Prozessorientierte Entwicklung von aufgaben- und ereignisorientierten Benutzungsschnittstellen für die Prozessführung mit Hilfe eines Usability-Engineering-Repositories (UsER)

Michael Herczeg, Marc Kammler & Amelie Roenspieß

Zusammenfassung

Der Beitrag stellt dar, wie mit Hilfe der kollaborativen Plattform UsER (Usability-Engineering-Repository) aufgaben- und ereignisorientierte interaktive Prozessführungssysteme und deren Benutzungsschnittstellen (Controls und Anzeigesysteme) entwickelt werden können. UsER hält über die den gesamten Entwicklungsprozess Informationen zu Organisation, Rollen, Aufgaben, Arbeitsobjekten, Benutzern, Kontexten, Szenarien, Screen-Mockups, Claims, Anforderungen, Features und Evaluationen in einem vernetzten Entwicklungsrepository verfügbar und erweiterbar vor, um durch mehrere Akteure kollaborativ und verteilt geeignete Lösungen informiert und hinterfragbar zu entwickeln, zu evaluieren und zu optimieren. Das Ziel ist dabei vor allem die Unterstützung von interdisziplinären Teams aus Analytikern, Gestaltern, Programmierern, Testern und nicht zuletzt Anwendern und Benutzern im informierten flexiblen Systementwicklungsprozess von der Zielsetzung über die Konkretisierung bis zur Evaluation. Das UsER-System wird von den Autoren in Zusammenarbeit mit Unternehmen entwickelt, erprobt, laufend durch neue Module angereichert und für diverse Anwendungsfelder wie Enterprise Resource Planning (ERP) und Prozessführungssysteme optimiert.

1 Einleitung

Prozessführungssysteme wie z.B. Kraftwerks- und Produktionsleitstände, Fahrzeugcockpits und Schiffsbrücken, Anästhesiemonitore oder Intensive-Care-Systeme sollen den Operateuren unter systemisch komplexen und zeitlich engen Randbedingungen aufgaben- und situationsgerechte, vor allem aber handlungsleitende Informationen und Interaktionsmöglichkeiten bieten.

Die Benutzungsschnittstellen von Prozessführungssystemen müssen dabei sowohl für Routineaufgaben (Normalbetrieb), wie auch für die Fortsetzung des gestörten Betriebs (Abweichungen, Anomalien, Ereignisse) ausgelegt sein. Dies erfordert zwei grundlegend unterschiedliche Denkweisen in der Konzeption von Benutzungsschnittstellen für sicherheitskritische Mensch-Maschine-Systeme, nämlich einerseits Systemkonzeptionen, die aus definierten *Aufgaben (Arbeits-*

DGLR-Bericht 2012-01/01

verfahren, Routineaufgaben) im Normalbetrieb abgeleitet werden und andererseits solche, die aus der Bearbeitung von Ereignissen (Abweichungen, Störungen, Störfällen, Unfällen) resultieren. Ein wichtiges Ziel ist es, dies in ein Gesamtparadigma in einem integrierten Prozessführungssystem zu verknüpfen.

Es gibt eine Vielzahl von Methoden und Hilfsmitteln zur Entwicklung gebrauchstauglicher Mensch-Maschine-Systeme, vor allem für die Analyse und Modellierung von *Organisationen, Benutzern, Aufgaben, Kontexten, Szenarien, Mockups* und den damit verbundenen *Evaluationsmethoden*. Diese Methoden stehen im Allgemeinen weitgehend isoliert nebeneinander und werden mit unterschiedlichsten Werkzeugen unterstützt, die meist nur gering oder überhaupt nicht integriert und vernetzt sind.

Die Usability-Engineering-Plattform *UsER (Usability-Engineering-Repository)* unterstützt die Entwicklung aufgaben- und ereignisorientierter interaktiver Systeme. Dabei können über den gesamten Entwicklungsprozess diverse Analysen und Konzepte entwickelt, in einem vernetzten, kollaborativ genutzten Entwicklungsrepository dokumentiert, evaluiert und verfeinert werden (siehe Bild 1):

- Organisationen (Organisationseinheiten, Stellen, Rollen),
- Aufgaben (Tätigkeiten, Aktivitäten, Handlungen),
- Arbeitsobjekte (Arbeitsgegenstände, Informationsobjekte),
- Benutzer (Benutzerklassen, Stereotypen, Personas),
- Szenarien (Problem- und Interaktionsszenarien) und Kontexte (räumlich, zeitlich, organisatorisch),
- multimediale Benutzungsoberflächen und Visualisierungen (Mockups, Graphiken),
- Ziele, Anforderungen und Funktionen (Claims, Requirements, Features) sowie
- Evaluationen (Befragungen und ihre Auswertungen).

Mit der Entwicklung der Plattform UsER ist das Ziel verbunden, geeignete Systemlösungen informiert und hinterfragbar zu entwickeln, zu evaluieren und zu optimieren und nicht etwa die bislang wenig erfolgreiche automatische Ableitung von Benutzungsschnittstellen aus Spezifikationen zu leisten (vgl. Meixner et al., 2011). Das System dient der Unterstützung von verteilten Teams aus Analytikern, Systemarchitekten, Gestaltern, Programmierern, Testern und nicht zuletzt auch Anwendern und Benutzern im offenen, aber informierten und hinterfragbaren kreativen Prozess von der Zielsetzung und der Analyse über die Konkretisierung bis zu ihrer Evaluation.

