

Entwicklungsprojekt interaktive Systeme

Wintersemester 2018/2019

Modelle und Modellierungsbegründungen Meet & Remind

Dozenten

Prof. Dr. Gerhard Hartmann

Prof. Dr. Kristian Fischer

Betreuer

Daniela Reschke

Markus Alterauge

Team

Johanna Mayer

Julian Schoemaker

Inhaltsverzeichnis

Einleitung	4
Vorgehensmodell	4
DIN EN ISO 9241-210	4
Usage Centered Design	5
Beschreibung	5
Begründung	6
Einsatz	6
Collaborative Requirements Dialog	7
Erfordernisse	7
Anforderungen	7
Funktionale Anforderungen	7
Organisationale Anforderungen	8
Qualitative Anforderungen	8
Technische Anforderungen	8
Fazit aus Anforderungen	9
Domain Model	9
Domänenmodell	9
Klassendiagramm	10
Stakeholderanalyse	10
Tabelle: Stakeholderanalyse	10
Role Model	11
User Roles	12
User Roles Vorgehen	12
User Role Tabelle	12
User Role Maps	15
Fazit aus User Roles	15
Operational Model (optional)	16
Operational Profiles	16
Fazit aus Operation Profiles	16
Task Model	16
Essential Use Cases	16
Use Case Maps	16
Fazit aus Use Cases	16
Content Model	16
Content Model	16
Navigation Map	16
Fazit aus Content Model	16

Implementational Model	17
Visual Design	17
Usability Inspection	17
Evaluierung	17
Systemkomponenten	17
Komponenten	17
Dienste im Web	17
Architektur	17
Kommunikation zwischen den Komponenten	17
Arten von Endgeräten	17
Datenstruktur	17
Persistente Datenhaltung	17
Datenformat	17
Quellen	19

Einleitung

In diesem Dokument geht es um die Modelle und die Modellierungsbegründungen zum im Konzept beschriebenen System Meet & Remind. Es befasst sich mit dem gewählten Vorgehensmodell der Software-Entwicklung und beantwortet die Fragen zur Erstellung des User Interfaces und der Umsetzung der weiteren Systemkomponenten.

Vorgehensmodell

DIN EN ISO 9241-210

Diese Norm befasst sich mit dem menschenzentrierten Gestaltungsprozess. Dies ist ein Vorgehen, das bei der Entwicklung von interaktiven Systemen eingesetzt werden kann. Dabei soll sich unter anderem auf die Verwendung des Systems und auf die Gebrauchstauglichkeit konzentriert werden.

Dieses Vorgehen passt zu dem zu entwickelnden System, da der Benutzer, und auch die Stakeholder im Allgemeinen, eine (im Alltag) unterstützende Anwendung erhalten sollen und ihre Aufgaben (und Ziele) im Bezug zum System effektiv, effizient und zufriedenstellend erledigen können sollen.

In Abbildung 1 wird der Prozess der menschenzentrierten Gestaltungsaktivitäten gezeigt. Zuerst muss der Nutzungskontext verstanden und beschrieben werden. Dies geschah für "Meet & Remind" bereits im Konzept. Hier wurden die Domäne recherchiert, Probleme beschrieben, Ursachen gesucht und Stakeholder aufgelistet. Somit wurde der Nutzungskontext genauer untersucht, jedoch ist dieser Teil ein iterativer Prozess und wird ggf. nochmals aufgegriffen und überarbeitet.

Wenn der Nutzungskontext verstanden wurde folgt die Spezifizierung der Anforderungen. Hierbei wird in funktionale, organisationale und technische Anforderungen unterschieden. Diese sind Voraussetzung für den weiteren Prozess und den ersten Gestaltungslösungen.

Bei der Entwicklung von Gestaltungslösungen wird ein weiteres Vorgehensmodell, das "Usage Centered Design" angewendet, um den Verwendungszweck des Systems besser zu verstehen und damit die Benutzung des Systems zu vereinfachen.

Am Ende des Gestaltungsprozess steht die Evaluierung. Hier muss die Perspektive der Benutzer eingenommen werden oder ggf. Tests mit ihnen durchgeführt werden, um zu entscheiden, welche Teilaspekte des Prozess iteriert werden sollten, um am Ende ein gebrauchstaugliches System zu besitzen.

