

Projektbericht

Software Engineering Projekt

18 | Erstellung und Verwaltung von Gruppen für Software Engineering Projekte

Namen: Julius Wyrembek, Vincent Weise, Phil Beck, Viktoria Karklin, Nuha Alhamiidi, Evelyn Dietrich

Datum: 26.06.23

Inhaltsverzeichnis

1. Aufgabenstellung und Auftraggeber	3
2. Projektplanung	3
2.1. Ausgangssituation	3
2.2. Teamaufstellung und Rollenverteilung	3
2.3. Kommunikation und eingesetzte Tools	4
2.4. Dokumentation und eingesetzte Tools.....	5
2.5. Praktiken und Bewertungen.....	6
3. Projektdurchführung	7
3.1. Einfindungsphase/ Entscheidungsphase.....	7
3.2. Grundbausteinphase.....	11
3.3. Bearbeitungsphase.....	14
3.4. Endphase Implementierung.....	16
3.5. Dokumentationsphase	20
4. Ergebnisse	22
4.1. Projektergebnisse.....	22
4.2. Gründe für Nichterreichung der Ziele.....	23
4.3. Fazit.....	24
4.4. Reflexionen.....	24

1. Aufgabenstellung und Auftraggeber

Unser Team erhielt im Rahmen des Moduls Softwareengineering den Auftrag, eine bestehende Software zur Erstellung und Verwaltung von Gruppen für zukünftige Software Engineering Projekte weiterzuentwickeln. Ziel war es die Software zu optimieren und um neue Funktionen zu erweitern, um so eine noch effektivere Unterstützung bei der Bildung von Teams zu bieten. Dabei lag der Fokus auf der Weiterentwicklung der Algorithmen zur Bildung der Teams, sowie auf dem Gestalten einer benutzerfreundlichen Oberfläche. Unser Auftraggeber ist Herr Zirkelbach, der uns ebenfalls als Coach zur Seite gestellt wurde.

In der zweiten Phase der Softwareentwicklung ist der Gruppe bewusst geworden, dass der vorhandene Code erhebliche Probleme für die weitere Bearbeitung aufweist. Der Zeitaufwand, um den Code zu verstehen und zu bearbeiten, ist sehr hoch. Das bestehende Softwaresystem ist nur schwer erweiterbar aufgrund der Unübersichtlichkeit und der teilweise hart kodierten Struktur. Infolgedessen traf die Gruppe die Entscheidung, dass eine Neuentwicklung als die effizientere Lösung angesehen wird.

2. Projektplanung

2.1. Ausgangssituation

Das Team bestand im ersten Modul von Softwareengineering aus 7 Mitgliedern. Die Zusammensetzung war wie folgt: fünf Wirtschaftsinformatikstudenten (evelyn.dietrich@stud.htw-dresden.de, julius.wyrembek@stud.htw-dresden.de, philbenedikt.beck@stud.htw-dresden.de, nuha.alhamiidi@stud.htw-dresden.de, hieutrong.thi@stud.htw-dresden.de), ein Wirtschaftsingenieurstudent (kilianluca.schinnerling@stud.htw-dresden.de) und eine Informatikstudentin (viktoria.karklin@stud.htw-dresden.de). Planmäßig verabschiedete sich Kilian Schinnerling nach SE I. Außerplanmäßig verließ Hieu Trung Thi die Gruppe. Verstärkend wurde uns der Wirtschaftsinformatikstudent Vincent Weise (vincent.weise2@stud.htw-dresden.de) zur Seite gestellt. Glücklicherweise hatte dies an unserer Rollenverteilung nichts geändert. Die ausgeschiedenen Studenten hatten die Rolle der Analysten übernommen. Diese Rolle wurde von Phil Beck nun zusätzlich übernommen. Vincent Weise wurde verstärkend in die Rolle des Implementierers aufgenommen.

2.2. Teamaufstellung und Rollenverteilung

In SE I wurden die Rollen je nach Erfahrungen und Vorlieben verteilt. Dadurch, dass sich jeder in seiner Rolle wohlfühlt hat, haben wir diese Aufteilung auch in SE II übernommen und dem neuen Mitglied eine ebenfalls geeignete Position zugeteilt.

Name	Rolle	Verwendete Namenszeichen
Evelyn Dietrich	Projektmanager/ Deployment	ED
Nuha Alhamidi	Implementierer	NA
Julius Wyrembek	Tester/ Analyse	JW
Viktoria Karklin	Implementierer	VK
Phil Benedikt Beck	Analyse/ Tester	PB
Vincent Weise	Implementierer	VW

2.3. Kommunikation und eingesetzte Tools

Zu Beginn von SE II gestaltete sich die interne Organisation in unserer Gruppe als recht unvorteilhaft. Wir trafen uns aller zwei Wochen zu einem Meeting, bei dem der genaue Termin für die nächste Iteration festgelegt wurde, durch Abstimmungen per Whatsapp (s. Abbildung 1). Diese Vorgehensweise führte allerdings dazu, dass die Dauer der Iteration um plus/minus 3 Tage schwankte.

Obwohl diese Art der Terminfindung in SE I noch akzeptabel war, erkannte unsere Gruppe in der ersten Iteration von SE II, dass ein besseres System entstehen muss. Dies war dringend von Nöten, da wir uns erst am Anfang von SE II darauf verständigt haben eine Neuimplementierung statt einer Weiterentwicklung durchzuführen. Dies hatte zur Folge, dass wir innerhalb derselben Zeitspanne mit einer größeren Anzahl an Aufgaben konfrontiert wurden. Dementsprechend beschlossen wir, unsere Iterationsdauer auf eine Woche zu begrenzen und uns alle zusammen wöchentlich am Mittwoch zur Aufgabenbearbeitung und Teambesprechung an der Hochschule zu treffen.

Die Flexibilität war hierbei ein wichtiger Faktor. Es war nicht zwingend erforderlich, dass alle Teammitglieder permanent anwesend waren und sich gegenseitig bei der Bearbeitung der Aufgaben kontrollierten. Stattdessen praktizierten wir ein agiles Arbeitsmodell, bei dem jeder die Freiheit hatte, zu kommen und zu gehen, wann immer es ihm am besten passte. Allerdings war es von entscheidender Bedeutung, dass jeder seine zugewiesenen Aufgaben fristgerecht erledigte. Ein weiterer wichtiger Punkt war, dass alle Teammitglieder an den Teambesprechungen teilnahmen. Hier wurden Probleme und Herausforderungen diskutiert und Lösungen gefunden. Die Besprechungen boten auch die Möglichkeit, die Fortschritte der Gruppe zu besprechen und aufeinander abzustimmen. So konnten wir sicherstellen, dass alle Mitglieder auf dem gleichen Stand waren und dass wir als Team effektiv und zielgerichtet arbeiten konnten.

Generell konnten wir durch die neue Herangehensweise Fragen und Probleme direkt und schnell klären, was uns ermöglichte, effizienter zu arbeiten und schneller Fortschritte zu erzielen. Zudem waren alle Mitglieder der Gruppe gewissermaßen dazu gezwungen, kontinuierlich zu arbeiten, was uns als Team näher ans Ziel brachte.

Ein weiterer Vorteil dieser Methode war, dass jeder in der Gruppe den aktuellen Fortschritt der anderen direkt einsehen konnte. Dadurch konnten wir uns gegenseitig unterstützen und motivieren, was zu einer höheren Produktivität und besseren Arbeitsergebnissen führte.

Unsere größte Errungenschaft bestand darin, dass wir als Gruppe durch die Erfolge, die wir zusammen erreicht haben, zusammengewachsen sind. Wir konnten gemeinsam unsere Fortschritte und Erfolge zelebrieren, was zu einem besseren Zusammenhalt und einer positiven Arbeitsatmosphäre beigetragen hat.

Ein weiteres Tools, welches wir für unsere Kommunikation einsetzten, war Discord. Discord bat uns die Möglichkeit themenbezogene Sprach- und Chaträume (S. Abbildung 2) zu eröffnen, um stets neue Veränderungen in jeweiligen Gebieten mitteilen zu können und immer auf dem aktuellsten Stand zu bleiben. Der Vorteil daran war, dass wir jeder Zeit wussten, wo wir benötigte Informationen finden. Weiterhin konnten wir online in Partnerarbeit an Dokumenten arbeiten, durch die Funktion der Bildschirmübertragung. Dies ersparte uns Anfahrtswege und ermöglichte eine perfekte Bearbeitung der Aufgaben.

Die Abstimmung der Termine mit dem Auftraggeber erfolgte durch den Austausch von E-Mails. Zu Beginn des Semesters baten wir ihn um eine Übersicht seiner verfügbaren Zeiten, um für die Treffen stets ein geeignetes Zeitfenster zu ermitteln. In der Regel wurden die Termine etwa eine Woche im Voraus per E-Mail mit dem Auftraggeber festgelegt. Die Treffen fanden dann an der HTW in einem geeigneten Raum statt. Die Dauer der Meetings variierte je nach Fortschritt des Teams und dauerte in der Regel zwischen 30 Minuten und anderthalb Stunden. Während dieser Treffen wurden der bisherige Projektstand, die erzielten Fortschritte und die Anmerkungen des Auftraggebers besprochen und formlos festgehalten. Die Treffen fanden etwa alle zwei Wochen statt und wurden im letzten Monat vor der Abgabe auf eine wöchentliche Regelmäßigkeit erhöht.

In manchen Fällen wurden dem Auftraggeber Fragen oder Entwürfe per Email zugesendet, woraufhin dieser Antworten bzw. Bewertungen der Entwürfe ebenfalls per Email beantwortete.

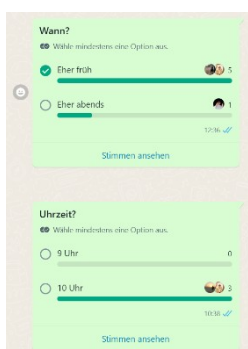


Abbildung 1: Abstimmung bei Whatsapp

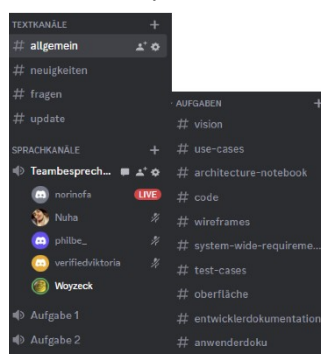


Abbildung 2: Sprach- und Chaträume in Discord

2.4. Dokumentation und eingesetzte Tools

Die Dokumentation erfolgt wie in SE I über das Hochladen bei GitHub. Dafür wurde ein neuer Ordner angelegt (belegabgabe_se2) in dem alle Dokumente hochgeladen wurden. Für die Dokumentationen haben wir Worddokumente angelegt. Jeweils eine für die Test-, Entwickler, Betriebs- und Anwenderdokumentation, sowie den Projektbericht. Die Entscheidung zur Benutzung von Word war einvernehmlich und beruhte auf den positiven und jahrelangen Erfahrungen der einzelnen Teammitglieder. Ein Problem, was die Benutzung von Word mit sich brachte, war die Bearbeitung an einem Dokument von mehreren Mitgliedern gleichzeitig. Dies ging leider nicht, sodass nur eine Person an einem Dokument arbeiten konnte und erst nach dem Hochladen der Datei, konnte ein anderes Mitglied das Dokument bearbeiten. Jedoch waren wir uns dessen schon im Vorhinein bewusst, weshalb wir damit keine Schwierigkeiten hatten. Eine weitere Herausforderung bestand darin, Hyperlinks zu implementieren. Leider ermöglichte unsere Version von Word diese Funktion nicht, wodurch es uns nicht möglich war, zwischen dem Inhaltsverzeichnis und den entsprechenden Überschriften hin und her zu springen. Trotz dieses Hindernisses entschieden wir uns dennoch für die Verwendung von Word aufgrund der vielen Vorteile, die es uns bot.