Im Folgenden werden die einzelnen Module von UsER anhand von Beispielen für die Entwicklung eines Alarmmonitors und seiner Visualisierung beschrieben.

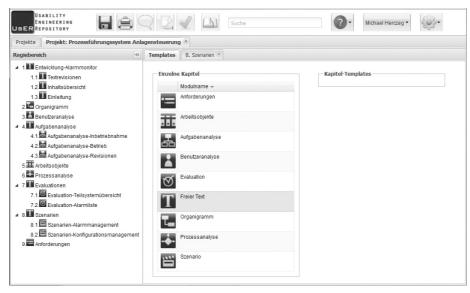


Bild 1: UsER-Rahmensystem mit den einzelnen Modulen (Mitte) und der damit definierten variablen, linearen Dokumentenstruktur (links)

2 Organisation, Aufgaben und Rollen – der Betrieb

Die Entwicklung eines Prozessführungssystems findet im Allgemeinen im Kontext wohldefinierter betrieblicher Strukturen statt. Ausnahmen sind beispielsweise die Nutzung von sicherheitskritischen Konsumgütern wie Pkws oder Hausinstallationen. Die wichtigsten betrieblichen Strukturen werden in Form von Organisationsstrukturen, Stellenplänen, Aufgaben, Rollen und mehr oder weniger konkreten Benutzern (siehe Abschnitt 4.1) beschrieben. Dies schafft den Rahmen für die Konzeption von Prozessführungssystemen, da so die Verteilung und Zuordnung von Aufgaben, Zuständigkeiten und Qualifizierungen geklärt werden. In UsER wurden dazu Modellierungsmodule geschaffen, mit denen Organisationsstrukturen und Aufgaben modelliert werden können.

2.1 Aufgabenanalyse

Es gibt eine Vielzahl von Methoden zur Aufgabenanalyse, wobei gerade die frühen Ansätze aus dem Bereich der Ingenieursysteme und Human Factors stammen, mit dem Ziel, klare Aufgabenstrukturen für Operateure auf Technik, vor allem auf Leitwarten, Cockpits und medizintechnische Systeme abzubilden (Kirwan & Ainsworth, 1992; Shepherd, 1998; Herczeg, 2001). In nahezu allen diesen Analysemethoden findet sich das Prinzip der hierarischen oder auch heterarchischen Aufgabenstrukturierung und Aufgabenverfeinerung. Entsprechend spricht man von *Hierarchical Task Analysis (HTA)*.

UsER stellt ein Modul für die HTA bereit (siehe Bild 2). Die Aufgabenknoten in der hierarchischen Aufgabenzergliederung können dort auch mit den Rollen

verknüpft werden, die diese Aufgaben ausführen sollen (siehe dazu auch Modul Organisationsanalyse in Abschnitt 2.2). Bei Bedarf können bei der Aufgabenmodellierung diese Rollen auch erzeugt werden, falls sie noch nicht existieren.

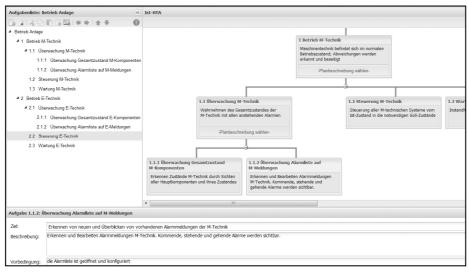


Bild 2: UsER-Modul Aufgabenanalyse mit einer Aufgabenmodellierung für die Nutzung einer Alarmliste auf einem Alarmmonitor

Neben den Rollen können in den Aufgabenknoten auch Verweise auf Dialoge in Form von Mockups (siehe Abschnitt 5) gesetzt werden, also auf Interaktionskomponenten, mit denen die Akteure die jeweilige Aufgabe durchführen können. Dies geschieht im Allgemeinen erst in der Konzeptionsphase. Die Aufgabenzerlegung dient aber bereits früh als Strukturierungsmittel, um Dialoge aufgabengerecht in Unterdialoge bzw. komplexe Anzeigen in ihre Teile zu zerlegen. Die Gestaltung eines interaktiven Systems aus einer Aufgabenmodellierung heraus, trifft den Kern des aufgaben- und benutzerzentrierten Entwickelns im Sinne des Usability-Engineerings. Für Routineaufgaben (Normalbetrieb) ist es ohnehin ein bewährtes Vorgehen, *Standard Operating Procedures (SOPs)* zu entwickeln. Für den Fall einer eventuellen Ereignisbearbeitung (Contingency, gestörter Betrieb), sind es Aufgaben, wie z.B. spezielle *Emergency Standard Operating Procedures (ESOPs)*, die für bestimmte Klassen antizierbarer Ereignisse als Aufgaben vordefiniert werden können.

