

Bachelorarbeit

**Eine gamifizierte Howto-App für
Bachelorarbeiten**

vorgelegt von

Tim-Pascal Lau

Ausgabedatum: 24. Mai 2018
Abgabedatum: 24. August 2018

Aufgabenstellung

Für Studierende im letzten Semester eines Bachelorstudiengangs umfasst **DIE** wesentliche Prüfungsleistung das Verfassen einer Bachelorarbeit. Die Auseinandersetzung mit komplexen Problemstellungen stellt jedoch erfahrungsgemäß für viele Studierende eine große Herausforderung dar, welche sich aus dem erstmaligen Zusammenspiel von selbständiger und eigenverantwortlichem Arbeiten sowie Problemlösen mittels erworbener Fach- und Methodenkenntnisse über einen längeren (dreimonatigen) Zeitraum ergibt.

Im Verlauf des Studiums sollten Studierende folgendes Wissen und folgende Fähigkeiten erworben haben und zielgerichtet anwenden können:

- Studiengang-spezifisches Grundlagenwissen
- Wissensansammlung über fachspezifische Methoden und deren Eigenschaften
- Die Fähigkeit, komplexe Probleme zu erkennen, zu strukturieren und systematisch mittels geeigneter Methoden zu bearbeiten

Es kommt im Kontext von Bachelorarbeiten dennoch oftmals zu Schwierigkeiten, das erworbene Wissen und Fähigkeiten zielgerichtet auf reale Probleme anzuwenden und deren Ergebnisse zusammenhängend zu dokumentieren.

Lösungsansatz

Ein möglicher Lösungsansatz wäre es, Studierenden eine Software bereitzustellen, welche unterstützend und wegweisend bei dem systematischen Vorgehen für komplexe Problemstellungen fungieren könnte. Hierbei soll es nicht darum gehen, dem Studierenden die eigentliche Arbeit abzunehmen, sondern vielmehr darum, Studierende hinsichtlich Vorgehen und Methodenauswahl zielgerichtet zu unterstützen.

Im Rahmen dieser Bachelorarbeit soll hierzu eine mobile Applikation entwickelt werden, die den Studierenden während der Dauer seiner Bachelorarbeit kontinuierlich „begleitet“. Dabei sollen Gamificationansätze realisiert werden, die motivierend im Rahmen einer Bachelorarbeit wirken sollen. Dies soll beispielhaft für den Studiengang Informatik/Softwareentwicklung erfolgen. Eine Erweiterbarkeit für andere Studiengänge ist hierbei jedoch von Anfang an konzeptionell vorzusehen.

Primäre Funktion der Software ist Studierenden bei den folgenden Aufgaben begleitend zu unterstützen und fortwährend zu motivieren “am Ball zu bleiben“:

- Brainstorming (zur Unterstützung der Ideenfindung für Bachelorarbeiten)
- Recherche und Literaturverwaltung
- Gliederung (unterschiedlicher Kategorien von Bachelorarbeiten, zum Beispiel mittels bewehrter Templates)
- Zeitplanung/Fortschrittsverfolgung sowie Erinnerungs- und Benachrichtigungsfunktion
- Problem-orientierte Anforderungsanalyse und deren Dokumentation
- Problem-orientierte Methoden- und Tool-/Framework-Selektion und deren Dokumentation
- Methoden-spezifische Aufbereitung von Ergebnissen
- Problem-orientierte Nachweisführung und deren Dokumentation

Die App soll mit Flutter für Android und iOS entwickelt werden. Dabei soll erhoben werden, inwiefern sich Flutter für die Entwicklung solcher Apps eignet (Lessons Learned).

Teilaufgaben

- Detaillierte Anforderungsanalyse oben angegebener Funktionen. Hierbei sind Studenten und Professoren des Studiengangs Informatik/Softwaretechnik geeignet einzubeziehen und relevante Literatur (insbesondere zu Gamification und Methoden der Informatik und des Softwareengineering) zu berücksichtigen.
- Architekturentwurf der Anwendung (Erweiterbarkeit für andere Studiengänge ist konzeptionell vorzusehen)
- Implementierung der Anwendung
- Die Funktionsfähigkeit der App soll mittels Softwaretests geeignet nachgewiesen werden.
- Die Nutzbarkeit der App soll systematisch evaluiert werden. Hierbei sind Studenten und Professoren des Studiengangs Informatik/Softwaretechnik geeignet einzubeziehen.
- Dokumentation der oben angegebenen Schritte inklusive Bewertung der Nutzbarkeit des Frameworks Flutter für solche Arten von Apps.

Es wird empfohlen die Dokumentation von Anfang an begleitend zur Arbeit zu erstellen und sich an einem Dokumentations-Template - wie bspw. diesem - zu orientieren.

Erklärung zur Bachelorarbeit

Ich versichere, dass ich die Arbeit selbstständig, ohne fremde Hilfe verfasst habe.

Bei der Abfassung der Arbeit sind nur die angegebenen Quellen benutzt worden.
Wörtlich oder dem Sinne nach entnommene Stellen sind als solche gekennzeichnet.

Ich bin damit einverstanden, dass meine Arbeit veröffentlicht wird, insbesondere dass
die Arbeit Dritten zur Einsichtnahme vorgelegt oder Kopien der Arbeit zur Weitergabe an
Dritte angefertigt werden.

(Datum)

Unterschrift

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation	1
1.1.1. Online-Umfrage zur Ermittlung der Bedarfssituation	2
1.2. Lösungsansatz und Zielstellung	4
1.3. Aufgabenbeschreibung	4
1.4. Ausblick auf die Bachelorarbeit	6
2. Beschreibung der Kernkomponenten	7
2.1. Gamification	7
2.1.1. Die Bedeutung von Gamification	7
2.1.2. Bekannte Spiel-Design-Elemente	8
2.1.3. Nutzergruppen nach Bartle	10
2.2. Vorstellung des Frameworks Flutter	12
2.2.1. Flutter Kerntechnologien	12
2.2.2. Plattformübergreifende Entwicklung von mobilen Applikationen . . .	13
2.2.3. Hot-Reload Funktion	14
2.2.4. Flutter Widgets	14
2.2.5. Flutter UI Rendering	15
2.3. Was ist Redux?	16
2.3.1. Actions	16
2.3.2. Reducers	16
2.3.3. Store	17
2.3.4. Flutter Redux Library	17
3. Untersuchung des Problembereiches	18
3.1. Hypothese	18
3.2. Identifikation der Interessengruppen	19
3.3. Wahl der Analysestrategie	20
3.3.1. Einbezug der Professoren	21
3.3.2. Einbezug der Studierenden	21
3.4. Ergebnisse der Professoren	22
3.4.1. Art der Arbeit	23
3.4.2. Erwartungen an den Bacheloranden	25
3.4.3. Häufig auftretende Probleme	27
3.4.4. Die Applikation – Wünsche und Anregungen	28
3.4.5. Die Applikation - Chancen und Risiken	29
3.5. Ergebnisse der Studierenden	31
3.5.1. Erwartungen an den Betreuer	31
3.5.2. Die größten Probleme der Studierenden	32
3.5.3. Wunsch nach besserer Vorbereitung auf die Bachelorarbeit	33

3.5.4. Einbezug der Studierenden in die Gestaltung der Applikation	34
3.6. Untersuchung der Ergebnisse	35
3.6.1. Identifikation des Problembereichs	35
3.6.2. Auswahl der Motivationsstrategie	36
3.6.3. Übersicht der abgeleiteten Produktfunktionen und Anforderungen . .	38
3.6.4. Übersicht der Risiken	40
3.6.5. Übersicht der Chancen	41
4. Konzeptvorstellung der Applikation	42
4.1. Beschreibung der Software	42
4.1.1. Meilensteinverwaltung	43
4.1.2. Bachelorarbeit Guide	44
4.1.3. Herausforderungen	46
4.1.4. Achievements	47
4.1.5. Sonstige Softwareinhalte	48
4.1.6. Dashboard	49
4.1.7. Übersicht der Produktfunktionen	51
4.1.8. Umgang mit Risiken	52
4.2. Beschreibung der Gamification-Elemente	53
4.2.1. Achievements	53
5. Architektur und prototypische Implementierungsdetails	55
5.1. Architekturübersicht	55
5.2. Detaillierte Beschreibung der Model Komponenten	57
5.2.1. AppState-Klasse	57
5.2.2. Milestone-Klasse	59
5.2.3. Task-Klasse	60
5.2.4. Challenge-Klasse	60
5.2.5. Achievement-Klasse	60
5.2.6. Property-Klasse	61
5.2.7. Content-Klasse	62
5.2.8. AppContentLoader-Klasse	63
5.2.9. JSONAppContentFile-Klasse	63
5.2.10. InformationToolContentBuilder-Klasse	63
5.2.11. Achievement-Notification Funktionen	63
5.3. Actions	64
5.3.1. Beispielhafte Demonstration einer Meilensteinbearbeitung	64
5.4. Reducer	65
5.4.1. App-Reducer	65
5.4.2. Beispielhafte Demonstration eines Meilenstein-Reducers	65
5.5. Container	66
5.5.1. Beispielhafte Demonstration einer Meilensteinbearbeitung	66
5.6. Presentation	68
5.6.1. Beispielhafte Demonstration einer Meilensteinbearbeitung	68
5.7. Middleware	69
5.7.1. Beispielhafte Demonstration der Notification Middleware	69

6. Evaluation	71
6.1. Ergebnisse der Softwaretests	71
6.2. Ergebnisse der Nutzerevaluation	72
6.2.1. Zusammenfassung der Testdurchführung mit den Studierenden	73
6.2.2. Zusammenfassung der Testdurchführung mit den Professoren	75
6.2.3. Ergebnis der Nutzerevaluation	77
6.3. Bewertung des Frameworks Flutter zur Entwicklung von mobilen Applikationen	78
6.3.1. Einführung und Dokumentation	78
6.3.2. Verwendete Programmiersprache	78
6.3.3. Handhabung und Anwendung	78
6.3.4. UI-Performance	79
6.3.5. Support und Community	79
6.3.6. Framework-spezifische Features	80
6.3.7. Zusammenfassung der Bewertung	80
7. Präsentation der Ergebnisse	81
7.1. Wichtigsten Erkenntnisse der Arbeit	81
7.2. Schwierigkeiten bei der Bearbeitung	82
7.3. Erfüllung der Aufgabenstellung	82
7.3.1. Unterstützung bei Bearbeitung der Bachelorarbeit	82
7.3.2. Unterstützung bei Zeitplanung	83
7.3.3. Motivation der Studierenden	83
7.4. Ausblick des Projekts	84
7.4.1. Ausblick auf weitere Untersuchungen	84
7.4.2. Ausblick auf Softwarefunktionen	84
Tabellenverzeichnis	86
Abbildungsverzeichnis	87
Listings	88
A. Anhang	89
A.1. Befragung der Beteiligten	90
A.1.1. Transkripte der Interviews mit den Professoren	91
A.1.2. Zusammenfassung der schriftlichen Befragung der Studierenden	128
A.1.3. Ergebnisse der Online-Umfrage für die Studierenden	135
Literatur	147

1. Einleitung

Für Studierende im letzten Semester eines Bachelorstudiengangs umfasst die wesentliche Prüfungsleistung das Verfassen einer Bachelorarbeit. Die Auseinandersetzung mit komplexen Problemstellungen stellt jedoch erfahrungsgemäß für viele Studierende eine große Herausforderung dar, welche sich aus dem erstmaligen Zusammenspiel von selbständiger und eigenverantwortlichem Arbeiten sowie Problemlösen mittels erworbener Fach- und Methodenkenntnisse über einen längeren (dreimonatigen) Zeitraum ergibt.

1.1. Motivation

Im Verlauf des Studiums sollten Studierende folgendes Wissen und folgende Fähigkeiten erworben haben und zielgerichtet anwenden können:

- Studiengang-spezifisches Grundlagenwissen
- Wissensansammlung über fachspezifische Methoden und deren Eigenschaften
- Die Fähigkeit, komplexe Probleme zu erkennen, zu strukturieren und systematisch mittels geeigneter Methoden zu bearbeiten

Des Weiteren stellt eine Online-Umfrage den Teil der Datenerhebung dar, die quantitative Ergebnisse erzielt und deren Aussage somit eine nicht durch die schriftlichen Befragungen abgebildete Menge darstellt.

Die Umfrage richtete sich nicht nur an die Studierenden des Studiengangs Informatik/Softwaretechnik, sondern an Studierende im Allgemeinen, welche sich ab dem 4. Semester bereits in der zweiten Hälfte des Studiums befinden und somit schon einen einschätzbareren Erfahrungsschatz aufgebaut haben. Diese Umfrage fließt jedoch nicht mit in die Anforderungsanalyse ein, sondern dient lediglich dazu, ein allgemeines Bild des Bedarfs der zu entwickelnden Software zu ermitteln.

1.1.1. Online-Umfrage zur Ermittlung der Bedarfssituation

Im Vorfeld zur Entwicklung der Applikation wurde eine Online-Umfrage an verschiedenen Hochschulen durchgeführt (siehe Anhang A.1.3), welche als Voruntersuchung des Problemfeldes dient und somit einen ersten Eindruck über den Bedarf an einer Software zur Unterstützung der Studierenden bei der Bearbeitung der Bachelorarbeit liefert.

Insgesamt haben 44 Studierende unterschiedlicher Hochschulen an der Umfrage teilgenommen, wobei sich diese in weitere drei Gruppen aufteilen lassen. Diese Unterteilung erfolgt nach den unterschiedlichen Phasen, in denen sich die jeweiligen teilnehmenden Studierenden zur Zeit der Umfrage befinden.

Im Folgenden werden die wichtigsten Erkenntnisse der zusammengefassten Umfrage (siehe Abbildung 1.1) beschrieben:

- **Vor Beginn der Bachelorarbeit (Ab 4. Semester)**

Teilnehmerzahl: 24

Die Ergebnisse der Befragung von Studierenden, die sich noch vor der Bearbeitung der Bachelorarbeit befinden, zeigen bezüglich der Kenntnisse über Ablauf der Bachelorarbeit und Erwartungen an den Studierenden, sowie deren Einschätzung zu Selbst- und Zeitmanagement ein relativ gleichmäßig verteiltes Bild der Situation auf.

Hierbei fällt auf, dass ein nicht geringer Teil explizit angibt, Schwierigkeiten mit den genannten Aspekten zu haben.

- **Während der Bachelorarbeit**

Teilnehmerzahl: 14

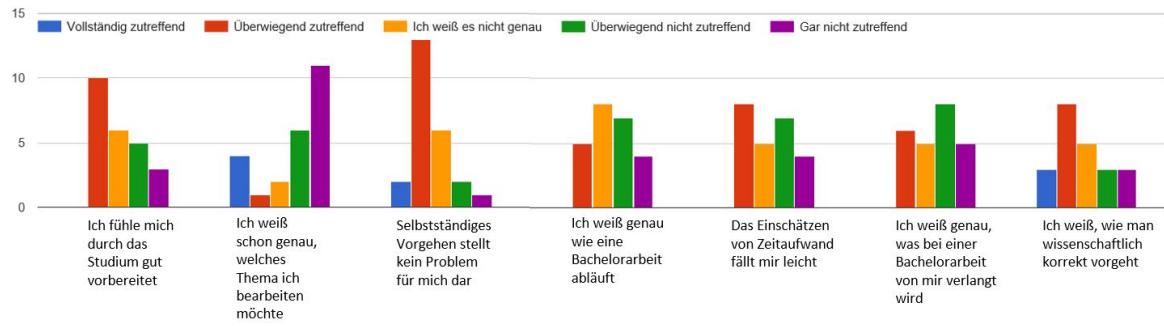
Bei Betrachtung der Umfrageergebnisse der Studierenden, die sich in der Phase während der Bachelorarbeit befinden, ist abzulesen, dass es im Kontext von Bachelorarbeiten unter einem nicht geringen Teil der Studierenden zu Schwierigkeiten kommt.

Hierbei gelten besonders das Zeitmanagement und das wissenschaftliche Arbeiten durchaus als problematisch. Weiterhin sagt über die Hälfte der Studierenden aus, dass sie mit der gesamten Bearbeitung der Bachelorarbeit überfordert sind und ein weiterer Teil, dass sie sich nicht ausreichend mit den Formalien einer Bachelorarbeit auskennen. Viele der Studierende fühlen sich unsicher gegenüber dem, was von ihnen erwartet wird, schätzen sich im Allgemeinen jedoch durch das Studium, als gut vorbereitet ein.

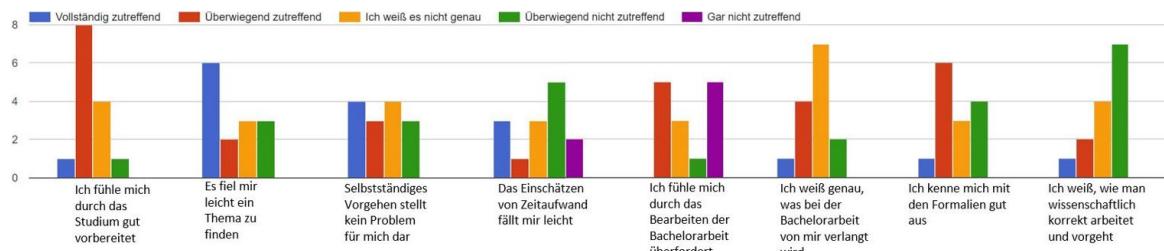
- **Nach Abschluss der Bachelorarbeit**

Teilnehmerzahl: 6

Auffällig einstimmig verhalten sich die Ergebnisse der Studierenden, die sich bereits nach Abschluss der Bachelorarbeit befinden. Das selbstständige Arbeiten sowie das Einschätzen von Zeitaufwand haben nach Aussage der Studierenden offensichtlich wenig bis keine Schwierigkeiten dargestellt. Weiterhin geben die Befragten an, sich bei der Fertigstellung der Bachelorarbeit nicht überfordert gefühlt zu haben. Ein wichtiges Detail ist jedoch, dass über die Hälfte der Befragten angeben, sich durch das Studium nicht ausreichend auf die Bachelorarbeit vorbereitet gefühlt zu haben.



(a) Vor Beginn der Bachelorarbeit



(b) Während der Bachelorarbeit



(c) Nach Abschluss der Bachelorarbeit

Abbildung 1.1.: Online Umfrage: Zusammenfassung der Umfrageergebnisse

Das Gesamtbild der Umfrage zeigt die verschiedenen Schwierigkeiten, mit denen Studierende im Laufe der Bachelorarbeit zu tun haben und stellt diese als wichtige Untersuchungselemente zur Analyse des Problembereichs heraus.

1.2. Lösungsansatz und Zielstellung

Ein möglicher zu verfolgender Lösungsansatz wäre es, eine Software zu entwickeln, welche unterstützend und wegweisend bei dem systematischen Vorgehen unter komplexen Problemstellungen fungieren könnte und auf diese Weise dabei hilft, den Ablauf der Bachelorarbeit für Studierende zugänglicher zu gestalten.

Hierbei soll es nicht darum gehen, den Studierenden die eigentliche Arbeit abzunehmen, sondern vielmehr darum, Studierende hinsichtlich Vorgehen und Methodenauswahl zielgerichtet zu unterstützen.

Mit einem solchen Ansatz soll es Studierenden ermöglicht werden, ihr gelerntes Wissen durch Fokussierung bestimmter Aufgaben und Zusammenhänge im Rahmen ihres eigenen Bachelorprojekts auf die Realität zu übertragen und somit einen motivierenden sowie weiterhin fordernden Rahmen zu schaffen, um ihr Bachelorprojekt erfolgreich abzuschließen.

1.3. Aufgabenbeschreibung

Im Rahmen dieser Bachelorarbeit soll hierfür eine mobile Applikation entwickelt werden, die den Studierenden während der Dauer der Bachelorarbeit kontinuierlich „begleitet“. Dabei sollen Gamificationansätze realisiert werden, welche motivierend bei der Bearbeitung und dem Vorgehen bei der eigenen Bachelorarbeit wirken sollen. Dies soll beispielhaft für den Studiengang Informatik/Softwareentwicklung erfolgen. Eine Erweiterbarkeit für andere Studiengänge ist hierbei jedoch konzeptionell vorzusehen. Primäre Funktion der Software ist, Studierenden bei den folgenden Aufgaben begleitend zu unterstützen und fortwährend zu motivieren „am Ball zu bleiben“:

- Brainstorming (zur Unterstützung der Ideenfindung für Bachelorarbeiten)
- Recherche und Literaturverwaltung
- Gliederung (unterschiedlicher Kategorien von Bachelorarbeiten, zum Beispiel mittels bewehrter Templates)
- Zeitplanung/Fortschrittsverfolgung sowie Erinnerungs- und Benachrichtigungsfunktion
- Problem-orientierte Anforderungsanalyse und deren Dokumentation
- Problem-orientierte Methoden- und Tool-/Framework-Selektion und deren Dokumentation
- Methoden-spezifische Aufbereitung von Ergebnissen
- Problem-orientierte Nachweisführung und deren Dokumentation

Die Applikation soll mittels Flutter für Android und iOS entwickelt werden. Dabei soll erhoben werden, inwiefern sich Flutter für die Entwicklung solcher Apps eignet(Lessons Learned).

Die im Rahmen der Aufgabenbeschreibung entstandenen Anforderungen werden durch die folgenden Teilaufgaben spezifiziert:

- Detaillierte Anforderungsanalyse oben angegebener Funktionen. Hierbei sind Studenten und Professoren des Studiengangs Informatik/Softwaretechnik geeignet einzubeziehen und relevante Literatur (insbesondere zu Gamification und Methoden der Informatik und des Softwareengineering) zu berücksichtigen.
- Architekturentwurf der Anwendung (Erweiterbarkeit für andere Studiengänge ist konzeptionell vorzusehen)
- Implementierung der Anwendung
- Die Funktionsfähigkeit der App soll mittels Softwaretests geeignet nachgewiesen werden.
- Die Nutzbarkeit der App soll systematisch evaluiert werden. Hierbei sind Studenten und Professoren des Studiengangs Informatik/Softwaretechnik geeignet einzubeziehen.
- Dokumentation der oben angegebenen Schritte inklusive Bewertung der Nutzbarkeit des Frameworks Flutter für solche Arten von Apps.

1.4. Ausblick auf die Bachelorarbeit

In der folgenden Dokumentation der Bachelorarbeit sind verschiedene aufeinander aufbauende Prozesse, Teilschritte und Ergebnisse der Softwareentwicklung dokumentiert.

Im Rahmen dieses Projekts werden verschiedene Inhalte aus dem Gebiet der Gamification bezüglich Software-Anwendungen behandelt. Um einen Eindruck über die Umsetzung und Bedeutung von gamifizierten Inhalten in modernen Software-Anwendungen zu bekommen, werden im **Kapitel 2**, die für diese Arbeit benötigten Grundlagen beschrieben.

Weiterhin enthält das Grundlagenkapitel wichtige Informationen für die Nachvollziehbarkeit der für die Entwicklung der Software gewählten Redux-Architektur unter Verwendung von Google's Framework Flutter und dessen Funktionsweise.

Das Durchführen der Anforderungsanalyse im Rahmen einer Softwareentwicklung, ist je nach Ausgangspunkt mit einem enormen Aufwand verbunden, da je nach Komplexität viele verschiedene Personen in ein ihnen fremdes Projekt einbezogen werden. Im Rahmen dieser Projekts haben sich sowohl Professoren der Fachhochschule Lübeck als auch Studierende bereit erklärt, an Interviews und weiteren Einbindungen in das Projekt teilzunehmen und somit das zu untersuchende Problemfeld schrittweise aufzudecken. Die aus der gewählten Befragungsstrategie resultierenden Ergebnisse werden im **Kapitel 3** diskutiert und durch das Ableiten von präzisen Lösungsmöglichkeiten beschrieben.

Basierend auf den abgeleiteten Anforderungen, werden im folgenden Verlauf der Dokumentation die verschiedenen Abstraktionsebenen der Software beschrieben. In **Kapitel 4** wird anhand der Oberfläche der Applikation das Gesamtkonzept unter Einbezug von Hintergründen der gewählten Lösungswege und Designentscheidungen beschrieben, wohingegen in Kapitel 5 die Beschreibung der wichtigsten Details zur technischen Umsetzung der Softwarelösung ausgeführt wird.

Nach der Entwicklung einer Softwarelösung sollte letztendlich immer der wichtige Schritt der Evaluation folgen, welcher in diesem Fall erneut durch das Einbeziehen von Professoren und Studierenden stattfinden konnte. Die Ergebnisse dieser Evaluation, der Softwaretest und die abschließende Bewertung des Frameworks Flutter werden im **Kapitel 6** behandelt.

Jeder dieser beschriebenen Kapitel trägt im Kern die Schwerpunkte einer jeden Softwareentwicklung, welche sich zwischen technischem Fachwissen und Kompetenzen (Hard Skills) und der zwischenmenschlichen Kommunikationsfähigkeit sowie der eigenen Persönlichkeit (Soft Skills) definieren lassen. Nach Durchführung des Projekts wurden durch alle einzelnen Schritte Teil- sowie Gesamtergebnisse erreicht, welche im abschließenden **Kapitel 7** zusammenlaufen und unter Schilderung eigener Erfahrungen diskutiert werden sowie einen Ausblick auf die Zukunft des Projekt ermöglichen.

2. Beschreibung der Kernkomponenten

In diesem Kapitel werden die Grundlagen zur Nachvollziehbarkeit dieser Projektarbeit vermittelt. Es wird somit empfohlen die folgenden Kapitel zu Gamification, Flutter und die für die Beschreibung der Implementierungsdetails relevanten Redux-Architektur zu lesen, um ein Gesamtverständnis für Begriffe und weitere Zusammenhänge des Projekts zu ermöglichen.

2.1. Gamification

Im folgenden Kapitel werden die für diese Arbeit relevante Informationen bezüglich Gamification vermittelt. Hierbei wird erst auf die grundlegende Bedeutung sowie deren Abgrenzung gegenüber verschiedenen Ansätzen eingegangen. Im weiteren Verlauf werden dann tiefgreifendere Informationen zu den verschiedenen Benutzerklassen und den jeweiligen kompatiblen Spiel-Design-Elementen beschrieben.

2.1.1. Die Bedeutung von Gamification

Im Kontext dieser Ausarbeitung wird sich auf die in [13] genannte Definition für den Begriff "Gamification" bezogen:

Gamification is the use of elements of game design in non-game contexts.

Was sich aus dieser Definition ableiten lässt, beschreibt Gamification als das Einsetzen spieltypischer Elemente in einem spielfremden Kontext.

Somit lässt sich ein Kontext bereits als "gamifiziert" beschreiben, sobald Spiel-Design-Elemente in den spezifischen Kontext integriert worden sind, weshalb besonders im Rahmen der Unternehmenseinführung von Gamification-Ansätzen zwischen **Enterprise Gamification** und **Serious Games** unterschieden wird[37, vgl. Kapitel 1.1 Gamification: Definition und Abgrenzung, Seite 4].

Enterprise Gamification bezeichnet das konkrete Einbinden von Gamification-Elementen in Arbeits- und/oder Lernprozessen, mit dem expliziten Ziel der Motivationssteigerung seitens der Anwender.

Gegenüber diesem Ansatz hat sich weiterhin der Begriff **Serious Games** integriert, welcher das explizite Einsetzen von Spielen in einem ernsten Kontext vorsieht und die Unterhaltung nicht im Fokus der Anwendung steht, sondern die Wertschöpfung von Wissen und Realwelterfahrungen.

2.1.2. Bekannte Spiel-Design-Elemente

Im folgenden Kapitel werden die wichtigsten interaktiven Spiel-Design-Elemente, welche in Tabelle 2.1 genannt werden, vorgestellt und beschrieben.

Der Einsatz von Gamification-Elementen basiert auf der Annahme, dass diese die grundsätzlichen Bedürfnisse der Menschen ansprechen (vgl. [37, Kapitel 1.1 Enterprise Gamification - Vorgehen und Anwendung, Seite 5]).

Spiel-Mechanik	Spiel-Dynamik	Motiv
Dokumentation von Verhaltensweisen	Exploration	Wissbegierde
Punktesysteme, Badges, Trophäen	Sammeln	Leistung
Ranglisten	Wettbewerb	Soziale Anerkennung
Ränge, Levels, Reputationspunkte	Statuswerbung	
Gruppenaufgaben	Zusammenarbeit	Sozialer Austausch
Zeitdruck, Aufgaben, Missionen	Herausforderung	Kognitive Stimulation
Avatare, virtuelle Welten, virtueller Handel	Entwickeln/Organisieren	Selbstbestimmung

Tabelle 2.1.: Übersicht der Spiel-Design-Elemente [7, vgl.]

Die folgenden aufgeführten Informationen wurden größtenteils aus den Werken [37, Kapitel 2.2.2 Analyse einzelner Spiel-Design-Elemente] und [44, Kapitel 4 The Gamification Toolkit Game Elements] entnommen, bei Abweichungen werden die konkreten Literaturwerke aus Gründen des Leseflusses explizit angegeben.

Dokumentation von Verhaltensweisen

Durch die ständige Dokumentation der eigenen Verhaltensweisen, werden persönliche Fortschritte für den Anwender sichtbar gemacht. Diese Fortschritte können ein Gefühl von hoher Leistungsfähigkeit auslösen, da auf diese Weise ein Vergleich zu den bisherigen erreichten Leistungen ermöglicht wird und somit bei Verbesserung der eigenen Leistung der persönliche Fortschritt verdeutlicht werden kann.

Punkte

Punktesysteme sind in ihrer Einfachheit eine weitverbreitete Form der Spiel-Design-Elemente, welche hauptsächlich eine Feedback-Funktion erfüllen. Diese Feedback-Funktion kann dem Spieler durch diverse weitere Möglichkeiten, wie zum Beispiel durch das Aufsteigen eines Levels, unterschiedlich stark verdeutlicht werden, weshalb Punkte häufig als Grundbestandteil von Gamification-Anwendungen auftreten. Weiterhin symbolisieren Punkte dem Spieler seinen derzeitigen Spielstand und bieten bei der Kombination mit Bestenlisten eine Vergleichsmöglichkeit zu anderen Spielern. In dieser Kombination können Punktesysteme also je nach Zielstellung auch Gewinner einer Herausforderung identifizieren.

In den zwei, in [37] behandelten, empirischen Untersuchungen [27] und [28], wurde nachgewiesen, dass der Einsatz von Punktesystemen durchaus einen leistungsfördernden Effekt auf die Spieler haben kann.

Abzeichen

Die Abzeichen, welche sich in der Regel durch erreichbare Achievements, Badges und/oder Trophäen innerhalb von Gamification-Anwendungen etabliert haben, dienen einerseits einer zielsetzenden und darüber Hinaus auch einer nicht-kontrollierenden Feedback-Funktion.

Zu erwähnen gilt, dass Badges und Trophäen nicht mit Achievements gleichzusetzen sind, sondern lediglich die visuelle Repräsentation eines Achievements darstellen.

Durch den Einsatz solcher Spiel-Design-Elemente soll der Benutzer sinnvoll gefordert werden und sich im Optimalfall durch das somit klar gesteckte Ziel motiviert fühlen.

Weiterhin lassen sich bei dem Einsatz von Achievements unterschiedliche strategische Ansätze identifizieren, die sich von der bereits genannten zielsetzenden Strategie abheben.

Beispielsweise kann bewusst auf die Angabe des Ziels eines Achievements verzichtet werden, um je nach Kontext die Neugierde des Benutzers zu fördern und den Effekt des Erreichens umso größer ausfallen zu lassen, da diese Achievements durch Ausführen verschiedener Tätigkeiten überraschend erreicht werden. Dies spricht aufgrund des erforschenden Charakters vor allem die Benutzerklasse **Explorer** an (siehe Kapitel 2.1.3).

Auf diese Weise kann auch eine stärkere Bindung einer Community entstehen, da der Austausch zwischen Benutzern, die bereits wissen wie ein bestimmtes Achievement erreicht wird und denjenigen, die es noch nicht erreicht haben, die Kommunikation untereinander fördern kann.

Ranglisten

Die sogenannten Rang- oder auch Bestenlisten dienen der Identifikation und Abgrenzung von unterschiedlichen Benutzern, die im Kontext der Gamification-Anwendung unterschiedliche Erfolge, häufig in Form von Punkten, erreicht haben.

Zentrale Absicht solcher Ranglisten ist das Darstellen der besten Spieler, was je nach Kontext für einen hohen kompetitiven Anwendungsanteil spricht.

Im Rahmen von Ranglisten-Spiel-Elementen können weiterhin verschiedene Kategorien identifiziert werden, welche sich auf unterschiedliche Weise durch verschiedene Vor- und Nachteile des jeweiligen Zielkontextes äußern.

Während individuelle Ranglisten die Leistung des Einzelnen aufzeigen und somit versuchen das Individuum von den restlichen Spielern abzugrenzen, ermöglichen team-basierte Ranglisten das kooperative Vernetzen unterschiedlicher Personen und Identifizieren ganzer Team-Leistungen.

Ränge, Level und Reputationspunkte

Wie in [36, Kapitel 4.5 Reputations, Ranks and Levels, Seite 75] beschrieben wird, können Reputationsmöglichkeiten für Benutzer ein wichtiges Identifikationsmittel sein. Durch die Möglichkeit bestimmte Ränge oder Level zu erreichen, sind Spieler besonders im Rahmen von Multiplayer-Anwendungen dazu angehalten, ihr Ansehen gegenüber anderen Spielern zu steigern oder zu festigen und somit den Status eines Veteranen zu repräsentieren.

Diese Art von Spiel-Elementen ermöglicht das Ausstrahlen von Kompetenzen, Talenten sowie Erfahrungen und somit den Platz, den der jeweilige Spieler in der Spielwelt einnimmt.

Avatare, virtuelle Welten, virtueller Handel

Durch die Möglichkeit der Erstellung eines persönlichen Avatars werden dem Spieler je nach Ausprägung hoch oder gering ausfallende Individualisierungs- und somit auch Identifikationsmöglichkeiten angeboten.

Ein wichtiger Aspekt, welcher je nach Anwendungsfeld unterschiedlich ausgelegt wird, ist das Interagieren mit der virtuellen Welt, in der sich der Avatar befindet. Hierbei kann es zur Interaktion mit anderen Spieler-Avataren oder mit definierten Teilen der virtuellen Welt kommen. Die virtuelle Welt kann hierbei beispielsweise Möglichkeiten bieten, sich frei in einer dreidimensionalen Umgebung zu bewegen.

Die virtuelle Welt muss nicht zwangsläufig dreidimensional ausfallen, sondern kann durch geschickte Konzipierung des Interface-Designs auch mit weniger Umgebungsreichweite den gewünschten Effekt erzielen.

Zeitdruck, Aufgaben, Missionen

In [30, Kapitel 6.1.1 Tasks] wird beschrieben, dass Aufgaben und Missionen im Kontext von Gamification-Anwendungen sehr breit vertretene Varianten von Spiel-Design-Elementen sind. Sie können sehr gut mit anderen Elementen wie Level-Systemen kombiniert werden und lassen sich durch die hohe Individualisierbarkeit der Aufgabeninhalte im Rahmen des jeweiligen Anwendungskontexts als ziel- oder wegweisende Spiel-Elemente benutzen, die gleichzeitig in direkter Form auf das tatsächliche Anwendungsfeld Bezug nehmen.

Gruppenaufgaben

Wie in [30, Kapitel 6.1.1 Tasks] weiterhin beschrieben wird, stehen die Gruppenaufgaben, wie auch die Einzelaufgaben, für eine zu erfüllende Gesamtaufgabe. Eine Gruppenaufgabe hebt sich jedoch insofern ab, dass sie aus mehreren kleinen Aufgaben besteht, welche im Team gelöst werden sollen. Wie auch in den Team-Bestenlisten rückt bei Einsatz dieser Elemente die kooperative Ausführung von Aufgaben in den Vordergrund und soll so durch eine starke Teamidentifikation für eine erhöhte Motivation des Einzelnen sorgen.

2.1.3. Nutzergruppen nach Bartle

Nach Bartle [6] werden vier verschiedene Nutzertypen voneinander unterschieden, welche in vielen Literaturwerken zur Gamification als Grundlage oder zumindest als Orientierung referenziert werden.

Die vier verschiedenen Nutzerklassen werden im folgenden Verlauf vorgestellt und hinsichtlich ihrer Bedürfnisse erläutert:

- **Der Killer**

Die Killer legen den Fokus auf den Wettkampf. Somit beruht ihre Motivationsquelle auf stark kompetitiv ausgeprägte Ansätze. Sie fühlen sich dadurch motiviert, den Kontakt mit anderen Individuen im Rahmen eines Wettkampfes zu suchen und sich durch den eigenen Sieg gegenüber anderen Personen zu behaupten.

- **Der Socialiser**

Die Gruppe der Socialiser sind auf der Suche nach sozialer Interaktion. Ohne diese können sie der Spielwelt wenig bis keine Motivation abgewinnen. Die Spielwelt, in der sie sich bewegen, ist vor allem nur das Mittel zum Zweck der Kommunikation. Der größte Anreiz der Socialiser ist die Freundschaft und der Kontakt zu anderen Personen.

- **Der Achiever**

Die Achiever konzentrieren sich auf das Erbringen von Leistungen und das Abschließen von Herausforderungen. Sie werden so eingeschätzt, dass sie die möglichst optimale geforderte Leistung erbringen wollen und daher ihre Motivation aus dem schrittweisen Sammeln von Erfolgen und dem Erfüllen von Aufgaben ziehen.

- **Der Explorer**

Die Explorer legen den Hauptfokus auf die Welt, in der sie sich bewegen und möchten im Zusammenspiel mit dieser Spielwelt durch immer wieder neu gestellte Aufgaben und erlebte Überraschungen alles entdecken, was es zu entdecken gibt. Sie sind sehr ambitioniert und empfinden Stolz für ihr Wissen und ihre Erfahrung in der Spielwelt.

Die Explorer haben vor allem Interesse an der Spielwelt und weniger an den Personen, die sie im Laufe der Erkundung treffen.

Wie aus den Beschreibungen zu erkennen ist, entsteht bei der allgemeinen Betrachtung aller Benutzerklassen ein Spannungsfeld der Interessen, welches in Abbildung 2.1 verdeutlicht werden soll.

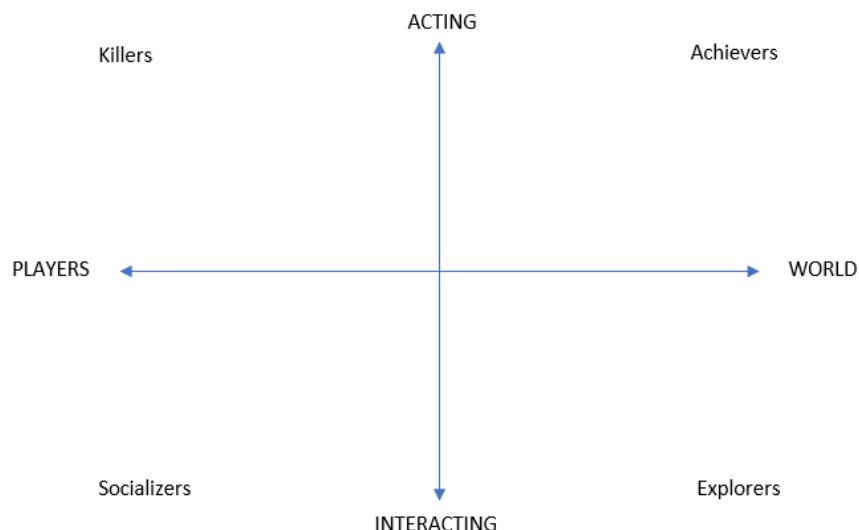


Abbildung 2.1.: Interest Graph nach Bartle [6]

Das Spannungsfeld wird weiterhin definiert durch die vier verschiedenen Interessenausprägungen der Spieler (Players), der virtuellen Welt (World), des Handelns in dieser Welt (Acting) und des Interagierens mit der virtuellen Welt (Interacting).

2.2. Vorstellung des Frameworks Flutter

Flutter ist ein von Google entwickeltes Open-Source-Framework, das auf die Entwicklung mobiler 2D-Applikationen für Android- und iOS-Betriebssysteme ausgelegt ist.

Beworben wird Flutter seit der Ankündigung der Beta-Version, welche in einem offiziellen Blogeintrag der **Google Developers**[3] im Rahmen des **Mobile World Congress 2018**[29] veröffentlicht wurde, unter anderem durch eine besonders flexibel handhabbare und mit Leichtigkeit umzusetzende Gestaltung von hochqualitativen Benutzeroberflächen. Weiterhin wird betont, dass die zu erzielenden Ergebnisse das Produkt einer extrem schnell umsetzbaren Entwicklung “in Rekordzeit“ sind.

Im folgenden Kapitel werden die Kernprinzipien von Flutter erläutert und die framework-spezifischen Besonderheiten beschrieben um Ihnen als Leser einen Eindruck von dem Framework zu ermöglichen und die grundsätzliche Funktionsweise sowie dessen Besonderheiten, hinter den werbe-zweckmäßigen Worten der öffentlichen Präsentation, nachvollziehbar zu gestalten.

2.2.1. Flutter Kerntechnologien

Das Flutter Framework besteht aus verschiedenen Basiskomponenten, welche aus aufeinander aufbauenden Systemschichten bestehen (siehe Abbildung 2.2) und sich auf oberster Abstraktionsebene auf die Flutter-Engine und das Flutter-Framework unterteilen lassen.

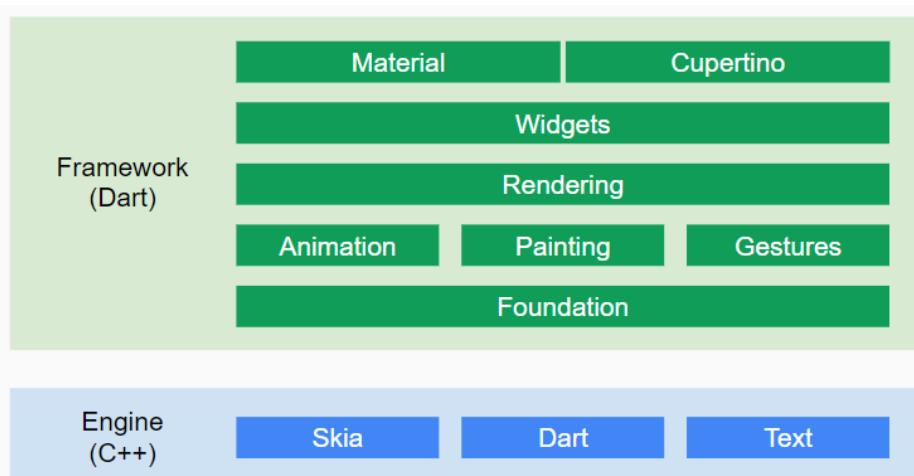


Abbildung 2.2.: Flutter High-Level Systemarchitektur [17]

Im folgenden Abschnitt werden die drei Basistechnologien von Flutter kurz beschrieben:

Flutter Engine

Wie in dem Flutter Wiki [43] beschrieben wird, setzt sich die C/C++ basierte Flutter Engine aus den drei Kerntechnologien Skia[38], eine open source 2D Graphics Library, die Dart Virtual Machine sowie die verschiedenen verwendeten Bibliotheken zum Rendern von Texten zusammen.

Foundation Library

Die Dart-basierte Foundation Library[18] stellt Basisklassen und -funktionen zur Verfügung, die von den anderen in der Flutter-Architektur überliegenden Schichten genutzt werden (siehe Abbildung 2.2) und stellt somit eine zentralen Grundlage bei der Konstruktion von Applikationen mittels dem Flutter-Framework dar.

Design-specific Widgets

Das Flutter-Framework stellt zwei verschiedene Design-Stile der Widgets zur Verfügung, welche den jeweiligen Design-Sprachen von **Google Material Design**[25], welche 2014 entwickelt wurde sowie die **iOS Design**[23] kopierende Design-Sprache **Cupertino**[11] entsprechen.

2.2.2. Plattformübergreifende Entwicklung von mobilen Applikationen

Flutter ermöglicht das plattform-unabhängige Entwickeln von mobilen Applikationen für Android- und iOS-Betriebssysteme und verarbeitet oberflächen-spezifische Darstellungsformen der jeweiligen Zielplattformen framework-intern. Dies bedeutet, dass zum Zeitpunkt der Entwicklung auf extrem abstrakter Ebene ein Oberflächendesign unabhängig von der Zielplattform konzipiert werden kann, da die plattform-spezifischen Darstellungsweisen von dem Framework selbst gelöst werden und von den jeweiligen Entwicklern nicht berücksichtigt werden müssen.

Hinweis: Die in Abbildung 2.3 gezeigten Oberflächen entstammen demselben Programmcode.

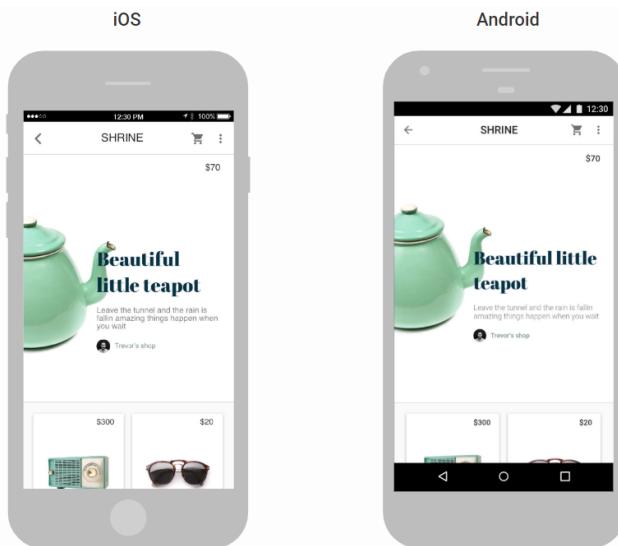


Abbildung 2.3.: Darstellung (iOS und Android): Plattformunabhängige Entwicklung von Applikationen [41, What is Flutter?]

2.2.3. Hot-Reload Funktion

Die sogenannte “Hot Reload“-Funktion spielt im Rahmen der App-Entwicklung mit Flutter eine primäre Rolle, da bei Änderungen an der Oberfläche während der Laufzeit ermöglicht wird, innerhalb von “Millisekunden“ die Oberfläche neu zu generieren und das System dabei den internen Zustand beibehält. Diese Funktion soll eine schnelle und experimentelle Vorgehensweise ermöglichen.[8]

2.2.4. Flutter Widgets

Alle User-Interface Grundbausteine in Flutter werden als sogenannte Widgets bezeichnet und dementsprechend gleich behandelt([17, vgl. Everything’s a Widget]). Diese Grundbausteine stellen die jeweiligen Design- und Interaktions-Elemente der Applikation dar und umfassen verschiedene spezialisierte Ausprägungen wie in Abbildung 2.4 beispielhaft zu sehen ist.

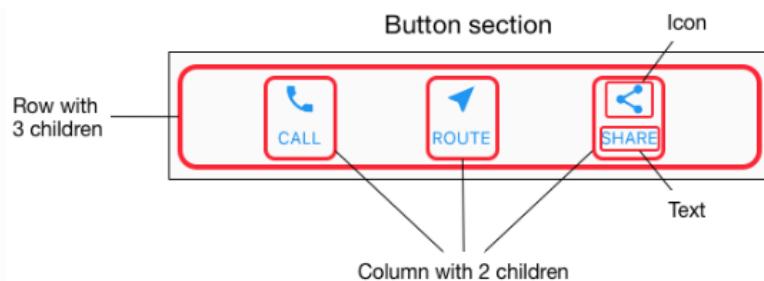


Abbildung 2.4.: Beispiel: Aufbau von Widgets [9]

Ein in dem Artikel [45, Widgets] beschriebener wichtiger Aspekt ist, dass Flutter im Gegensatz zu anderen Alternativen eine eigens erstellte Systemschnittstellen-Architektur besitzt und nicht auf OEM-Widgets oder DOM-Web Views zurückgreift, wie in Abbildung 2.5 verdeutlicht wird. Dies ermöglicht eine extrem freie Handhabung und Individualisierung im Umgang mit den Widgets und Plattform spezifischen Diensten.

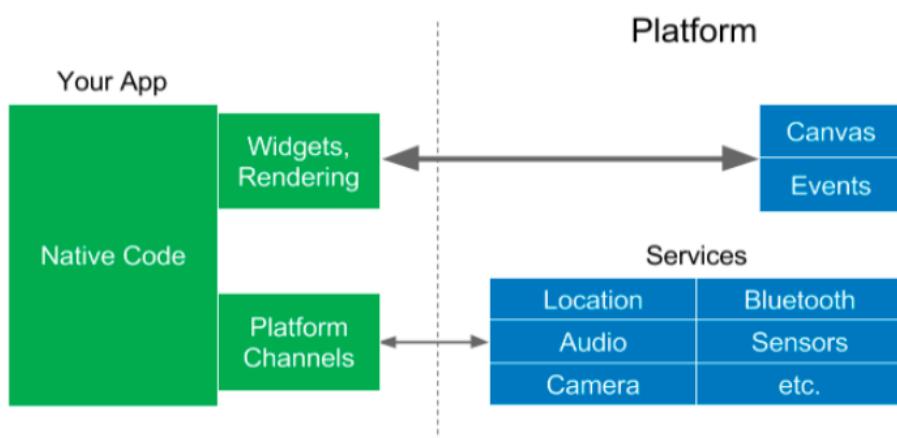


Abbildung 2.5.: Flutter Widgets Architektur [45]

Diese Widgets bilden zur Laufzeit eine hierarchische Struktur und setzen sich selbst aus weiteren Widgets zusammen. Somit ist der Ursprung einer jeden Programm-Oberfläche ein Wurzel-Widget, welches wiederum Kinder-Widgets enthält und diese somit über mehrere Ebenen eine Gesamtkomposition bilden, den sogenannten **Widget Tree**.

