

USEKIT¹ – Ein integrierter Ansatz zur nutzerzentrierten Software-Entwicklung für unternehmenskritische Anwendungen

Karl Tröger
PSIPENTA Software Systems GmbH

Torstrasse 49
10119 Berlin

Kurzfassung

Methoden des Software Engineering (SE) bieten softwareentwickelnden Unternehmen eine breite Palette praktisch erprobter und wissenschaftlich begründeter Methoden und Techniken zur Steigerung der Qualität von Entwicklungsprozessen und Systemen. Usability Engineering (UE) bietet weit weniger verbreitete Methoden zur Festlegung, Bewertung und Steigerung der Gebrauchstauglichkeit (engl. Usability) von Softwaresystemen. Wenngleich beide Engineering- Herangehensweisen für sich erprobt und in sich schlüssig sind, mangelt es an einer Integration beider Welten.

Unternehmen der Softwarebranche beschränken sich längst nicht mehr auf die reine Bereitstellung von Funktionalität und somit auf die Erhebung von funktionsrelevanten Anforderungen, allerdings finden eher technisch motivierte Qualitätsaspekte wie Zuverlässigkeits- oder Wartbarkeitsaspekte Beachtung. Anforderungen an die Usability sind eng verknüpft mit Kundenzufriedenheit und somit in einem dienstleistungsorientierten Markt im zunehmenden Maße entscheidend für den Erfolg der Produkte. Anwender beurteilen den Wert einer Software überwiegend an der Erfüllung ihrer Bedürfnisse. Doch nicht alleine die Zufriedenheit der Anwender ist die entscheidende Motivation für die Einführung von Usability- Methoden. Usability- Aspekte beeinflussen darüber hinaus in hohem Maße die

¹ Das Projekt USEKIT wird durch das Bundesministerium für Bildung und Forschung (BMBF) unter dem Kennzeichen 01|SC23 gefördert. Weitere Informationen erhalten Sie unter www.usekit.de. Das Projektkonsortium wird von der PSIPENTA Software Systems GmbH geleitet und setzt sich aus den Forschungsinstituten Fraunhofer IESE, Institut für Technologie und Arbeit und dem Unternehmen DaimlerChrysler AG zusammen.

Effizienz der Arbeitsabläufe, die mit der Software unterstützt werden. Auf diese Weise besteht ein direkter Zusammenhang zwischen Usability und dem Return on Investment (ROI) bei Herstellern und Anwendern.

Es existieren zahlreiche Beispiele für den Einfluss von Usability auf wirtschaftliche Kenngrößen. So wurde 1995 geschätzt, dass in der US-Industrie jährlich 30 Mrd. US-\$ Kosten durch mangelnde Produktivität der Software-Systeme entstehen bzw. entstanden sind. Dies wird im Wesentlichen auf fehlende Berücksichtigung von Usability-Engineering-Methoden im Entwicklungsprozess zurückgeführt [LAN05].

Der Zweck des Projekts USEKIT ist die Entwicklung und Erprobung eines nutzerzentrierten Vorgehensmodells. Dies soll Methoden des Software Engineering und des Usability Engineering (UE) integrieren und die spezifischen Randbedingungen von kleinen und mittelständischen Unternehmen (KMU) (Kosten, Kapazität, Marktanforderungen, Tätigkeitsgebiet) berücksichtigen. Der Schwerpunkt der Integration liegt dabei auf der Anforderungs- und der Testphase.

1. Einleitung und Vorstellung des Themenkomplexes

1.1 Allgemeine Situation

Seit langem bekannt und wieder in den Fokus von Wissenschaft und Technik gelangt ist die Lücke zwischen eingeführten Software-Engineering Methoden und Methoden zur Festlegung, Bewertung und Steigerung der Usability von Softwaresystemen (auch Reifegraderhöhungen). Für sich betrachtet existieren in beiden Welten Methoden und Hilfsmittel, die relativ gut eingeführt sind.

Bei Möglichkeiten und Lösungen zur Integration der bekannten UE-Methoden in die in der Industrie gelebten System- und Software-Entwicklungsprozesse steht die wissenschaftliche Welt erst am Anfang (Derzeit breitere Diskussion: „Bridging the Gap between Usability Engineering and Software-Engineering“).