2.2 Organisationsanalyse

Die Organisationsanalyse in UsER orientiert sich am klassischen Modell der hierarchischen Strukturierung einer Organisation in Organisationseinheiten und Stellen, die dann am Ende der Zergliederung von konkreten Rollen funktional ausgefüllt werden müssen (siehe Bild 3). Wenn nicht schon in der Aufgabenanalyse geschehen, können hierbei auch Rollen neu definiert und mit Aufgaben

verbunden werden. Alle Aufgaben, die im Rahmen einer Organisation anfallen, müssen auf definierte Rollen und entsprechende Stellen verteilt werden.

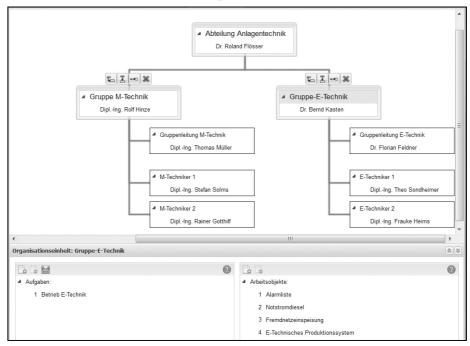


Bild 3: UsER-Modul Organisationsanalyse mit Organisationseinheiten, Rollen und Stellen (Konkretisierungen sind Personas; siehe Abschnitt 4.1.3)

Rollen in Verbindung mit Aufgaben und Organisationsstrukturen liefern wichtige Hinweise auf die Kompetenzen der Benutzer, bzw. auf die Kompetenzbildung im Sinne von Ausbildung, Weiterbildung und Training, um die Aufgaben mit Hilfe der Arbeitsmittel, hier den Dialogen des Prozessführungssystems, leisten zu können (Weiteres dazu unter Benutzermodellierung in Abschnitt 4.1).

3 Arbeitsobjekte und Informationsmodellierung – die Arbeitsgegenstände

Gerade im Zusammenhang mit Prozessführung ist die Informationsmodellierung und Informationsergonomie von zentraler Bedeutung (Herczeg & Stein, 2012). Die Informationsflüsse aus dem bzw. in den Prozess (Sensoren und Aktuatoren) bestimmen letztlich das Geschehen für die Operateure. Es sind die wahrgenommenen Ist-Zustände bzw. die geplanten Soll-Zustände des Prozesses, die das aufgaben- sowie das ereignisbasierte Handeln antreiben (vgl. Rasmussen, 1984, 1985). Aus diesem Grund leiten sich auch die Dialoge in Prozessführungssystemen weitgehend aus der Darstellung von Zuständen von mehr oder weniger abstrakten Arbeitsobjekten ab.

Auch die Unterstützung der Situation Awareness als die Grundlage und Triebfeder der Prozessführung hängt davon ab, wie gut die Operateure ein mentales Abbild des bisherigen, aktuellen und künftigen Anwendungsprozesses herstellen können, das sie befähigt, geeignete Entscheidungen und Handlungen vorzunehmen, um Ist-Zustände in geeignete Soll-Zustände zu transformieren (Endsley et al., 2003). Bei Prozessführungssystemen, anders als bei vielen anderen Applikationen, wie z.B. solchen im Bürobereich, sind Echtzeitanforderungen zu stellen, so dass es möglich wird, die aktuellen Zustände noch als ausreichend gültig zu erklären, sowie Handlungsmöglichkeiten zu schaffen, um Zielzustände im verfügbaren Zeitfenster erreichen zu können.



Bild 4: UsER-Modul zur Modellierung von Arbeits- und Informationsobjekten

Im UsER-System wird ein bislang noch einfaches Modul zur Informationsmodellierung angeboten, das es ermöglicht, Informationsstrukturen (Objektklassen, ihre Attribute und Modellwerte) zu definieren, die dann Ausgangspunkte und Quellen für interaktive Displaysysteme (Controls und Anzeigen) darstellen können (siehe Bilder 4 und 6). Dabei können einzelne Arbeits- oder Informationsobjekte (z.B. Systemkomponenten und ihre Zustände) oder auch listenartige Informationsmengen (z.B. Komponentengruppen oder Alarmmeldungen) definiert werden. Diese Objekte dienen später vor allem auch als Bezugspunkte beim Erstellen von Mockups für Bildschirmdarstellungen und Dialoge. Den Objekten können wiederum Aufgaben und Rollen zugeordnet werden.

Bei der benutzer- und aufgabenzentrierten Entwicklung ist bei der Arbeitsobjektund Informationsmodellierung darauf zu achten, dass diese aus Sicht des Operateurs und seiner Aufgaben erfolgt. Je besser diese mit der späteren Informationsmodellierung des sichtbaren Prozessführungssystems korrespondiert, desto direkter wird der Informationsabruf bzw. die Einwirkmöglichkeiten auf den Prozess werden (Herczeg, 2009). Kongruente Informationsmodelle zwischen Benutzer und System sind auch beim *Supervisory Control*, einer sehr eng verzahnten Mensch-Maschine-Interaktion mit dynamischer Teilautomatisierung, von zentraler Bedeutung. Sheridan (1988, S. 170) weist auf die Schwierigkeiten und die Bedeutung der Verbindung mentaler Modelle und computerbasierter Entscheidungshilfen hin.

