



HOCHSCHULE BREMEN  
FAKULTÄT 4 – ELEKTROTECHNIK UND INFORMATIK  
INTERNATIONALER STUDIENGANG MEDIENINFORMATIK (B.Sc.)

## EXPOSÉ

---

### Human Centred Design und agile Entwicklungsmethoden

– Durchführung und Evaluation anhand der MyCompetence  
Applikation der Lufthansa Industry Solution TS GmbH –

---

Dr. Andreas Teufel Teschke (5034226)  
mit Prof. Dr. Martin Hering

5. September 2018 (Version 1.01)

# 1 Einleitung

Diese Bachelorarbeit beschreibt eine mögliche Integration des User Centered Design (UCD)-Prozesses in agile Methode am Beispiel Scrum. Im Rahmen einer Überarbeitung der DIN EN ISO 9241 wurde der Prozess 2006 von „User Centered Design“ in „Human Centered Design“ umbenannt. In der Literatur wird jedoch noch häufig der Begriff „User Centered Design“ verwendet. Da beide Begriffe analog genutzt werden wurde auf die, zur Zeit noch, ungebräuchliche Verwendung des offiziellen Prozessnamens verzichtet. Im ersten Teil dieser Arbeit wird der UCD-Prozess und einige der darin verwendeten Methoden vorgestellt. Der zweite Teil erläutert den Scrum-Prozess. Im abschließenden dritten Teil wird versucht, eine mögliche Verbindung der beiden Prozesse zu zeigen und an Hand eines Beispiels erläutert. Für die Erstellung dieser Bachelorarbeit wurde der englische Scrum-Guide benutzt. Die jeweiligen Begriffe für Meetings, Rollen, etc. wurden daher nicht übersetzt. Die LH IND TS GmbH wurde durch einen Kunden beauftrag eine Analyseapplikation neu zu entwickeln. Der Kunde vertritt über 3000 aus metallverarbeitenden Unternehmen. Auf Grund von Größe und Umfang des Projektes (geschätzt etwa 900 Personentage) wurde zusammen mit dem Auftraggeber vereinbart, das Projekt mit Scrum als Projektmanagement-Framework umzusetzen. Da das Produkt, neben dem technischen Teil (Verfügbarkeit und Performance), für die Benutzer leicht zu bedienen sein sollte, wurde bereits in der Angebotsphase die Integration des Bereiches UCD empfohlen. Der UCD-Bereich der Cocomore AG ist zuständig für die konzeptionelle Entwicklung von benutzerfreundlichen Bedienkonzepten. Der Kunde (Kunde) folgte dieser Empfehlung.

## 2 Problemstellung und Lösungsansatz

Der Kunde hatte in der Vergangenheit mit mindestens drei verschiedenen Agenturen ein Redesign seines E-Commerce-Angebots durchgeführt. Die Ergebnisse waren nicht das was sich der Kunde erhofft hatte. Es sollte eine Performance-Steigerung sowie eine Erhöhung der Besucherzahlen erreicht werden. Beide Anforderungen konnten in den vorangegangenen Redesigns nicht erfüllt werden.

In den vorangegangenen Redesigns wurde auf klassische Projektmanagement-Frameworks zurückgegriffen. Der Kunde war von der Idee Scrum zu nutzen, und den Entwicklungsprozess aktiv zu begleiten und mitzugestalten zu können, leicht zu überzeugen. In den vorherigen Redesigns konnte der Kunde erst sehr spät Entwicklungsergebnisse überprüfen und aufgetretene Probleme zögerten eine Fertigstellung immer hinaus.

Ein weiterer Grund für den Einsatz von Scrum lag in dem großen Projektumfang (geschätzt etwa 900 Personentage). Der zuständige Projektmanager, sowie der Leiter des IT-Bereiches der Cocomore AG, schlugen dem Kunden auch aus diesem Grund Scrum als Projektmanagementtool vor. Durch die Verwendung von Scrum konnte der Kunde ständig den Fortschritt des Projektes überprüfen und bei Problemen eingreifen. Der Einsatz linearer Entwicklungsmethoden hätte aus Sicht des Projektmanagers und des IT-Leiters evtl. zu einer Wiederholung der Probleme aus den vorangegangenen Redesigns geführt.

### 2.1 Problemstellung des

#### 2.1.1 Was ist Human Centered Design?

Human Centered Design (UCD) ist ein Prozess um eine möglichst hohe positive User Experience (Nutzungserlebnis) zu erreichen. Die User Experience wird durch die Usability in starkem Masse beeinflusst.

