

Projektbegründungen

“MDKS”

im Rahmen der Veranstaltung
Entwicklung Interaktiver Systeme

Veranstaltung betreut von
Prof. Dr. Kristian Fischer | Prof. Dr. Gerhard Hartmann

im
WS 2015/2016

verfasst von
Kevin Apitz | Alexander Miske

Gruppe betreut von
Sheree Saßmannshausen | Ngoc-Anh Dang

Inhaltsverzeichnis

1	Einleitung
2	Ideenfindung
3	Betrachtung des Nutzungskontextes
4	Zielhierarchie
6	Alleinstellungsmerkmale
7	Kommunikationsmodell
8	Risiken
9	Proof of Concepts
10	MCI-Vorgehen
10.1	Abwägung der MCI-Vorgehensmodelle
10.2	Abwägung der Methoden
10.2.1	Verstehen und Beschreiben des Nutzungskontextes
10.2.2	Spezifizieren der Nutzungsanforderungen
10.2.3	Entwerfen der Gestaltungslösungen
	papierbasierte vs. computerbasierte Prototypen
	Techniken des Prototyping
10.2.4	Testen und Bewerten der Gestaltung
11	Benutzermodellierung
12	Aufgabenmodellierung
13	Use Cases
14	Anforderungen
15	Entwerfen der Prototypen
16	Evaluation
17	Poster
18	Prozessassessment
19	Fazit
20	Projektplan
	Quellenverzeichnis

1 Einleitung

Im Rahmen des Projektes “Entwicklung interaktiver Systeme” soll ein verteiltes System konzipiert und entwickelt werden. In diesem Dokument sollen alle Entscheidungen bezüglich des Projektes festgehalten und begründet werden, sodass der Verlauf der Entwicklung des Projektes erkennbar und nachvollziehbar ist. Die Begründungen stützen sich in erster Linie auf verschiedene Artefakte, in denen konkrete Thematiken der Projektentwicklung aus Sicht der “MCI” und der “WBA” im Bezug auf das zu entwickelnde Projekt betrachtet und analysiert werden.

2 Ideenfindung

Zum Start des Projektes müssen zunächst Ideen für Projekte gesammelt werden. Brainstorming ist hierbei eine geeignete Methode. Es wurden verschiedene Ansätze zur Ideenfindung eingesetzt. Einerseits wurden Projektideen aus dem eigenen Alltag und Umfeld miteinbezogen. Zum Anderen wurden speziell Domänen gesucht, die im Markt weniger vertreten sind oder in der Qualität der Produkte Mängel aufweisen. Eine weitere Methode zur Ideenfindung ist die Betrachtung von Problemräumen.

Im Anschluss wurden die Ideen diskutiert, kritisch betrachtet und priorisiert. Als Ergebnis wurden drei Ideen festgehalten, aus denen dann Exposés entwickelt wurden.

- siehe [Exposé - Drohnenflug-Netzwerk](#)
- siehe [Exposé - Go Get It](#)
- siehe [Exposé - Medikations-System](#)

In den Beratungsterminen mit den Betreuern wurden die Exposés kritisch hinterleuchtet und einige Mängel herausgestellt. So zeigte sich, dass die Grundidee von “Go Get It” bereits im Markt vertreten ist. Auch die Domäne erwies sich als ungeeignet für das

EIS-Projekt, da die Benutzergruppe “Käufer und Verkäufer” schwer zu identifizieren sind. Bei der Idee “Drohnenflug-Netzwerk” stellte sich heraus, dass ein verteiltes System nicht notwendig ist für die Umsetzung der Idee. Das “Medikations-System” kurz “MDKS” hob sich als einzige Idee mit Potential hervor. Daher soll das Projekt nun mit dieser Idee begonnen werden.

Der nächste Schritt sollte eine Domänenrecherche sein, damit der Nutzungskontext beschrieben werden kann.

