Interfacekonzept für einen „Role-Based Client“ als Anwendung im gesamten Produktlebenszyklus

Ralph Tille
Hochschule der Medien
Wolframstr. 32
70191 Stuttgart
tille@hdm-stuttgart.de

Michael Burmester
Hochschule der Medien
Wolframstr. 32
70191 Stuttgart
burmester@hdm-stuttgart.de

Abstract
Für einen großen Automobilzulieferer soll das Software-Interface eines „Role-Based Client“ entwickelt werden. Dies soll maßgeschneidert für die verschiedenen Rollen und wesentlichen Aufgaben der Mitarbeiter entworfen und umgesetzt werden. Das Hauptziel eines rollenspezifischen Interfaces besteht darin, dass die Mitarbeiter ziel-, aufgaben- und effizienzorientierte Funktionsumfänge angeboten bekommen. Die dazugehörige Fragestellung war, wie die tatsächliche Rolle der Mitarbeiter im Unternehmen aussieht, da sich Themen und Aufgaben je nach Rolle und Nutzergruppe unterscheiden. Zunächst gilt es, die alltägliche Nutzung und den daraus resultierenden Funktionsumfang festzustellen, um danach ein nutzerorientiertes und aufgabenspezifisches Interfacekonzept zu erstellen. Für ein Role-Based Interface stand die Frage im Vordergrund, wie dies beschaffen sein muss. Um die Aufgabenstrukturen detailliert und genauer zu verstehen, wurde ein methodischer, teilweise partizipativer Ansatz gewählt, der es erlaubt, einen übergreifenden Blick auf den Ablauf eines vollständigen Projektes und den Alltag der Mitarbeiter zu werfen. Diese Methodenkombination hat sich sehr gut bewährt um einen Makro- und Mikroblick auf die Projektarbeit zu werfen und daraus entsprechende Anforderungen an die Konzeptentwicklung abzuleiten. Die Erkenntnisse aus der Nutzerstudie lieferten die Basis für die Modellierung einer Persona und einem Anwendungsszenario sowie einem Szenario-Mockup für ein maßgeschneidertes und konfigurierbares Interface.

Keywords:
/// Interfacekonzeption
/// Role-Based Interface
/// Produktentwicklung
/// PLM

1. Ausgangssituation

Für einen großen Automobilzulieferer sollte ein für verschiedene Rollen und wesentliche Aufgaben maßgeschneidertes Software-Interface – einen „Role-Based Client“ – entwickelt werden. Die Mitarbeiter nutzen innerhalb der Software-Umgebung im Unternehmen verschiedene Anwendungen für spezifische Aufgaben innerhalb der technisch-kaufmännischen Entwicklungsprozesse im Automobilsektor. Das Spektrum der zu entwickelnden Anwendung erstreckt sich über den gesamten Produktlebenszyklus: von Akquisition und Pflichtenhefterstellung, Konzeptions- und Mustererstellungsphase sowie nachfolgenden Produkt- und Prozessentwicklungsphasen bis zur Überführung in Serienprozesse. Aufgrund umfangreicher und heterogener Aufgabenbereiche wurden als erste Zielgruppe die technischen Teamleiter in der Entwicklung ausgewählt. Der Schwerpunkt liegt auf Produktplanung, der Produkt- und Prozessentwicklung sowie der Fertigungsvorbereitung. [Abb. 1]

Die technisch-kaufmännischen Teamleiter verantworten im Unternehmen umfangreiche Schlüsselfunktionen bis hin zum Kunden. Sie koordinieren Mitarbeiter, Prozesse und Ressourcen im Unternehmen

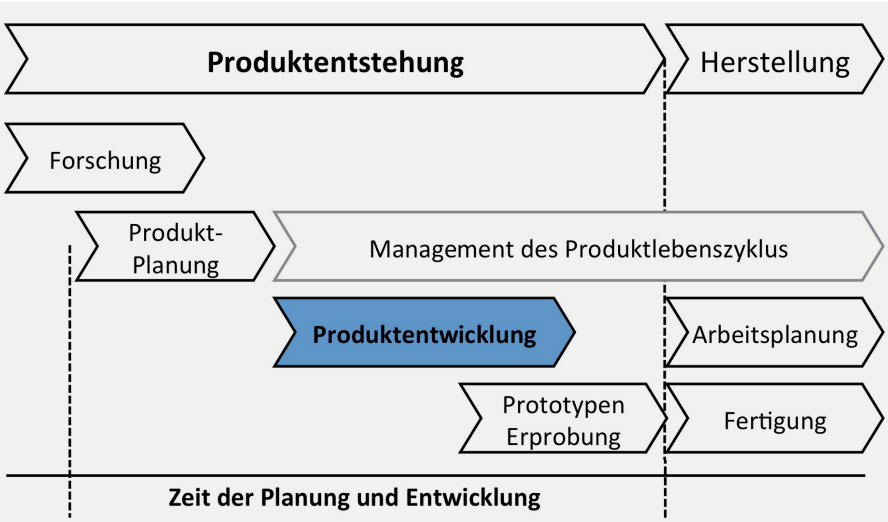

Abb. 1.
Produktentstehungsprozess nach Westkämper (2006)

sowie bei Lieferanten und weltweit verteilten Produktionswerken. Eine erste Bestandsaufnahme im Unternehmen zeigt, dass die Bandbreite der Software-Anwendungen ebenfalls sehr heterogen und umfangreich ist. Es werden bspw. ein E-Mail-Client mit Kalenderfunktion eingesetzt, übliche Office-Softwarepakete, diverse ToDo-Listen für die Teams auf der Basis unterschiedlicher Softwareprodukte, spezielle Anwendungen für die Abwicklung von Prüfaufträgen, Applikationen für interne und externe Bestellung und Beauftragungen, sowie Anwendungen für das Produktdatenmanagement im SAP-Umfeld und viele weitere spezialisierte Anwendungen. Zudem gibt es Zugänge zu weiteren Anwendungen im Intranet und zu kollaborativen Dateiplattformen. Im Rahmen der Softwareerneuerung des Produktdatenmanagements soll das Role-Based Client-Interface den Zugang zu Software-Anwendungen erleichtern und eine einfache, effektive und effiziente digitale Arbeitsumgebung bieten.