Bei der Konzeption von Produkten nach UCD wird der Benutzer in den Mittelpunkt des Entwicklungsprozesses gestellt. Es wird dabei versucht, den Benutzer während der gesamten Entwicklung eines Projektes in den Entwicklungsprozess einzubinden und die Benutzeroberfläche und Interaktionen so zu optimieren, dass die fertige Anwendung den Nutzer in seiner Aufgabe optimal unterstützt.

Der äußere orange Rahmen symbolisiert die Anforderungen an einen benutzerorientierten Entwicklungsprozess (User Centered Design). Der grüne Rahmen steht für das Usability Engineering, das einen Ausschnitt des User Centred Design bildet. Usability Engineering konzentriert sich zwar auf das Überprüfen und Optimieren der Usability des Produktes („während der Nutzung“), ist jedoch nicht unabhängig davon, was vor und nach der Nutzung passiert: bereits generierte Erwartungen und Vorerfahrungen sowie während der Nutzung neu gewonnene Erfahrungen spielen ebenfalls eine wichtige Rolle. User Centered Design bietet den Rahmen in dem die Methoden des Usability Engineering angewandt werden.

#### 2.1.2 Die vier Phasen des User Centered Design

1. Analyse des Nutzungskonzepts
2. Definition der Anforderungen
3. Konzeption und Entwurf
4. Evaluation

#### 2.1.3 Agile Entwicklungsmethoden

Agile Entwicklungsmethoden basieren auf der Idee, durch häufige Iterationen die Entwicklungszeit für Projekte zu verkürzen und ein flexibles System zur Problemlösung zu benutzen. Anders als bei linearen Entwicklungsmethoden wird ein zu entwickelndes System nicht vollständig geplant bevor die eigentliche Umsetzung erfolgt. Viel mehr ist das Ziel agiler Entwicklung, dem Kunden nach jeder Iteration ein funktionierendes Produkt zu liefern. Der Kunde wird bereits sehr früh in den Entwicklungsprozess eingebunden und begleitet diesen bis zum fertigen Produkt.

Seit den Anfängen der agilen Entwicklung haben sich verschiedene Methoden etabliert, einige der bekanntesten sind

- Extreme Programming
- Feature Driven Development
- Crystal (oder auch Methodensammlung)

- Usability Diven Debelopment
- Scrum

Agile Entwicklungsmethoden bieten einige Vorteile, sowohl für die Entwickler, als auch für den Kunden:

- Agile Entwicklungsmethoden lassen sich auf nahezu alle Projekte anwenden
- Frühe Beteiligung des Kunden am Entwicklungsprozess
- Nach jeder Iteration erhält der Kunde ein funktionsfähiges Produkt
- Mehr Freiheit für den Entwickler bei der eigentlichen Umsetzung / Problemlösung
- Auf Grund der Flexibilität können Änderungen am Projekt schnell und meist auch kostengünstig realisiert werden.

Die bei linearen Entwicklungsmethoden teilweise sehr aufwendigen Iterationen werden von Anfang an in den Prozess integriert und dienen der dynamischen Entwicklung eines Projektes.

### 2.2 Lösungsansatz

In diesem Abschnitt wird zusammengefasst, was bei der Lösung des zu bearbeitenden „Problems“ – nämlich der Erstellung eines Exposés bzw. der Anfertigung einer Bachelor- oder Masterarbeit – in inhaltlicher, methodischer und formaler Hinsicht zu beachten ist. Weitere Informationen zum wissenschaftlichen Arbeiten finden sich in entsprechenden Ratgebern (siehe z. B. [1]).

**Inhaltliches: Was ist eigentlich eine geeignete Problemstellung?** Problemstellungen für eine Bachelor- oder Masterthesis können durch Ausschreibungen eines Professors gegeben sein, aus persönlichem Interesse erwachsen oder – wie häufig anzutreffen – aus einem Unternehmen stammen, bei dem man gerade sein Praktikum abgeschlossen hat. Gerade bei letzteren ist es häufig der Fall, dass die vorliegenden Probleme nur auf die Belange des Unternehmens zugeschnitten und damit sehr speziell sind („Implementieren Sie doch noch Modul X für unsere Anwendung!“). Zudem bieten entsprechende Probleme in konzeptioneller Hinsicht oftmals wenig Spielraum.