Durch die Neuentwicklung mussten auch das Architecture Notebook, die Use Cases, das Use Case Model, die Vision, das Glossary, die Systemweiten Anforderungen und das Domainmodel aus SE I angepasst werden. Dafür haben wir in dem vorgefertigten Ordner belegabgabe_se1 die bereits vorhandenen Dateien verändert und mit AsciiDoc überarbeitet. Für Modelle und Diagramme nutzten wir Visual Paradigm, sowie den Essence Navigator.

Für bestimmte Dokumente haben wir teilweise auf vorgefertigte Vorlagen zurückgegriffen, die von uns angepasst wurden. Die Iterationspläne behielten das Muster bei, das uns in SE I vorgegeben wurde. Für die Testdokumentation nutzten wir hingegen eine eigens erstellte Vorlage. Dies erwies sich als äußerst hilfreich, da wir manuelle Tests durchführten und somit eine einheitliche und übersichtliche Struktur schaffen konnten. Darüber hinaus führte die Verwendung der selbsterstellten Vorlage für die Testdokumentation zu einer zeitsparenden Durchführung des Testprozesses.

Für die Projektplanung nutzen wir das Tool „Projects“ in GitHub. Dabei erstellten wir Issues, die bestimmten Teammitgliedern zugeordnet und während einer Iteration bearbeitet werden sollten. In der ersten Phase der Softwareentwicklung hatten wir Schwierigkeiten, das Tool so einzusetzen, dass es eine unterstützende Rolle anstatt einer zusätzlichen Belastung für uns darstellte. Allerdings wurde es im Verlauf der zweiten Phase zu einem hilfreichen Werkzeug, um Struktur in unser Projekt zu bringen und unsere Arbeit effizient zu organisieren.

2.5. Praktiken und Bewertung

Bewertung 1 sehr schlecht, 2 schlecht, 3 mittel, 4 gut, 5 sehr gut

Praktiken	Beschreibung	Bewertung
Weeklys	Eine Methode, bei der sich die Gruppenmitglieder einmal die Woche treffen um erzielte Ergebnisse und die nächsten Schritte zu besprechen. Damit wird sichergestellt, dass alle Mitglieder auf demselben Projektstand sind.	5
Risikobesprechung	Eine Methode zur frühzeitigen Reduzierung von möglichen Risiken. Dabei werden auch bereits vorhandene Risiken	4 – Erleichterte die Arbeit stark, jedoch

	angesprochen und zeitnah eliminiert. (Risiken und Lösungsansätze sind in der Rikslist zu finden)	ergeben sich wieder neue Probleme, die vorher nicht berücksichtigt wurden
Agile Entwicklung	Eine iterative und inkrementelle Entwicklungsmethode, die es Teams ermöglicht, schnell auf Änderungen zu reagieren und qualitativ hochwertige Software zu liefern.	5
Clean Code	Ein Ansatz, bei dem besonderer Wert auf lesbaren, wartbaren und gut strukturierten Code gelegt wird, um die Wartbarkeit, Qualität und Erweiterbarkeit der Software zu verbessern. Diese Entscheidung basierte auf dem Vorgängercode, da dieser, unserer Meinung nach, die oben beschriebenen Eigenschaften kaum aufwies.	5
Lean Development	Eine Methode, die sich auf die Reduzierung von Verschwendung konzentriert, indem unnötige Schritte, Verzögerungen und Ressourcenverschwendung minimiert werden, um Effizienz und Produktivität zu steigern.	4 - Einschätzung, ob Verschwendung vorliegt oder nicht, nicht immer gegeben
Pair Programming	Eine Methode, bei der zwei Entwickler zusammenarbeiten und den Code gemeinsam entwickeln, um die Qualität zu verbessern und das Wissen auszutauschen.	4- Nimmt einen Ressourcen, da zwei Mitglieder an einer Aufgabe arbeiten

3. Projektdurchführung

3.1. Einfeldungsphase/ Entscheidungsphase (22.03.23-22.04.23)

Zu Beginn des Semesters fand am 22.03.23 ein kurzes Meeting der Gruppe statt, um sich wieder in die Thematik einzufinden und organisatorische Fragen zu klären, wie zukünftige Treffen mit dem Auftraggeber und der weitere Verlauf des Projektes. Ein großes Problem, mit dem wir konfrontiert waren, bestand darin, dass unsere vorgegebene Webseite nur lokal funktionierte. In der vorherigen Phase SE I hatten wir alle unsere Ressourcen darauf konzentriert, die Webseite zum Laufen zu bringen, allerdings ohne Erfolg. Daher stellte sich die grundlegende Frage: Schaffen wir es, die Webseite wieder zu aktivieren, oder sollten wir eine komplett neue Seite erstellen? Diese Bedenken wurden am 23.03.23 mit dem Coach besprochen, und wir erhielten neue Hilfestellungen, um das Problem zu beheben. Unter anderem wurden diese Hinweise zu einzelnen Issues zusammengefasst:

- Lokale Installation von Docker und Einführung
- Problembehebung

- Implementierung des ID-Feldes
- Recherche nach geeigneten Oberflächenframeworks
- Erstellung einer Grundstruktur mittels HTML/CSS/JavaScript

Herr Zirkelbach verwies ebenfalls auf unsere Herangehensweise mit den Issues. Obwohl wir das Organisationstool nicht umfassend genutzt haben und Schwierigkeiten hatten, es in unseren Prozess zu integrieren, haben wir dennoch versucht, es verstärkt einzubeziehen.

Lokale Installation von Docker und Einführung

Wir begannen direkt mit den Aufgaben und trafen uns am 25.03.23 in der Slub, um Docker gemeinsam zu installieren und einzuführen. Wir haben uns für Docker entschieden, da wir eine Schnittstelle für unsere Datenbank, das Front- und Backend brauchten. Mit diesem Tool hat auch unsere Vorgängergruppe gearbeitet, weshalb wir VK sich bereits in SE I damit auseinandergesetzt haben und somit das nötige Wissen hatte. Sie gab der restlichen Gruppe eine kurze Einführung in das Thema. Die Installation verlief reibungslos bei allen Teilnehmern außer ED, da sie Docker aus uns unverständlichen Gründen nicht installieren konnte. Wir verblieben so, dass sie mit NA zusammen arbeitet, falls sie Docker braucht. Die Entscheidung für Docker brachte mehrere Vorteile mit sich. Durch die Verwendung von Containern konnten wir eine isolierte Umgebung schaffen, in der wir die einzelnen Komponenten unserer Anwendung unabhängig voneinander entwickeln und testen konnten. Dies ermöglichte uns eine effiziente Entwicklung und erleichterte auch die Skalierbarkeit und Bereitstellung der Anwendung in verschiedenen Umgebungen.

Problembehebung

Aufgrund unserer Ratlosigkeit, wie wir die Webseite wieder auf dem Server zum Laufen bringen sollten, holten wir uns Unterstützung von zwei Mitgliedern der vorherigen Gruppe. Wir vereinbarten ein Online-Treffen am 01.04.23. Vor dem Treffen stellten wir uns Fragen, die uns einerseits beim besseren Verständnis des Codes und andererseits bei der weiteren Implementierung helfen sollten (s. Abbildung 2). Obwohl alle unsere Fragen beantwortet werden konnten, wiesen uns die Mitglieder der vorherigen Gruppe darauf hin, dass an mehreren Stellen im Code, die für uns relevant sind, hartkodierte Werte verwendet wurden, was zu neuen Herausforderungen bei der Weiterentwicklung führen würde. Darüber hinaus empfahlen sie uns, anstelle der Erweiterung ihrer Webseite eine komplett neue Webseite zu entwickeln, da die Weiterentwicklung schwieriger wäre als eine Neugestaltung. Die Zusammenarbeit mit den Mitgliedern der vorherigen Gruppe war ein wichtiger Wendepunkt für unser Projekt. Ihre Unterstützung half uns nicht nur dabei, die Probleme mit der bestehenden Webseite zu verstehen, sondern gab uns auch wertvolle Einblicke in bewährte Praktiken und Möglichkeiten zur Verbesserung der Softwareentwicklung.

SE 2 Fragen

Code:

Warum überschreibt die Webseite Änderungen (nach Änderung kurz auf Webseite sichtbar, dann läd diese neu und zeigt ursprüngliche Webseite an)

Wo überall (konkrete Datei + Zeilen) muss man etwas ändern um z.B.:

- einen Button
- eine Tabelle/Liste
- einen Datensatz
- ein Feature

ändern/erstellen zu können? Und wie, mit welchem Keyword sucht man mit Steuerung F am besten dannach? (Mit ID kommt man nicht weit)

Was genau muss man ändern/ einfügen? (z.B. neue Klasse anlegen? in vorhandener Klasse etwas ändern?)

Wo ändert man das Design? (Bilder in Ordner legen alte löschen, Namen im Code ändern, was ist dieses css und json)

Eine Übersicht (in Form eines Baumes) die alle Dateien mit ihrer Funktion beschreiben, welche mit Pfeilen verbunden sind an denen steht weshalb diese Elemente verbunden sind, wäre sehr hilfreich um sich nicht alles merken zu müssen und auch für die Doku von Wert. Am besten noch schreiben was für Elemente (z.B. Progression Bar oder Menüleiste) in jenen definiert sind.

Abbildung 3: Fragen an Vorgängergruppe

Am 04.04.23 traf sich unsere Gruppe erneut und traf schnell und einvernehmlich die Entscheidung, eine Neuimplementierung durchzuführen. Dabei diskutierten wir direkt die Tools, mit denen wir weiterarbeiten wollten. Basierend auf den individuellen Erfahrungen der einzelnen Mitglieder entschieden wir uns für Java als Programmiersprache, MySQL als Datenbank und Spring Boot als Framework. Ähnlich wie die vorherige Gruppe wollten wir diese Tools in Docker-Containern betreiben.

Die Abstimmung dieser Entscheidung mit dem Auftraggeber war noch ausstehend. Am 06.04.23 besprachen wir unsere Erkenntnisse mit ihm und erhielten seine Zustimmung, jedoch mit Skepsis, da noch viel Arbeit vor uns lag und wir bisher nur theoretische Grundlagen vorweisen konnten. Wir gingen nochmals die Use Cases mit ihm durch, um sicherzustellen, dass alle Anforderungen berücksichtigt wurden. Wir waren uns bewusst, dass alle von uns identifizierten Use Cases nun berücksichtigt und neu implementiert werden mussten (s. Abbildung 3). Der Auftraggeber merkte jedoch an, dass der Use Case "Backup erstellen/laden" vorerst in den Hintergrund treten könne und nur bei ausreichender verbleibender Zeit berücksichtigt werden sollte. Für die nächste Iteration standen der Import der Studentendaten und die Bearbeitung des Fragebogens im Vordergrund. Bei der Neuentwicklung stützten wir uns auf den Prototypen, den wir in SE I erstellt hatten.