Usability gewinnt auch in Deutschland zunehmend an Bedeutung [PR02]. Ein Trend, der in Amerika, bedingt durch die stärker Konsumgüter- orientierte Branchenausrichtung schon lange erkannt ist. Die Kaufmotivation der Vergangenheit - ein Unternehmen kauft ein Software-Produkt, weil es funktioniert - wird künftig nicht mehr ausreichen, weil die Produktivität nicht nur durch die software-technische Qualität, sondern vor allem durch die Nutzungs-Qualität beeinflusst wird (vgl. [ISO 9241], Qualitätskriterien Effektivität und Effizienz).

Grossunternehmen haben diesen Trend erkannt und beschäftigen zu diesem Zweck Usability-Experten, deren Aktivitäten parallel bzw. überlappend zur Software- Entwicklung und weitgehend unabhängig von dieser stattfinden. Da die Schnittstellen zwischen UE und SE nur unzureichend verstanden sind, gibt es kein standardisiertes Modell, das eine Verknüpfung beider Disziplinen beschreibt. Unternehmen müssen diese Verbindung immer wieder neu erarbeiten, potentielle Synergieeffekte bleiben häufig ungenutzt.

1.2 Gründe für die Thematisierung

Unternehmen der Softwarebranche stehen vor der Aufgabe, Anforderungen an zu entwickelnde Software aufzunehmen, zu verifizieren und laufend mit dem Entwicklungsprozess abzugleichen. Dies betrifft gleichermaßen Individualsoftware wie auch Standardsoftware. In diesem Zusammenhang sind Anforderungen an die

Gebrauchstauglichkeit (engl. Usability) der Software in zunehmenden Maße entscheidend für den Erfolg der Produkte.

- Usability ist eng verknüpft mit Kundenzufriedenheit. Die Zufriedenheit des Anwenders mit der entstandenen Lösung ist die Basis für eine weitere, für alle beteiligten Parteien gewinnbringende, Zusammenarbeit. Anwender beurteilen den Wert einer Software überwiegend an der Erfüllung ihrer Bedürfnisse [CGEY02].
- Des Weiteren hat Usability einen großen Einfluss auf die Effizienz der Arbeitsabläufe, die mit der Software unterstützt werden. Auf diese Weise besteht ein direkter Zusammenhang zwischen Usability und ROI [Mar02].

Doch obwohl die Bedeutung von Usability als wichtiges Qualitätsattribut erkannt ist, haben Unternehmen bei der Umsetzung Schwierigkeiten. Hierfür gibt es eine Vielzahl von Ursachen:

- Information über Anwender und Nutzungskontext, die zur Entwicklung gebrauchstauglicher Software notwendig ist, wird nicht systematisch erfasst oder nur unzureichend ins Entwicklungsteam kommuniziert. Mangelnde Benutzer-Integration ist eine der Hauptursachen für das Scheitern von Software-Entwicklungsprojekten [CHAOS99].
- Bekannte Software-Engineering Prozesse [BD95] ebenso wie Agile Prozesse [Coc02] lassen Usability als nichtfunktionale Anforderung außen vor. Dies führt unter anderem dazu, dass Usability als „Kontrollgröße“ im Entwicklungsprozess nicht berücksichtigt wird, und damit Zielabweichungen nicht rechtzeitig bemerkt werden können.
- Methoden aus dem Bereich Usability Engineering besitzen keine definierten Schnittstellen zum Softwareentwicklungsprozess [May99].

Die Verfügbarkeit einer standardisierten integrierten Methode zur nutzerzentrierten Entwicklung würde die Wettbewerbssituation der betroffenen Softwarehersteller deutlich verbessern, da sie eine systematische Berücksichtigung der Anwenderbedürfnisse im gesamten Softwareentwicklungszyklus vorsieht (Fokus auf Nutzerbedürfnisse, Senkung von Herstellungskosten durch fokussierte Entwicklung der Software). Die Erfüllung der Anwenderbedürfnisse erzeugt Zufriedenheit mit der geschaffenen Lösung und stärkt die Position des Herstellers beim konkreten Anwender und marktübergreifend.