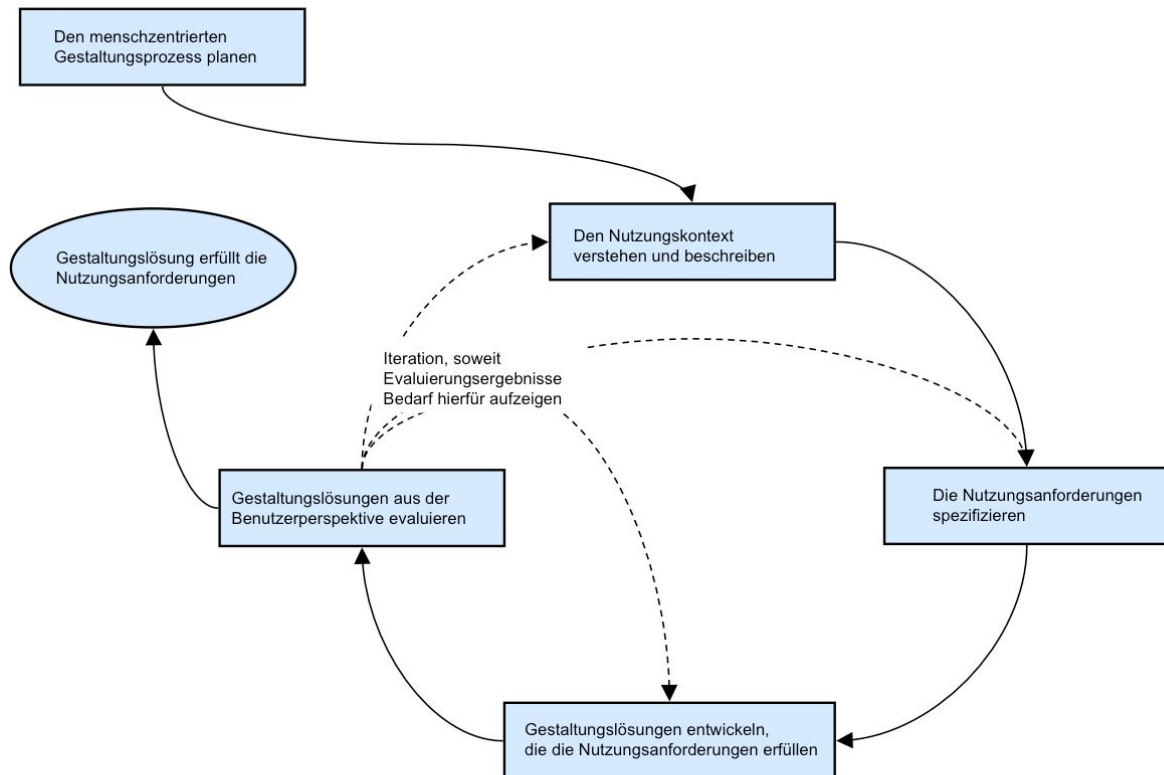


Abb. 1: aus DIN EN ISO 9241-210: Wechselseitige Abhängigkeit menschzentrierter Gestaltungsaktivitäten

Usage Centered Design

Beschreibung

Das Vorgehensmodell des Usage Centered Design basiert auf der modellgetriebenen Entwicklung in der Software-Entwicklung. Aus dem Namen lässt sich erkennen, dass hierbei vor allem die Einsatzmöglichkeit und Benutzung des Systems im Vordergrund steht. Die drei Modelle Role Model, Task Model, Operational Model und Content Model bilden die Grundlage für die Erstellung des Visual Design im Implementational Model. Außerdem liefern sie Erkenntnisse für die fachlichen Datenmodelle und beeinflussen damit nachhaltig die Systemarchitektur.