2. Referenzen und Vorüberlegungen

Typische Probleme bei der Nutzung von Anwendungen für das Produktmanagement sind fundamental. Gundelsweiler&Reiterer (2008) identifizieren typische Usability-Fehler: Navigationsprobleme, Probleme bei Auswahl und Informationssuche sowie Fehler bei der Darstellung oder Verzögerungen beim Arbeiten mit Anwendungen. Während der Recherche wurden typische Anwendungen mit rollenbasiertem Interface wie bspw. Microsoft Dynamics NAV (2013) im Bereich des Customer Relation Managements (CRM) oder ABAS ERP (2013) im Bereich der Enterprise Ressource Planning (ERP) Software genauer betrachtet. Dabei ist wesentlich, dass die Systeme eine höchst umfangreiche und flexible Datenbank bzw. Datenbasis als Grundlage haben, um unterschiedlichen Nutzern für die vielfältigsten Anwendungsfälle die gewünschten Informationen liefern zu können. Problematisch sind daher eingeschränkte oder nicht vorhandene Schnittstellenzugänge sowie inhomogene Datenbasen. Dadurch dass unterschiedliche Nutzergruppen auf dieselben Daten zugreifen können, ist eine freie Konfigurierbarkeit des Interfaces naheliegend und ermöglicht Flexibilität. Diese Parameter sind jedoch auch Faktoren für die Komplexität eines Interfaces und potentielle Fehlerquellen zugleich. Die Fokussierung und Auswahl auf wenige, wichtige Funktionen ist notwendig. Das Zusammenlegen von thematischen und aufgabenspezifischen Funktionen bietet eine weitere Reduktion der Komplexität, ohne dass auf Funktionen verzichtet werden muss. Die Anpassbarkeit und Flexibilität der o.g. Systeme ist so weitreichend, dass dies für bestimmte Nutzergruppen auch ein Hindernis bedeuten kann. Banovic et al. (2012) berichten mit Marathe&Sundar (2011) auch davon, dass die Konfigurationsfreudigkeit vom Leistungsniveau der Nutzer abhängen kann. Eine umfangreiche Liste förderlicher und hinderlicher Faktoren von Konfigurierbarkeit findet man nach wie vor bei Mackay (1991). Für Konfigurierbarkeit spricht, dass man in unterschiedlichen Projektphasen arbeitet, bestimmte Abläufe jedoch individuell abweichen und daher Nutzer selbst Einstellungen vornehmen müssen. Insofern kann Konfigurierbarkeit zwar hilfreich sein, aber eine Maßanfertigung für den Nutzer verspricht viele Fehlerquellen zu vermeiden. Vorab angepasste Interfaces haben Vorteile, für gleich ablaufende Tätigkeiten. Somit ist ein klares Verständnis der Rollen und Aufgaben der Nutzer notwendig (Findlater et al., 2008), was in der nachfolgend beschrieben Studie beschrieben wird.

3. Ziele, Fragestellungen und Vorgehen

Das Hauptziel dieses Projektes ist die Entwicklung eines rollenspezifischen Interfaces, um administrative Aufgaben im Produktentwicklungsprozess zu erleichtern und zu beschleunigen. Das Interface soll daher ziel-, aufgaben- und effizienzorientiert sein, d.h. es soll den Nutzern für spezifische Aufgaben die Informationen direkt, schnell und ohne Umwege bereitstellen. Die Aufgaben und Rollen betreffen laut den Angaben des Unternehmens einen großen Teil des Produktlebenszyklus. Die Herausforderung bestand darin, dass die Tätigkeiten im Tagesverlauf und über den gesamten Projektablauf unterstützt werden sollen. Dabei ist es durchaus üblich von Projekt zu Projekt zu wechseln und situationsabhängig und zeitlich sehr fragmentiert zwischen unterschiedlichen Aufgaben und Tätigkeiten zu springen (z.B. Besprechungen mit Teammitgliedern, To-Do-Listen pflegen, etc.). Ineffizienzen in Detailinteraktionen (z.B, unnötige Mehrfacheingaben in unterschiedlichen Anwendungen) sollen vermieden werden. Zudem stand fest, dass die bisherigen Software-Anwendungen sehr unterschiedliche Usability-Qualitäten aufwiesen und teilweise nicht aufeinander abgestimmt waren. Die umfangreichen Tätigkeiten der Primärzielgruppe greifen in viele Unternehmensbereiche ein, was für einen Abgleich und die Übertragung der Erkenntnisse auf die Prozessmodelle anderer Nutzergruppen spricht. Basis ist ein für das gesamte Unternehmen gültiger, formalisierter Produktentstehungsprozess, der i.d.R. aus den Komponenten Prozesse, Phasen, Meilensteine, Aufgaben, Methoden und Reifegrade besteht (vgl. Ristic, 2011). Allerdings gibt es Unterschiede zwischen idealem Prozessmodell und „gelebten" Abläufen. Die zentrale Fragestellung für diesen Teil war, aus welchen Komponenten sich die Rolle des Teamleiters zusammensetzt.

Im ersten Schritt wurde mit den Teilnehmern deren mentales Modell der Unternehmensprozesse anhand eines oder mehrerer Projekte besprochen und visualisiert. Diese Informationen sollten den Interviewern Detailkenntnisse zu gelebten Prozessen und verwendeten Fachbegriffen liefern, um im nächsten Schritt gezielte Fragen zu konkreten Abläufen der täglichen Projektarbeit und zum Arbeitsplatz stellen zu können. Daraus lassen sich Funktionsumfang, Probleme, Wünsche und Bedürfnisse der Nutzer feststellen sowie etwaige Abweichungen zum formalen Prozessmodell identifizieren. Die anschließende Analyse der erhobenen Daten ergibt ein Anforderungsmodell für die Interface-Konzeptentwicklung und Prototypenerstellung und soll die Fragestellung beantworten, wie ein Role-Based Interface beschaffen sein soll, um die Rolle und deren Aufgaben zu unterstützen. Eine Persona soll als prototpisches

Abb. 2.
Momentaufnahme einer CARD-Session für die Prozess-Sicht.