Eine wichtige Anforderung an ein geeignetes Thema für eine Thesis ist, dass ein hinreichend *allgemeines* Problem betrachtet wird, dessen Lösung dann für viele interessant sein kann. Ein solches allgemeines Problem kann vielfach ausgehend von einem speziellen Fall (z. B. aus dem erwähnten Praktikum) durch Abstraktion, d. h. Generalisierung gewonnen werden. Im Rahmen der Arbeit sollte dann dieses generalisierte Problem analysiert und ein möglichst allgemeines Konzept entwickelt werden, das dann, z. B. unter Verwendung einer bestimmten Plattform und ggf. wieder eingegrenzt auf das speziellere Problem, im Sinne

eines „Proof of Concepts“ prototypisch umgesetzt werden kann, um letztendlich zu zeigen, dass das (allgemeine) Konzept funktioniert.

Sie (und evtl. beteiligte Unternehmen) sollten aber davon ausgehen, dass das Ergebnis einer Bachelor- oder Masterarbeit typischerweise kein fertiges Produkt ist, das im Unternehmen direkt eingesetzt werden kann. In der Regel ist es eher so, dass auf Basis der Thesis und der prototypischen Implementierung anschließend ein System entwickelt werden kann, das auch produktiv eingesetzt werden kann.

**Methodisches: Wie gehe ich denn nun vor?** Die Bachelor- oder Masterthesis zeigt ihre Fähigkeit, eine Aufgabe mit wissenschaftlichen Mitteln in begrenzter Zeit zu bearbeiten. Ein wissenschaftliches Vorgehen sieht immer einen *Analyseteil* („Was sind die Anforderungen, welche Rahmenbedingungen sind zu beachten, was gibt es für Möglichkeiten?“) und einen *Syntheseteil* („Was habe ich weshalb wie gemacht?“) vor. Im Folgenden ein grober Gliederungsrahmen als Beispiel<sup>1</sup>:

1. Einleitung: Hier führt eine kurze Beschreibung des Problemfeldes bzw. der Motivation, der konkreten Aufgabe(n), der angestrebten Vorgehensweise und ggf. auch ein erster Blick auf mögliche Lösungsansätze in das Thema der Arbeit ein („Ziel der Arbeit ist ...“). Danach sollte noch ein Abschnitt folgen, der kurz in Prosa die weitere Struktur der Arbeit und die wesentlichen Inhalte der Kapitel darstellt. Danksagungen (und Verwünschungen) gehören ins Vorwort.
2. Analyse: Der „Fleißteil“ mit viel Literaturrecherche, z. B. zu Fragestellungen wie
  - Problemanalyse: Welche Anforderungen sollen erfüllt werden?
  - Verwandte Arbeiten: Welche alternativen Lösungsansätze gibt es schon in diesem Problemfeld? Welche Stärken und Schwächen haben diese Ansätze?
  - Grundlagen: Welche Konzepte und Methode(n) sind für diese Arbeit relevant? Welche (Software-)Werkzeuge können benutzt werden?

Der Analyseteil ist ggf. in mehrere Kapitel unterteilt.

3. Synthese: Hier geht es um das eigene „Werk“: Konzeption und Realisierung, gerne in zwei Kapiteln (in dieser Reihenfolge!).
  - Konzeption: Abstrakte, aber dennoch möglichst klare Spezifikation Ihres Lösungsansatzes. Mithilfe von Text und Diagrammen (z. B. zu Interaktionsstruktur, Software-Architektur, Datenbankschema) wird ein Modell Ihrer Lösung (Struktur und Dynamik) beschrieben und darüber hinaus die zu deren Erstellung verwendeten Methoden.
  - Realisierung (natürlich nur, wenn was realisiert wurde): Beschreibung ausgewählter und interessanter Aspekte einer konkreten Umsetzung Ihres Konzepts auf einer (Programmier-)Plattform. Dazu können beispielsweise der Einsatz von besonderen Werkzeugen, Optimierungen oder spezielle Problemlösungen gehören, die aufgrund der gewählten Plattform erforderlich werden und daher nicht

---

<sup>1</sup>Ein etwas konkreteres Muster für eine (generische) Gliederung findet sich in Kapitel 4.

bereits durch das Konzept abgedeckt sind.

Ein gutes Konzept zeichnet sich vielfach dadurch aus, dass der Realisierungsteil eher klein bleiben kann, weil die Realisierung ja eben „nur“ die Umsetzung eines Konzepts in einer Programmierplattform darstellt und daher nicht jedes Detail aufgeführt werden muss (dafür gibt es als Anlage eine CD/DVD). Dennoch gilt: Wissenschaftliche Ergebnisse müssen reproduzierbar sein. Daher ist es wichtig, dass die Arbeit möglichst alle relevanten Details für eine Re-Implementierung und Reproduktion der Ergebnisse enthält.