Widgets können also je nach Ausprägung beispielsweise strukturierende Elemente, wie Spalten und Zeilen sowie stil-definierende Elemente in Form von Farbdefinitionen darstellen. Weiterhin können Widgets auch Interaktion-Elemente, wie Buttons und Textfelder und noch viele weitere Elemente mit unterschiedlichen Eigenschaften definieren.

2.2.5. Flutter UI Rendering

Flutter verwendet im Gegensatz zu anderen Alternativen nach eigener Aussage eine selbst entwickelte und optimierte Rendering-Engine, die die Benutzeroberfläche nach einer Veränderung des aktuellen Status neu generiert und diese Veränderung dem Benutzer darstellt (vgl. [42, What makes Flutter unique?]). Die folgende Abbildung 2.6 stellt einen solchen Ablauf dar.

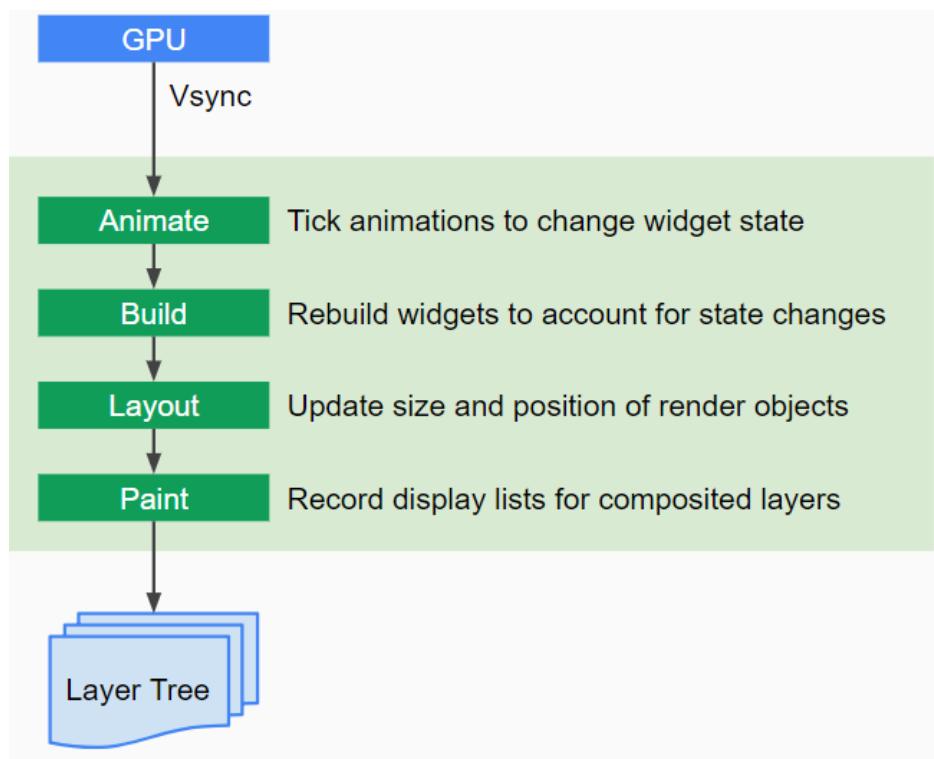


Abbildung 2.6.: Flutter Rendering [17]

Sollte sich der aktuelle Zustand der Applikation beispielsweise durch eine Benutzeraktion verändern, so werden die in der Baumstruktur vorhandenen Widgets neu generiert, deren Eigenschaften aktualisiert und dem Benutzer letztendlich angezeigt.

2.3. Was ist Redux?

Redux ist eine von Dan Abramovic und Andrew Clark entwickelte, im Jahr 2015 veröffentlichte open-source Bibliothek für JavaScript-Anwendungen (vgl. [35]). Durch die zentrale Verwaltung von Zustandsinformationen soll eine erleichterte Pflege der Konsistenz und aufgrund dessen auch eine “einfache“ Wartbarkeit und Testbarkeit der Software ermöglicht werden. [32, vgl. Abschnitt Read Me]

*Hinweis: Wie auch in [32, vgl. Abschnitt Read Me] angemerkt wird, wird an dieser Stelle darauf hingewiesen, dass es sich hierbei nicht um das WordPress-Framework **Redux Framework**[33] handelt.*

Innerhalb einer Redux-Architektur existieren verschiedene Komponenten, welche jeweils einen besonderen Zweck erfüllen und durch das modulare Zusammenspiel mit anderen Komponenten in dem System eine Gesamtkomposition bilden.

Die wichtigsten Redux-spezifischen Komponenten sind **Actions**, **Reducers** und der **Store**, welche hinsichtlich Bedeutung und Funktionsweise in den nachfolgenden Kapiteln beschrieben werden:

2.3.1. Actions

Wie in den Basics der Online-Dokumentation [1] beschrieben wird, definieren Actions im Rahmen von Redux hauptsächlich Informationspaket, welche unter dieser Bedeutung auch als “Payload“ bezeichnet werden. Sie enthalten in erster Linie nur Informationen über die Action-spezifische Operation, wie beispielsweise das Ändern bestimmter Bestandteile des aktuellen Systemzustands.

Hinweis: Es könnte beispielsweise denkbar sein, dass ein neuer Eintrag einer ToDo-Liste angelegt werden soll. Hierbei würde die Action vermutlich einen Namen wie “AddTodoAction“ tragen und die Informationen des Titels des neuen Eintrags enthalten.

Das Ausführen dieser Zustandsänderung wird also nicht von den Actions selbst ausgeführt, für diesen Zweck dienen die sogenannten Reducer-Funktionen.

2.3.2. Reducers

Reducer-Funktionen erfüllen den Zweck, die durch die jeweilige Action übermittelten Informationen zu verarbeiten und die eigentliche, durch die Action spezifizierte, Zustandsänderung durchzusetzen. (vgl. [34])

Reducer liegen im Allgemeinen als Pure Function (auch Reine Funktionen) vor. Dies bedeutet im Rahmen der Redux-Architektur, dass die Funktionen keine direkten Änderungen an den durch die Actions übermittelten Informationen oder an dem Status selbst vornehmen dürfen und unter keinen Umständen Seiteneffekte produzieren dürfen. Die Auswirkungen von Pure Functions müssen deterministisch bestimmt sein, also muss unter den gleichen Eingabebedingungen immer das gleiche zu erwartende Ergebnis folgen.

Im Detail werden also nicht direkte Änderungen an Objekten vorgenommen, sondern der gesamte aktuelle Zustand durch das Neuerzeugen überschrieben.

Hinweis: Eine nicht den Eigenschaften von Pure Functions entsprechenden Funktion wäre das Bestimmen des Datums mit der JavaScript-Funktion Date.now(), da hier bei Ausführung immer ein unterschiedliches Ergebnis herauskommt.

2.3.3. Store

Der Store ist in der Redux-Architektur das zentrale Element, welches nicht nur für die Datenhaltung des aktuellen Zustands (State) der Applikation zuständig ist, sondern auch als Schnittstelle für die Bereitstellung und Veränderung des aktuellen Systemzustands dient (vgl. [39]).

Im Gegensatz zu den anderen Komponenten ist der Store innerhalb der Architektur einmalig, was viele Vorteile der Redux-Architektur überhaupt erst ermöglicht.

Hinweis: Der Store enthält alle für die Applikation relevanten Informationen an zentraler Stelle im System in einem State Objekt.

Die gespeicherten Informationen liegen im besten Fall in normalisierter Form vor. Insgesamt ist ein flacher Key-Value Store dem Handhaben komplexer Objekte vorzuziehen

2.3.4. Flutter Redux Library

Die externe Flutter Libary **flutter_redux**[16] setzt die auf JavaScript basierende Redux-Funktionsweise für die Entwicklung mobiler Applikation mit Flutter und Dart um, dadurch wird die beschriebene komfortable Nutzung der Redux-Funktionsweise ermöglicht.

Grundsätzlich werden also dieselben Funktionsweisen durch die Nutzung der Flutter Redux Library abgebildet. Es folgt dennoch eine kurze Beschreibung der spezifischen Redux Widgets, welche sich jedoch alle auf die Interaktion mit dem zentralen Store beziehen:

- **StoreProvider**

Der StoreProvider ist ein Schnittstellen-Widget, das allen unterliegenden Widgets Zugang zum Store ermöglicht.

- **StoreBuilder**

Der StoreBuilder holt sich den aktuellen Zustand von dem in der überliegenden Hierarchie am nächsten platzierten StoreProvider und bietet eine builder-Funktion um die fallspezifischen Informationen zu verwenden.

- **StoreConnector**

Der StoreConnector trägt die Aufgabe, den von dem StoreProvider übergebenen Store durch eine converter-Funktion in ein ViewModel umzuwandeln, welches die benötigten fallspezifischen Informationen in sich hält und durch die builder-Funktion diese Informationen kontextspezifisch verwendet. Weiterhin wird der StoreConnector bei jeder Änderung des Zustands neu generiert und somit die durch das ViewModel repräsentierten Informationen aktualisiert.

3. Untersuchung des Problembereiches

Im folgenden Kapitel werden die Rahmenbedingungen und die Analysestrategie zu den durchgeführten Befragungen von Professoren und Studierenden aus dem Studiengang Informatik/Softwaretechnik der Fachhochschule Lübeck dargestellt. Die daraus resultierenden Ergebnisse und Erkenntnisse beider Seiten werden in zusammengefasster Form beschrieben. Die Diskussion dieser Ergebnisse und die, aus den durchgeführten Befragungen abgeleiteten Anforderungen an die zu entwickelnde Software, stellt das Produkt der Anforderungsanalyse und somit den Ausgangspunkt für die gesamte Arbeit dar.

3.1. Hypothese

Die Auseinandersetzung mit komplexen Problemstellungen, wie einer, als abschließende Prüfungsleistung des Studiums zu erarbeitende, Bachelorarbeit, stellt erfahrungsgemäß für viele Studierende eine große Herausforderung dar. Diese Herausforderung ergibt sich aus dem erstmaligen Zusammenspiel von selbstständigem und eigenverantwortlichem Arbeiten sowie dem Problemlösen mittels erworbener Fach- und Methodenkenntnisse über einen längeren Zeitraum. Trotz des Verlaufs des Studiums, des angeeigneten Wissens und der somit zahlreich erworbenen Fähigkeiten kommt es im Kontext von Bachelorarbeiten dennoch oftmals zu Schwierigkeiten, diese Kompetenzen auf reale Probleme abzubilden und zu dokumentieren.

3.2. Identifikation der Interessengruppen

Die folgende Tabelle 3.1 bietet eine Übersicht der, im Rahmen des Projektes identifizierten, Interessengruppen und welche Stellung diese zu dem Projekt bezüglich Einfluss, Einstellung und Erwartungen einnehmen.

Bezeichnung	Einfluss	Einstellung	Erwartungen	Bemerkungen
Studierende	Hoch	Positiv	Erhoffen sich optimale Ergebnisse und weniger Fallstricke bei dem Bearbeiten der Bachelorarbeit und diesbezüglich eine insgesamt umfangreich ausfallende Hilfestellung.	Zielgruppe der zu entwickelnden Applikation.
AStA	Hoch	Positiv	Erhoffen sich eine angemessene Förderung und Entlastung der Studierenden bei der Bearbeitung der Bachelorarbeit.	Sprachrohr und Interessenvertreter der Studierenden.
Leiter des Bachelorseminars	Hoch	Positiv	Erhofft sich eine steigende Bereitschaft der Studierenden, im Rahmen des Bachelorseminars Beiträge zu erbringen sowie eine steigende Qualität der Vorbereitung, Bearbeitung und Fertigstellung von Bachelorarbeiten.	Leitet das Bachelorseminar, hat somit direkten Kontakt mit der Zielgruppe und trägt wichtige Erfahrungswerte bezüglich der Probleme der Studierenden mit sich.
Professoren	Hoch	Positiv	Erhoffen sich eine steigende Qualität der Bearbeitung der Bachelorarbeit sowie eine Entlastung bei dem Betreuen von Studierenden, hinsichtlich sich wiederholenden Erklärungen und weiterer Trivialitäten.	Tragen wichtige Erfahrungswerte durch den detaillierten Einblick als Betreuer von Bacheloranden sowie dem Bewerten von Bachelorarbeiten mit sich.
Präsidium	Mittel	Positiv	Erhoffen sich eine Steigerung der Leistung der Studierenden an der Fachhochschule Lübeck.	Macht promotor der Fachhochschule Lübeck.

Tabelle 3.1.: Übersicht der Interessengruppen

3.3. Wahl der Analysestrategie

Um den Problembereich zu ermitteln und somit eine Analysegrundlage zu schaffen, werden unterschiedliche Personengruppen der Fachhochschule Lübeck durch verschiedene Befragungs- und Analysemethoden in das Projekt miteinbezogen. Dies soll einen detaillierten Einblick in die Sichtweisen der unterschiedlichen beteiligten Personen und Interessengruppen ermöglichen und somit eine Grundlage für das Verständnis der aktuellen Situation bilden.

Als primäre Einflussgeber, die in die Entwicklung der Applikation stark eingebunden werden, wurde die Gruppe der Professoren und die Gruppe der Studierenden identifiziert. Diese Entscheidung wurde aufgrund der im Rahmen der Bachelorarbeit existierenden unterschiedlichen Sichtweisen sowie der unmittelbaren Berührung mit dem Problembereich der Beteiligten und den unterschiedlichen Erfahrungsständen von Betreuern und Bacheloranden getätigt (siehe Abbildung 3.1). Darüber hinaus handelt es sich bei den Studierenden um die Zielgruppe und potentielle spätere Anwender, weshalb die Aufnahme der Probleme und Meinungen der Studierenden als essenziell angesehen wird.

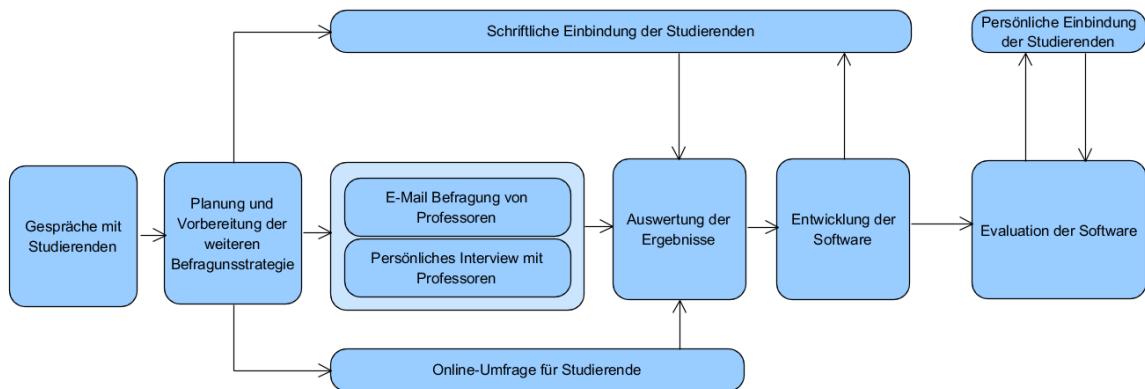


Abbildung 3.1.: Beschreibung der Analysestrategie

Durch die initial durchgeführten Gespräche mit den Studierenden ließen sich Erkenntnisse über den Umfang des Problems gewinnen. Auf Grundlage dieser Erkenntnisse wurde die weitere Analysestrategie verfeinert und für die Umsetzung durch die Erstellung von Fragenkatalogen und Interview-Leitfaden vorbereitet (siehe Abbildung 3.1), welche im folgenden Verlauf beschrieben werden.

Die Interviews mit den Studierenden sollen somit als Grundlage zu Konzeption und Design weiterer Befragungsstrategien dienen und stellen in diesem Umfang einen Ausgangspunkt für die Problemanalyse dar. Weiterhin wurden die gewählten Interviewpartner bei Zustimmung über das gesamte Projekt und nach Absprache auch in die Evaluation der Applikation miteinbezogen.

Durch das Durchführen von Gruppeninterviews sollen die Studierenden zu Diskussionen angeregt werden, welche durch den Leiter des Interviews gezielt angeregt werden können. Dies hat den Zweck, die Situation und die Rolle der Studierenden für den Interviewer erkenntlich zu machen. Diese persönlichen Interviews dienen in erster Linie also nicht der Erhebung konkreter Daten, sondern als erster Berührungspunkt mit dem Problembereich und in diesem Rahmen als strategischer Orientierungspunkt für das weitere Vorgehen.

3.3.1. Einbezug der Professoren

Für die Interessengruppe der Professoren aus dem Fachbereich Informatik wurden als Grundlage der Datenerhebung Einzelinterviews mit einer ausgewählten Gruppe von Professoren durchgeführt, während eine weitere Gruppe von Professoren schriftlich per E-Mail befragt wurde (siehe Abbildung 3.1). Dies bietet sowohl den Zugriff auf die unmittelbaren Erfahrungen der einzelnen Professoren als Spezialisten in ihren jeweiligen Fachgebieten, als auch auf die Erfahrungen der Professoren in der Position eines Betreuers und Ansprechpartners für Bacheloranden.

Durch das Durchführen von Einzelinterviews wird ermöglicht, die Erwartungen seitens der Professoren an die Bacheloranden im Detail zu identifizieren und die in dieser Hinsicht priorisierten inhaltlichen und methodischen Aspekte bei der Bearbeitung einer Bachelorarbeit herauszuarbeiten. Die Aufteilung auf Einzelinterviews und E-Mail Befragungen bietet den Vorteil, beide Strategien simultan zu verfolgen und nach Abschluss der Datenerhebung sowohl die detaillierten Einzelinterviews, als auch die oberflächlicher ausfallenden E-Mail Antworten in bereits dokumentierter Form vorliegen zu haben, um diese dann auszuwerten.

3.3.2. Einbezug der Studierenden

Nach Absprache mit den interviewten Studierenden werden diese weiterhin per schriftlicher Befragung in die Entwicklung des Projektes, unter ergänzender Zunahme anderer Studierender, eingebunden. In diesem Zusammenhang werden die teilnehmenden Studierenden mittels Bildern, Fragen und Gestaltungsideen der Applikation regelmäßig über den aktuellen Stand der Entwicklung informiert. Auf diese Weise sollte es möglich sein, ein breites, jedoch persönliches Feedback zu erhalten, da sich die Einbindung der Studierenden auf diese Weise sehr gut automatisieren lässt. Wesentlich dabei ist auch die Tatsache, dass die Reaktionsfreudigkeit der Studierenden höher ausfällt, als für die im Vergleich existierende Alternative der zeitaufwändigeren persönlichen Interviews.

Aufgrund der Durchführung der schriftlichen Befragung wird der Ablauf bereits dokumentiert und der Fragesteller sowie die Studierenden haben jederzeit die Möglichkeit Fragen und Anregungen zu teilen. Miteinbezogen werden vorzugsweise Studierende der oberen Semester, unabhängig davon, ob sie sich vor Beginn, während der Bearbeitung oder nach Abschluss der Bachelorarbeit befinden, da die unterschiedlichen Sichtweisen und Erfahrungsstände wichtige Impulse für die Entwicklung der Applikation geben.

3.4. Ergebnisse der Professoren

Im Folgenden sind die gewonnenen Eindrücke und Kenntnisse der Einzelinterviews mit den Professoren des Fachbereichs Informatik durch Themenkategorien geordnet und in zusammengefasster Form dokumentiert.

Es haben insgesamt sechs Professoren an persönlichen Einzelinterviews teilgenommen, wobei es für fünf Interviews gestattet wurde, eine Tonaufzeichnung anzufertigen.

Des Weiteren wird auch eine schriftliche Befragung miteinbezogen, die jedoch das gleiche Fragespektrum wie die Interviews einnimmt.

Allgemeine Informationen

Der zeitliche Rahmen der fünf aufgezeichneten Interviews erstreckt sich über einen Zeitraum von etwa 30 bis 45 Minuten. Für die Auswertung der aufgezeichneten Interviews wurden die Aussagen der Interviewpartner, auf Grundlage der vorliegenden Audioaufnahmen, unter Berücksichtigung des Kontextes aufbereitet und werden nachfolgend dargestellt. Ein weiteres Einzelinterview, dass nicht aufgezeichnet wurde, erstreckte sich über einen Zeitraum von 75 Minuten. Für dieses Interview wurden lediglich begleitende Feldnotizen angefertigt. Diese Feldnotizen wurden im Anschluss des Interviews aufbereitet und fließen zusammen mit den Ergebnissen der schriftlichen Befragung in die folgende Beschreibung ein.

Die gewählten Kategorien ergeben sich aus dem gewählten Auswertungsverfahren, was sich an der qualitativen Inhaltsanalyse nach Mayring[26] orientiert und basieren in diesem Kontext auf die, gemäß des Verfahrens, herausgearbeiteten Codings.

Die Interviews mit den einzelnen Professoren sind in transkribierter Form im Anhang zu finden (siehe Anhang A.1.1).

3.4.1. Art der Arbeit

Im Laufe der Interviews wurden verschiedene Typen von Bachelorarbeiten versucht zu identifizieren. Dabei geht es vor allem darum, die Vielfalt der typischen Arbeiten des Studiengangs Informatik/Softwareentwicklung zu erfassen und somit einen Überblick über die Situation zu bekommen.

Als allgemeine auftretende Arten von Bachelorarbeiten wurden die Klassen *Entwickelnde Arbeit* und *Evaluierende Arbeit* identifiziert. Weiterhin gibt es auch *Reine Literaturarbeiten*, welche in dem Studiengang Informatik/Softwareentwicklung jedoch nicht oder nur in einem sehr geringen Vorkommen auftreten.

Es folgt eine stichpunktartige Aufführung der gewonnenen Erkenntnisse:

Konstruktiv/Entwickelnd - Durchlauf des Softwareentwicklungszyklus

Die Art der konstruktiven/entwickelnden Arbeiten, befassen sich hauptsächlich mit dem Durchlaufen des Softwareentwicklungszyklus. Im Rahmen der Bachelorarbeit werden die verschiedenen Aufgaben der Softwareentwicklung, je nach Schwerpunkt der Arbeit, mehr oder weniger intensiv von dem Bacheloranden bearbeitet und beschrieben.

Typischer fachlicher Inhalt einer solchen Arbeit könnte sein:

- 1.** Anforderungsanalyse
- 2.** Entwurf einer Softwarearchitektur
- 3.** Implementierung eines Softwareprototyps
- 4.** Evaluation des Softwareprototyps

Typische Aufgabenstellungen:

- Entwicklung einer mobilen Applikation zur Interpretation von Bildmaterial.
- Entwicklung einer mobilen Applikation zur Steigerung der Bereitschaft bei Senioren und Seniorinnen Fitnessaktivitäten auszuführen, unter Einbezug von Gamification-Elementen.
- Entwicklung einer Software zur Optimierung der täglichen Arbeitsabläufe im Unternehmen A.

Ein bei externen Arbeiten wichtiger Aspekt sind die möglichen Interessenunterschiede zwischen dem externen Unternehmen und dem internen Betreuer der Fachhochschule, welche einen Einfluss auf die Inhalte der Bachelorarbeit haben können. Externe Unternehmen sind tendenziell eher an dem resultierenden Ergebnis interessiert, während die internen Betreuer darüber hinaus einen hohen Wert auf nachvollziehbare Methodik, Herangehensweise sowie Planung und das saubere wissenschaftliche Arbeiten legen und somit ein hohes Interesse an dem Gesamtprozess haben.

Analytisch/Evaluierend - Vergleich, Auswertung und/oder Nachweis eines Aufgabengegenstandes

Die Art der analytisch/evaluierenden Arbeiten beinhalten hauptsächlich die Analyse und Auswertung eines Aufgabengegenstandes oder mehrerer verschiedener Aufgabengegenstände. Es wird der gesamte Weg der Analysestrategie, der Durchführung der Analyse und der Auswertung behandelt.

Typischer fachlicher Inhalt einer solchen Arbeit könnte sein:

- 1.** Erstellen eines Kriterienkatalogs
- 2.** Aufbau des Experiments
- 3.** Durchführung des Experiments
- 4.** Evaluation und Ergebnisauswertung

Typische Aufgabenstellungen könnten sein:

- Evaluation der Gesichtserkennungsdienste von Unternehmen A, Unternehmen B und Unternehmen C.
- Untersuchung des Verhaltens einer neuen Technologie A, im Vergleich mit einer alten Technologie B.
- Datenbankanalyse unter Anwendung von Machine-Learning-Algorithmen.

Reine Literaturarbeiten

Reine recherchierende Arbeiten finden in dem Studiengang Informatik/Softwareentwicklung aufgrund dem geringen Interesse seitens der Studierenden kaum statt und werden aus Gründen der Vollständigkeit lediglich erwähnt und nicht ausgeführt.

3.4.2. Erwartungen an den Bacheloranden

Im Laufe der Interviews wurden die Professoren hinsichtlich Ihrer Erwartungen an die Bacheloranden befragt und haben in diesem Rahmen häufig gleiche oder ähnliche Punkte ausgeführt. Aus diesem Grund werden im folgenden Verlauf die Meinungen der befragten Professoren aus der Sicht als Betreuer zu den jeweiligen Aspekten als zusammengefasstes Meinungsbild wiedergegeben.

Es folgt die Ausführung der gewonnenen Erkenntnisse:

Selbstständiges Arbeiten

Das selbstständige Arbeiten und Vorgehen ist eine der am häufigsten genannten Erwartungen, was sich in unterschiedlichen Punkten äußert. Dazu zählen vor allem das selbstständige Kommunizieren von Ergebnissen und das Einholen von Feedback sowie die Transparenz bei Problemen oder Schwierigkeiten, um sich Hilfe von dem Betreuer zu holen. Es wurde mehrfach betont, dass es im Allgemeinen nicht die Aufgabe des Betreuers ist, nachzufragen und aufzufordern. Das Einbinden des Studierenden in die Erarbeitung der Aufgabenstellung ist ein häufig genannter Punkt, der bereits frühzeitig Engagement des Studierenden erfordert. Des Weiteren stellt das selbstständige Einarbeiten in die Probleme und die Auswahl geeigneter Methoden und Werkzeuge einen wesentlichen Inhalt der Arbeit dar.

Die Vorgehensweise

Sehr häufig wird betont, dass die Vorgehensweise hinsichtlich der wissenschaftlichen Arbeitsweise von essenzieller Bedeutung ist. In diesem Rahmen soll der Studierende auch zeigen, dass er in der Lage ist, große Probleme systematisch in Teilprobleme zu zerlegen und diese unter Berücksichtigung der im Studium vermittelten Methoden, Modelle und Techniken zu bearbeiten. Sehr wichtig ist dabei das vorausschauende Planen von Teilprozessen, wie beispielsweise das Evaluieren der Ergebnisse. Dies sollte von Anfang an berücksichtigt werden und die in diesem Rahmen getroffenen Entscheidungen sollten nachvollziehbar erklärt werden können. Dazu zählt weiterhin das Erstellen eines Zeitplans, welcher sich im Verlauf jedoch durchaus verändern kann. Es wird sehr viel Wert darauf gelegt, zu sehen, dass die Studierenden einen weiten Blick auf das gesamte Projekt entwickeln und pflegen.

Der Literaturteil

Es wird betont, dass vor allem Wert auf einschlägige Quellen gelegt wird. In diesem Zusammenhang ist es wichtig, dass die Studierenden Literaturquellen verwenden sollten, die bereits eine anerkannte längere Gültigkeit besitzen. Weiterhin sollten die Studierenden über den Umfang von Grundlagenliteratur hinaus blicken und je nach Themengebiet und Arbeitsstand spezifischere Fachliteratur in die Arbeit einbeziehen. Dies kann auch bedeuten, dass auf wissenschaftliche Papiere und Primärquellen verwiesen werden soll. Je nach Thema und Aufgabenstellung kann der Literaturteil mehr oder weniger umfangreich ausfallen, was sich durch den praktischen Teil der Arbeit ausgleichen lässt.

Der praktische Teil

Im Wesentlichen soll der praktische Teil den Umfang der Aufgabenstellung abdecken und gegebene und/oder erhobene Anforderungen erfüllen. Je nach Thema und Aufgabenstellung der Arbeit nimmt dieser Teil einen höher oder niedriger ausfallenden Umfang ein. Der Studierende soll bei der praktischen Bearbeitung der Aufgabe zeigen, dass er in der Lage ist, im Studium gelerntes Wissen sowie kennengelernte Methoden und deren Ausführung umzusetzen.

Das Ergebnis der Bachelorarbeit

Grundsätzlich soll die Bachelorarbeit aus zwei Teilen von Leistungen bestehen. Die von dem Studenten durchgeführte Literaturarbeit nimmt einen Teil der Arbeit ein, während die eigenständige praktische Leistung den anderen Teil erfüllt. Der Umfang der jeweiligen Anteile kann dabei je nach Themengebiet und Aufgabenstellung stark variieren.

Das angestrebte Ziel der Aufgabenstellung einer Bachelorarbeit, welches sich je nach Art der Arbeit stark voneinander unterscheiden kann, soll wissenschaftlich erarbeitet und somit nachvollziehbar und belegbar sein und im Optimalfall die Aufgabenstellung erfüllen. Es kann jedoch auch vorkommen, dass das angestrebte Ziel aus verschiedenen Gründen nicht erreicht wurde. Dies muss nicht bedeuten, dass es zu einer schlechten Bewertung der Bachelorarbeit kommt, sofern der Grund oder die Erkenntnis über ein bestimmtes aufgetretenes Problem belegbar und nachvollziehbar dokumentiert ist.

3.4.3. Häufig auftretende Probleme

Es stellte sich im Verlauf des Interviews heraus, dass Studierende immer wieder mit gleichen oder ähnlichen Problemen zu kämpfen haben. Im folgenden Verlauf werden diese genannten Probleme in aufbereiteter Form stichpunktartig beschrieben.

Es folgt die Ausführung der gewonnenen Erkenntnisse:

Zeitmanagement

Der am häufigsten genannte Aspekt ist das mangelhafte Zeitmanagement der Studierenden. Im Laufe der Bearbeitung der Bachelorarbeit kommt es häufig zur Unterschätzung des nötigen Zeitaufwandes, besonders hinsichtlich des Schreibens der Dokumentation.

Viele Studierende verschieben das Schreiben der Dokumentation auf einen späteren Zeitpunkt und geraten im späteren Verlauf der Bearbeitungszeit somit unter Zeitdruck. Die Betreuer haben unter diesen Umständen wenige Möglichkeiten rechtzeitiges und hilfreiches Feedback zu liefern. Es wird häufig empfohlen frühzeitig mit dem Schreiben zu beginnen und begleitend zum Arbeitsfortschritt die Dokumentation an mehreren Stellen wachsen zu lassen. Trotz vieler Hinweise seitens der Betreuer kommt es in diesen Belangen häufig zu großen Problemen. Das Problem des mangelnden Zeitmanagements äußert sich auch darin, dass die Studierenden sich bei der Bearbeitung in nicht relevanten Details verlieren, da sie die Schwerpunkte der Arbeit nicht erkennen.

Wissenschaftliches Arbeiten

Die Studierenden erfassen teilweise nicht die Bedeutung des wissenschaftlichen Arbeitens. Es kommt immer wieder zu Schwierigkeiten und Unklarheiten über den eigentlichen Umfang der Arbeit und wodurch sich das wissenschaftliche Arbeiten auszeichnet. Oft wird der falsche Umgang mit Literatur und Quellen als negatives Beispiel genannt.

Ein Kritikpunkt ist, dass es in dem Studiengang Informatik/Softwareentwicklung keinen Kurs gibt, welcher die Studierenden auf das wissenschaftliche Arbeiten vorbereitet. In einigen Wahlpflichtmodulen werden diesbezüglich zwar Ansätze im Rahmen von Projekten integriert, jedoch gilt dies somit nur für die an dem Wahlpflichtmodul teilnehmenden Studierenden und steht auch nicht im Fokus der Projektarbeit.

Kommunikation

Kommunikation und Transparenz ist ein weiteres angesprochenes Problem. Es kommt vor, dass Studierende und Betreuer unterschiedliche Ansichten über die Zusammenarbeit haben, die sich dadurch äußern, dass die Studierenden auf Aufforderungen bezüglich Leistungsständen oder Ergebnissen seitens der Betreuer warten oder sich sogar aus diversen Gründen nicht trauen, ihren aktuellen Arbeitsstand oder ihre Probleme mit dem Betreuer zu teilen.

Die Vorbereitung der Studierenden

Bei der Vorbereitung der Studierenden nennen die beteiligten Interviewpartner unterschiedliche Aspekte, es wird zum einen das mangelhafte selbständige Informieren der Studierenden kritisiert, andererseits wird jedoch auch eine optimaler zu gestaltende Vorbereitung der Studierenden seitens der Fachhochschule für den Studiengang Informatik/Softwareentwicklung betont.

Das angebotene Bachelorarbeit-Seminar wird positiv erwähnt, da es einen positiven Einfluss auf die Arbeit der Studienenden hat. Es müssen weniger Aspekte einer Bachelorarbeit erklärt werden, jedoch müssen viele Aspekte mehrfach wiederholt werden.

Es wird betont, dass auch viele Informationsmaterialien im Lernraum der Fachhochschule Lübeck zu finden seien, auf die auch oft hingewiesen wird, jedoch von den Studierenden nicht in dem Umfang beachtet werden, für den die Materialien gedacht sind. Dabei wird unter anderem auch kritisiert, dass die Informationsmaterialien teilweise schwer auffindbar sind, da sie nicht an einer zentralen Stelle, sondern verteilt im Lernraum liegen.

Weiterhin wird jedoch auch betont, dass die Studierenden zu wenig Engagement aufbringen, sich trotz vieler Möglichkeiten selbstständig zu informieren.

3.4.4. Die Applikation – Wünsche und Anregungen

In den Interviews wurden die Professoren nach ihren persönlichen Wünschen, Ideen und Anmerkungen bezüglich der Applikation gefragt. Es folgt die Ausführung der aufgenommenen Wünsche und Ideen.

Neuer Kanal zu den Studierenden

Es wird der Wunsch geäußert, dass die Applikation verwendet werden kann, um in zentraler Form konkrete interne oder externe Bachelorarbeit-Themen sowie Beispieltiteln angeben zu können, da dies an der Fachhochschule Lübeck bisher nicht ermöglicht ist.

Plattform als Informationsquelle

Es besteht der Wunsch, die im Lernraum vorliegenden Informationsmaterialien, durchaus auch in aufbereiteter Form, durch die Applikation den Studienreden zugänglicher zu machen.

Applikation zur Unterstützung des Zeitmanagements

Es wird der Wunsch geäußert die Studierenden bei dem Zeitmanagement, unter anderem durch Erinnerungen, zu unterstützen. In diesem Rahmen wird der Vorschlag eingebracht möglichst detaillierte Beschreibungen von Arbeitspaketen in dem Tool zu verlangen, damit die Studierenden dazu gezwungen sind, sich rechtzeitig mit der Aufwandseinschätzung zu beschäftigen.

3.4.5. Die Applikation - Chancen und Risiken

In jedem Interview bekamen die Professoren abschließend die Möglichkeit, ihre Erwartungen an eine solche Applikation auszuführen und besonders auf die, aus ihrer Sicht, möglichen Chancen und Risiken einzugehen.

Diese Anmerkungen werden im folgenden Verlauf zusammengefasst dargestellt:

Chancen

- Die Applikation als neuer Kanal für die Studierenden, der dafür dienen kann, dass Studierende sich besser informieren können. Dies wird besonders in Bezug auf die Formalien einer Bachelorarbeit betont, da viele Studierende gar nicht wissen, was die Rahmenbedingungen einer Bachelorarbeit sind oder welche Regeln und Anforderungen es überhaupt gibt.
- Die Applikation kann im Gegensatz zum Bachelorseminar begleitend während der eigenen Bachelorarbeit genutzt werden kann. Dies kann dafür sorgen, dass die Aufnahmefähigkeit der Studierenden für Tipps, Empfehlungen und weiteren Aspekten gesteigert wird, da sie sich zu diesem Zeitpunkt mit dem Problem konfrontiert sehen und somit der Lerneffekt am höchsten ist. In diesem Ansatz wird auch die Chance erkannt, dass der Fokus der Studierenden zum richtigen Zeitpunkt auf bestimmte wichtige Fragen gelenkt werden kann und somit grobe Fehler minimiert werden können.
- Besseres Zeitmanagement der Studierenden und die somit geförderten organisatorischen Fähigkeiten der Studenten.
- Die App könnte Probleme im Projektmanagement und bei der Gestaltung der Dokumentation minimieren.
- Das Senken des Beratungsaufwandes für Professoren und somit das Minimieren von sich wiederholenden Arbeitsabläufen für die unterschiedlichen Bacheloranden wird als Chance genannt, da in einfacher Form auf eine Applikation verwiesen werden kann, die alle nötigen Informationen enthält.

Risiken

- Die Applikation kennt nicht den realen Stand der Bachelorarbeit, sondern die Studierenden sind für die Verwaltung selbst zuständig. Wenn der Benutzer eine der Aufgaben abhakt, stellt dies unter Umständen nicht den echten Zustand der Bachelorarbeit dar und könnte dem Studierenden einen falschen Eindruck des Fortschritts geben.
- Die Applikation regt dazu an, sich durch das Zeitmanagementtool zu über-planen, was dafür sorgt, dass der Benutzer von der eigentlichen Arbeit abgehalten wird. In diesem Rahmen kann die Applikation dem Benutzer nicht die Eigenverantwortung abnehmen. Der Studierende kann der Applikation nicht die Schuld für einen Misserfolg geben.
- Gamification-Elemente könnten unter Umständen einen sehr begrenzten Effekt haben, da sie kein Interesse an einem Thema wecken können, sondern das Grundinteresse aus der Eigenmotivation erzeugt werden muss.
- Befürchtung, dass Studenten gegebenenfalls die Applikation als Leitfaden als unumstößlich ansehen könnten und somit durch unterschiedliche Ansichten in einen Konflikt mit dem Betreuer geraten können.
- Die App könnte missverstanden werden als Ersatz für die persönliche Betreuung – insbesondere fachliche Aspekte wird eine App naturgemäß nicht abdecken können. Es könnte weiterhin zu einem “Device Mismatch“ kommen, da Bachelorarbeiten üblicherweise nicht an mobilen Endgeräten entstehen man müsste also immer zwei Geräte bedienen: Notebook/Desktop-PC und Smartphone.
- Die Applikation könnte den Studierenden zu sehr unterstützen, sodass die eigentliche Arbeit der Prüfungsleistung gar nicht von den Studierenden selbst ausgeführt wird, sondern durch die Applikation. Dies könnte zu einer Minderung des Lerneffektes bei der Bearbeitung der Bachelorarbeit führen.

3.5. Ergebnisse der Studierenden

Das folgende Kapitel liefert einen Einblick in die Ergebnisse der Befragung der Studierenden und beschreibt die in diesem Rahmen gewonnenen Erkenntnisse.

Im Fokus der Befragung stehen die Erwartungen an die Bachelorarbeit und den Betreuer und die typischen Probleme und Sorgen der Studierenden. Die Erkenntnisse der Befragung werden abschnittsweise in zusammengefasster Form dargestellt. Die gesamten Antworten der schriftlich befragten Studierenden ist in kategorisierter Form im Anhang zu finden (siehe A.1.2). Weiterhin stehen auch die nicht aufbereiteten Originalaufzeichnungen der schriftlichen Befragung zur Verfügung, allerdings aufgrund des Umfangs nur in elektronischer Form auf der CD im Anhang (siehe TODO).

Informationen zu der schriftlichen Befragung:

- Insgesamt wurden 17 Studenten schriftlich befragt.

3.5.1. Erwartungen an den Betreuer

In diesem Kapitel werden die ermittelten Erwartungen der Studierenden an den Betreuer einer Bachelorarbeit zusammengefasst beschrieben.

Teilen von Erfahrungen

Es werden sich Erkenntnisgewinne durch das Teilen von Erfahrungen der Betreuer gewünscht, was sich besonders auf die Arbeit mit externen Unternehmen bezieht. Hierzu zählen Hinweise zu Besonderheiten und die wichtigen Unterschiede zu einer internen Bachelorarbeit.

Weiterhin wird das gemeinsame Erarbeiten eines Themas, was bezüglich des Umfangs durch die Erfahrungen des Betreuers eine angemessene Form annehmen sollte, betont.

Kommunikation

Was die Kommunikationsbereitschaft betrifft, wird vor allem eine hohe Erreichbarkeit, das (schnelle) Beantworten von Fragen und die Möglichkeit, persönliche Treffen wahrnehmen zu können, von dem Betreuer erwartet.

Feedback

Grundsätzlich wird von dem Betreuer erwartet, dass dieser sich mit der Bachelorarbeit der Studierenden insofern beschäftigt, als dass die einzelnen Feedback-Punkte zum aktuellen Stand, Inhalt, Ideen und Umfang konstruktiv diskutiert werden können. Darüber hinaus erwarten die Studierenden, dass der Betreuer Hinweise bei Unverständlichkeiten der Dokumentation gibt und gegebenenfalls schlechte Ideen und Entwurfsentscheidungen hinterfragt.

In diesem Rahmen wird auch vereinzelt erwartet, dass der Betreuer Interesse an der Arbeit des Studierenden hat sowie Bereitschaft für die Beantwortung von Fragen und auf das Diskutieren von Lösungsmöglichkeiten eingeht.

Impulse des Betreuers

In wenigen Fällen erwarten die befragten Studierenden, dass der Betreuer aktiv an dem Prozess der Erstellung der Bachelorarbeit teilnimmt. Das bezieht sich auf die Unterstützung bei Zeitmanagement, dem Geben von Denkanstößen und Anregungen und das vom Betreuer ausgehende Einfordern von Leistungen.

3.5.2. Die größten Probleme der Studierenden

Die Studierenden wurden zu ihren, nach eigener Einschätzung, größten Unsicherheiten und Problemen gefragt und haben einige, sich im Laufe der Befragung wiederholende, Aspekte angesprochen, die im folgenden Verlauf zusammengefasst werden.

Schreiben der Dokumentation

Die von den Studierenden angesprochenen Punkte befassen sich in erster Linie mit Unsicherheiten bei der Einschätzung von Umfang, Inhalt und Aufbau der Dokumentation. Des Weiteren werden Rechtschreibung und der korrekte Umgang mit Quellen genannt.

Zeitmanagement

Der Punkt Zeitmanagement stellt für viele Studierende eine grundsätzliche Herausforderung dar.

Fachliche Probleme

Teilweise werden auch fachliche Probleme unterschiedlicher Natur als große Herausforderung genannt. Dazu zählen zum einen Probleme mit der im Rahmen der Bachelorarbeit zu verwendenden Hardware, andererseits aber auch Softwareprobleme, das Entwickeln von sinnvollen Tests und das präzise Ermitteln von Anforderungen.

Sonstige Probleme

Weitere Probleme, die vereinzelt genannt werden, sind Probleme mit der Themenfindung, die Kommunikation zwischen internem Betreuer und der externen Firma oder die allgemeine Überforderung bei der Bearbeitung der Bachelorarbeit.

3.5.3. Wunsch nach besserer Vorbereitung auf die Bachelorarbeit

Um das Bild der Situation der Studierenden besser zu verstehen und darüber hinaus auch Impulse für die Applikation zu erfassen, wurden die Studierenden hinsichtlich Verbesserungsvorschlägen für den Verlauf des Studiums, unter dem Aspekt, was ihnen zu einer besseren Vorbereitung zur Bearbeitung der Bachelorarbeit verhelfen würde oder verholfen hätte, befragt.

Zeitmanagement

Im Allgemeinen wünscht sich ein großer Teil der Studierenden eine bessere Vorbereitung bezüglich des Zeitmanagements. Die Studierenden bemängeln eine unzureichende Vorbereitung auf das Abschätzen von Zeitaufwand sowie Planung von größeren Projekten und nennen in diesem Zusammenhang den Wunsch nach Möglichkeiten und Maßnahmen im Studium, wie das häufigere Einbinden von Projektarbeiten.

Wissenschaftliches Arbeiten

Häufig wird der Wunsch nach besserer Vorbereitung auf das wissenschaftliche Arbeiten genannt, es gibt Unsicherheiten bezüglich des Aufbaus einer wissenschaftlichen Arbeit, des Umgangs mit Quellen und des korrekten Einbindens von Quellcode-Inhalten. Als vorkommender Lösungsvorschlag wird ein Kurs, welcher zur besseren Vorbereitung vor dem 6. Semester stattfinden sollte, genannt.

Aufklärung zur Bachelorarbeit

Viele der befragten Studierenden wünschen sich vorbereitende und informierende Maßnahmen zur Bachelorarbeit. Dies bezieht sich vor allem auf das Klären organisatorischer Fragen, der wichtigen Formalien und des Umfangs der Bachelorarbeit. Im Detail werden auch vorbereitende Maßnahmen gewünscht, die sich auf die Vor- und Nachteile einer extern oder intern betreuten Bachelorarbeit beziehen.

Sonstige Anmerkungen

Weiterhin treten bei der Befragung auch Fälle auf, bei denen keine expliziten Veränderungen gewünscht werden, da die Studierenden zufrieden sind. Der Wunsch nach einem stärkeren Fokus auf Design-Pattern im Laufe des Studiums wurde einmal geäußert.

3.5.4. Einbezug der Studierenden in die Gestaltung der Applikation

Die Einbindung der Studierenden befasst sich mit spezifischeren Fragen, hinsichtlich der motivierenden Gamification-Inhalte der zu entwickelnden Applikation. Der Verlauf der folgenden Befragungen fand somit nicht in einem Durchlauf statt, sondern ergab sich aus dem kontinuierlichen Einbinden der Studierenden. Es wurden bei der Befragung zwei Gamification-Ansätze vorgestellt.

Der erste Ansatz fällt unter anderem in die Benutzergruppe der **Socialiser** (Siehe Kapitel 2.1). Inhalt des Ansatzes ist es, die Motivation durch den sozialen Kontakt und Austausch mit anderen Teilnehmern zu erzeugen, indem diese den Fortschritt bei der Bearbeitung der Bachelorarbeit anderer Studierender sehen können.

Als Gegengewicht zum ersten Ansatz orientiert sich der zweite Ansatz an der Benutzergruppe **Achievers** (Siehe Kapitel 2.1). Dieser Ansatz beruht vor allem auf dem Abschließen und Sammeln von Erfolgen. Im Rahmen der Applikation wird es dem Benutzer bei dieser Idee ermöglicht, Erfolge durch Erfüllen von definierten Aufgaben zu erlangen und somit den Fortschritt des Projektes durch das Abschließen von geeigneten Aufgaben zu verfolgen.

Primäres Ziel des Einbezugs der Studierenden ist, zu bestimmen, welche Einbindung von Gamification-Strategien ansprechend auf die Zielgruppe der Studierenden wirken und in welcher Weise sich das Einbinden von Gamification-Elementen auf die Steigerung der Motivation von Studenten bei der Bearbeitung der Bachelorarbeit auswirkt.

In der folgenden Tabelle werden die Ergebnisse der Befragung verdeutlicht:

Art der Gamification-Strategie	Art der Reaktion	Häufigkeit
Motivation durch soziale Interaktion	Positiv	3
	Negativ	12
	Neutral	2
Sammeln von Achievements und Erfüllen von Aufgaben	Positiv	15
	Negativ	1
	Neutral	1
		17

Tabelle 3.2.: Einbezug der Studierende in Wahl der Gamification-Strategie

3.6. Untersuchung der Ergebnisse

Im folgenden Kapitel werden alle relevanten Kenntnisse, die aus der Datenerhebung und -auswertung hervorgehen, untersucht und analysiert. Dazu gehört die Identifikation des Problembereiches, die Auswahl der zu verfolgenden Motivationsstrategie bezüglich der Wahl der Spiel-Design-Elemente sowie die relevanten Produktfunktionen und deren abgeleitete Anforderungen, Chancen und Risiken.

3.6.1. Identifikation des Problembereichs

Durch die Befragung der Beteiligten und der Analyse der gewonnenen Informationen haben sich die Erwartungen der Beteiligten an die Benutzung der Applikation herausarbeiten lassen, welche in der folgenden Tabelle (siehe Listing 3.3) dargestellt sind.

Die dabei genannten Ziele und Erwartungen entsprechen dem allgemeinen Bild, das durch die Interviews und schriftlichen Befragungen ermittelt werden konnte, weshalb kleine Abweichungen in dieser Tabelle nicht mit aufgenommen wurden, da diese einen zu großen verzerrenden Einfluss auf das Gesamtbild hätten.