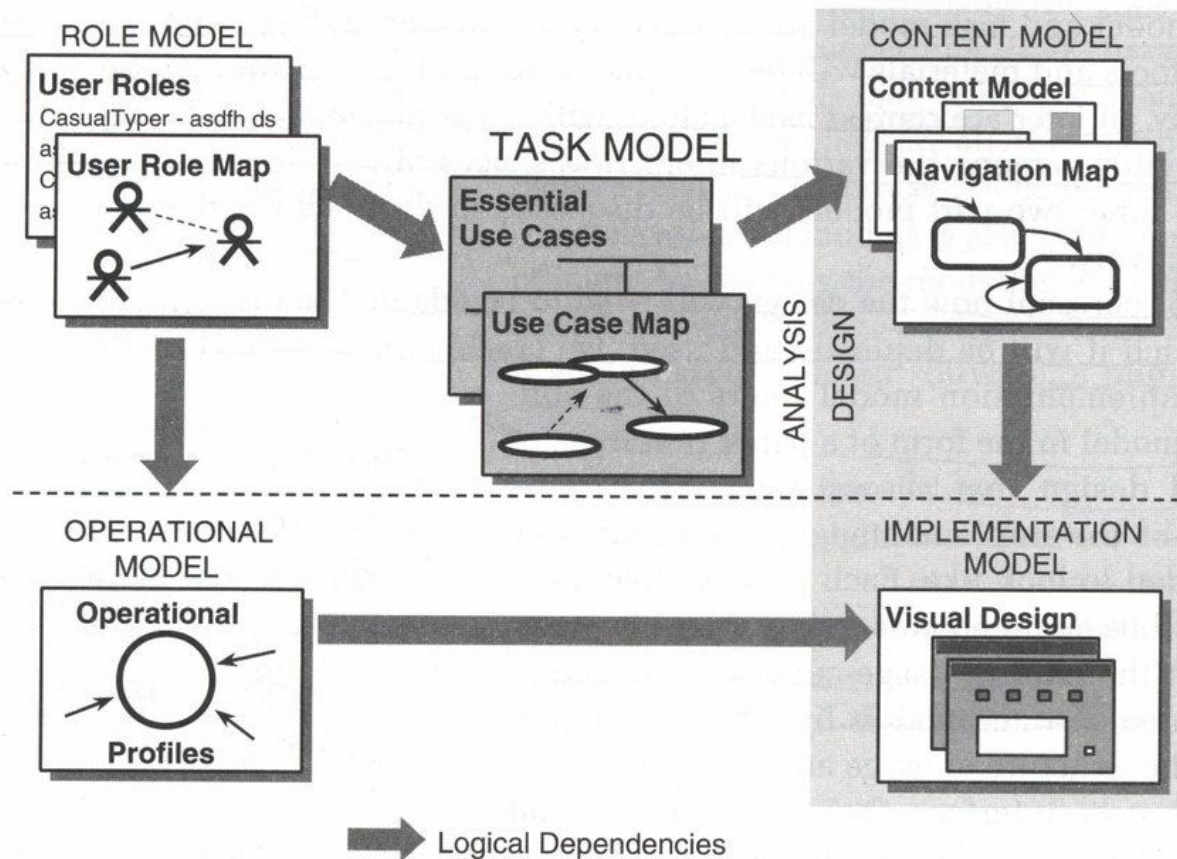


Abb. 2: aus Software for use: Essential models

Begründung

Im Konzept wurde bereits kurz beschrieben, warum dieses Vorgehensmodell positiv zur Entwicklung von Meet & Remind beitragen würde. Zudem lässt sich aus dem vorhergehenden Abschnitt erkennen, dass das Usage Centered Design für den Entwicklungsprozess förderlich ist.

Hier wurde auch das Scenario Based Usability Engineering als passend genannt, welches jedoch bei der weiteren Betrachtung der Modellierung als weniger geeignet eingeschätzt wurde. Die erarbeiteten Problemszenarien für dieses Vorgehensmodell wichen nicht weit genug voneinander ab, als dass man aus ihnen Schlüsse für das visuelle Design ziehen konnte.

Bei der ersten Stakeholderanalyse und dem Aufstellen der Kommunikationsmodelle konnte festgestellt werden, dass sich die Benutzer im Bezug auf das System nicht weit voneinander unterscheiden. Die Benutzer spielen also weniger eine Rolle für das System als die Art der Aufgaben, die sie mit dem System bearbeiten. Aus diesem Grund wurde auch das Vorgehensmodell des User Centered Design ausgeschlossen.

Einsatz

Beim Einsatz des Usage Centered Design wurde auf iterative Anwendung des Vorgehensmodells geachtet. Das bedeutet, dass alle Modellierungen mehrmals durchlaufen wurden und dabei Verbesserungen vorgenommen worden sind. Dadurch konnte sichergestellt werden, dass die Schlüsse aus späteren Modellierungen auch mit in der

Überlegungen für die vorhergehenden Modellierungen genommen wurden. Zur besseren Übersicht wurden die Iterationen in diesem Dokument zusammengefasst.

Collaborative Requirements Dialog

Der Prozess des Vorgehensmodell des Usage Centered Design beginnt mit dem Collaborative Requirements Dialog, der die Erfordernisse und Anforderungen der Stakeholder an das System aufzeigt. Hierbei wird sowohl die Sicht der Benutzer als auch der Entwickler und Administratoren des Systems beschrieben.

<https://www.procontext.de/aktuelles/2012/06/nutzungskontext-erfordernisse-anforderungen-und-loesung-das-arbeitsmodell-des-usability-engineering.html>

Erfordernisse

“Eine notwendige Voraussetzung, die es ermöglicht, den in einem Sachverhalt des Nutzungskontexts enthaltenen Zweck effizient zu erfüllen” [QUELLE https://www.dakks.de/sites/default/files/71_sd_2_007_leitfaden_usability_1.3_0.pdf]

Ein System, das

1. den Benutzer an erstellte Notizen erinnert, sobald sich zwei Personen treffen.
2. Kontakte, mit denen man Erinnerungen verknüpfen möchte, speichert.
3. mobil, also auch mit weniger guter Internetverbindung, verfügbar ist.
4. für Gesprächspartner passende Themenvorschläge liefert.
5. die Datensicherheit beachtet.
6. auf möglichst vielen aktuellen Android Endgeräten zur Verfügung steht.