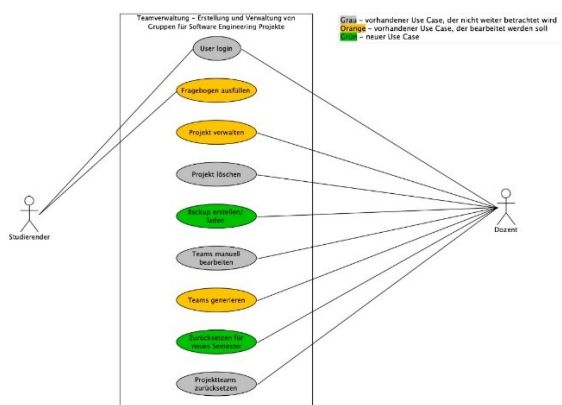


Abbildung 4: Altes Use Case Model

Implementierung des ID-Feldes

Dieses Issue ist durch die komplette Umstrukturierung des Projektes in den Hintergrund gerückt und musste erstmal außen vor gelassen werden. Jedoch wurde diese Anforderung in unserem Prototypen implementiert, weshalb wir diesen in der nächsten Iteration ausbauen wollten.

Recherche nach geeigneten Oberflächenframeworks

JW übernahm die Verantwortung für die Recherche nach einem geeigneten Oberflächentool. Nach gründlicher Überlegungen entschied er sich für Bootstrap, da es den Anforderungen des Auftraggebers entsprach und nahtlos mit unserem Backend kompatibel war. Die Gestaltung der Benutzeroberfläche wurde sofort angegangen und die Ergebnisse wurden dem Auftraggeber per Email am 16.04.23 präsentiert. Er hatte nur geringfügige Optimierungsvorschläge, war aber insgesamt zufrieden mit dem Design der Oberfläche (s. Abbildung 4).

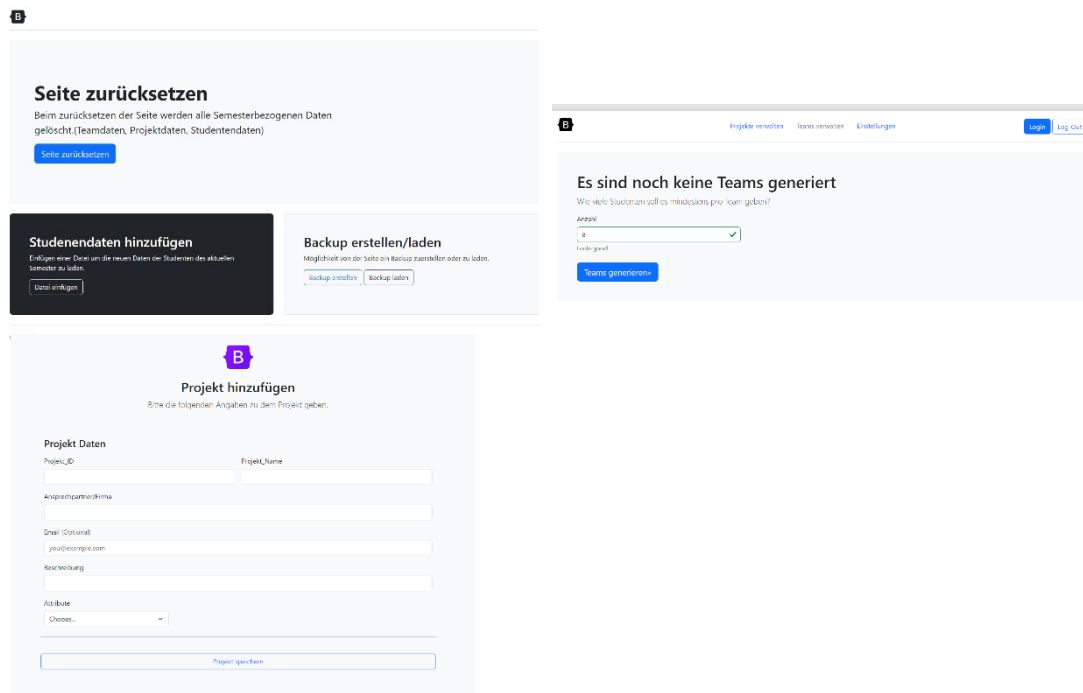


Abbildung 5: Oberflächengestaltung

Erstellung einer Grundstruktur

VW erstellte eine Grundstruktur mittel CSS, HTML und JavaScript, die groben Orientierung für das Projekt galt. So wollten wir erstmals den Aufbau unserer Software veranschaulichen. Diese verfügte jedoch noch über keine Funktionen.

Team_U1			Team_U2		
Vorname	Nachname	Studiengang	Vorname	Nachname	Studiengang
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik

Team_U3			Team_U4		
Vorname	Nachname	Studiengang	Vorname	Nachname	Studiengang
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik
Mark	Otto	Wirtschaftsinformatik	Mark	Otto	Wirtschaftsinformatik

zur

Bewertung Einfindungsphase/ Entscheidungsphase

Wir haben viel Zeit damit verbracht, die Entscheidung zur Neuimplementierung gründlich zu überdenken, da wir besorgt waren, ob wir unsere Ziele erreichen würden. Letztendlich stellte sich jedoch heraus, dass die Entscheidung die richtige war, da wir mit dem alten Code völlig überfordert waren. Durch die Neuimplementierung konnten wir zumindest Kenntnisse in ausgewählten Tools erlangen.

Trotzdem müssen wir feststellen, dass die Organisation innerhalb der Gruppe nicht ausreichend war, um ein gutes Ergebnis zu erzielen. Meetings wurden spontan angesetzt, der Kontakt zum Auftraggeber war unregelmäßig und vor allem waren die zugewiesenen Aufgaben zu umfangreich, um sie in ihrer Komplexität zu verstehen und angemessen zu bearbeiten.

Essence Navigator Einfindungsphase/ Entscheidungsphase

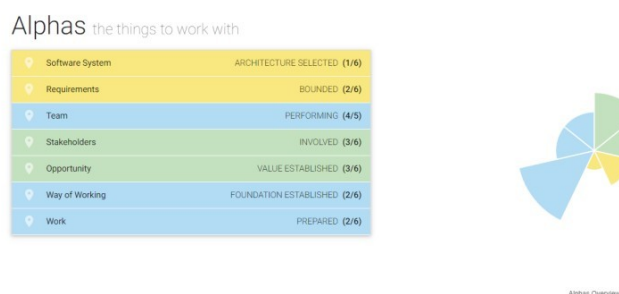


Abbildung 6: Essence Diagramm nach Einfindungsphase/Entscheidungsphase

3.2. Grundsteinphase (23.04.23- 10.05.23)

In unserem Meeting am 23.04.23 besprachen wir den weiteren Implementierungsplan. Unser erster Schritt bestand darin, den Prototypen aus SE I in Java umzuschreiben, um einen Ausgangspunkt für die Weiterentwicklung des Systems zu schaffen. Da wir nur begrenzt Zeit zur Verfügung hatten, mussten wir direkt mit der Implementierung beginnen, um letztendlich eine funktionsfähige Webseite vorweisen zu können. Zudem wurde ein Verantwortlicher für die Erstellung der Dokumentationen bestimmt. Die festgelegten Issues für diese Iteration waren wie folgt:

- Klassen in Java übernehmen
- Einbindung einer Loginansicht
- Verbindung der Datenbank mit dem vorhandenen Code
- Erstellung von Docker-Containern für die Datenbank, das Backend und das Frontend
- Vorbereitung einer Präsentation für den fachlichen Austausch im Projektmanagement

In einem Meeting mit unserem Coach am 04.05.23 wurden uns die Fehler in unserem bisherigen Projekt deutlich gemacht. Insbesondere wurde die unzureichende Nutzung von GitHub bemängelt. In den Meetings hatten wir Aufgaben erteilt, die von allen verstanden wurden, jedoch zu umfangreich waren, um sie in der Kürze der Zeit zu bewältigen. Dies führte dazu, dass wir nur wenige Issues hatten, aber viel Aufwand betreiben mussten und die Issues immer wieder in die nächste Iteration übernehmen mussten. Unser Auftraggeber empfahl uns, die Aufgaben kleiner zu halten und sie einzeln aufzuschreiben. Daher sollten wir für die nächste Iteration 12 Issues erstellen. Diese Methode erwies sich als effizienter für unsere Gruppe und ermöglichte es uns besser abzuschätzen, wie weit die Mitglieder mit ihren jeweiligen Aufgaben gekommen waren. Darüber hinaus wurde die mangelnde Organisation in unserem GitHub-Repository kritisiert, da unser bisheriger Code nicht für alle zugänglich war. Wir änderten dies noch am selben Tag.

Da unser Coach auch unser Auftraggeber ist, führten wir anschließend ein Gespräch mit ihm. Er präsentierte uns seine Prioritäten für die Implementierung. An erster Stelle stand das Importieren der Studentendaten aus einer CSV-Datei, gefolgt von der Möglichkeit, Teams anzulegen. Zuerst sollte die Erstellung und Speicherung des Fragebogens in der Datenbank erfolgen und schließlich das Generieren der Teams. Basierend auf diesem Gespräch nahmen wir zusätzliche Issues in Angriff:

- Import der Studentendaten aus einer CSV-Datei
- Erweiterung des Fragebogens
- Erstellung einer Schnittstelle von Java mit dem Fragebogen

Klassen in Java übernehmen

Zu Beginn haben wir den Code von der Vorgängergruppe analysiert und alle notwendigen Klassen in Java übertragen. Wir haben uns an der Anzahl und den Inhalten der Klassen orientiert und diese

übernommen, um im späteren Verlauf damit weiterarbeiten zu können und somit eine Basis geschaffen.

Einbindung einer Loginansicht

VW übernahm die Implementierung des Logins in die Backend-Struktur von Java Spring. Dazu musste der Login-Controller in Java Spring entsprechend angepasst werden. VW erstellte zusätzliche Klassen und implementierte neue Interfaces, um die Funktionalität des Logins umzusetzen. Dabei integrierte er die grundlegende HTML- und CSS-Struktur in das Backend. Durch diese Anpassungen war es möglich, den Login-Vorgang im Java Spring-Backend zu verarbeiten.

Verbindung der Datenbank mit dem vorhandenen Code

Um die Projekte anzulegen, war es notwendig, eine Datenbank einzurichten, um die eingegebenen Werte zu speichern. NA übernahm diese Aufgabe und erstellte eine Datenbank mit MySQL. Sie stellte eine Verbindung zwischen der Datenbank und dem Code her, indem sie eine REST-API einrichtete. Um die Datenbank optimal zu nutzen, wurde ein Docker-Container erstellt, der die notwendige Umgebung bereitstellte. Dadurch konnte die Datenbank effizient in das System integriert werden.

Erstellung von Docker-Containern für die Datenbank, das Backend und das Frontend

VK hat zu Beginn den Grundstruktur Container für die Webseite erstellt. Dann gab es technische Schwierigkeiten einen Container für die Datenbank zu erstellen, da sie ratlos war welche Vorlage sie für diesen Container brauchte. Sie suchte sie sich erfolgreich Hilfe bei Ruben und setzte gemeinsam mit NA den Docker-Container auf.

Vorbereitung einer Präsentation für den fachlichen Austausch im Projektmanagement

Um die Präsentation kümmerte sich ED, wobei sie sich an der bereits geschriebenen Planung im Projektbericht gut orientieren konnte. Diese Präsentation erfolgte am 08.05.23.

Import der Studentendaten aus CSV-Datei

Der Import der Studentendaten wurde in drei separate Issues aufgeteilt. PB war dafür zuständig, eine CSV-Datei mit selbst gewählten Beispieldaten zu erstellen. Dabei musste darauf geachtet werden, dass die Spalten in der richtigen Reihenfolge benannt wurden, damit die Attribute der Tabelle in der Datenbank korrekt ausgelesen werden konnten. VW und JW kümmerten sich um das Auslesen und Einbinden der CSV-Datei. Sie orientierten sich am vorherigen Code und übertrugen die Logik des CSV-Dateiauslesens auf Java. Zu diesem Zeitpunkt funktionierte das Auslesen und Einbinden der Daten noch einwandfrei.