4 Benutzer, Szenarien und Kontexte – die Situationen der Nutzung

Eine vollständige Modellierung eines Arbeitsbereiches unter Berücksichtigung komplexer Kontexte ist praktisch nicht möglich. Deshalb konzentriert man sich im Usability-Engineering auf schwierige und kritische Situationen, teils aus antizipativer Sicht (Risikoanalysen), teils abgeleitet aus betrieblichen Erfahrungen (Incident- oder Accident-Analysen). Solche Situationen beschreiben, wie Operateure ihre Aufgaben unter mehr oder weniger konkreten Randbedingungen und Anforderungen leisten oder auch nicht leisten können.

4.1 Benutzermodellierung

Für die Beschreibung von Szenarien werden *Benutzermodelle* benötigt. Diese können mehr oder weniger abstrakt von *Benutzerklassen* über *Stereotypen* bis hin zu konkreten *Personas* reichen (Herczeg, 2009). Die Modellierung von Benutzern als die späteren Platzhalter für Rollen im System ist wichtig für eine möglichst realistische Einschätzung des Wissens, der Fähigkeiten und der Fertigkeiten von Operateuren, die mehr oder weniger erfolgreich bestimmte Aufgaben durchführen oder auftretende Probleme zu lösen im Stande sind. Im UsER-System lassen sich alle genannten Abstraktionsformen mit dem *Benutzermodul* modellieren, um diese dann an Rollen und Stellen zu binden oder in Szenarien zu verwenden. Mit dem Benutzermodul werden auf diese Weise letztlich Kompetenzen, Qualifikationen und Verhaltensweisen von Operateuren anschaulich modelliert.

4.1.1 Benutzerklassen

Benutzerklassen sind abstrakte Beschreibungen von Zielgruppen, die mit dem interaktiven System, hier dem Prozessführungssystem erfolgreich arbeiten können sollen. Sie charakterisieren

- Ziele und Randbedingungen bei der Erfüllung von Aufgaben, die mit einem bestimmten Anwendungssystem bearbeitet werden sollen;
- vorhandene Erfahrungen mit anderen interaktiven Systemen, die bislang genutzt wurden und möglicherweise weiterhin parallel genutzt werden;
- besondere sensomotorische Fertigkeiten (Automatismen) mit interaktiven Systemen, z.B. bezüglich Tastaturen und spezieller Ein- und Ausgabegeräte;
- Erwartungen und Präferenzen in Bezug auf ein neues oder bestehendes System.

4.1.2 Stereotypen

Anstatt Benutzer nur durch Benutzerklassen zu abstrahieren, hat es sich bewährt, Stereotypen für Benutzer als weitere Konkretisierungen von Benutzerklassen zu beschreiben.

So kann ein Operateur beispielsweise als "technikverliebter Ingenieur" beschrieben werden, der sein System auch im Detail gut kennt, aber die Meinung anderer eher nur marginal berücksichtigt. Auf diese Weise kann beispielweise in Szenarien dargestellt werden, wie Konfliktsituationen durch menschliches Verhalten entstehen können und wie sichergestellt wird, dass möglichst alle relevanten Informationen in den Entscheidungs- und Handlungsprozess einfließen, z.B. als Szenarien für das *Crew Resource Management (CRM)* oder für die Arbeit von Krisenstäben.

4.1.3 Personas

Geht man in der Methodik der Stereotypen einen Konkretisierungsschritt weiter, so beschreibt man konkrete, fiktive Benutzer. Als Methode wurde hierzu vorgeschlagen, Benutzer als sogenannte *Personas* zu beschreiben (Cooper, 1999).

Personas zwingen dazu, über konkrete Benutzer anstatt über Benutzer im Allgemeinen nachzudenken. Designalternativen können an Personas direkt geprüft und entschieden werden. Alan Cooper (1999) schreibt dazu:

"As we begin to develop ideas for design solutions, we can constantly hold them up against our personas to see how well we have done."

Personas sind somit keine abstrahierten Benutzer wie Benutzerklassen oder Stereotypen, sondern Konkretisierungen, die man detailreich hinsichtlich Zielen, Fertigkeiten, Kenntnissen, Präferenzen und Verhalten hinterfragen kann. Personas sind Persönlichkeiten und Charaktere. Sie sind vorstellbar und glaubhaft und werden in Teamarbeit erarbeitet, geprüft und dokumentiert. Sie werden für die Entwicklung interaktiver Systeme als besonders hilfreich beschrieben.

Personas werden prosaisch, am besten unter Verwendung einiger Zitate (z.B. fiktiver O-Ton) und fotografischem Portrait (leihweise von einer sonst fremden Person) beschrieben. Es kann praktisch sein, möglichst detailliert biografische Daten einzuflechten. Noch wichtiger aber sind Ausführungen, die es erlauben, sich in die Person hineinzuversetzen. Typische Persona-Beschreibungen geben Informationen zu:

- Name, Vorname, Titel,
- Geschlecht, Alter, Familienstand und Familienverhältnisse,
- Nationalität und Sprachkenntnisse,
- erlernter und praktizierter Beruf,
- Fachkenntnisse, Rollen, Aufgabengebiete und Interessen,
- Erfahrungen und Probleme mit Computern und anderer Technik,
- Einstellungen zu Firma, Arbeit, Familie, eigener und fremder Kompetenz.