Anforderungen

Die Anforderungen werden von den Erfordernissen abgeleitet und werden im späteren Verlauf eingesetzt um bei der Evaluierung des System zu prüfen, ob die gestellten Anforderungen auch erfüllt wurden.

“Erfordernis oder Erwartung, das oder die festgelegt, üblicherweise vorausgesetzt oder verpflichtend ist.” [QUELLE

https://www.dakks.de/sites/default/files/71_sd_2_007_leitfaden_usability_1.3_0.pdf]

Sie werden in funktionale, organisationale, qualitative und technische Anforderungen unterteilt.

Funktionale Anforderungen

1. Das System muss den Benutzern die Möglichkeit bieten sich mittels Bluetooth zu koppeln.
 - 1.1. Das System muss eine Liste aller verfügbaren Bluetooth Geräte anzeigen.
 - 1.2. Das System muss dem Benutzer die Möglichkeit bieten ein Gerät zum Koppeln auszuwählen.

- 1.3. Das System muss auf beiden Geräten einen Dialog starten, ob die Kopplung erwünscht ist.
2. Das System muss fähig sein zu erkennen, wenn sich zwei bereits gekoppelte Geräte in der Nähe befinden.
 - 2.1. Das System besitzt diese Fähigkeit auch dann, wenn zwischenzeitlich Bluetooth ausgeschaltet wurde.
3. Das System muss eine Übersicht aller bereits gekoppelten Geräte bieten.
4. Das System muss dem Benutzer die Möglichkeit bieten ein bereits gekoppeltes Gerät auszuwählen, um für dieses Gerät eine neue Erinnerung zu erstellen.
5. Das System muss dem Benutzer die Möglichkeit bieten Erinnerungen zu erstellen.
 - 5.1. Das System muss mindestens ein Textfeld zur Erstellung bieten.
 - 5.2. Das System muss die Möglichkeit bieten, Themen-Labels zu der jeweiligen Erinnerung auszuwählen.
6. Das System muss fähig sein dem Ersteller die jeweilige Erinnerung als Benachrichtigung anzuzeigen, sobald er sich genähert hat.
7. Das System muss erkennen, wenn die Gesprächspartner in einem gewissen Zeitraum dieselben Themen-Labels gesetzt haben.
8. Das System muss bei gleichen Themen-Labels beiden Gesprächspartnern denselben informativen Text (Artikel etc.) zur Verfügung stellen.
9. ...

Organisationale Anforderungen

1. Das System muss im Play Store für alle Android Geräte ab Version 7.0 verfügbar sein.
2. Das System muss sowohl auf Tablet als auch auf Smartphones nutzbar sein.

Qualitative Anforderungen

1. Das System muss problemlos das Koppeln von Bluetooth Geräten ermöglichen.
2. Das System muss in jeder Situation Reaktionszeiten von unter einer Sekunde haben.
3. Das System darf nicht unerwartet abstürzen oder einfrieren.
4. Das System muss die Datensicherheit beachten.
5. Das System soll unabhängig von der Größe des Geräts uneingeschränkt nutzbar sein.
6. Das System muss auch für die Zielgruppe der älteren vergesslichen Leute gut lesbar sein.
7. Das System soll gebrauchstauglich im Sinne der Benutzer sein.

Technische Anforderungen

1. Das System muss einen Zuweisungs-Algorithmus der Themen-Labels auf Serverseite besitzen.
2. Der Server soll eine REST Schnittstelle besitzen.
3. Der Server soll Zugriff auf eine externe Datenhaltung besitzen.
 - 3.1. Die externe Datenhaltung soll über Googles Firebase realisiert werden.

- 3.2. Die externe Datenhaltung soll unabhängig vom Client sein, sodass bei Umstellung des externen Datensystems nur die Serverseite angepasst werden muss.
4. Die Kommunikation zwischen Client-App und Server, sowie zwischen Server und externer Datenhaltung, muss sicher sein.
5. ...

Fazit aus Anforderungen

Die Erfordernisse und Anforderungen geben an, was vorausgesetzt wird, damit der Benutzer seine Ziele mit dem System bewerkstelligen kann. Sie bilden die Grundlage für die weiteren Modellierungen und liefern einen tieferen Einblick über den Nutzungskontext des Systems. Aus den Anforderungen können zudem die ersten Schlüsse für Systemarchitektur und Datenstruktur gezogen werden, die durch die weiteren Schritte des Vorgehensmodells überprüft werden.