4. Evaluation: Teil einer wissenschaftlichen Arbeit ist die kritische Überprüfung und Diskussion der erzielten Ergebnisse (Konzept und Realisierung), wenn möglich auch der Vergleich mit bestehenden Ansätzen. Welche Aspekte dabei im Vordergrund stehen, hängt vom Themenfeld Ihrer Arbeit ab: Performance einer Anwendung, Usability eines UI, Sicherheit einer Schnittstelle, Flexibilität einer Software-Architektur... Machen Sie zum Abschluss Ihrer Arbeit deutlich, welche Ziele Sie zu welchem Grad erreicht haben (auch wenn vielleicht nicht alles zum Besten steht).
5. Zusammenfassung und Ausblick: Das Schlusskapitel führt die wichtigsten Ergebnisse der Arbeit auf und zeigt, wo es weitergehen kann oder was noch verbessert werden müsste. Ein (eiliger) Leser von ausschließlich Einleitung sowie Zusammenfassung und Ausblick sollte bereits eine Vorstellung davon haben, welche Ziele die Arbeit verfolgt hat und welche davon inwieweit erreicht wurden.

Wie bereits in den Anmerkungen zum Analyseteil angedeutet, kommt der Literaturrecherche eine besondere Bedeutung zu: Welche verwandten Arbeiten gibt es? Wo finde ich Informationen zu relevanten Konzepten, Methoden oder Technologien? Entsprechende Literatur ist zu recherchieren und – im kleineren Umfang – bereits im Exposé zu referenzieren.

Recherchearbeit ist ein oftmals aufwändiger und langwieriger Prozess, bei dem man in der Regel erst allmählich durch das Lesen vieler Quellen und das Verfolgen von Referenzen zu „guter“ Literatur gelangt. Lassen Sie sich nicht verleiten, sich auf bequem erreichbare Online-Quellen wie Blogs oder Wikipedia zu verlassen. Auch wenn Wikipedia immerhin eine Art Qualitätssicherung durch die „Crowd“ bietet und teilweise auch für den Einstieg brauchbare Ergebnisse liefert, stellen derartige Quellen in aller Regel Informationen zur Verfügung, die nicht objektiv von Fachexperten auf ihre Richtigkeit geprüft wurden (Sie werden im Web auch viele Quellen finden, die scheinbar glaubhaft „belegen“, dass es keinen Klimawandel gibt oder die Evolution nie stattgefunden hat). Online-Quellen – sofern es sich dabei nicht um (auch) im Netz auffindbare wissenschaftliche Dokumente wie Konferenzbeiträge oder technische Berichte mit Autoren, Titel, Organisation und Datum handelt – sind keine Literatur im engeren Sinne und sollten (sofern sie denn verwendet werden) bevorzugt in einem gesonderten Verzeichnis gelistet werden. Besser ist es in jedem Fall, sich auf Fachbücher [2] und wissenschaftliche Beiträge [3] zu Konferenzen zu konzentrieren. Hier kann man davon ausgehen, dass diese Publikationen einen Review-Prozess durchlaufen haben und somit durch Fachexperten geprüft und für (hinreichend) gut befunden wurden.

Für die Recherche gibt es eine Reihe von Quellen, die Sie als Studierende der Hochschule Bremen nutzen können:

- Staats- und Universitätsbibliothek Bremen (SuUB)<sup>2</sup>,
- digitale Bibliotheken von Fachgesellschaften wie die ACM Digital Library<sup>3</sup> und IEEE Xplore Digital Library<sup>4</sup>
- Google Scholar<sup>5</sup>

Sofern Sie im Hochschulnetz sind oder sich von zuhause über das VPN der Hochschule Bremen mit dem Hochschulnetz verbunden haben, können Sie online verfügbare Texte der SuUB runterladen und auch die genannten digitalen Bibliotheken kostenlos nutzen.

Für die Literaturverwaltung bieten sich Werkzeuge wie z. B. *Mendeley*<sup>6</sup> oder *Zotero*<sup>7</sup> an. Diese bieten neben der Möglichkeit zur Verwaltung von Literatur in der Cloud und den Export von bibliographischen Einträgen in verschiedene Zielformate weitere Features wie z. B. die Möglichkeit zur Suche nach Literatur innerhalb des Werkzeugs.