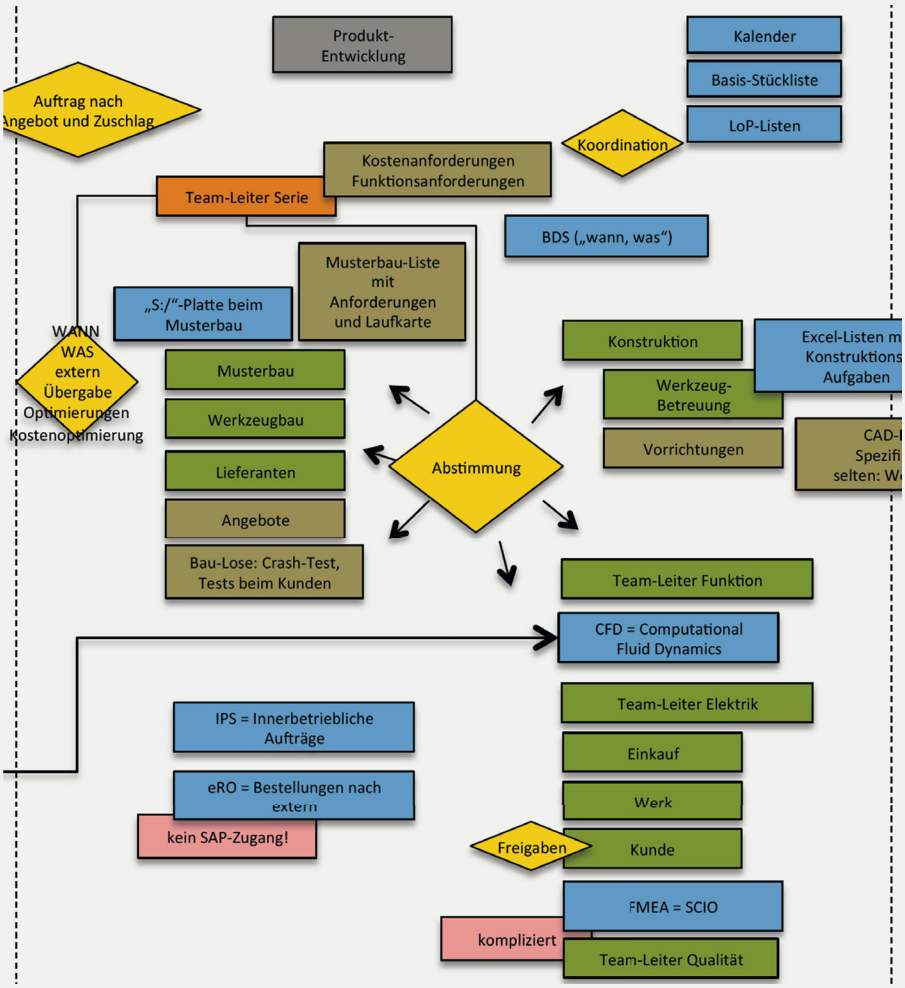

Abb. 3.
Ausschnitt eines visualisierten Prozessmodells der Nutzer

Entwicklungswerkzeug modelliert werden, welche die Eigenschaften, Wünsche und Bedürfnisse der Nutzer beinhaltet. Danach soll ein Aktivitäts-Szenario für typische Abläufe im Projektalltag entwickelt werden. Am Ende sollen dann Interface- und Interaktionskonzepte entworfen werden. Dieses Vorgehen entspricht dem szenariobasierten Gestaltungsansatz von Rosson&Carroll (2002).

In einem Anschlussprojekt werden die späteren Prototypen bzw. Mock-Ups empirisch mit denselben Nutzern validiert (Tille et al., 2013), um das Interface weiter zu detaillieren und ein Gestaltungskonzept zu erstellen. Die daraus gewonnenen Erkenntnisse fließen als Anforderungen in ein Lastenheft für die Software-Entwicklung.

Das Vorgehen in der Studie lässt sich wie folgt darstellen:
- Anforderungs-Erhebung aus Nutzer-Perspektive
- Sammlung von Hinweisen direkt von den Mitarbeitern
- Verständnisaufbau der Arbeitsabläufe und Arbeitsweisen
- Herstellen der Prozesssicht („was“) und Workflow-Sicht („wie“)
- Meta-Analyse der Zusammenhänge für erste Konzepte
- Anforderungen an das Interface für einen Role-Based Client
- Persona- und Szenariomodellierung sowie Anwendungsszenario als Mock-Up

4. Methoden

Das Verständnis der Rollen und Aufgaben der entsprechenden Zielgruppe zu erlangen, ist wesentlich um passende Interfacekonzepte zu erstellen. Die vom Unternehmen definierte Rolle der Teamleiter lässt sich anhand folgender Aufgabenschwerpunkte im administrativen und technisch-kaufmännischen Bereich charakterisieren:
- Kontrolle von Aufwänden, Projektstand, Termine, Zeitbedarfe
- Steuerung von Anpassungen im technischen Bereich (Produktauslegung etc.)

TN	Index	Aussagen	Themen	Kategorien
TN2	15	(Aufträge rausschreiben) „Und für jede Änderung mach ichs dann halt wieder“	Abarbeiten von sich wiederholenden Aufgaben	Arbeitsablauf
TN4	24	„Und der Rest vom Tag ist jetzt zumindest mal mit Besprechungen noch nicht verplant. Da kann man dann ein bisschen eher was wegarbeiten, grad im Hinblick auf seine ganzen ToDo-Listen die man dann so hat“	Arbeitsweise	Arbeitsablauf
TN4	25	„Die wichtigen Punkte schreib ich mir in der Regel auch immer aufn Block und äh so machs ich zumindest und hak dann ab“	Arbeitsweise Arbeitswerkzeug	Arbeitsablauf
TN4	36	Probleme beim switchen von Projekten „Ne eigentlich nicht“ „Aber auch da ist eigentlich, in unserem Team gibt's nur 2 Arten von Kühlerbau da könnte man zumindest gedanklich relativ schnell umschalten“	Arbeitsweise Arbeitsfluss	Arbeitsablauf
TN5	19	„Ich habe auch am Arbeitsplatz zwei Bildschirme, wo ich...ich habe einen größeren separaten Bildschirm, auf dem ich arbeite und nutze eigentlich den Laptop, der daneben steht und habe da meine P-Platte auf, dass ich da auch Daten irgendwo ablegen kann. Das ist eigentlich so der Einsteig, dass ich auf dem Bildschirm mein LOTUS habe und auf dem kleinen Bildschirm die P-Platte“	Arbeitsweise	Arbeitsablauf
TN5	24	„generell ist es eigentlich vieles was wir doppelt und dreifach machen. Stichwort Terminpläne. Stichwort der PSF“ ... „vieles was da drin steht möchten andere wieder in anderen Präsentationen haben“	Arbeitsweise Zeitaufwand	Arbeitsablauf
TN6	6	„Ja, also so was wirklich Wichtiges würde ich mir dann notieren. Und äh, was weniger Wichtiges würde ich vielleicht erst mal gar nicht lesen“	Arbeitsweise Priorisierung	Arbeitsablauf