Domain Model

Bei der Vorgehensweise des Usage Centered Design wird die Prozess des Domain Model für die weitere Modellierungsbegründung genutzt. Hier kommen unsere Domänenrecherchen aus dem Konzept zu tragen.

Domänenmodell

Um einen besseren Überblick für den Modellierungsschritt Role Model zu erhalten, haben wir uns das Domänenmodell zur Domäne "Gespräch" wieder einbezogen. Innerhalb der Erarbeitung des Konzeptes wurde das Domänenmodell bereits iteriert.

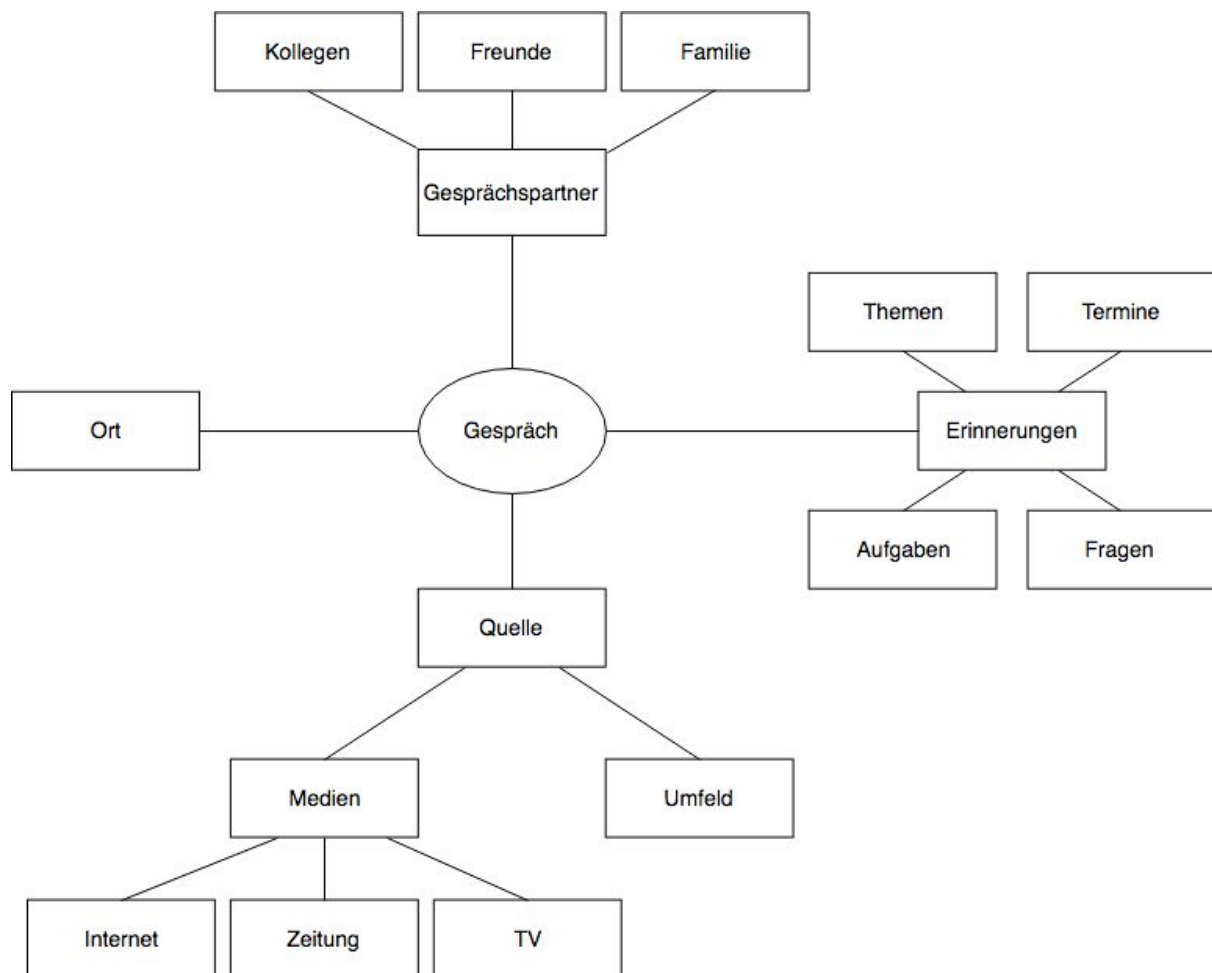


Abb. 3: Domänenmodell zur Domäne "Gespräch"

Klassendiagramm

// Ungefähr das, was wir im Konzept zunächst als Domänenmodell abgeben wollten

Stakeholderanalyse

Desweiteren bringt uns die Stakeholderanalyse aus dem Konzept Einblicke in die Rollen des Systems. Zusammen mit den Anforderungen und dem Domänenmodell sind hiermit die Grundkenntnisse über die Domäne und die Beziehung zum System vorbereitet um sie im Role Model auszuarbeiten.