Von besonderer Bedeutung in einer wissenschaftlichen Arbeit ist es, dass Sie Ihre Bewertungen (z. B. von Lösungsansätzen) grundsätzlich deutlich machen und daraus resultierende Entscheidungen für den Leser nachvollziehbar begründen, sowohl in der Analyse als auch in der anschließenden Konzeption und Realisierung. Stellen Sie sich dazu stets Fragen Ihres Gutachters wie „Warum haben Sie das so gemacht?“ oder „Welche Alternativen haben Sie betrachtet?“ und beantworten Sie die Fragen in der Arbeit (und nicht erst im Kolloquium).

**Formales:** Was muss ich alles beim Schreiben eines wissenschaftlichen Textes beachten? An wissenschaftliche Texte werden eine Reihe von formalen Anforderungen gestellt, die letztendlich das Ziel haben, die Lesbarkeit und Verständlichkeit des Textes zu erhöhen [4]. Einige davon werden hier aufgeführt.

**Schreibstil:**

- Wissenschaftliche Distanz: Schreiben Sie nicht in der ersten Person (also nicht „ICH habe mich dafür entschieden, dies und das so und so zu machen.“), sondern wählen Sie stattdessen lieber eine passive, distanzierte Form („Im Rahmen dieser Arbeit wurde entschieden, ...“ oder „Die Entscheidung, ..., war begründet durch ...“).
- Inhaltliche Übernahmen aus der Literatur müssen selbstverständlich durch entsprechende Quellenangaben kenntlich gemacht werden.
- Verzichten soweit wie möglich auf Zitate, sondern finden Sie Ihre eigenen Worte für Inhalte, die Sie aus der Literatur übernehmen. Zitate bieten sich nur dann an, wenn die wörtliche Wiedergabe von Inhalten von Bedeutung ist (und das ist zumeist nicht der Fall). Zitate müssen durch Anführungszeichen und Quellenangabe kenntlich gemacht werden.
- Plagiate sind jedwede Übernahme von Inhalten anderer, die *nicht* als solche gekenn-

---

<sup>2</sup><https://www.suub.uni-bremen.de>

<sup>3</sup><https://dl.acm.org>

<sup>4</sup><https://ieeexplore.ieee.org/Xplore/home.jsp>

<sup>5</sup><https://scholar.google.de>

<sup>6</sup><https://www.mendeley.com/>

<sup>7</sup><https://www.zotero.org>

zeichnet sind. Verstoßen Sie dagegen, führt das zwingend zum Nichtbestehen, ggf. sogar zur Exmatrikulation<sup>8</sup>. Also: nutzen Sie das (gesicherte) Wissen anderer, aber weisen Sie unbedingt auf die Quellen hin.

- Prüfen Sie Ihre Rechtschreibung! Und falls Sie da nicht über das richtige Händchen verfügen sollten: es gibt eine Reihe von Programmen mit einer automatischen Rechtschreibüberprüfung. Keine Ausreden!
- Viele Seiten sind kein Zeichen von Qualität! Gewürdigt wird, wenn das Wesentliche kompakt dargestellt und auf den Punkt gebracht wird, ohne dabei die Nachvollziehbarkeit zu beeinträchtigen. Das kann z. B. durch die Darstellung von Zusammenhängen durch selbst erstellte Grafiken, Übersichtstabellen und Spiegelstriche statt Prosa *unterstützt* werden.

Als geeigneter Umfang für eine Bachelorarbeit haben sich 40-60 Seiten gezeigt, bei einer Masterarbeit 60-80 Seiten (jeweils ohne Anhänge).

Referenzen:

- *Jede* Abbildung, Tabelle und jedes Listing etc. bekommt eine Kategorie, Nummer und Über- oder Unterschrift (z. B. *Listing 1: Hocheffizienter Suchalgorithmus*). Da Tabellen und Listings über eine Seite hinausgehen können, werden ihre Beschriftungen oberhalb der Tabelle bzw. des Listings angegeben, um frühzeitig die Bedeutung der Tabelle bzw. des Listings deutlich zu machen. Abbildungen werden unterhalb beschriftet.
- *Jede* Abbildung, Tabelle und jedes Listing etc. muss mindestens einmal im Text referenziert werden und sollte in diesem Zusammenhang zudem im Text diskutiert werden, um die wesentliche Botschaft deutlich zu machen. (z. B. „Listing 1 zeigt *<Botschaft>*“ oder „*<Diskussion...>* (siehe Listing 1)“).
- *Jede* Literaturquelle muss mindestens einmal im Text referenziert werden. Die Angabe der Seitenzahl ist in der Informatik unüblich, schadet aber auch nicht.
- Wenn Sie URLs zu Organisationen wie der Object Management Group (OMG)<sup>9</sup>, Firmen wie z. B. Apple<sup>10</sup> oder Produkten wie Mendeley<sup>6</sup> angeben wollen, geschieht das sinnvollerweise in einer Fußnote. Online-Quellen [5], die „lesbare“ und mit Autor und Titel versehene Texte bezeichnen, gehören dagegen ins Literaturverzeichnis.