Abb. 4.
Ausschnitt aus der inhaltlichen Analyse der Interviews

– Überwachung der Kosten (Finanz-controlling)
– Dokumentation der Projektfortschritte
– Ansprechpartner für alle technischen Fragestellungen im jeweiligen Bereich

Diese Beschreibungen sind sehr generisch, da sie für eine große Anzahl von Mitarbeitern gelten sollen. Der gewählte methodische Ansatz erlaubt es, detaillierte Kenntnis der Aufgabenstrukturen und Abläufen der Teamleiter im Alltag zu erlangen und einen übergreifenden Blick auf ein vollständiges Projekt zu werfen. Dazu wurde eine angepasste Form der CARD-Methode (Tudor et al., 1993, Muller, 1993) eingesetzt. Die Methode hat stark partizipativen Charakter, da die Teilnehmer selbst ihre Sicht auf die Prozesse herstellen, bewerten oder verändern. Hier sollten die Teamleiter, die Aktivitäten, Abläufe und Verzweigungspunkte eines kürzlich bearbeiteten Projektes mit Hilfe von zu beschriftenden Karten auf einem Tisch visualisieren und im Detail erläutern. **[Abb. 2]**

Dies wurde mit 3 erfahrenen Teamleitern durchgeführt. Die Visualisierungen können vom Teilnehmer selbst korrigiert werden, da Reihenfolgen, Lücken oder wichtige Aspekte schnell und einfach dargestellt und verändert werden können. Für den detaillierten Blick in die alltägliche Arbeit in Projekten wurde eine Kombination aus Contextual Inquiry (Beyer&Holtzblatt, 1998) und der Methode „A Day in a Life“ (Gillen et al., 2007) bzw. „The Day Reconstruction Method“ (Kahneman et al., 2004) eingesetzt. Hier geht es darum, den Arbeitsplatz, die Arbeitseinflüsse, die tatsächlichen Bedingungen und Abläufe zu erfassen. Aus diesem Grund wurden weitere 3 Teamleiter in ihrer Arbeitsumgebung besucht. Sie erhielten die Aufgabe ihre aktuellen Arbeiten zu erläutern und den Arbeitstag des Besuches vor Ort von morgens bis abends genau zu beschreiben. Dabei nahm der Interviewer die Rolle des Lernenden im klassischen Master-Apprentice-Model des Contextual Inquiries ein. Der Interviewer verhält sich dabei als Beobachter, Zuhörer und Nachfragender. Die Teilnehmer erläutern die aktuelle, tatsächliche Arbeit und schildern zusätzlich möglichst präzise anhand ausgewählter Arbeitstage, wie die Arbeit verläuft. Durch die Konkretheit der eigenen Arbeitsumgebung reduzieren sich Abweichungen, Idealisierungen oder Verfälschungen. Diese Methodenkombination gab einen genauen Einblick in den Tagesablauf, beispielsweise den häufigen Wechsel von operativ-administrativer Arbeit am Arbeitsplatz sowie Besprechungen an anderen Orten. Um die Erkenntnisse der empirischen Studie in ein Interfacekonzept zu überführen, wurde der zuvor beschriebene, nutzerzentrierte Prozess, bestehend aus der Persona-Modellierung nach Cooper (1999, 2004, 2007) sowie der szenariobasierten Gestaltung nach Rosson&Carroll (2002) bzw. adaptiert nach Goodwin (2009) gewählt. Das Anwendungsszenario, basiert auf dem Aktivitätsszenario, wurde detailliert und als verständliches Informations- und Interaktionsszenario erweitert.

5. Auswertung und Ergebnisse zu Projekt- und Tagesablauf

Die 6 Interviewsitzungen wurden audiovisuell aufgezeichnet und anschließend transkribiert. Das gesamte Material wurde qualitativ ausgewertet, indem zunächst die Themen in den Äußerungen und Erläuterungen identifiziert und anschließend klassifiziert wurden. **[Abb. 3]** Zur Beantwortung der Frage nach dem Ablauf von Entwicklungsprojekten bei den Teamleitern lagen dann visualisierte Prozessmodelle **[Abb. 4]** des Workflows innerhalb kompletter Projekte vor.

Zur Klärung der täglichen Arbeitsabläufe wurden aus den Themen und den zugeordneten Aussagen, Vorgänge, Bedürfnisse sowie Probleme bei der täglichen Arbeit der Teamleiter extrahiert.

