1 day?	☒	12/15/22 8:00 AM	12/15/22 5:00 PM
Quiz-App - page1						

Quelle: Erstellt in Project Libre Software

Anhang 2: Gantt-Diagramm



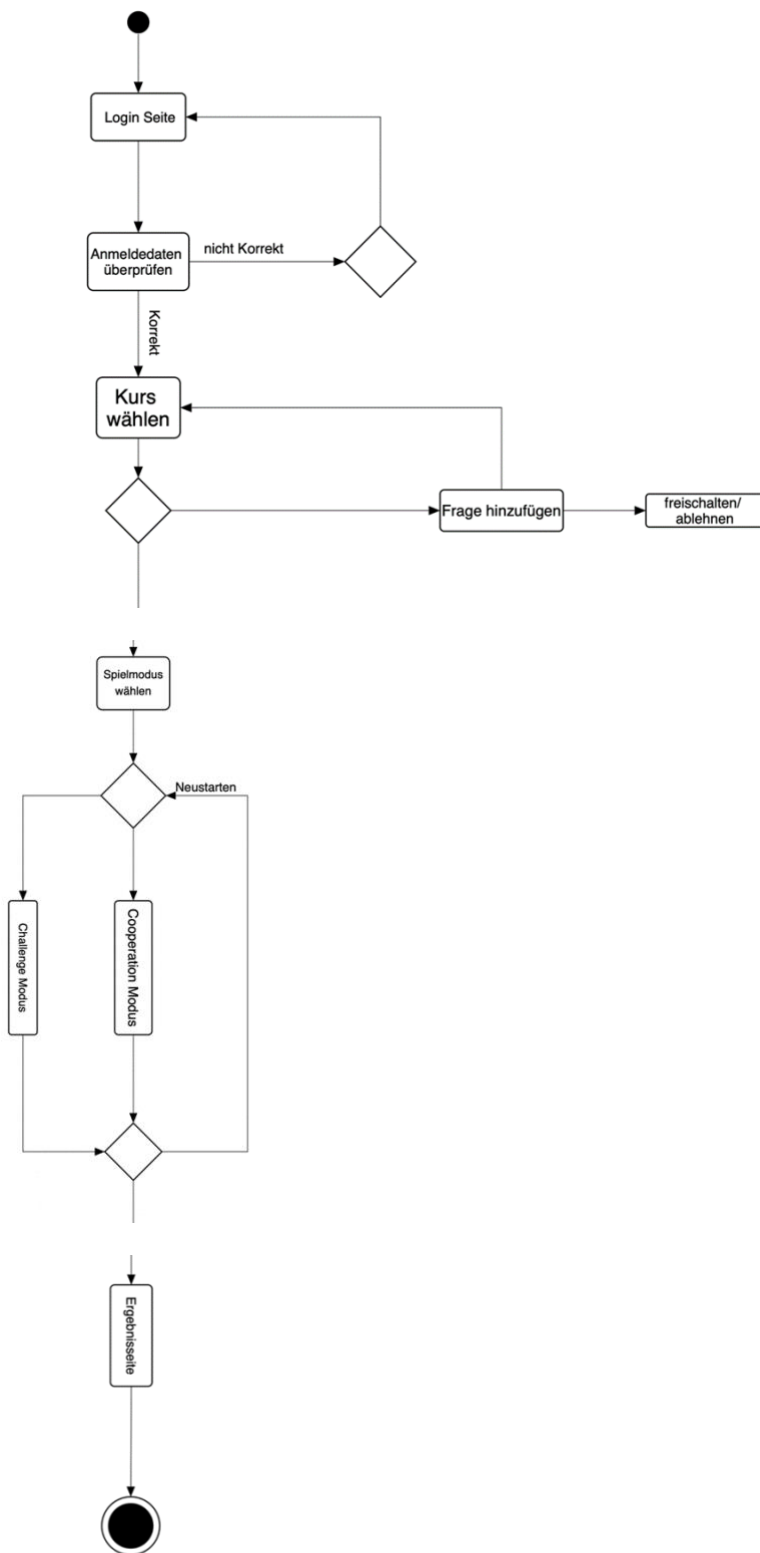
Quelle: Generiert mit Projekt Libre Software