Bezeichnung der Beteiligten	Wünsche der Beteiligten
Professoren	Steigerung der Fragebereitschaft von Studierenden Weniger Aufwand in das sich wiederholende Informieren des Studierenden stecken Zielgerichtetere Vorbereitung auf Besprechung mit dem Betreuer Höhere Qualität von Abschlussarbeiten und Dokumentationen Bessere Zeitplanung der Studierenden Systematische(re) Vorgehensweise der Studierenden Optimale Vorbereitung auf die Bearbeitung der Bachelorarbeit
Studierende	Optimales Ergebnis der Bachelorarbeit Kenntnisse über Formalien, Umfang und Ablauf der Bachelorarbeit Kontrolle über die Bachelorarbeit und den Ablauf Unterstützung Weniger Fallstricke bei der Bearbeitung der Bachelorarbeit

Tabelle 3.3.: Erwartung der Beteiligten an die Applikation

Aus den Ergebnissen der Befragungen geht hervor, dass die Probleme nicht in den unterschiedlichen Ansichten der Studierenden und der Professoren liegen, da beide beteiligte Gruppen sehr kompatible Erwartungen und Ziele besitzen.

Häufig wird von den Studierenden geäußert, dass sie sich unzureichend vorbereitet für die Bachelorarbeit fühlen und die größten Probleme der Studierenden im Bereich Zeitmanagement und dem wissenschaftlichen Arbeiten liegen. Zusätzlich äußern sich die Studierenden zu

fehlenden oder unzureichend aufgebauten Kenntnissen bezüglich der Rahmenbedingungen und weiteren Informationen zu Bachelorarbeiten als Ganzes.

Die Professoren sehen die Probleme bei den Studierenden eindeutig auch bei dem Zeitmanagement, dem wissenschaftlichen Arbeiten, der Kommunikation und einer mangelnden Vorbereitung, bezüglich der Formalitäten und des Ablaufs einer Bachelorarbeit.

Hervorzuheben ist der Aspekt, dass die existierenden Probleme sowohl den Professoren, als auch den Studierenden bewusst sind und unabhängig der unterschiedlich ausfallenden Sichtweisen übereinstimmen.

Somit lässt sich erwarten, dass die angestrebten Ziele der Studierenden und die Erwartungen der Professoren aufgrund dieser Wissens- und Erfahrungslücken unter Umständen nicht reibungslos im Rahmen der Bachelorarbeit erfüllt werden können. Auf diese Weise werden die Erwartungen der Professoren sowie die Ziele der Studierenden gefährdet und können unter Umständen nicht erfüllt werden, obwohl dies im Sinne beider Beteiligten ist.

3.6.2. Auswahl der Motivationsstrategie

Wie aus den Ergebnissen in Kapitel 3.5 in Tabelle 3.2 abzulesen ist, äußern sich die befragten Studierenden gegenüber dem vorgestellten Designvorschlag, welcher die soziale Einbindung der Studierenden vorsieht, überwiegend sehr kritisch. Im Vergleich dazu wird die vorgestellte Variante des Einsetzens von Achievements und Aufgaben in Kombination mit Herausforderungen stark positiv bewertet.

Aufgrund der eindeutigen Reaktionen der eingebundenen Studierenden wird die Entscheidung getroffen, den ersten Ansatz nicht weiter zu verfolgen, dafür aber den zweiten Ansatz auszubauen. Somit fallen die Nutzergruppen überwiegend in den Spannungsbereich der **Explorers** und **Achievers**, wie in Abbildung 3.2 verdeutlicht wird.

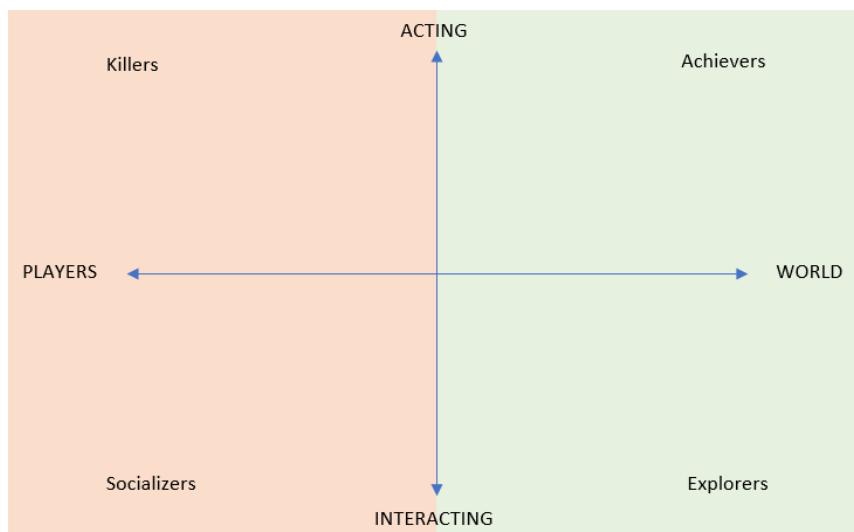


Abbildung 3.2.: Benutzeroberfläche Fortschritts-Management

Durch den vorgesehenen Kontext der Applikation, die Studierenden während der Bearbeitung der Bachelorarbeit wegweisend zu unterstützen und zu motivieren, bietet sich somit an,

Herausforderungen und die damit verbundenen Achievements in Form von kontextbezogenen fokuslenkenden Game-Design-Elementen (siehe 2.1) in die Applikation als festen Bestandteil der Motivationsstrategie zu integrieren.

Wie in [37][S. 32 - 34] beschrieben wird, besitzen Achievements verschiedene Eigenschaften, die in Bezug auf Zielstellungen einer steuernden und in diesem Sinne auch fordernden Funktion dienen können. Ein wichtiges Kriterium bei der Wahl der Achievements als Spiel-Design-Elemente ist, dass sie dem Benutzer ein nicht-kontrollierendes, positives Feedback geben und somit nicht unangenehm kontrollieren oder den ohnehin schon sehr hoch ausfallenden Druck auf die Studierenden bei der Bearbeitung der Bachelorarbeit erhöhen. Es sollte in diesem Sinne nicht aus den Augen verloren werden, dass die Applikation den Studierenden positiv unterstützen soll.

Im weiteren Verlauf werden, den Nutzergruppen entsprechend, auf Spiel-Design-Elemente wie Bestenlisten oder andere soziale Motivationsmodelle verzichtet.

Weiterhin wird das Integrieren eines Punkte- oder Level-Systems sowie einer Avatar-Funktion nicht verfolgt. Dies dient der Absicht, dass die Applikation zwar motivierende Spiel-Design-Elemente enthalten soll, jedoch in einem Umfang, der keine an Gamification desinteressierten Benutzer ausschließen und somit in diesem Fall weiterhin eine brauchbare Hilfestellung bieten soll.

3.6.3. Übersicht der abgeleiteten Produktfunktionen und Anforderungen

Im folgenden Verlauf sind die aus der Aufgabenstellung und der Anforderungsanalyse erhobenen Produktanforderungen in natürlicher Sprache vorzufinden. Diese wurden an einer Anforderungsschablone aus dem Lehrbuch der Softwaretechnik von Helmut Balzert orientiert [5, Kapitel 19 - Natürlichsprachliche Anforderungen, Seite 481] und nach Bedarf spezifiziert:

PR1 Unterstützung bei Zeit- und Fortschrittsmanagement

[Req1.1] Die Software muss den Studierenden ermöglichen, Meilensteine zu verwalten.

[Req1.1.1] Das Anlegen von Meilensteinen

[Req1.1.2] Das Entfernen von Meilensteinen

[Req1.1.3] Das Ändern von Meilensteinen

[Req1.1.4] Das Verschieben von Meilensteinen

[Req1.2] Die Software muss dem Studierenden ermöglichen, dem Meilensteinen zugehörige Aufgaben zu verwalten.

[Req1.2.1] Das Anlegen von Aufgaben

[Req1.2.2] Das Entfernen von Aufgaben

[Req1.2.3] Das Ändern von Aufgaben

[Req1.2.4] Das Markieren von Aufgaben als *abgeschlossen*

[Req1.3] Die Software soll die Studierenden auf bevorstehende Meilensteine und Aufgabenpakete aufmerksam machen.

[Req1.3.1] Das Auslösen von Erinnerungen/Benachrichtigungen

PR2 Unterstützung bei der Bearbeitung der Bachelorarbeit.

- [Req2.1] Die Software muss den Studierenden allgemeine Informationen zur Bearbeitung der Bachelorarbeit bereitstellen.
 - [Req2.1.1] Die Software muss den Studierenden Informationen über die Rahmenbedingungen und Formalien der Bachelorarbeit an der Fachhochschule Lübeck bereitstellen können.
 - [Req2.1.2] Die Software muss den Studierenden Hinweise und Empfehlungen zu Aufbau und Struktur von Bachelorarbeiten bereitstellen können.
 - [Req2.1.3] Die Software muss den Studierenden Hinweise und Empfehlungen zur Durchführung von Bachelorarbeiten bereitstellen können.
-
- [Req2.2] Die Software muss den Studierenden eine Übersicht über studiengang-spezifische und -übergreifende relevante Techniken und Methoden bereitstellen können.
- [Req2.4.1] Hilfestellung/Übersicht zu Themenfindung für Bachelorarbeit
 - [Req2.4.2] Hilfestellung/Übersicht von studiengang-spezifischen Inhalten
 - [Req2.4.3] Hilfestellung/Übersicht zu Aufbereitung von Ergebnissen
 - [Req2.4.4] Hilfestellung/Übersicht zu Nachweisführung
-
- [Req2.3] Die Software muss den Umgang von zielgruppenspezifischen Anpassungen der Inhalte ermöglichen können.
-
- [Req2.4] Die Anpassungen der Inhalte soll software-extern realisierbar und integrierbar sein.

PR3 Einsatz von Gamification-Elementen zur Steigerung der Motivation

- [Req3.1] Die Software muss im Rahmen der Gamification-Strategie Spiel-Design-Elemente bereitstellen, die den ermittelten Benutzerklassen entsprechen.
-
- [Req3.2] Die Spiel-Design-Elemente sollen unter Berücksichtigung der wegweisenden Eigenschaft geeignet an dem Ablauf einer Bachelorarbeit orientiert sein.

PR4 Weitere Anforderungen

- [Req4.1] Die Software muss sämtliche Inhalte und Leistungen auch für fremdsprachige Personengruppen enthalten.
-
- [Req4.2] Die Software kann die Möglichkeit bieten, dass Betreuer ihre persönlichen Themenvorschläge für die Studierenden sichtbar per Weboberfläche hinzufügen/entfernen können

3.6.4. Übersicht der Risiken

Im Laufe der mit den Professoren durchgeführten Interviews wurden häufig Bedenken bezüglich der Risiken geäußert, die der Einsatz der Software für die Studierenden mit sich bringen könnte.

Es folgt die Aufzählung der ermittelten Risiken sowie einer Gewichtung in Bezug auf die Stärke der negativen Auswirkungen bei Eintreten dieser Risiken:

[R1] Auswirkungen: Hoch

Die Applikation kennt den realen Zustand und Fortschritt der Bachelorarbeit nicht und könnte somit zu einer Verfälschung des realen Bildes führen. Studierende könnten somit den eigentlichen Zustand der Arbeit aus den Augen verlieren.

[R2] Auswirkungen: Hoch

Studierende könnten die Hilfestellung der Applikation als unumstößlich ansehen und so unter Umständen in Konflikt mit dem Betreuer kommen.

[R3] Auswirkungen: Hoch

Die Applikation kann als Ersatz für die Betreuung von einem Professor oder einer Professorin wahrgenommen werden, was zu weitreichenden Konsequenzen für die Qualität der Abschlussarbeit führen könnte.

[R4] Auswirkungen: Hoch

Durch die Nutzung der Applikation wird dem Studierenden so viel Arbeit abgenommen, dass der Lerneffekt der Bachelorarbeit zu gering ist.

[R5] Auswirkungen: Mittel

Die Applikation könnte im Rahmen des Zeitmanagement-Tools zu einer Überplanung führen, was den Studierenden von der eigentlichen Arbeit abhalten könnte.

[R6] Auswirkungen: Gering

Gamification-Elemente können einen sehr geringen Effekt haben, da nicht sie die Motivation aufrechterhalten, sondern das Interesse des Studierenden.

3.6.5. Übersicht der Chancen

Aus den gewonnenen Informationen, die sich aus dem Verlauf der Interviews mit den Professoren und der schriftlichen Befragung mit den Studierenden ergeben haben, ließen sich Chancen für Betreuer und Studierende durch Einsatz der Applikation identifizieren, welche im Folgenden genannt werden.

Es folgt die Aufzählung der ermittelten Chancen sowie einer Gewichtung in Bezug auf die Stärke der positiven Auswirkungen bei Erreichen dieser Chancen:

[C1] Auswirkungen: Hoch

Aufnahmebereitschaft gegenüber Tipps, Empfehlungen und anderen Aspekten könnte im Gegensatz zum Bachelorarbeit-Seminar gesteigert werden, da der Einsatz der Applikation von begleitender Natur ist und sich die Studierenden somit besser mit dem Problem identifizieren könnten.

[C2] Auswirkungen: Hoch

Vermeiden von großen Fehlern durch Lenken des Fokus der Studierenden auf wichtige Aspekte der Bachelorarbeit, bei denen üblicherweise häufig Probleme auftreten.

[C3] Auswirkungen: Hoch

Besseres Zeitmanagement der Studierenden und die Steigerung der organisatorischen Fähigkeiten durch Anwendung der Applikation.

[C4] Auswirkungen: Hoch

Systematischere Vorgehensweise der Studierenden bei Problemanalyse, Anforderungserhebung und Nachweisführung

[C5] Auswirkungen: Hoch

Senken des Beratungsaufwandes seitens der Betreuer bei Trivialitäten und bei sich wiederholenden Fragestellungen.

[C6] Auswirkungen: Mittel

Neuer Informationskanal für Studierende, welcher Wissenslücken bezüglich der Rahmenbedingungen und Regeln einer Bachelorarbeit schließen kann.

[C7] Auswirkungen: Mittel

Zielgerichteterer Methoden- und Werkzeugeinsatz sowie Anwendung fachspezifischer und fachübergreifender Techniken.

[C8] Auswirkungen: Mittel

Minimieren von Problemen bei der Gestaltung der Dokumentation.

[C9] Auswirkungen: Gering

Bessere Vorbereitung der Studierenden auf Gesprächstermine mit dem Betreuer.

4. Konzeptvorstellung der Applikation

4.1. Beschreibung der Software

Die zu entwickelnde Applikation kann in verschiedene Kern-Softwareabschnitten unterteilt werden. Diese Abschnitte lassen sich weiterhin auf folgende namensgebende Aufgabenbereiche aufteilen, welche sich an den erhobenen Anforderungen und der Aufgabenstellung orientieren (siehe Kapitel 3.6.3).

- Meilensteinverwaltung
- Bachelorarbeit Guide
- Herausforderungen
- Achievements
- Dashboard
- Sonstige Softwareinhalte

4.1.1. Meilensteinverwaltung

Die Meilensteinverwaltung ist eine Funktion, die das Anlegen, Planen und Verwalten von Meilensteinen sowie der zugehörigen Unteraufgaben ermöglichen soll.

Ferner soll die Meilensteinverwaltung nicht den detaillierten Prozess des gesamten Projektmanagements abbilden, sondern die Aufgabe der übersichtlichen Handhabung von Unteraufgaben und Verdeutlichung von Fristen in Form einer dynamischen ToDo-Listenverwaltung unterstützen. Hierbei soll dem Anwender weiterhin die größtmögliche Freiheit der Gestaltung des Plans zukommen, um individuelle Arbeitsweisen zu berücksichtigen.

Der im Projektmanagement definierte Begriff "Meilenstein" bezeichnet hierbei einen Zeitpunkt, zu welchem der vom Nutzer definierte Arbeitsstand erreicht sein soll.

Es folgt eine stichpunktartige Auflistung und Beschreibung der Kernfeatures des Tools:

- **Erstellung und Verwaltung von Meilensteinen**

Es können Meilensteine erstellt [Req1.1.1], bearbeitet [Req1.1.3], verschoben [Req1.1.4] und entfernt [Req1.1.2] werden.

- **Erstellung und Verwaltung von Unteraufgaben**

Jeder Meilenstein enthält optionale Unteraufgaben [Req1.2], die von dem Benutzer einem Meilenstein hinzugefügt [Req1.2.1], bearbeitet [Req1.2.3] und gelöscht [Req1.2.2] werden können. Weiterhin lassen sich die Unteraufgaben als *abgeschlossen* markieren [Req1.2.4], um dem Benutzer die Möglichkeit zu bieten, seine Aufgaben zu organisieren.

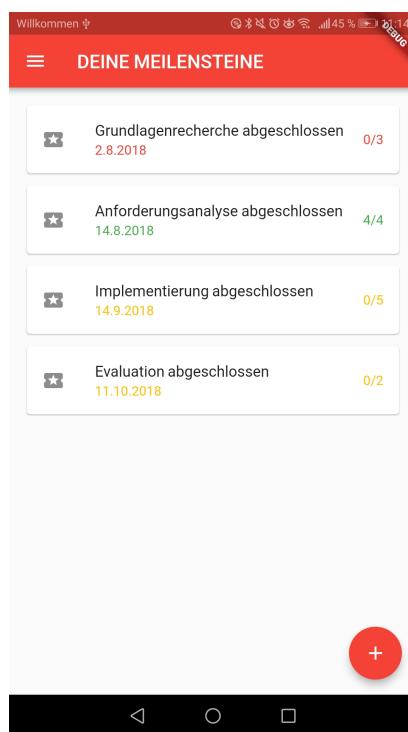


Abbildung 4.1.: Benutzeroberfläche Meilensteinverwaltung

4.1.2. Bachelorarbeit Guide

Der Guide stellt den Teil der Applikation dar, der die Bereitstellung von Hinweisen, Tipps und weiteren hilfreichen Informationen zur Bearbeitung der Bachelorarbeit abdecken soll. Dieser Inhalt ergibt sich aus der Verwendung des von der Fachhochschule Lübeck ausgehändigten Dokuments, welches als Ratgeber bei der Erstellung von Bachelorarbeiten funktionieren und somit die speziellen Anforderungen der Fachhochschule Lübeck berücksichtigen soll [2, vgl. Kapitel 1].

In dieser Hinsicht soll der Guide als genereller Anlaufpunkt fungieren, der eine Hilfestellung für Bacheloranden darstellt/*Req2.2*.

Die Bandbreite der Hinweise, Tipps und weiteren Informationen sollen somit den gesamten Verlauf der Bachelorarbeit abdecken und lässt sich in diesem Zusammenhang in folgende Bereiche und Inhalte zerlegen, welche sich an dem genannten Leitfaden der Fachhochschule Lübeck zur Erstellung einer Bachelorarbeit orientieren.

The image displays two screenshots of the Bachelorarbeit Guide application. Both screenshots show a red header bar with the text "WILLKOMMEN" and "BACHELORARBEIT GUIDE". The left screenshot, labeled (a), shows the "Hauptbildschirm" (main screen) with a sidebar containing links: "Formalien, Voraussetzungen und Fristen", "Betreuung der Bachelorarbeit", "Formulierung des Themas", "Aufbau und Struktur", "Hinweise zum Schreiben", "FAQ", and "Methodenübersicht". The right screenshot, labeled (b), shows the "Themen-Ebene" (topic level) with a sidebar containing links: "Was macht der Erstprüfer?", "Wer kann dein Erstprüfer sein?", "Zweitprüfer" (expanded), and "Unterschiede bei der Betreuung". The "Zweitprüfer" section contains a detailed text about the role of the second supervisor. Both screenshots have a black footer bar with navigation icons: back, forward, and search.

Abbildung 4.2.: Benutzeroberfläche Guide

Es folgt eine Interpretation und Beschreibung der Unterteilung der Themengebiete:

Anmerkung: Die folgenden Themengebiete stellen nur eine Möglichkeit der Aufteilung dar. Bei Einsatz der Software sollen die hier dargestellten Inhalte individuell erweitert oder verändert werden können([Req2.3], [Req2.4]).

- **Formalien, Voraussetzungen und Fristen**

Dieser Abschnitt beinhaltet die allgemeinen Informationen, die vor allem zur Klärung der Formalien bei der Bearbeitung der Abschlussarbeit wichtig sind. [Req2.1.1]

- **Formulierung des Themas**

In dieser Sektion befinden sich zahlreiche Tipps und hilfreiches Wissen zur Formulierung des eigenen Themas und der zu verfassenden Aufgabenstellung.

- **Betreuung der Bachelorarbeit**

Informationen zur Betreuung von Bachelorarbeiten und was hinsichtlich des Engagements und der Vorgehensweise von den Studierenden erwartet wird, sind in diesem Abschnitt zu finden.

- **Aufbau und Struktur**

Dieser Abschnitt befasst sich näher mit dem Aufbau der Bachelorarbeit und stellt vor allem Informationen und Empfehlungen bereit, welche sich auf die Strukturierung und die Bedeutung der einzelnen Kapitel der Bachelorarbeit beziehen. [Req2.1.2]

- **Hinweise zum Schreiben**

In diesem Abschnitt werden tiefgehende Informationen und Empfehlungen behandelt, welche sich auf die praktische Umsetzung des Schreibens im Detail beziehen. Beispielahaft hierfür sind der Schreibstil, die Verwendung von Zeiten oder die Einbindung von Bildern, Tabellen und Programmcode. [Req2.1.3]

- **Methodenübersicht**

Dieser Abschnitt bietet eine Übersicht wichtiger Methoden der Informatik, die sich bei der Bearbeitung und Dokumentation der Bachelorarbeit als hilfreich erweisen könnten. [Req2.2]

- **FAQ**

Dieses Kapitel behandelt oft gestellte Fragen und verdeutlicht die Informationen in der Form eines Frage-Antwort Schemas.

4.1.3. Herausforderungen

Die sogenannten Herausforderungen [Req3.1] sollen im Rahmen der Applikation, im Gegensatz zu den frei definierbaren Meilensteinen, eine fest definierte kategorisierte Aufteilung von typischen im Verlauf einer Bachelorarbeit zu leistenden Aufgaben darstellen [Req3.2].

Diese Aufgaben sollen vor allem als interaktiver Leitfaden dienen, an welchem sich die Studierenden orientieren können. Hierbei wird das Ziel angestrebt, den Studierenden eine Perspektive auf alle Phasen einer Bachelorarbeit sowie den jeweiligen einzelnen Arbeitsschritten zu geben, welche von den Studierenden abgehakt werden können.

Durch die in der Anforderungsanalyse ermittelten Daten hat sich der Eindruck ergeben, dass viele Studierende gerade zu Beginn der Bearbeitungszeit dazu neigen, die Arbeit, aufgrund der noch viel verbleibenden Zeit, aufzuschieben. Unter diesem Aspekt wird hierbei versucht, durch das Abhaken von kleinen konkreten Aufgaben bereits frühzeitig einen fließenden Arbeitsablauf (Workflow) seitens der Studierenden anzustreben.

Hinweis: Der Inhalt und Ablauf der dargestellten Herausforderungen ist im Rahmen der Prototypentwicklung lediglich beispielhaft erfolgt und stellt somit keinen korrekten und vollständigen Entwurf dar

The image shows two screenshots of a mobile application interface. The left screenshot, labeled (a), shows the main menu with three categories: 'Anfangsphase' (0/15), 'Bearbeitungsphase' (0/5), and 'Abschlussphase' (0/6). The right screenshot, labeled (b), shows a detailed view of the 'ANFANGSPHASE' (Initial Phase) with a list of tasks:

- 1. Beantrage deine Zulassung zur Bachelorarbeit
- 2. Finde das Thema deiner Bachelorarbeit
- 3. Suche dir deinen Betreuer/deine Firma für die Bachelorarbeit
- 4. Bereite deinen Arbeitsplatz für die Arbeit vor
- 5. Schreibe deine Aufgabenstellung
- 6. Erstelle die Gliederung für deine Arbeit
- 7. Erstelle deinen Zeitplan
- 8. Arbeite dich in deine Schreibumgebung ein (Beispiel „LaTeX“)

Abbildung 4.3.: Benutzeroberfläche Herausforderungen

4.1.4. Achievements

Der Achievement-Hauptbildschirm [Req3.1] enthält eine Übersicht der verschiedenen Kategorien von Achievements (siehe Abbildung 4.7). Die nächste Ebene dieses Abschnitts zeigt die detailliertere Ansicht der jeweiligen Achievements an, bei welcher der Benutzer einsehen kann, welche Achievements er bereits erhalten hat.

Hinweis: Weitere Informationen zu den einzelnen Achievements und deren Kategorien finden sich im Kapitel 4.2.

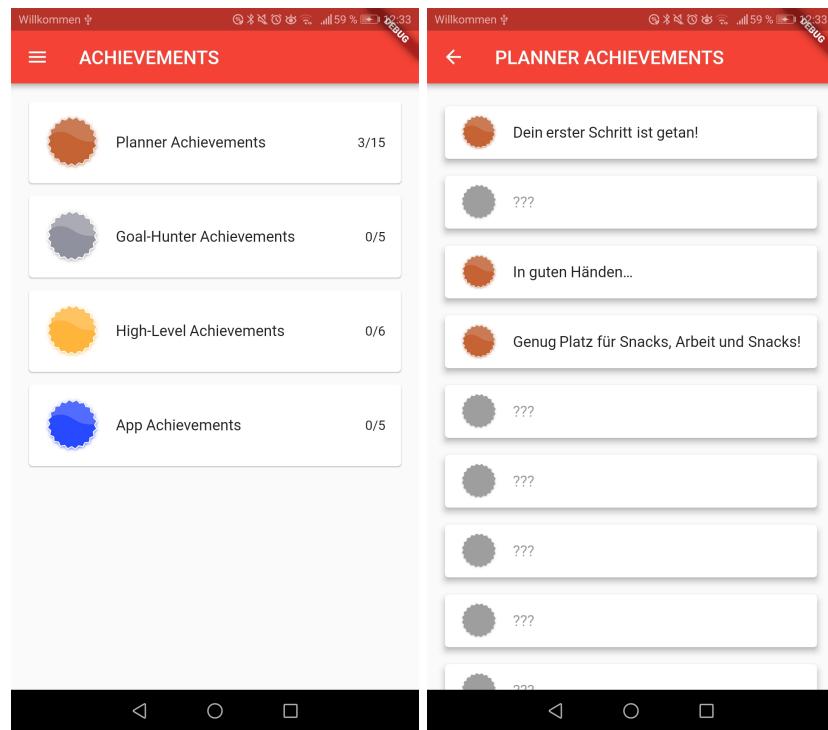


Abbildung 4.4.: Benutzeroberfläche Achievements

4.1.5. Sonstige Softwareinhalte

Über die eigentlichen individuellen Funktionalitäten hinaus sind weitere Grundfunktionalitäten vorhanden, welche unter den Punkt **Sonstige Softwareinhalte** fallen.

Es folgt eine stichpunktartige Auflistung und Beschreibung der Inhalte:

- **Menü**

Das Menü zeigt die existierenden Softwareabschnitte in einer klassisch gelisteten Menüstruktur (siehe Abbildung 4.5).

- **Einstellungen**

Die Einstellungen beinhalten Optionen, die sich auf die Eigenschaften der Applikation auswirken, beispielsweise das Verändern des Designs oder der Sprache/*Req4.1*

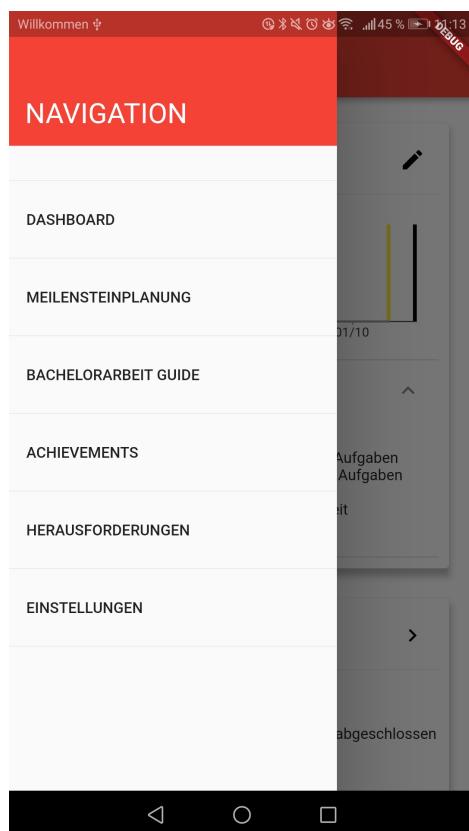


Abbildung 4.5.: Benutzeroberfläche Navigations-Menü

4.1.6. Dashboard

Das Dashboard stellt den Ausgangspunkt und Hauptbildschirm der Applikation dar und trägt somit die Aufgabe, aktuelle Informationen für den Nutzer aufbereitet anzuzeigen und die verschiedenen Funktionen der Applikation im Bund darzustellen.

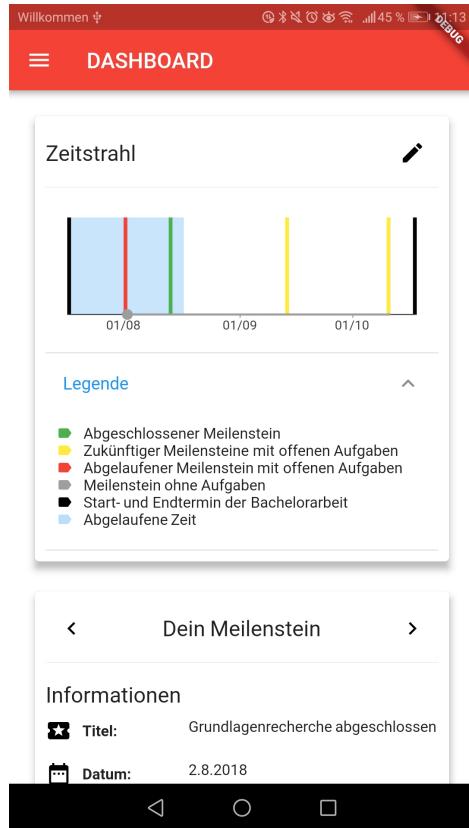


Abbildung 4.6.: Benutzeroberfläche Dashboard

Der auf Abbildung 4.6 Ausschnitt zeigt einen Zeitstrahl als primäres Fotschrittsverfolgungs-Tool. Dieser Zeitstrahl zeigt die von dem Nutzer erstellte Meilensteine sowie deren zugehörigen Status in Form einer spezifischen Farbvisualisation. Die hierbei übermittelten Informationen ermöglichen dem Benutzer einen Überblick über abgelaufene (rot), aktive (gelb) erfüllte (grün) und statuslose (grau) Meilensteine.

Des weiteren wird dem Benutzer durch einen blau visualisierten Bereich die bereits abgelaufene Zeit verdeutlicht, welche zwischen Start und Endpunkt der Bearbeitungszeit auf einfache Weise eine bessere Einschätzung von Fortschritt- und Zeitmanagement ermöglichen soll.

Auf dem Dashboard befindet sich auch ein Abschnitt, welcher den aktuellen ausgewählten Meilenstein anzeigt. Dieser kann durch Interaktion mit dem Zeitstrahl oder mit den dafür vorgesehenen Pfeilen ausgewählt werden, so dass jederzeit Unteraufgaben abgehakt oder hinzugefügt werden können.

Es folgt eine stichpunktartige Auflistung und Beschreibung weiterer möglicher Ideen zu Inhalten des Dashboards, welche bei der Planung der Software entwickelt wurden und durch einen frühen Oberflächenprototyp in die Anforderungsanalyse eingebunden wurde (siehe Anhang TODO):

- **Dynamische Anzeige des Fortschritts**

Der bisher geleistete Gesamtfortschritt wird verdeutlicht, indem zu sehen ist, wie viele Meilensteine in Relation zu den Gesamtmeilensteinen bisher abschlossen wurden. Weiterhin wird der bisher geleistete Wochenfortschritt verdeutlicht, indem zu sehen ist, wie viele Aufgabenpakete in der Phase bis zum nächsten Meilenstein, in Relation zu den vorher definierten gesamten Arbeitspaketen, schon erfüllt wurden.

- **Anzeige der noch offenen Herausforderungen**

Es wird angezeigt, wie viele Herausforderungen der jeweiligen Kategorien abgeschlossen wurden, in Relation zu deren Gesamtanzahl.

- **Dynamische Anzeige der letzten erreichten Achievements**

Die Anzeige zeigt die letzten erreichten Achievements an. Zugehörig sind in diesem Fall die Darstellung des Typs des Achievements in Form der jeweiligen Färbung und des Titels der Herausforderung. Das zuletzt erreichte Achievement wird hierbei durch Größe der Darstellung und Beschreibung hervorgehoben.

- **Dynamische Anzeige von allgemeinen Tipps**

Die Anzeige zeigt zufällig ausgewählte allgemeine Tipps zur Bearbeitung der Bachelorarbeit, die im Laufe der Zeit wechseln.

4.1.7. Übersicht der Produktfunktionen

Die unterschiedlichen Abschnitte der Software stehen unterschiedlich stark in Verbindung mit den verschiedenen definierten Produktanforderungen. Um eine Übersicht über die groben Zusammenhänge zu ermöglichen, folgt eine zusammenfassende Tabelle (Siehe Tabelle 4.1).

Bezeichnung	Beschreibung	A1	A2	A3	A4	A5	A6
[Req1.1]	(Verwaltung von Meilensteinen)	●				●	
[Req1.2]	(Verwaltung von Aufgaben)	●				●	
[Req1.3]	(Erinnerung an Meilensteine)	●				●	
[Req2.1.1]	(Hinweise zu Rahmenbedingungen und Formalien)		●				
[Req2.1.2]	(Hinweise zu Aufbau und Struktur)		●				
[Req2.1.3]	(Hinweise zu Durchführung)		●				
[Req2.2]	(Übersicht von Methoden und Techniken)		●				
[Req2.3]	(Anpassbarkeit an Zielgruppe)		●				
[Req2.4]	(Software-externe Anpassung)		●				
[Req3.1]	(Benutzerklassen-orientierte Spiel-Design-Elemente)			●	●	○	
[Req3.2]	(Spiel-Design-Elemente sinnvoll an Bachelorarbeit orientieren)			●	●		
[Req4.1]	(Mehrsprachigkeit)						○
[Req4.2]	(Bachelorarbeitsthemen über Applikation ausschreiben)						○

A1 - Meilensteinverwaltung

A2 - Bachelorarbeit Guide

A3 - Herausforderungen

A4 - Achievements

A5 - Dashboard

A6 - Sonstige Softwareinhalte

● - Teil der Applikation

○ - Als Erweiterung vorgesehen

Tabelle 4.1.: Abdeckung der Produktfunktionen

4.1.8. Umgang mit Risiken

Im folgenden Verlauf wird der Umgang mit den, in Kapitel 3.6.4 dargestellten, Risiken diskutiert und beschrieben.

Da sich die Herausforderungen an dem Ablauf einer Bachelorarbeit orientieren und den Studierenden somit vor typische Aufgaben stellt, kam es im Verlauf der Interviews zu den Befürchtungen [R1], dass die Applikation nicht den realen Zustand der Bachelorarbeit kennt und somit der Studierende durch das Erfüllen von Herausforderungen ein Gefühl von Erfolg hat, obwohl diese Herausforderungen nicht den geforderten Qualitäten entsprechen und der Studierende dies durch das positive Feedback nicht merkt.

Um diesem Risiko entgegenzuwirken sollen die erstellten Herausforderungen möglichst auffordernder Natur sein (Beispiel: Beginne deine Einleitung). Bei Einsatz von Herausforderungen, welche auf inhaltliche Aspekte abzielen, sollen Checklisten zum Abschließen der Herausforderungen genutzt werden, um den Fokus auf eine allgemeingültige Vorgehensweise zu reduzieren, für deren Erfüllung die Studierenden jedoch selbst verantwortlich sind. Auch hier wird eine klare Kommunikation durch die Applikation als zwingend erforderlich angesehen.

Die Befürchtung [R4], dass die Applikation den Studierenden so viel Arbeit abnimmt, dass sie zu wenig eigenständige Erfahrung sammeln und somit der Lerneffekt zu gering ist, wird berücksichtigt, indem die Applikation einen eher informierenden Charakter aufweist, und keine Funktionalität bezüglich dem Liefern von Ergebnissen vorsieht.

Die Gefahren der Risiken [R2] und [R3], die sich damit befassen, dass die Applikation fälschlicherweise als Ersatz eines Betreuers wahrgenommen oder der Inhalt der Applikation als unumstößlich angesehen werden könnte, soll durch eine klare Kommunikation mit den Anwendern reduziert werden. Es werden als Maßnahmen einerseits immer die Quellen der Informationen angegeben und weiterhin eine allgemein sichtbare Erklärung zur Anwendung oder Fehlanwendung der Applikation deutlich kommuniziert.

Das identifizierte Risiko [R5], welches die Befürchtung einer Überplanung der Studierenden beinhaltet, was die Auswirkung haben kann, dass die Studierenden von der eigentlichen Arbeit abgehalten werden, wird als nicht ausschlaggebend betrachtet. Studierende, die zur Überplanung neigen, würden dies auch ohne Applikation tun.

Wichtiger wird in dieser Hinsicht die Freiheit der einzelnen Planungs-Bedürfnisse durch die individuelle Arbeitsweise der Studierenden angesehen, weshalb keine weiteren Maßnahmen zur Reduzierung der Planungsfreiheit getroffen werden. Weiterhin steht die Intensität der Planung nicht in der Verantwortung der Applikation, sondern in der Verantwortung des Benutzers.

Das Risiko [R6], das sich mit dem Nichtwirken der Gamification-Motivation befasst, wird als nicht weitreichend betrachtet, da die Gamification-Inhalte zwar Bestandteil der Applikation, jedoch nicht zwangsläufig zur Benutzung der Applikation sind. Die Spiel-Design-Elemente sollen nicht das Interesse an einem bestimmten Thema steigern, sondern dienen im Kontext der Bachelorarbeit einer wegweisenden Natur, welche motivierende Einflüsse auf die Bearbeitenden hervorrufen können, jedoch nicht müssen, um ihren Nutzen zu erfüllen.

4.2. Beschreibung der Gamification-Elemente

Im folgenden Kapitel werden die Gamification-Elemente, die in der Applikation zum Einsatz kommen, näher beschrieben. Dabei wird vor allem auf die identifizierten Kategorien der Achievements eingegangen, welche den Kern der gewählten Motivationsstrategie bilden. Darüber hinaus wird auch beschrieben, in welcher Form die gewählte Motivationsstrategie in der Realität, in Bezug auf den Fortschritt bei der Bearbeitung der Bachelorarbeit, funktioniert.

4.2.1. Achievements

Die Achievements verfolgen im Rahmen der Applikation die Aufgabe, motivierende Inhalte vor allem für die identifizierten Benutzergruppen der **Achiever** und **Explorer** (siehe 2.1.3) zu bieten. Dies soll erreicht werden, indem die Achievements dem Benutzer als Belohnung für eine abgeschlossene Herausforderung als Sammelstück hinzugefügt werden.

Im Rahmen der gewählten Motivationsstrategie werden verschiedenen Arten von Achievements in die Applikation integriert. Dies passiert unter anderem mit dem Ziel, eine an den Ablauf einer Bachelorarbeit gekoppelte Abstufung von Fortschritt und Erfolgen zu ermöglichen. Weiterhin werden auch die sogenannten *App-Achievements* Einzug in die Applikation erhalten. Diese sollen weitestgehend die Bedürfnisse der Benutzergruppe **Explorer** ansprechen und im Gegensatz zu den anderen Achievements nicht an einer konkreten Herausforderung gekoppelt sein, sondern im Laufe der Nutzung der Applikation durch *versteckte Aufgaben* freigeschaltet und offen gelegt werden können.

Um den Achievements, über das Sammeln und die Vervollständigung hinaus, eine Wertigkeit zu geben, werden diese durch verschiedene visualisierte Medaillen dargestellt.



Abbildung 4.7.: Visualisierung der Achievements

Diese Achievements basieren auf einem Bild aus der gemeinfreien Bilderdatenbank Pixabay (vgl. [31]) und wurden mit einer Bilderbearbeitungssoftware für die eigenen Zwecke bearbeitet.

Es folgt eine Übersicht und Beschreibung der verschiedenen Achievement-Typen:

- **Anfangsphase-Achievements** (Bronze)

Beinhaltete Achievements, die an die Herausforderungen gekoppelt sind, die nach Beginn der Bachelorarbeit erfolgen, sich aber noch nicht mit den einzelnen inhaltlichen Schritten der Bachelorarbeit auseinandersetzen.

- **Bearbeitungsphase-Achievements** (Silber)

Beinhaltete Achievements, welche an die Herausforderungen gekoppelt sind, die nach Beginn der Bachelorarbeit erfolgen und sich mit der eigentlichen Bearbeitung der Bachelorarbeit auseinandersetzen.

- **Abschlussphase-Achievements** (Gold)

Beinhaltete Achievements, welche an die Herausforderungen gekoppelt sind, die sich hauptsächlich mit der Abschlussphase einer Bachelorarbeit beschäftigen.

- **App-Achievements** (Blau)

Beinhaltete Achievements, welche an keinen offensichtlichen Herausforderungen gekoppelt sind, sondern nur durch Benutzung der Applikation erhalten werden können. Die hierbei von den klassischen Abstufung der Wertigkeit (Bronze, Silber, Gold) gewählte Farbe (Blau) soll die semantische Differenz zwischen den Herausforderungen und den, durch die Benutzung der App ausgelösten, Achievements verdeutlichen.

Der Inhalt wird durch unterschiedliche, den Achievements entsprechenden, Texten dargestellt. Hierbei geht es darum, die mit den Herausforderungen gekoppelten Achievements durch das Verfassen passender Texte eine Ausstrahlung zu verleihen, welche dem Benutzer, als Reaktion auf die Erfüllung eines Achievements, als Titel angezeigt wird. Der Inhalt der Texte ist nicht festgelegt, sollte jedoch so gewählt werden, dass eine positive oder ironische Stimmung gegenüber dem Benutzer vermittelt wird.

Hinweis: Eine für den Prototypen gewählte Übersicht der Achievements ist im Anhang zu finden (siehe Angang TODO).

5. Architektur und prototypische Implementierungsdetails

Das Ziel dieses Kapitels ist es, das Verständnis für die architektspezifischen Zusammenhänge sowie die jeweiligen softwaretechnischen Abläufe zu vermitteln.

Auf dieser Basis werden die einzelnen Architektur-Komponenten, deren Beziehungen zueinander sowie die Funktionsweise des Gesamtsystems im folgenden Verlauf erläutert und in Kapitel Die detaillierte Beschreibung der einzelnen Komponenten folgt in Kapitel 5.2 an geeigneter Stelle durch beispielhafte Abläufe von Nutzeraktionen beschrieben.

5.1. Architekturübersicht

Aufgrund der Vorzüge einer konsequenten zentralen Verwaltung des Zustands der Applikation, welche in diesem Rahmen auch mit einer komfortablen Wartbarkeit und den strikt getrennten und auf diese Weise auch eindeutig nachverfolgbaren Datenflüssen und Operationen einhergeht, wurde eine Architektur konstruiert, die sich an dem Redux-Architekturmuster orientiert. Die folgende Abbildung 5.1 soll in vereinfachter Form verdeutlichen, wie die verschiedenen Komponenten der gewählten Architektur hinsichtlich des Aufbaus und des Datenflusses zueinander in Beziehung stehen.

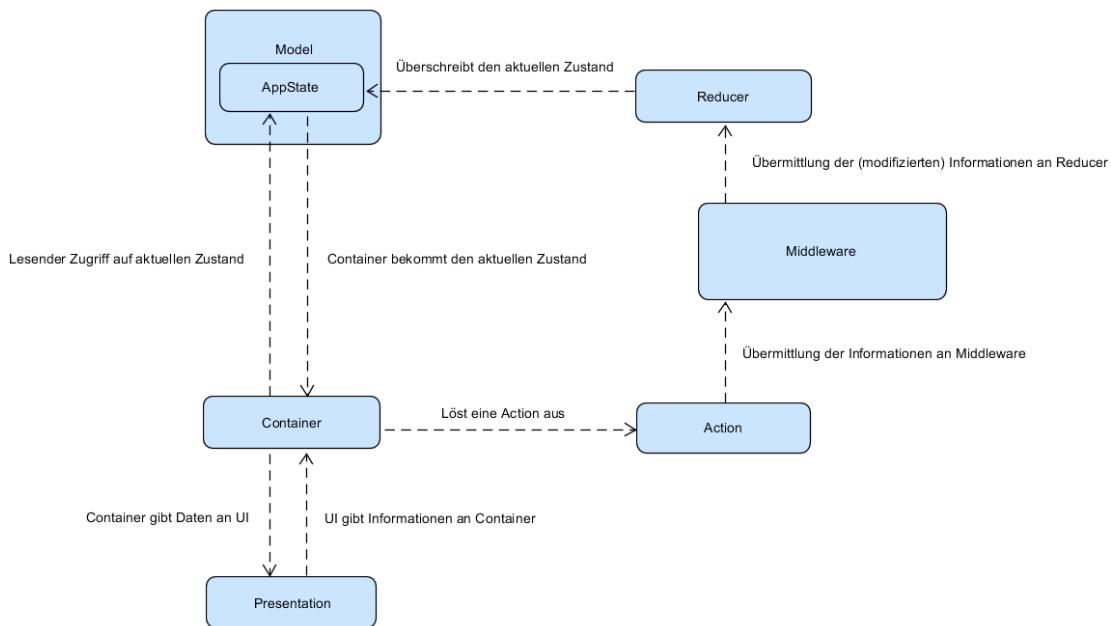


Abbildung 5.1.: Architekturübersicht

Model

Die Model-Komponenten stellen den aktuellen internen Zustand des Systems dar und enthalten somit in erster Linie die hierfür verwendeten Objekt-Klassen.

Die für die gewählte Architektur durch ihre Wichtigkeit hervorzuhebende Klasse AppState, ist die zentrale Verwaltungsklasse für den gespeicherten Status der Applikation und trägt alle relevanten Informationen in sich, die den aktuellen Zustand des Systems abbilden. Die Klasse AppState selbst dient also lediglich als zentrales Element zur Datenhaltung in der gewählten Architektur.

Presentation

Die Presentation-Komponenten sind alleinig für die Visualisierung der gegebenen Informationen sowie für die Entgegennahme der Benutzereingaben zuständig.

Container

Die Container-Komponenten lesen den aktuellen Zustand der Applikation aus dem AppState ein und bereiten diesen vor der Übergabe an die Presentation-Komponenten vor. Weiterhin lösen die Container-Komponenten die Actions aus, welche auf den von dem Benutzer getätigten Interaktionen basieren.

Action

Eine Action-Komponente beinhaltet Informationen über die jeweilige auszuführende Operation und dient somit als Identifikationsmittel und Informationspaket, die von der Middleware und/oder den Reducer-Komponenten aufgegriffen und verarbeitet werden.

Middleware

Die Middleware verarbeitet und/oder manipuliert die, in den Actions übermittelten, Informationen, bevor diese zu den Reducer-Komponenten gelangen. Weiterhin können auch in der Middleware selbst Actions ausgelöst werden, wie im späteren Verlauf in Kapitel 5.7 nachzuvollziehen ist.

Reducer

Die Reducer-Komponenten nehmen die eigentliche Änderung am Zustand vor, indem sie einen völlig neuen Zustand generieren und den vorherigen Zustand überschreiben. Durch die, über die Actions erhaltene, Informationen weiß die Reducer-Komponente, welche Aktion der Benutzer ausgeführt hat und folglich auch, welche Information sich an dem aktuell existierenden Zustand der Applikation ändern soll. In diesen Komponenten wird also der aktuelle Zustand in einen neuen Zustand transformiert.

5.2. Detaillierte Beschreibung der Model Komponenten

Die nun folgenden Unterkapitel beschreiben die Klassen, die Teil der Model Komponente sind und teilweise wichtige Datenfelder und Operationen zur Verwaltung des Redux-Stores enthalten.

Hinweis: Alle verwendeten Klassen deren Objekt-Instanzen in dem Redux-Store liegen, sind mit dem Schlüsselwort `@immutable` versehen und können zur Laufzeit nicht verändert werden. Dies ist hinsichtlich der Redux-Architektur vorgesehen, da der Redux-Store und dessen Inhalt nicht verändert, sondern überschrieben wird. Weiterhin liegen die Attribute innerhalb der Objekt-Klassen im Rahmen des flach ausgerichteten Key-Value-Stores des AppStates im Allgemeinen nicht als Objektreferenzen, sondern als IDs in textueller Form vor.

5.2.1. AppState-Klasse

Die Klasse AppState stellt den aktuellen Status der Applikation dar und trägt somit eine Vielzahl an Informationen in sich. Diese Informationen werden im Rahmen der an Redux orientierten Architektur durch das Einsetzen von Key-Value Stores strukturiert und verwaltet.