Listing 1: Hocheffizienter Suchalgorithmus

```
1 void bubblesort(int[] zahlen) {
2     for (int i=1; i<zahlen.length; i++) {
3         for (int j=0; j<zahlen.length-i; j++) {
4             if (zahlen[j] > zahlen[j+1]) {
5                 swap(zahlen[j], zahlen[j+1]);
6             }
7         }
8     }
9 }
```

---

<sup>8</sup>Siehe jeweils gültige Fassung der Bachelor- und Masterprüfungsordnung (Allgemeiner Teil) unter <http://www.hs-bremen.de/internet/de/hsb/hip/dokumente/po/>.

<sup>9</sup><https://www.omg.org>

<sup>10</sup><https://www.apple.com/de/>



```
8   }  
9 }
```

---

Struktur:

- Ein neues *Kapitel* (nicht Abschnitt oder Unterabschnitt) in einer Bachelor- oder Masterarbeit beginnt üblicherweise auf einer neuen Seite. Bei beidseitigem Layout beginnt ein neues Kapitel zudem auf einer ungeraden (rechten) Seite.
- Nach einer Überschrift folgt nicht direkt eine zweite (Unter-)Überschrift, sondern zumindest eine kurze Beschreibung dessen, was in den folgenden Unterabschnitten zu erwarten ist (vgl. Kapitel 2).
- Ein Kapitel oder Abschnitt mit weiterer Untergliederung enthält mindestens zwei Unterabschnitte. Oder anders formuliert: wenn es einen Abschnitt x.y.1 gibt, dann gibt es auch einen Abschnitt x.y.2 (ansonsten gibt es eben nur Abschnitt x.y)!
- Verzichten Sie auf Vorwärtsreferenzen! Es ist im Allgemeinen schlechter Stil, bereits in Kapitel 2 mittels eines „siehe Abschnitt 4.3.1“ auf Ergebnisse zu verweisen, die erst später folgen (Ausnahmen bestätigen die Regel).

Sonstiges:

- Sehr spezifische Fachbegriffe sollten in einem Glossar mit ein bis zwei Sätzen erklärt werden. Hilft auch dem Schreiber! Neben Literaturverzeichnis und Glossar sind noch Verzeichnisse für Abbildungen, Tabellen und Listings gebräuchlich.
- Wenn Abkürzungen verwendet werden sollen, muss die Abkürzung gemeinsam mit dem abzukürzenden Begriff explizit eingeführt werden. Also erst „Asynchronous JavaScript and XML (AJAX) ist ein ...“, bevor später dann nur noch mit „AJAX“ gearbeitet wird. Ein Abkürzungsverzeichnis kann unterstützen, ersetzt aber die explizite Einführung der Abkürzung im Text nicht.
- Code-Fragmente bitte nicht mittels Screenshot als Abbildung einbinden. Das ist zwar einfach, jedoch ist das Resultat oft optisch unschön und – und das ist wichtiger – spätestens im Ausdruck in der Regel nicht mehr lesbar (und auch nicht durchsuchbar). Fügen Sie Code-Fragmente wie in Listing 1 gezeigt als formatierte Texte ein.
- Skalieren Sie Abbildungen gleichen Typs (z. B. UML-Diagrammen) oder mit gleichen Fonts einheitlich. Ggf. erfordert das dann etwas Arbeit am Layout, um eine Abbildung mit vorgegebener Skalierung auf die Abmessungen einer Seite anzupassen.
- Zahlen bis einschließlich zwölf schreibt man üblicherweise aus. Also „die sieben Zwerge“, aber „die wilde 13“!
- *LaTeX* [6] ist schön und gerade im Wissenschaftsbetrieb der Informatik verbreitet, aber nicht Pflicht.

Die Erstellung des Exposés stellt grundsätzlich eine gute Möglichkeit dar, sich in das Textverarbeitungssystem der Wahl einzuarbeiten und dabei auch Features wie automatische Nummerierung (z. B. von Kapiteln, Abbildungen), Querverweise, Erstellung von Verzeichnissen, Literaturverwaltung etc. als Vorbereitung auf die Thesis praktisch

anzuwenden.