Weitere Details können hilfreich sein. Personas können in verschiedenen Typen charakterisiert werden, wie z.B. Primary, Secondary oder Negative Personas.

4.2 Szenarien und Kontexte

Szenarien dienen dazu, ein realistisches und am Einzelfall tiefes Verständnis über Situationen zu erlangen, in denen Benutzer ihre Systeme erfolgreich oder auch defizitär bedienen. Dazu bedarf es Beschreibungen, bei denen mehr oder weniger konkret bestimmte Benutzer bestimmte Aktivitäten in einer durch das System bestimmten Weise mit den für das System typischen Wirkungen ausführen.

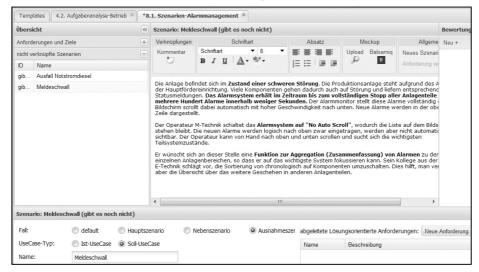


Bild 5: UsER-Modul Szenarien-Editor mit einem Interaktionsszenario zum Umgang mit einem Meldeschwall auf dem Alarmmonitor

Die Situationen und Kontexte werden dabei zunächst in *Problemszenarien* (Problemsituationen und Zielbeschreibungen) beschrieben, die schrittweise über *Aktivitätsszenarien* (Handlungsbeschreibungen) und *Informationsszenarien* (Erwerb und Produktion von Information) bis hin zu *Interaktionsszenarien* konkretisiert werden können (vgl. Rosson & Carroll, 2002). Hierbei lassen sich auch System-Mockups (z.B. Bildschirmentwürfe) gut einsetzen. Die Akteure sind, je nach Abstraktionsgrad der Szenarien, die schon oben beschriebenen Benutzerklassen, Stereotypen oder konkreten Personas, die dann auch im UsER-System als Referenzen zur Beschreibung von Szenarien benannt, charakterisiert und vernetzbar sind.

Das Szenarien-Modul in UsER ist ein Rich-Text-Editor, der es erlaubt, Referenzen auf schon vorhandene UsER-Entitäten wie Benutzer, Rollen, Aufgaben, Organisationseinheiten zu setzen (siehe Bild 5). Dadurch entstehen hypermediale Beschreibungen, ähnlich wie durch HTML, deren Knoten die Szenarien selbst und deren Links UsER-Entitäten sowie Annotationen darstellen.

5 Lösungsansätze und Entwürfe – Ideen und Lösungen

Aus all den genannten Analysen und Modellierungen sollen Schritt für Schritt verfeinerte, mehr oder weniger interaktive Benutzungsoberflächen (Dialoge, Visualisierungen) entwickelt werden. In zunehmender Konkretisierung können *Mockups* oder *Screenshots* im UsER-System modelliert oder importiert und in Storyboards sequenziert werden. Ein *User-Interface-Styleguide* wird integriert, um externe oder eigene Standards zu sichten und zu prüfen.

5.1 Mockups

Im UsER-System wurde mit *Balsamiq* ein *Mockup-Editor* integriert, der es mit wenig Aufwand erlaubt, Bildschirmoberflächen zu entwerfen und in UsER zu importieren. Gerade im Sinne des Rapid Prototyping und anderer agiler Entwicklungsmethoden ist es wichtig, schnell und mit wenig Aufwand anschauliche Bildschirmentwürfe zu erstellen, mit denen erste Konzepte für Visualisierungen und Interaktionen entwickelt und diskutiert werden können. Eine Vorstufe wären klassische *Paper-Prototypes*, die aber durch schnelle Mockup-Systeme weitgehend entbehrlich werden.

Durch eine Kopplung mit einem Styleguide- und Recommender-System (siehe Abschnitt 5.4) kann das Mockup-Modul auch in Abhängigkeit vom Informationsmodell Vorschläge für Gestaltungsmuster liefern (z.B. Formulare für Einzelobjekte, Tabellen für Mengen gleichartiger Daten oder Sterndiagramme für ein aggregiertes Zustandsobjekt).

5.2 Screenshots

Je weiter Systemkonzepte gediehen sind, desto konkreter können Dialogentwürfe dann mit Mitteln wie *Graphikeditoren* (z.B. Adobe Photoshop, Microsoft Visio), Tabellensystemen mit Graphikintegration (z.B. Microsoft Excel) oder speziellen *User Interface Buildern* entwickelt werden. Graphiken können aus diesen Werkzeugen in gängigen Bildformaten exportiert, anschließend in UsER importiert und dort annotiert werden (siehe Bild 6).