Jede für den Zustand der Applikation relevante Information ist auf diese Weise an einer zentralen Stelle im System gespeichert (siehe Listing 5.1). Durch einen lesenden Zugriff können die Oberflächenkomponenten alle benötigten aktuellen Informationen ermitteln und abbilden. Weiterhin kann der Status der Applikation durch das Ausführen von Aktionen des Benutzers, was das Auslösen der sogenannten Actions mit sich trägt, verändert werden.

```
1 class AppState{  
2  
3     final Map<String, Milestone> currentMilestones;  
4     final Map<String, Map<String, Achievement>> achievedAchievements;  
5     final Map<String, Property> properties;  
6     final Map<String, Challenge> challenges;  
7     final Map<String, Content> informationToolContent;  
8  
9     final Milestone selectedMilestone;  
10    final DateTime begin;  
11    final DateTime end;  
12  
13    ...  
14 }
```

Listing 5.1: Datenfelder der AppState-Klasse

Aktueller Zustand der Meilensteine

Die aktuellen Meilensteine werden durch das Datenfeld `currentMilestones` abgebildet. Diese Datenstruktur enthält alle vom Benutzer angelegten Meilensteine und deren jeweilige Unteraufgaben und wird als SplayTreeMap initialisiert. Die selbst-sortierende Eigenschaft dieser Datenstruktur bietet einige Vorteile im Rahmen der Handhabung von Meilensteinen in der Implementierung, da deren einzigartige IDs auf dem angegebenen Meilensteindatum basieren.

Aktueller Zustand der Achievements

Die von dem Benutzer der Applikation zu erreichenden Achievements werden im Datenfeld achievements gespeichert. Dieses Datenfeld weißt einige Besonderheiten vor, auf die ich im Folgenden hinweisen möchte:

Hinweis: Es handelt sich bei dieser Datenstruktur im Detail um eine HashMap, welche weitere HashMaps beinhaltet.

Diese HashMap wird mit genau vier verschiedenen konstanten Einträgen initialisiert.

- **ALL_ACHIEVEMENTS**

Enthält alle vorhandenen Achievements.

- **ACHIEVED**

Enthält alle Achievements, die von dem Benutzer gesammelt wurden.

- **RECOGNIZED**

Enthält alle gesammelten Achievements, die von dem Benutzer betrachtet wurden.

- **NOT_RECOGNIZED**

Enthält alle gesammelten Achievements, die von dem Benutzer noch nicht betrachtet wurden.

Diese Unterscheidung ist durch die Idee begründet, die unterschiedlichen Zustände der Achievements bei der Handhabung der Benutzerinteraktion zu unterschiedlicher Zeit zu berücksichtigen. Beispielsweise sollen neu erworbene Achievements, welche vom Benutzer noch nicht eingesehen worden sind, bis zu dem Moment, in dem sie betrachtet wurden, gesondert behandelt werden.

Aktueller Zustand der Properties

Die zu verwaltenden Eigenschaften zur Erfüllung der Achievements werden in dem Datenfeld properties gespeichert. Dies bietet den Vorteil, dass die Properties, welche von unterschiedlichen Achievements geteilt werden können, an einer zentralen Stelle im State verwaltet werden können.

Aktueller Zustand der Herausforderungen

Die für den Benutzer zu erledigenden Herausforderungen werden in dem Datenfeld challenges verwaltet. Über die bereits genannten Verwaltungsstrukturen hinaus beinhaltet die Klasse weiterhin folgende einzelne Informationen:

Aktueller Zustand des Bachelorarbeit-Guide Inhaltes

Der aus einem Git Repository ausgelesene und durch Content-Objekte strukturierte Inhalt des Bachelorarbeit-Guides wird in dem Datenfeld informationToolContent verwaltet.

Weitere Datenfelder

Der aktuell, im Rahmen der Bedienung der Applikation von dem Nutzer ausgewählte, Meilenstein wird im Datenfeld selectedMilestone gespeichert.

Die beiden Felder begin und end, bezeichnen den durch den Benutzer angegebenen Bearbeitungszeitraum der Bachelorarbeit.

5.2.2. Milestone-Klasse

Ein Meilenstein besitzt folgende Attribute und Methoden:

String id

Jeder Meilenstein besitzt eine ID, die immer auf dem jeweils festgelegten Datum des Meilensteins basiert.

String title

Der eigentliche Titel beschreibt den, von dem Benutzer gegebenen, Namen des Meilensteins und dient als Identifikations- und Anzeigemittel für den Benutzer.

String description

In dem Datenfeld description befindet sich die von dem Benutzer eingetragene Beschreibung des Meilensteins.

DateTime date

Jeder Meilenstein muss ein Ablaufdatum enthalten, welches den Tag darstellt, an dem der Meilenstein eingetragen ist.

Map<String, Task> tasks

Meilensteine besitzen, von dem Benutzer hinzugefügte, Unteraufgaben, die in dem Datenfeld tasks liegen.

Map<String, Task> getCompletedTasks(), getNotCompletedTasks()

Durch die Methoden getCompletedTasks() und getNotCompletedTasks() ist die Möglichkeit gegeben, abgeschlossene und noch offene Unteraufgaben der Meilensteine herauszufiltern.

MilestoneState getMilestoneState()

Durch die Methode getMilestoneState() bietet sich die Möglichkeit, den aktuellen Status des Meilensteins zu bestimmen, welcher auf abgeschlossene Unteraufgaben, noch offene Unteraufgaben sowie der Relation zwischen dem aktuellen Zeitpunkt und dem definierten Ablaufzeitpunkt basiert. Für diesen Zweck wird weiterhin die Enumeration MilestoneState verwendet, die die vier Zustände Empty, AllTasksCompleted, SomeTasksNotCompleted und outOfDateAndSomeTasksNotCompleted darstellt.

5.2.3. Task-Klasse

Ein Task besitzt folgende Attribute:

String id

Jeder Task besitzt eine ID als Identifikationsmöglichkeit.

String title

Der Titel eines Tasks stellt die, von dem Nutzer erstellte, inhaltlich zu lösende Aufgabe dar.

TaskState taskState

Ein Task kann die Zustände **Completed** und **NotCompleted** annehmen, welche durch eine Enumeration dargestellt werden. Eine Enumeration wurde einem booleschen Wert vorgezogen, da weitere Zustände, wie zum Beispiel das Favorisieren von Unteraufgaben, bei Bedarf hinzugefügt werden könnten.

5.2.4. Challenge-Klasse

Die von dem Benutzer zu absolvierenden Herausforderungen enthalten folgende Attribute:

String title

Jede Herausforderung besitzt einen Titel, welche den Inhalt der zu absolvierenden Aufgabe darstellt. Dies ist also der Inhalt, welcher der Benutzer in der Applikation angezeigt bekommt.

ChallengeType type

Der Typ einer Challenge lässt sich durch eine definierte Enumeration ausdrücken. Vorgesehen ist dies, um den jeweiligen Herausforderungen einen Kontext zu verleihen. Als Beispiel kann hier eine Aufteilung in verschiedene Phasen der Bachelorarbeit erfolgen.

bool completed

Ob eine Herausforderung abgeschlossen wurde oder noch offen ist, wird über das Feld **completed** angegeben.

5.2.5. Achievement-Klasse

Ein Achievement besitzt folgende Attribute:

String title

Der Titel des Achievements ist der Teil, der dem Nutzer als Identifikationsmittel des jeweiligen Achievements angezeigt wird.

bool completed

Die Information, ob ein Achievement erreicht wurde, wird durch den Wert des Attributs **completed** dargestellt.

List<String> properties

Weiterhin beinhaltet ein Achievement bestimmte Eigenschaften, die erfüllt werden müssen, damit das Achievement als erreicht bezeichnet werden kann. Ein Achievement wird also nur erreicht, wenn alle seine Properties als erfüllt gelten.

AchievementType type

Der Typ, welcher durch eine Enumeration verdeutlicht wird, stellt die Art des Achievements dar.

Hinweis: Bei der nicht ganz einfachen Implementierung von Achievements wurde sich nach ausgiebiger Recherche an einem im Internet geteilten Beitrag (vgl. [21]) orientiert und die dort beschriebene Umsetzung für die eigene Redux-basierte Variante adaptiert.

5.2.6. Property-Klasse

Die Klasse Property enthält folgende Attribute und Funktionen:

String name

Der Name einer Property-Instanz stellt die Bezeichnung dar, welche dem Nutzer als Identifikationsmöglichkeit dient und im Rahmen der Applikation als Instruktion angezeigt werden kann, weshalb diese Bezeichnung wohlüberlegt gewählt werden sollte.

int initialValue

Der Initialwert beschreibt, wie der Name bereits impliziert, den Startwert, der bei Initialisieren des Objektes angenommen wird.

int currentValue

Ein aktueller Wert stellt den aktuellen Fortschritt der jeweiligen Eigenschaft, welche es zu erfüllen gilt, dar.

int activationValue

Das Datenfeld activationValue stellt den Zielwert dar, den es zu erfüllen gilt.

String activationRule

Die genannten Felder initialValue, currentValue und activationValue werden durch eine definierte Regel in Beziehung zueinander gesetzt. Diese Regel kann verschieden ausgeprägt sein, um dem Ersteller von Achievements sowie deren Kriterien zur Erfüllung eine gewisse Freiheit zu ermöglichen. Im Rahmen des Projektes wurden drei verschiedene Regeln in Form von konstanten Variablen des Datentyps String aufgestellt, welche die Regeln “<“, “>“ und “==“ enthalten.

bool isActive()

Diese Regeln werden als Werkzeug benutzt, um die Semantik der zu erfüllenden Eigenschaften auf die logische Ebene zu übertragen. Beispielsweise kann somit in der Methode isActive() überprüft werden, ob der Benutzer bereits drei Meilensteine erstellt hat, also die jeweilige Eigenschaft erfüllt wurde.

5.2.7. Content-Klasse

Eine Objektinstanz der Content-Klasse stellt den Inhalt der “Guide“-Funktion dar und enthält folgende Attribute:

String id

Die ID eines Content-Objekts dient als Zuweisungsmittel von weiteren Content-Objekten als Unterabschnitten. Die verwendeten IDs werden in der ausgelesenen JSON-Datei, welche in dem Git-Repository liegt, zugewiesen

String title, description

Der Titel des Content-Objekts ist gleichzusetzen mit einer Überschrift für den jeweiligen Inhalt, der durch die Beschreibung ausgedrückt wird.

List<String> subsections

Das Datenfeld subsections enthält Referenzen durch in textueller Form zugewiesenen IDs weiterer Content-Objekte, welche in Beziehung mit dem jeweils übergeordneten Content-Objekts als Unterabschnitte zu bezeichnen sind.

String type

Der Typ enthält Informationen darüber, welche Art von Darstellung innerhalb des “Guide“-Tools für die Benutzeroberfläche verwendet werden soll. Das bietet den Vorteil, dass bei dem Erstellen der JSON-Datei für den Inhalt dieser Funktion keine Kenntnisse über die softwareinterne Lösung vorhanden sein muss, sondern der Ersteller nur vordefinierte Begriffe eintragen muss, die dann softwareintern den jeweiligen Bildschirminhalt generieren.

Hinweis: Im Rahmen der Prototypentwicklung sind zwei verschiedene Arten (type) der Darstellung von Content-Objekten vorgesehen:

1. *Die erste Möglichkeit ist das Erzeugen eines Buttons (“ButtonListWidget“), welcher den Titel des Content-Objektes anzeigt und bei Antippen auf eine nächste Seite führt und dort die jeweiligen subsections anzeigt.*
2. *Die zweite Möglichkeit ist das Generieren eines ausklappbaren Elements (“ExtensionPanelWidget“), welches den Titel des Content-Objektes anzeigt und dessen Beschreibung bei Ausklappen sichtbar wird.*

5.2.8. AppContentLoader-Klasse

Die Klasse ContentLoader stellt die Schnittstelle zum gewählten Git-Repository dar und trägt in diesem Umfang die Aufgabe, die in dem Repository-Ordner angelegten JSON-Dateien durch HTTP-Requests einzulesen und dem System zur weiteren Verarbeitung zur Verfügung zu stellen. Dies wird umgesetzt, indem diese eingelesenen Dateien in lokaler Form auf dem Endgerät gespeichert werden.

Die Klasse besitzt lediglich ein Datenfeld (`String httpRequest`), welches die auszuführende HTTP-Anfrage beinhaltet.

5.2.9. JSONAppContentFile-Klasse

Eine Instanz dieser Klasse, die durch eine Objektinstanz der Klasse AppContentLoader erzeugt wird, stellt eine aus dem Repository eingelesene JSON-Datei dar und enthält den Namen der im Repository vorliegenden JSON-Datei, welche auch weiterhin als Dateiname in dem lokalen Speicher des Endgeräts verwendet.

Die durch den ContentBuilder ermittelte jeweilige Download-Adresse der verschiedenen im Repository liegenden Dateien, dient zum Herunterladen des jeweiligen Inhaltes der jeweiligen JSON-Datei, was wiederum durch die Methode `Future<String> getJsonContent()`, im Detail durch ein HTTP-Request umgesetzt wird.

5.2.10. InformationToolContentBuilder-Klasse

Die Hilfsklasse InformationToolContentBuilder dient als Werkzeug zur Generierung des Inhaltes der Software-Sektion “Bachelorarbeit Guide“ der Applikation. Die Content-Objekte werden durch die jeweilige JSON-Datei, welche zuvor aus dem Git-Repository ausgelesen und gespeichert wurde generiert. Jedes Content-Objekt kann mehrere Content-Objekte in Form der jeweiligen IDs referenzieren, welche in diesem Rahmen als subsections bezeichnet werden.

5.2.11. Achievement-Notification Funktionen

Für die Implementierung der Notification wurde ein externes Plugin genutzt (siehe [14]), da die grundsätzliche Möglichkeit der Handhabung von Notifications durch das Framework Flutter nicht gegeben ist. Durch die Verwendung des Plugins sind jedoch einige zielplattformabhängige Konfigurationen nötig, welche sich unter [14, Android Integration] und [14, iOS Integration] nachvollziehen lassen.

Bei der Umsetzung der Notifications wurde sich an den dort gegebenen Beispielen orientiert, wobei deren Funktionsweisen für die eigene Umsetzung adaptiert wurden.

Die Verwendung der Achievement-Notifications wird im Rahmen der prototypischen Implementierung durch spezielle Top-Level-Funktionen gelöst, welche bei Aufrufen die jeweiligen Achievement-Actions auslösen. Die dadurch hervorgerufene Aktualisierung der zu den Achievements zugehörigen Bedingungen findet in der `updateProperties Reducer-Funktion` statt. Hier werden die Nutzeraktionen an zentraler Stelle im System abgefragt und die Änderungen an den aktuellen, im Zustand gespeicherten, Bedingungen durchgeführt. Somit ist dies die zentral zu erweiternde Stelle im Code, wenn neue Achievements Einzug in das System erhalten.

Weitere Informationen zum Ablauf und der Funktionsweise von Notifications sind in Kapitel 5.7 zu finden.

5.3. Actions

Im Rahmen der auf der Anforderungsanalyse basierenden Konzeption der Softwarelösung wurden verschiedene Benutzeraktionen identifiziert, welche sich größtenteils in den sogenannten Actions wiederfinden. Für jede dieser Actions existiert eine eigene Klasse, welche im folgenden Verlauf beispielhaft an einem für alle Actions repräsentativen Fall erläutert wird.

5.3.1. Beispielhafte Demonstration einer Meilensteinbearbeitung

Das gewählte Beispiel bezieht sich auf die Nutzeraktion des Änderns eines Meilensteins: Wie in Abbildung 5.2 zu sehen ist, enthält die Klasse EditMilestoneAction vier verschiedene Datenfelder. Der zu ändernde Meilenstein (`milestone`) sowie die jeweiligen neuen Werte `newTitle`, `newDate` und `newDescription`, welche die zu dem vorhandenen Meilenstein zugehörigen Werte überschreiben sollen.

Es wird deutlich, dass eine solche Klasse außer dem Beinhalten der spezifischen Payload-Daten keine tiefe Funktionalität bieten soll.

Die weitere Verarbeitung der Daten wird im Kapitel 5.4 verdeutlicht.

```
1 class EditMilestoneAction {  
2     Milestone milestone;  
3     String newTitle;  
4     DateTime newDate;  
5     String newDescription;  
6  
7     EditMilestoneAction(this.milestone, this.newTitle, this.newDate,  
8         this.newDescription);  
9 }
```

Listing 5.2: Meilenstein Action Beispiel

5.4. Reducer

Die verschiedenen Reducer-Funktionen tragen die eigentliche Aufgabe der Datenmanipulation und Verwaltung. In diesen Komponenten passieren die entscheidenden Änderungen am aktuellen Zustand der Applikation.

5.4.1. App-Reducer

Der App-Reducer ist die Top-Funktion der Reducer, welche nach jeder durchgeführten Action einen neuen Zustand der Applikation generiert. Jedes Datenfeld des aktuellen Zustands wird in diesem Rahmen durch die Delegation an untergeordnete Reducer-Funktionen neu generiert. Dabei wird immer der aktuelle Zustand des Systems sowie die in der Action-Klasse enthaltenen Informationen übergeben.

Im folgenden Verlauf wird dieser Ablauf an einem repräsentativen Beispiel verdeutlicht.

5.4.2. Beispielhafte Demonstration eines Meilenstein-Reducers

Innerhalb der von dem AppReducer aufgerufenen updateMilestones-Funktion (siehe Abbildung 5.3) wird überprüft, um welche Action es sich handelt. Sollte es sich um keine der angegebenen Actions handeln, so wird der aktuell existierende Zustand unverändert zurückgegeben (vgl. Zeile 16). Sollte die Action jedoch übereinstimmen, so wird anschließend die jeweils definierte Folge von Befehlen ausgeführt und der neue Zustand der jeweiligen Daten zurückgegeben (vgl. Zeile 12).

```
1 Map<String, Milestone> updateMilestones (Map<String, Milestone> current, action){  
2     Map<String, Milestone> ret = new SplayTreeMap<String, Milestone>();  
3  
4     ...  
5  
6     if(action is EditMilestoneAction) {  
7  
8         Milestone editedMilestone = new Milestone(action.newTitle, action.newDate,  
9                 action.newDate.toIso8601String(), action.milestone.tasks,  
10                action.newDescription);  
11            current.remove(action.milestone.id);  
12            current[editedMilestone.id] = editedMilestone;  
13  
14        return current;  
15    }  
16    ...  
17    return current;
```

Listing 5.3: Meilenstein Reducer Beispiel

5.5. Container

Die Container-Komponenten dienen in dieser an Redux orientierten Architektur als Schnittstelle zwischen der Benutzeroberfläche (Presentation) und dem internen Zustand der Applikation (Model). Sie stehen somit einerseits in der direkten lesenden Beziehung zu dem Store, der den aktuellen Status der Applikation beinhaltet und andererseits in Beziehung zu den Reducer-Komponenten, welche diesen Status durch das Auslösen von Actions schreibend aktualisieren.

Im folgenden Verlauf wird dieser Ablauf an einem repräsentativen Beispiel verdeutlicht.

5.5.1. Beispielhafte Demonstration einer Meilensteinbearbeitung

Die Converter-Funktion (siehe Abbildung 5.4, vgl. Zeile 8) des StoreConnectors konvertiert den aktuellen Store in ein ViewModel, welches die für die Presentation-Komponente relevanten Daten in der Builder-Funktion (vgl. Zeile 11) bereitstellen kann.

```
1 class EditMilestone extends StatelessWidget {
2     final Milestone milestone;
3     EditMilestone(this.milestone);
4
5     @override
6     Widget build(BuildContext context) {
7         return new StoreConnector(
8             converter: (Store<AppState> store) {
9                 return _ViewModel.fromStore(store, this.milestone);
10            },
11            builder: (BuildContext context, _ViewModel vm) {
12                return AddEditMilestoneScreen(vm.editMilestone, vm.currentMilestones,
13                    vm.initialDate, true, milestone);
14            },
15        );
16    }
}
```

Listing 5.4: Container-Komponente einer Meilensteinänderung

In der ViewModel-Klasse werden einerseits die Informationen aus dem Store geladen (vgl. Zeile 20) und weiterhin auch die getätigten Actions gesendet (vgl. Zeile 17). Letzteres funktioniert durch das Übermitteln von Callback Funktionen an die Presentation-Komponente (siehe Abbildung 5.4, vgl. Zeile 13).

```
1 class _ViewModel {
2     final Milestone currentMilestone;
3     final Function(String, DateTime, String) editMilestone;
4     final List<Milestone> currentMilestones;
5     List<String> currentMilestonesDates;
6     DateTime initialDate;
7
8     _ViewModel(
9         {this.currentMilestone, this.editMilestone, this.currentMilestones,
10          this.currentMilestonesDates}) {
11         // ... Initialisieren von currentMilestonesDates und initialDate
12     }
13
14
15     factory _ViewModel.fromStore(Store<AppState> store, currentMilestone) {
16         return _ViewModel(
17             editMilestone: (String title, DateTime date, String description) =>
18                 store.dispatch(new EditMilestoneAction(
19                     currentMilestone, title, date, description)),
20             currentMilestones: store.state.currentMilestones.values.toList(),
21         );
22     }
23 }
```

Listing 5.5: ViewModel von Container-Komponente einer Meilensteinänderung

5.6. Presentation

Die Presentation-Komponenten stellen die Schnittstelle zwischen System und Benutzer dar. Die Aufgabe dieser Komponenten ist, die über die Nutzereingaben erhaltenen Informationen an die jeweiligen Container-Komponenten zur Weiterverarbeitung zu vermitteln und den aktuellen Systemzustand für den Anwender geeignet darzustellen. Eine Presentation-Komponente erhält alle nötigen Informationen von der jeweils übergeordneten Container-Komponente und soll nicht direkt mit dem Store interagieren. Ziel ist es, die Darstellung von der direkten Systeminteraktion strikt zu trennen, um Austauschbarkeit und Änderungen an der Benutzeroberfläche zu erleichtern.

Im folgenden Verlauf wird dieser Ablauf an einem repräsentativen Beispiel verdeutlicht.

5.6.1. Beispielhafte Demonstration einer Meilensteinbearbeitung

Wie in Abbildung 5.6 zu sehen ist, wird die von der Container-Komponente übermittelte Callback-Funktion addEditMilestone bei Bestätigen der vom Nutzer eingegebenen Inhalte ausgeführt. Darunter zählen in diesem Fall die der Titel, das Datum und eine optionale Beschreibung des bearbeiteten Meilensteins.

```
1 void submit(context) {  
2  
3     if (_formKey.currentState.validate()) {  
4         addEditMilestone(_titleInputField.currentState.value, chosenDate,  
5                           _optionalDescriptionInputField.currentState.value);  
6  
7     ...  
8 }  
9 }  
10  
11 }
```

Listing 5.6: Bestätigung einer Meilensteinänderung Methode

5.7. Middleware

Im Folgenden wird die eingesetzte Middleware beschrieben, welche im Rahmen der prototypischen Entwicklung der Applikation für die Verwaltung von Push-Notifications und das lokale Speichern und Laden des aktuellen Zustands bei Schließen und Öffnen der Applikation zuständig ist.

5.7.1. Beispielhafte Demonstration der Notification Middleware

Wie in Abbildung 5.7 zu sehen ist, wird auch hier wie in den Reducer-Komponenten innerhalb dieser Middleware eine Abfrage zu der Art der ausgelösten Actions getätigt (vgl. Zeile 6). Sollte die Bedingung, dass es sich um eine SendAchievementNotificationAction handelt, nicht bestätigen, so wird die eingegangene Action sofort weiter an die Reducer-Komponenten geleitet(vgl. Zeile 21). Falls sich die Bedingung jedoch als wahr herausstellt, so wird der darauffolgende Ablauf von Befehlen ausgeführt, welcher sich hier auf das Auslösen einer Notification bezieht.

Eine Besonderheit innerhalb dieser Komponente ist die Verwendung eines Completers, welcher als Teil der Action Payload übermittelt wurde.

Hinweis: Ein Completer ist eine von Dart gegebene Klasse, die es in diesem Fall ermöglicht, eine zukünftig ausgeführte Funktion so zu behandeln, als wären der Rückgabewert bereits bekannt. Weitere Informationen hierzu, sind in der Dart Dokumentation zu finden (vgl. [10]).

```
1 void notificationMiddleware(
2     Store<AppState> store,
3     dynamic action,
4     NextDispatcher next,
5     ) {
6     if (action is SendAchievementNotificationAction) {
7         new Future.delayed(new Duration(seconds: 1), () {}).then( (result) {
8             if (NEUE ACHIEVEMENTS VORHANDEN) {
9                 ...
10                showAchievementNotification(
11                    //... spezifischer Inhalt eines Achievements
12                    );
13                );
14                ...
15                action.completer.complete();
16                store.dispatch(new ClearAchievedAction());
17            },
18        },
19    );
20 }
21 next(action);
22 }
```

Listing 5.7: Notification Middleware

In diesem Rahmen wird immer, wenn eine Actions in der Applikation ausgeführt wird, das Senden einer Notification durch das Aufrufen der Top-Level-Funktion checkAchievementNotification() „vorbereitet“ (siehe Abbildung 5.8, vgl. Zeile 9 - 11).

Um zu gewährleisten, dass neue Achievements registriert werden, wird vor Ausführung der SendAchievementNotificationAction in Abbildung 5.8(vgl. Zeile 3 - 7) die Methode CheckForAchieveAction() ausgelöst, die für das Aktualisieren der aktuellen Achievements sorgt.

Sollten folglich die Bedingungen für das Ausführen dieser Notification zutreffen, so wird wie in Abbildung 5.7 zu sehen ist, die „versprochene“ Action ausgeführt (vgl. Zeile 25) und in diesem Fall die Notification angezeigt.

```
1 Future<dynamic> checkAchievementNotification(context) {  
2     StoreProvider.of<AppState>(context).dispatch(new CheckForAchieveAction(  
3         StoreProvider  
4             .of<AppState>(context)  
5             .state  
6             .achievedAchievements[ALL_ACHIEVEMENTS]));  
7  
8     final action = new SendAchievementNotificationAction();  
9     StoreProvider.of<AppState>(context).dispatch(action);  
10    return action.completer.future;  
11}  
12 }
```

Listing 5.8: checkAchievementNotification Methode

6. Evaluation

In den folgenden Kapiteln werden die unternommenen Evaluationsschritte des Projekts beschrieben und ausgewertet. Hierbei werden die entwickelten Softwaretests, die durchgeführte Nutzerevaluation sowie die abschließende Bewertung des verwendeten Frameworks Flutter aufgeführt.

6.1. Ergebnisse der Softwaretests

Im Rahmen der Softwaretests wird speziell das Testen der Reducer-Funktionen im Kontext der Redux-Architektur als essenziell betrachtet, da diese als zentrale ausführende Einheit der jeweiligen Nutzeraktionen identifiziert werden und somit auch auf die jeweiligen Anforderungen abzubilden sind.

Es wurden verschiedene allgemeine Interaktionen mit dem System identifiziert, welche durch die folgenden Testfälle im Rahmen der Reducer-Tests abgedeckt werden:

Testgruppe	Testfall	Status	Anforderungen
Milestone Reducer	Add Milestone	Erfolgreich	[Req1.1.1]
	Remove Milestone	Erfolgreich	[Req1.1.2]
	Edit Milestone	Erfolgreich	[Req1.1.3]
	Add Task to Milestone	Erfolgreich	[Req1.2.1]
	Remove Task from Milestone	Erfolgreich	[Req1.2.2]
	Edit Task from Milestone	Erfolgreich	[Req1.2.3]
Challenge Reducers	Check Toggle Challenge State	Erfolgreich	[Req3.2]
Achievement Reducers	Check Challenge Achievement	Erfolgreich	[Req3.2]
Initialization Test	AppContentLoader/Generator GitRequest Test	Erfolgreich	[Req2.4]

Tabelle 6.1.: Übersicht der Reducer-Tests

6.2. Ergebnisse der Nutzerevaluation

Insgesamt wurden jeweils 6 Personen einem Test unterzogen, der als zentralen Bestandteil die Nutzung der entwickelten Applikation beinhaltet.

Unter den genannten 6 Personen befinden sich 2 Professoren und 4 Studierende des Studien-gangs Informatik/Softwareentwicklung der Fachhochschule Lübeck.

Weiterhin befinden sich 3 der 4 Studierenden in der Phase während der Bachelorarbeit und 1 Studierender am Ende des 5. Semesters vor der Bearbeitung der Bachelorarbeit.

Der Test wurde unter Moderation mit verschiedenen vordefinierten Aufgaben durchgeführt (siehe TODO), wobei hier aufgrund der vergleichbar zu behandelnden getesteten Nutzerer-fahrung sowie der abschließenden Beurteilung der Tests nicht zwischen Professoren oder Studierenden unterschieden, sondern der jeweils gleiche Testablauf gewählt wurde.

Bei der Durchführung der Tests gab es dennoch einige wichtige Unterschiede.

Zum einen unterlagen die Test mit den Professoren einer ausführlicher gestalteten Vorstellung und Beschreibung der Features, da diese durch die Erfahrung der Professoren hinsichtlich ihres Nutzens als sehr gut beurteilbar angenommen wurden.

Weiterhin wurde bei den Testdurchführungen mit den Studierenden ein abschließender Fragebogen ausgehändigt, in dem die Testpersonen noch einmal die Möglichkeit hatten weitere Anmerkungen zu der Software festzuhalten.

Die eigentliche Testdurchführung wurde auf dem eigenen, dafür vorbereiteten, Smartphone durchgeführt, wobei das Bildschirmgeschehen aller Tests mithilfe einer Software aufgezeichnet wurde, jedoch ist eine der Dateien durch einen unbekannten Datenfehler nicht korrekt aufgezeichnet worden und somit nicht verwendbar.

Des Weiteren wurden nach vorher eingeholter Erlaubnis der Testpersonen ein Audiomitschnitt erzeugt. Dieser Audiomitschnitt wurde von beiden Professoren und insgesamt 3 von 4 Studierenden zugelassen. Diese Audiomitschnitte unterliegen jedoch aus nicht bekannten Gründen einer schlechten Aufnahmqualität. Es wurden jedoch unmittelbar nach jedem Interview Notizen angefertigt, welche im Anhang dieses Dokuments zu finden sind (siehe Anhang TODO).

Der verwendete Testablauf, welcher sich aus nacheinander gestellten Aufgaben zusammensetzt, ist im Anhang zu finden und umfasst zehn Aufgaben (siehe TODO ANHANG).

Die Aufgaben orientieren sich an den verschiedenen Softwarefunktionen der entwickelten Applikation und umfassen das Verwenden aller Kernfunktionen.

Die Reihenfolge der gestellten Aufgaben ist im Rahmen des Tests jedoch nicht fest definiert, sondern wurde je nach Situation frei gewählt.

6.2.1. Zusammenfassung der Testdurchführung mit den Studierenden

Zusammenfassend hat sich ergeben, dass alle Testpersonen die vorgegebenen Aufgaben fast ohne Schwierigkeiten lösen konnten und der Großteil der Aufgaben ohne Hilfestellung des Moderators erfüllt werden konnten.

Falls Unklarheiten bezüglich des Lösens der Aufgaben bei den Testpersonen aufgetreten sind, so konnten diese größtenteils selbstständig beseitigt werden.

Im folgenden Verlauf werden die Ergebnisse des Tests, nach dem Ablauf der jeweiligen Aufgaben kategorisiert, dargestellt. Weiterhin werden, falls vorhanden, die Kernaussagen der aufgabenspezifischen Kommentare und Wünsche zur Softwaregestaltung und den Features der Testpersonen beschrieben.

Aufgabe 1 - Anweisungen des Startbildschirms befolgen

Insgesamt haben 4 von 4 Testpersonen diese Aufgabe erfolgreich abgeschlossen. Alle Studierenden konnten den Anweisungen folgen und haben einen Bearbeitungszeitraum für ihre Bachelorarbeit eingetragen und bestätigt, woraufhin sie auf das Dashboard gelangt sind. Es gab keine negativen Anmerkungen zur Softwareumsetzung dieser Funktionalität.

Aufgabe 2 - Beschreibe die Elemente des Dashboards

Alle 4 Testpersonen konnten die auf dem Dashboard befindlichen Elemente, also deren Bedeutung und Nutzen korrekt deuten.

Aufgabe 3 - Informiere dich über die Anmeldung der Bachelorarbeit.

Es konnten 3 von 4 Testpersonen ohne Schwierigkeiten Informationen über die Anmeldung einer Bachelorarbeit durch die Nutzung der Applikation gewinnen.

Eine der Testpersonen hatte große Schwierigkeiten, die Sektion "Bachelorarbeit Guide" als solche zu identifizieren und musste durch Hilfestellung des Moderators geleitet werden. Ab diesem Punkt gelang es der Testperson jedoch problemlos, sich über das Thema zu informieren.

Aufgabe 4 - Erstelle zwei Meilensteine

Alle 4 von 4 Testpersonen konnten diese Aufgabe erfolgreich lösen, wobei jedoch 2 von 4 Personen fälschlicherweise als erste Reaktion auf den Dashboard-Zeitstrahl getippt haben, da dieser in die unmittelbare Verbindung mit den zu erstellenden Meilensteinen gebracht wurde.

Aufgabe 5 - Was ist passiert? Begib dich zu dem Achievement-Bildschirm.

Insgesamt haben 4 von 4 Testpersonen diese Frage beantworten können. Eine der Testpersonen hat das, durch eine Push-Benachrichtigung verdeutlichte, Achievement nicht als solches wahrgenommen und war verwirrt. Nach einer Erklärung hat diese Person verstanden was passiert ist und hat sich wie die anderen Personen erfolgreich zu dem Achievement-Bildschirm begeben.

Aufgabe 6 - Füge den Meilensteinen Unteraufgaben hinzu.

Alle 4 von 4 Testpersonen haben diese Aufgabe abschließen können, wobei 3 von 4 Personen Schwierigkeiten hatten, die gegebene Android-Tastatur nicht nur als Eingabewerkzeug, sondern auch das erforderliche Betätigen mithilfe des Enter-Buttons zu identifizieren. Dies führte dazu, dass die Testpersonen einen Bestätigungsbutton auf dem Bildschirm gesucht haben. Es wurde der Wunsch nach einem Feedback bei Hinzufügen von Aufgaben geäußert, um Kenntnisse über den erfolgreichen Abschluss des Vorgangs zu gewinnen.

Weiterhin führte die Darstellung der ein- und ausklappbaren Aufgabenliste vermehrt zu Irritationen, da offenbar das Gefühl vermittelt wurde, dass die eingetragenen Aufgaben verschwunden seien.

Aufgabe 7 - 10 Allgemeine Operationen ausführen

Aufgabe 7 bis Aufgabe 10 befassen sich mit weiteren allgemein in der Software zu tätigen Benutzerinteraktionen, welche aufgrund der gleich ausfallenden Ergebnisse im Folgenden zusammengefasst werden:

- Aufgabe 7 - Das Bearbeiten von Meilensteinen (Datum ändern)
- Aufgabe 8 - Die Navigation innerhalb der Software
- Aufgabe 9 - Das Abschließen von Meilenstein-Unteraufgaben
- Aufgabe 10 - Das Abschließen definierter Herausforderungen

Alle Testpersonen konnten diese Aufgabe ohne Schwierigkeiten abschließen.

6.2.2. Zusammenfassung der Testdurchführung mit den Professoren

Insgesamt wurden alle im Rahmen der Testdurchführung gestellten Aufgaben auch von den Professoren erfolgreich gelöst. Der Schwerpunkt der Tests lag hierbei jedoch auf der Betrachtung der einzelnen Features aus der Meta-Perspektive, welche sich durch die Erfahrung der Professoren auszeichnet und die Beurteilung der Softwareumsetzung hinsichtlich des Inhalts vorsieht.

Im folgenden Verlauf ist eine nach den Features der Software kategorisierte Zusammenfassung der Ergebnisse zu finden:

Kategorie 1 - Der Startbildschirm bei erstmaligem Ausführen der Applikation

Es wurden von einer der beiden Testpersonen verschiedene Anmerkungen zu dem Bildschirm geäußert, auf dem die Datumseingabe für den Bearbeitungszeitraum der Bachelorarbeit erfolgt. Zum einen wurde angemerkt, dass es sich bei dem Datum um das amerikanische Datumsformat (MM/DD/YYYY) handelt und dies zu Verwirrungen führt. Weiterhin wurde gewünscht, dass Informationskästchen mit Inhalten zur Bearbeitungszeit der Bachelorarbeit auf dem Bildschirm platziert werden könnten.

Kategorie 2 - Bachelorarbeit Guide

Es liegen keine konkreten Anmerkungen zu dieser Funktion vor.

Kategorie 3 - Meilensteinverwaltung

Grundsätzlich wurden alle, das Meilenstein-Tool betreffende, Aufgaben erfolgreich gelöst, jedoch kam es bei dem Bestätigen von Unteraufgaben aufgrund der nicht erkennbaren Bestätigungsmöglichkeit in beiden Testdurchführungen zu Schwierigkeiten.

Diese Darstellung dieser Option wird jedoch vielen Geräten in unterschiedlicher Weise visualisiert, weshalb es sich hierbei wohl um eine Frage der Gewohnheit handelt. Dies ist insofern wichtig, da alle Testdurchführungen auf einem für die Testpersonen fremden Gerät ausgeführt wurden.

Kategorie 4 - Achievements

Bei Verwendung der Achievement-Funktion wurde angemerkt, dass die zusammenhängenden Achievements und Herausforderungen gegebenenfalls nicht trennscharf genug gehandhabt werden. Es wird gewünscht, dass klassische Gamification-Badges Einzug in die Applikation erhalten.

Kategorie 5 - Herausforderungen

Die Herausforderungen wurden als äußerst sinnvoll bezeichnet, da sie bei richtiger Gestaltung die verschiedenen folgenden Checklisten-Eigenschaften erfüllen können:

- Man kann überprüfen, ob man alle relevanten Inhalte einer Bachelorarbeit bedacht hat.
- Man wird auf Dinge hingewiesen, die man sonst nicht rechtzeitig bedacht hätte.
- Herausforderungen können einen auffordernden Charakter haben.
- Gute Ergänzung zum individuellen Meilenstein-Tool durch stärkere inhaltliche Vorgabe.
- Orientierungsmöglichkeit für Studierende.

Weiterhin wurden die folgenden Wünsche für eine gute Umsetzung von Herausforderungen betont:

- Das selbständiges Erinnern an Herausforderungen könnte eine nützliche Funktion sein.
- Verbindung zu anderen Features der Applikation (Bachelorarbeit Guide) sollte gegeben sein, da die Herausforderungen nicht zu isoliert vorliegen sollten und das Gesamtpotential der Software genutzt werden sollte.
- Die korrekte Formulierung der Herausforderungen ist essenziell, da es nicht sinnvoll wäre, wenn man nur Achievements bekommt, wenn man etwas angefangen hat, sondern auch wenn man etwas fertigstellt.
- Eine Mechanik, um Selbstbetrug zu verhindern, wäre wünschenswert.

Kategorie 6 - Dashboard

Das Dashboard und der auf diesem befindliche Zeitstrahl wurde im Rahmen der Tests als sehr sinnvoll eingeschätzt, da die folgenden Aspekte bei den Studierenden gefördert werden könnten:

- Motiviert Studenten dazu, sich überhaupt einen Plan zu machen und sich folgende Fragen zu stellen:
 - Welche Meilensteine habe ich?
 - Wie lange brauche ich?
- Die einfache Möglichkeit der Übersicht und das Fördern folgender Fragen:
 - Wie weit bin ich gerade?
 - Bin ich früh oder zu spät dran?

Weiterhin wurde die Anmerkung gemacht, dass die Auswahl von Meilensteinen auf dem Zeitstrahl erleichtert werden könnte, wenn die Meilensteine über eine andere Visualisierung dargestellt werden, da diese zu klein sind und gerade bei mehreren Meilensteinen nicht auswählbar sein könnten.

Kategorie 7 - Allgemeine Anmerkungen

Eine der Testpersonen hat die Idee geäußert, einen initialen vorbildlichen Meilensteinplan zu erstellen, an dem sich die Anwender für ihre eigenen Zwecke orientieren können.

6.2.3. Ergebnis der Nutzerevaluation

Zusammenfassend lässt sich aus der Evaluation mit den Professoren und den Studierenden ein konkreter aktueller Stand ableiten, welcher als Orientierungspunkt für die weitere Entwicklung der Software dient.

Im Allgemeinen gilt der Software-Prototyp als noch nicht reif für den realen Einsatz, da er sowohl hinsichtlich der Usability, als auch der inhaltlichen Umsetzung im Rahmen der Tests als ausbaufähig und unfertig erscheint.

Die folgenden Punkte dienen als Konkretisierung der größten zu leistenden Änderungen oder Erweiterungen der Software:

- Das Zeitstrahl-Element auf dem Dashboard sollte durch eine geeigneter Visualisierung ersetzt oder zwecks besserer Bedienbarkeit bei mehreren eingetragenen Meilensteinen entsprechend angepasst werden.
- Achievements sollten stärker von den Herausforderungen differenziert sein und durch eine stärkere der Gamification entsprechenden Visualisierung, wie beispielsweise Badges statt Farben, verfügen.
- Herausforderungen sollten mit den jeweiligen Informationen des Bachelorarbeit Guide verbunden werden, da sie zu isoliert voneinander vorliegen und das mögliche Potential der Funktion nicht genutzt wird.

Des Weiteren sollen die von den Testpersonen angesprochenen Details im Umgang mit der Software berücksichtigt und eine Lösung geeignet umgesetzt werden, worauf eine weitere Testiteration durchgeführt werden soll.

6.3. Bewertung des Frameworks Flutter zur Entwicklung von mobilen Applikationen

Im Laufe der in dieser Arbeit durchgeführten Entwicklung einer Applikation mit dem Frameworks Flutter haben sich sowohl subjektive Erfahrungen als auch objektive Fakten ergeben, die im folgenden Verlauf unter der Fragestellung, inwiefern sich Flutter für die Entwicklung von mobilen Applikationen eignet, bewertet werden.

Die Wahl der Bewertungskriterien orientiert sich zum einen an den, im Rahmen einer Softwareentwicklung bezeichnenden, Eigenschaften eines Frameworks. Dazu gehören die Qualität der Dokumentation, die Komplexität des Erarbeitungsaufwandes, die Wahl der zu verwendenden Programmiersprache, die Handlichkeit der allgemeinen Anwendung, die Performance-Eigenschaften und der Support sowie die zugehörige Community-Aktivität des Frameworks.

Zum anderen werden die framework-spezifischen Merkmale mit in den Bewertungsumfang einbezogen, um die klaren, den Anwendungsfall betreffenden, Stärken des Frameworks abzubilden.

6.3.1. Einführung und Dokumentation

Das Framework Flutter weist eine ausführlich gestaltete und sehr gut strukturierte Online-Dokumentation vor, welche besonders auf die, für Einsteiger wichtigsten und interessantesten, Funktionen mit ausführlichen Beispielanwendungen und Schritt-für-Schritt Anleitungen eingeht. Besonders wird dies durch die Hilfestellungen der zielplattform-spezifischen **Get Started Anleitung**[20] verdeutlicht.

Hierbei werden ausführliche Anleitungen zur Einrichtung und Installation der benötigten Softwarekomponenten für die verbreiteten Betriebssysteme Windows, MacOS und Linux gegeben. Nach Abschluss der Installationsschritte wird der Flutter-Newbie im zweiten Teil des **Get Started** Programms mit dem Erstellen einer Beispielapplikation unter detaillierter Anleitung und einer abschließenden Tour durch das Flutter-Framework in die App-Entwicklung entlassen und die benötigten Grundlagen zur selbstständigen Weiterbildung durch die Dokumentation erlernt haben sollte.

6.3.2. Verwendete Programmiersprache

Mit **Dart**[12] erhält man bei der Entwicklung mit Flutter eine Google zugehörige, moderne, ebenfalls gut dokumentierte und für die mobile Entwicklung optimierte Programmiersprache, die eine objektorientierte Programmierung unterstützt.

6.3.3. Handhabung und Anwendung

Durch die Handhabung der Flutter-Widgets, welche in einer Baumstruktur organisiert sind, die die Grundbausteine einer jeder Benutzeroberfläche darstellen, sollten Entwickler im Allgemeinen einen sehr intuitiven Umgang erwarten, welcher weiterhin durch die Dokumentation unterstützt wird. Weiterhin gestaltet es sich aufgrund der einfach gehaltenen Grundstrukturen als sehr einfach, in kurzer Zeit eine lauffähige Minimalsoftware zu erstellen, was erneut für eine sehr gute Erlernbarkeit der Funktionsweise von Flutter spricht.

Flutter enthält von Haus aus eine große Vielfalt an Widgets, welche dem Entwickler große Freiheiten bei der Gestaltung der Software lassen und hinsichtlich der Plattformunabhängigkeit viel Arbeit abnehmen.

Es kann jedoch je nach Anwendungsfall abseits des reinen Oberflächendesigns schnell zum Erreichen dieser Grenze kommen, da viele Services, wie beispielsweise Push-Notifications, nicht von Flutter unterstützt werden. Im Rahmen dieser Arbeit wurde hierfür ein externes Plug-In verwendet, wobei zusätzliche zielpflicht-spezifische Konfigurationen zwingend notwendig waren.

Als nicht sehr komfortabel hat sich das Ausführen auf einem iOS Endgerät herausgestellt, sofern man keinen Zugriff auf ein MacOS-Gerät oder -Betriebssystem hat. Dies lässt sich zwar nicht auf das Framework zurückführen, sondern auf die Philosophie des Unternehmens **Apple**, sollte jedoch trotzdem an dieser Stelle Erwähnung finden.

Es ist zwar möglich, eine virtuelle Maschine mit einem MacOS-Betriebssystem einzurichten, jedoch gestaltet sich das eventuell sehr umständlich und nicht sehr praktikabel. Da jedoch bei der Entwicklung einer plattformübergreifenden Applikation das Ausführen auf allen Zielplattformen aus Testzwecken vorgesehen sein sollte, erfährt man als Entwickler unter Umständen Unannehmlichkeiten.

6.3.4. UI-Performance

Da Flutter-Applikationen nativ kompiliert ausgeführt werden, sollte es laut Dokumentation auf einfache Weise möglich sein, bei der Ausführung von Software konstante 60 Bilder in der Sekunde zu erreichen. [42, Abschnitt “What kind of App Performance can I expect?”]

Weiterhin bietet Flutter den sogenannten **Flutter Performance Profiler**[15], der es ermöglicht, die Performance-Eigenschaften der Applikation bei Laufzeit zu Überwachen.

6.3.5. Support und Community

Da sich Flutter noch in der Beta-Phase der Entwicklung befindet, ist mit einigen Problemen beim Einsatz zu rechnen, jedoch kann bei Nachverfolgung der im Git-Repository erfolgenden Fortschritte und Meilensteine im Allgemeinen davon ausgegangen werden, dass kontinuierlich an regelmäßigen Updates und Softwareverbesserungen gearbeitet wird.

Weiterhin hat sich die Verwendung von Flutter im Falle dieser Arbeit als sehr stabil und zuverlässig geäußert.

Aufgrund des derzeitig hohen und weiter steigenden Interesses an dem Framework ist eine sehr aktive Community zu erwarten, welche auch zur Lösung von Verständnisproblemen, dem Einholen von Ratschlägen und zum einfachen Interessenaustausch als Anlaufstelle herangezogen werden kann.

6.3.6. Framework-spezifische Features

Das wohl ausschlaggebendste Feature von Flutter ist die sogenannte **Hot Reload Funktion**. Es wird während der Laufzeit ermöglicht, innerhalb von Millisekunden die UI-Komponenten neu zu generieren ohne den inneren Zustand des Systems zurückzusetzen. Dies sorgt für eine extrem flexible Gestaltung und Änderung des Oberflächen-Designs während der Laufzeit, was nicht nur zur Einarbeitung in das Framework besonders nützlich ist, sondern auch bei der direkten Entwicklung mit Kunden oder den späteren Anwendern. Beispielsweise könnten Kunden bei Präsentation der Software Änderungswünsche äußern, welche je nach Aufwand der Änderung direkt umgesetzt werden können. Dies könnte bei den typischerweise häufigen Zwischenpräsentationen im Rahmen von agilen Softwareprojekten zu einer starken Senkung von Kommunikationsbarrieren sorgen.

6.3.7. Zusammenfassung der Bewertung

Zusammenfassend lässt sich sagen, dass sich Flutter ausgesprochen gut für die zielgruppennahe Front-End-fokussierte Entwicklung von mobilen Applikationen nutzen lässt. Die größten Stärken von Flutter liegen ganz eindeutig in den schnell und effizient zu erreichenden Ergebnissen, was Oberflächendesign betrifft. Die Kunden und späteren Nutzer können unkompliziert in die Entwicklung und Veränderung der Oberfläche miteinbezogen werden, was sich in der Entwicklung von mobilen Applikationen stark anbietet.

Allerdings muss darauf hingewiesen werden, dass Flutter aus der eigenen Natur heraus kein Allesköninger ist, was sich durch die Präsentation des Frameworks seitens Google in der Öffentlichkeit erst auf den Zweiten Blick erkennen lässt, da der Konsument der Dokumentation oder der verschiedenen veröffentlichten Vorträge von dem vermeintlichen Überschuss an Vorteilen überwältigt wird.

Ab einem gewissen Punkt müssen die jeweiligen Entwickler selbst tätig werden oder auf bereits entwickelte externe Plug-Ins zurückgreifen und können sich, wie beispielsweise bei den in Kapitel 6.3.3 angesprochenen Push-Notifications, nicht allein auf Flutter stützen. Dies kann wiederum unter Umständen zu einer erhöhten Komplexität hinsichtlich der Wartbarkeit der Software führen, da sich somit auf eigene Verantwortung von Googles Versprechen einer plattformunabhängigen Entwicklung entfernt wird.