Erweiterung des Fragebogens

Der Auftraggeber stellte die Anforderung, den Fragebogen um die Frage nach einer Fachinformatikausbildung zu erweitern. JW übernahm diese Erweiterungsaufgabe. Er verwendete die HTML-Seite der vorherigen Gruppe und fügte eine Checkbox mit dem Titel "Vorerfahrung" hinzu, bei der man ankreuzen konnte, ob man eine Fachinformatikausbildung absolviert hat. Jedoch wurde diese Erweiterung in dem nächsten Issue noch nicht berücksichtigt.

Erstellung einer Schnittstelle von Java mit dem Fragebogen

NA übernahm die Aufgabe, den Fragebogen in Java einzubinden. Zunächst erstellte sie die benötigten Klassen in Spring, wie zum Beispiel die Klasse "Project", in der alle angelegten Projekte gespeichert wurden. Ebenso wurden Klassen wie "Role" für verschiedene Rollen und "Skill" für die abgefragten Fähigkeiten erstellt. Zusätzlich wurden für die Antworten jeweils eigene Klassen erstellt.

NA verband alle diese Klassen mithilfe einer REST-API mit dem Frontend. Anschließend stellte sie eine Verbindung zur Datenbank her, um sicherzustellen, dass die Antworten des Fragebogens in der Datenbank gespeichert werden konnten.

Bewertung

Nachdem wir den Rat erhalten hatten, die Issues kleiner zu halten und mehr davon zu erstellen, lief die Bearbeitung in der nächsten Iteration deutlich besser. Wir setzten uns gemeinsam in Meetings und schrieben für jeden Einzelnen die Aufgaben der Iteration auf. Dadurch waren alle Mitglieder auf dem gleichen Stand und es gab eine klare Aufgabenverteilung. Allerdings bemerkten wir in dieser Phase, dass wir zu wenig Zeit hatten, um alles zu bewältigen. Obwohl wir bereits viel geschafft hatten, war das Tempo nicht nachhaltig. Aus diesem Grund einigten wir uns darauf, die Iterationsphasen zu verkürzen und uns wöchentlich zu treffen, um die Aufgaben zu bearbeiten. Diese Umstellung legte einen bedeutenden Grundstein für das System und sorgte für Motivation innerhalb der Gruppe, aufgrund der kleinen Erfolge.

Die Software selbst hatte noch viel Potenzial zur Weiterentwicklung. Wir waren uns jedoch bewusst, was noch zu tun war, und hatten einen Überblick über den aktuellen Stand. In dieser Phase wurde deutlich, dass wir versucht hatten, die Webseite der Vorgängergruppe so gut wie möglich wiederherzustellen und gleichzeitig die neuen Anforderungen des Auftraggebers zu erfüllen. Allerdings hatten wir unsere Aufgaben immer getrennt voneinander bearbeitet, sodass wir die alten Anforderungen oft nicht mit den neuen verknüpft hatten. Wir hatten von allem ein bisschen, aber nichts war vollständig integriert.

Essence Navigator

Software System	ARCHITECTURE SELECTED (1/6)
Requirements	COHERENT (3/6)
Team	PERFORMING (4/5)
Stakeholders	INVOLVED (3/6)
Opportunity	VALUE ESTABLISHED (3/6)
Way of Working	IN USE (3/6)
Work	STARTED (3/6)



Alphas Overview

Abbildung 7: Essence Diagramm nach Grundsteinphase

3.3. Bearbeitungsphase (11.05.23-24.05.23)

In dieser Phase haben wir uns intensiv mit den Use Cases "Teams generieren" und "User einloggen" auseinandergesetzt und gleichzeitig die Anforderungen aus der vorherigen Phase erfüllt. Dabei wurden die folgenden Aufgaben aufgeteilt und bearbeitet:

- Anpassung des Algorithmus zur Teamgenerierung
- Erstellung des Logins für die Studierenden
- Erstellung der Docker.yml-Datei
- Verfassen der Projektplanung im Projektbericht
- Entwicklung von Testmethoden

Zusätzlich wurde in einem Meeting mit dem Coach am 25.05.23 erneut unsere GitHub-Struktur kritisiert. Wir wurden angewiesen, alle Fortschritte und Zwischenstände der einzelnen Teammitglieder regelmäßig auf GitHub hochzuladen. Bisher hatten wir dieses Vorgehen vernachlässigt. Jeder von uns war zwar mit seinen eigenen Aufgaben beschäftigt, die relativ unabhängig von den Aufgaben der anderen waren, und deshalb haben wir es nicht für notwendig erachtet, unsere Ergebnisse kontinuierlich hochzuladen. Wir haben uns jedoch dieser Anforderung angenommen und daran gearbeitet, unsere Ergebnisse regelmäßig auf GitHub zu aktualisieren.

Anpassung des Algorithmus zur Teamgenerierung

JW und VW waren für die Anpassung des Algorithmus zur Teamgenerierung im Pair Programming verantwortlich. Dabei mussten sie den vorhandenen Code der Vorgängergruppe auf Java umstellen. Eine Anforderung des Auftraggebers war es, dass eine Mindestanzahl an Studierenden pro Team festgelegt werden kann. Obwohl wir bereits einen vorgegebenen Algorithmus hatten, stellte sich die Anpassung an das neue System als äußerst herausfordernd heraus. Die theoretische Herangehensweise wurde ausgearbeitet, aber die Implementierung nahm viel Zeit in Anspruch. Daher konnten sie dieses Issue in dieser Phase nicht vollständig abschließen.

Login für den Studenten erstellen

NA war für die Erstellung des Backends für den Login verantwortlich. Hierfür benötigten wir eine Zertifikatskette. Wir nahmen die Zertifikatskette der vorherigen Gruppe, da uns zugesichert wurde, dass wir diesen nutzen könnten. Es stellte sich jedoch heraus, dass die Implementierung des Servers mühevoll und somit nach etlichen Versuchen nicht erfolgreich war. Die Bearbeitung dieses Issues erwies sich als äußerst zeitaufwendig, da viel Zeit und Mühe in die Recherche investiert wurde, um eine Verbindung mit dem LDAP-Server herzustellen. Leider scheiterten alle Versuche, den Server erfolgreich einzubinden. Aufgrund dieser Herausforderungen konnte das Backend nur teilweise erstellt werden und die Integration des LDAP-Servers konnte in dieser Phase nicht erreicht werden, sodass weiterhin mit hart codierten Logindaten weitergearbeitet werden musste.

Docker.yml file anlegen

In diesem Ordner befinden sich die Container. Der Ordner heißt Docker compose und wird benötigt um die Docker-Container im späteren Verlauf zu öffnen und die Webseite anzeigen zulassen. Diese Anwendung von der Vorgängergruppe übernommen und in unser Projekt implementiert.

Projektplanung im Projektbericht schreiben

Für den Projektplan war ED zuständig. Durch die Hilfestellungen, die Herr Anke uns in Opal gegeben hat, war dies eine schaffbare Aufgabe und wurde erledigt.

Entwicklung von Testmethoden

Die Entwicklung der Testmethoden konnte in dieser Phase nicht erfolgen, da die Konzentration der Gruppe auf dem Implementieren der Use Cases lag. Somit wurde diese Aufgabe in die nächste Phase geschoben.

Bewertung Bearbeitungsphase

In dieser Phase traten die größten Herausforderungen im gesamten Projekt auf. Der Algorithmus für die Teamsgenerierung und der Login waren entscheidende Komponenten unseres Systems. Wir waren uns bewusst, dass die Webseite ohne die Implementierung dieser Anforderungen nur halb so wertvoll sein würde. Daher haben wir in dieser Phase alle verfügbaren Ressourcen auf diese Bereiche konzentriert. Wir haben intensiv recherchiert, Hilfe in Anspruch genommen und alles Mögliche ausprobiert. Obwohl wir einige kleinere Aufgaben erledigt haben, wie das Ergänzen von Dokumenten oder das Anpassen des Codes an einigen Stellen, konnten wir die wichtigsten Aufgaben nicht abschließen. Daher fällt diese Phase eher negativ aus.

Dennoch gab es positive Aspekte. Wir haben erfolgreich eine Struktur in unserem GitHub-Repository aufgebaut, in der jeder von uns arbeiten konnte. Darüber hinaus haben wir in dieser Phase erstmals die Iterationsphasen auf eine Woche verkürzt und waren zufrieden mit diesem Ansatz. In den Meetings konnten wir unsere Probleme persönlich besprechen und direkt neue Lösungsansätze im

Team ausprobieren. Dadurch wurde der Zusammenhalt und die Zusammenarbeit in der Gruppe erheblich gestärkt.

Insgesamt lässt sich sagen, dass unsere Software noch erhebliches Potenzial zur Weiterentwicklung bot, jedoch haben wir in dieser Phase wichtige Fortschritte in Bezug auf die Gruppendynamik gemacht.

Essence Navigator

Alphas *the things to work with*

Software System	DEMONSTRABLE (2/6)
Requirements	COHERENT (3/6)
Team	PERFORMING (4/5)
Stakeholders	INVOLVED (3/6)
Opportunity	VIALE (4/6)
Way of Working	WORKING WELL (5/6)
Work	STARTED (3/6)



Alphas Overview

Abbildung 8: Essence Diagramm nach Bearbeitungsphase

3.4. Endphase Implementierung (25.05.23-14.06.23)

Im Meeting vom 25.05.23 haben wir den aktuellen Stand unserer Software besprochen. Das Backend war bereits weitgehend entwickelt und enthielt alle wesentlichen Funktionen gemäß den Anforderungen. Es gab einen Login, eine Dozentenansicht mit der Möglichkeit, Projekte anzulegen und einen Überblick über die Teams zu erhalten. Ebenso gab es eine Studentenansicht, in der der Fragebogen ausgefüllt werden konnte. All diese Funktionen waren erfolgreich mit der Datenbank verbunden. Mit diesem Wissen konnten wir uns auf die wichtigsten verbleibenden Aufgaben konzentrieren:

- Anpassung des Algorithmus für die Teamgenerierung und den Fragebogen
- Einbindung des LDAP-Servers in den Login-Prozess
- Aktualisierung der Dokumente aus SE I, um sie an die neuen Anforderungen anzupassen
- Verknüpfung des Backends mit dem Frontend
- Oberflächengestaltung der HTML Seiten mit Bootstrap/ CSS
- Erstellung einer Präsentation für den Entwurf und die Testphase
- Beginn der Entwicklerdokumentation
- Ausarbeitung der Projektdurchführung im Projektbericht

Diese Aufgaben waren von großer Bedeutung, um die Software weiterzuentwickeln und die verbleibenden Anforderungen des Auftraggebers zu erfüllen. Zudem war es wichtig, den Fortschritt und die Arbeitsschritte in Dokumenten festzuhalten, um die Entwicklung nachvollziehbar und transparent zu gestalten.