	ALARMLISTE									
Fehlerklasse	Status	Teilsystem	Komponente	Attribut	ID	Datum	Uhrzeit	OP	#	FK
WARNUNG		Verpackung	Folie	NORM	730229	08.07.2012	09:58:23	OFFEN	120708 63	WA
ALARM		Reinigung	Wassertank 7	EMPTY	740882	08.07.2012	09:52:52	OFFEN	120708 62	AL
ALARM		Notdiesel	Zündung	NORM	307663	08.07.2012	09:45:32	ок	120708 61	AL
ALARM		Verteiler 2	Bremse	NORM	200309	08.07.2012	09:31:37		120708 60	AL
INFO		Verpackung	Folie	WARTUNG	730229	08.07.2012	09:26:18	OFFEN	120708 59	IN
WARNUNG		Antrieb 3	Schmierung	NORM	540600	08.07.2012	09:16:26	ок	120708 58	WA
WARNUNG		Verpackung	Druck	NORM	730800	08.07.2012	09:14:20		120708 57	WA
WARNUNG		Reinigung	Wassertank 3	NORM	740874	08.07.2012	09:05:29		120708 56	WA
WARNUNG		Umluft	Ventilator 12	NORM	630901	08.07.2012	09:02:39		120708 55	WA
WARNUNG		Verteiler 1	Separator 4	LOW	201443	08.07.2012	08:51:36	OFFEN	120708 54	WA
WARNUNG		Reinigung	Wassertank 3	LOW	740874	08.07.2012	08:45:12		120708 53	WA
WARNUNG		Reinigung	Wassertank 7	LOW	740882	08.07.2012	08:40:46	OFFEN	120708 52	WA
INFO		Notdiesel	Zündung	WARTUNG	307663	08.07.2012	08:25:50	ок	120708 51	IN
WARNUNG		Verteiler	Motor 3	NORM	200486	08.07.2012	08:17:03	ок	120708 50	WA
INFO		Antrieb 3	Schmierung	WARTUNG	540600	08.07.2012	08:02:13		120708 49	IN
WARNUNG		Verpackung	Folie	LOW	730229	08.07.2012	08:01:30	OFFEN	120708 48	WA
WARNUNG		Verteiler	Motor 3	SLOW	200486	08.07.2012	07:59:04	ок	120708 47	WA
WARNUNG		Verpackung	Druck	LOW	730800	08.07.2012	07:48:45		120708 46	WA
ALARM		Abgas	Filter	CLEAR	309550	08.07.2012	07:39:02	ок	120708 45	AL
WARNUNG		Antrieb 3	Schmierung	LOW	540600	08.07.2012	07:30:01	ок	120708 44	WA
ALARM		Verteiler 2	Bremse	нот	200309	08.07.2012	07:15:10		120708 43	AL
ALARM		Notdiesel	Zündung	STOERUNG	307663	08.07.2012	07:12:20	ОК	120708 42	AL

Bild 6: Mockup-Graphik eines Systemdialogs

5.3 Storyboards

Existiert eine zeitliche Folge von graphisch dokumentierten Dialogschritten, so kann daraus mit Hilfe des Szenario-Editors ein *Interaktionsszenario (eine Art Storyboard)* entwickelt werden, das die einzelnen Interaktionsschritte des Operateurs zeigt und diskutierbar macht. Solche Diskussionen werden im Szenario annotiert und stehen für spätere Optimierungen des Systems dokumentiert und vernetzt mit Benutzern, Rollen, Aufgaben und Mockups zur Verfügung.

5.4 Styleguides

Für UsER wird ein *Styleguide-Modul* entwickelt, das sowohl die Erstellung eines Styleguides als auch die Überprüfung der Entwürfe und spätere Implementierungen auf Styleguide-Konformität unterstützen wird. Die Entwicklung von Styleguides ist für interaktive Systeme im Allgemeinen und für Prozessführungssysteme im Besonderen mit hohem Aufwand verbunden. UsER wird deshalb ein flexibles Rahmensystem anbieten, mit dem firmen- oder anwendungsspezifische Styleguides entwickelt und im Rahmen der Evaluation (siehe Abschnitt 7) auf Einhaltung überprüft werden können. Werden aus dem Styleguide Gestaltungsmuster entnommen, so ist bei konsequenter Umsetzung in der Implementierung die Styleguidekonformität bereits im Vorfeld in hohem Maße gesichert.

6 Claims, Requirements und Features – die Umsetzung und Realisierung

Im Rahmen des *Requirement-Engineering*, verbunden mit *Contextual Design*, kommt man von den ersten Ideen über konkrete Anforderungen bis zur Spezifikation der Funktionalität des Systems. UsER sieht vor, dass *Claims (Ziele, Ideen)* in *Requirements (Anforderungen)* und zuletzt in *Features (Funktionen)*

systematisch und verknüpft übergeführt werden können (vgl. Rosson & Carroll, 2002).

Tabellarische Darstellungen mit eindeutigen Schlüsseln ermöglichen Revisionen und Roll-Out-Pläne eines Systems im Sinne des klassischen Systems- und Software-Engineerings zu planen und zu dokumentieren. Verknüpfungen zwischen Claims, Requirements und Features erlauben später zu hinterfragen, woraus ein Feature oder eine Anforderung abgeleitet worden ist.