Flutter wird im Rahmen dieser Diskussion als sehr geeignet für die Entwicklung mobiler Applikationen bewertet, da das Framework viele Funktionen bietet oder vereinfacht, welche ihre Stärken besonders in der benutzernahen Softwareentwicklung hinsichtlich der schnell zu generierenden prototypischen Fortschritte und Änderungen von Oberflächen-Konzepten zeigen. Mit der Programmiersprache Dart enthält Flutter weiterhin eine optimierte und mächtige Programmiersprache, welche im Gegensatz zu der fokussierten Front-End-Ausrichtung von Flutter auch ein breites Spektrum von Möglichkeiten für die Back-End-Entwicklung bietet.

Es bleibt jedoch abzuwarten, ob das Framework Flutter bezüglich der Relevanz in den nächsten Jahren von Bedeutung bleibt oder ob es genauso schnell wieder vom Markt verschwindet, wie es erschienen ist.

7. Präsentation der Ergebnisse

Die Erstellung einer Bachelorarbeit ist mit dem Durchlaufen vieler verschiedener Teilprozesse verbunden, wobei jeder dieser Teilprozesse unterschiedliche Erkenntnisse, Zusammenhänge sowie auch Probleme offenbaren kann.

Im folgenden zusammenfassenden Kapitel werden die erreichten Ergebnisse sowie die während der Arbeit aufgetretenen Schwierigkeiten und Erkenntnisse resümiert. Außerdem wird ein Ausblick der Software gegeben und die im Laufe der Arbeit gesammelten und entwickelten Ideen beschrieben.

7.1. Wichtigsten Erkenntnisse der Arbeit

Die Untersuchung dieses Problembereiches brachte im Allgemeinen das Ergebnis hervor, dass Studierende und Professoren im Kern sehr kooperierende Erwartungen und Ansichten bezüglich dem Bearbeiten von Bachelorarbeiten haben und das eigentliche Problem in der Ausführung des Projekts selbst liegt.

Studierende sehen sich zum ersten Mal mit einem umfangreichen und komplexen Projekt konfrontiert, wofür sie nicht ausreichend Vorwissen besitzen. Die fehlenden Kompetenzen umfassen somit überwiegend das Wissen um die Eigenschaften und die Durchführung einer wissenschaftlichen Arbeit, sowie der mangelhaften Zeitplanung, die laut der ermittelten und überwiegend einheitlich ausfallenden Aussagen der befragten Personen im Laufe des Studiums nicht ausreichend ausführlich vermittelt wird.

Das größte identifizierte Problem hierbei ist, dass sich Zeitplanung nur durch eigens gesammelte Praxiserfahrung und nicht durch das theoretische Aneignen von Fachwissen oder Besuchen des, an der Fachhochschule Lübeck angebotenen, Bachelorarbeit-Seminars erlernen lässt. Da die im Studium durchlaufenden Projekte jedoch meist in Gruppenarbeit verlaufen und zudem, besonders aus diesem Grund, sehr überschaubar ausfallen, kommt es unter Umständen nicht zu dem gewünschten Lernprozess. Die somit erste **Forderung** von Zeitmanagement mündet offenbar in einer **Überforderung** bei der Bearbeitung der Bachelorarbeit.

Hinsichtlich der fachspezifischen Kompetenzen der Informatik gibt es auffallend wenig Schwierigkeiten, da Studierende im Laufe des Studiums an der Fachhochschule Lübeck größtenteils die für die Bachelorarbeit benötigten Methoden, Werkzeuge und Fähigkeiten durch die zahlreichen und vielseitigen Module vermittelt bekommen haben.

Das Ergebnis der Nutzungsevaluation brachte hervor, dass die Softwarelösung durchaus Potential zur Qualitätssteigerung der Bachelorarbeiten von Studierenden haben kann. Besonders die enthaltene Funktion der Herausforderungen als Leitfaden sowie der Bachelorarbeit-Guide als Informationsquelle stoßen auf hohen Zuspruch seitens der Professoren und der Studierenden. Um eine ausreichend fundierte Aussage über den Nutzen der Software zu ermöglichen, muss jedoch ein Praxis-Test durchgeführt werden, welcher sich im Optimalfall auf mehrere Iterationen von Bachelorarbeit-Studiengängen bezieht.

7.2. Schwierigkeiten bei der Bearbeitung

Als persönliche größte Schwierigkeit betrachte ich das Zeit- und Aufwandsmanagement des Gesamtprozesses, da die zugehörigen Teilprozesse auf eine unvorhergesehene hohe Komplexität gewachsen sind. Konkret zähle ich hierzu besonders das Durchlaufen der Anforderungsanalyse, welche durch die in diesem Projekt detaillierte Vorgehensweise der Durchführung und Auswertung der Interviews und Befragungen, einen enormen Arbeitsaufwand eingenommen hat. Ferner führte dies jedoch auch zu einem sehr detaillierten Bild des Problembereiches, welches in Kapitel 3 ausführlich beschrieben wird. Dies wird im Rahmen der Entwicklung dieser Applikation als essenziell und von hohem Wert betrachtet.

Durch diesen betriebenen Aufwand kam es somit bei den darauf folgenden, nicht parallelisierbaren Teilprozessen zu einer starken Verschiebung im Zeitplan, was wiederum zu einer Minderung der Gesamtqualität des Projektausgangs geführt hat.

Dies wurde vor allem in der Evaluationsphase als enorme zu bewältigende Schwierigkeit wahrgenommen, weshalb die Auswertung des Nutzens der Applikation als nicht ausreichend fundiert angesehen werden kann. Dies liegt jedoch weiterhin an den eingeschränkten Evaluationsmöglichkeiten, die durch den Rahmen des Projekts innerhalb so kurzer Zeit gegeben ist, da hierfür eigentlich ein groß angelegter Einbezug von Studierenden und Professoren über mehrere Iterationen von Bachelorarbeiten nötig wäre, um die gewünschte Aussagekraft zu erhalten.

Dennoch werden die umgesetzten Bestandteile der Aufgabenstellung im folgenden Verlauf erläutert und dienen somit als neuer Ausgangspunkt des Projekts.

7.3. Erfüllung der Aufgabenstellung

Bei der Durchführung des Projektes wurde die Erfüllung der Aufgabenstellung als zentrales Projektziel angesehen. Hierbei gestaltete es sich jedoch aus Zeitgründen schwierig, die angestrebte Qualität in voller Gänze zu erreichen und es mussten bezüglich der Vollständigkeit einige Abstriche gemacht werden.

Im Verlauf des Kapitels werden die erreichten Ziele beschrieben und die nicht vollständig umgesetzten Lösungen begründet. Die dabei gewählte Kategorisierung setzt sich aus den, aus der Aufgabenstellung hervorgehenden, Aufgabenfeldern der Software zusammen.

7.3.1. Unterstützung bei Bearbeitung der Bachelorarbeit

Um den Studierenden Hilfestellungen, in Form von Richtlinien und Ratschlägen zu ermöglichen, wurde einerseits die Bachelorarbeit-Guide Funktion implementiert, welche als Nachschlagewerk zu Rahmenbedingungen, Hinweisen und Bearbeitungsdetails dienen soll.

Der Inhalt dieser Softwarefunktion wird durch das Einlesen einer JSON-Datei aus einem Git-Repository ermöglicht. Diese Schnittstelle sorgt dafür, dass diese Inhalte beliebig verändert oder erweitert werden können, ohne einen Eingriff in den Programmcode zu erfordern.

Weiterhin liegen die bereits eingefügten Wissenssegmente bei Ende dieses Projekts aufgrund des zu betreibenden Zeitaufwandes nur in beispielhafter Form vor und gelten vor allem hinsichtlich der Methodenübersicht nicht als vollständig.

Wie bereits erwähnt kann der Inhalt jedoch beliebig erweitert werden, was konkret bedeutet, dass beispielhafte Gliederungen und alle gewünschten Hinweise, Übersichten und Anleitungen zu fachspezifischen Methoden und Werkzeugen zur Anforderungserhebung, Nachweisführung und/oder Aufbereitung von Ergebnissen beliebig verändert oder hinzugefügt werden können.

Ein weiterer wichtiger Bestandteil der Unterstützung der Studierenden liegt in der Definition von Herausforderungen. Diese Herausforderungen sollen einen Leitfaden für die Studierenden darstellen, welcher so allgemein wie nötig und so detailliert wie möglich gewählt werden soll. Die Idee hierbei ist, dass die Studierenden sich an dem, durch die Herausforderungen definierten, Rahmen und Ablauf einer Bachelorarbeit orientieren können, um von Anfang an einem konkreten Roten Faden folgen zu können und nicht "einfach drauf los" zu arbeiten. Es ist von essenzieller Bedeutung, dass die Inhalte so gewählt werden, dass sie die Inhalte einer themenspezifischen Lösung innerhalb einer Bachelorarbeit nicht beeinflussen, da dies einen zu großen Eingriff in die Prüfungsleistung der Studierenden bedeuten würde und dies sich durch einen manipulativen Charakter sogar negativ auswirken könnte.

Hinsichtlich der Erweiterbarkeit ist es bei den Herausforderungen und Achievements im Gegensatz zu dem Bachelorarbeit Guide zum Ende des Projekts nicht möglich, diese softwareextern zu bearbeiten oder hinzuzufügen, jedoch lässt sich auch diese Funktion bei mehr Investition von Zeit und Aufwand nachträglich in das Projekt integrieren.

7.3.2. Unterstützung bei Zeitplanung

Um die Studierenden bei der Zeitplanung der Bachelorarbeit zu unterstützen, wurde eine Softwarefunktion implementiert, welche es den Studierenden ermöglicht, Meilensteine und Teilaufgaben individuell zu verwalten und diese durch eine geeignete Oberflächenpräsentation in Form eines Zeitstrahls als permanenten Begleiter und somit auch als Erinnerungsmöglichkeit innerhalb der Applikation zu visualisieren. Eine konkrete Lösung durch interaktive Erinnerungen, welche beispielsweise mittels Push-Benachrichtigungen ausgelöst werden könnten ist zum Ende des Projektes nicht umgesetzt worden. Dies lässt sich jedoch durch die gewählte Architektur nachträglich in das Projekt integrieren.

7.3.3. Motivation der Studierenden

Um Studierende bei der Durchführung der Bachelorarbeit zu motivieren, wurde ein Gamification-System implementiert, welches den Studierenden bei Erfüllung von Herausforderungen und Benutzung der Applikation ein positives Feedback vermitteln soll. Dies geschieht bei Erfüllen von Herausforderungen durch das Auslösen von Notifications, welche den Herausforderungen entsprechend ironische, sarkastische und lobende Texte oder Sprüche enthalten. Diese Texte und Sprüche bilden gemeinsam mit sich durch die Wertigkeit unterscheidenden Medaillen, den Inhalt der sammelbaren Achievements. Inwiefern diese Strategie ihre Wirkung zur Motivationssteigerung beitragen kann, ist als sehr Zielgruppen-spezifisch anzusehen und sollte weiterhin untersucht werden. Bei einigen Personen wirkt das Feedback laut der Evaluation durchaus positiv auf die Motivation, bei anderen jedoch nicht.

7.4. Ausblick des Projekts

Der Abschluss dieses Projekts definiert einen Orientierungspunkt, welcher einen Ausblick auf weitere zukünftig zu bearbeitende Fragestellungen und die Gestaltung der Softwarelösung zulässt. In diesem Kapitel werden die Fragen und Ansätze, welche im Rahmen dieses Projekts nicht untersucht oder realisiert werden konnten, angesprochen.

7.4.1. Ausblick auf weitere Untersuchungen

Im Laufe der Anforderungsanalyse wurde im Rahmen der Interviews oft über die zeitliche Konstellation und Abfolge der an der Fachhochschule Lübeck verfügbaren Module diskutiert, da zum jetzigen Zeitpunkt kein konkretes Modul zur Vorbereitung auf das wissenschaftliche Arbeiten angeboten wird. Eine Problemuntersuchung hinsichtlich des Bedarfs an einem solchen Modul oder einer Alternativlösung wird nach Abschluss dieses Projekts empfohlen.

7.4.2. Ausblick auf Softwarefunktionen

Der im Rahmen der Bachelorarbeit entwickelte Software-Prototyp enthält bereits unterschiedliche Funktionen, welche zur Lösung der auftretenden Probleme dienen sollen. Die hierbei entwickelten Software-Inhalte verfügen im Rahmen einer prototypischen Entwicklung über eine nicht ausreichende Qualität zum Einsatz in einer realen Umgebung. Es wäre zwingend notwendig die vorhandenen Funktionen besonders hinsichtlich der Usability und Wirkungsweise zu überarbeiten oder zu erweitern.

Im folgenden Verlauf werden einige konkrete Ideen zur Erweiterung oder Überarbeitung aufgeführt:

- Erweiterung der Software für andere Studiengänge**

Die entwickelte Software wurde nur für den Studiengang Informatik/Softwaretechnik der Fachhochschule Lübeck entwickelt. Es wäre denkbar, zu untersuchen, inwiefern andere Inhalte von Studiengängen und Hochschulen in die Software integrierbar sein können.

Denkbar wäre eine studiengangspezifische Auswahl bei erstmaligem Öffnen der Software.

- Externe definierbare Erinnerungen**

Eine mögliche Funktion wäre es, durch das externe Absenden von Benachrichtigungen an Studierende, client-seitige Push-Benachrichtigungen auszulösen. Dies könnte beispielsweise in Form von Newsletter-Benachrichtigungen über neue Bachelorthemen sowohl von internen Betreuern, als auch externen Firmen geschehen, die auch in der Software selbst angezeigt werden.

- Auswahl von Sprachpaketen**

Eine wichtige Funktion, welche bis zum Ende des Projekts nicht realisiert wurde, ist die Auswahl von verschiedenen Sprachen, da auch viele Studierende aus dem Ausland die Fachhochschule Lübeck besuchen.

- **Eine geeignetere Darstellung von Informationen in Bachelorarbeit Guide**

Zum Ende des Projekts liegen die Vorbereiteten Informationen in einer sehr rudimentären Fassung vor. Im Rahmen der Entwicklung des Prototyps ist lediglich das Darstellen von Buttons, welche auf den nächsten Bildschirm leiten sowie die Darstellung des Inhaltes durch ausklappbare Oberflächen-Elemente gegeben. Hier können weiterhin vielfältigere Darstellungsmöglichkeiten wie Grafiken oder eingebettete Videos ihren Platz finden, welche die Übermittlung von Informationen auf eine höhere und zeitgemäßere Qualitätsebene bringen könnten.

- **Dynamische Wochenberichte**

Ein gutes Feature um dem Benutzer zu zeigen was dieser diese Woche geschafft hat, könnte durch abschließende Berichte einer Woche umgesetzt werden. Hierbei könnte dem Benutzer durch eine Übersicht gezeigt werden welche Aktionen er an welchem Tag durch nutzen der Meilensteinplanung getätigter hat. Beispielsweise wären hierfür das Erfüllen von eigens erstellten Unteraufgaben der Meilensteine oder das abschließen eines Meilenstein als ganzes. Weiterhin könnte dem Nutzer hier auch angezeigt werden welche Erfolge er hinsichtlich den Herausforderungen und Achievements geschafft hat. Auf diese Weise könnte der bereits vorhandene Softwareinhalt genutzt werden und ohne großen Erweiterungsaufwand zu einer Qualitätssteigerung der Benutzererfahrung führen.

Tabellenverzeichnis

2.1. Übersicht der Spiel-Design-Elemente [7, vgl.]	8
3.1. Übersicht der Interessengruppen	19
3.2. Einbezug der Studierende in Wahl der Gamification-Strategie	34
3.3. Erwartung der Beteiligten an die Applikation	35
4.1. Abdeckung der Produktfunktionen	51
6.1. Übersicht der Reducer-Tests	71

Abbildungsverzeichnis

1.1.	Online Umfrage: Zusammenfassung der Umfrageergebnisse	3
2.1.	Interest Graph nach Bartle [6]	11
2.2.	Flutter High-Level Systemarchitektur [17]	12
2.3.	Darstellung (iOS und Android): Plattformunabhängige Entwicklung von Applikationen [41, What is Flutter?]	13
2.4.	Beispiel: Aufbau von Widgets [9]	14
2.5.	Flutter Widgets Architektur [45]	14
2.6.	Flutter Rendering [17]	15
3.1.	Beschreibung der Analysestrategie	20
3.2.	Benutzeroberfläche Fortschritts-Management	36
4.1.	Benutzeroberfläche Meilensteinverwaltung	43
4.2.	Benutzeroberfläche Guide	44
4.3.	Benutzeroberfläche Herausforderungen	46
4.4.	Benutzeroberfläche Achievements	47
4.5.	Benutzeroberfläche Navigations-Menü	48
4.6.	Benutzeroberfläche Dashboard	49
4.7.	Visualisierung der Achievements	53
5.1.	Architekturübersicht	55

Listings

5.1.	Datenfelder der AppState-Klasse	57
5.2.	Meilenstein Action Beispiel	64
5.3.	Meilenstein Reducer Beispiel	65
5.4.	Container-Komponente einer Meilensteinänderung	66
5.5.	ViewModel von Container-Komponente einer Meilensteinänderung	67
5.6.	Bestätigung einer Meilensteinänderung Methode	68
5.7.	Notification Middleware	69
5.8.	checkAchievementNotification Methode	70

A. Anhang

A.1. Befragung der Beteiligten

A.1.1. Transkripte der Interviews mit den Professoren

Interview I

1
2 **Frage: Ihr Fachgebiet wird auf der Internetseite der Fachhochschule Lübeck als**
3 **“Data-Management, Software Engineering” angegeben. Wenn ein Studierender Sie als**
4 **Betreuer für die Bachelorarbeit wünscht und mit einem Themenvorschlag zu Ihnen**
5 **kommt, der nicht Ihrem Fachgebiet entspricht, gibt es dann Themen bei denen Sie**
6 **den Studierenden an Ihre Kollegen weiterleiten?**

7
8 Antwort: Offen bin ich grundsätzlich schon, aber wenn ich das Gefühl habe, dass ich aus
9 meiner fachlichen Kompetenz das Thema nicht gut betreuen kann, würde ich denjenigen
10 weiterdelegieren. Also Rechnernetze wäre zum Beispiel ein Beispiel wo ich sagen würde, es
11 gibt noch Andreas Hanemann, der das wahrscheinlich besser betreuen kann weil es eben
12 dann auf tieferer Protokollebene stattfindet. Das was ich jetzt so mache ist eben HTTP
13 Protokolle und darüber. Und wenn es um Protokolle darunter geht, bin ich nicht mehr der
14 richtige Ansprechpartner.

15
16 **Und andersherum gefragt, was sind typische Arbeiten, die man bei Ihnen schreibt und**
17 **wie sind sie im allgemeinen aufgebaut? Es gibt ja zum Beispiel Arbeiten, bei denen**
18 **die Entwicklung einer Software im Vordergrund steht.**

19
20 Ich versuche das mal zu Strukturieren.

21
22 Also das wäre aus meiner Sicht der erste Typ. Der Studierende entwirft und implementiert
23 Softwareprototyp. Aus meiner Sicht ist es immer ein Prototyp bei einer Abschlussarbeit und
24 ich erwarte kein fertiges Produkt, was nach der Arbeit in Betrieb genommen direkt werden
25 kann. Und es geht ja auch nicht darum, jetzt vollständig möglichst viele Use Cases zu
26 implementieren, sondern es geht mehr um den Entwurf einer Architektur, die vielleicht sich
27 im Laufe der Zeit auch weiterentwickeln kann und man sollte dann immer Grundlegend
28 zeigen das man in der Lage ist als Studierender ein paar Use-Cases zu implementieren und
29 dann werden ja meistens einige auch rausgesucht und die werden dann in der Arbeit
30 dargestellt.

31
32 Dann haben wir den ganzen typischen Softwareentwicklungszyklus mit dabei, so wie
33 Anforderungen muss man gucken, woher kommen die eigentlich. Hat der Student überhaupt
34 die Aufgabe die Anforderung zu erheben oder sind die Anforderungen bereits erhoben
35 worden aus einem externen Umfeld? Wenn er es im Umfeld einer Firma macht oder einer
36 anderen Organisation, dann sind die Anforderungen manchmal schon gegeben, dann muss
37 er sie nur noch verstehen und durchdringen und kann sie vielleicht nochmal infrage stellen.
38 Also es kann seine Aufgabe sein die Anforderungsanalyse, muss aber nicht.

39
40 Dann die Entwurfsphase sollte auf jeden Fall seine Aufgabe sein. Es kann auch sein, dass
41 er nicht auf der grünen Wiese aufsetzt, sondern es kann ja schon Software existiert, die
42 erweitert werden soll. Dann gibt es eine Ist-Architektur und eine Soll-Architektur.
43 Oder er fängt auf der grünen Wiese an, dann gibt es eine komplett neue Architektur.
44 Dann kommt die Implementierungsphase, Das will man natürlich auch sehen, das der
45 Student in der Lage ist ein Stück Software zu produzieren und dann noch das Thema
46 Testing, er sollte sich auch Gedanken zu dem Testing machen und das Thema

47 Inbetriebnahme spielt meistens keine Rolle. Das wäre so der erste Typ von Arbeit, wo es
48 aus meiner Sicht schon darum geht, dass der Student ein Softwareprojekt an sich
49 organisieren kann und das auch technisch im Griff hat. Sowohl Entwurfs- als auch
50 Implementierungs- und Testphase. Um was es da inhaltlich geht - welche Technologie - das
51 spielt eigentlich gar keine Rolle.

52

53 Dann der zweite Typ von Arbeit, wären empirische Arbeiten, wo meistens irgendwas
54 evaluiert wird. Da kommen natürlich auch viele Unternehmen mit dem Ziel, den Studenten
55 auf so eine Problemstellung anzusetzen. Zum Beispiel "Es gibt eine neue Technologie und
56 wir wollen mal gucken wie die sich eigentlich Verhält im Vergleich zu bisher gesetzten
57 Technologien", dann ist meistens die erste Aufgabe, sich ein Kriterienkatalog zu überlegen.
58 Wie kann man es nachher ausmessen und dann kommt danach der eigentliche
59 Experimentaufbau und die Durchführung dieser Messung, die auch wieder ganz
60 unterschiedlich ausgeprägt sein kann. Es kann Fragebögen geben, so wie jetzt in dieser
61 Situation, wo man Leute befragen muss. Es gibt natürlich auch die Möglichkeit mit
62 Onlinefragebögen eine größere Stichprobe zu sammeln. Dann gibt es auch Fallstudien, wo
63 ich mir die Technologien anschau - Technologie A und Technologie B für den gleichen Use-
64 Case - und muss mir vorher Kriterien überlegt haben. Und am Ende ist dann natürlich die
65 eigentlich Evaluation der Beitrag des Studierenden - die Ergebnisauswertung des
66 Experiments. Das wäre die zweite Klasse.

67

68 Und da jetzt die Frage, ob die dritte Klasse mit darein gehört? Das ist manchmal schwer zu
69 trennen. Die könnte man auch mit zur zweiten Klasse zählen.

70 Das ist so in dem Bereich Datenanalyse/Machine Learning. Manchmal Arbeiten wo dann
71 öffentlich ausgeschriebene Wettbewerbe im Bereich Datenanalyse angegangen werden.
72 Also zum Beispiel über Kaggle oder jetzt hatten ich eine Arbeit mit diesem "Spotify Million
73 Playlist Dataset". Das heißt, da ist eine Datenanalyseaufgabe - die sind natürlich dann offen
74 vom Ergebnis her - und aus meiner Sicht geht es darum das der Student sich erstmal
75 explorativ den Datensatz anschaut.

76

77 Ach ihr habt sowas ähnliches auch gemacht. Nicht als Abschlussarbeit, sondern in dem
78 Wahlpflichtmodul.

79

80 **Ja, genau.**

81

82 Also erstmal explorativ den Datensatz anschauen und dann überlegen, mit welchen Machine
83 Learning Verfahren könnte ich denn im Kontext dieser Aufgabenstellung zu guten
84 Ergebnissen kommen. Dann hat man in der Regel ja auch so ein Benchmarking, wo man
85 das Ergebnis dann hochlädt und dann ein Feedback bekommt.

86 Hier ist das Lernziel aus meiner Sicht, dass der Studierende zeigen soll, dass er die Machine
87 Learning Algorithmen versteht und auch zur Anwendung bringen kann. Bei einer sehr guten
88 Arbeit, wendet der Studierende nicht nur einen Ansatz an, sondern mehrere und vergleicht
89 die. Da hat man teilweise auch Herausforderungen aufgrund der Größe der Datenmenge.
90 Das sind dann so diese Benchmarkanwendungen. Es gibt auch so
91 Datenbankbenchmarks, da ist das das gleiche. Das hatte ich auch schon im
92 Wahlpflichtmodul. Da war es dieser Big Data Benchmark. Das könnte man natürlich auch in
93 die Abschlussarbeit reinbringen. Da hat man am Ende dann auch ein Evaluationsergebnis,

94 aber der Kontext und die Rahmenbedingungen sind durch so ein Wettbewerb meistens
95 enger gesteckt.
96
97 Eigentlich sind es diese zwei Arten (von Arbeiten). Einmal konstruktiv, irgendetwas
98 erschaffen oder analytisch irgendetwas evaluieren/analysieren.
99
100 Was wir kaum haben (im Studiengang Informatik/Softwareentwicklung). In der BWL zum
101 Beispiel, gibt es auch reine Literaturarbeiten
102
103 **Dankeschön. Sie haben es ja bereits angerissen, was ihre Erwartungen betrifft. Aber**
104 **was sind Ihre allgemeinen Erwartungen an den Bacheloranden?**
105
106 Ich erwarte von einer Bachelorarbeit erstmal keine neuen Erkenntnisse. Das würde ich von
107 einer Dissertation erwarten - wirklich neue Erkenntnisse. Und die Masterarbeit ist irgendwo
108 dazwischen. Ich erwarte aber, dass der Studierende sich selbstständig - nicht so stark
109 begleitet, wie in den Praktika in den Semestern davor - in ein Thema einarbeitet, indem er
110 sich vielleicht vorher noch gar nicht so gut auskannte. Da erwarte ich eigentlich strukturiertes
111 Vorgehen des Studenten und das er sich am Anfang sowas wie einen Zeitplan macht und
112 Meilensteine überlegt. Ob er das dann mit einem Projektmanagement Tool macht ist mir
113 eigentlich ganz egal. Er kann es auch einfach auf einen Zettel schreiben. Aber das merkt
114 man ja schon relativ früh - bei den ersten beiden Treffen - dann, ob der Studierende sich
115 einen Plan über die gesamte Laufzeit gemacht hat oder ob das einfach so laufen lässt.
116 Je nachdem, sehe ich das als meine Aufgabe ihn häufiger zu mir zu bitten oder weniger. Aber
117 meistens lasse ich ihn relativ frei Laufen. Beispielsweise die beiden chinesischen Studenten
118 kommen einmal in der Woche, weil man das Gefühl hat, dass sie das brauchen. Bei den
119 Online Studenten habe ich sogar das Gefühl, dass sie noch selbstständiger sind. Aber das ist
120 jetzt ein subjektiver Eindruck.
121
122 Was da nachher das Ergebnis betrifft, sehe ich die Arbeit immer in zwei Teilen. Es sollte einen
123 Literaturteil geben, wo der Studierende es geschafft hat, das was er auf technischer Ebene
124 implementiert hat, auch zu vergleichen mit verwandten Ansätzen - Related Work. Zu
125 versuche, dass auf eine höhere Abstraktionsebene zu bringen und grundlegende Muster zu
126 erkennen, diese Muster zu beschreiben - z.B. Architekturmuster, Entwurfsmuster, die nicht
127 nur die nächsten zwei Jahre eine Rolle spielen, sondern länger schon eine Gültigkeit haben.
128 Und dann eben auch möglichst einschlägige Quellen verwenden. Damit meine ich, sich nicht
129 nur auf allgemeine Softwaretechnik-Lehrbücher zu beziehen, sondern das man in diesen
130 Softwaretechnik-Lehrbüchern Unterkapitel findet - zum Beispiel zum Thema Requirements
131 Engineering - und sich daraufhin spezielle Bücher zu dem Thema Requirements
132 Engineering anschaut und aus denen zitiere. Und dann kann ich natürlich immer noch ein
133 Schritt weitergehen und sagen, dann finde ich am Ende der Kapitel der Lehrbücher
134 Verweise auf wissenschaftliche Papiere und gucke mir die auch noch an und suche mal
135 nach den Primärquellen. Und das unterscheidet eine bessere Arbeit, von einer die nicht so
136 gut ist - das man die Suche nach den Primärquellen und die Diskussion des Themas
137 erkennt. Das so eine Arbeit im Grundlagenteil nur an einer Quelle orientiert ist und man
138 eigentlich nichts anderes getan hat, als diese Quelle umzuschreiben oder zu übersetzen,
139 sondern das da mehrere Quellen ernsthaft eingeflossen sind und manchmal gibt es ja auch
140 Aussagen in Quellen die sich widersprechen. Und das wäre dann die Literaturarbeit.
141

142 Der andere Teil wäre der praktische Teil. Hier ist es Aufgabe des Studierenden und des
143 Betreuers zu Anfang zusammen die Komplexität so zu wählen, dass es nicht zu viel oder zu
144 wenig ist. Wenn es zu wenig ist, kann man sagen "Das was du gemacht hast war zu trivial
145 und führt dann auch nicht zu einer guten Note" und wenn es zu viel ist, dann kommt es
146 entweder nicht zu einem runden Ende oder es ist alles nur so angerissen. Das heißt, dann
147 muss der Betreuer mit seinem Erfahrungswissen ein bisschen Steuern. Der Studierende
148 aber auch, vor allem wenn es eine externe Arbeit aus einem Unternehmenskontext ist. Dann
149 fällt es mir manchmal schwer, es von Anfang an richtig einzuschätzen - wie viel Aufwand
150 sich da wirklich verbirgt.

151 Das Ergebnis soll natürlich am Ende laut Aufgabenstellung gelöst sein - dafür hat man ja die
152 Aufgabenstellung, wo drinnen steht, welche Funktionalitäten abgedeckt werden sollen - und
153 zum anderen sollen natürlich externe Anforderungen auch abgedeckt werden. Die
154 Anforderungen kann man ja aufschreiben im Rahmen der Arbeit.

155

156 Das wäre so der praktische Teil. Und da gibt es durchaus ja Unterschiede, je nach Art der
157 Arbeit. Wenn jemand das Gefühl hat, der Praxisteil war nicht so ergiebig oder etwas
158 einfacher und war schnell fertig, der hat natürlich die Möglichkeit das im Literaturteil wieder
159 auszugleichen. Grundsätzlich würde ich sagen, die Empfehlung ist 50/50, aber es gibt auch
160 Schwankungen von 30/70 und 70/30 und das ist auch absolut ok.

161

162 **Darauf aufbauend wäre eine Frage, wo denn häufig Probleme auftreten? Gibt es
163 Probleme seitens der Studierenden, die immer wieder auftreten? - Stichwort
164 Zeitmanagement.**

165

166 Es gibt Studierende, die neigen dazu alles auf den letzten Drücker zu machen. Das ist
167 natürlich nicht gut und das wissen die Studierenden meistens auch selbst. Es ist von Vorteil,
168 wenn man sich alle 2 oder 3 Wochen mit dem Betreuer trifft und dann eine Vorversion hat,
169 oder eine Prototypen den man Vorstellen kann. Es ist immer besser über irgendwelche
170 Artefakte zu sprechen als über das was man noch tun möchte und welche Probleme einen
171 davon abhalten, das man das noch nicht getan hat. Also eine Empfehlung an die
172 Studierenden wäre "Wenn man einen Termin mit seinem Betreuer eingeht, am besten immer
173 die Entwicklung zeigen können".

174

175 Beim ersten Termin erwarte ich immer eine Gliederung und 3 oder 4 Einträge im
176 Literaturverzeichnis als Hauptquellen. Und beim nächsten treffen sollte ich dann immer was
177 neues zeigen können und nicht nochmal das gleiche. Ein paar Einträge im
178 Literaturverzeichnis hin und herschieben geht innerhalb von 5 Minuten - das meine ich damit
179 nicht. Und wenn der Studierende sich hauptsächlich mit lesen beschäftigt hat, kann man
180 trotzdem in den Unterkapiteln zeigen, welche Quellen man eigentlich zitieren möchte.
181 Beispiel "Thema A nimm ich aus Kapitel 5 von dem Buch; Thema B aus einer anderen
182 Quelle; ..." Also auch wenn man die Zeit nur zum Einarbeiten genutzt hat, kann man das
183 auch zu Papier bringen.

184

185 Dann ein Thema. Wissenschaftliches Arbeiten an sich. Und da müssen wir als
186 Fachhochschule, speziell auf diesen Studiengang (Informatik/Softwareentwicklung) uns mal
187 an die Nase fassen, dass wir wenig vorarbeiten.

188 Es gibt bei uns in der Form keine Übung in Form einer wissenschaftlichen Seminararbeit
189 oder Hausarbeit, die sich als Übung für die Bachelorarbeit eignet. Ich kenne das von einer

190 anderen Hochschule - da sind 2 Seminararbeiten vorher Pflicht - und dann kommt natürlich
191 auch tendenziell ein besseres Ergebnis in der Abschlussarbeit dabei raus. Weil der
192 Studierende dann genau das schonmal gemacht hat und erkennt, woraus diese
193 Literaturarbeit eigentlich besteht. Es ist ja nicht die Aufgabe, wenn man ein scheinbar
194 relevantes Paper mit 20 Seiten bekommt, dass ich diese 20 Seiten im Detail durchlese und
195 alles verstehe.

196 Es geht ja meistens darum die Idee des Papers zu verstehen - Literaturscreening gehört ja
197 auch dazu - und die paar Seiten, die wirklich relevant sind, für das was man tut, zu lesen
198 und zu verstehen. Ein Lehrbuch liest man ja auch nicht von vorne bis hinten.

199 Der praktische Teil - Die Beschreibung was man gemacht hat - gelingt da meistens besser,
200 aber sich mit mehreren Quellen auseinanderzusetzen und diese ineinander fließen zu
201 lassen gelingt nicht immer.

202

203 Das nächste ist dann der wissenschaftliche Anspruch, Struktur und der Versuchsaufbau.
204 Das ich mir vorher klar mache, was das Ergebnis meiner Arbeit ist und wie der Weg dahin
205 aussieht, das man messbare Kriterien braucht und welche Fragen man stellen muss um
206 dieses Kriterien dann zu erfüllen. Da gibt es ja diesen GQM Ansatz, was vielleicht auch ein
207 Vorschlag für das Bachelorseminar wäre.

208

209 Goal Question Metric

210 Goal - Man überlegt sich Ziele die man erfüllen will.
211 Question - Fragen, um diese Ziele zu erfüllen.
212 Metric - Und die Antworten auf die Questions sind meistens messbare
213 Metriken.

214

215 Diese Methode ist natürlich auf einem sehr hohen Abstraktionsniveau aber grundsätzlich
216 vielleicht mal nicht schlecht, sich das als Student zu fragen. Wie kann ich das eigentlich
217 quantifizieren. Im Evaluations Kapitel steht drinnen, was gut geklappt hat und was nicht
218 geklappt hat, aber besser wäre es, wenn das durch Zahlen belegbar ist, aber das gelingt
219 (den Studenten) nicht immer.

220

221 Ich fasse nochmal zusammen:

222 - Zeitmanagement
223 - Wie funktioniert eigentlich wissenschaftliches Arbeiten? Literaturrecherche?
224 - So etwas wie ein strukturierter Experimentaufbau, vor allem bei dem zweiten Typ von
225 Arbeit. Bei dem Ersten Typ habe ich eigentlich Anforderungen und ich muss belegen,
226 dass diese erfüllt werden.

227 **Sie haben es ja bereits angerissen mit Ihrem Gedanken, dass die Hochschule kein
228 Modul anbietet, welche die Studierenden auf diese Problematik vorbereitet. Wie
229 würden sie diese Probleme versuchen zu lösen, wenn sie könnten?**

230

231 Das ist ja erstmal kein grundsätzlich neues Thema. Als ich Student war, hatte ich
232 wissenschaftliches Arbeiten im ersten oder zweiten Semester und es ist total demotivierend.
233 Da geht nachher kein Student mehr hin, man kann auch schlecht eine Prüfung in Form einer
234 Klausur machen und es hat aus meiner Sicht auch keinen Sinn das so früh zu machen. Ich
235 finde das hier schon gut, dass es das Bachelorarbeit Seminar gibt - es findet natürlich im 6.
236 Semester statt, das heißt es ist parallel zur Bachelorarbeit. Vielleicht würde es Sinn machen,
237 wenn es im 5. Semester liegen würde, aber auch nicht viel früher. Das was man in so einem

238 Kurs an Theorie im Frontalunterricht machen kann, ist in der Regel nicht sehr spannend und
239 schwierig zu gestalten. Also würde ich versuchen diesen Teil möglichst kurz und prägnant
240 zu halten und stattdessen den Student selbst eine Art Mini-Übung zu einer
241 wissenschaftlichen Arbeit in Form einer Seminararbeit schreiben zu lassen. Und dann eine
242 Form der Arbeit, die sich ganz stark auf den Literaturteil bezieht. Das heißt man bekommt
243 ein oder zwei Quellen vorgegeben zu einem bestimmten Verfahren oder einer
244 mathematischen Methode und soll die dann beschreiben und an einem Beispiel dann auch
245 zeigen in Form einer Präsentation am Ende. So würde ich das machen, das haben wir auch
246 häufig schon diskutiert. Das kann dann genauso heißen wie jetzt auch und ist vielleicht nicht
247 nur einem Kollegen zugeordnet, sondern jeder Student sucht sich wie bei einer
248 Bachelorarbeit auch einen Betreuer. Vielleicht kann man auch einen Pool aufbauen an
249 Themen. Alle Kollegen dürfen dann 5 Themen angeben und die werden dann unter den
250 Studierenden dann irgendwie verteilt.

251 Das muss natürlich vom Aufwand reduziert sein, da es ja nur um das wissenschaftliche
252 Schreiben und die wissenschaftliche Literaturarbeit - der Umgang mit Quellen - geht.
253 Das könnte man machen aber man darf nicht vergessen, dass es sich dabei "nur" um die
254 Bachelorarbeit dreht. Früher gab es das Grund- und das Hauptstudium und am Ende nach
255 10 Semestern musste man erst nachweisen, dass man das wissenschaftliche Arbeiten
256 beherrscht und da hatte man natürlich zwischendurch auch Hausarbeiten auf dem Weg
257 dahin. Jetzt könnte man ja so argumentieren, dass die Masterarbeit eigentlich der Part ist, bei
258 dem Sie das zeigen müssen und die Bachelorarbeit ist die Übung, aber nun hören halt viele
259 nach der Bachelor Arbeit auf.

260 Das darf man nicht vergessen. Die Bachelorarbeit muss aus meiner Sicht nicht perfekt sein,
261 sondern ist nur eine Vorübung zur Masterarbeit.

262
263 Mein Gestaltungsvorschlag wäre also, das Seminar aus dem 6. Semester in das 5.
264 Semester zu verschieben und dann so ein Themenpool aufzubauen. Bei der Bachelorarbeit
265 erwarte ich ja vom Studenten dann, das er sich auch selbst ein bisschen um das Thema
266 kümmert. Hier würde ich sagen, der Student soll sich nicht selbst darum kümmern, sondern
267 es werden geeignete Themen aus unterschiedlichen Bereichen der Informatik
268 zusammengestellt, so das jeder sich was aussuchen kann. Die Themen könnten sich dann
269 natürlich auch wiederholen.

270
271
272 Es gibt ja auch im Internet Bachelorarbeiten die man durchlesen kann, man muss dann
273 natürlich auch selbst in der Lage sein, eine schlechte Bachelorarbeit von einer guten zu
274 unterscheiden, aber ich glaube das wird ja auch im Bachelorseminar so gemacht und das
275 klappt ja auch erstaunlich gut. Ein Student kann ja eine schlechte Bachelorarbeit von einer
276 guten unterscheiden und deshalb kann man ja auch seine eigene Arbeit relativ gut
277 einsortieren in diese Bandbreite.

278 Was mir auch auffällt ist, dass sich viele Studierende nicht rechtzeitig mit der
279 Themenfindung beschäftigen. Vielleicht sollte man da einen Hinweis geben, das kann ich
280 nicht so ganz nachvollziehen, weil es ja eigentlich bekannt ist, das man in dem 6. Semester
281 diese Abschlussarbeit schreibt.

282
283 **Das ist auch etwas, was ich von vielen Studierenden gehört habe. Die Studierenden**
284 **wissen meistens gar nicht was eigentlich "erlaubt" ist, bezüglich der Themenwahl**
285 **und aus meiner Sicht tun sich viele sehr schwierig damit, sich zu informieren.**

286 **Vielleicht haben die Studierenden Berührungsängste oder einfach Angst mal**
287 **nachzufragen, weil sie denken, dass sie dann einen schlechten Eindruck machen?**

288

289 Ja, da seh ich ganz klar eine hohe Schuld bei den Studierenden. Im 6. Semester hat man im
290 Laufe des Studium viele Personen kennen gelernt. Wie haben sehr viele professionelle
291 Lehrkräfte an der Fachhochschule und einer sollte wohl dabei sein, zu dem man schon ein
292 gewisses Vertrauen gefunden hat und bei dem man sich auch traut zu fragen. Und man
293 kann ja wirklich jeden Fragen. Wenn man sich nicht für Rechnernetze interessiert, aber das
294 Gefühl hat, dass der jeweilige Kollege nett ist, dann kann man ja hingehen und einfach mal
295 nach dem Prozess hinter der Bachelorarbeit fragen - dann wird man sicherlich auch
296 Antworten bekommen.

297

298 Was wir ja nicht wollen - und da glaube ich für mehrere Professoren zu sprechen - , ist das
299 ganze hier noch weiter zu "verschulen". Dafür gibt es ja letzten Endes auch das Curriculum.
300 Da stehen die Module drinnen, da steht die Abschlussarbeit im 6. Semester drinnen. Man
301 kann auch in der Prüfungsordnung lesen, was eigentlich alles zulässig ist - es liegt nur
302 keiner.

303

304 **Genau da ist das Problem, wer und ob jemand die Schuld dafür trägt, ist ja erstmal zu**
305 **vernachlässigen. Es existieren ja auf jeden Fall große Probleme bei diesem Prozess.**
306 **Meine Hoffnung mit der Applikation ist es zum einen, auch die Aufmerksamkeit und**
307 **den Fokus frühzeitig auf Dinge zu lenken, die immer wieder falsch gemacht werden,**
308 **so das dieser Prozess bei den Studierenden vielleicht früher ausgelöst wird.**

309

310 Die Idee von dieser App finde ich grundsätzlich nicht schlecht - das es erstmal überhaupt ein
311 neues Medium ist oder ein neuer Kanal um Informationen zu erhalten. Scheinbar reicht es
312 nicht, diese Informationen auf die Website zu schreiben.

313

314

315 **Viele Studierende wissen, dass diese Infomaterialien existieren aber teilweise finden**
316 **sie sie nicht oder suchen nicht danach. Ein Ansatz wäre es beispielsweise, diese**
317 **Informationen mit der Fachhochschule zusammen aufzubereiten oder durch**
318 **Verlinkungen zugänglicher zu machen.**

319
320 Ja. Das eine ist das organisatorische, was aus meiner Sicht vielen Studierenden fehlt. Es
321 gibt ja auch die Möglichkeit sich von externe Unternehmen die AUfgabenstellungen zu
322 holen. Das machen aus meiner Sicht erstaunlich wenige, obwohl es sehr viele Unternehmen
323 gibt, die händeringend Informatikstudenten suchen. Was man überlegen könnte ist, ob man
324 als zusätzlicher Kanal mit einer solchen App auch konkrete (externe) Abschlussarbeiten
325 angeben könnte. Oder ein Pool von möglichen Themen "triggert vielleicht ein paar
326 Studierende an". Das gibt es von seitens unserer Hochschule nicht wirklich. Zumindest nicht
327 koordiniert. Professor Kratzke hat auf seiner Webseite auch Themen, die er gerne bearbeitet
328 hätte aber ansonsten hat das glaube ich keiner. Ich hab das auch mal gemacht, aber ich
329 glaube nur ein Student hat sich daraufhin gemeldet - also auch relativ wenig Reaktion
330 darauf. Könnte man ja drüber nachdenken, ob man von der Hochschule aus so einen
331 Themenpool angeht, aber da braucht man natürlich auch jemanden, der den pflegt - nicht
332 nur von externen sondern auch von internen Themen. Jeder ist natürlich bereit, seine
333 eigenen Themen zu pflegen, aber jemand muss natürlich auch das gesamtsystem pflegen.
334

335 Ich hab da ja immer Angst. Man baut eine App, die dann für ein Jahr "cool" ist und ab dann
336 eigentlich durch neuen Kontent lebt. Wo kommt dieser weitere Kontent her und wie könnte
337 der aussehen?

338
339 **Wie würden Sie Ihre Einstellung zu einer solchen App beschreiben?**

340
341 (lacht) Das hängt vom Endergebnis der App ab, aber grundsätzlich würde ich das begrüßen,
342 dass ein neuer Kanal existiert, über den sich die Studierenden informieren können.

343
344 Was ich begrüßen würde ist, wenn es eine Plattform geben würde, wo ich jedes Semester 3-
345 4 Themen eintragen könnte. Das sollte nicht zu aufwändig sein. Irgendwie nur so Vorschlag
346 für eine Überschrift und eine kurze Beschreibung der Idee des Themas. Da ist mir ganz egal
347 wie das Technisch gelöst ist - idealerweise natürlich was browserbasiertes. Dann erscheint
348 das was ich da eingetragen habe, in dieser App die für Studierende gedacht ist.

349 Das würde natürlich nicht funktionieren, wenn da nur meine Vorschläge drinnen stehen,
350 sondern natürlich müssten da auch Vorschläge von anderen drinnen stehen. Sowohl intern
351 und extern. Für externe Unternehmen gibt es das ja schon. Sowas wie
352 Abschlussarbeiten.de, wo die Unternehmen Praktika oder Abschlussarbeiten inserieren.
353 Das ist natürlich deutschlandweit. Wenn man dann den Filter auf Lübeck stellt ist ja nichts
354 mehr da. Wenn man dann doch nach Informatik filtert ist noch weniger da. Oder es gibt nur
355 ein prominentes Unternehmen in Lübeck.

356
357 **Abschluss des Interviews**

Interview II

1
2 **Frage:** Ihr Fachgebiet wird auf der Internetseite der Fachhochschule Lübeck als
3 "Rechnernetze, Web-Technologien" angegeben. Wenn ein Studierender Sie als
4 Betreuer für die Bachelorarbeit wünscht und mit einem Themenvorschlag zu Ihnen
5 kommt, der nicht Ihrem Fachgebiet entspricht, gibt es dann Themen bei denen Sie
6 den Studierenden an Ihre Kollegen weiterleiten?

7
8 **Antwort:**

9 Üblicherweise sind die Themen liegen die Themen schon in diesen Bereichen, aber wenn
10 Studierende selbst ein Thema vorschlagen und von mir betreut haben wollen , dann ist das
11 schon mal etwas weiter ausgelegt. Aber sonst in manchen anderen Fällen, wenn ich denke
12 das ein Kollege oder eine Kollegin besser geeignet ist, werden die Studierenden auch
13 weitergeschickt.

14
15 Bei der Zweitbetreuung habe ich das bisher recht Großzügig betrachtet. Das läuft ja oft auch
16 etwas unterschiedlich, wenn zum Beispiel der Studierende schon einen
17 Zweitbetreuerwunsch äußert.Der Zweitbetreuer beschäftigt sich ja nicht während der
18 Arbeitszeit mit der Abschlussarbeit, sondern erst zum Ende der Arbeit, richtung Kolloquium.
19 Es ist aus Zeitgründen nicht möglich das zwei Personen kontinuierlich die Arbeit betreuen.

20
21 Vielleicht könnten Sie auch mal darüber nachdenken, ob die Applikation nur für
22 Bachelorarbeiten ausgelegt ist oder nicht vielleicht auch für Masterarbeiten.
23 Für Masterarbeiten haben wir hier ja 6 Monate Zeit und manche Themen müsste man dann
24 noch vertiefter betrachten.

25
26 Was ich eigentlich sagen wollte. Es müsste bei dem Ablauf der App ja so sein, dass der
27 Studierende den Prozess selbst steuert. Ich mache das Grundsätzlich so, das ich am
28 Anfang erkläre, dass es um eine selbständige wissenschaftliche Arbeit geht und ich den
29 Studierenden nicht alles hinterhertragen möchte. Die App müsste auch so "programmiert"
30 sein, das die App diesen Prozess auch unterstützt. Das ist bei den deutschen Studierenden
31 nicht so das Problem, während chinesische Studierende sich an das selbstständige Arbeiten
32 erst gewöhnen müssen.

33
34 Haben Sie schon eine Idee entwickelt?

35
36 **Ja, die Idee bisher liegt so vor, dass Studierende selbständig Meilensteine, mit
37 Unteraufgaben planen, erstellen und verändern müssen und seitens der App
38 Erinnerungen ausgelöst werden.**

39
40 Beim Ablauf habe ich das oft so gemacht:
41 Man macht gewisse Zwischenversionen und ich kommentiere dann was besser/anders
42 gemacht werden muss. Und dann erwarte ich beim nächsten Feedback, dass diese
43 Kommentare auch umgesetzt wurden und neuer Inhalt zum weiteren Kommentieren
44 enthalten ist. Ich möchte nicht zweimal die gleichen Kommentare geben müssen.