3 Betrachtung des Nutzungskontextes

Um das Projekt zu konzipieren muss eine erste Domänenrecherche durchgeführt werden. Dabei wurden alle Stakeholder, Vorgänge, Metaphern sowie Paradigmen recherchiert. Die Ergebnisse sollen als Grundlage für weitere projektspezifische Entscheidungen dienen.

Das zu entwickelnde System soll in Krankenhäusern eingesetzt werden und die dort angewendete Medikation unterstützen. Dem Medikationsprozess liegt eine hohe Komplexität zugrunde, weshalb die Arzneimitteltherapie viele Risiken birgt und als “der fehleranfälligste Teil der medizinischen Versorgung”(Ärztliche Direktor der Frankfurter Uniklinik, Jürgen Schölmerich[1]) gilt. Dies resultiert aus den unterschiedlichen Personen, die in diesem Prozess involviert sind, der Vielzahl an Vorgängen, die für die Medikation notwendig sind, dem Daten- und Informationsaustausch unter den beteiligten Personen, sowie der Vielfalt an Medikamenten, die im Krankenhaus eingesetzt werden.

Die Etablierung eines Systems im Krankenhaus birgt außerdem noch rechtliche Risiken. Daher müssen weitere Untersuchungen im Bereich Datenschutz und Datensicherheit unternommen werden.

Viele Begrifflichkeiten aus der Medizin, die in dem MDKS verwendet werden, wurden in der Domänenrecherche zusammengestellt und erläutert. Diese Liste wird im Verlauf des Projekts stetig bearbeitet.

□ siehe [Domänenrecherche](#)

Nach der ersten Betrachtung der Domäne ist es nun wichtig, die Ziele des MDKS klar zu definieren. Diese sollen in einer Zielhierarchie aufgeführt werden

4 Zielhierarchie

Die Ziele für das Projekt wurden aus dem anfänglichen [Exposé](#) und der [Domänenrecherche](#) abgeleitet. Die in der Domänenrecherche beschriebenen Probleme und Risiken in dem Medikationsprozess wurden dabei berücksichtigt. Die Zielhierarchie soll vor allem als Leitlinie für das Projekt dienen. Abschließend können die Ziele herangezogen werden, um ein Fazit für das Projekt zu verfassen.

□ siehe [Zielhierarchie](#)

Da die Verwaltung von Medikamenten und Verordnungen ein großes Thema in unserer Gesellschaft ist, gibt es viele Unternehmen, die sich diesem Problem bereits gewidmet haben. Um einen Überblick über die Konkurrenzprodukte zu bekommen und die Qualität dieser abzuschätzen ist es notwendig eine Marktrecherche durchzuführen.

5 Marktrecherche

Für die Marktrecherche wurden drei Produkte aus der Domäne ausgewählt, die den Medikationsprozess vereinfachen und unterstützen. Ziel ist es positive wie negative Aspekte der Konkurrenzprodukte gegenüber dem MDKS heraus zu kristallisieren. Anhand von Broschüren und Webseiten der Unternehmen konnten ausreichende Recherchen durchgeführt werden.

□ siehe [Marktrecherche](#)

6 Alleinstellungsmerkmale

Die [Marktrecherche](#) soll nun verwendet werden, um Alleinstellungsmerkmale des MDKS zu definieren. Dafür wurden die negativen Aspekte der Konkurrenz zusammengefasst und mit den [Zielen](#) des MDKS verglichen.

In keinem der Produkte wurde der Patient aktiv in den Medikationsprozess einbezogen. Die Entwicklung einer Patientenschnittstelle für den Medikationsprozess stellt daher ein Alleinstellungsmerkmal dar.

□ siehe [Alleinstellungsmerkmale](#)

Nach der Betrachtung des Medikationsprozesses und der beteiligten Personen kann nun ein Kommunikationsmodell dazu dienen den deskriptiven Zustand der Kommunikation in Krankenhäusern abzubilden. Ziel ist es, anhand der aufgestellten [Ziele](#) ein präskriptives Kommunikationsmodell zu entwickeln. Dabei sollen vor allem die [Alleinstellungsmerkmale](#) ersichtlich werden.