7 Evaluationen – die Zielerreichung, Bewertung und Optimierung

Im Usability-Engineering beginnt man bereits mit den ersten Ideen und Konzepten diese systematisch zu evaluieren und zu dokumentieren. Man spricht hierbei auch von *formativer Evaluation*, um darauf hinzuweisen, dass das System durch eine laufende kritische Reflektion geformt wird. Mehr im Sinne der klassischen Qualitätssicherung wird das erstellte Gesamtsystem dann am Ende der Entwicklung auf die Einhaltung von Anforderungen überprüft. Man spricht hierbei auch von *summativer Evaluation*. UsER unterstützt grundsätzlich beide Evaluationsformen

7.1 Formative Evaluationen

Die formative Evaluation mit Hilfe des UsER-Systems ergibt sich aus folgenden Ansätzen:

- a) die Benutzermodellierungen erlauben eine ständige informierte Diskussion über die Fähigkeiten und Erwartungen der Benutzer;
- b) die Realisierung von Mockups ermöglicht die frühe Diskussion von Entwürfen für die Benutzungsschnittstelle mit Benutzern und Gebietsexperten;
- c) mit Hilfe von Szenarien können kritische Nutzungssituationen und Aufgaben durchgespielt und frühzeitig evaluiert werden.

Die Erkenntnisse der formativen Evaluationsschritte können in Form von Annotationen an den einzelnen Entitäten dokumentiert oder direkt in neue Revisionen der einzelnen Analysen und Konzepte umgesetzt werden.

7.2 Summative Evaluationen

Während formative Evaluationen den Entwicklungsprozess antreiben sollen, dienen summative Evaluationen eher im Sinne einer abschließenden Qualitätssicherung der Prüfung, ob das interaktive System seinen Zweck und die damit verbundenen Anforderungen erfüllt.

Im UsER-System wird zu diesem Zweck ein Evaluationsmodul bereitgestellt, mit dem man im Stil von Benutzer- und Anwenderbefragungen, z.B. nach ISONORM mit Bezug auf die ISO 9241-110, oder auch anderen Evaluationsmethoden, webbasierte Fragebögen generieren, Benutzerbewertungen einholen, halbautomatisch auswerten und über Systemversionen Trends verfolgen kann (siehe Bild 7).

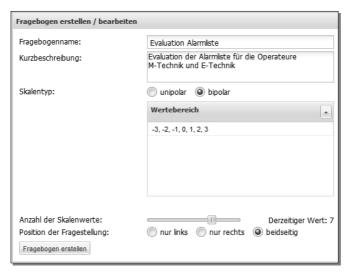


Bild 7: UsER-Modul Evaluation zur Generierung, Dokumentation und Auswertung von Fragebögen

8 Entwicklungsstrategien

Mit Hilfe der UsER-Umgebung lassen sich individuelle Entwicklungsstrategien und Entwicklungsprozesse unterstützen. Mit Ausnahme der Realisierung und der Ereignisanalyse lassen sich alle Projektphasen mit UsER unterstützen.

8.1 Aufgabenbasierte Entwicklung

In einer typischen Vorgehensweise bei hochdefinierten Arbeitssystemen dienen die Aufgaben als Ausgangspunkt. Mit Hilfe von UsER ließe sich hierzu, ohne Darstellung von Iterationen, in folgenden Schritten vorgehen:

- a) Formulieren von Zielen in Form von Claims
- b) Erstellen hierarchischer Aufgabenmodelle
- c) Erstellen von Informations- und Arbeitsobjekten
- d) Zuordnen der Aufgaben, Informations- und Arbeitsobjekte zu organisatorischen Rollen und Stellen
- e) Modellieren der Benutzereigenschaften mittels Benutzerklassen und Personas
- f) Formulieren von Anforderungen
- g) Entwerfen der Benutzungsschnittstelle in Form von Mockups
- h) Entwickeln verfeinerter Entwürfe mittels User Interface Builder oder Graphikeditor
- i) Formulieren von Features und Festlegung in Systemversionen
- j) Realisieren des Systems
- k) Evaluieren des Systems mit dem Evaluationsmodul nach ISONORM

8.2 Ereignisbasierte Entwicklung

Im Fall von ereignisbasierter Optimierung von Systemen wäre folgende Vorgehensweise mit UsER möglich:

- a) Erstellen einer ganzheitliche Ereignisanalyse
- b) Überprüfen und Optimieren der Informations- und Arbeitsobjekte
- c) Identifizieren von Schwachstellen am bestehenden Systems mit Hilfe von annotierten Screenshots oder Fotos mit Hilfe des Szenarien-Editors
- d) Überprüfen und Optimieren des Aufgabenstruktur und der Zuständigkeiten
- e) Entwerfen eines verbesserten Systems mit Hilfe von Mockups und anschließende Diskussion dieser Mockups mit Operateuren und Domänenexperten
- f) Formulieren von neuen Anforderungen
- g) Entwerfen detaillierter Bildschirmlayouts mit einem Graphiksystem und Formulieren neuer Anforderungen
- h) Umsetzen der Anforderungen in funktionale Features oder funktionale Änderungen
- i) Realisieren des optimierten Systems
- j) Evaluieren der n\u00e4chsten Revision des Systems mit dem Evaluationsmodul nach ISONORM und Vergleich mit den fr\u00fcheren Ergebnissen