Die Aussagen wurden in 27 Themen (Tätigkeiten, Werkzeuge, Strukturkomponenten usw.) eingeordnet, welche wiederum zu fünf nachfolgend aufgelisteten, übergreifenden Kategorien aggregiert wurden. Diese Komponenten bilden das Skelett eines Rollenmodells:

1. Arbeitsablauf
2. Arbeitswerkzeuge
3. Datenablage
4. Kommunikation
5. Kontrolle

Auffällig geworden ist, dass bestimmte Werkzeuge und Aufgaben nur in bestimmten Projektphasen relevant sind und zudem nur für bestimmte Teamleiter von Bedeutung waren, was auf eine notwendige Anpassbarkeit auf Nutzerebene hindeutet. Des Weiteren wurde festgestellt, dass bestimmte Werkzeuge sehr wichtig sind und oft genutzt wurden, andere zwar wichtig sind aber selten genutzt wurden, und auch unwichtige und selten genutzte Werkzeuge genannt wurden. Für die Arbeitsstruktur der Teamleiter lassen sich folgende Erkenntnisse ableiten:

- Je nach Produktkategorie wird ein komplexes Projekt oder mehrere parallele Projekte in unterschiedlichen Prozessphasen bearbeitet
- Projektübergreifende wiederholende Verwaltungstätigkeiten
- Zentrale Rolle der Teamleiter im Projekt
- Hohes Fachwissen und hohe Verantwortung hinsichtlich Budget, Qualität, Termin, Kunde, Mitarbeiter
- Die Tages- und Kommunikationsstrukturen sind höchst dynamisch und teilweise unplanbar
- Rückmeldung zu komplexen Baugruppen und Prozessen müssen schnell erfolgen
- Häufiger Zugriff auf Statusinformationen
- Sehr heterogene, individuelle Datenstrukturen
- Ständige Priorisierung von Aufgaben und Prozessen

Daraus lassen sich erste Anforderungen an die Konzeption eines Role-Based-Interfaces ableiten:

- Individualisierbare Konfigurierbarkeit mit Vorkonfiguration (Templates) ermöglicht eine projekt- und aufgabenorientierte Einstellung ohne Nachteile der Konfigurierbarkeit
- Klare Übersicht bestimmter, individueller Informationen aus den spezifischen Anwendungen („Dashboard-Charakter")
- Schneller Wechsel zur gewünschten Anwendung unter Berücksichtigung der aktuell ausgewählten Datensicht (bspw. Projektnummer, Teilenummer)
- Applikations- und projektübergreifende Suchfunktion
- Zusammenführung der heterogenen Datenstrukturen („Parallelwelten" vermeiden)

Ob und welche Komponenten für andere Rollen übertragen werden können, kann – neben verallgemeinerbaren Anforderungen wie bspw. Übersichtlichkeit oder schnelle Programmwechsel – abschließend nur vermutet werden und sollte daher mit den modellierten Prozessen im Unternehmen abgeglichen und über weitere Studien überprüft werden. Aus den Interviews konnten wir entnehmen, dass die Zusammenarbeit im Unternehmen sehr stark verzahnt ist, insofern liegt es nahe, dass bspw. die Projektarbeitsstruktur (mehrere niederkomplexe Produkte vs. ein komplexes Produkt), dynamische Tagesstruktur, dynamische Kommunikationsstruktur sowie die heterogenen, individuellen Datenstrukturen auch bei anderen Mitarbeitern von Relevanz sind.

6. Konzeptentwicklung

Grundlage der Konzeption bildete ein Anwendungsszenario mit einer Persona namens Stefan Kessler. Während eines realistischen Tagesablaufes wurden Aufgaben eines typischen Teamleiters mit dem Behr Workspace beschrieben. Es wurden Konzeptideen entwickelt, als Scribbles bzw. Wireframes visualisiert **[Abb. 5]** sowie mit den Anforderungen aus den Interview-Analysen harmonisiert und wiederum in das Anwendungsszenario überführt. Die Grundkonzeption zeigt hier schon wesentliche Merkmale des Workplace-Ansatzes: individuelle Konfigurierbarkeit und Anordnung von aufgaben- und projektorientierten Werkzeugen sowie programmspezifische und individuell einstellbare Informationssichten aktuell relevanter Datensätze.

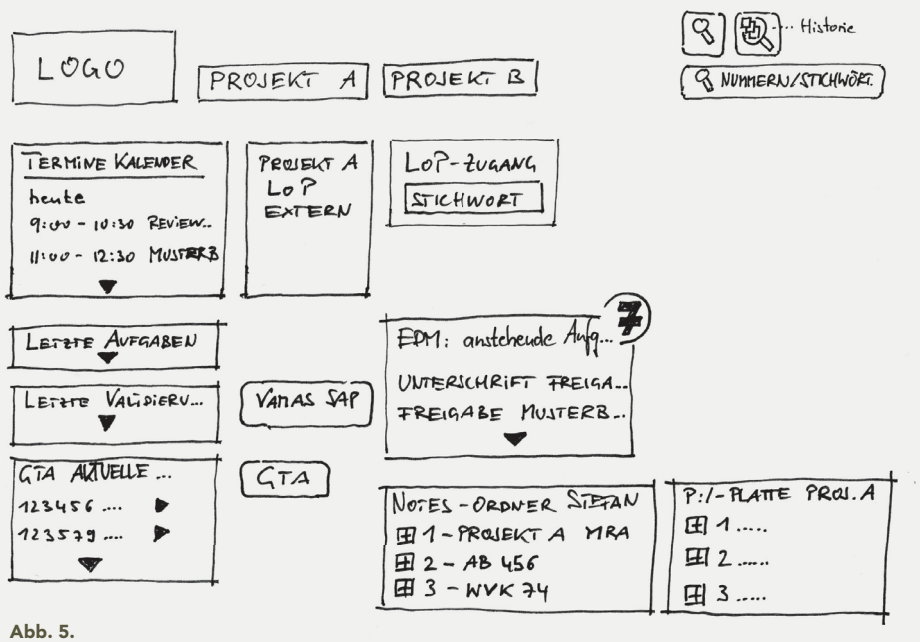

Abb. 5.
Scribble erster Konzeptideen

Dieses Szenario zeigt und beschreibt konkrete Aktivitäten der Persona entlang eines Tagesablaufs: Arbeitsbeginn, Starten der letzten Sitzung, Konfiguration bestimmter Elemente, Suche neuer Aufträge, Prüfen anstehender Aufgaben etc. Das Interfacekonzept wurde als Wireframe-Darstellung mit einer hochwertigen, realistischen, aber nicht zu final wirkenden Anmutung entworfen, die zu Anmerkungen und Änderungsvorschlägen animiert.