Algorithmus anpassen für die Teamgenerierung und den Fragebogen

JW und VW widmeten sich beharrlich der Anpassung des Algorithmus. Sie unternahmen verschiedene Ansätze, wie das manuelle Umschreiben des alten Algorithmus, die Verwendung einer Java-Erweiterung zum Auslesen des Python-Codes der Vorgängergruppe und sogar den Versuch, einen eigenen Algorithmus zu schreiben. Leider waren alle ihre Bemühungen erfolglos. In einem gemeinsamen Treffen mit NA diskutierten sie das Problem und erhielten Unterstützung bei der Umsetzung des vorherigen Algorithmus in Java, um Fehlermeldungen zu vermeiden. Zu diesem Zeitpunkt hatten wir einen funktionierenden Algorithmus, waren uns jedoch nicht sicher, ob er einwandfrei funktionierte. Darüber hinaus war die Berücksichtigung der Fachinformatikausbildung im Algorithmus noch nicht erfolgt.

LDAP Server in den Login einbinden

Nach zahlreichen Versuchen, den LDAP-Server einzubinden, erhielten wir in der Mitte dieser Phase die Information, dass sich die Zertifikatskette zur Überprüfung des LDAP-Servers geändert hatte. Als Reaktion darauf erhielten wir eine neue Zertifikatskette. Trotz dieser Aktualisierung traten aufgrund der Komplexität des Backends weiterhin schwerwiegende Fehler auf, die nur schwer zu beheben waren. Daraufhin haben wir Kontakt zur Vorgängergruppe aufgenommen, um Rat einzuholen. Sie schlugen vor, vorerst einen unverschlüsselten Login zum Testen zu implementieren, anstelle eines verschlüsselten Logins. Leider führte auch dieser Ansatz nicht zum gewünschten Erfolg.

In der Zwischenzeit erstellte VW einen einfachen Prototypen mit geringer Komplexität, um mögliche Fehler in anderen Java-Klassen auszuschließen, beispielsweise in den Interface-Klassen. Dieser Prototyp diente ausschließlich der Kommunikation mit dem LDAP-Server und konnte erfolgreich ohne Verschlüsselung implementiert werden. Jedoch erwies sich die verschlüsselte Variante erneut als problematisch. Anschließend versuchte VW, die gewonnenen Erkenntnisse auf unser bestehendes System zu übertragen, jedoch mit nur teilweisem Erfolg. Aufgrund der Ausschöpfung unserer Ressourcen und der vielen gescheiterten Versuche entschieden wir, diese Anforderung vorerst in den Hintergrund zu stellen, da die Implementierung dieser Aufgabe zu viel Zeit in Anspruch nahm und unsere zeitlichen Ressourcen stark beanspruchte.

Dokumente aus SE I auf die neuen Anforderungen anpassen

Während dieser Phase nahmen PB und VW die Aufgabe der Überarbeitung der Dokumente in Angriff. Durch unsere gemeinsamen Meetings und die kontinuierliche Aktualisierung unserer Stände auf GitHub waren alle Teammitglieder stets auf dem neuesten Stand und VW und PB konnten somit problemlos die Aufgaben zur Überarbeitung der Dokumente angehen. Mit einem tiefen Verständnis für den aktuellen Stand des Projekts konnte sie effektiv an der Aktualisierung der Dokumente

arbeiten. Dabei achteten sie darauf, dass alle neuen Informationen und Anforderungen angemessen in die Dokumente integriert wurden. Durch diese enge Zusammenarbeit und ständige Kommunikation innerhalb des Teams konnten wir sicherstellen, dass die Dokumentation den aktuellen Stand unseres Projekts genau widerspiegelte.

Backend mit Frontend verbinden

Für das Meeting mit dem Auftraggeber am 01.06.23 wurde eine vorläufige Oberfläche präsentiert, um den bisherigen Stand visualisiert zu veranschaulichen. Das Backend wurde erfolgreich mit dem Frontend verbunden. Am 09.06.23 wurden dem Auftraggeber die Implementierungen der Loginansicht und der Einstellungsansicht in der Dozentenansicht vorgelegt. Es wurde vereinbart, in der nächsten Phase eine fertige Webseite bereitzustellen, damit der Auftraggeber die Funktionen eigenständig testen und Wünsche äußern kann.

Präsentation für Entwurf/ Test erstellen

Die Ausarbeitung der Präsentation des Entwurfs war ein Gemeinschaftswerk der gesamten Gruppe. Dabei wurde uns bewusst, dass wir die Dokumente aus SE I vernachlässigt hatten und unsere Implementierung auf unvollständigen Notizen und teilweise aus theoretischen Beschreibungen aus den SE I-Dokumenten basierte. Es war daher dringend erforderlich, die SE I-Dokumente zu überarbeiten. Obwohl wir bisher keine Zeit für umfassende Tests hatten, haben wir uns dennoch Gedanken darüber gemacht, welche Bereiche getestet werden sollten, und ein Testprotokoll für manuelle Tests erstellt, damit die zukünftige Bearbeitung der Testfälle zeitlich weniger Aufwand mit sich zieht.

Entwicklerdokumentation beginnen

Die Dokumentationen sind leider etwas zu kurz gekommen. Ursprünglich hatten wir geplant, die Software fertigzustellen und dann innerhalb von anderthalb Wochen die Dokumentationen zu verfassen. Glücklicherweise hat PB bereits mit der Erstellung der Entwicklerdokumentation begonnen, was uns einen kleinen Vorsprung verschaffte.

Projektdurchführung im Projektbericht schreiben

Die Projektdurchführung wurde von ED sorgfältig dokumentiert und kontinuierlich an die jeweiligen Phasen der Gruppe angepasst. Bei Fragen oder Unsicherheiten hinsichtlich der Vorgehensweise bei den Aufgabenstellungen suchte sie aktiv den Kontakt zu ihren Teammitgliedern und gemeinsam erarbeiteten wir die bestmögliche Herangehensweise.

Bewertung

Diese Phase diente als letzter Versuch, unsere Probleme aus der vorherigen Phase zu lösen und das Projekt zum Abschluss zu bringen. Leider waren unsere Bemühungen nicht so erfolgreich, wie wir es

uns erhofft hatten. Obwohl wir einige kleine Fortschritte erzielen konnten, war die Webseite insgesamt nicht vollständig funktionsfähig und bedurfte weiterer Verbesserungen. Da wir dem Auftraggeber jedoch ein Produkt präsentieren mussten, einigten wir uns darauf, in der nächsten Phase nur noch kleinere Anpassungen vorzunehmen und die Software trotz fehlender Funktionalitäten abzuliefern.

Wir hatten uns in dieser Phase nur wenig auf das Testen konzentriert, weshalb wir unseren Fokus nun verstärkt darauf legen mussten. Ebenso waren die Dokumentationen kaum vorhanden, was sich ebenfalls negativ auf den Erfolg unseres Projekts auswirkte. Um diese Mängel auszugleichen, waren wir entschlossen, uns in den kommenden Phasen intensiver mit dem Testen und der Dokumentation zu befassen, um das Projekt zum erfolgreichen Abschluss zu bringen.

Essence Navigator

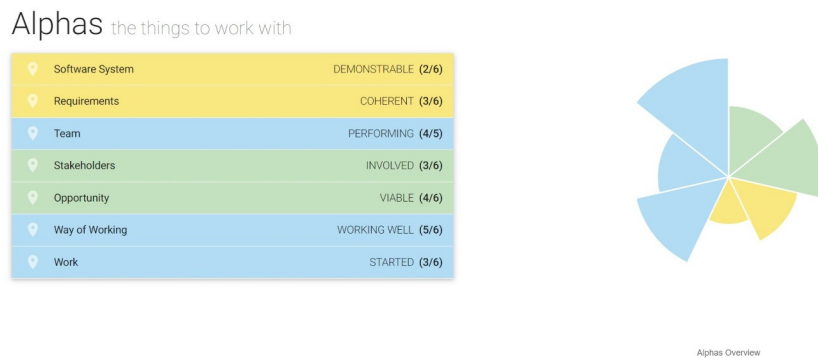


Abbildung 9: Essence Diagramm nach Endphase Implementierung

3.5. Dokumentationsphase

In der abschließenden Phase des Projekts lag unser Hauptfokus auf der Fertigstellung der Dokumentationen und des Projektberichts. Da das Testen zuvor vernachlässigt wurde, mussten wir in dieser Phase zusätzliche Anstrengungen unternehmen, um so viele Tests wie möglich an unserer Software durchzuführen und Fehler in unserem Code zu beheben. Dabei ergaben sich verschiedene Aufgaben und Herausforderungen:

- Abschluss des Projektberichts
- Durchführen von Tests
- Erstellen der Testdokumentation
- Verfassen der Anwenderdokumentation
- Erstellen der Betriebsdokumentation
- Verfassen der Entwicklerdokumentation
- Implementierung der während der Tests aufgetretenen Fehler
- Vorbereitung und Durchführung der Softwarepräsentation

Diese Aufgaben hatten hohe Priorität, da sie entscheidend waren, um eine vollständige und funktionierende Softwarelösung bereitzustellen. Durch eine gründliche Testphase und die sorgfältige

Erstellung der Dokumentationen strebten wir an, das Projekt erfolgreich abzuschließen und den Auftraggeber zufriedenzustellen.

Am 16.06.23 fand ein letztes Gespräch mit dem Auftraggeber vor der Abgabe statt. Wir präsentierten ihm die Webseite und er hatte die Möglichkeit, einige Funktionen zu testen. Dabei wies er erneut darauf hin, dass ein unverschlüsselter Login über den LDAP-Server implementiert werden sollte und dass wir ebenso hart codierte Eingabewerte verwenden sollten. Außerdem äußerte er den Wunsch nach einer Übersicht aller Projekte und kritisierte die uneinheitlichen Begrifflichkeiten. Die zahlreichen Kritikpunkte des Auftraggebers führten zu einem Verlust unserer Motivation. Wir einigten uns darauf, die noch umsetzbaren Anforderungen zu implementieren und ihm zeitnah unser "fertiges" Produkt zu präsentieren.

Am 21.06.23 stellten wir ihm über unser GitHub-Repository unsere Webseite zur Verfügung, damit er sie für das Abnahmeprotokoll testen konnte.

Projektbericht beenden

Im Projektbericht fehlte lediglich noch der Abschnitt "Ergebnisse". Das Festhalten der Ergebnisse stellte sich als schwieriger heraus als erwartet, da die Software zu diesem Zeitpunkt noch nicht vollständig abgeschlossen war und es unklar war, wie weit sie bis zur Abgabefrist fertiggestellt werden würde. Schließlich wurden die Ergebnisse zwei Tage vor der Abgabe dokumentiert.

Zusätzlich dazu musste jedes Teammitglied eine Reflexion verfassen, die auf den Fragen von Herr Anke basierte. Diese Reflexionen ermöglichten es uns, unsere persönlichen Eindrücke und Erfahrungen im Projekt zu reflektieren und daraus wertvolle Erkenntnisse zu gewinnen.

Testen

Das Testen wurde von JW, ED und PB durchgeführt, die sich gemeinsam zusammensetzten und einen Unittest entwickelten. Sie erstellten Testfälle anhand eines vordefinierten Testprotokolls und führten diese parallel durch. Aufgrund der Tatsache, dass viele Anforderungen nicht implementiert wurden bzw. nicht ausreichend implementiert wurden, gestaltete sich das Testen umso herausfordernder.

Wir trafen zudem die Entscheidung, alle zu testenden Objekte zu dokumentieren, um sie für zukünftige Gruppen festzuhalten. Dies ermöglichte es uns, das Wissen und die Erfahrungen aus dem Testprozess weiterzugeben und die Effizienz für kommende Entwicklungsphasen zu steigern.