Tabelle: Stakeholderanalyse

Bezeichnung	Beschreibung	Beziehung zum System	Priorität für das Projekt
Menschen im Arbeitsumfeld	Benutzer die viel unterwegs sind und dadurch oft Kontakten begegnen. Sowohl innerhalb eines Unternehmens, als auch	Interesse	sehr hoch

	außerhalb (Außendienstmitarbeiter, Berater).		
Menschen mit vielen Kontakten	Benutzer, die durch einen gewissen Umstand (Vermögen, Berühmtheit) viele Kontakte haben und diese Kontakte pflegen wollen.	Interesse	niedrig
Gestresste Menschen	Benutzer, die aufgrund von privatem Umfeld (Eltern, Alleinstehende) oder hohen Verantwortungen (Geschäftsführer, Ärzte) im Alltag viele Dinge im Kopf haben müssen.	Interesse	sehr hoch
Vergessliche Menschen	Benutzer, die wegen ihrem Alter und einer Erkrankung (Demenz, Alzheimer)	Interesse	hoch
Schüchterne Menschen	Benutzer, denen es schwer fällt Gespräche aufzubauen, da sie keinen Einstieg, zum Beispiel in Form von Themengebieten kennen.	Interesse	mittel
Ersteller von Medienprodukten (Wissenschaftler, Journalisten usw.)	Personen, die durch das Veröffentlichen von Medien (z.B. Zeitungsartikel) Themen für Gespräche liefern.	Interesse	mittel
Alle	Datensicherheit: Daten müssen abgesichert sein. Wie Daten gehalten und benutzt werden muss transparent sein.	Anrecht	hoch

Role Model

Für das Usage Centered Design liegt der Fokus eher auf dem Task Model, aber die Perspektive der Benutzer spielt trotzdem eine Rolle. Es ist wichtig, dass Entwickler und Benutzer in Interaktion treten, damit beide Perspektiven berücksichtigt werden. Nur so kann ein gutes System gewährleistet werden. Diese Perspektiven werden mit dem Role Model analysiert.

Das Role Model ist in simpler Form eine Liste von Benutzerrollen, beschrieben in Bedürfnisse, Interessen, Erwartungen, Benehmen und Verantwortlichkeiten.

Eine Benutzerrolle ist eine abstrakte Klasse, definiert durch eine bestimmte Beziehung zum System. Es wäre zu grob gesagt, dass sie eine Gruppe von Benutzern zusammenfassen. Denn User Roles sind eine Abstraktion und stellen keine echten Benutzer dar.

User Roles

User Roles Vorgehen

Das Vorgehen bei User Roles sieht folgendermaßen aus:

1. Compile: Zuerst wird Brainstorming verwendet, wobei dabei nicht diskutiert werden darf. Alle Ideen sind akzeptiert und es werden Aspekte der User Roles gesammelt, also einzelne Eigenschaften, Bedürfnisse etc.
2. Organize: Der erste Schritt wurde sorgfältig überprüft und nun geht es darum die gesammelten Aspekte zu sortieren und mit aussagekräftigen Titeln zu gruppieren.
3. Detail: Da es nun eine Gruppierung gibt und die entsprechenden Merkmale zugeordnet wurden, können diese nun verfeinert werden und Lücken geschlossen werden.
4. Refine. Nun wird das organisierte und detaillierte Model verfeinert und komplettiert. Hier wird es genauestens überprüft und Kritik geäußert.

Bei den User Roles sollten folgende Fragen beantwortet werden:

1. Wer würde oder könnte das System benutzen?
2. Welcher Klasse oder Gruppe gehören sie an?
3. Was zeichnet sie aus, wie sie das System benutzen?
4. Was charakterisiert ihre Beziehung zum System?
5. Was brauchen sie typischerweise vom System?
6. Wie verhalten sie sich gegenüber dem System und was erwarten sie vom Verhalten des Systems?

User Role Tabelle

Abstrakte Klasse und Wer? (1., 2.)	Wie ist die Benutzung des Systems? (3.)	Beziehung zum System (4.)	Bedürfnisse (5.)	Verhalten gegenüber System (6.)	Erwartetes Verhalten vom System (6.)
Durch Alltagsstress Vergessliche (Berufstätige, Familien, Freunde)	Schnelle und häufige Benutzung, viele Kontakte, viele Erinnerungen	Funktion der Erstellung der Erinnerungen sowie die Funktion der Themenvorschläge wichtig	Effiziente Bedienung, lieber Symbole als Text	schnell, oft, kurzlebig	Zuverlässigkeit, Schnelligkeit, Gebrauchstauglichkeit