45

46 Die Arbeit wird im Laufe der Zeit immer Umfangreicher und man fragt sich als Dozent dann
47 irgendwann "Was soll ich denn nun genau lesen".
48 Es wäre hilfreich wenn dann in der Mail steht, oder markiert ist, was genau neu ist oder
49 nochmal gelesen werden sollte, damit man nicht erst suchen muss. Bei einigen Texten weiß
50 man sonst nicht ob es nur um Kleinigkeiten geht oder sich das überhaupt lohnt.
51
52 Eine weitere Sache ist, die Sache mit den 3 Monaten. Wenn man angemeldet ist, dann gilt
53 folgendes: Die Zeit läuft für 3 Monate und man hat somit ein gewisses Enddatum, was nicht
54 verlängert werden kann, außer man wird Krank oder es ist eine externe Arbeit und die Firma
55 kann die nötige Hardware nicht liefern, die man benötigt. Bei Gründen die man nicht
56 vertreten kann, kann verlängert werden, aus Selbstverschulden jedoch nicht, z.B bei zu
57 optimistischen Zeitplan. Aber man hat ja noch vor der offiziellen Anmeldung die Möglichkeit
58 sich vorzubereiten.
59 Für die App wäre es also nicht schlecht, wenn man den Hinweis auf das eben gesagte
60 geben würde. Es kam schon oft vor, dass die Studierenden noch am letzten Tag an der
61 Bachelorarbeit gearbeitet haben.
62
63 Noch ein wichtiger Punkt ist: Es ist ein wesentlicher Unterschied, ob man allgemein
64 schmierpapiermäßig etwas aufschreibt oder ob es ein guter Text (gutes Deutsch, klar
65 formuliert, sauber zitiert, formatierte Bilder) ist. Das wirkliche Aufschreiben kostet immer viel
66 Zeit. Es ist ganz wichtig darauf hinzuweisen. Ein wichtiger Tipp ist hier (verweis auf
67 Studentische Vorlagen zum wissenschaftlichen Arbeiten), von Anfang an nach diesen
68 Vorlagen zu arbeiten. Nicht erst in vielen verschiedenen Dokumenten Sachen festhalten
69 oder provisorischen Inhalt verfassen. Bei relevanten Literaturstellen sollten gleich alle
70 Quellen sauber angegeben werden und auch von Anfang an überlegen, wie man was
71 dokumentiert. Vorher nur in Stichpunkten arbeiten und wenn klar ist, dass die Struktur so
72 bleibt, dann ausformulieren.
73
74 Man muss halt überlegen wie man das mit der App macht. Bringt man die Studierenden
75 schon zu Anfang in eine gewisse Reihenfolge? Schon am Anfang mal überlegen - komplette
76 Arbeit? Ich würde es nicht empfehlen, Kapitel 1 fertig zu stellen und dahinter steht noch gar
77 nichts.
78
79 Und sonst die Frage mit der Gliederung. Ich habe in der Vorlage eine Art
80 Standardgliederung, die meines Erachtens nach auf möglichst viele Bachelorarbeiten
81 passen kann. Es gibt unterschiedliche Themen. Ist es ein Programmierthema, wo viel
82 umgesetzt werden muss, dann ist der Text vielleicht nicht so lang. Oder ist es ein
83 vergleichendes Thema und dann ist natürlich mehr Text und weniger praktische arbeit da?
84
85 (Persönliche Einschätzung zu typischen Informatikthemen) Ein Tipp zur Gliederung wäre
86 noch, das man klar erkennen kann was der Student eigentlich gemacht hat und was ist
87 allgemeine Kenntnis aus der Literatur.
88 Häufig habe ich so Themenstellung die heißen so "Konzeption und prototypische
89 Implementierung von ...". Da kann man dann am Anfang darstellen, wie die
90 Aufgabenstellungen und was die Anforderungen an das Ziel sind. Dann guckt man in der
91 Literatur was andere schon gemacht haben und was man davon erben kann. Dann macht
92 man ein Konzept Kapitel und ein Umsetzungskapitel. Und das es "prototypische
93 Entwicklung" heißt, hat den Hintergrund, dass man sich da selber nicht so in Schwierigkeiten

94 bringt. Wenn man schwerwiegende Programmierfehler hat, sieht das sonst vielleicht etwas
95 blöd aus. Dann kann man immernoch sagen, das die Umsetzung eher eine Fleißarbeit ist,
96 das Konzeptionelle aber enthalten ist. Das ist aber eher eine persönliche Einschätzung zu
97 typischen Informatikthemen. Aber wenn das andere Themen z.B. aus IDT sind, dann würden
98 die das natürlich anders machen.
99

100 Zu den Programmierthemen hat man typischerweise die Anforderungen, erstmal das Thema
101 "Anforderungen" klar aufschreiben, dazu dann MUSS/KANN-Kriterien abwägen und mit
102 Literaturwerken vergleichen. Dann das Konzept ausarbeiten und die folgende Umsetzung
103 erarbeiten. Bei einer professionellen Softwareentwicklung würde man ja nicht einfach was
104 "loshacken". Das wäre ja nicht die Professionelle Herangehensweise. Sondern erst muss
105 man sich ja was konzeptionell überlegen und das bei der Umsetzung ggf. sonst nur zum Teil
106 umsetzen.
107

108 (Zusammenfassung der Antwort des Interviewenden)

109
110 Ein bisschen wollte ich noch auf das Thema mit einem Firmenbetreuer eingehen. Wir haben
111 häufig den Fall das die Studierenden eine Bachelorarbeit in einer Firma erstellen. Die Firma
112 hat andere Interessen als die Fachhochschule. Die Firma hat oftmals die Situation, ein
113 konkretes Problem zu lösen wo vielleicht schon ein spezielles Konzept für vorgesehen ist
114 und somit den Fokus eher auf die praktische Umsetzung legen. Das ist aber nicht ganz der
115 Anspruch an die Bachelorarbeit. Bei der Bachelorarbeit ist es schon wichtig, dass man da
116 verschiedene Lösungsmöglichkeiten anschaut und die sinnvoll miteinander vergleicht. Das
117 ist aber keine Programmieraufgabe wo das Konzept schon vorhanden ist und Sie es nur
118 umsetzen sollen. Das wäre zu wenig. Eine richtige konzeptionelle Arbeit wird hier von den
119 Studierenden erwartet - soll heißen, THEORIE + Praxis. Da steht unter Umständen der
120 Fokus auf den Frühzeitigen Dialog zwischen Betreuer der Firma und Betreuer der
121 Fachhochschule um frühzeitig unterschiedliche Erwartungen zu identifizieren.
122

123 (Schilderung der Erwartung der Studierenden an das Thema Bachelorarbeit, Bezug 124 auf Übergang von bisherigem Studiumsverlauf zu Bachelorarbeit. Studierende haben 125 oft Unsicherheiten überhaupt zu verstehen, was eine Bachelorarbeit ist und was 126 erwartete wird und wo die Freiheiten liegen)

127
128 **Antwort:**
129 Ganz andere Situation als in den Studienmodulen, da in den Modulen oft der Gedanke
130 existiert, das der Code sowieso nicht weiterverwendet wird, sondern nur Lerngegenstand ist.
131 Gerade auch in den Softwaretechnikprojekten sieht das anders aus. Da geht es auch darum
132 den Code den Nachfolgern sauber zu hinterlegen und zu pflegen. Und Es ist im Allgemeinen
133 viel offener als in den Modulen.
134

135 (Unterschied nochmal zusammenfassen)

136 **(Die Brücke von Studium zur Bachelorarbeit aus Sicht der Studierenden schildern -**
137 **Problem: Übergang zu Hart?)**

138

139 Sie würden lieber noch ein anderes Modul dazwischen haben, in dem Sie schon mal viel
140 offener in Berührung zu solchen Themenstellungen kommen oder vielleicht sogar das
141 Softwaretechnikprojekt vorziehen?! Wir haben derzeit das Problem, dass wie nur 6
142 Semester haben. Früher mit dem Diplom, mit den 4-5 Jahren konnte man deutlich mehr
143 machen. Es ist jetzt nicht so einfach ein Auslandssemester zu integrieren. Da soll sich jetzt
144 einiges tun, dass man ein sogenanntes Mobilitätsfenster reinbringt, und somit in einem
145 Semester ausschließlich Wahlpflichtfächer hat und so leichter mal ins Ausland gehen kann.

146

147 *Ich habe selbst mal in meinem Softwaretechnikprojekt mal gefragt ob Interesse besteht,
148 Auslandssemester innerhalb des Bachelorstudiengangs zugänglicher zu machen und so
149 Erfahrungen (besonders bezüglich eigenständiges Arbeiten) zu sammeln. Von den 10
150 Personen hat nur einer sich gemeldet und ausgesagt, dass er nicht so viel Wert drauf legt.
151 Die anderen haben gar nichts gesagt. Man weiß nicht ganz genau woran das liegt- Vielleicht
152 an den organisatorischen Schwierigkeiten? Oder die Finanzierung? Ob die Studierenden nur
153 schnell das Studium abschließen wollen und Geld verdienen wollen? Oder es einfach nicht
154 nötig ist sowas Vorzuweisen, da die Arbeitslage sehr gut ist?(Vorteil derzeit nicht erkennbar)*

155

156 Man kann die Bachelorarbeiten in Firmen machen oder man kann sie intern in der
157 Fachhochschule machen. Früher war es sehr üblich extern zu machen. Jetzt wo die
158 Fachhochschule viel Forschung vor ort hat, ist es sehr üblich interne Bachelorarbeiten zu
159 machen.

160

161 **(Fokus wird auf Applikation gelegt, was die Aufgabe der applikation sein soll.
162 Fokussieren der eigentlichen Idee und die Idee das Bachelor Seminar vielleicht
163 vorzuziehen)**

164

165 Ich finde es nicht so gut, dass die Leistungspunkte so 'kuddelmuddel' sind (Hier sind 8 CP,
166 da sind 5 CP), das gibt immer Probleme mit diesen Anerkennungsverfahren. In anderen
167 Studiengängen, z.B. ITD hat jedes Modul grundsätzlich 5 Leistungspunkte oder bei dem
168 großen Projekt ein Vielfaches davon (10 CP). Das ist so ein bisschen einfacher als wenn
169 jedes Modul seine Leistungspunkte hat. Hier (für Softwaretechnik) wurde auch schon einiges
170 herumgeschoben. Es ist vorgesehen, dass man halt ungefähr auf 30 CP pro Semester
171 kommt. Wenn man das Bachelorseminar in das 5. Semester zieht, haben wir im letzten
172 Semester nur noch 25 CP. Dann müssten wir wieder ein anderes Fach einbringen und wenn
173 das ein Fach mit inhalten zu z.B. formalen Sprachen wäre, dann wären diejenigen
174 benachteiligt, die eine Bachelorarbeit in diesem Gebiet schreiben wollen.

175

176 *(Abschweifen des eigentlichen Themas)*

177

178

179 **Frage: Was halten Sie von der Idee der App, haben Sie Anregungen, sehen Sie
180 Chancen oder sogar Risiken?**

181

182 **Antwort:**

183 Günstig finde ich es, dass die Personen an den Zeitplan erinnert werden. Denn mit dem
184 Zeitmanagement hapert es manchmal ein bisschen. Für manche sind es vielleicht unnötige
185 Erinnerungen aber für andere ist das denke ich nicht verkehrt.

186

187 Es bleibt die Gefahr, dass wenn die App einen Fortschrittsbalken anzeigt und der bei 100%
188 liegt, die App ja nicht den tatsächlichen Fortschritt prüfen kann und man (der Studierende)
189 sich somit denkt, dass "alles toll" ist, obwohl man zum Beispiel alles vollgeschrieben hat,
190 dass jedoch (qualitativ) nicht so toll ist. Vielleicht könnte man ja auch inhaltliche Fragen
191 einbringen. Zum Beispiel wenn man Grafiken hat (Entsprechen die Grafiken einem
192 Standard). Diagramme wurden schon in eigenen Notationen geschrieben, was zu Fehlern
193 und missverständnissen führt - ein Zustandsdiagramm, welches fehlerhafte Wege
194 zugelassen hat. Wenn man da eine vernünftige Notation eingehalten hätte wäre es deutlich
195 besser gewesen.

196

197 Man könnte in die App vielleicht eine Checkliste integrieren, die diese Fragen an bestimmten
198 Stellen abfragt. Eine Mögliche Frage wäre, ob bekannte Notationen verwendet wurden oder
199 "Keine Abbildung soll als selbsterklärend betrachtet werden". Das man so ein paar Punkte
200 als Checkliste als Checkliste herausnimmt. Oder bei jeder Quelle alle Angaben aufgeführt.
201 Wenn es eh schon passt und das abgefragt wird, dann macht das ja nichts.

202 Es wäre natürlich auch nicht gut, wenn diese Abfragen erst kurz vor der Abgabe kommen
203 und der Studierende dann merkt "Oh, Mist! Ist ja gar nicht..." und alle 20 Grafiken neu
204 überarbeiten muss. Und natürlich sollte diese Abfrage auch nicht dazu führen, dass die
205 Studierenden "wie bei Softwareinstallationen" sie einfach mit dem immer wieder
206 aufeinanderfolgenden Bestätigen wegklicken. Also sollte es auch eher eine übersichtlich
207 ausgelegte Auswahl an Fragen bleiben.

208

209 Ich denke, da müssen Sie am besten Überlegen in welcher Reihenfolge das ganze kommt.
210 Wenn die erste Grafik kommt, dann sollte die Frage auftreten.

Interview III

1
2 **Frage: Ihr Fachgebiet wird auf der Internetseite der Fachhochschule Lübeck als**
3 **“Informatik” angegeben.**

4
5 Ja, das ich auf der Webseite der Fachhochschule nichts weiter angegeben habe liegt daran,
6 dass ich da zwei Links habe, wo ich die Informationen regelmäßig pflege. Der Vorteil liegt
7 dabei, dass ich diese Informationen selbst pflegen kann. An der Webseite der
8 Fachhochschule kann ich nichts ändern. Deswegen habe ich damals gebeten nur die Links
9 auf meine CoSA Profilseite und noch ein weiteren angibt. Die aktuellen Sachen passieren
10 auf diesen externen Seiten. Ich will nicht ständig dafür sorgen, das irgendwas aktualisiert
11 wird. Wenn Sie da auf der CoSA Seite schauen, werden Sie mehr Informationen finden.

12
13 **Wenn ein Studierender Sie als Betreuer für die Bachelorarbeit wünscht und mit einem**
14 **Themenvorschlag zu Ihnen kommt, der nicht Ihrem Fachgebiet entspricht, gibt es**
15 **dann Themen bei denen Sie den Studierenden an Ihre Kollegen weiterleiten?**

16
17 Ja, ich betreue relativ breit gestreut. Ich unterrichte ja auch an mehreren Studiengängen. Ich
18 unterrichte bei ITD genauso viel wie in Informatik zum Beispiel. Dadurch bin ich gewohnt
19 Arbeiten zu betreuen, die eigentlich gar nicht meinen fachlichen Kern betreffen.
20 Aber es ist so. Bei Bachelorarbeiten gibt es eben ganz schön viele Dinge, die eben einfach
21 “bachelorarbeitstypisch” sind. Die kann man dann beurteilen, auch wenn es nicht exakt das
22 eigene Fachgebiet ist. Bei einer Promotion muss dass das eigene Forschungsgebiet sein.
23 An der Fachhochschule arbeiten wir denke ich auf einem Level bei dem 70 bis 80 Prozent
24 der Professoren das Wissen haben sollten. Wir sind ja nicht alle als Lehrstuhlinhaber eines
25 Spezialgebiets berufen, sondern so ein bisschen sind wir schon generalisten. Insofern
26 kommt es auch vor, dass ich Studentin an andere Professoren verweise. Der Grund ist aber
27 meistens, das ich nicht noch mehr Leute Betreuen kann, aber es kam auch schon einmal
28 vor, dass es aus fachlichen Gründen war. Das war dann ein Bereich, wo ich mich wirklich
29 nicht so gut ausgekannt habe und dann Kollege X oder Kollege Y empfohlen habe. Das
30 kommt vor. Aber dafür gibt es ja auch dieses Vorgespräche.

31
32 Bei mir läuft es normalerweise so, dass Leute mich dann ansprechen, weil sie gerne die
33 Bachelorarbeit bei mir schreiben wollen und dann treffen wir uns, um
34 interessenschwerpunkte zu besprechen und am Ende des Gesprächs ist dann für beide
35 Seiten klar, ob das sinnvoll ist oder ob das Thema vielleicht jemand anders betreuen sollte.

36
37 **Lassen sich Bachelorarbeiten grob nach Typen kategorisieren? Was sind da Ihre**
38 **Erfahrungen? Stichwort - “entwicklend” oder “vergleichend”.**

39
40 Ja. Man muss da auch nochmal unterscheiden zwischen interne und externe Arbeiten. Das
41 kann auch noch große Unterschiede machen.

42
43

44 Es gibt die entwickelnde externe Arbeit. Das ist ganz klar eine Kategorie, welche bedeutet
45 dass die Person die eine Arbeit schreibt, in einer Firma sitzt oder zumindest für eine Firma
46 primär arbeitet. Das bedeutet auch, dass das eigentliche Ziel, das dabei rauskommt von
47 dem Unternehmen gesteckt wird und in dem Fall, sehe ich mich als Betreuer eher in der
48 Position eines Projektleiters oder Projektmanagers. Ich mische mich da dann sehr selten in
49 die inhaltlichen Ziele ein, solange ich das Gefühl habe, dass die Arbeit das Niveau einer
50 Bachelorarbeit erfüllt.

51 Also wenn unsere Leute dann als Arbeitssklaven eingesetzt werden, muss ich das
52 verhindern. Aber wenn die dann genau etwas brauchen, werde ich nicht nein sagen. Also in
53 diesem Fall ist meine Rolle eine ganz spezielle Rolle, weil man dann mit der Firma klären
54 muss, welche Punkte in die Arbeit reinkommen weil es eine Bachelorarbeit ist. Das sind oft
55 Dinge, die der Firma eigentlich völlig egal sind. Also ich möchte dann natürlich trotzdem in
56 der Arbeit sehen, warum dieser (ein bestimmter) Weg gegangen wurde und welche
57 alternativen es gegeben hätte. Das ist den Firmen meistens ganz egal, weil sie das eine
58 Ergebnis haben soll - und das soll dann auch funktionieren. Aber die Leute die wir hier in die
59 Welt entlassen, sollen schon in der Lage sein zu entscheiden, zu begründen, abzuwägen
60 und "ingenieurmäßig" zu denken. Da sollten sie auf jeden Fall sagen können, warum sie
61 sich für zum Beispiel genau ein Framework entschieden haben. Das sind so Aspekte, die
62 den Firmen oft egal sind und nicht im Vordergrund stehen.

63 Da ist das Besondere der Arbeit dann, das man versuchen muss was in die Arbeit
64 hineinzubringen, was keine Praktikantenarbeit ist, sondern den Rahmen einer
65 Bachelorarbeit erreicht. Das ist also insgesamt eine spezielle Sache, die auch nicht mit allen
66 Firmen gleich läuft aber im Mittel würde ich sagen, sind meine Erfahrungen deutlich positiv.
67 Die meisten Firmen sehen das ein, dass die Arbeit nicht identisch mit dem Produkt ist, was
68 letzten Endes entsteht.

69
70 Dann gibt es natürlich die interne Bachelorarbeit, welche bei mir den Schwerpunkt meistens
71 im Softwaretechnik Bereich liegt. Ich erwarte in diesem Fall, das die Leute mit der Methodik
72 arbeiten, die sie hier (an der Fachhochschule Lübeck) gelernt haben. Da kommt dann ein
73 Produkt heraus, von dem man die Eigenschaften kennt, wo man sorgfältig geplant hat und
74 die Anforderungen erhoben hat. Es ist dann wirklich ein Softwareentwicklungsprozess, den
75 die Leute da dann einmal durchlaufen und mir geht es dann meistens nicht Primär um die
76 Qualität des Ergebnisses, sondern um den Weg.

77 Da liegt dann der Unterschied zu der entwicklungsarbeit in Firmen. Da muss dann ein
78 bestimmtes Ergebnis herauskommen und man versucht dann unter den bestimmten
79 Randbedingungen, das dann hinzubekommen. Wenn es intern passiert, habe ich das
80 Gefühl, das die Leute dann mehr das Interesse daran haben, was den fachlichen Kern
81 ausmacht und wenn man dann am Ende feststellt, dass man das Ziel nicht vollständig
82 erreicht hat und dann erklären kann warum das passiert ist, dann kann das trotzdem eine
83 sehr gute Arbeit sein. Der Fokus liegt also ganz klar auf der Entwicklung.

84
85 Dann gibt es natürlich auch die vergleichenden Arbeiten. Beispielsweise vier prominente
86 Webframeworks, worüber sich die Gemeinden streiten, welches nun das beste ist. Eben
87 eine vergleichende Studie zu führen - sowas kann man natürlich auch machen. Das betreue
88 ich aber eher selten, was aber daran liegen könnte, dass die Leute die mich fragen offenbar
89 an solchen Sachen nicht so ein Interesse haben.

90 Was ich häufig habe - besonder im Bereich Künstliche Intelligenz -, sind so eine Art
91 Machbarkeitsstudien. Das wäre noch ein dritter Typus. Zum Beispiel, wie gut die
92 Gesichtserkennungsdienste von Amazon, Microsoft oder andere funktionieren.
93 Sind die schon so gut, dass man mit einer einfachen Drohnenkamera auf 10 Meter
94 Entfernung Personen erkennen kann? Das war tatsächlich mal der Inhalt einer Arbeit.
95 Auch hier war es nicht Ziel der Arbeit, ein fertiges Produkt zu entwickeln. Es war im Kern
96 auch keine vergleichende Studie, sondern es ist mehr die Frage "Ist die Technologie so weit
97 entwickelt, dass man sowsas daraus ausbauen könnte?" Das Ergebnis war eben nur ein
98 Prototyp, mit dem dann verschiedene Untersuchungen gemacht wurden. Das wäre dann
99 auch so ein Typus von arbeit - eben die technische Machbarkeit -, die dann in technische
100 Grenzbereiche vordringt, wo der Betreuer vorher meisten auch nicht weiß, ob das
101 funktionieren wird oder ob man auf Hindernisse stößt. Und in diesem Fall geht es dann ja
102 wirklich weniger darum ein Produkt als Ziel zu erreichen, sonder darum eine Frage zu
103 beantworten. Man würde am Anfang vielleicht Anforderungen aufschreiben, aber mehr mit
104 dem Hintergrund, ob die sich erfüllen lassen oder nicht. Dadurch kann das eine sehr
105 anspruchsvolle Studie werden, weil man da natürlich auch sehr viel testen muss. Man muss
106 ja sicherstellen, dass es nicht deswegen nicht funktioniert, weil man den Prototyp falsch
107 gebaut hat. Das wäre also auf jeden Fall ein Typus von Arbeit, den es gibt.
108

109 Im Studiengang ITD gibt es da ja noch ganz andere Dinge. Sowas wie eine Studie über die
110 Bedienbarkeit, Usability oder User Experience. Da kommt dann die Frage auch auf, ob man
111 etwas entwickeln kann, dass bisher schon existiert aber "nicht spannend" genug für die
112 Leute wirkt. Da kann durchaus auch viel Technik für ins Spiel kommen. Da wäre das Ziel
113 auch eher eine Entwicklung oder manchmal auch eine Machbarkeitsstudie, aber mit einem
114 nicht nur technischen Ziel. Ich vermute aber mal, das Ihr Fokus eher auf Informatik Arbeiten
115 beschränkt ist?

116
117 **Genau, aber es ist im Sinne der Erweiterbarkeit ja durchaus interessant. Diese
118 Studien, die Sie angesprochen haben (Machbarkeitsstudien, etc.) gehen ja schon eher
119 in eine forschende Richtung.**

120
121 (zögern) Forschend wäre vielleicht etwas übertrieben. Ich glaube Machbarkeit trifft es am
122 besten. Da gibt es wenig zu erforschen, sondern man guckt ja nur ob etwas funktioniert.
123 Aus meiner Sicht ist das eher Entwicklung. Entwicklung heißt ja nicht, das man wie ein
124 Handwerker nach einem bekannten Rezept etwas zusammenbaut, sondern eher das Rezept
125 erstmal zu finden. Von Firmen wird es immer gerne euphemistisch als Forschung
126 bezeichnet. Sobald die was machen, wo nicht jeder sofort sagen kann ob es geht, nennen
127 die es Forschung. (lacht) Aber es ist eigentlich nur etwas mehr als Handwerk.
128

129 Unsere Absolventen sollen in der Lage sein, für Firmen zu überprüfen ob es zum Beispiel
130 Möglich ist einen bestimmten Dienst mit einer Sprachsteuerung auszustatten. Das heißt
131 aber nicht gleich das man einen Forscher braucht, (lacht) sondern dann braucht man einen
132 Ingenieur, der mit Technologien umgehen kann und weiß wie man sowsas testet und
133 ausprobiert. Und wenn er dann herausfindet, das es funktioniert, dann ist das ja kein
134 Forschungsergebnis, sondern dann hat er die Machbarkeit nachgewiesen. Eine
135 Bachelorarbeit sollte eigentlich auch nicht weiter gehen. Dafür gibt es ja dann die
136 Masterarbeit - um zu versuchen der Welt eine neue Erkenntnis zu eröffnen. Wenn man

137 methodisch weiterentwickelt, dann forscht man. Das ist aber nicht die Aufgabe einer
138 Bachelorarbeit.
139 Aber trotzdem geht es natürlich schon manchmal so ein bisschen in diese Richtung. Das
140 sind dann häufig Leute, die vorhaben noch einen Master zu belegen, weil man so die
141 Grenze zur Forschung tatsächlich ankratzen kann. Die Leute kriegen dann durch so ein
142 Gefühl dafür was man sich fragt und wo diese Fragen herkommen.
143
**144 Es ist schwierig das richtige Wort dafür zu finden. Forschung ist es natürlich nicht,
145 aber recherche wäre dann auch zu wenig.**
146
**147 Sie haben es bereits etwas ausgeführt, aber ich würde gerne nochmal fragen, was
148 Ihre Erwartungen an die Studierenden sind, die Sie betreuen?**
149
150 In der Bachelorarbeit soll man ja Eigenständigkeit nachweisen, deswegen finde ich es sehr
151 wichtig, dass man sich natürlich regelmäßig trifft und dann alles bespricht, aber im
152 Vordergrund sollten dann da die Arbeitsergebnisse stehen. Also aus meiner Sicht ist eine
153 Bachelorarbeit immernoch eine Art Lehrveranstaltung. Die Leute schreiben eine solche
154 ARbeit zum ersten Mal, also kann ich nicht davon ausgehen, dass sie alles wissen. Aber Sie
155 sollten bereits im Stande sein, eigene Zwischenergebnisse zu erarbeiten, so das wir dann
156 darüber diskutieren können und ich auf dieser Basis dann weiterhelfen kann. Das heißt,
157 dass das im Studium vorher vermittelte dann einfach schonmal da ist und die Leute dann
158 auch selber mal auf die Idee kommen irgendwo nachzuschlagen und sich Dinge anzulesen
159 und selbst merken, welche Dinge sie in Erfahrung bringen müssen. Da möchte man die
160 Studierenden dann nicht jedes Mal darauf hinweisen, was sie machen sollen. Die Leute
161 sollten möglichst herkommen und schon Ideen im Kopf haben und sich Rückkopplung
162 einholen - bin ich auf dem richtigen Weg? So stelle ich mir das vor und in der Regel läuft das
163 auch so.
164
**165 Also sollte es so sein, das man sich trifft um eher die Qualität der Ergebnisse zu
166 reflektieren und so zu überlegen wo die geleistete Arbeit zu wenig oder vielleicht
167 sogar zu viel war.**
168
169 Genau, auch das kann sehr wichtig sein. Bei der Bachelorarbeit verrennen sich manchmal
170 Studierende in irgendwelchen Seitenzweigen die sie vielleicht ganz spannend finden und
171 dann kommt die Gefahr auf, das große Ziel aus den Augen zu verlieren. Dann überlegt man
172 da, wie man die Kurve bekommen kann. Solche Dinge wären die Dinge über die man (bei
173 einem Treffen) am liebsten spricht.
174
**175 Was ich von vielen Studierenden gehört habe ist, dass sie zu Anfang extrem viel Zeit
176 in die Grundlagen gesteckt haben und durch Feedback auch darauf hingewiesen
177 worden, das es vielleicht zu viel ist.**
178
179 Das ist keine Seminararbeit, wo man wochenlang Grundlagen darstellt. Die Grundlagen
180 stellt man ja nicht dar, um zu zeigen dass man die Grundlagen beherrscht, sondern
181 Grundlagen sollen in der Bachelorarbeit ja dargestellt werden um zu sehen worauf die Arbeit
182 basiert. Ich empfehle oft, dass man sich für das Grundlagenkapitel erstmal nur Skizzen
183 macht, zum Beispiel durch Stichwortartige Inhalte. Richtig Schreiben kann man erst, wenn

184 man den Kern der Arbeit im Griff hat, weil man erst dann weiß was man in den Grundlagen
185 erklären muss und was eben nicht. Das ist auch so ein Punkt.
186
187 **Was ich auch wahrgenommen habe, ist dass viele Studierende die Arbeit sequenziell**
188 **angehen, obwohl sogar auch im Bachelorseminar darauf hingewiesen wurde, dies**
189 **nicht zu tun. Da wären wir dann schon bei dem nächsten Thema. Diese Probleme, die**
190 **aufreten und vielleicht sogar immer wieder auftreten. Welche Problem treten denn**
191 **immer wieder auf?**

192
193 Es kommt häufig vor, das Leute Probleme mit dem Schreiben haben und das ist diesen
194 Leuten eben vorher nicht wirklich bewusst, weil in unserem Studium wenig vorkommt, wo
195 man lange Texte schreiben muss. Meist sind es eher kurz gefasste Dinge und das etwas
196 längeres vorkommt ist eher selten. Es fällt vielen Leuten schwer den Fokus beim Schreiben
197 zu behalten - die driften oft weg. Unter anderem war das auch der Anlass, weswegen wir im
198 5. Semester im Fach Intelligente Systeme angefangen haben diese Berichte einzuführen.
199 Das war nicht von ANfang an so. Ich glaube Sie mussten das auch schon machen oder?
200

201 **Ja**
202

203 Da merken dann alle bei der ersten Aufgabe einen Bericht zu schreiben, weil niemand
204 genau weiß was da jetzt eigentlich reingehört. Und bei der Bachelorarbeit stellt sich dann die
205 Fragen in einem ganz anderen Umfang - Was muss da eigentlich rein? Ich hoffe dadurch zu
206 erreichen, dass die Leute zumindest für einen Typus von Arbeit ein Gefühl dafür entwickeln,
207 was es Wert ist reingeschrieben zu werden und was nur "Füllmasse" ist.
208 Da passiert dann eben das, was Sie auch schon gesagt haben. Man spricht die Gliederung
209 ab und dann fangen die Leute meistens an von vorne zu schreiben, was oft keine gute Idee
210 ist. Es gibt einige die bekommen das hin, aber die haben wohl eine literarische Begabung.
211 (lacht) Denen floss das aus der Feder. Den meisten Informatikern fließt das eben nicht aus
212 der Feder und da empfehle ich dann eben, dass sie die Arbeit an allen Stellen wachsen
213 lassen. Man skizziert alles was man haben will, hat den großen Rahmen und fängt dann an
214 mehreren Stellen zu arbeiten. Im Grundlagenkapitel hält man erstmal nur fest, was da rein
215 muss und welche Punkte wichtig sein könnten. Dann fängt man eigentlich mit dem Teil an,
216 welcher einem am leichtesten fällt, nämlich zu beschreiben was man wirklich gemacht hat.
217 Und während man das beschreibt, fällt einem oft ein, was man jetzt gerne so sagen würde
218 und was man vorher eigentlich in den Grundlagen beschreiben muss. Der Inhalt entsteht
219 also so ein bisschen aus der Arbeit selbst. Ich glaube das es unmöglich ist, das linear
220 entstehen zu lassen, wenn man zum ersten Mal so eine Arbeit schreibt. Diesen großen Plan
221 und diese Weitsicht hat man einfach nicht. Mir selbst fällt es immer noch schwer.
222 Wann immer ich etwas abfasse mache ich das immer noch in diesem Stil. Ich überlege mir
223 zuerst, was ist denn das wichtigste überhaupt an dem ganzen Ding, danach wird mir dann
224 klar wie ich da heranführen muss. Von wo muss ich kommen? Was muss ich erklären, damit
225 ich den Leser dann da habe, wo ich ihn haben will, wenn der interessante Teil kommt.
226 Ich glaube das ist eine Sache, die heutzutage sowieso wichtig ist. Die Leute haben immer
227 weniger Geduld zu lesen. Wenn zehn Seiten Grundlagen kommen und es dann spannend
228 wird, ist es oft heikel und deswegen ist es wichtig so zu schreiben ,das der Leser von anfang
229 an das Gefühl hat, dass alles was kommt wichtig ist. Das ist sowieso wichtig. Diese
230 zielgerichtete Schreiben zu lernen. Das ist dann eben nicht dieser klassische Seminar Stil,
231 zu zeigen was man alles recherchiert hat und der Leser sich dann fragt, wozu er das alles

232 erzählt bekommt. Deswegen nenne ich das oft am Anfang als eine Option, einfach mal
233 darüber nachzudenken ob das so der richtige Weg ist und empfehle es das die Arbeit an
234 mehreren Stellen wächst. Die Einleitung denke ich, sollte man als allerletztes schreiben.
235 Die Arbeit soll am Ende eben aus einem Guss wirken und eben nicht, das sie sich im Laufe
236 der Zeit entwickelt hat. Am Ende weiß man, wie man die Einleitung verfassen muss, damit
237 sie zur Arbeit passt. Das sind Dinge, die man dann wirklich steuern kann. Damit kann man
238 wahrscheinlich schon mal viel Gutes tun.
239 Viele haben eben genau dieses Problem. Studierende arbeiten vor sich hin und bekommen
240 immer mehr Angst es dann zu schreiben, sie versuchen dann ganz am Anfang anzufangen,
241 wissen nicht was sie dann schreiben sollen und damit wird die Hürde dann immer höher.
242
243 Viele Leute fangen zu spät an. Das wäre auch noch so ein Punkt. Sie unterschätzen dann
244 die Zeit, die man wirklich braucht um wirklich das, was man nachher druckt und abgibt
245 entstehen zu lassen. Viele sehen dann im Vordergrund, beispielsweise bei der Entwicklung,
246 die Software entstehen zu lassen und beißen sich dann daran fest, um sich vielleicht sogar
247 selbst mit Arbeit zu beschäftigen, die einen davon abhält mit dem Schreiben anzufangen.
248 Wenn man es schafft, die Leute dazu zu bekommen von Anfang an das Gerüst der
249 gesamten Arbeit vor sich zu sehen, würde es glaube ich so manche "Fast-Katastrophen"
250 verhindern, wo Leute erst drei Wochen vor der Abgabe anfangen den Text zu schreiben.
251 Das kann eigentlich nicht gut gehen, wenn man nicht gerade eine Ausnahmefähigkeit hat,
252 was Schreiben angeht. Deswegen ist es glaube ich gut, in die Breite zu gehen. Sich klar
253 machen, dass es nicht nur um die Software geht die man entwickelt, sondern das es letzten
254 Endes hauptsächlich um die Darstellung geht und deswegen immer die Arbeit als Ganzes im
255 Auge behalten muss, ohne sich am Text fest zubeißen. Da kann man wahrscheinlich viel
256 Gutes mit tun, mit dem was Sie da vorhaben.
257
258 **Lustigerweise wird es ja alles immer wieder erwähnt. Es gibt das Bachelorseminar,**
259 **wo auch viele Tipps und Hinweise gegeben werden, aber trotzdem machen alle die**
260 **gleichen Fehler. Es ist sehr interessant, warum es denn nicht funktioniert. Vielleicht**
261 **kommen die Studierenden mit der eigenen Vorstellung in die Quere. Einleitung und**
262 **Grundteil hören sich an, wie etwas wo man ansetzen könnte.**
263
264 Ich glaube es ist sehr schwierig bei so etwas wie die Bachelorarbeit die Ratschläge in einem
265 Paket zu geben, und dann zu glauben, dass es die Leute verinnerlicht haben und so
266 anwenden werden. Ich glaube, dass so etwas nur funktionieren kann, wenn es
267 arbeitsbegleitend stattfindet. Zu Verstehen, warum etwas eine gute Idee ist, verinnerlichen
268 die Leute erst, wenn sie das Problem wirklich haben.
269 Das ist so ähnlich wie mit dem Projektmanagement. Da kann man sich natürlich hinstellen
270 und betonen, was die üblichen Fehler sind, aber ein ist klar. Wenn man die Leute auf ein
271 Projekt loslässt, werden sie alle genau diese Fehler machen. So funktionieren Menschen
272 irgendwie nicht. Es gibt Dinge, die sind nicht lehrbar indem man sie erklärt, sondern sie sind
273 nur lehrbar, indem man es die Leute machen lässt und sie dabei begleitet. Es gibt auch
274 andere Dinge, die man sehr gut so lehren kann. Man erklärt etwas und es macht klick weil
275 sie verstanden wurden. Aber es gibt einfach Dinge, die man wirklich erst versteht, in dem
276 Moment in dem man sie lernt und das ist in vielen Bereichen so.
277 Das kennt man auch aus dem Sport. Wenn man den Kindern im Training erklärt, was man
278 im Spiel machen soll, hören alle Kinder sehr aufmerksam zu aber im Spiel machen alle
279 genau den einen Fehler, wobei man vorher stundenlang erzählt hat, dass sie es auf keinen

280 Fall so machen sollen. Ich hab einige Jahre als Jugendtrainer im Fußball gearbeitet und
281 daher weiß ich das es die echt Situation ist, in der es plötzlich klick macht.
282 Ich glaube, dass es bei der Bachelorarbeit auch so ist. Erst wenn die Leute an ihrer eigenen
283 Bachelorarbeit sitzen und die Zeit tickt ist die Aufnahmebereitschaft da, zu erkennen wo da
284 der Sinn liegt. Deswegen glaube ich, es könnte eine tolle Sache sein wenn das mit ihrer
285 Arbeit gut läuft, da das Ziel ja ist, die Leute genau im Prozess zu begleiten. Das wird ja kein
286 Tutorial, wo man ein Video anguckt und dann werdet ihr alle tolle Arbeiten schreiben. Genau
287 das kann nicht funktionieren meiner Ansicht nach. Man braucht etwas, was einen genau in
288 der Situation begleitet, wo man das Problem hat. genau da muss man ansätzen und dann
289 glaube ich kann man eben solche Sachen verhindern.

290 Dann brauch man sich auch gar nicht mehr fragen, warum die Leute das nicht verstanden
291 haben, sondern ich glaube, dass viele der Probleme gar nicht mehr auftreten werden, wenn
292 man sie in dem Moment mit richtig begleitet. Das wäre so meine Hoffnung.

293

294 Deswegen würde ich gar nicht so viel darüber brüten woran das liegt, sondern ich glaube es
295 ist eher diese Trennung. Die Leute sind einfach noch nicht in dieser Situation und es ist wie
296 Leuten das Schwimmen zu erklären, bevor sie das erste Mal das Wasser berühren.

297 Da kann man auch nicht einfach sagen, das es einfach ist sich an der Oberfläche zu halten
298 und das man eigentlich kaum etwas dafür machen muss. Das verstehen die Leute alles,
299 aber in dem Moment wo sie im Wasser sind, ist die Situation eine völlig andere und die
300 fangen dann an herumhampeln und vergessen dabei völlig, dass man eigentlich gar nichts
301 machen muss. Deswegen braucht man eben die Person, die am Beckenrand steht und
302 guckt was da passiert.

303

304 **Also würden sie dieser Applikation, oder der Idee positiv gegenüberstehen? Oder als
305 sinnvoll ansehen?**

306

307 Die Idee ist eine sehr sinnvolle. Ob es dann im ersten Anlauf gleich funktioniert weiß man
308 nicht. Sie probieren es ja eben zum ersten mal. (lacht)

309 Manchmal scheitern Dinge ja auch an Kleinigkeiten. Aber als Ansatz überhaupt, klingen die
310 Dinge für mich sehr vielversprechender, als vorher die Einweisung zu machen und dann zu
311 hoffen, dass die Leute es richtig machen. Da sehe ich die Chancen auf jeden Fall hier viel
312 größer, das sowas (die Leute) voranbringen kann.

313 Ich sehe das als sinnvoll an. Zumindest so sinnvoll, das es sich lohnt, da mal ein versuch zu
314 machen.

315

316 **Das Problem ist ja in meinem Fall, das es natürlich auch meine Bachelorarbeit ist, und
317 ich ...**

318

319 Ja, sie haben natürlich die Probleme die jeder hat, und zusätzlich erwartet man Wunder
320 (lacht)

321

322 **Genau. Die Zeit reicht letzten Endes ja gar nicht aus um das Konzept ordentlich zu
323 auszuwerten und ob es funktioniert. Man müsste ja eigentlich mindestens eine ganze
324 Iteration erstmal ausprobieren, ob es überhaupt einen Effekt hat.**

325

326 **Haben Sie sonst noch Anmerkungen oder Ideen, die sie noch mitteilen wollen?**

327

328 Nö, vielleicht nur eine sehr profane Anmerkung. Wenn man schonmal so etwas macht, wäre
329 es vielleicht nicht verkehrt wenn man in das was letzten Endes das Tool wird, die Formalien
330 aufnimmt. Auch das steht überall und wird zigfach erklärt, aber auch das erkläre ich immer
331 nochmal den Bacheloranden - wie es mit den Formalien aussieht, wie es mit der Anmeldung
332 aussieht und wie es mit dem Kolloquium läuft. Es wäre schön, wenn man dann so ein Infoteil
333 mit drinnen hätte. Die Hoffnung ist natürlich, dass die Leute es am Ende annehmen und
334 nutzen wollen. Wir hätten dann eine Stelle, wo sie freiwillig hingucken. Wenn man jetzt sagt,
335 das steht doch in irgendwelchen Lernraum-Kursen, dann sagen alle dass sie das wissen,
336 das es irgendwo steht, aber (es ist zu aufwändig zu finden).
337 Wenn es aber jetzt eine Stelle gibt, von der man weiß, die Leute die Interesse haben,
338 werden sich diese Stelle angucken und da dann diese Sachen stehen, dann wäre das
339 Klasse.
340

**341 Das besondere ist ja, das Ich falle in diesem Fall ja auch in die Zielgruppe. Und da
342 spiegelt sich meine eigene Erfahrung auch wieder und deckt sich mit den
343 Erfahrungen, die ich von anderen Studierenden gehört habe. Jeder weiß, das es
344 irgendwo steht, aber man weiß gar nicht so richtig was und wo man es suchen muss.**

345

346 Wo muss ich suchen, wie heißt das Ding was ich suche und warum soll ich überhaupt ein
347 Formblatt suchen, wenn man nicht mal weiß das man es überhaupt ausfüllen muss. Das ist
348 im Moment total diffus verteilt und das an Orten, wo die Leute aus eigenem Antrieb gar nicht
349 gucken und deswegen wäre das eine geeignete Stelle um diese Dinge zu bündeln oder
350 zumindest die Links zu finden. Es muss ja nicht nochmal da dupliziert sein. Aber das eine
351 Stelle da ist, wo einem das Tool dann auch gleich sagt, welche Dinge alle wichtig sind und
352 wo man die Hinweise findet und was für Hinweise man gar nicht findet, die dann aber explizit
353 genannt werden. Das fänd ich klasse.
354

355 Ansonsten fällt mir eigentlich nichts weiter ein. Ich denke es wird Dinge geben, die man gut
356 generisch vorbereiten kann, man sollte sich aber auch bewusst sein, dass Arbeiten immer
357 auch einen individuellen Charakter hat und das es Dinge gibt, die man nicht so allgemein
358 abhandeln kann. Das ist glaube ich sonnenklar. Deswegen kann der Versuch nicht der sein,
359 es in seiner Gesamtheit abzuhandeln, sondern eher - und ich glaube auf der Spur sind sie
360 auch - was die Grundprobleme sind. Was ist allen gemeinsam? Wo kann man ansetzen um
361 Alle voran zu bringen. Man sollte auf jeden Fall vermeiden, die Leute mit irgendetwas zu
362 gängeln. Also das man zu sehr versucht, die Leute in irgendwelchen Spuren zu halten, die
363 dann dich nicht zu jeder Arbeit passen. Aber mehr als so eine diffuse Äußerung kann ich
364 auch nicht machen, weil ich das auch nicht weiß. Das muss man dann mal ausprobieren und
365 mal gucken wie das dann ist. Da muss man einfach Mut haben.
366

**367 Ja, mir ist auch klar, das man nicht so einfach den Weg gehen kann und die App
368 "fragen" kann was man machen soll und dann weiß was er braucht und machen
369 muss. Aber ich bin auf jeden Fall dabei, zu untersuchen wo die Probleme liegen, um
370 von da aus dann die Schwachstellen zu finden und zu gucken, was man verbessern
371 kann.**

372

373 Ja, da gibt es dann ja viele Formalien, die vielleicht gar nicht so kompliziert sind bei denen
374 man unterstützt werden kann, wie das Erstellen einer Literaturverzeichnisses. Das man diese
375 Sachen dann schonmal sammelt und dann eben schon mal so eine Liste hat. Also es gibt da

376 ganz viel auf der Handwerklichen Ebene, wo so ein Tool auf jeden Fall unterstützen kann -
377 jeden, ganz egal wie die Arbeit ist - aber auch bei den Sachen die in richtung der Art der
378 Vorgehensweisen gehen gibt es sicherlich auch allgemeine Sachen. Ich glaube da haben
379 sie schon exakt die richtigen Fragen gestellt.

380
381 Für mich klang es so, als wären Sie mit Ihren Fragen auf der richtigen Spur.

Interview IV

1

2 **Frage:** Ihr Fachgebiet wird auf der Internetseite der Fachhochschule Lübeck als
3 "praktische Informatik angegeben". Wenn ein Studierender Sie als Betreuer für die
4 Bachelorarbeit wünscht, und mit einem Themenvorschlag zu Ihnen kommt, der nicht
5 Ihrem Fachgebiet entspricht, gibt es dann Themen bei denen Sie den Studierenden an
6 Ihre Kollegen weiterleiten?

7

8 **Antwort:**

9 Ja, da gibt es Themen. Alle Themen die ich nicht selber fachlich kompetent erschöpfen und
10 betreuen kann. Die sollten schon in meinem fachlichen Bereich liegen - den setze ich Breit
11 an - aber angenommen da kommt jetzt jemand mit der ANforderung "Spieleprogrammierung
12 mit dem Fokus auf künstlerische Gestaltung", dann gibt es hier kompetentere Betreuer.
13 Wenn ich einen Betreuer ausmachen kann, der deutlich kompetenter wäre hier an der FH,
14 würde ich erstmal sagen "gehen Sie mal dorthin und fragen Sie mal", wenn nicht "kommen
15 Sie wieder zurück, dann schauen wir weiter". Also da wird jetzt keiner komplett
16 weggeschickt, nur nochmal umhorchen, ob es einen vielleicht besseren Betreuer gibt.

17

18 **Frage:** Also würden Sie schon sagen, das man hier etwas abgrenzen kann?

19

20 **Antwort:**

21 Ja, das hängt immer vom konkreten Thema ab, das ist nicht so fest als Grenze definiert. Die
22 ist natürlich weit - ein Graubereich. Eine Variante wäre noch als Zweitbetreuer einen
23 Kollegen aus einem anderen Fachbereich hinzuzuziehen.

24

25 **Frage:**Kann man bei Bachelorarbeiten, die Sie betreuen grob gesagt "eine Art"
26 identifizieren?

27

28 **Antwort:**

29 Die Mehrheit hat eine entwickelnde Natur. Das ist aber daraus begründet, dass Studiere
30 etwas entwickeln wollen. Die wenigsten kommen zu mir und sagen, sie wollen irgendwas
31 vergleichen und einen Bericht schreiben. Das habe ich nur in extremen Ausnahmefällen
32 erlebt.

33

34 Die Aufgaben die ich vergeben habe, lassen sich im allgemeinen immer auf ein verteiltes
35 System abbilden.

36

37

38 **Frage: Wo liegen aus Ihrer Sicht die Prioritäten bei dem Bearbeiten der**
39 **Bachelorarbeit. Was verlangen Sie von den Studierenden? Wo liegen die für Sie die**
40 **Punkte bei der Bachelorarbeit, wo Sie sagen würden "Das ist der Inhalt der**
41 **Bachelorarbeit"?**

42

43 **Antwort:**

44 Methodische Vorgehensweise, Systematische Vorgehensweise unter Anwendung der
45 Kenntnisse die die Studierenden im Laufe des gesamten Studiums gelernt, gehört und
46 praktiziert haben. Angefangen bei der Anforderungsanalyse bis hin zum Test.

47

48 Die Vorgehensweise, die den üblichen Modellen entspricht, die ausgiebig diskutiert wurden.
49 Also nicht einfach anfang zu codieren. Das wäre weit weg von unserem Studiengang
50 "Informatik/**SOFTWARETECHNIK**". Entsprechend dem Stand der Kunst, den wir hier lehren.