7 Kommunikationsmodell

Die Vorgänge des Medikationsprozesses wurden im Hinblick auf die Kommunikation der Stakeholder untersucht. Das Kommunikationsdiagramm baut auf den in der [Domänenrecherche](#) beschriebenen Medikationsprozess auf.

□ siehe [Kommunikationsmodell](#)

Außer der verbalen Kommunikation dienen Datensammlungen, wie die Patientenakte dem Personal als Kommunikationsmedium. Daher wurden Daten als weitere Instanz in dem Diagramm hinzugefügt, um die Kommunikation konkreter zu beschreiben. Des Weiteren wurden Datenzugriffe des Arztes auf Medikamentendaten berücksichtigt.

Die Instanz "Einrichtungen" wurde herangezogen, um Schritte, die von dem Kommunikationsablauf abhängig sind, beschreiben zu können. So muss zum Beispiel das Pflegepersonal den Bestand in der Station prüfen bevor eine Kommunikation zum Apotheker stattfindet.

Aus dem deskriptiven Kommunikationsmodell wurde das präskriptive Modell entwickelt. Da in dem zu entwickelnden System die benötigten Patienten-, Verordnungs-, und Medikamentendaten elektronisch vorliegen sollen, ist eine Kommunikation über die Patientenakte nicht mehr nötig. Der Kommunikationsprozess findet über das System statt. Dennoch können aus rechtlichen Gründen (siehe [Domänenrecherche](#)) nicht alle Kommunikationsprozesse über das System stattfinden. Daher müssen Apotheker und Ärzte weiterhin persönliche Gespräche mit den Patienten über Verordnungen von Medikamenten führen.

Die erarbeiteten [Alleinstellungsmerkmale](#) sind in der Kommunikation zwischen Patienten und System ersichtlich. Der Patient kann Informationen zu seiner Medikation erhalten und Kontrolle über Selbstmedikationen über das System erhalten.

8 Risiken

Die Risiken sollten bereits in der frühen Entwicklungsphase des Projekts identifiziert und priorisiert werden, damit mögliche Probleme, die in der späteren Entwicklungsphase auftreten können vermieden werden können.

Nach der Betrachtung des Nutzungskontextes konnten bereits einige Risiken bei der Behandlung des Datenschutzes bei der Entwicklung ermittelt werden. Weitere Risiken konnten aus den [Architekturbegründungen](#) des MDKS abgeleitet werden. Viele der Techniken, die in dem System zum Einsatz kommen sollen sind dem Team bisher unbekannt, wodurch ein Risiko für die Realisierung der Architektur besteht.

□ siehe [Risiken](#)

Die Risiken wurden anschließend quantitativ bewertet, sodass diese priorisiert werden konnten.

□ siehe [Risiken-Ranking](#)

9 Proof of Concepts

□ siehe [POC 1 Publish-Subscribe mit RabbitMQ](#)

□ siehe [POC 2 Notification-Service auf einem Android-Client](#)

□ siehe [POC 3 Synchroner Datenaustausch zwischen Android und NodeJS mit Speicherung in MySQL](#)

10 MCI-Vorgehen

Das Vorgehen aus der MCI-Sicht ist für die Konzeption des Projekts grundlegend. Deswegen muss die Wahl des Vorgehensmodells ausreichend abgewägt werden. Die Argumente für die Wahl eines Vorgehensmodells müssen immer in Bezug zu dem vorhandenen Problemraum stehen. Aus diesem Grund müssen die Ergebnisse aus der Untersuchung der [Domäne](#) an dieser Stelle herangezogen werden.

10.1 Abwägung der MCI-Vorgehensmodelle

Aus dem Nutzungskontext ist ersichtlich, dass viele Benutzertypen im Medikationsprozess involviert sind. Diese haben verschiedene komplexe Aufgaben und Perspektiven, die maßgeblich entscheidend für die Entwicklung des Systems sind. So hat der Patient ein völlig anderes Interesse an dem System als das Pflegepersonal. Ein besonderer Fokus im Vorgehensmodell sollte daher der Benutzer mit seinen Aufgaben sein. Dennoch muss das System ein gutes Werkzeug sein, da der Medikationsprozess wichtige und risikoreiche Vorgänge beinhaltet.