Dokumentationen schreiben

In Bezug auf die Dokumentenbearbeitung haben wir uns dafür entschieden, Word als unser Hauptwerkzeug zu verwenden. Jedes Dokument erhielt einen Verantwortlichen, der über das entsprechende Fachwissen verfügte und somit am besten geeignet war, das Dokument inhaltlich zu gestalten. Darüber hinaus wurde jedem Dokument ein Nebenverantwortlicher zugewiesen, der bei Fragen, Hilfestellungen und dem Korrekturlesen unterstützen konnte. Die Zuweisung von Nebenverantwortlichkeiten spielte ebenfalls eine wichtige Rolle, da sie uns ermöglichte, eventuelle Lücken zu schließen und Fehler schneller zu erkennen. Durch die gemeinsame Verantwortung konnten wir auch voneinander lernen und uns gegenseitig unterstützen, um die Qualität der

Dokumente zu verbessern. Auf Grund von sprachlicher Barrieren konnte ein Mitglied kaum an Dokumenten schreiben, was wir zu unserem Vorteil nutzten und sie die Korrektur der Testfälle an der Software übernahm und unterstützend die inhaltliche Fülle der Dokumente gewährleistete.

Diese Vorgehensweise hat sich als äußerst effektiv erwiesen, da sie es uns ermöglichte, sowohl an der Softwareentwicklung als auch an der Erstellung der Dokumentation kontinuierlich zu arbeiten. Wir konnten das Fachwissen unserer Gruppenmitglieder optimal nutzen und sicherstellen, dass alle Aspekte des Projekts angemessen abgedeckt wurden.

Am Ende dieser Phase korrigierten wir gemeinsam alle Dokumente auf Inhalt, Rechtschreibung und Grammatik.

Implementieren der Fehler beim Testen

Während des Testens und im Gespräch mit dem Auftraggeber ist uns aufgefallen, dass beim Projektanlegen die ID mehrfach vergeben werden konnte. Weiterhin konnte man Projekte speichern, auch wenn Felder frei gelassen wurden. Diese Fehler wurden an NA weitergetragen, die sie zusammen mit ED, verbesserte. Bei der Behebung dieser Anforderungen stellten wir fest, dass unsere Docker-Container häufig das Frontend verschoben. Das Beheben dieses Problems erforderte erneut viel Zeit und Aufwand. Das Einlesen der CSV-Datei funktionierte ebenfalls nicht mehr. Allerdings hatten wir Bedenken, dass die Integration dieser Anforderung zu neuen Problemen mit dem Docker-Container führen würde, daher haben wir beschlossen, diese Anforderung zu streichen.

Als nächster Schritt wurde die Implementierung des Zurücksetzens der Eingabedaten für das kommende Semester übernommen. Aufgrund der begrenzten Zeit konnten wir jedoch nicht mehr erreichen.

Bewertung Dokumentationsphase

Diese Phase war eine enorme Herausforderung für uns, da wir den Großteil unserer Zeit in sie investiert haben. Trotz des holprigen Weges haben wir gelernt, uns niemals entmutigen zu lassen. Wir konnten weitere Anforderungen erfolgreich umsetzen und alle Dokumente in einer vorzeigbaren Form präsentieren, worauf wir sehr stolz sind. Während dieser Phase ist unser Team noch enger zusammengerückt und die gegenseitige Unterstützung wurde besonders deutlich.

Obwohl die Software nur teilweise den Anforderungen entspricht, haben wir in der kurzen Zeit viel erreicht.

Essence Navigator

Alphas the things to work with

Software System	DEMONSTRABLE (2/6)
Requirements	ACCEPTABLE (4/6)
Team	ADJOURNED (5/5)
Stakeholders	INVOLVED (3/6)
Opportunity	VIALE (4/6)
Way of Working	RETIRED (6/6)
Work	STARTED (3/6)



Alphas Overview

Abbildung 10: Essence Diagramm nach Dokumentationsphase

4. Ergebnisse

4.1. Projektergebnisse

Das Ziel unseres Projekts bestand darin, eine Webseite zu entwickeln, die zukünftig im Software-Engineering-Modul eingesetzt werden kann. Die Software sollte dazu dienen, Gruppen für Software-Engineering-Projekte zu generieren.

Das System hat einen separaten Studenten- und Dozentenlogin, jedoch sind die Zugangsdaten hart codiert und nicht wie verlangt mittels LDAP Server integriert.

Ein Student kann den Fragebogen ausfüllen und seine Eingabedaten speichern, sowie zu einem späteren Zeitpunkt einsehen und gegebenenfalls ändern. Die Anforderung den Fragebogen um die Vorqualifikation einer Fachinformatikausbildung zu erweitern wurde erreicht.

Der Dozent kann Projekte erstellen, bearbeiten und löschen. Die gewünschten Anforderungen ein separates ID- Feld anlegen zu können und Pflichtfelder anzupassen wurden erreicht. Ebenfalls kann er eine CSV Datei mit allen Studierenden, die an dem Modul teilnehmen möchten, hochladen.

Das System besitzt eine Datenbank, die alle zuvor genannten Daten speichert und an gewünschten Stellen ausgibt. Weiterhin können alle Daten für ein neues Semester zurückgesetzt werden.

Die Software besitzt eine schmalere Oberfläche mit einer vereinfachten Menüführung. Der Platz auf der Webseite wurde ebenfalls besser ausgenutzt.

Ebenfalls ist ein Algorithmus hinterlegt.

Aus unserer Sicht wurden folgende Use Cases erreicht: Projekte verwalten, Fragebogen ausfüllen, Projekte löschen und Dateien zurücksetzen.

Folgende Anforderungen wurden nicht erreicht:

Die Software kann keine Teams generieren. Der hinterlegte Algorithmus wurde falsch implementiert, sodass man die Richtigkeit des Algorithmus nicht testen kann. Ebenso kann man nicht abschätzen, ob die erstellte Ansicht der Teams implementiert wurde, da keine Teams generiert werden können.

Das System kann keinen Login mittels LDAP Server herstellen. Zeitweise funktionierte diese Anforderung unverschlüsselt, diese wurde jedoch ohne Rest API implementiert, was zu Problemen mit Docker geführt hat. Die Behebung dieses Fehlers hätte zu erneuten Verschiebungen der Oberfläche geführt und hätte das Projektergebnis stark gefährdet.

Die Software kann Teams nicht manuell bearbeiten, da bislang keine Teams generiert werden konnten. Ebenso können aus diesem Grund Projektteams nicht zurückgesetzt werden.

Die Use Cases Teams generieren, User Login, Teams manuell bearbeiten, Projektteams zurücksetzen und Backup erstellen/laden wurden nur teilweise oder fehlerhaft implementiert, sodass sie nicht funktionsfähig sind. Insgesamt wurde das Ziel, die Software im nächsten Semester einzusetzen, nicht erreicht. Der Hauptgrund dafür war der Zeitmangel.

4.2. Gründe für Nichterreichung der Ziele

Die Entscheidung zur Neuentwicklung fiel sehr spät, sodass uns nur etwa zweieinhalb Monate blieben, um die Webseite zu implementieren. In dieser Zeit mussten jedoch auch Dokumente erstellt werden und wir mussten uns in neue Tools einarbeiten. Es ist ebenfalls zu bemerken, dass einige Teammitglieder nur wenig an der Implementierung beteiligt waren. Dies wurde damit begründet, dass diesen Mitgliedern das nötige Wissen für die Implementierung fehlte. Daher entschieden wir uns, sie bei der Dokumentbearbeitung einzusetzen, um so viel wie möglich zu erreichen und keine Zeit und Ressourcen zu verschwenden.

Eine weitere Herausforderung bestand in den Sprachbarrieren innerhalb der Gruppe. Das fachliche Wissen war zwar vorhanden, aber die genauen Anforderungen konnten nicht immer verstanden werden, wodurch einige Aufgaben unvollständig waren. Dies wiederum führte dazu, dass uns die Zeit für Verbesserungen fehlte. Ein weiterer Grund für das Nichterreichen des Ziels war, dass zu Beginn von SE II, als unsere Organisation noch nicht ausgereift war und viele Dopplungen bei der Aufgabenbearbeitung auftraten. Es gab zum Beispiel zeitweise mehrere Login-Ansichten. Zudem investierte ein Teammitglied viel Zeit in Docker, während ein anderes Mitglied bereits Docker installiert hatte, da es dringend benötigt wurde. Dadurch gingen wertvolle Ressourcen verloren, die an anderen Stellen möglicherweise effektiver eingesetzt worden wären.

Das Ziel wurde auch nicht erreicht, da die Aufgabe äußerst komplex war und in vielen Bereichen spezifisches Wissen erforderlich war, wie zum Beispiel bei der Implementierung des LDAP-Servers. Die Bearbeitung dieser Aufgabe nahm viel Zeit in Anspruch, ohne dass nennenswerte Fortschritte erzielt wurden.

In der letzten Phase wurde die Software noch verschiedenen Tests unterzogen, wobei sich unser Verdacht bestätigte, dass die Webseite noch viele Fehler und Ungenauigkeiten aufwies. Das Testen der Webseite erfolgte ebenfalls erst spät aufgrund des Zeitmangels und weil jedes Mitglied zu diesem Zeitpunkt wichtigere Aufgaben zu erledigen hatte. Die Ergebnisse des Tests werden in der Testdokumentation festgehalten.

Das Ergebnis der Abnahme durch den Auftraggeber wird nachgereicht, da die Abnahme erst am 29.06.23 stattfindet.

4.3. Fazit

Abschließend können wir sagen, dass die Webseite noch erhebliches Verbesserungspotenzial aufweist. Jedoch wurde unser Ziel, eine solide Basis für eine Weiterentwicklung zu schaffen, erreicht. Das Backend haben wir nach bestem Wissen strukturiert und dynamisch implementiert, sodass es eine hohe Erweiterbarkeit aufweist. Alles in Allem ist die Gruppe mit dem Ergebnis zufrieden, da wir wissen, wie viel Mühe wir investiert haben und stolz auf die Erkenntnisse sind, die wir aus dem Projekt gewonnen haben.

4.4. Reflexionen

Reflexion Evelyn Dietrich

Über das gesamte Modul Software Engineering habe ich die Position der Teamleiterin übernommen, da ich mir in Zukunft auch eine Tätigkeit in diesem Bereich vorstellen kann. Ich habe gelernt, dass diese Aufgabe sehr viel Verantwortung mit sich trägt. Man muss Durchsetzungskraft haben, aber gleichzeitig ein gesundes Arbeitsklima schaffen können. Diese Balance ist nicht immer einfach, da eine falsche Kommunikation mit den Teammitgliedern leicht zu Missverständnissen und somit zu einem schlechten Arbeitsklima führen kann. Ebenso konnte ich mein fachliches Wissen etwas aufbessern. Meine Vorerfahrungen im Bereich Informatik waren kaum gegeben, aber durch die Position als Teamleiterin muss man sich in jedem Aufgabenbereich einen Überblick verschaffen, um die Richtigkeit der Aufgaben zu überprüfen. Dies war auch herausfordernd, aber hilfreich für die Zukunft. Weiterhin habe ich gelernt, dass die Zusammenarbeit mit Auftraggebern sich schwieriger gestaltet, als man vermutet. Es können nicht immer die Anforderungen erfüllt werden auf Grund von Unwissenheit des Teams oder zu hohen Erwartungen des Auftraggebers. Man muss mit ständiger Kritik und keinem Lob vom Auftraggeber umgehen können und sich ein hartes Fell zulegen, da dieser nicht immer die Mühen und Erfolge der Gruppe zu schätzen weiß. Weiterhin muss man damit rechnen, dass man bei Unwissenheit innerhalb der Gruppe keine Hilfestellungen erwarten kann und selber irgendwie eine Lösung finden muss. Aber dieser schwere Weg zum Ziel hat mich stressresistenter werden lassen, was auch seine Vorteile hat.