51

52 Die Herangehensweise an das Problem. Wie nähre ich mich dem Problem? Wie zerlege
53 ich das Problem, weiß ich was eine ANforderungsanalyse ist. Ein Pflichtenheft.

54

55 **Frage: Das geht ja jetzt schon mal in die Richtungen der Erwartungen an den**
56 **Studierenden. Was erwarten Sie denn von den Studierenden im Allgemeinen?**
57 **Dazu zählt zum Beispiel die Kommunikation.**

58

59 **Antwort:**

60 Entscheidend ist wirklich die Vorgehensweise, wie ich vorhin bereits betont habe. Welche
61 sich am Softwarelebenszyklus orientiert. Das man vergleichende Analysen durchführt, wenn
62 es notwendig ist. Das man seine Entscheidungen begründen kann - sei es die Auswahl der
63 Programmiersprache, die Auswahl der Softwarearchitektur - im Vergleich zu den die man
64 nicht verwendet hat und die Bewertung gegenüber den Alternativen.

65

66

67 **Frage: Und zum Stichwort Kommunikation? Was erwarten Sie, wenn der Studierende**
68 **zu Ihnen kommt. Nach dem Motto "So ich bin jetzt zu Ihnen gekommen aber ich hab**
69 **noch nichts aufgeschrieben. Was soll ich tun?".**

70

71 **Antwort:**

72 Dann trete ich mit dem Studierenden in die Aufgabenfindungsanalyse ein und das schon bei
73 dem genauen Formulieren der Aufgabenstellung. In den seltensten Fällen bekommt der
74 Studierende eine fertige Aufgabenstellung, da Sie das ja im späteren Berufsleben auch
75 machen müssen. Aufgaben definieren, abgrenzen und Anforderungen schreiben.

76

77 Die erste Phase ist es, die Aufgabenstellung zu formulieren.

78 Das kann durchaus lange dauern, dass wir eine Aufgabenstellung haben. Das geht über
79 mehrere Treffen hinweg. Darauf lege ich letzten Endes den Fokus, dass dort genau auch
80 darauf geachtet wird, was der Studierende von sich selbst letzten Endes fordert.

81 Ist das erfüllbar? Wie kann er es bewerten?

82

83 Und was die Kommunikation angeht, das er sich regelmäßig meldet und nicht wartet bis er
84 kontaktiert wird. Er MUSS selbst proaktiv den Kontakt zu seinem Betreuer suchen. Das er,
85 wenn er Probleme hat auch möglichst schnell den Kontakt zu mir sucht.

86 Das verstehe ich auch unter selbstständigem Arbeiten.

87

88 **Frage: Gibt es denn typische Probleme die während der Arbeit mit den Studierenden**
89 **(immer wieder) auftreten?**

90

91 **Antwort:**

92 Ja, die gibt es ganz naturgemäß. Das ist halt die erste größere Arbeit, die die Studierenden
93 schreiben.

94

95 Bei der Programmierung gibt es interessanterweise nicht so typische Probleme die sich
96 wiederholen, eher bei der Ausfertigung der Arbeit.

97 Zum Beispiel die Forderung, dass Aussagen begründet werden müssen unter Bezugnahme
98 auf anerkannter Fakten, Publikationen oder selbst durchgeföhrter Analysen. Das sind
99 Erkenntnisprozesse, die fast jeder durchmacht. Bei der Aussage "Das Programm ist schnell"
100 - warum ist es schnell, in welchem Sinne ist es schnell - oder es wird einfach nur gesagt, die
101 eine Programmiersprache ist besser geeignet als die andere - Warum? Aus welchen
102 Gründen? Was sind die Kriterien?.

103

104 Da haben fast alle Studierenden ihre Problem, bis hin zu Details und Fakten, das
105 ordnungsgemäße nachvollziehbare (seitenweise) Zitieren von Quellen oder im Internet
106 Permanent Links, die nachvollziehbar sind. Dann erkennt der Studierende erst warum er die
107 Arbeit so gestalten muss - damit sie für einen Nachfolgenden komplett nachvollziehbar ist.

108

109

110 Das ist glaube ich das, was zum schluss rauskommt - Warum schreiben wir die
111 Bachelorarbeit in der Art und Weise wie wir sie schreiben? - Weil sie eine wissenschaftliche
112 Arbeit ist, die in all Ihren Begründungen nachvollziehbar sein muss für einen externen
113 Beobachter. Das ist eine neue Erkenntnis, die so noch nicht erarbeitet werden konnte im
114 Studium.
115
116 Aber auch das Projektmanagement, das Selbstmanagement, das Zerlegen in Arbeitspakete.
117 Das bekommen einige Studierende besser hin als andere Studierende. Das schätzen von
118 Zeitaufwand ist generell eine große Schwierigkeit bei Softwaretechnik Projekten in der
119 Praxis. Da gibt es auf jeden Fall Unterschiede. Das wurde im Studium ja auch schon
120 stellenweise behandelt - das merkt man auch.
121
122 Wichtiger Rat den ich immer mitgebe "Fangen Sie mit der Dokumentation direkt während der
123 Arbeit von Anfang an - sonst wird es später sehr Schwer und kommt an dieser Stelle dann
124 zu Blockaden"
125
126 Aber was gut funktioniert, ist das Verständnis für den Softwareentwicklungszyklus - das
127 kriegen sie gut hin. Alle wissen es geht mit der Anforderungsanalyse los, über die
128 Entwurfsphasen bis hin zum Testen. Beim Testing wird es wiederum Schwierig, es kann
129 sehr Aufwändig werden ein System zu testen, gerade wenn es ein Benutzerinterface hat.
130 Dann muss ich ja Nutzer davor setzen und eine Nutzeranalyse machen. Das sprengt dann
131 den Rahmen.
132 Aber Testen ist immer ein Problem - vielleicht aus Zeitgründen: nicht früh genug Modul- und
133 Unit-Tests geschrieben.
134 Ich vermute mal, das liegt daran, dass wir an der Fachhochschule kein Wahlpflichtmodul zu
135 dem Thema Testing anbieten. Das ist insgesamt ein Schwachpunkt, der nicht von den
136 Studierenden verschuldet ist. Da wünsche ich mir auf jeden Fall, das das gemacht wird.
137
138

139 **Frage: Können Sie aus Ihrer Sicht den Zeitpunkt konkretisieren, wo der Studierende**
140 **den größten Erkenntnisgewinn hat - also merkt "Oh, das ist das was von mir verlangt**
141 **wird!"**

142

143 Ich glaube, das es zu dem Zeitpunkt ist, bei dem ein bestimmter Umfang an Dokumentation
144 geschrieben ist. Spätestens wenn die Inhaltsstruktur aufgebaut werden soll. Das ist ja auch
145 eins der Themen, die man mit dem Studierenden bespricht. Wie ist der Aufbau der Arbeit?
146 Die Kapitelstruktur? - Ich glaube zu diesem Zeitpunkt dämmert es. Das ist die erste große
147 Schreibaufgabe. Gar nicht mal bei dem Programmieren, das haben Sie im Laufe des
148 Studiums ja schon häufig gemacht. Spätestens beim ersten Feedback zu einem
149 geschriebenen Text und ich die Punkte nochmal bringe, die ich schon vorher angemerkt
150 habe.

151

152 Wobei ich sagen würde, dass sich vieles gebessert hat durch das Bachelorseminar.
153 Ich muss nicht mehr so viel sagen was ich früher bei der Diplomarbeit sagen musste. Ich
154 muss zwar alles nochmal sagen, aber nicht alles nochmal neu erklären. Es ist definitiv eine
155 gute Hilfe. Es ersetzt zwar nicht die Beratung - ich muss vieles wiederholen.

156

157 Man merkt, die Studierenden haben davon schonmal was gehört.

158

159 **Frage: (Bezug auf die Applikation) Glauben Sie, dass es so möglich ist, Studierende**
160 **zielgerichtet zu unterstützen und die vielleicht sogar die Fragebereitschaft zu**
161 **erhöhen?**

162

163 **Antwort:**

164 Auf jeden Fall, viele Beratungsaspekte wiederholen sich im Laufe der Jahre - und da wäre es
165 schon sehr hilfreich auf die Applikation zu verweisen, weil man somit einen ganz neuen
166 Diskussionshintergrund hätte. Nach dem Gespräch ist der Kontakt ja erstmal unterbrochen
167 für eine Weile, das kommt der selbstständige Arbeitsabschnitt. Und für den Zeitabschnitt
168 benötigt der Studierende meiner Meinung nach auch nochmal einen Leitfaden - deswegen
169 wäre das in meinen Augen als Webapp sehr sinnvoll. Auf die wir als Betreuer auch
170 verweisen können.

171

172 Früher hatten wir nur einen Termin, es gab dann zum Ende des 5. Semesters einen
173 Beratungstermin, wo alle Studierende gekommen sind.

174 Da haben wir kurz vorgestellt was eine Bachelorarbeit eigentlich ist, die Studierenden
175 konnten Fragen stellen - daraus hat sich auch das Bachelorseminar entwickelt.

176

177 Daher wäre Ihre WebApp an dieser Stelle unbedingt notwendig.

178

179

180 **Frage: Sehen Sie Probleme durch die Applikation entstehen könnten?**

181

182 **Antwort:**

183 Ja, ganz klar die Wartung der Applikation. Sie schreiben Ihre Bachelorarbeit und haben
184 dann Ihre Note und haben einen lukrativen Job. Aber wer kümmert sich dann um die App.
185 Es muss natürlich gewartet werden, sonst ist es am Ende nur ein totes Konstrukt, was
186 keiner mehr nimmt.

187

188 **Frage: Sehen Sie Risiken durch die Applikation, bei der Nutzung von den
189 Studierenden?**

190

191 **Antwort:**

192 Ja, es könnte sein, dass die Studierenden sich festbeißen und bestimmte Punkte aus Ihrem
193 "Leitfaden" als unumstößlich ansehen und somit in den Konflikt mit den Betreuer kommen,
194 wodurch die Studierenden in eine Abwehrhaltung geraten. Es (Leitfaden) sollte also immer
195 noch als Interpretierbar - als Ratschlag - gelten.

196

197 Es sollte klar erkennbar sein, dass es nicht so gedacht ist. Es kann ja auch vorkommen,
198 dass es bei der App zu inhaltlichen Fehlern kommen kann, welche durch wartung und
199 Einsatz erst aufgedeckt werden können.

200

201

202

Interview V

1
2 **Frage: Ihr Fachgebiet wird auf der Internetseite der Fachhochschule Lübeck als**
3 **“Software Ergonomie, Mensch-Maschine-Interaktion, Softwaretechnik” angegeben.**
4
5 **Würden Sie sagen, das die Bachelorarbeit üblicherweise in diesen Fachgebieten**
6 **stattfinden oder halten Sie das eher offener?**

7
8 Überwiegend ja, allerdings ist das Themenspektrum - gerade auch in der Medieninformatik,
9 wo ich viel lerne und auch in ITD relativ breit - und da kommen da auch mal Fragestellungen
10 die dann eher am Rande liegen. Aber klar! Die Hauptexpertise liegt ja dann in dem Bereich
11 und wenn Anfragen zu Themen kommen, mit denen ich eher gar nichts zu tun habe, würde
12 ich die dann auch eher ablehnen, wenn jetzt nicht zwingende Gründe dafür sprechen, wie
13 zum Beispiel, das derjenige niemand anderen findet oder ganz verzweifelt ist. Üblicherweise
14 liegen die (Themen) in diesem Themenschwerpunkt.

15
16 **Es ist ja eher schwierig einen konkreten Typus von Bachelorarbeiten zu definieren -**
17 **was ich meine ist sowas wie in der Softwareentwicklung eine “entwickelnde” Arbeit**
18 **ist. Kann man Bachelorarbeiten aus Ihrer Sicht (grob) kategorisieren?**

19
20 Das sind verschiedene Typen zum kategorisieren. Zum einen Softwareentwicklungsprojekte,
21 wo im Mittelpunkt steht möglichst auch funktionsfähige Prototypen - meistens sind es ja
22 Prototypen - aber funktionsfähige Applikationen zu entwickeln. Auch dann durchaus den
23 Fokus auf der technischen Umsetzung. Häufig aber auch mit Designaspekten dabei.

24
25 Dann gibt es eher die Designlastigen Arbeiten -die kommen häufig aus ITD - wo es nicht
26 darum geht funktionsfähige Prototypen zu entwickeln, sondern verschiedene
27 Interfacedesigns oder Design Alternativen auszuprobieren und dann zu evaluieren, zum
28 Beispiel hinsichtlich Ihrer Wirkung - was kommt besser an? Dann gibt es auch Arbeiten, die
29 in Richtung Corporate Design oder ähnliches gehen, wo beispielsweise ein neuer
30 Webauftritt entwickelt wird, mit neuem Logo, mit neuen Informationsmaterialien, wie
31 zum Beispiel ein Flyer oder ähnliches für eine Organisation beispielsweise. Das wären so
32 die ITD Arbeiten. Teilweise sind es eben auch wissenschaftlich geprägte Arbeiten, wo es
33 dann auch eine ausgedehnte Literaturrecherche gibt und man mit Hilfe von empirischen
34 Methoden wie beispielsweise Fragebogen- oder Laboruntersuchungen bestimmte
35 Fragestellungen beantworten möchte.

36
37 **Sie sind nun schon auf die Aufgabenstellung im Detail eingegangen. Könnten Sie**
38 **auch etwas zu den Erwartungen an die Studierenden im Allgemeinen sind? Was soll**
39 **der Studierende im Rahmen seiner Bachelorarbeit leisten?**

40
41 Also wichtig ist ja bei einer Bachelorarbeit zum einen, dass es ein eigenständiges größeres
42 Projekt ist, das man in einer überschaubaren Zeitspanne bearbeitet. In den anderen
43 Projekten, die sie im Studium haben, sind das dann ja häufig noch Gruppenarbeiten. Auch
44 wenn da schon größere Projekte bearbeitet werden, macht man das selten alleine. Kommt
45 auch ab und an mal vor, je nach Studiengang. In der Bachelorarbeit haben Sie zum ersten
46 mal die Herausforderung, dass sie allein verantwortlich eine Fragestellung entwickeln, die

47 Methoden planen, das Vorgehen planen, die Umsetzung in die Tat umsetzen und das
48 entsprechend verschriftlichen.

49

50 Diese Eigenständigkeit ist ein Merkmal von (diesen) Arbeiten. Es ist in meinen Augen nicht
51 die Aufgabe des Betreuers jeden Schritt vorzugeben und zu sagen wie sie das machen
52 müssen, denn dann wäre es keine Eigenständige Arbeit mehr. Klar! Als Betreuer begleitet
53 man das, gibt Hilfestellungen, gibt Ratschläge und Feedback, aber letzten Endes liegt die
54 Verantwortung für die Umsetzung immer beim Bearbeiter selber.

55

56 Das andere was ich wichtig finde ist, dass es eine wissenschaftliche Arbeit ist. Das kann von
57 der Herangehensweise unterschiedlich sein. Je nachdem ob ich einen Schwerpunkt in der
58 Softwareentwicklung habe oder ich wirklich eine wissenschaftliche Fragestellung
59 untersuche. Nichtsdestotrotz ist die Vorgehensweise so, dass ich mir nicht einfach selbst
60 was aus den Fingern sauge, sondern mir eine sinnvolle Grundlage erarbeite, anhand von
61 Literaturrecherche oder eben auch indem ich mir verwandte Systeme anschau und gucke
62 wie da bestimmte Dinge umgesetzt sind. Also das ich mir erst einmal eine Grundlage lege
63 und systematisch mein Vorgehen plane und das auch transparent und überprüfbar umsetze.
64 Das ich jeden einzelnen Schritt auch begründen kann und dass das nicht einfach so aus
65 dem Bauch heraus passiert. Das ist unabhängig davon, wie die Fragestellung ist und
66 unabhängig davon, wie die konkrete Methodik ist. Das kann sehr unterschiedlich sein. Es ist
67 halt eine Grundvoraussetzung für das wissenschaftliche Arbeiten, das ich begründet und
68 nachprüfbar vorgehe.

69

70 **Was ist denn wenn ein Prototyp entwickelt werden soll und dieses Ziel nicht erreicht
71 wird, der Weg dahin aber korrekt und begründet dokumentiert ist. Liegt der Fokus
72 dann eher auf dem Weg dahin?**

73

74 Das kann passieren, denn das ist letzten Endes die Natur von Wissenschaft. Ich stelle
75 Hypothesen auf oder habe bestimmte Vermutungen oder Fragestellungen, die ich sicherlich
76 auch gut begründen muss. Die können ja auch nicht aus der Luft kommen, Es kann sich
77 aber trotzdem herausstellen, das die Hypothese falsch war. Das mach dann nicht unbedingt
78 das Ergebnis wertlos, denn auch daraus kann ich dann ja eine Erkenntnis ziehen. Insofern,
79 wenn ich ein Vorgehen plane, das gezielt und begründet umsetze und am Ende merke, aus
80 bestimmten Gründen habe ich die falsche Vorgehensweise gewählt, kann ich daraus jede
81 Menge Lernerfahrung ziehen und das ist auch eine wertvolle Erkenntnis für andere Leute,
82 die dann dieselben Fehler nicht noch einmal machen müssen oder von einer anderen
83 Grundlage starten können. In diesem Sinne ist das dann keine gescheiterte Arbeit.

84

85 **Gibt es denn Fehler und Probleme bei der Bearbeitung der Bachelorarbeit, die immer
86 wieder auftreten? Wo liegen allgemeine Probleme, gerade in Bezug auf die
87 Erwartungen, die Sie angesprochen haben.**

88

89 Also ich glaube ein grundsätzliches Problem - nicht nur bei Bachelorarbeiten, sonder auch
90 bei umfangreichen Projektarbeiten - ist die mangelnde Zeitplanung. Das haben Sie auch
91 schon häufig beobachtet im Studium - vielleicht nicht bei sich selber, aber bei anderen - ,
92 das man Dinge gerne nach vorne schiebt, gerade wenn das nicht so in kleine Häppchen
93 verteilt ist, wo ich genau weiß, (wann ich welche Aufgabe habe). Dann schiebt man das

94 gerne vor sich her und am Ende kommt man dann nicht mehr mit der Zeit aus, die man noch
95 hat.

96

97 Was ich auch empfehle ist zum Beispiel frühzeitig mit dem Schreiben anzufangen. Das wird
98 zwar unterschiedlich gehandhabt. Ich glaube aber, das es nicht hilfreich ist, wenn ich
99 zweieinhalb Monate was tue und in der verbleibenden zwei Wochen alles nochmal schnell
100 aufschreiben und dokumentieren will, was ich gemacht habe. Das ist zum einen viel mehr
101 Aufwand, weil ich mir vieles nochmal neu erschließen muss, wenn ich zum Beispiel eine
102 Literaturarbeit gemacht habe und das dann zwei Monate liegen lasse, dann muss ich im
103 Grunde nochmal alles lesen, wenn ich das dann aufschreiben will. Also dieses zeitnahe
104 verschriftlichen und gutes dokumentieren ist ein Erfolgsfaktor, beziehungsweise kann
105 umgekehrt dazu führen, das man sehr viel mehr Arbeit hat.

106

107 Dann natürlich auch unklare Absprachen. Das passiert häufiger mal, wenn die
108 Bachelorarbeiten in Unternehmen geschrieben werden. Das man sich dann vielleicht nicht
109 genau geeinigt hat, was die Erwartungen sind oder dass das Unternehmen vielleicht auch
110 unrealistische Erwartungen an die Bachelorarbeit hat. Es ist wie gesagt eine
111 wissenschaftliche Arbeit, es kann eben sein, das da was anderes herauskommt als man
112 eben denkt. Dann noch, das (von dem Unternehmen) teilweise sehr stark eingegriffen wird
113 und somit gar nicht mehr diese freie Methodenwahl hat, die man eigentlich braucht, sondern
114 dass das Unternehmen dann eben vorgibt, wie es gemacht werden soll oder Ergebnisse
115 umsetzen will, wobei es gute Gründe geben würde es anders zu machen. Dann ist man als
116 Bachelorarbeiter, der vielleicht auch ein Angestelltenverhältnis in der Firma hat in einer
117 schlechten Position, weil man kaum die Freiheit hat, sich beispielsweise gegen den Chef zu
118 stellen. Dann muss man sehr behutsam Lösungen finden, wie man das dann zumindest
119 auch nochmal reflektieren kann in der Arbeit, wo eben Entscheidungen anders getroffen
120 werden mussten, als man es getan hätte. Manchmal ist es eben auch so, das Kunden sich
121 für eine Designlösung entscheiden, die eigentlich schlecht ist. Wo man dann selber sagt,
122 das es eine schlechte Lösung ist, der Kunde aber darauf besteht, dass es so gemacht
123 werden muss, hat man in so einem Vertragsverhältnis nicht die Möglichkeit es anders zu
124 machen. Das sind Dinge, die muss man dann reflektieren und das kann dann schon zu
125 Schwierigkeiten im Prozess führen.

126

127 **Würden Sie sagen, dass Unternehmen den Bacheloranden bewusst (als billige
128 Arbeitskraft) auf den Praxisteil lenkt, wobei der interne Betreuer ja eher den ganzen
129 Weg sehen möchte?**

130

131 Genau, das kommt vor. Das sollte man dann versuchen, während des Betreuungsprozesses
132 zu lenken, indem man sich zum Beispiel regelmäßig Teilergebnisse vorlegen lässt,
133 zumindest die Gliederung sehr gut bespricht und dann eben auch darauf hinweist, wenn die
134 Grundlagenrecherche nicht so ausgeprägt ist, wie das der Fall sein sollte, damit man da
135 auch möglichst noch gegensteuern kann. Es gibt aber auch durchaus auch Bachelorarbeiter,
136 die sich drei Monate nicht melden und die einfach in ihrem Unternehmen vor sich hin
137 werkeln und letzten Endes eine mehr oder weniger fertige Arbeit vorlegen. Wenn das dann
138 zwei Wochen vor Abgabe ist, hat man auch wenige Möglichkeiten noch korrigierend
139 einzugreifen. Dann bringt es auch nichts zu sagen, das der Grundlagenteil nicht
140 wissenschaftlich genug ist, da man das in zwei Wochen nicht vollends beheben kann. Und
141 das liegt letzten Endes auch in der Verantwortung (des Studierenden). Es ist eine

142 Eigenständige Arbeit. Wenn ich schon wochenlang nichts gehört habe, kann ich mal
143 nachfragen wie der Stand der Dinge ist, aber letzten Endes ist es auch die verantwortung
144 des Einzelnen sich zu melden und Unterstützung einzufordern. Es ist nicht meine Aufgabe
145 den Leuten hinterherzulaufen und zu gucken, dass sie jeden Schritt richtig machen. Das
146 kann dann manchmal zu einem unangenehmen Erwachen führen.
147

148 **In einem solchen Fall wo das Unternehmen ganz klar andere Interessen hat als der
149 Betreuer, ist die Aufgabe dann auch ganz klar bei dem Studierenden zu vermitteln
150 oder zumindest eine Grundkommunikation aufzubauen?**

151
152 Die Aufgabe des Studierenden ist in erster Linie erstmal für Transparenz zu sorgen und sich
153 Hilfe zu holen. Wenn ich den Eindruck habe, dass es in eine falsche Richtung geht, würde
154 ich durchaus auch anbieten, das ich mich mit dem Unternehmen in verbindung setze und
155 dort dann auch das Gespräch suchen mit den Betreuern vor Ort. Das kann ich aber nur dann
156 tun, wenn ich gezielt darauf angesprochen werde. Meistens kenne ich die Leute ja nichtmal,
157 die da in dem Unternehmen sind und das wäre auch von den Ressourcen her nicht zu
158 leisten, ständig vor Ort irgendwelche Besuche zu machen. Das wird manchmal angefragt
159 oder angeboten, aber das kann ich gar nicht leisten. Die Leute sind in allen möglichen
160 Unternehmen. Dann wäre ich ja nurnoch am rumreisen, um dann vor Ort irgendwelche
161 Termine auszumachen. Das ist auch nicht meine Aufgabe. Aber wenn jemand zu mir
162 kommt, mit dem Gefühl in eine falsche Richtung gedrängt zu werden oder nur eine zu kleine
163 Aufgabe zugeteilt bekommen zu haben. Das versucht man im Vorhinein zu vermeiden. Das
164 man gerade bei größeren Projekten sicherstellt, das derjenige auch wirklich eigenständig
165 etwas bearbeiten kann.
166

167 Das kann auch zu Problemen führen - ist tatsächlich auch schon vorgekommen - das dann
168 Dinge in der Bachelorarbeit präsentiert werden, die dann gar nicht von der Person selber
169 erarbeitet wurden. Wir hatten mal einen Fall - da war ich Zweitprüfer - da ist das erst im
170 Kolloquium aufgefallen, weil ich bestimmte Designentscheidungen hinterfragt habe und dann
171 die Person gar nichts dazu sagen konnte. Ich habe mich dann gewundert, und habe die
172 Person darauf angesprochen, dass sie es doch entschieden hat und dann doch begründen
173 müsste, aber dann stellte sich heraus, das sie es nicht entschieden hat. Sie hat das Design
174 auch nicht erstellt. Das haben dann die Kollegen in der Grafikabteilung gemacht. In der
175 Arbeit wurde es dann aber so präsentiert, als wäre es von der Person erstellt worden, was
176 natürlich hochproblematisch ist. Dann muss man sich fragen, ob es überhaupt noch eine
177 Arbeit ist, die als "bestanden" gewertet werden kann. Das sind eben so ein bisschen die
178 Sachen, wo man vorher drauf gucken muss. Sind das wirklich eigenanteile? Kann ich das
179 auch ein Stück weit eigenständig bearbeiten? Habe ich da auch Entscheidungsbefugnisse?
180 Wenn ich letzten Endes nur ein Befehlsempfänger bin, dann ist das nicht geeignet für eine
181 Bachelorarbeit. Das sollte man versuchen im Vorfeld zu klären und meistens gelingt das
182 dann auch.

183 **Die Punkte die sie bisher genannt haben, zielen ja alle eher darauf ab, das es bei der
184 Durchführung der Bachelorarbeit Probleme gibt. Gerade was das Thema
185 wissenschaftliches Arbeiten angeht gehen die Erwartungen von Studierenden und
186 Professoren durchaus auseinander. Wie würden Sie denn die fachliche Kompetenz
187 der Studierenden bewerten, also das was besonders im Studium gelehrt wurde?
188 Gibt es dort auch Probleme?**

189

190 Natürlich gibt es da Unterschiede. Es gibt Leute, die sind besser und es gibt Leute, die sind
191 schlechter. Das ist letztlich ganz klar. Insofern haben wir da auch ein Spektrum bei
192 Bachelorarbeiten, wie in jedem anderen Modul auch. Es gibt Leute, die das alles
193 ausgezeichnet machen und es gibt Leute, bei denen hapert das. Aber es ist selten so, dass
194 die Bearbeitung, die Methodik und die Vorgehensweise und die Planung exzellent sind, aber
195 kein fachliches Wissen da ist, um die Aufgabe zu bearbeiten. Das habe ich eigentlich noch
196 nie erlebt. Das geht so ein Stück weit Hand in Hand. Das würde im Grunde ja bedeuten,
197 dass man sich für ein komplett falsches Thema entschieden hat, wo man zum Beispiel in
198 Informatik dann anfängt, ein Design Thema zu bearbeiten, obwohl man das im Studium
199 eigentlich gar nicht gelernt hat. Oder Umgekehrt, jemand der seinen Schwerpunkt im Design
200 gelegt hat, eine Hardcore Aufgabenstellung aus der Eingebetteten Systeme Welt bearbeiten
201 würde und dann feststellt, das er gar nicht die fachlichen Kompetenzen trägt, um das zu
202 machen. Aber das weiß man ja eigentlich oder kann es im Vorfeld einschätzen, ob es
203 erfolgversprechend ist oder nicht. Typischerweise suchen sich die Leute ja auch Arbeiten,
204 die ihren Schwerpunkten entsprechen und auch ihren Interessen. Das ist ja durchaus auch
205 ein wichtiger Entscheidungspunkt, wenn ich mich für eine Arbeit entscheide. Dann ist das ja
206 auch etwas, wo ich mich später mit bewerben kann, oder wo ich sagen kann "Das habe ich
207 erarbeitet" - als Arbeitsprobe. Und insofern geht das selten komplett auseinander, das
208 jemand komplett exzellent im Vorgehen ist und im Fachlichen schlecht. Umgekehrt kann es
209 durchaus sein, dass man fachlich ordentlich was drauf hat aber es "schlampig" angeht. Das
210 kann man aber mit ein wenig Selbstdisziplin in den Griff bekommen.

211
212 **Betreuen Sie viele Arbeiten aus dem Studiengang Informatik/Softwareentwicklung?**
213 **(Anmerkung: Weil eigentlich Studiengangsleiter für Medieninformatik Online**
214 **gewesen)**

215
216 Es ist häufig so ein "Drittelsmix" gewesen aus den Studiengängen in denen ich überwiegend
217 gelehrt habe, wie Medieninformatik Online - Bachelor und Master, Informationstechnologie
218 und Design und Informatik und daraus rekrutieren sich dann auch die Bachelorarbeiten die
219 ich angefragt bekomme. Mal ein bisschen mehr und mal ein bisschen weniger. Das
220 schwankt immer ein bisschen aber es ist in der Regel eine ganz gut Mixtur.
221
222

223 **Bei IDT gibt es doch ein recht frühes Modul, wo sich mit dem wissenschaftlichen**
224 **Arbeiten beschäftigt wird, oder?**

225

226 Bei IDT nicht, sondern in der Medieninformatik gibt es "Einführung in wissenschaftliche
227 Projektarbeit". Das ist in der Tat - ganz früh nicht, sondern im 4. Semester - schonmal der
228 Bachelorarbeit vorgeschaltet, wo man sich grundsätzlich schon mal mit solchen Dingen, wie
229 Recherche beschäftigt. Wie mache ich ein Literaturverzeichnis? Wie mache ich
230 Literaturangaben? Was bedeutet eigentlich wissenschaftliches Vorgehen?

231

232 **Achso. Hat das einen großen Einfluss auf die Bachelorarbeit? Merkt man da einen**
233 **Unterschied? (Zwischen den Studiengängen)**

234

235 Die Bachelorarbeiten sind insofern schwierig zu vergleichen, weil das Klientel einfach
236 unterschiedlich ist. Wie haben in der Medieninformatik Online sehr viele Berufstätige, die
237 berufsbegleitend studieren und insofern auch länger brauchen. Das ist nicht der Faulheit
238 geschuldet, sondern weil sie geplant weniger Zeit für das Studium aufwenden und die
239 Bachelorarbeiten entstehen dann häufig aus dem Arbeitskontext heraus. Weil es natürlich
240 effektiv ist, wenn ich in einem Unternehmen tätig bin, wo man Projekte teilweise in einer
241 relativ hohen Position betreut, dann ist das auch naheliegend, dass ich sowas eben auch für
242 eine Bachelorarbeit suche. Das ist dann auch eine ganz andere Position als jemand, der ein
243 Praktikum im Unternehmen macht oder eben als Bachelorstudent oder Werksstudent
244 angestellt wird für eine Zeit. Die Projekte sind dann teilweise auch schon ein bisschen
245 komplexer und anspruchsvoller, weil die Leute dann teilweise auch schon 10 Jahre
246 Berufserfahrung haben und häufig mit diesen praxisorientierten herangehensweisen sehr
247 vertraut sind. Da ist meistens nicht das Problem, das so auszuarbeiten wie man das in der
248 Praxis dann eben macht. Da ist eher dann die Herausforderung es wissenschaftlich zu
249 betrachten und ich hoffe das dieses Modul dabei hilft aber ich könnte es jetzt nicht 1 : 1 mit
250 anderen Studiengängen vergleichen und sagen, dass die Bachelorarbeiten deshalb besser
251 wären. Das ist auf jeden Fall etwas, was ich jedem Studiengang empfehlen würde das
252 frühzeitig mit einzubringen. Auch für Projektarbeiten kann es sehr nützlich sein.

253

254 **Also es wäre Ihrerseits auf jeden Fall ein Ansatz so ein Modul einzuführen, um damit**
255 **die von Ihnen angesprochenen Probleme anzugehen oder zu lösen. Wie sie schon**
256 **gesagt haben, machen Studierende (im Laufe des Studiums) mehrere Projekt -**
257 **größere Projekte, kleinere Projekte alleine oder auch als Gruppenarbeit. Haben Sie**
258 **weitere Anregungen, wie man diesen Übergang aus dem "Studium" in die**
259 **Bachelorarbeit "besser" gestalten könnte oder diese auftretende Kluft zu minimieren?**

260

261 Ich versuche das teilweise in meinen Lehrveranstaltungen zu tun, indem ich nicht Klausuren
262 als Prüfungsleistung nutze, sondern eben da schon semesterbegleitende Projektarbeiten,
263 die dann oft schon solche Komponenten beinhalten. Sie haben ja auch Module von mir
264 belegt. Es gibt dann ja auch Themen zu wählen, dass eine wissenschaftliche
265 Herangehensweise verlangt. Das mache ich nicht als Pflicht, weil das ja nicht der primäre
266 Inhalt des Moduls ist, aber viele tun das. Dann bekommen sie natürlich auch Feedback zur
267 Herangehensweise und können das schon mal üben. Was bedeutet das eigentlich eine
268 sinnvolle Quellenrecherche zu machen? Wie kann ich beurteilen, ob die Quelle die ich
269 gefunden habe seriös ist? Wie mache ich einen sinnvollen Quellenverweis? Das sind alles
270 Dinge die man so schonmal übt. Alles in einem geschützten Rahmen, weil da noch nicht der

271 Hauptfokus darauf liegt. Es gibt natürlich Feedback, aber es wird nicht erwartet, dass es da
272 schon perfekt läuft. Das finde ich schon relativ wichtig, das man da ab der Hälfte des
273 Studiums wenn man in das 4. oder 5. Semester kommt, dass man da schonmal etwas
274 anspruchsvollere Aufgaben bearbeitet und nicht dieses typische Auswendiglernen für eine
275 Klausur und danach hat man alles einfach abgehakt.

276

277 **(Kurze beispielhafte Beschreibung der im Rahmen der Bachelorarbeit zu**
278 **entwickelnden Applikation)**

279

280 **Was ich von studierenden in Interviews wahrgenommen habe ist, das sie den Zeitplan**
281 **zum einen völlig unterschätzen. Einer hat sogar schon damit gerechnet, das es am**
282 **Ende eng wird, schiebt seine Arbeit nach eigener Aussage trotzdem immer weiter auf.**
283 **Da soll die App zum einen Ansätzen. Das der Fokus auf Dinge gelenkt wird, die**
284 **teilweise im Vorfeld unangenehm zu betrachten sind, aber im Nachhinein den Nachteil**
285 **ausgleichen. Die Hoffnung ist, dass die Studierenden den Fokus haben. Frühzeitig**
286 **erkennen, wo die Schwierigkeiten liegen. Das "Zeitmanagement Tool" ist dabei nur**
287 **ein Aspekt. Darüber hinaus, sammle ich viele Tipps und Hinweise, die Studierende**
288 **sonst viel zu spät selbst erfahren durch Fehler. Die ganz Idee ist natürlich, dass das**
289 **begleitend funktioniert.**

290

291 **Wie das Beispiel "Schwimmen lernen". Wenn man vorher erklärt, wie es funktioniert**
292 **(wie im Bachelorseminar) rennen am Ende alle rein und machen die üblichen Fehler**
293 **trotzdem (wie bei der Bachelorarbeit).**

294

295 **Erkennen Sie bei einer solchen Applikation Chancen oder vielleicht sogar Risiken für**
296 **die Bearbeitung der Bachelorarbeit?**

297

298 Ich denke, die Unterstützung beim Zeitmanagement ist essenziell. Da habe ich ja vorhin
299 ausgeführt, dass man Dinge gerne vor sich herschiebt. Unterstützung beim Zeitmanagement
300 würde ja auch ein bisschen "Unterstützung beim Planungsprozess" implizieren. Ich kann ja
301 nur dann ein gutes Zeitmanagement haben, wenn meine Planung gut ist. Wenn ich sage
302 "Ich brauche drei Wochen für die Literaturrecherche" und habe mich dabei aber grandios
303 verschäzt, dann bringt mir die App aber auch nichts wenn sie mir sagt, dass der Literaturteil
304 bald dran ist, also wenn die ganze Zeitplanung letzten Endes unrealistisch war. Da könnte
305 man sich überlegen, was ich dieses Semester ausprobiert habe, dass man eine sehr
306 Strukturierte Arbeitspaketbeschreibung macht. So, wie man es bei Projekten in der
307 Wirtschaft oder in der Forschung auch machen muss. So, dass ich mir überlege, wieviele
308 Stunden ich in etwas rein stecke, welche Tätigkeiten es konkret sind und man nicht nur so
309 etwas schwammiges wie Literaturrecherche oder Anforderungsanalyse schreib, sondern mir
310 wirklich konkret überlege, welche Datenbanken ich zum Beispiel anschauen muss. Komme
311 ich da überhaupt heran? Habe ich da Zugang? Wenn ja, habe ich den Zugang auch von
312 Zuhause oder muss ich dafür beispielsweise in die Bibliothek, um mich mit einem Rechner
313 vor Ort einzuloggen? Wie viele Rechercheergebnisse erwarte ich mir denn? Reicht es wenn
314 ich am Ende 5 einschlägige Publikationen habe oder erwarte ich eher fünfzehn? Sowas
315 kann ich ja auch mit dem Betreuer besprechen. Wie ausführlich muss alles sein? Mache ich
316 in der Anforderungsanalyse eher Interviews oder Fragebögen? Wenn ja, wie viele Personen
317 muss ich eigentlich befragen? Komme ich an die Leute leicht heran oder ist das total
318 schwierig? Muss ich damit rechnen, dass ich erstmal zehn Leute bevor einer mir zusagt?

319 Das sind ja alles Dinge, die letzten Endes die Komplexität bestimmen und der Unterschied
320 dann darin liegen kann, ob ich eine Woche brauche oder drei Wochen. Insofern denke ich,
321 dass nicht nur Zeitmanagement, sondern auch eine Unterstützung bei der Planung wäre
322 eine wichtige Sache, indem man zum Beispiel bestimmte Templates vorgibt, die mir
323 bestimmte Informationen abverlangen. Das ich mir über bestimmte Punkte eben Gedanken
324 machen muss. habe ich schon die Fähigkeiten die ich dafür brauche? Muss ich mit
325 Einarbeitung rechnen? Wie ist der Umfang konkret zu benennen? Woraus speist sich das?
326 Also das ich wirklich gezwungen bin eine konkrete Planung zu machen und dann wird auch
327 meine Zeitplanung besser sein, weil ich das dann viel besser einschätzen kann. Das ist ja
328 alles was letzten Endes in der agilen Softwareentwicklung. Ich muss letztlich möglichst
329 kleine Pakete schnüren, die ich dann gut abschätzen kann. Wenn ich ein riesenpaket habe,
330 schreibe ich dann vielleicht 10 Personenmonate dran, aber ob das nun realistisch ist, weiß
331 ich nicht. Wenn ich aber sage, dass ich drei Klassen implementieren muss, dann habe ich
332 ein viel handhabbareres Paket letzten Endes, dass ich überschauen kann. Deshalb glaube
333 ich, kann man relativ viel aus der agilen Softwareentwicklung entnehmen für so eine
334 Bachelorarbeit, denn das ist eigentlich ein sehr gutes Beispiel dafür. Man hat einen
335 begrenzten Zeitraum und ein im Vergleich zu anderen Softwareentwicklungsprojekten
336 kleines Projekt. Dann kann man möglichst viele kleine Arbeitspakete verteilen, die ich dann
337 gut Plane und abarbeiten kann und wo man später gegebenenfalls auch gut gegensteuern
338 kann, wenn man merkt, das man falsch geplant hat. Dann habe ich die Möglichkeit im
339 nächsten Arbeitspaket gegenzusteuern. Wenn Ihre App dabei unterstützen würde, wäre das
340 denke ich sehr nützlich.

341

342 Erkennen Sie vielleicht auch Risiken, was die Benutzung der App angeht?

343

344 Man kann sich natürlich zum einen "trotz planen". Das kann natürlich auch passieren, das ich
345 dann vor lauter Planung nicht zum Bearbeiten komme. Das sollte nicht passieren. Und
346 weiterhin kann die App natürlich nicht die Eigenverantwortung abnehmen. Man kann am
347 Ende nicht sagen, das "die App mich nicht rechtzeitig erinnert hat und deswegen ist meine
348 Bachelorarbeit nichts geworden". Das ist dann eher so eine Grundhaltung. Die Leute die das
349 für sich annehmen - Das ist meine Arbeit, das ist mein Projekt, das ist auch so ein bisschen
350 mein Baby und da stecke ich Herzblut rein. Das ist glaube ich schon erforderlich um so eine
351 Arbeit über die Runden zu kriegen. Das man davon überzeugt ist und dass man das will.
352 Immer dann wenn man gar nicht wirklich weiß, ob einen das Thema so richtig interessiert, ist
353 es schon so ein Alarmsignal. Das ist dann keine gute Voraussetzung. Man muss eben schon
354 ein bisschen für die Sache "brennen" und das kann keine App erzeugen. Das muss aus der
355 Eigenmotivation heraus kommen.

356

357 Also sollte die App das natürlich auch klar kommunizieren, was die Aufgabe der App 358 ist.

359

360 Genau

A.1.2. Zusammenfassung der schriftlichen Befragung der Studierenden

Ergebnisse der schriftlichen Befragung von Studenten Teil 1

Beschreibung:

Die wörtlichen Aussagen der befragten Studierenden wurden im folgenden Dokument verschiedenen ermittelten Kategorien zugeordnet. Die gewählten Kategorien (**Erwartungen, Einschätzung der größten Probleme und Wünsche für bessere Vorbereitung durch das Studium**) gehen der Fragestellung hervor, während die jeweiligen Unterkategorien durch sortieren und filtern von Antworten entstanden sind.

Insgesamt ausgewertete Gespräche mit Studierenden: 17

Erwartungen

Teilen von Erfahrungen

- Ich erwarte, dass er mich bei der Arbeit mit der Firma unterstützt, indem er mir zum Beispiel Dinge sagt die ich nicht wissen kann. Wie z. B. dass das zeitlich nicht passen wird, oder ein Teil der Arbeit nicht in eine BA gehört. Ich erwarte, dass er seine Erfahrungen mit mir teilt.
- Dass er mich in meiner Bachelorarbeit unterstützt, auch was die Kommunikation mit der Firma betrifft.
- Dass er bei der Themenfindung mögliche Richtungen aufzeigen kann, die für ein geeignetes Thema eingeschlagen werden können.
- Erarbeitung eines sinnvollen Themas
- Es würde mich freuen, wenn er mich über den zeitlichen Ablauf einer BA informiert also wie viel Zeit man hat was wann ansteht
- Zudem soll er den Umfang der Arbeit richtig einschätzen können.

Kommunikation

- Erreichbarkeit, das Fragen schnell beantwortet werden oder man sich treffen kann, wenn was ist und Kritik.
- Die Möglichkeit, überhaupt mal kommunizieren zu können, Treffen bei Erreichen eines neuen Meilensteins.
- Ich erwarte, dass der Betreuer die nötige Zeit aufbringt, um mich betreuen zu können und auch ansprechbar und aufgeschlossen gegenüber Fragen ist.
- Ich erwarte das ich mit meinem Betreuer jeder Zeit über das Projekt reden kann, wenn ich wenn ich Fragen habe

Feedback

- Und dass er sich mit meinen Sachen auseinandersetzt und nicht nur hofft das ich das schon gut mache.
- Bei Fragen Hilfestellung geben.
- Von meinem Betreuer erwarte ich konstruktives Feedback in Bezug auf Ideen die ich ihm mitteile, außerdem konstruktive Kritik an meiner Dokumentation.
- Grundsätzliches Interesse an der BA des Betreuers, konstruktives Feedback und eine gewisse Diskussionsbereitschaft.
- An den vereinbarten Feedackpunkten Rückmeldung zu meiner Arbeit geben mich dabei davor bewahren überdurchschnittlich dumme Dinge zu tun
- regelmäßiges Feedback über aktuellen Arbeitsstand und das er sagt, wenn was scheiße ist

- Einfach nur Feedback. Quasi eine Person, die Korrektur liest. Also jemand, der auf Unverständlichkeiten hinweist bzw. bei groben schlechten Entscheidungen dazwischenfunkt
- Das der Betreuer sich genügend Zeit für Fragen/Antworten nimmt und einen nicht ignoriert.
- Dass er sich Zeit nimmt für den Studenten und den Studenten dabei unterstützt eine wirklich professionelle wissenschaftliche Arbeit zu schreiben
- Sinnvolles Feedback während der Entstehung der Arbeit und gegebenenfalls Denkanstöße bei auftretenden Problemen
- Dass er für Fragen offensteht
- Mir würde sehr viel Tipps zu Struktur der Bachelorarbeit helfen.

Eingreifen des Betreuers

- Wenn er erkennt, dass ich was falsch mache, das er mich in die richtige Richtung lenkt
- Weiterhin kann manchmal ein kleiner Schubser, was das Zeitmanagement betrifft nicht schaden denke ich.
- Anregungen für Erweiterungen. Andere Denkweisen anstoßen. Aber auch Einforderungen für Leistungen.
- Dass er von sich aus sagt welche Teile meiner BA er sehen will, also wann er meine ersten Implementierung-Versuche sehen will oder die Gliederung oder ob da erst was Sinnvolles drinstehen soll.
- Dass er die Zielstellung klar definiert

Einschätzung der größten Probleme

Schreiben der Dokumentation

- Unsicher war ich mir beim Gewichten der Ausführungen. Also wie viel muss ich zu einem Abschnitt schreiben und was davon alles mit Quellen belegen.
- Ausführlichkeit der Doku
- Die Doku was drinstehen soll. Also die Gliederung. Seitenanzahl. Rechtschreibung.
- Die Einschätzung vom Umfang der Arbeit
- Eine sinnvolle Doku parallel zum Code zu erstellen und eine gute Gliederung für diese zu finden
- Die größten Problemen in meinem Fall würden sehr wahrscheinlich bei der Dokumentation der Bachelorarbeit liegen. Vor allem - man weiß nie, ob man sich in der richtigen Richtung bewegt (zumindest kann man in dem Fall nicht hundert Prozent sicher sein). Das zweite Problem ist, dass Bachelorarbeit schon eine Art wissenschaftlicher Arbeit ist. Umgangssprache passt in dem Fall nicht.

Zeitmanagement

- Definitiv das Zeitmanagement. Ich muss dringend auspassen das ich das schreiben nicht vergesse und nur Code.
- Das größte Problem wird bei mir darin liegen, einen Zeitplan zu erstellen
- Zeitmanagement
- Zeitmanagement
- ZEIT
- Zeitmanagement wird sehr wahrscheinlich ein Problem sein
- Einschätzung des Umfangs

Fachliche Probleme

- Konkrete Anforderungen
- Weiterhin werde ich vor einem Problem stehen, welches wie folgt aussieht:

Die von mir programmierten HMI-Module sind mit einem Bus mit dem Host verbunden. Sollten jetzt 2 Module ausfallen und ausgetauscht werden, soll die Software des Hosts in der Lage sein, neue Module zu erkennen und mit denselben Aufgaben zu bestücken, wie ihre Vorgänger.

Beispielsweise sind die beiden Module für "Anker rauf/runter" und "Notausstieg auf/zu" defekt, werden ausgetauscht und die Funktionalitäten sollen dann bei den nächsten Modulen an derselben Stelle liegen.

- Hardware zum Laufen kriegen
- sinnvolles Testen
- Unsicherheiten bei den implementation, wenn es nicht so wird wie ich es erhofft hätte

Sonstige Probleme

- Die Kommunikation zwischen Betreuer, Firmenmitarbeiter und mir zu koordinieren. soweit ich weiß, wechselt mein Betreuer in der Firma oder es müsste mit mehreren Personen und dem Betreuer kommuniziert werden. Hier hat mein Betreuer schon gesagt, dass der gern EINE Ansprechperson hätte und nicht mit der ganzen Firma diskutieren müsste.
- Schwer zu sagen, gefühlt alles
- Größtes Problem sehe ich immer noch in der Themenfindung.
- Meine größten Probleme liegen bei der Themenfindung für eine Bachelorarbeit zu finden

Wünsche für bessere Vorbereitung durch das Studium

Zeitmanagement

- Bessere Vorbereitung im Bereich Planung und Zeitmanagement. Wie lange brauchen verschiedene Sachen wirklich. Welche Probleme treten auf, mehr mit größeren Projekten arbeiten. Das Softwaretechnik Projekt ist dafür sehr gut, nur leider kommt es ganz zum Schluss und auch nur ein Modul. Webtech war zwar ähnlich, aber da hat man wenig mit Management gemacht. Da haben wir meistens "einfach drauf los programmiert".
- Mehr Projektarbeiten evtl. auch bestehende Projekte übernehmen (z.B. SWT-Projekt AutoSail), evtl. Aufgaben in denen man mal so Zeitschätzungen machen soll, beispielsweise, wie lange dauert das Modellieren einer Autonomen Steuerung für das AutoSailProjekt mit anschließender Auflösung, um zu gucken, dass man eine völlig falsche Zeitvorstellung hat. Den Studenten auf den Boden der Tatsachen zurückholen und ihm klar machen das nicht alles mal eben so runterprogrammiert wird in 2 Stunden.
- Zeitaufwände besser abschätzen können
- Zeitmanagement ob man noch in der Zeit liegt. Das mit dem Feedback von anderen bachelorstudenten würde die Motivation sicherlich erhöhen jedoch nur wenn diese auch selber Fortschritte vorweisen.
-

Wissenschaftliches Arbeiten und Schreiben

- Ein Kurs "Wissenschaftliches Schreiben", am besten speziell für Informatiker. Darin können die Unsicherheiten beseitigt werden. Viele haben ja auch Probleme damit, wie Quellcode in den Text untergebracht werden kann. Gibt zwar häufig Tipps dazu (auch im Lernraum) aber trotzdem sind mit einem kleinen Absatz über den Umgang mit Quellcode längst nicht alle Fragen geklärt. Außerdem müssten die Betreuer dann nicht jedes Mal aufs neue Hinweise dazu geben. Dieser Kurs sollte aber vor dem 6. Semester angeboten werden (als NT-WPF?).
- Richtiger Aufbau einer wissenschaftlichen Arbeit, Sinnvolle Quellen angeben und Zitate etc. einbringen
- Mehr Wert auf die Dokumentation in allen Projekten legen. Diese Dokumentation muss aber in allen Modulen einheitlich und gleich sein (zumindest demselben Muster folgen). Vielleicht sollen die Studierenden sich mehr mit Papers beschäftigen (z.B. Studenten erzwingen, so was wie Paper von Google zu Big Table zu lesen, damit man sich an "wissenschaftlichen Styl" gewöhnt).