Und zu guter Letzt noch ein wichtiger Gedanke: Immer an den Leser denken! Der weiß noch nicht alles, was Sie wissen (oder im Laufe der Arbeit gelernt haben). Versuchen Sie daher, sich in seine Rolle hineinzuversetzen und führen Sie ihn möglichst so durch den Text, dass er Ihre Gedankengänge nach und nach verstehen kann.

### 3 Konkrete Aufgaben

Für die Erstellung eines Exposés sind die folgenden Aufgaben durchzuführen:

- Recherche: Ermittlung von Literatur zum wissenschaftlichen Arbeiten, Vergleich mit ähnlichen Dokumenten
- Anforderungsanalyse: Zusammenstellung von Anforderungen an das zu erstellende Exposé bzw. die Bachelor- oder Masterarbeit.
- Konzeption: Entwicklung eines Konzepts für das Exposé / die Thesis: Textverarbeitung / Textsatzsystem, Gliederung, Konventionen, ...
- prototypische Implementierung: Schreiben des Exposés bzw. der Thesis
- Evaluation: Bewertung und Diskussion der Ergebnisse (machen auch die Prüfer ;-))
- Qualitätssicherung: sorgfältige Überprüfung des Dokuments auf inhaltliche und formale Fehler

### 4 Vorläufige Gliederung

*Hinweis: Im Folgenden wird exemplarisch eine generische Gliederung gezeigt, die selbstverständlich abhängig vom Thema angepasst werden muss.*

Eigenständigkeitserklärung

Zusammenfassung / Abstract

1. Einleitung
  - 1.1. Problemfeld
  - 1.2. Ziele der Arbeit
  - 1.3. Lösungsansatz
  - 1.4. Aufbau der Arbeit
2. Anforderungsanalyse
  - 2.1. Funktionale Anforderungen
  - 2.2. Nicht-funktionale Anforderungen
  - 2.3. Zusammenfassung
3. Grundlagen und verwandte Arbeiten
  - 3.1. User Centered Design
  - 3.2. Grundlagenthema 2
  - 3.3. ...

- 3.4. Verwandte Arbeiten
- 4. Konzeption
  - 4.1. Entwurf einer Software-Architektur
  - 4.2. Design des User Interfaces
  - 4.3. Entwurf der Komponenten
    - 4.3.1. Komponente 1
    - 4.3.2. Komponente 2
    - 4.3.3. ...
  - 4.4. Spezifikation der Schnittstellen
  - 4.5. Zusammenfassung
- 5. Prototypische Realisierung
  - 5.1. Wahl der Realisierungsplattform
  - 5.2. Festlegung des Realisierungsumfangs
  - 5.3. Ausgewählte Realisierungsaspekte
    - 5.3.1. REST-API mit Bibliothek XYZ
    - 5.3.2. ...
  - 5.4. Qualitätssicherung
  - 5.5. Zusammenfassung
- 6. Evaluation
  - 6.1. Überprüfung funktionaler Anforderungen
  - 6.2. Überprüfung nicht-funktionaler Anforderungen
  - 6.3. Zusammenfassung
- 7. Zusammenfassung und Ausblick
  - 7.1. Zusammenfassung
  - 7.2. Ausblick

Anhang

A Anhang <Thema A>

B Anhang <Thema B>

...

Literaturverzeichnis

Ergänzende Anmerkungen:

- Wenn die Evaluation eher knapp ausfällt, kann sie ggf. auch zu einem Abschnitt des Kapitels 5 (Prototypische Realisierung) werden.
- Die eigentliche (nummerierte) Gliederung wird noch durch Eigenständigkeitserklärung, Zusammenfassung / Abstract, Verzeichnisse und evtl. Anhänge ergänzt:
  - In der Eigenständigkeitserklärung versichern Sie unter anderem, die Arbeit selbstständig, ohne unzulässige fremde Hilfen und unter Angabe aller verwendeten Quellen und Hilfsmittel angefertigt zu haben. Das Prüfungsamt der Hochschule Bremen stellt auf seiner Web-Seite ein Muster<sup>11</sup> bereit, das allerdings noch angepasst werden muss.
  - Zusammenfassung bzw. englischer Abstract enthalten jeweils nur einen Abschnitt, der den Inhalt der Arbeit und die erzielten Ergebnisse ohne gesonderte Einführung in das Thema zusammenfasst.
  - Neben dem Literaturverzeichnis sind noch Verzeichnisse für Abbildungen, Tabellen, Listings und Abkürzungen gebräuchlich. Ein Glossar kann helfen, spezielle und dem Leser im Allgemeinen nicht bekannte Fachbegriffe zu erläutern.
  - Anhänge können dazu dienen, der Arbeit wichtige Inhalte beizufügen, die für die Wiedergabe im Hauptteil nicht geeignet erscheinen, weil Sie z. B. nicht für jeden Leser relevant sind oder allgemein den Lesefluss stören.