Der Arbeitsbereich oder Workspace **[Abb. 6]** ist eine Arbeitsfläche auf dem Bildschirm vergleichbar zu einem „Desktop-Dashboard" und bietet für die verschiedenen Programme eine aktuelle Vorschau, die es dem Nutzer ermöglicht, Kontrollinformationen in unterschiedlichen Detailgraden einzusehen, bevor der Absprung in das weiterführende Programm (bspw. SAP) erfolgt. Es können mehrere Workspaces angelegt werden. Die Vorschau ist in drei verschiedenen Größen verfügbar (klein, mittel, groß). Der Nutzer kann seine Workspaces frei konfigurieren und dazwischen hin und her wechseln. Diese Konzepte sollen es ermöglichen, den Anforderungen paralleler, phasenübergreifender Projektarbeit und einem schnellen Wechsel zur entsprechenden Anwendung gerecht zu werden und eine möglichst einfache, vorkonfigurierte und trotzdem individuelle Anpassbarkeit abzubilden.

Um einen neuen Workspace anzulegen, klickt der Nutzer auf den Bereich „Auswahl Spaces" und bekommt alle vorhandenen Workspaces angezeigt, so wie ein Feld mit der Aufschrift „+", über das ein neuer Workspace angelegt werden kann. **[Abb. 7]**

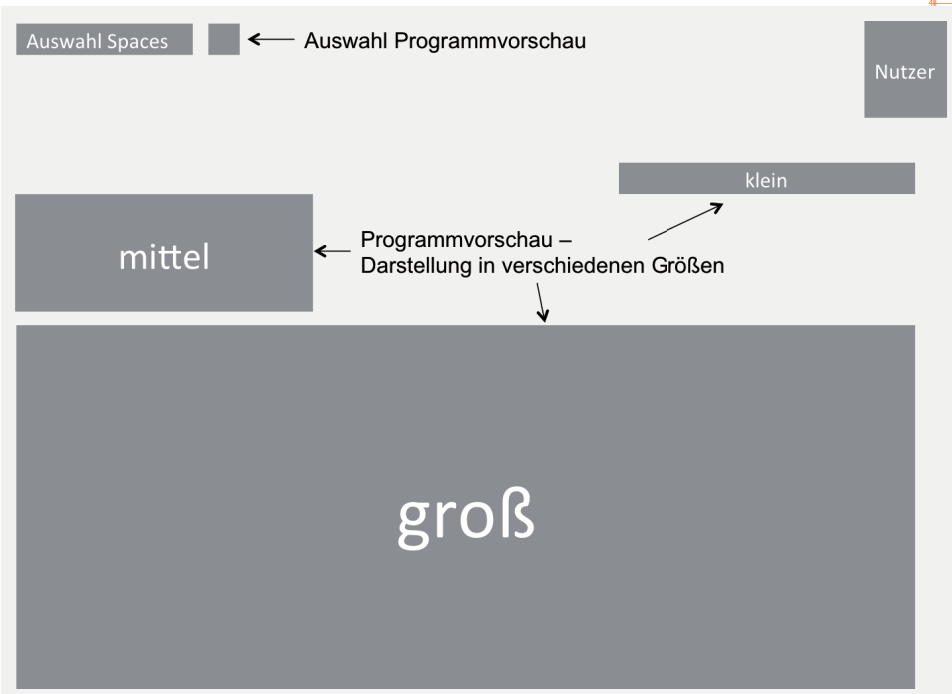

Abb. 6.
Aufbau eines Workspace

Den Nutzern können außerdem Mustervorlagen zur Verfügung gestellt werden, auf deren Basis sie ihren Workspace konfigurieren und wiederum als eigene Vorlage abspeichern können. Die Mustervorlagen könnten die in verschiedenen Phasen erforderlichen Programm-Informationen enthalten. Dadurch wird die Komplexität reduziert und auf wesentliche Aufgaben und deren Werkzeuge fokussiert. Über die Option „Auswahl Programme" bekommt der Nutzer alle Vorschaumöglichkeiten angezeigt, die er auf seinem Workspace platzieren kann. **[Abb. 8]** Der Nutzer kann frei wählen, welche Vorschau er sich wo anzeigen lässt und auf dem Workspace die Vorschaugröße ändern.

Ein Hauptmerkmal des Konzeptes ist die Programmvorschau **[Abb. 9]** mit den Stati der aktuellen Projektsituation für einen Überblick, ohne dass weitere Anwendungen gestartet werden müssen. Die dreistufige Darstellungsgröße bietet entsprechende Detailansichten. Vorkonfigurierte Informationselemente können innerhalb eines Rasters positioniert werden. Prinzipiell dürfen sich Informationselemente