Das Usage Centered Design, dass von Lockwood und Constantine 1996 entwickelt wurde, fokussiert sich auf den Verwendungszweck des zu entwickelnden interaktiven Systems. Dieses System soll ein gutes Werkzeug für die Benutzer darstellen, wodurch die Betrachtung des eigentlichen Benutzers des Systems in den Hintergrund rückt.

Das User-Centered Design fokussiert sich dagegen auf den Benutzer und seine Bedürfnisse. Demnach muss der Benutzer des zu entwickelnden Systems stetig in den Entwicklungsprozess miteinbezogen werden.

Aus der Argumentation heraus sollte also von einem benutzungszentrierten Ansatz abgesehen werden und ein benutzerzentriertes Vorgehensmodell gewählt werden. Grund dafür ist vorwiegend die inhomogene Zielgruppe des Systems.

Einige Modelle geben dabei konkrete Methoden vor. Im Usability-Engineering nach Rosson und Carroll wird hauptsächlich mit Szenarien gearbeitet. Diese werden als Methode für jegliche Aktivitäten bei der Entwicklung des Systems vorgesehen.

Das Vorgehensmodell "Usability engineering lifecycle" nach Deborah Mayhew ist iterativ und skalierbar und besteht aus drei Prozessbestandteilen. Zunächst müssen die Anforderungen der Stakeholder analysiert werden, die sich aus den User Profiles und der Aufgabenanalyse ableiten lassen. Der nächste Schritt ist es aus den Anforderungen Prototypen zu entwickeln und Style-Guides festzulegen. Dieser Schritt muss solange evaluiert und getestet werden, bis die Prototypen valide sind und alle Probleme im Hinblick auf die präskriptive Aufgabenmodellierung beseitigt sind. Die Prototypen müssen abschließend dahingehend überprüft werden, ob alle Anforderungen adressiert werden. Der letzte Bestandteil des Vorgehensmodells ist es, das System zu installieren. Auch dieser Schritt sieht Iterationen vor. Das System muss mit Zielbenutzern evaluiert und gegebenenfalls weiterentwickelt werden.

Auffallend bei diesem Vorgehensmodell ist besonders die Iterationen in allen qualitätsrelevanten Aktivitäten. Dies kann im Hinblick auf den Problemraum sehr zeitaufwendig werden und ein echtes Risiko für die Realisierung des Projekts darstellen.

Das "Discount Usability Engineering" Vorgehensmodell nach Nielsen beschreibt, wie man mit Probanden durch die Methode des think alouds sein System mit kostengünstigem Aufwand gebrauchstauglicher gestalten kann. Dabei werden Beobachtung angestellt, während die Probanden ihre Aufgaben mit einem Prototypen des Systems lösen. Die Evaluierung wird durch die 10 usability-heuristiken nach Nielsen durchgeführt. Dadurch kann ein für die Benutzer und ihren Fähigkeiten angepasstes System konzipiert werden, wodurch die Gebrauchstauglichkeit erhöht wird und gleichzeitig das Risiko vermindert wird, dass die Anforderungen der Stakeholder nicht erfüllt werden. Vorteilhaft ist vor allem an diesem Modell, dass die

Gebrauchstauglichkeit durch wenig aufwand gesteigert werden kann. Dazu werden konkrete Techniken vorgegeben, die im Entwicklungsprozess verwendet werden müssen.[2]

Aus dem Problemraum wird ersichtlich, dass die Aufgaben sehr komplex werden können und verschiedene Anwendungsfälle betrachtet werden müssen. Dabei ist es wichtig Unteraufgaben und Kommunikationsnetze aufzuzeigen. Durch Szenarien kann dies beschrieben werden. Dennoch ist es sinnvoll weitere Methoden zu den Szenarien zu verwenden, um die Aufgaben zu beschreiben. Das Vorgehensmodell sollte daher keine konkreten Methoden vorschreiben und modifizierbar sein. Dadurch kann nach Bedarf weiteres Wissen durch andere Methoden erarbeitet werden. Demnach sollten auch Iterationen im Vorgehensmodell einplanbar sein. Die Wahl fällt aufgrund der beschriebenen Umstände auf das ISO 9142 Teil 210[3] Vorgehensmodell. Dieses wird mit einigen Methoden modifiziert.