Anhang 3: Use-Case-Diagramm



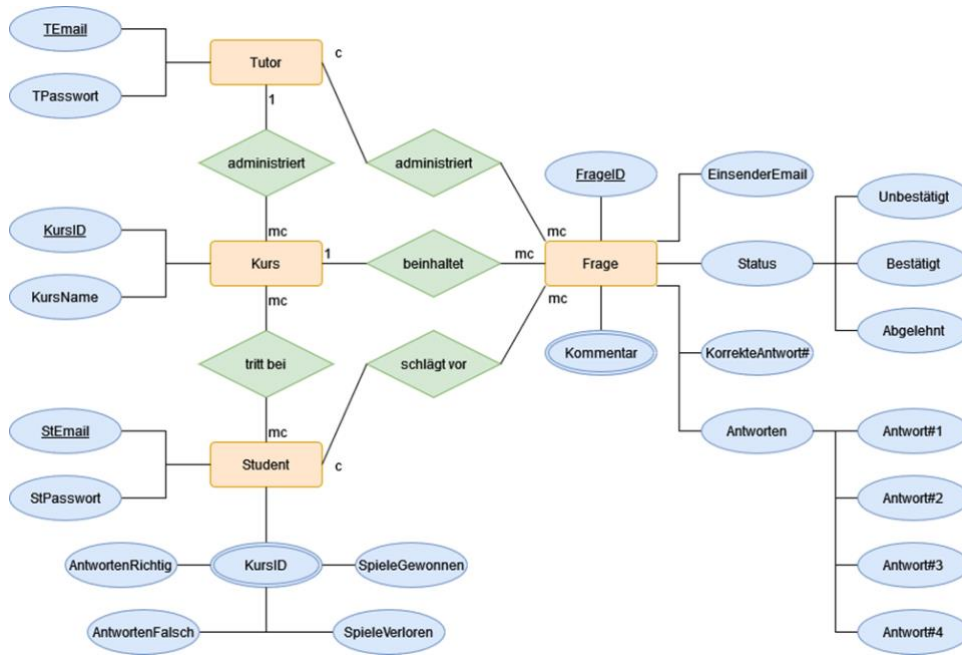
Quelle: Eigene Darstellung

Anhang 4: UML-Zustandsdiagramm



Quelle: Eigene Darstellung

Anhang 5: Entity-Relationship-Diagramm



Quelle: Eigene Darstellung

Nr.	Anforderung	Erfüllt	Nicht erfüllt	Anmerkung
1	Zwei verschiedene Spielmodi: Einen Kooperativen Modus und einen Spielmodus, in dem die Studierenden gegeneinander antreten können	Ja		Im Challenge Modus muss man die Frage anklicken, um zur nächsten Frage zu wechseln
2	Datenbanksystem zur persistenten Speicherung implementieren	Ja		Daten werden in der Google Firebase Datenbank gespeichert
3	Kursauswahl	Ja		
4	Fragenpool für jeden Kurs	Ja		
5	Verschiedene Berechtigungsstufen für Student, Tutor und Administrator	Ja		Es gibt auch Demo-Zugänge zu jeder Berechtigungsstufe über die Startseite
6	Login-Seite und Benutzerkonten für die Studierenden mit Benutzername und Passwort	Ja		
7	Studenten können sich registrieren	Ja, Bestätigungsmail wird evtl. vom IU-Mailserver blockiert		Bestätigungsmail wurde in manchen Tests vom IU-Mailserver blockiert

8	• Möglichkeit weitere Kurse hinzuzufügen für den Tutor/ Admin	Ja		
9	• Möglichkeit für die Studierenden weitere Fragen einzureichen	Ja		
10	• Möglichkeit für den Tutor fragen freizugeben oder abzulehnen	Ja		
11	• Fragenkatalog, zu dem Studierende Fragen inkl. Antworten hinzufügen können.	Ja		Fragen können im Kooperativen Modus von jedem Studenten bearbeitet und korrigiert werden
12	• Verschiedene Bereiche für einzelne Kurse	Ja		
13	• App soll an das Design der IU angepasst sein	Ja		
14	• Studenten können ihr Benutzerprofil einsehen und bearbeiten	Ja		Studenten können Anzeigenamen bearbeiten, Vollständiger Name und E-Mail-Adresse können nicht bearbeitet werden.
15	• Kurse als Favoriten speichern	Ja		
16	• FAQ-Seite mit den wichtigsten Fragen und Antworten	Ja		
17	• Die Studierenden sollen die Möglichkeit haben eigene Einstellungen	Ja		Studenten können die Fragen bewerten, ob sie

	vornehmen zu können, um den Schwierigkeitsgrad des Fragekatalogs zu bestimmen.			hilfreich waren und den Schwierigkeitsgrad angeben. Nach diesen Kriterien können die vorhandenen Community Fragen gefiltert werden.
--	---	--	--	--

Quelle: Eigene Darstellung

Anhang 7: Link zum Github Repository

<https://github.com/Rakantor/iu-quiz-app>

Anhang 8: Adresse der Webanwendung

<https://iu-quiz-app.web.app>