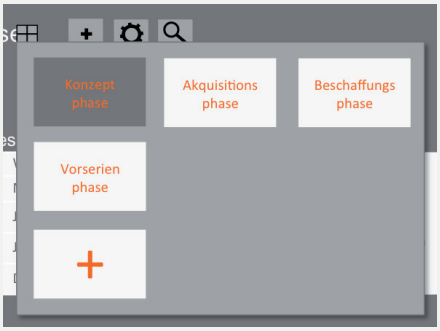

Abb. 7.
Auswahl Workspace

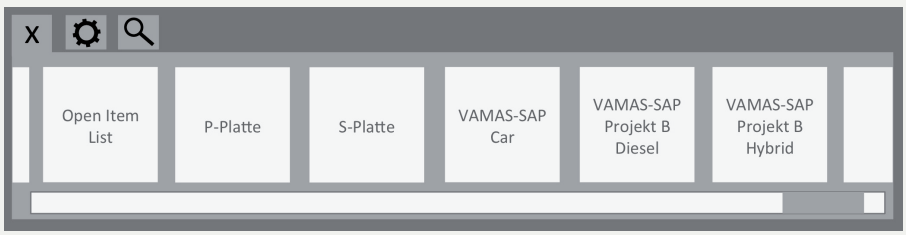

Abb. 8.
Auswahl der Programmvorschau

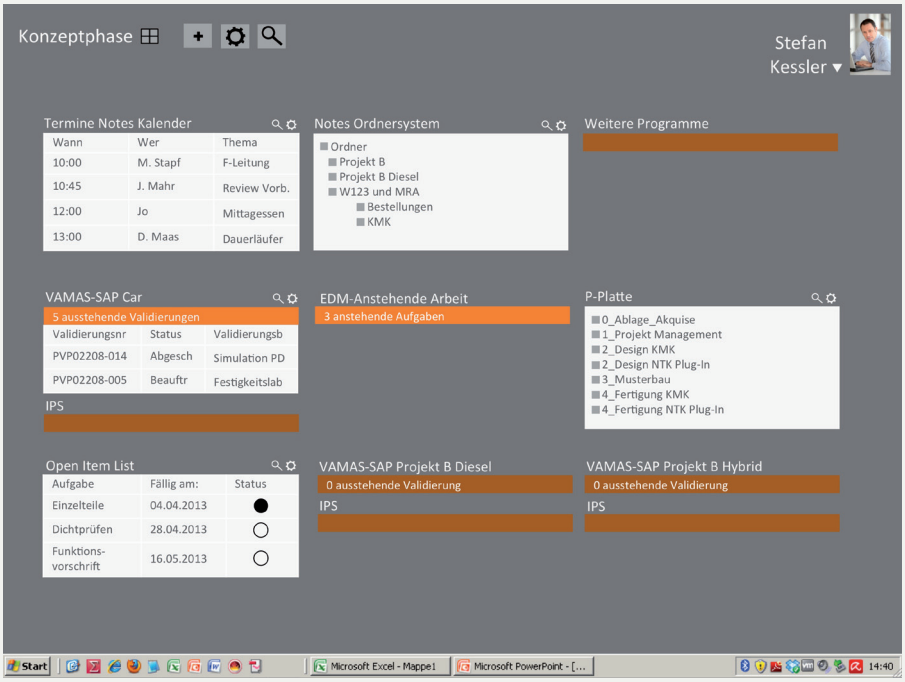

Abb. 9.
Basisansicht des Workspace

nicht überlappen, um dem Nutzer immer Zugriff auf sämtliche Informationskanäle zu liefern. Details zu den schmalen, einzeiligen Vorschau-Elementen (bspw. „3 anstehende Aufgaben") können über ein einfaches „Mouse-Over" bezogen werden.

Das Konzept der mehrstufigen Ansichten in Verbindung mit einer ständigen Sichtbarkeit aller Komponenten verspricht den Vorteil, dass der Überblick nicht verloren geht – eine zentrale Anforderung aus der Studie. Zwar ist die aktuell ausgewählte Informationseinheit **[Abb. 13]** im Fokus, doch auch andere Statusinformationen sind parallel für Entscheidungen verfügbar – das entspricht der Anforderung, Aufgaben und Prozesse ständig priorisieren zu können. Die Programmvorschau ist das „Herzstück" des Konzeptansatzes. Wesentlich ist dabei, dass die Nutzer aktuelle Informationen aus den von ihnen ausgewählten Datensätzen der Programme angezeigt bekommen, was eine ständige Kontrolle und Übersicht der wesentlichen Informationen ermöglicht und den Anforderungen nach Konfigurierbarkeit und individueller, aufgabenspezifischer Übersicht entspricht.

Der Absprung in das Programm erfolgt von jeder Größe aus über einen Doppelklick. Über der Vorschau stehen jeweils der Programmname und der Projektname, falls die Vorschau projektbezogen ist. Der „Landepunkt" innerhalb der Ziel-Anwendung sollte möglichst korrespondierend zur Vorschau-Ansicht sein, damit sich der Nutzer schnell in der jeweiligen Anwendung und den entsprechenden Kennzahlen wiederfindet. Die kleine Vorschau zeigt nur den Programmnamen und eine Statusmeldung. Über die Pfeilspitze kann in die mittlere Größe gewechselt werden. Der hell orangene Vorschaubalken zeigt an, dass Aufgaben oder Meldungen im Programm ausstehen. **[Abb. 10]** Gibt es keine Statusmeldungen aus dem Programm, ist der Balken im Prototyp in einem dunkleren orange gestaltet. **[Abb. 9]**

Abb. 10.
Vorschau EDM klein

Die mittlere Vorschau zeigt einen Ausschnitt aus dem Programm, wie hier am Beispiel VAMAS zu sehen ist. **[Abb. 11]** VAMAS ist eine Anwendung für das Management von Validierungsaufträgen im Rahmen von Qualitätsmaßnahmen der Produktentwicklung. Über die Pfeilspitzen kann in die kleine oder große Darstellung gewechselt werden.

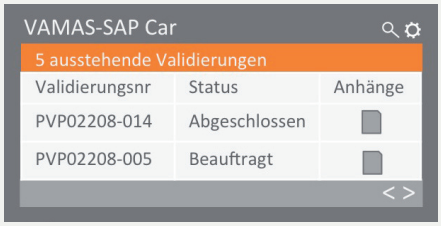

Abb. 11.
Vorschau VAMAS mittel

Über das Zahnrad kann die Ansicht konfiguriert werden Beispielsweise können in VAMAS Tabellenspalten für die Vorschau gewählt werden. **[Abb. 12]**

Die größte Darstellung eines Programmes zeigt in der Detailansicht die gesamte Vorschau. **[Abb. 13]** Der Nutzer kann hier scrollen, um alle Informationen einzusehen.

7. Erkenntnisse und nächste Schritte

Gerade die Methodenkombination hat sich sehr gut bewährt, um einen Makro- und Mikroblick auf Projektarbeit von Teamleitern zu werfen. Auf Basis dieser empirischen Erkenntnisse konnte eine Persona modelliert und ein umfangreiches Szenario entwickelt werden. Die Konzeptvorschläge basieren ebenfalls auf den nutzerzentrierten Befunden und ergaben ein stringentes und kompaktes Interface. Die nächsten Schritte sind die Validierung des aktuellen Konzeptes mit den Teamleitern sowie die Überprüfung, welche Aspekte auf andere Rollen und deren Anforderungen übertragen werden können, um danach in die Software-Entwicklung zu gehen und funktionsfähige Prototypen für Feldversuche zu entwickeln.