□ siehe [MCI-Rahmen](#)

10.2 Abwägung der Methoden

Die Methoden dienen uns für das Erarbeiten von Wissen für die Konzeption des Systems. Die ISO Norm[3] nach der das Vorgehensmodell erstellt werden soll, legt keine Methoden fest. Daher müssen mögliche Methoden für die einzelnen Aktivitäten aus dem [MCI-Rahmen](#) gewählt werden. Die ersten Recherchen aus dem Nutzungskontext (siehe [Domänenrecherche](#)) bieten uns auch an dieser Stelle Argumente zur Wahl von Methoden.

10.2.1 Verstehen und Beschreiben des Nutzungskontextes

Nach der Norm bestimmen die Benutzermerkmale, Arbeitsaufgaben und die organisatorische, technische und physische Umgebung den Kontext. Daher müssen

zuerst die Benutzer identifiziert und analysiert werden. Eine geeignete Methode dafür ist die Stakeholderanalyse. Der Benutzer werden in primäre, sekundäre und tertiäre Benutzergruppen aufgeteilt. Vorteilhaft ist hierbei, dass sofort ersichtlich ist welche Benutzer im Fokus stehen und das System häufig nutzen werden. Trotzdem können Nutzer, die zwar in Verbindung mit dem System stehen, aber es nicht häufig oder garnicht benutzen mit in den Entwicklungsprozess einbezogen werden.

Anschließend können User Profiles für diese Benutzer angelegt werden, die die Benutzermerkmale aufzeigen. Dies ist für unseren Problemraum sehr wichtig, da hier die Fähigkeiten und Einschränkungen der Benutzer ersichtlich werden, die für die Gestaltung eines barrierefreien Systems von Bedeutung sind.

Aus den User Profiles können Personae entwickelt werden, die die Benutzerperspektive zum System verdeutlichen können.

Die Aufgaben der Benutzer sind verschieden. Diese unterscheiden sich in Komplexität und Häufigkeit der Durchführung. Des Weiteren können mehrere Benutzer bei den Aufgaben involviert sein. Es bietet sich an Problemszenarien als Methode zur Beschreibung für die komplexen Aufgaben zu verwenden. Die Beschreibungen sind deskriptiv. Aus zeitlichen Gründen können nicht alle Aufgabenbereiche durch Szenarien abgedeckt werden, da das Schreiben von Szenarien zeitaufwendig ist.

Anschließend können Fehleranfällige Aufgabenbereiche spezifischer durch Use Cases betrachtet werden. Use Cases beschreiben Aufgaben genauer als Hierarchical Task Analysen. Dabei sollen die Use Cases in narrativer Form erstellt werden, wodurch verschiedene Aspekte besser beschrieben werden können. Die Use Cases sollten vor allem auf folgende Fragestellungen eingehen:

- Wie häufig wird die Aufgabe durchgeführt?
- Welche Risikofaktoren gibt es bei der Durchführung?

- Welche Fertigkeiten und Fähigkeiten werden benötigt?
- Ist der Vorgang kontinuierliche oder wird dieser temporär immer wieder aufgegriffen?
- Welche Faktoren sind kritisch bei dem Prozess (Zeit, Informationen)?
- Welche Benutzer sind beteiligt?

10.2.2 Spezifizieren der Nutzungsanforderungen

Es ist wichtig, dass das deskriptive Modell der Aufgaben ausführlich betrachtet wird, um funktionale sowie nicht-funktionale Anforderungen zu ermitteln. Die Use Cases als präskriptives Modell sollen die Anforderungen beinhalten. Eine mögliche Iteration wäre hierbei, den Nutzungskontext nochmals zu betrachten, um mögliche weitere Anwendungsfälle zu finden. Die Anforderungen werden demnach aus den Problemszenarien ermittelt. Allerdings sollte die Domänenrecherche auch für die Ermittlung der Anforderungen herangezogen werden, da eventuell nicht alle Aufgaben modelliert werden können oder wichtige Aspekte nicht in den Szenarien betrachtet wurden.