## 5 Zeitplanung

Geplanter Starttermin<sup>12</sup>: 1. September 2018

Bearbeitungsdauer: 9 Wochen (bei Bachelorarbeiten) / 22 Wochen (bei Masterarbeiten)

Tabelle 1 stellt die geplanten Arbeitspakete und Meilensteine dar:

---

<sup>11</sup>[https://www.hs-bremen.de/mam/hsb/dezernat/d3/erklärung\\_über\\_das\\_eigenständige\\_erstellen\\_der\\_arbeit.doc](https://www.hs-bremen.de/mam/hsb/dezernat/d3/erklärung_über_das_eigenständige_erstellen_der_arbeit.doc)

<sup>12</sup>Hinweis: Das Prüfungsamt erbittet sich 14 Tage Vorlauf. Dementsprechend muss der Starttermin ausgehend vom Zeitpunkt der Anmeldung einer Bachelor- oder Masterarbeit mindestens zwei Wochen in der Zukunft liegen.

Tabelle 1: Arbeitspakete und Meilensteine

M1	Offizieller Beginn der Arbeit	01.09.2018
	<ul style="list-style-type: none"><li>• Einrichtung der Textverarbeitung</li><li>• Anforderungsanalyse</li><li>• Verfassen der Thesis: Kapitel 2 (Anforderungsanalyse)</li></ul>	1,5 Wochen
M2	Abschluss der Analysephase	11.09.2018
	<ul style="list-style-type: none"><li>• Recherche</li><li>• Verfassen der Thesis: Kapitel 3 (Grundlagen)</li></ul>	1,5 Wochen
M3	Abschluss der Recherchephase	21.09.2018
	<ul style="list-style-type: none"><li>• ...</li><li>• Verfassen der Thesis: ...</li></ul>	n Wochen
M4	Abschluss der ...phase	...
...		
My	Erste Fassung vollständige Thesis	27.10.2018
	<ul style="list-style-type: none"><li>• Korrekturlesen</li><li>• Drucken / binden lassen</li></ul>	1 Woche
Mz	Abgabe der Thesis	02.11.2018

## 6 Unterschriften

Student\_in:

Name: *<Vor- und Nachname>*

Adresse: *<Straße und Hausnummer, PLZ, Stadt>*

Telefon: *<Nummer, unter der man Sie bei wichtigen Fragen schnell erreichen kann.>*

E-Mail: *<mail@stud.hs-bremen.de>*

---

Unterschrift (Student\_in)

Erstgutachter\_in / Betreuer\_in: *<Prof. Dr. Ernst>*

---

Unterschrift (Erstgutachter)

Zweitgutachter\_in: *<Prof. Dr. Lustig>*

## Literatur

### Gedruckte Quellen

- [1] Kristina Folz und Detlef Jürgen Brauer. *Studi-SOS Bachelor- und Masterarbeit – Erste Hilfe fürs wissenschaftliche Arbeiten*. Verlag Wissenschaft und Praxis, 2017.
- [2] Erich Gamma u. a. *Design Patterns – Elements of Reusable Object-Oriented Software*. Addison Wesley, Professional Computing Series, 1995.
- [3] Artem Syromiatnikov und Danny Weyns. „A Journey Through the Land of Model-View-\* Design Patterns“. In: *Proceedings - Working IEEE/IFIP Conference on Software Architecture 2014, WICSA 2014*. Sydney, Australia: IEEE Computer Society, 2014, S. 21–30.
- [4] Christoph Prevezanos. *Technisches Schreiben: Für Informatiker, Akademiker, Techniker und den Berufsalltag*. Carl Hanser Verlag, 2013.
- [6] Joachim Schlosser. *Wissenschaftliche Arbeiten schreiben mit LaTeX: Leitfaden für Einsteiger*. mitp Professional, 2016.

### Online-Quellen

- [5] John Gossman. *Introduction to Model/View/ViewModel pattern for building WPF apps*. Microsoft. Okt. 2005. URL: <https://blogs.msdn.microsoft.com/johngossman/2005/10/08/introduction-to-modelviewviewmodel-pattern-for-building-wpf-apps/> (besucht am 03.08.2018).