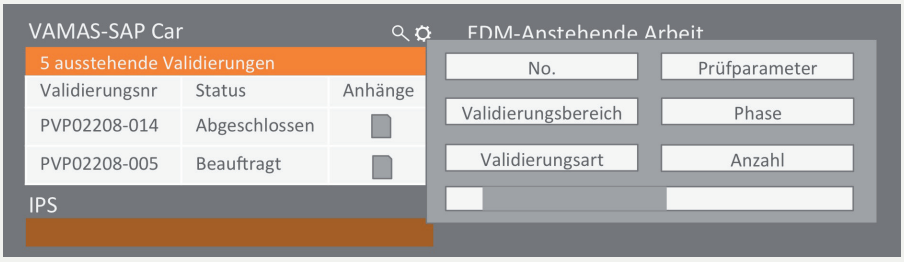

Abb. 12.
Spaltenauswahl in VAMAS

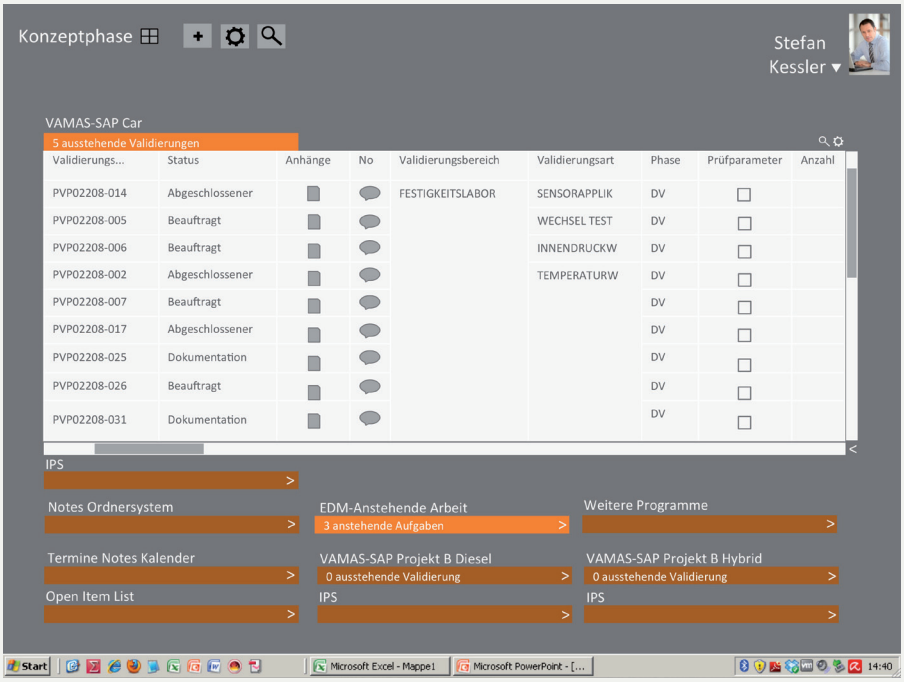

Abb. 13.
Große Vorschau

Literatur

1. ABAS ERP Software. (2013). Zugriff am 30.04.13 unter http://www.abas.de/de/download/abas_erp/produktbroschuere_2013.pdf
2. Banovic, N., Chevalier, F., Grossman, T., Fitzmaurice, G. (2012). Triggering Triggers and Burying Barriers to Customizing Software. In Proc. CHI'12.ACM, S. 2717–2726.
3. Cooper, A. (1999). The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (1st Edition).
4. Cooper, A. (2004). The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition).
5. Cooper, A. (2007). About Face 3: The Essentials of Interaction Design. Indianapolis: Wiley Publishing.
6. Findlater, L., McGrenere, J., Modjeska, D. (2008). Evaluation of a role-based approach for customizing a complex development environment. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08). ACM, New York, NY, USA, 1267–1270. DOI=10.1145/1357054.1357251.
7. Gillen, J., Cameron, C. A., Tapanya, S., Pinto, G., Hancock, R., Young, S., &Accorti Gamannossi, B. (2007). 'A Day in the Life': advancing a methodology for the cultural study of development and learning in early childhood. Early Child Development and Care, 177 (2), S. 207–218.
8. Goodwin K. (2009). Designing for the Digital Age: How to Create Human-Centered Products and Services. Indianapolis: Wiley Publishing.
9. Gundelsweiler, F.; Reiterer, H. (2008). Advanced User Interfaces for Product Management Systems. Proceedings of the 3rd IASTED International Conference on Human Computer Interaction (IASTED-HCI 08), Acta Press, Canada, p. 180–188, Jun 2008.
10. Kahneman, D., Krueger, A.B., Schkade, D.A., Schwarz, N. & Stone, A.A. (2004). A Survey Method for Characterizing Daily Life Experience: The Day Reconstruction Method. Science 3 December 2004: 306 (5702), 1776–1780. [DOI:10.1126/science.1103572]
11. Mackay, W.E. (1991). Triggers and barriers to customizing software. In Proc. CHI '91. ACM, S. 153–160.
12. Marathe, S. &Sundar, S.S. (2011). What drives customization?: control or identity?. In Proc. CHI '11. ACM, S. 781–790.
13. Microsoft Dynamics NAV. (2013). Zugriff am 30.04.13 unter http://www.microsoft.at/admp/70D3B4F4–4BE9–4BD0–844C-F5462E2004EB/NAV2009+Rollencenter+Uebersicht.pdf
14. Muller, M.J. (1992). Retrospective on a year of participatory design using the PICTIVE technique. In: Striking a Balance: Proceedings of CHI'92. Monterey CA: ACM, S.455–462.
15. Ristic, S. (2011). An Overview of the Approaches for A PLM Application's Customization. XV International Scientific Conference on Industrial Systems (IS'11).
16. Rosson, M.B. & Carroll (2002). J.M. Usability Engineering – Scenario-based development of human-computer interaction. San Francisco: Morgan Kaufmann.
17. Tille, R., Burmester, M., Schippert, K. (2013, akzeptiert). Role-Based-Client Workspace – Entwicklung von Dashboard-Interfaces im Product Lifecycle Management (PLM). Im Tagungsband der 10. Berliner Werkstatt Mensch-Maschine-Systeme, Oktober 2013.
18. Tudor, L.G., Muller, M.J., Dayton, T., Root, R.W. (1993). A participatory design technique for high-level task analysis, critique, and redesign: The CARD method. In: Proceedings of the Human Factors and Ergonomics Society 1993 Meeting, Seattle WA, Oktober 1993, S.295–299