10.2.3 Entwerfen der Gestaltungslösungen

Es gibt zwei verschiedene Methoden zur Erstellung von Prototypen[4], die im Folgenden diskutiert werden sollen.

papierbasierte vs. computerbasierte Prototypen

Der Vorteil von papierbasierten Prototypen sind, dass sie in Handskizzen oder -zeichnungen erstellt werden und deshalb weniger zeitaufwendig sind. Zudem sind sie leicht änderbar und auch zeitliche Abfolgen können über Storyboards dargestellt werden. Der Nachteil ist, dass keine Interaktionsmöglichkeit bestehen, wie bei den computerbasierten Prototypen. Im Hinblick auf den zeitlichen Rahmen des Projekts eignen sich daher papierbasierte Prototypen mehr. Die Erstellung von computerbasierten Prototypen könnte ein echtes zeitliches Risiko für das Projekt

darstellen, da alle Prototypen immer wieder geändert und validiert werden müssen. Des Weiteren können durch papierbasierte Prototypen besonders wichtige Aspekte im Interface hervorgehoben werden, indem andere Aspekte nicht in der Gestaltung miteinbezogen werden.

Techniken des Prototyping

Requirements Animation zielt darauf ab, funktionale Anforderungen im frühen Projektphasen im Prototypen zu gestalten und diese mit den Stakeholdern zu validieren. Für unser Projekt eignet sich diese Variante sehr gut, da ein komplexes System zu erstellen ist. Daher sollten Fehler der Prototypen schon im frühen Stadium der Entwicklung erkannt und beseitigt werden. Um die Ergebnisse im Rahmen des Projektes im vollen Funktionsumfang präsentieren zu können, müssen die Prototypen vertikal entwickelt werden.

10.2.4 Testen und Bewerten der Gestaltung

Die erste Evaluierung der Prototypen soll mithilfe von Gestaltungsregeln durchgeführt werden. Es soll überprüft werden, ob die Prototypen Mängel aufweisen, die durch für das Projekt erstellten Gestaltungsregeln identifiziert werden sollen. Anschließend müssen die Mängel beseitigt werden und die Evaluierung muss erneut durchgeführt werden. Dieser Prozess soll wiederholt werden, bis keine Mängel mehr vorhanden sind und die Prototypen valide sind.

Der nächste Schritt ist es, die Prototypen mit einem Benutzer des Systems zu evaluieren. Ziel ist es die Evaluation formativ durchzuführen, um weitere Anforderungen zu erheben und Fehler zu beseitigen. Dabei soll ein qualitatives Verfahren verwendet werden, damit der Benutzer seine Meinung beim Testen frei äußern kann.

11 Benutzermodellierung

Der nächste Schritt ist zunächst die Stakeholder konkret zu benennen und sie in dem Nutzungskontext einzuordnen. Dazu werden die Ergebnisse aus der [Domänenrecherche](#) verwendet.

□ siehe [Stakeholder](#)

Um aus den ermittelten Stakeholdern Personae zu generieren, müssen User Profiles angelegt werden. An dieser Stelle kann der Aufwand immens werden, da gerade zu unzählige User Profile erstellt werden können. Daher muss hier eine Entscheidung getroffen werden, auf welche Benutzer wir uns besonders in diesem Projekt fokussieren wollen. Für die Benutzung des Systems sind bisher Patienten, Ärzte, Pflegepersonal und Apotheker vorgesehen. Die Hauptaufgaben im Medikationsprozess liegen bei den Ärzten und dem Pflegepersonal. Der Patient soll, wie in den Alleinstellungsmerkmalen beschrieben, mit in den Prozess einbezogen werden. Aufgrund der zeitlichen Einschränkung des Projekts müssen die Stakeholder priorisiert werden. Demnach soll bei dem weiteren Vorgehen der Benutzermodellierung der Fokus auf folgende Benutzertypen fallen:

- Patienten
- Ärzte
- Pflegepersonal

Die User Profiles wurden zunächst grob mit einigen Benutzermerkmalen und Ausprägungen angelegt. Nach weiterer Betrachtung des Nutzungskontextes wurden die User Profiles verfeinert. Es wurden nun weitere Ranges für das Alter der Benutzer angelegt, da die Merkmale sich dahingehend teilweise unterscheiden. Diese Änderung lieferte uns auch weitere Ausprägungen der Merkmale sowie Sichtweisen der Benutzer. Oft ist es der Fall, dass ältere Menschen Einschränkungen wie eine Sehschwäche

aufweisen und eine andere Einstellung zu elektronischen Systemen und Technologien haben als jüngere Benutzer.

□ siehe [User Profiles](#)

Um diese Unterschiede und Auswirkungen der Merkmale konkret in Szenarien zu beschreiben wurden zunächst Personae aus bestimmten Altersgruppen erstellt. Dabei wurde ein besonderer Fokus auf kritische Benutzermerkmale, wie die Einschränkung der Fähigkeiten oder Technologieängste gelegt. Um die Unterschiede der Patienten aufzuzeigen, wurden zunächst zwei Personae geschrieben, die als Patienten in den Szenarien verwendet werden sollen.

□ siehe [Personae](#)

12 Aufgabenmodellierung

Um weitere Aufgaben der Pfleger zu sammeln, wurde ein Interview mit einem Pflegepersonal geführt. In dem Gespräch wurden neben weiteren Aufgaben auch weitere Risiken diskutiert. Die Aufgaben und Risiken wurden in den Problemszenarien aufgegriffen.

□ siehe [Interview mit Krankenschwester](#)

Grundlage für die deskriptive Aufgabenmodellierung sind die [Personae](#). Um konkrete Probleme zu skizzieren, wurde die [Domänenrecherche](#) und das Interview als Quelle hinzugezogen. Die dort beschriebenen Risiken in der Medikation konnten für die Erstellung der Problemszenarien verwendet werden. Die Problemszenarien sollen vor allem die Probleme verdeutlichen, die ohne ein Medikationssystem entstehen.

□ siehe [Problemszenarien](#)

Um die Probleme aus den Szenarien zu verdeutlichen, wurde eine Claimsanalyse zu den Problemszenarien durchgeführt. Dabei wurden Vor- und Nachteile zu dem Ist-Zustand der Aufgabenmodellierung herausgestellt. Diese sollen der Ermittlung von Anforderungen dienen.

□ siehe [Claimsanalyse](#)

13 Use Cases

Die Use Cases sollen die präskriptive Aufgabenmodellierung darstellen. Diese wurden auf Grundlage der deskriptiven Aufgabenmodellierung entwickelt. Probleme, die während der Medikation entstehen sollen hier bereits gelöst werden. Dadurch können weitere Anforderungen erhoben werden.

Aufgrund der [Evaluationsergebnisse](#) mussten Änderungen in einigen Use Cases vorgenommen werden.

□ siehe [Use Cases](#)

14 Anforderungen

Die Anforderungen wurden aus den Erfordernissen der [Stakeholdern](#), den [Claimsanalysen](#) und den [Use Cases](#) entnommen. Des weiteren wurde die Domänenrecherche verwendet, um weitere Anforderungen zu erheben. Bei der Auflistung der Anforderungen wurde zwischen funktionalen und non-funktionalen Anforderungen unterschieden.

Die Liste der Anforderungen wurde im Laufe des Projekts stetig erweitert, da bei erneuter Betrachtung des Nutzungskontextes neue Anforderungen gefunden wurden.

Die Formulierungen der Anforderungen wurden anhand von Schablonen[5] von Chris Rupp angepasst.