9 Zusammenfassung und Ausblick

UsER ist eine Plattform zur Praktizierung der Methoden und Prinzipien des Usability-Engineering. Durch eine Reihe von Modulen lassen sich alle wichtigen Analyse-, Entwurfs- und Evaluationsschritte systemunterstützt durchführen und dokumentieren. Die Dokumentation ist eine datenbankgestützte Vernetzung von Entitäten, die in einzelnen Editoren (Modulen) dargestellt und bearbeitet werden können. Als Ergebnis erhält man eine flexible linearisierte, kapitelartige Dokumentenstruktur, die im Entwicklungsprozess verschiedene Funktionen wie Lastenheft, Arbeitsanalyse, Risikoanalyse, Pflichtenheft, Designmodell oder auch Evaluationsbericht einnehmen kann. UsER unterstützt Entwickler komplexer interaktiver Systeme ohne den Anspruch der Vollständigkeit der Modellierungen zu erheben. Es liegt bei den Entwicklern, die wichtigen Modellierungen herzustellen, zu kommunizieren, zu annotieren und in einen iterativen kollaborativen Prozess einzuspeisen. Gerade im Bereich sicherheitskritischer Systeme können so auf Grundlage von Aufgaben- und Ereignisanalysen Alternativen entwickelt, dokumentiert und kritisch reflektiert werden.

Literatur

Cooper, A. (1999). *The Inmates are Running the Asylum*. Indianapolis: SAMS. Endsley, M.R., Bolté, B. & Jones, D.G. (2003). *Designing for Situation Awareness*. London: Taylor & Francis.

- Herczeg, M. (2001). A Task Analysis and Design Framework for Management Systems and Decision Support Systems. ACIS International Journal of Computer & Information Science, 2(3), 127-138.
- Herczeg, M. (2003). Diagnostische Repositorien zur Unterstützung kollaborativer Entscheidungsprozesse. In M. Grandt (Hrsg.) *Entscheidungsunterstützung für die Fahrzeug- und Prozessführung* (DGLR-Bericht 2003-04, S. 117-131). Bonn: Deutsche Gesellschaft für Luft- und Raumfahrt e.V.
- Herczeg, M. (2006a). Interaktionsdesign. München: Oldenbourg-Verlag.
- Herczeg, M. (2006b). Analyse und Gestaltung multimedialer interaktiver Systeme. In B. Zimolong & U. Konradt (Hrsg.), *Ingenieurpsychologie, Enzyklopädie der Psychologie* (D-III, Bd. 2, S. 531-562). Göttingen: Hogrefe.
- Herczeg, M. (2009). Software-Ergonomie (3. Auflage). München: Oldenbourg-Verlag.
- Herczeg, M. & Stein, M. (2012). Human Aspects of Information Ergonomics. In M. Stein & P. Sandl (Eds.), *Information Ergonomics* (pp 59-98). Berlin, Heidelberg: Springer-Verlag.
- Kammler, M., Roenspieß, A. & Herczeg, M. (2012). UsER: Ein modulares Usability-Engineering-Repository. In *Tagungsband Mensch & Computer 2012* (09.-12.09.2012, Konstanz). München: Oldenbourg-Wissenschaftsverlag.
- Kirwan, B. & Ainsworth, L.K. (Eds.) (1992). *A Guide to Task Analysis*. London: Taylor & Francis.
- Meixner, G., Paternò, F. & Vanderdonckt, J. (2011). Past, Present, and Future of Model-Based User Interface Development. *i-com*, 10(3), 2-11.
- Rasmussen, J. (1984). Strategies for State Identification and Diagnosis. In W.B. Rouse (Ed.), *Advances in Man-Machine Systems* (Vol. 1, pp 139-193). Greenwich: JAI Press.
- Rasmussen, J. (1985). The Role of Hierarchical Knowledge Representation in Decisionmaking and System Management. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(2), 234-243.
- Rosson, M.B. & Carroll, J.M. (2002). *Usability Engineering Scenario-Based Development of Human-Computer-Interaction*. San Francisco: Morgan Kaufmann.
- Shepherd, A. (1998). HTA as a framework for task analysis. Ergonomics, 41 (11), 1537-1552.
- Sheridan, T.B. (1988). Task Allocation and Supervisory Control, In: M. Helander (Ed.). *Handbook of Human-Computer Interaction* (pp 159-173). Amsterdam: Elsevier Science Publishers B.V. (North Holland).

Autoren

Prof. Dr. M. Herczeg
Dipl.-Inf. M. Kammler
A. Roenspieß, M.Sc.
Universität zu Lübeck
Institut für Multimediale und Interaktive Systeme

Kontakt: herczeg@imis.uni-luebeck.de