Durch Arbeitsstress Vergessliche	Schnelle und häufige Benutzung, Arbeitskollegen als Kontakte, viele Erinnerungen	Funktion der Erstellung der Erinnerungen wichtig, Themenvorschläge unwichtig, da sonst nur Themen über die Arbeit	Effiziente Bedienung	schnell, oft, kurzlebig	Zuverlässigkeit, Schnelligkeit, Gebrauchstauglichkeit
Durch Haushalt Vergessliche	Häufige Benutzung, eher Familie als Kontakte, viele Erinnerungen, zu Hause in Ruhe Erinnerungen erstellen	Funktion der Erstellung der Erinnerungen sowie die Funktion der Themenvorschläge wichtig	Effiziente Bedienung	in Ruhe, oft, kurzlebig	Zuverlässigkeit, Gebrauchstauglichkeit
Organisatorische Benutzer (Außendienst)	Wohl überlegte Benutzung, ausgewählte Kontakte, Erinnerungen spezifisch.	Funktion der Erstellung der Erinnerungen wichtig, Themenvorschläge eher nicht	Effiziente Bedienung	in Ruhe, wohl überlegt, oft überprüfend	Zuverlässigkeit, Gebrauchstauglichkeit
Krankheitsbedingt Vergessliche (Alte Menschen)	langsame Benutzung, wenig Kontakte, simple Erinnerungen	Nur Funktion der Erstellung der Erinnerungen wichtig.	Viel Text, wenig Symbole, große, gut lesbare Buchstaben mit hohem Kontrast. Verständliche, einfach zu erklärende Bedienung.	langsam, ausgewählte Erinnerungen	Zuverlässigkeit, Einfachheit

Introvertierte Menschen (Schüchterne, ruhige Personen)	Ausgewählte Kontakte, wenige aber ausführliche Erinnerungen	Funktion der Themen-vorschläge besonders hilfreich, da gemeinsame Interessen schnell gefunden werden können.	Fachliche Informationen der Themenvorschläge, keine kritischen "Fake News"	schnell, oft, kurzlebig	Zuverlässigkeit, Schnelligkeit, Gebrauchstauglichkeit
Extrovertierte Menschen (viele Freunde, viele Gesprächsthemen)	Viele Kontakte, schnelle und häufige Benutzung, viele Erinnerungen	Funktion der Erstellung der Erinnerungen sowie die Funktion der Themen-vorschläge wichtig, da sie so neue interessante Themen ansprechen, die dem Gegenüber auch wirklich interessieren	Gute Übersicht aller Kontakte, keine Belästigung wegen zu häufigen wiederkehrenden Erinnerungen, effiziente Bedienung, lieber Symbole als Text	schnell, oft, kurzlebig	Zuverlässigkeit, Schnelligkeit, Gebrauchstauglichkeit

User Role Maps

Die User Role Map verbildlicht die Beziehungen der einzelnen User Roles zueinander. Somit ist auf einem Blick zu erkennen, wer die Benutzer des Systems sind und wie sie dieses nutzen. Die User Roles können auf 3 unterschiedliche Weisen verbunden sein:

Affinity

Die User Roles besitzen viele Gemeinsamkeiten. Sie besitzen ähnliche Interaktionen, Erwartungen und Charakteristiken. (Dargestellt durch eine gestrichelte Linie und dem Stichwort "resembles").

Classification

Manche User Roles stellen eine Unterklasse einer generellen User Role dar. Diese sind eine spezialisierte Version der Oberklasse und besitzen spezifische Eigenschaften, Interaktionen oder Bedürfnisse etc. (Dargestellt durch Pfeil von Unterklasse zur Oberklasse und dem Stichwort: "specializes").

Composition

Manche User Roles kombinieren die Eigenschaften und Charakteristiken von zwei oder mehr anderen Rollen. Das heißt, dass ein User die Rolle von zwei anderen User Roles einnehmen kann, auch wenn sich diese grundlegend unterscheiden. (Dargestellt durch Pfeil von kombinierter Klasse zu den speziellen Klassen und dem Stichwort "includes").