Ich bin sehr stolz auf die Zusammenarbeit und den Zusammenhalt der Gruppe. Wir haben uns in jeder Situation, bei jedem mental breakdown und bei allen Schwierigkeiten unterstützt und konnten jederzeit auf die Hilfe innerhalb der Gruppe bauen, was allen Hoffnung und Optimismus brachte. Ich bin ebenso auf das Ergebnis unserer Arbeit stolz. Man muss berücksichtigen, dass wir von Anfang an eine Seite bekommen haben die so gut wie gar nicht ausbaufähig war und wir uns auf Grund dessen für eine Neuentwicklung entscheiden mussten. Wir mussten innerhalb von 3 Monaten alle Dokumente bearbeiten, 5 neue Schreiben und eine ganze Webseite auf die Beine stellen. Wir haben uns selber die Hilfe gesucht, die wir brauchten und kriegen konnten und haben selber Lösungen

gefunden. Daher bin ich umso stolzer auf unser Ergebnis und auch, wie die Gruppe mit der Situation umgegangen ist.

Gut funktioniert hat das Verteilen der Aufgaben und die Bearbeitung der Aufgaben zusammen. Die Methode sich einmal die Woche zu treffen und gemeinsam am Projekt zu arbeiten, kann ich nur weiterempfehlen. Auch das Einigen auf Tools und Vorgehensweisen und generelle Einigungen innerhalb der Gruppe verliefen einwandfrei, da das Arbeitsklima so angenehm war.

Beim nächsten Projekt würde ich (wenn es eine Weiterentwicklung ist) definitiv als erstes die Vorgängergruppe kontaktieren und mir anhören, was diese zu der Software zu sagen haben, da nur sie die Komplexität, den Aufwand und die Möglichkeit der Weiterentwicklung abschätzen können. Weiterhin würde ich von Anfang an das Tool Projects in GitHub nutzen. Es erleichtert die Arbeit ungemein und sorgt für den perfekten Überblick über die Gesamtsituation, aber auch über die einzelnen Iterationsphasen. Ein anderer Punkt, der von unserem Auftraggeber kritisiert wurde, war, dass er unsere Fortschritte nicht sehen konnte. Das lag vor Allem daran, dass wir zunächst nur am Backend gearbeitet haben und man dieses dem Auftraggeber nur schwer verständlich nahe bringen kann. Daher würde ich beim nächsten Projekt darauf achten, dass das Frontend von Anfang an eine genauso große Rolle spielt, damit man dem Auftraggeber bei jedem Meeting einen kleinen Fortschritt visualisiert darstellen kann.

Insgesamt bin ich dankbar für die wertvollen Erkenntnisse, die ich aus diesem Modul gewonnen habe. Es hat meine Fähigkeiten als Teamleiterin gestärkt und mir ein tieferes Verständnis für die Herausforderungen und Möglichkeiten im Bereich Software Engineering vermittelt. Ich bin zuversichtlich, dass ich diese Erfahrungen in meiner zukünftigen beruflichen Laufbahn nutzen kann und freue mich darauf, weitere Projekte gemeinsam mit engagierten Teammitgliedern umzusetzen.

Reflexion Julius Wyrembek

Während der Durchführung dieses Informatikprojekts habe ich eine Vielzahl von Lernmöglichkeiten und Herausforderungen erlebt. Eine der wichtigsten Lektionen, die ich gelernt habe, betrifft die Beherrschung neuer Programmiersprachen und Tools. Da wir alle unerfahren waren und wenig Vorwissen hatten, mussten wir uns intensiv in neue Tools und Sprachen einarbeiten. Diese Fähigkeit, sich schnell in neue Technologien einzuarbeiten, ist von unschätzbarem Wert und wird in der Zukunft von großer Bedeutung sein. Die Auseinandersetzung mit neuen Tools und Programmiersprachen hat nicht nur dazu beigetragen, das aktuelle Projekt abzuschließen, sondern bietet auch einen langfristigen Nutzen. Die erworbenen Fähigkeiten und Kenntnisse können in zukünftigen Projekten und beruflichen Herausforderungen angewendet werden. Die Bereitschaft, sich mit neuen Technologien vertraut zu machen und diese effektiv einzusetzen, eröffnet neue Möglichkeiten und erweitert das eigene Fachwissen. Ein weiterer Aspekt, auf den ich stolz bin, ist die Entwicklung der Team-Website selbst. Trotz der anstrengenden Natur des Projekts haben wir es geschafft, eine Website zur Erstellung von Teams zu entwickeln die zwar nicht vollständig nutzbar ist, aber in Zukunft die Möglichkeit bietet eine funktionierende Website fertigzustellen. Wir konnten die neu erworbenen Fähigkeiten und Kenntnisse in der Praxis anwenden, auch wenn am Ende nicht das gewünschte Ziel erreicht werden konnte. Trotz Probleme bei der Fertigstellung hat mir das Projekt

gezeigt, dass unsere Anstrengungen, uns in neue Tools und Sprachen einzuarbeiten, von großem Wert waren und sich gelohnt haben. Ein Aspekt, der gut funktioniert hat, war die Teamarbeit und das Gruppenklima. Obwohl wir alle unerfahren waren, haben wir effektiv zusammengearbeitet und uns gegenseitig unterstützt. Die offene Kommunikation und der Austausch von Ideen haben dazu beigetragen, dass wir die Hindernisse überwinden konnten, mit denen wir konfrontiert waren. Die enge Zusammenarbeit im Team war entscheidend für unseren Vorrangkommen und hat gezeigt, wie wichtig es ist, auf die Stärken jedes Einzelnen zu setzen. Bei der Reflexion auf dieses Projekt gibt es jedoch auch Bereiche, in denen wir beim nächsten Mal anders vorgehen sollten. Insbesondere der Umgang mit externen Funktionen, auf die wir keinen Einfluss hatten und die uns Probleme bereitet haben, war eine Herausforderung. Für zukünftige Projekte sollten wir versuchen, frühzeitig mögliche Hindernisse zu identifizieren und alternative Lösungen zu planen. Eine bessere Vorbereitung auf unvorhergesehene Probleme könnte uns helfen, die Auswirkungen solcher Schwierigkeiten zu minimieren. Insgesamt war dieses Projekt eine wertvolle Erfahrung. Es hat mir gezeigt, dass Teamarbeit, kontinuierliches Lernen und eine gute Kommunikation entscheidend für den Erfolg eines Projekts sind. Die erworbenen Fähigkeiten in neuen Tools und Sprachen werden nicht nur in der Zukunft von Nutzen sein, sondern haben bereits dazu beigetragen, ein funktionsfähiges Produkt zu entwickeln. Die gewonnenen Erkenntnisse und Erfahrungen werde ich nutzen, um mich weiterzuentwickeln und meine Herangehensweise an zukünftige Projekte zu verbessern.

Reflexion Nuha Alhamiidi

Im Bereich Frontend habe ich meine Kenntnisse in HTML, JavaScript und CSS erweitert. Ich habe gelernt, wie man ansprechende und benutzerfreundliche Benutzeroberflächen entwickelt, Interaktionen implementiert und das Erscheinungsbild einer Webseite verbessert. Ich habe auch verschiedene Frameworks und Bibliotheken wie React oder Angular verwendet, um die Entwicklung effizienter und skalierbarer Frontend-Anwendungen zu ermöglichen.

Im Backend-Bereich habe ich mich auf Java Spring spezialisiert. Ich habe gelernt, wie man mithilfe des Spring-Frameworks robuste und skalierbare Backend-Services entwickelt. Ich konnte meine Kenntnisse in Java erweitern und verschiedene Spring-Module wie Spring Boot, Spring MVC und Spring Data verwenden, um effektive Lösungen für Backend-Herausforderungen zu implementieren.

In Bezug auf Docker habe ich gelernt, wie man Container erstellt und verwaltet, um Anwendungen in einer isolierten Umgebung auszuführen. Ich habe gelernt, wie man Docker-Images erstellt, Container konfiguriert und die Containerisierungstechnologie effektiv in Entwicklungs- und Bereitstellungsprozesse integriert. Docker hat es uns ermöglicht, Anwendungen unabhängig von der zugrunde liegenden Infrastruktur zu entwickeln und bereitzustellen.

Auf das Team bin ich stolz, weil wir eine starke Zusammenarbeit hatten und gemeinsam großartige Ergebnisse erzielt haben. Jedes Teammitglied hat seine Fachkenntnisse und Fähigkeiten eingebracht und wir haben uns gegenseitig unterstützt, um die besten Lösungen zu finden. Die offene Kommunikation und der respektvolle Umgang miteinander haben zu einer positiven Teamdynamik beigetragen.

Was besonders gut funktioniert hat, war die regelmäßige Kommunikation und der Wissensaustausch im Team. Wir haben regelmäßige Meetings abgehalten, um den Fortschritt zu besprechen,

Herausforderungen anzugehen und Erfahrungen zu teilen. Dadurch konnten wir effizienter arbeiten und unsere individuellen Stärken nutzen, um das Projekt voranzubringen.

Beim nächsten Projekt würde ich versuchen, noch mehr Kontakt und Austausch mit den Ansprechpartnern zu haben. Eine enge Zusammenarbeit und regelmäßige Kommunikation mit den Ansprechpartnern hätte dazu beigetragen, deren Anforderungen und Erwartungen besser zu verstehen und potenzielle Missverständnisse frühzeitig zu klären.

Darüber hinaus würde ich beim nächsten Projekt versuchen, die Projektplanung und -organisation weiter zu verbessern. Dies umfasst das Festlegen klarer Ziele, Deadlines und Zuständigkeiten, um den Projektfortschritt effizienter zu verfolgen. Eine gute Projektplanung ermöglicht es uns, Engpässe und Verzögerungen frühzeitig zu erkennen und entsprechende Maßnahmen zu ergreifen.

Insgesamt bin ich stolz auf das, was das Team erreicht hat, und ich bin zuversichtlich, dass wir aus den Erfahrungen und Herausforderungen des aktuellen Projekts lernen können, um beim nächsten Projekt noch erfolgreicher zu sein.

Reflexion Viktoria Karklin

Im sozialen Aspekt habe ich gelernt, wie man möglichst konstruktiv und konfliktvermeidend miteinander kommuniziert. Des Weiteren habe ich gelernt, eine erledigte Aufgabe aus meinem Fachgebiet anderen Menschen so zu erklären, dass sie grob verstehen können, was ich gemacht habe. Ich habe gelernt, Aufgaben von anderen zu übernehmen oder gemeinsam mit anderen daran zu arbeiten, wenn sie Probleme bereiten und auch meine eigenen Aufgaben mit anderen zu teilen, wenn sie zu anspruchsvoll für mich alleine sind. Schnell merkt man, wie schwierig es ist, gemeinsame Termine für Team-Meetings, Bearbeitungszeit und Coach-/Arbeitgeber-Meetings zu finden. Oft muss man am Wochenende oder zu persönlichen Terminen arbeiten, um das Arbeitspensum zu erreichen. Man lernt, mit Frustration umzugehen, wenn die Team- und Einzelarbeit gut funktioniert, man weit vorankommt und sich große Mühe gibt, gute Ergebnisse zu liefern, die jedoch nicht anerkannt werden.