Informationen zur Bachelorarbeit

- Ich hätte mir gewünscht besser auf die BA vorbereitet zu werden. Sowas wie im 4ten Semester mal eine Infoveranstaltung, was man machen könnte, worauf man achten muss bei dem planen und das man jetzt schon mal anfangen sollte zu gucken. Vielleicht sogar vor und Nachteile und/oder Erfahrungen von BA in einer Firma oder im Ausland. Sowas fehlt komplett!
- Dass man mehr auf die Formalitäten einer Bachelor Arbeit vorbereitet wird (richtiges zitieren, Schriftform etc.)
- Überschaubarkeit des Projektes. Welcher Aufwand ist wirklich notwendig.
- Vielleicht mehr Informationen über das organisatorische (vielleicht hab ich die auch noch nicht, weil ich mich da noch nicht mit befasst hab), ansonsten fällt mir da leider nicht viel ein
- Im 5 Semester Einführungsveranstaltungen und Beratung, falls kein Thema gefunden wird

- Eine Infoveranstaltung für die Bachelorarbeit. Also früher, irgendwann im 3/4 Semester. Ein paar grobe Regularien. Also wie das mit Prüfern ist, was ich alles fertig haben muss, ob ich früher abgeben kann, wie funktioniert das Kolloquium...
-

Sonstige Anmerkungen

- Eigentlich nicht wirklich viel mehr, da ich mich mit der Einstellung angemeldet habe, dass es eher allgemein wird und dass ich viele Teilbereiche kennen lerne.

Ich finde dieses Studium ist mehr drauf aufgebaut einen breiten Umfang des Themen Bereichs Informatik zu geben und die BA nur Mittel zum Zweck ist, den Zettel zu bekommen von daher finde ich es so in Ordnung wie es ist. Halt nur wenn man nicht drauf vorbereitet wird, ist es wichtig das der Mentor möglichst vieles erklärt

- Ich finde, dass alles wichtige gezeigt/beigebracht wurde damit man für die Bachelorarbeit vorbereitet ist.
- Bisher bin ich zufrieden mit dem Inhalt
- Viel mehr verschiedene Design Pattern während des Studiums Kennenlernen und auch durchgehend in den Praktika einsetzen und viel mehr testen von Applikationen während des studiums
- Da würde mir jetzt nichts konkretes einfallen
- nichts

Ergebnisse der schriftlichen Befragung von Studenten Teil 2

Beschreibung:

Befragung der Studierenden unter Angabe, ob die Studierenden positiv oder negativ auf die Ideen Vorschläge reagiert haben

Motivation durch soziale Einbindung anderer Studierende

Die Antworten lassen sich in drei Kategorien unterteilen, welche sich aus den positiven, negativen oder neutralen Antworten zusammensetzen.

Art der Reaktion	Häufigkeit der Reaktion
Positiv	3
Negativ	12
Neutral	2

Motivation durch das Sammeln von Achievements

Die Antworten lassen sich in drei Kategorien unterteilen, welche sich aus den positiven, negativen oder neutralen Antworten zusammensetzen.

Art der Reaktion	Häufigkeit der Reaktion
Positiv	15
Negativ	1
Neutral	1

Ergebnisse der schriftlichen Befragung von Studenten Teil 3

6. Würdest du die App nutzen um eine der folgenden Zwecke zu erfüllen?

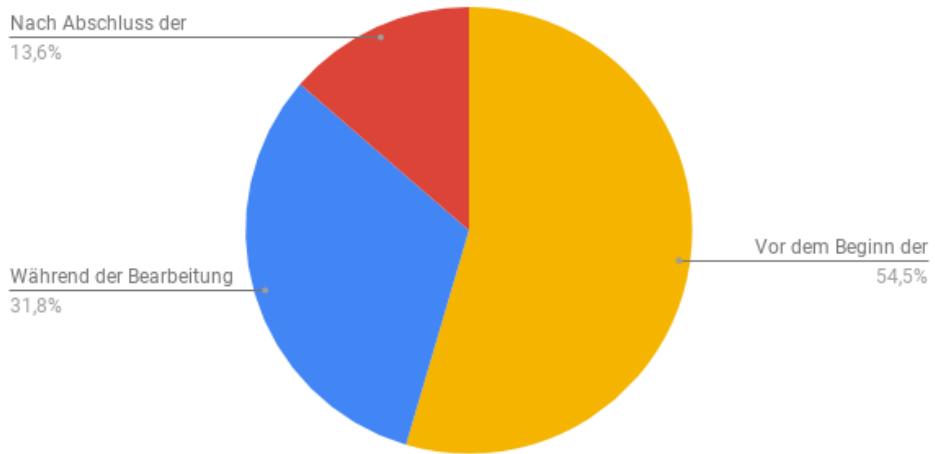
- Strukturierte Informationssammlung zum wissenschaftlichen Arbeiten, Gliederung und weiteren Informationen zu Kernpunkten der Bachelorarbeit?
- Spielerische Einbindung der Bachelorarbeit durch sammeln von Achievements?
- Planung und Übersicht von Meilensteinen, Aufgaben

Art der Antwort	Häufigkeit der Antwort
Positiv	14
Negativ	0
Neutral	3

A.1.3. Ergebnisse der Online-Umfrage für die Studierenden

Allgemeine Fragen

In welcher Phase des Studiums/der Bachelorarbeit befindest Du dich? Bitte wähle die am ehesten zutreffende Antwort.

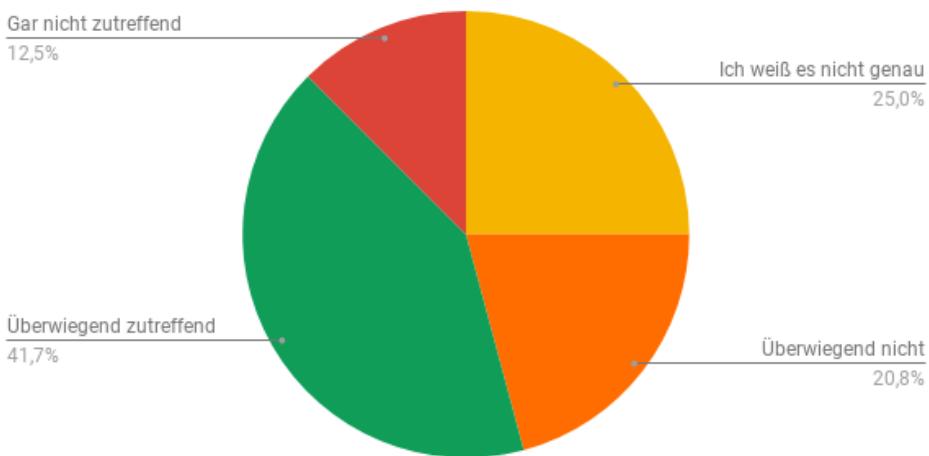


Ca. 55% der Befragten befinden sich zum Zeitpunkt der Umfrage vor dem Beginn der Bachelorarbeit (d.h. auch in den vorherigen Semestern), ca. 33% befinden sich bei der Bearbeitung und etwa 13% haben bereits eine Bachelorarbeit geschrieben.

Die nachfolgenden Fragen wurden dem jeweiligen Status der Befragten angepasst.

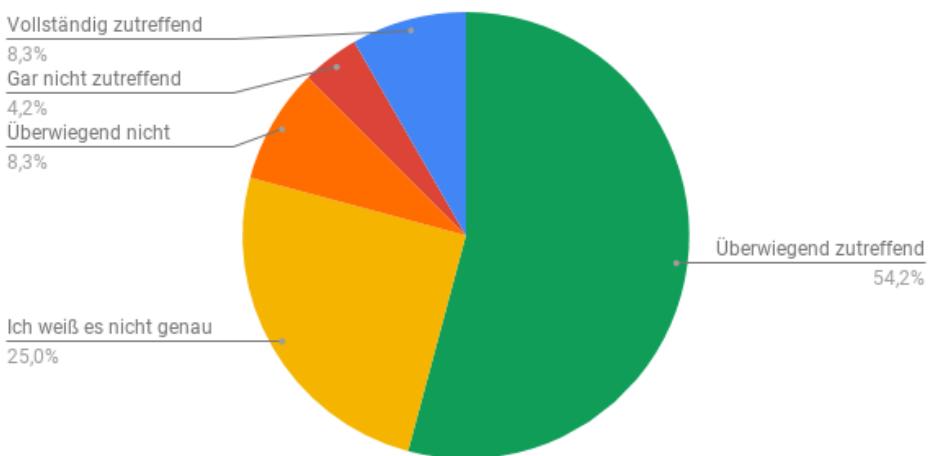
Vor dem Beginn der Bachelorarbeit

Ich fühle mich durch das bisherige Studium sehr gut auf das Bearbeiten der Bachelorarbeit vorbereitet



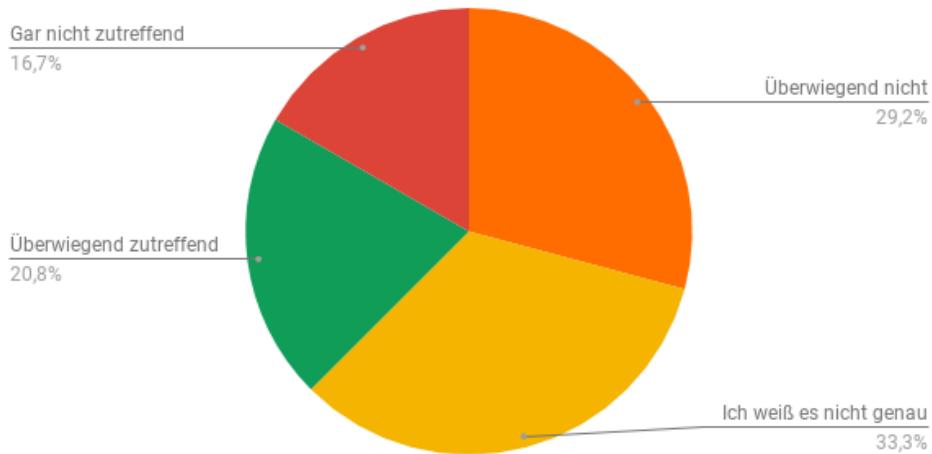
Etwa 40% der Studenten fühlen sich durch das Studium überwiegend auf das Bearbeiten der Bachelorarbeit vorbereitet. Der Rest fühlt sich unsicher oder überwiegend nicht bis gar nicht durch das Studium auf die Bachelorarbeit vorbereitet.

Das selbstständige Vorgehen wird kein Problem für mich darstellen



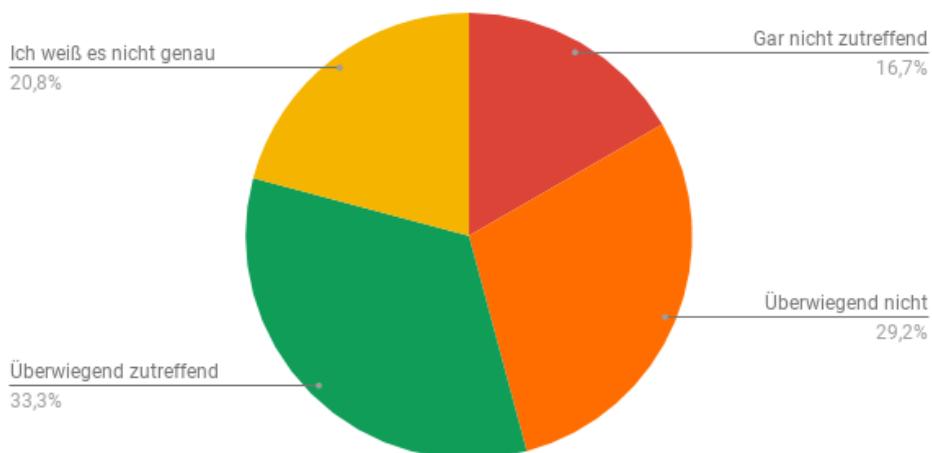
Knapp über die Hälfte der Studenten sehen überwiegend keine Probleme beim selbstständigen Vorgehen, 8% sehen sich auch vollständig dazu in der Lage. Ein Viertel ist sich unsicher und der Rest fühlt sich überwiegend nicht bis gar nicht dazu fähig, selbstständig vorzugehen.

Ich habe eine genaue Vorstellung davon, wie die Bearbeitung einer Abschlussarbeit abläuft



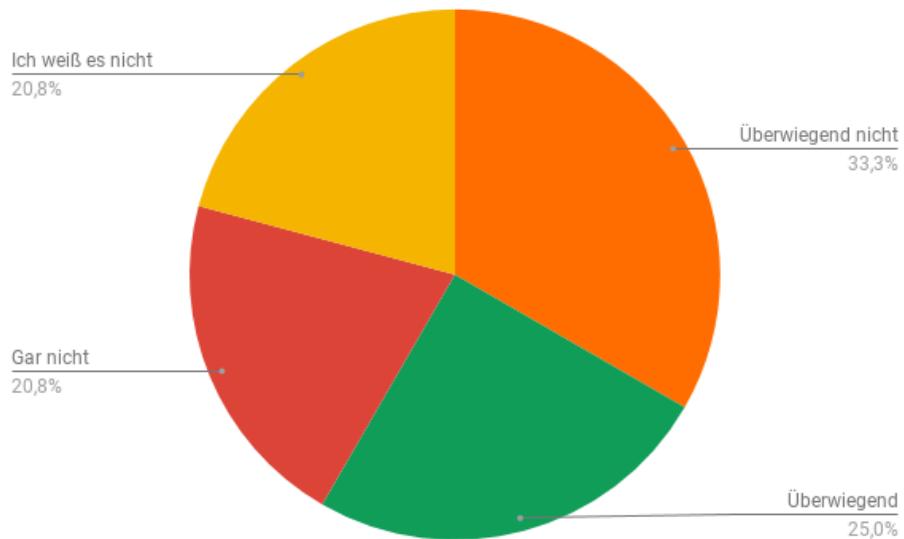
Knapp die Hälften der Studenten haben überwiegend keine oder gar keine genaue Vorstellung davon, wie die Bearbeitung einer Abschlussarbeit abläuft. Ein Drittel ist sich unsicher und die restlichen 20% haben überwiegend eine Vorstellung davon.

Das Einschätzen von Zeitaufwand (Meilensteine, Teilziele) fällt mir leicht



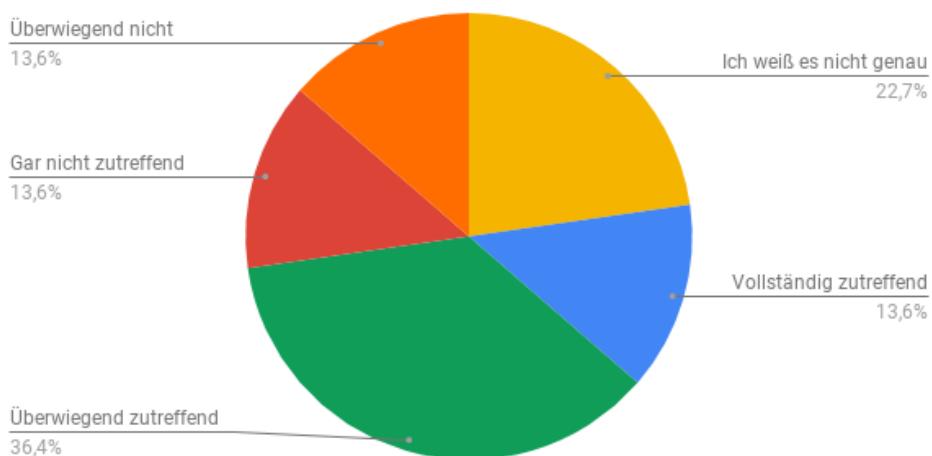
Ein Drittel der Studenten fühlt sich überwiegend sicher mit dem Einschätzen von Zeitaufwand, 20% sind sich unsicher und die restlichen ca. 45% fühlen sich überwiegend nicht oder gar nicht sicher dabei.

Ich weiß ganz genau, was bei der Bearbeitung meiner Abschlussarbeit von mir verlangt wird und bei welchen Punkten die Prioritäten liegen



Ein Viertel der Befragten weiß überwiegend, was von ihnen verlangt wird. 20% sind sich unsicher, ein Drittel ist sich überwiegend unsicher und 20% wissen gar nicht was von ihnen erwartet wird.

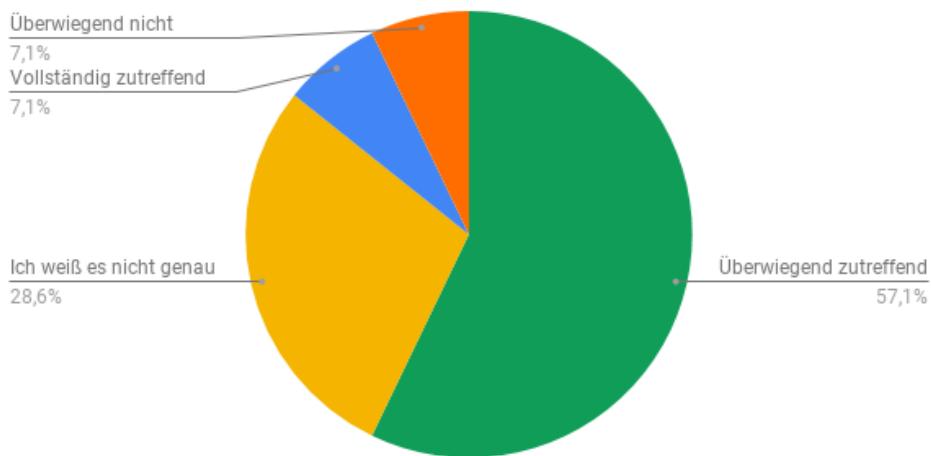
Ich weiß, wie man korrekt wissenschaftlich arbeitet und vorgeht (Literaturarbeit, Quellenangaben)



Die Hälfte der Studenten weiß mindestens überwiegend, wie man korrekt wissenschaftlich arbeitet, 13% davon sind sich vollständig sicher. Fast ein Viertel weiß es nicht genau und der Rest weiß es überwiegend nicht oder gar nicht.

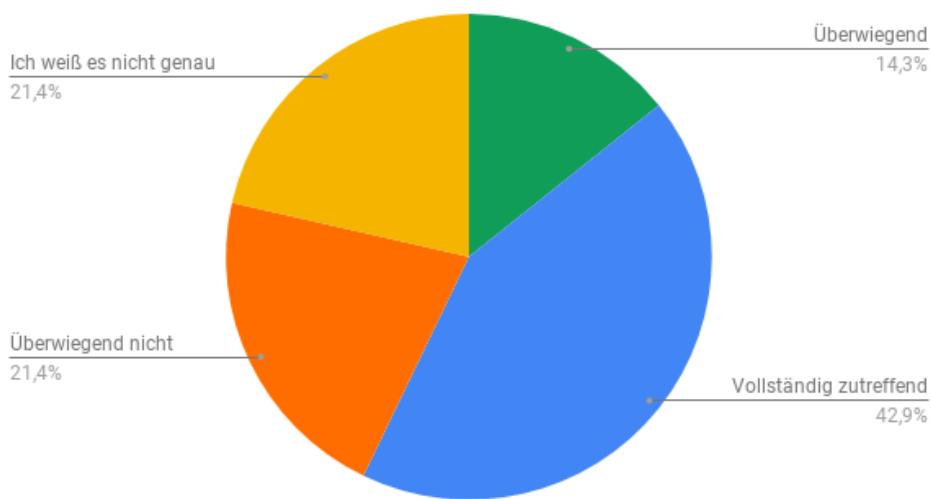
Während der Bearbeitung der Bachelorarbeit

Ich fühle mich durch das Studium sehr gut auf das Bearbeiten der Bachelorarbeit vorbereitet



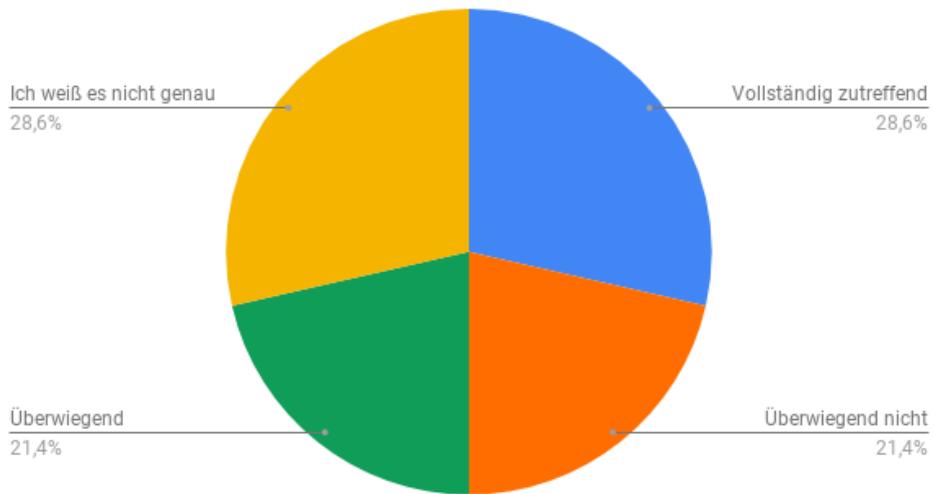
Knapp 65% der Studenten sehen sich mindestens überwiegend auf das Bearbeiten einer Bachelorarbeit vorbereitet, 7% davon auch vollständig. Der Rest ist sich mindestens unsicher, 7% davon fühlen sich überwiegend unvorbereitet.

Es fiel mir leicht, das Thema meiner Bachelorarbeit zu finden



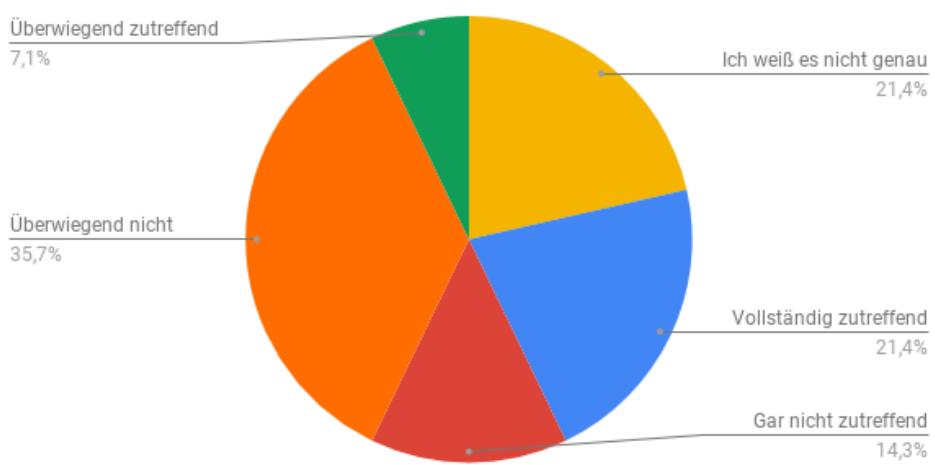
Einem großen Teil fiel es vollständig leicht, ein Thema für die Arbeit zu finden, 14% fielen es überwiegend leicht. Dagegen war sich der Rest unsicher oder überwiegend nicht in der Lage, leicht ein Thema zu finden.

Das selbständige Vorgehen stellt kein Problem für mich dar



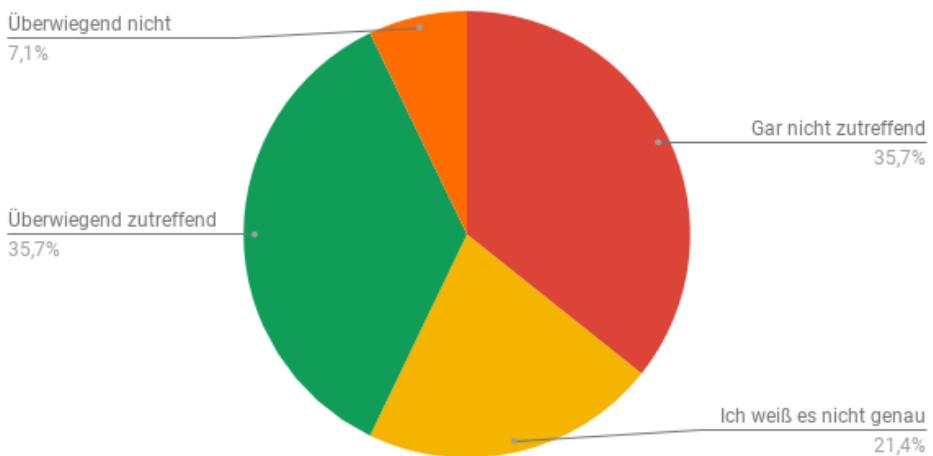
Für die Hälfte der Studenten stellt selbständiges Arbeiten mindestens überwiegend kein Problem dar, 29% sehen sich vollständig dazu in der Lage. Die andere Hälfte ist sich mindestens unsicher, 21% sehen sich überwiegend nicht zu selbstständigem Arbeiten in der Lage.

Das Einschätzen von Zeitaufwand (Meilensteine, Teilziele) fällt mir leicht



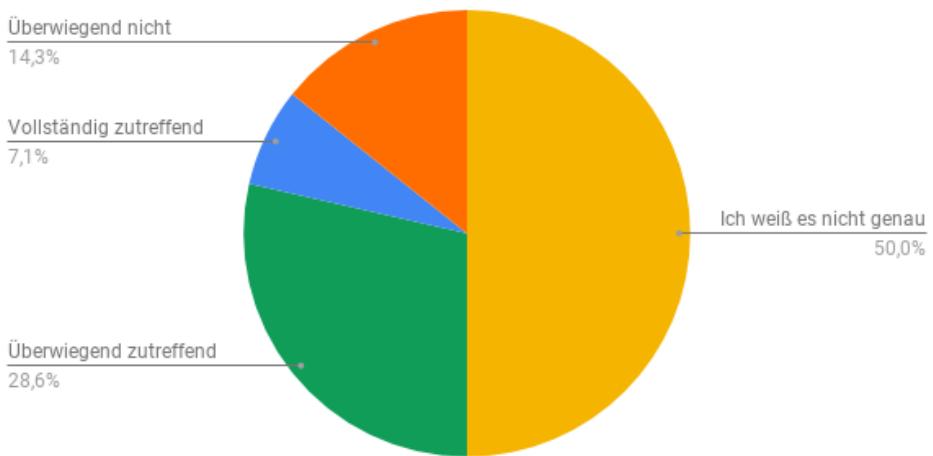
50% der Befragten fällt es mindestens überwiegend schwer, 14% auch komplett schwer, Zeitaufwand einzuschätzen. Etwa 20% fühlen sich unsicher, vom Rest fühlen sich etwa 7% überwiegend sicher dabei, ca. 21% auch vollständig sicher.

Ich fühle mich durch das Bearbeiten der Bachelorarbeit
überfordert



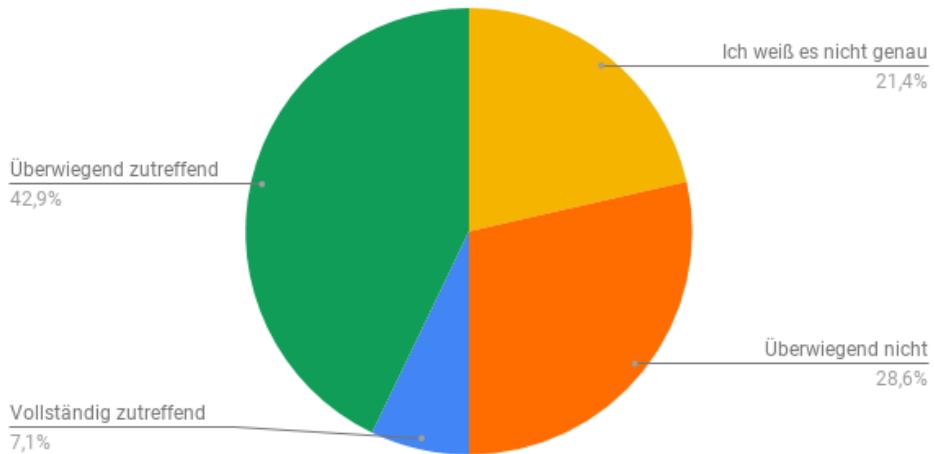
Etwa 35% der Befragten fühlen sich überwiegend durch das Bearbeiten der Bachelorarbeit überfordert, genauso viele fühlen sich gar nicht überfordert. Ca. 21% sind sich unsicher und der Rest fühlt sich überwiegend nicht überfordert.

Ich weiß ganz genau, was bei der Bearbeitung meiner Abschlussarbeit von mir verlangt wird und bei welchen



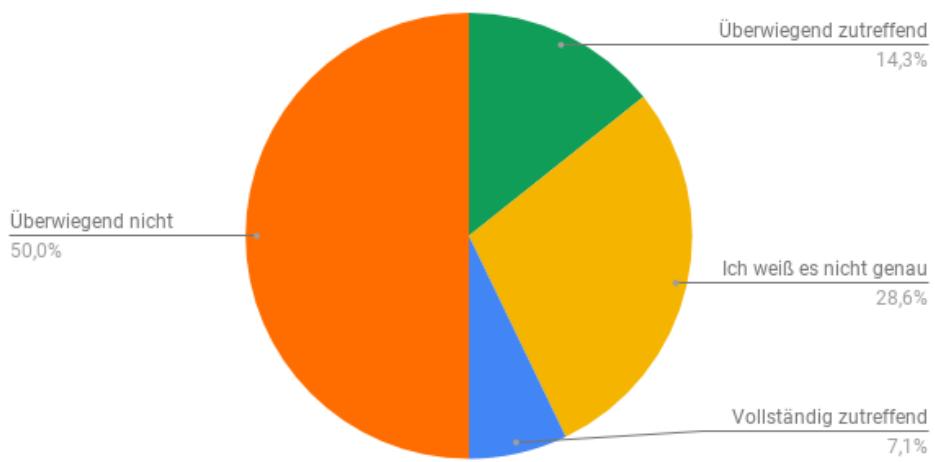
Die Hälfte der Studierenden ist sich unsicher, was von ihnen verlangt wird. Ca. 14% wissen es überwiegend nicht. Der Rest weiß es mindestens überwiegend, etwa 7% weiß es vollständig.

Ich kenne mich gut mit den Formalien zur Bachelorarbeit aus
(Anmeldung, Ablauf, Zulassungsanforderungen)



Die Hälfte der Befragten kennt sich mindestens nicht genau aus, ca. 28% davon kennt sich überwiegend nicht mit den Formalien zur Bachelorarbeit aus. Die andere Hälfte kennt sich mindestens überwiegend aus, 7,1% auf vollständig.

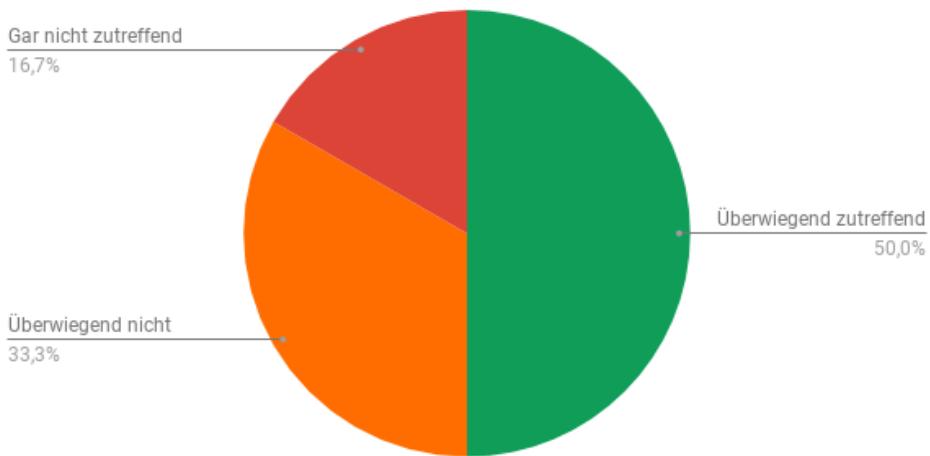
Ich weiß, wie man korrekt wissenschaftlich arbeitet und vorgeht (Literaturarbeit, Quellenangaben)



Die Hälfte der Studierenden kennt sich überwiegend nicht mit wissenschaftlichem Arbeiten aus. Etwa 29% sind sich dabei unsicher, der Rest weiß mindestens überwiegend, 7,1% auch vollständig, wie man wissenschaftlich arbeitet.

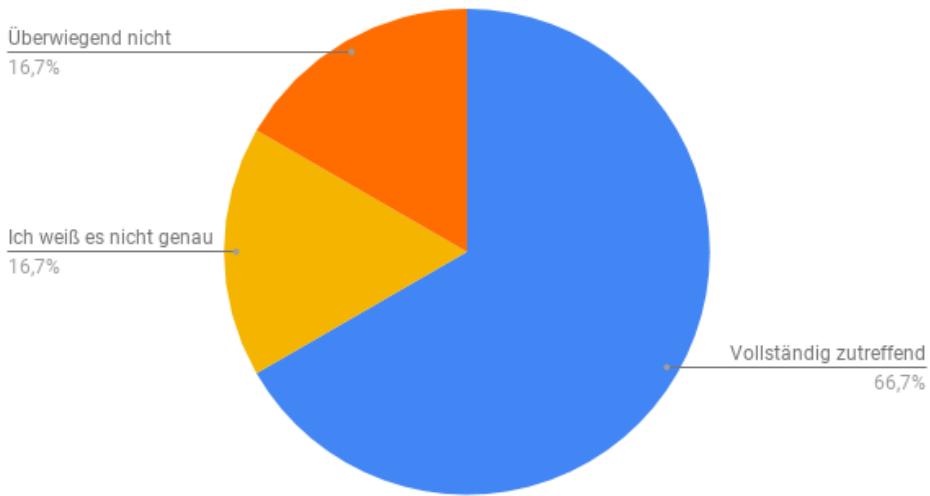
Nach Abschluss der Bachelorarbeit

Das Studium hat mich auf das Bearbeiten der Bachelorarbeit sehr gut vorbereitet



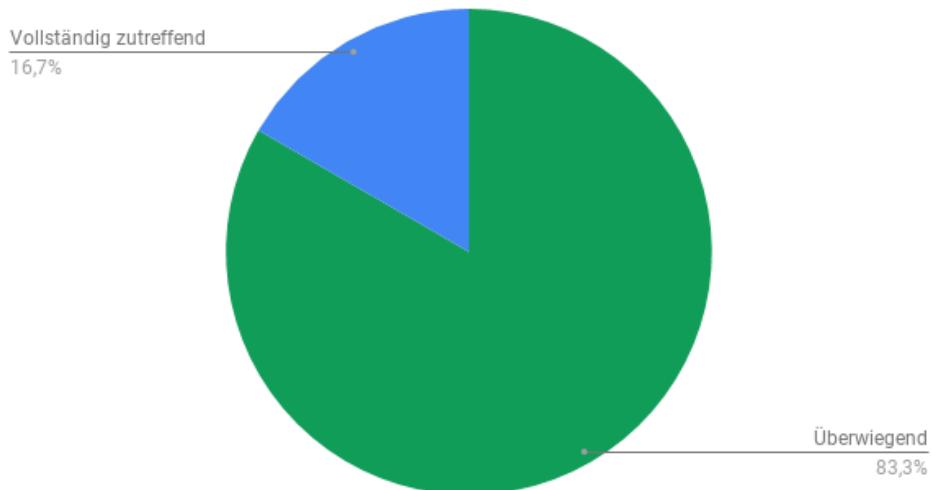
50% der Absolventen fühlten sich durch das Studium überwiegend auf die Bachelorarbeit vorbereitet. Ein Drittel fühlte sich überwiegend nicht, und der Rest fühlte sich gar nicht vorbereitet.

Es fiel mir leicht, das Thema meiner Bachelorarbeit zu finden



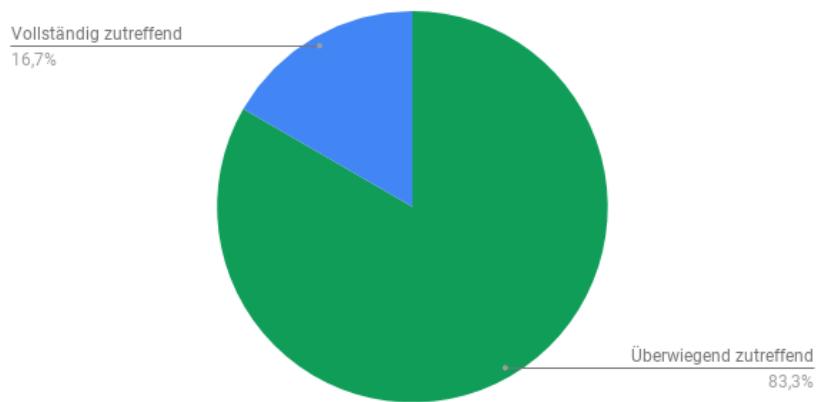
Zwei Drittel der Befragten fiel es leicht, ein Thema für die Bachelorarbeit zu finden. Der Rest war sich unsicher, oder ihm fiel es überwiegend nicht leicht.

Das selbständige Vorgehen stellte kein Problem für mich dar



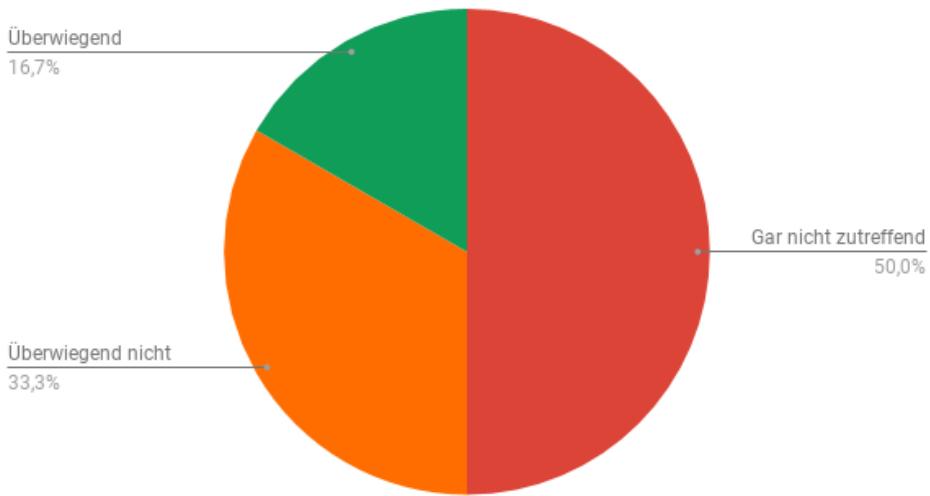
Alle Befragten fühlten sich mindestens überwiegend sicher beim selbstständigen Vorgehen, ca. 17% fühlten sich auch vollständig sicher.

Das Einschätzen von Zeitaufwand (Meilensteine, Teilziele) fiel mir leicht



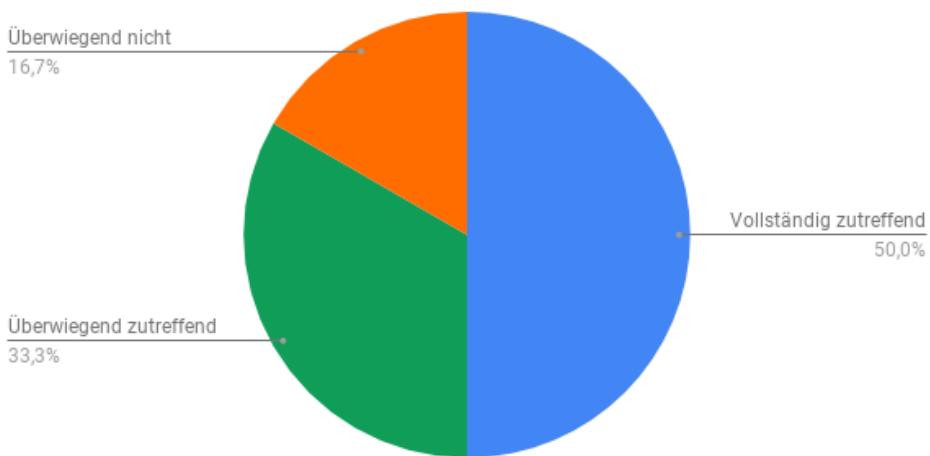
Alle Befragten fühlten sich mindestens überwiegend sicher beim Einschätzen von Zeitaufwand, ca. 17% fühlten sich auch vollständig sicher.

Das Fertigstellen der Bachelorarbeit hat mich überfordert



Die Hälfte der Absolventen hat sich gar nicht mit der Bachelorarbeit überfordert gefühlt. Ein Drittel fühlte sich überwiegend nicht überfordert und der Rest fühlte sich überwiegend überfordert.

Die Benotung meiner Abschlussarbeit entsprach genau meinen Erwartungen



Die Benotung der Arbeit entspricht bei 50% der Absolventen genau der Erwartung, bei einem Drittel überwiegend und beim Rest überwiegend nicht.

Literatur

- [1] *Actions*. (Zugriff Juni 8, 2018). URL: <https://redux.js.org/basics/actions>.
- [2] (Prof. Dr. rer. nat. Andreas Hanemann). “Anleitung zum Erstellen einer Abschlussarbeit”. Herausgegeben an Studenten der Fachhochschule Lübeck, die das Bachelorseminar besuchen.
- [3] *Announcing Flutter beta 1: Build beautiful native apps*. Februar 27, 2018 (Zugriff Juni 3, 2018). URL: <https://developers.googleblog.com/2018/02/announcing-flutter-beta-1.html>.
- [4] *Apple Homepage*. (Zugriff Juni 10, 2018). URL: <https://www.apple.com/>.
- [5] Helmut Balzert. *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering* -. Berlin Heidelberg New York: Springer-Verlag, 2010. ISBN: 978-3-827-42247-7.
- [6] Richard Bartle. “Hearts, clubs, diamonds, spades: Players who suit MUDs”. In: *Journal of MUD research* 1.1 (1996), S. 19.
- [7] Ivo Blohm und Jan Marco Leimeister. “Gamification: Gestaltung IT-basierter Zusatzdienstleistungen zur Motivationsunterstützung und Verhaltensänderung”. In: *Wirtschaftsinformatik* 55.4 (2013), S. 275–278.
- [8] *Build beautiful native apps in record time*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/>.
- [9] *Building Layouts in Flutter*. Februar 27, 2018 (Zugriff Juni 3, 2018). URL: <https://flutter.io/tutorials/layout/>.
- [10] *Completer<T> class*. (Zugriff Juni 16, 2018). URL: <https://api.dartlang.org/stable/2.0.0/dart-async/Completer-class.html>.
- [11] *Cupertino (iOS-style) Widgets*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/widgets/cupertino/>.
- [12] *Dart Homepage*. (Zugriff Juni 10, 2018). URL: <https://www.dartlang.org/>.
- [13] Sebastian Deterding u. a. “Gamification: Toward a definition”. In: *CHI 2011 gamification workshop proceedings*. Bd. 12. Vancouver BC, Canada. 2011.
- [14] *Flutter Local Notification Plugin*. (Zugriff Juni 16, 2018). URL: https://pub.dartlang.org/packages/flutter_local_notifications.
- [15] *Flutter Performance Profiler*. (Zugriff Juni 10, 2018). URL: <https://flutter.io/ui-performance/>.
- [16] *Flutter Redux*. (Zugriff Juni 8, 2018). URL: https://pub.dartlang.org/packages/flutter_redux.
- [17] *Flutter System Architecture*. April 23, 2017 (Zugriff Juni 3, 2018). URL: https://docs.google.com/presentation/d/1cw7A4HbvM_Abv320rVgPVGiUP2msVs7tfGbkgdrTy0I/edit#slide=id.p.

- [18] *Foundation Library*. (Zugriff August 6, 2018). URL: <https://docs.flutter.io/flutter/foundation/foundation-library.html>.
- [19] *Foundation Library*. (Zugriff Juni 3, 2018). URL: <https://docs.flutter.io/flutter/foundation/foundation-library.html>.
- [20] *Get Started*. (Zugriff Juni 10, 2018). URL: <https://flutter.io/get-started/>.
- [21] *How to Code Unlockable Achievements for Your Game (A Simple Approach)*. (Zugriff August 16, 2018). URL: <https://gamedevelopment.tutsplus.com/tutorials/how-to-code-unlockable-achievements-for-your-game-a-simple-approach--gamedev-6012>.
- [22] *Introduction*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/faq/#introduction>.
- [23] *iOS Design Themes*. (Zugriff Juni 3, 2018). URL: <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/>.
- [24] *Material Components Widgets*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/widgets/material/>.
- [25] *Material Design*. (Zugriff Juni 3, 2018). URL: <https://material.io/design/>.
- [26] Philipp Mayring. *Qualitative Inhaltsanalyse - Grundlagen und Techniken*. Langensalza: Beltz, 2015. ISBN: 978-3-407-25730-7.
- [27] Elisa D Mekler u. a. “Disassembling gamification: the effects of points and meaning on user motivation and performance”. In: *CHI'13 extended abstracts on human factors in computing systems*. ACM. 2013, S. 1137–1142.
- [28] Elisa D Mekler u. a. “Do points, levels and leaderboards harm intrinsic motivation?: an empirical analysis of common gamification elements”. In: *Proceedings of the First International Conference on gameful design, research, and applications*. ACM. 2013, S. 66–73.
- [29] *Mobile World Congress*. (Zugriff August 6, 2018). URL: <https://www.mobileworldcongress.com/about/>.
- [30] Nicolas Pflanzl. *Gamification for business process modeling*. 2018.
- [31] *Pixabay*. (Zugriff August 18, 2018). URL: <https://pixabay.com/de/abzeichen-blau-button-aufkleber-150755/>.
- [32] *Read Me*. (Zugriff Juni 8, 2018). URL: <https://redux.js.org/>.
- [33] *Read Me*. (Zugriff Juni 8, 2018). URL: <https://reduxframework.com/>.
- [34] *Reducers*. (Zugriff Juni 8, 2018). URL: <https://redux.js.org/basics/reducers>.
- [35] *Redux (JavaScript-Bibliothek)*. (Zugriff Juni 8, 2018). URL: [https://de.wikipedia.org/wiki/Redux_\(JavaScript-Bibliothek\)](https://de.wikipedia.org/wiki/Redux_(JavaScript-Bibliothek)).
- [36] Byron Reeves und J Leighton Read. *Total engagement: How games and virtual worlds are changing the way people work and businesses compete*. Harvard Business Press, 2009.
- [37] Michael Sailer. *Die Wirkung von Gamification auf Motivation und Leistung - Empirische Studien im Kontext manueller Arbeitsprozesse*. Berlin Heidelberg New York: Springer-Verlag, 2016. ISBN: 978-3-658-14309-1.

- [38] *Skia Graphics Library*. (Zugriff Juni 3, 2018). URL: <https://skia.org/>.
- [39] *Store*. (Zugriff Juni 8, 2018). URL: <https://redux.js.org/basics/store>.
- [40] Susanne Strahringer und Christian Leyh. *Gamification und Serious Games - Grundlagen, Vorgehen und Anwendungen*. Berlin Heidelberg New York: Springer-Verlag, 2017. ISBN: 978-3-658-16742-4.
- [41] *Technical Overview*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/technical-overview/>.
- [42] *Technology*. (Zugriff Juni 3, 2018). URL: <https://flutter.io/faq/>.
- [43] *Welcome to the Engine Wiki*. (Zugriff Juni 3, 2018). URL: <https://github.com/flutter/engine/wiki>.
- [44] Kevin Werbach und Dan Hunter. *For the win: How game thinking can revolutionize your business*. Wharton Digital Press, 2012.
- [45] *What's Revolutionary about Flutter*. (Zugriff Juni 3, 2018). URL: <https://hackernoon.com/whats-revolutionary-about-flutter-946915b09514>.