□ siehe [Anforderungen](#)

15 Entwerfen der Prototypen

Die Gestaltungsgrundsätze nach ISO 9241-110[6]

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Konformität mit Benutzererwartungen
- Lernförderlichkeit
- Steuerbarkeit
- Fehlertoleranz
- Individualisierbarkeit

wurden bei der Gestaltung des interaktiven Systems berücksichtigt, um eine angemessene User Experience zu erreichen . Eine daraus resultierende Entscheidung war es, den Android Style-Guide in den Gestaltungsprozess miteinzubeziehen, um die Lernförderlichkeit und Benutzerkonformität der mit Androidgeräten vertrauten Nutzer zu steigern. Konkretisiert werden die Gestaltungslösungen schließlich mittels papierbasierter Prototypen, welche im Verlauf des 4. und 5. Meilensteins mehrfach überarbeitet und an die [Anforderungen](#) angepasst wurden.

Aufgrund der großen Tabellen, die für den Medikations- und Verabreichungsplan erstellt werden müssen, wird vorausgesetzt, dass das MDKS auf dem Android Tablet ausschließlich im Landscape-Modus genutzt wird.

□ siehe [PrototypenUI](#)

16 Evaluation

Die Evaluation soll Rückmeldungen über die Stärken und Schwächen der Gestaltungslösungen aus Nutzerperspektive liefern und dabei helfen, den Erfüllungsgrad der Anforderungen zu bestimmen.

Für die Evaluationsmethode “think aloud” werden die Rollen Proband (Medieninformatiker eines höheren Semesters), Moderator (führt die Evaluation durch und stellt sicher, dass die Aussagen des Probanden verwertbar sind) und Versuchsleiter (überwacht und protokolliert) benötigt.

Getestet wird das User Interface für Patient und Pfleger. Aufgezeichnet wird der Evaluationsvorgang via Audioaufnahme.

Die Evaluation teilt sich in 2 Phasen auf. In Phase 1, der Aufwärmphase, wird dem Probanden die Möglichkeit geboten, anhand der Android Applikation “Google Mail” das “think aloud” einzuüben.

In Phase 2, der Evaluationsphase, wird dem Probanden Zweck und Einsatzbereich des Systems erläutert. Da das User Interface des Pflegers ein besonderes Vorwissen des Testers über den organisatorischen Ablauf des Medikationsprozesses voraussetzt, wird dieser ihm ebenfalls verständlich erklärt. Daraufgehend erhält der Proband papierbasierte Prototypen des Systems und soll sich in diesem mit möglichst wenig Anweisungen des Moderators, frei bewegen und seine Gedanken laut aussprechen.

Im Anschluss werden sowohl das schriftliche Protokoll als auch die Audioaufnahme aufbereitet und ausgewertet.

□ siehe [EvaluationsergebnisseUI](#)

→ [Audioaufnahme](#)

17 Poster

- siehe [Posterkonzept](#)
- siehe [Poster](#)

18 Prozessassessment

- siehe [Prozessassessment](#)

19 Fazit

- siehe [Fazit](#)

20 Projektplan

- siehe [Projektplan](#)

Quellenverzeichnis

- [1] <http://www.n-tv.de/wissen/Medikamente-sind-grosse-Fehlerquelle-article14803246.html> (Sichtung: 17.10.2015 18:15)
- [2] <http://www.nngroup.com/articles/discount-usability-20-years/>
(Sichtung: 23.10.2015)
- [3] DIN EN ISO 9241. Ergonomie der Mensch-System-Interaktion: Teil 210: Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme. DIN Deutsches Institut für Normung, 2010.
- [4] MCI-Draft von Gerhard Hartmann, S 500 20.2.2 Prototypen in der Mensch-Computer Interaktion
- [5] <http://www4.in.tum.de/lehre/vorlesungen/re/ws12/uebung/REuM5Auf1Kap7.pdf>
(Sichtung: 21.12.2015)
- [6] DIN EN ISO 9241. Ergonomie der Mensch-System-Interaktion: Teil 110: Grundsätze der Dialoggestaltung. DIN Deutsches Institut für Normung, 2010