Technisch gesehen habe ich folgende Themen gelernt/angewandt und musste sie von Grund auf erschließen: Docker, Container, Starten eines lokalen Webservers in Verbindung mit Docker, Bedienung/Einrichtung von virtuellen Maschinen mithilfe der Bash, Verwaltung von GitHub-Repositories und deren Funktionen, AsciiDoc zur Erstellung umfangreicher Dokumente, MySQL, LDAP, HTML, CSS, Bootstrap, SpringBoot und andere Frameworks, Nginx, Apache Server, Python, Verständnis von Proxys, umfangreiche Fehlerbehandlung in verschiedenen Bereichen, Webseitendesign, Zertifikate und Testing. Den Großteil dieses Wissens hatte ich zuvor noch nie oder nicht in diesem Ausmaß an der Hochschule gelernt.

Ich bin stolz auf mein Team, da wir trotz der Aufgabe, die weit über unserem Bildungsstand im Studium lag, niemals die Motivation verloren haben, uns dieser Herausforderung zu stellen. Wir haben alle einen großen Teamgeist gezeigt und eine fleißige Arbeitsweise an den Tag gelegt. Wir haben Ergebnisse erzielt, da wir wöchentlich pro Mitglied ungefähr 6 Stunden investiert haben. Bis auf wenige Eingriffe durch Ruben haben wir alles vollkommen eigenständig erarbeitet. Wir hatten einen erschwerenden Faktor durch unser Vorgängerteam: eine Webseite, die wir neu implementieren mussten, obwohl das vorherige Team und andere Dritte davon abrieten, nachdem wir bereits ein

Semester daran gearbeitet hatten. Diese und viele weitere Gründe machen mich sehr stolz auf mein Team, mich selbst und die Arbeit, die wir geleistet haben.

Sollte ich erneut an einem Projekt arbeiten, würde ich sofort die Kommunikationswege klären und Regeln für das GitHub-Repository aufstellen. Sobald wir unsere Aufgaben erhalten, würde ich sie von einem Spezialisten einschätzen lassen, der die Arbeitszeit für einen Neuling abschätzen kann. Diese geschätzte Zeit würde ich dann dem Arbeitgeber mitteilen, damit er entscheiden kann, ob er diese Zeit akzeptieren kann oder ob ihm ein anderer Bereich seines Projekts wichtiger ist. Sollte der Arbeitgeber ein Tool voraussetzen, das im Studium nicht gelehrt wurde (z. B. Docker), würde ich genau zwei Stunden für Recherchen nutzen. Wenn ich dann immer noch keine Ahnung von dem Tool habe oder es nach sechs Stunden Lernzeit nicht einsetzen kann, würde ich sofort den Arbeitgeber, zuständige Mentoren und Kommilitonen um Hilfe bitten. Wenn das Problem innerhalb einer Woche nicht gelöst ist, würde ich nicht noch einmal den Fehler machen, wertvolle Arbeitszeit auf Tools zu verschwenden, die nicht in einem angemessenen Zeitrahmen für das Projekt erlernt werden können, und versuchen, eine Alternative zu finden. Wenn es keine Alternative gibt, müsste ich dies dem Arbeitgeber mitteilen.

Reflexion Phil Beck

Als ich zum ersten Mal von Prof. Dr.-Ing. Anke hörte, der behauptete, dass das Modul Software Engineering einen erheblichen Zeitaufwand erfordern würde, ließ ich mich zunächst nicht weiter davon beeindrucken. Ich war skeptisch und dachte, dass es sich um die übliche akademische Rhetorik handeln würde, die in der Praxis kaum Nutzen hätte.

Im Rückblick lässt sich zweifellos feststellen, dass Prof. Anke mit seiner Einschätzung bezüglich des Zeitaufwands absolut ins Schwarze getroffen hat. Es war eine wahre Herausforderung, den Anforderungen des Moduls gerecht zu werden und gleichzeitig den umfangreichen Lehrplan zu bewältigen. Der Stundenplan war gespickt mit Vorlesungen, Praktika und regelmäßigen Iterationstreffen, die es erforderten, unseren Kalender sorgfältig zu koordinieren. Doch das war längst nicht alles. Die eigentliche Essenz des Moduls lag in der intensiven Projektarbeit, die uns unzählige Stunden in der Woche abverlangte. Es war ein wahrer Balanceakt zwischen Aufgaben aus anderen Kursen und dem unermüdlichen Einsatz, den wir in das Projekt stecken mussten. Es gab Momente, in denen wir uns fragten, ob es überhaupt möglich war, alles unter einen Hut zu bringen. Allerdings gewinnt man durch diese intensive Erfahrung nicht nur ein enormes praktisches Wissen, sondern auch wertvolle Einblicke und Fähigkeiten, die sich als äußerst nützlich erweisen werden, wenn es darum geht, zukünftige Projektarbeiten im Bereich des Software Engineerings erfolgreich umzusetzen. Das Modul ermöglichte uns, die theoretischen Konzepte in die Praxis umzusetzen und eine Vielzahl von Herausforderungen zu meistern, die uns in der realen Welt der Softwareentwicklung begegnen werden.

Meine drei Top Learnings aus diesen 2 Semestern waren:

1. Eine gute Zusammenarbeit im Team ist entscheidend, um den Unterschied zwischen Erfolg und Niederlage zu machen. Beispielsweise waren die Meetings mit den Auftraggebern und Coaches manchmal demotivierend, da sie unsere Fehler genau aufzeigten und uns den noch langen Weg verdeutlichten. Ein weiteres Beispiel ist, dass wir Mitte des 2. Semesters die gesamte Infrastruktur

überdenken mussten und größtenteils von vorne anfangen mussten. Dadurch konnten wir nicht alle Aufgaben bewältigen, aber ich hatte anfangs kaum für möglich gehalten, dies überhaupt umzusetzen. Dennoch hat das gesamte Team hervorragend zusammengearbeitet und sein Bestes gegeben, worauf ich unglaublich stolz bin in Bezug auf die Entwicklung des Teams.

2.

Kürzere Iterationen sind vorteilhafter, da sie sicherstellen, dass jeder stets auf dem neuesten Stand bleibt und Teammitglieder effektiver bei individuellen Problemen unterstützen können. Aufgrund unserer Entscheidung für eine Neuentwicklung hatten wir eine größere Anzahl an Aufgaben zu bewältigen, als ursprünglich geplant, hätten wir uns für eine Weiterentwicklung entschieden. Um den großen Aufgabenberg zu bewältigen, haben wir beschlossen, unsere Iterationstreffen jeden Mittwoch von 09:00 bis 13/14 Uhr abzuhalten. Dank dieser Treffen konnten wir das Projekt schneller vorantreiben als erwartet.

3. Mein bedeutendstes Learning während des Moduls war zweifellos die bahnbrechende Erkenntnis, dass die agile Arbeitsweise ein absoluter Gamechanger in Projekten ist. Die Effektivität und Effizienz, mit der agile Methoden und Prinzipien angewendet werden können, hat mich zutiefst beeindruckt und lässt mich erahnen, dass sie die Zukunft des Projektmanagements maßgeblich prägen werden. Deshalb will ich in Zukunft noch mehr in diese Thematik vertiefen.

Wenn ich das nächste Mal etwas anders machen würde, gäbe es zwei wesentliche Änderungen. Zum einen würde ich die Iterationszeiträume von Anfang an auf eine Woche verkürzen, anstatt auf zwei Wochen. Zum anderen würde ich zuallererst eine Bestandsanalyse durchführen, die mindestens zwei bis drei Wochen dauert. Dabei würde ich beispielsweise untersuchen, welche Kenntnisse für welche Aufgaben erforderlich sind, wer welche Fähigkeiten besitzt und vor allem den Code gründlich analysieren. Es wäre wichtig zu überprüfen, ob wir mit den vorhandenen Technologien arbeiten können und ob der Code sauber geschrieben ist.

Abschließend kann ich sagen, dass das Modul Software Engineering eine außerordentliche Herausforderung war, die mir jedoch unzählige Erkenntnisse und Fähigkeiten vermittelt hat. Trotz des anfänglichen Zweifels an der Bedeutung und dem Zeitaufwand des Moduls wurde mir klar, dass Prof. Anke mit seiner Einschätzung vollkommen richtig lag. Die intensive Projektarbeit, die Zusammenarbeit im Team und die Anwendung agiler Methoden haben mir wertvolle Lektionen gelehrt.

Reflexion Vincent Weise

Während der Durchführung meines Software Engineering Projekts konnte ich viele wertvolle Erfahrungen sammeln und eine Vielzahl von Erkenntnissen gewinnen. In meiner Reflexion möchte ich die wichtigsten Punkte, die mich während des Projekts beeinflusst haben, hervorheben.

Eine entscheidende Erkenntnis war die Bedeutung einer gut funktionierenden Gruppe. Ich hatte das Glück, im zweiten Semester in einem Team zu arbeiten, in dem die Mitglieder kommunikativ und kooperativ waren. Diese Gruppe stellte das genaue Gegenteil der Gruppe des ersten Semesters dar, worüber ich sehr glücklich war. Diese positive Atmosphäre hat es uns ermöglicht, effizient zusammenzuarbeiten und unsere Aufgaben erfolgreich zu erledigen. Im Laufe der Zeit entwickelte sich ein gutes Gruppenklima, was das Arbeiten noch angenehmer machte.

Die Aufgabenteilung war ein weiterer Aspekt, der zum Erfolg unseres Projekts beigetragen hat. Wir haben uns gegenseitig unterstützt und dafür gesorgt, dass jeder seinen Teil beigetragen hat. Wenn Probleme auftraten, wurden die Aufgaben offen kommuniziert und an andere Teammitglieder weitergegeben, um eine zügige Lösung zu finden. Dieses kooperative Vorgehen hat uns geholfen, Herausforderungen erfolgreich zu bewältigen.

Jedoch stellte sich heraus, dass das Projekt wesentlich mehr Zeit in Anspruch nahm als erwartet. Es wurde sehr anstrengend, da wir mit unvorhergesehenen Schwierigkeiten konfrontiert wurden. Insbesondere stellten sich die Erarbeitung von verschiedenen Elementen und auch Techniken zum Erreichen eines solchen Projektes als erheblich schwieriger heraus als angenommen. Da wir alle ein so großes Projekt zum ersten Mal durchführten, hatten wir wenig Erfahrung mit vielen Aspekten einer solchen Gruppenarbeit und vieles war wesentlich komplexer zu erlernen als wir es bisher in Praktika gelernt hatten. Wir hatten teilweise so komplexe Probleme, dass wir nicht mehr wussten an wen wir uns wenden sollten, um eine geeignete Lösung zu finden. Dies führte zu einer großen Belastung und ließ die Motivation zum Ende hin etwas nachlassen.

Trotz dieser Herausforderungen habe ich eine klarere Vorstellung davon erhalten, wie Software entwickelt wird. Ich konnte wichtige Einblicke in den Entwicklungsprozess gewinnen und habe gelernt, dass eine gründliche Dokumentation von entscheidender Bedeutung ist. Mir wurde klar warum es wichtig ist eine solide Planung und Struktur vorher zu erstellen, um ein erfolgreiches Ergebnis zu erzielen und es nicht funktioniert ungezielt Komponenten zu programmieren.

Insgesamt war das Software Engineering Projekt eine wertvolle Erfahrung für mich. Ich konnte nicht nur meine technischen Fähigkeiten verbessern, sondern auch wichtige Erkenntnisse über Teamarbeit, Kommunikation und den Entwicklungsprozess gewinnen. Obwohl es anstrengend war und meine Motivation zeitweise schwankte, bin ich stolz darauf, was wir als Team erreicht haben. Diese Erfahrung hat mich nachhaltig geprägt und wird sicherlich einen positiven Einfluss auf meine zukünftigen Projekte haben.