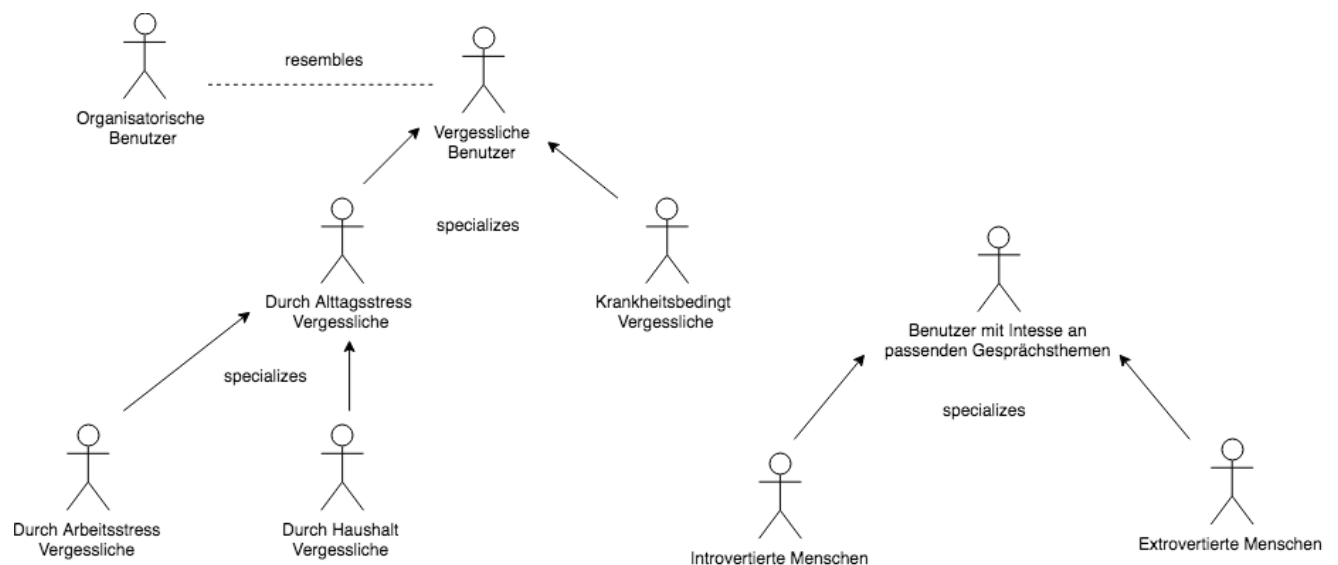


Abb 2: User Role Map

Fazit aus User Roles

Die User Roles ähneln sich alle dahingehend, dass es normale Benutzer des Systems sind. Es gibt keine Personen mit besonderen Rechten oder Pflichten, wie bspw. bei Planungssystemen.

Grundlegende Unterscheidungen gibt es bei den Benutzern, die vergesslich sind und denjenigen, für die das Gesprächsthema eine wichtige Rolle spielt. Dies spiegelt sich auf die zwei Hauptfunktionen des Systems.

Ein besonderer Fokus muss auf die krankheitsbedingten Vergesslichen Benutzern gesetzt werden. Diese haben als einzige Benutzerrolle wichtige zu berücksichtigende Bedürfnisse. Das System muss für ältere Personen, die nicht oft mit Technik umgehen, verständlich sein und das User Interface muss für Personen mit Sehschwächen angepasst sein.

Operational Model (optional)

// Modelliert den Kontext in dem das System benutzt wird

Operational Profiles

Fazit aus Operation Profiles

// wir sollten möglichst aus jedem Teilabschnitt ein Fazit ziehen um einen roten Faden zu bilden

Task Model

// Maßnahmen in strukturierter Form beschrieben, welche die Benutzer zur Erledigung ihrer Aufgaben ergreifen und welche Rollen sie einnehmen.

Essential Use Cases

// Sammlung von Beschreibungen der einzelnen Elemente

Use Case Maps

// Beziehungen der Elemente untereinander als Karte modelliert

Fazit aus Use Cases

// wir sollten möglichst aus jedem Teilabschnitt ein Fazit ziehen um einen roten Faden zu bilden

Content Model

// Artefakte, die das User Interface den Benutzern zu Verfügung stellen. Es wird beschrieben, wie der Inhalt repräsentiert wird.

Content Model

// Beziehungen der Elemente untereinander als Karte modelliert

Navigation Map

// Beziehungen der Elemente untereinander als Karte modelliert

Fazit aus Content Model

// wir sollten möglichst aus jedem Teilabschnitt ein Fazit ziehen um einen roten Faden zu bilden

Implementational Model

// Beinhaltet das visuelle Design des User Interface sowie die Beschreibungen seiner Operationen

Visual Design

Usability Inspection

Evaluierung

Systemkomponenten

Komponenten

// Nur beschreiben der einzelnen Komponenten

// Wo liegen die Komponenten

Dienste im Web

// Firebase beschreiben

Architektur

// Architekturdiagramm

// Beziehungen zwischen den Komponenten

Kommunikation zwischen den Komponenten

// asynchron /synchron in Bezug auf das Architekturdiagramm

Arten von Endgeräten

// Android ab 7.0

Datenstruktur

Persistente Datenhaltung

Datenformat

Quellen

1. Beispielquelle: <https://www.paulwatzlawick.de/axiome.html> (abgerufen am 08.11.2018)