Es existieren zahlreiche Beispiele für den Einfluss von Usability auf wirtschaftliche Kenngrößen. So wurde 1995 geschätzt, dass in der US-Industrie jährlich 30 Mrd. US-\$ Kosten durch mangelnde Produktivität der Software-Systeme entstehen bzw. entstanden sind. Dies wird im Wesentlichen auf fehlende Berücksichtigung von Usability-Engineering-Methoden im Entwicklungsprozess zurückgeführt [LAN95]. Frühzeitige Berücksichtigung von Usability Aspekten reduziert die Produkt- Entwicklungs- und Wartungskosten (Product Lifecycle Costs) [GIL98]. Etwa 80% der Kosten über den Produktlebenszyklus entstehen in der Wartungsphase durch die Korrektur unvorhergesehener oder falsch verstandener Anforderungen [NIEL93].

1.3 Abgrenzung des Problembereiches

Das Projekt USEKIT hat die Zielsetzung, insbesondere für KMU Methoden für eine nutzerzentrierte Softwareentwicklung zur Verfügung zu stellen, die auf folgende Weise Wettbewerbsvorteile für KMU schaffen:

- Der Entwicklungsprozess wird in seiner Effizienz wegen der Nutzerzentrierung gesteigert (Senkung der Herstellkosten).
- Die entwickelte Software zeichnet sich durch hohe Nutzerakzeptanz aus.
- Die Software unterstützt einen effizienten Arbeitsablauf und führt damit zu Wettbewerbsvorteilen beim Kunden und schnellstmöglichen Return on Investment.

Dies wird erreicht, indem **bereits vorhandene und beschriebene Methoden** des Usability Engineering (UE) und Software-Engineering (SE) verzahnt und entsprechend der KMU-Ressourcen und Anforderungen angepasst werden.

Zweck des Projekts ist die Entwicklung und Erprobung des nutzerzentrierten Vorgehensmodells USEKIT. Dies soll Methoden des Software-Engineering (Schwerpunkt: Anforderungsphase und Testen) und des Usability Engineering integrieren und die spezifischen Randbedingungen von KMU (Kosten, Kapazität, Marktanforderungen, Tätigkeitsgebiet) berücksichtigen. Der Schwerpunkt der Integration liegt dabei auf der Anforderungs- und der Testphase.

1.4 Themenbeschreibung

Ziel des Software Engineering ist es, in definierten und planbaren Schritten Software zu entwickeln, die zuverlässig auf realen Maschinen arbeiten [Bau72]. Dieses Verständnis von SE macht deutlich, dass die Entwicklung in festem Kosten- und Zeitrahmen, sowie die Zuverlässigkeit des Programms im Mittelpunkt der Methoden des SE stehen. Entwicklungen des Softwaremarktes zeigen allerdings, dass darüber hinaus heute Faktoren wie Kundenzufriedenheit und Gebrauchstauglichkeit über den Erfolg bzw. Misserfolg eines Produktes entscheiden. Software Engineering lässt dies bis heute meist außen vor.

Allerdings ist die unzureichende Beachtung von Usability Aspekten im Software Engineering erkannt, was in verschiedenen Ansätzen jüngster Entwicklungen, sowie durch Workshops auf internationalen Konferenzen sichtbar wird. Vielversprechend sind hier die agilen Prozesse. In XP versucht man durch den „Onsite-Customer“ [Bec99] die Perspektive des Endanwenders im Entwicklungsteam zu repräsentieren. Zur Steigerung der Gebrauchstauglichkeit und Kundenzufriedenheit ist dieser Schritt allerdings nicht ausreichend, da zum einen Kunde und Endanwender nicht die gleiche Personengruppe repräsentieren, zum anderen der „Onsite-Customer“ mit Fortschreiten des Projektes immer mehr die Entwicklersicht einnimmt und die Endanwendersicht verliert.

Andere Ansätze aus dem Gebiet der Objekt- Orientierung versuchen ebenfalls die Schnittstelle zwischen SE und UE zu schließen, in dem sie zum Beispiel geeignete Notationen oder Einzelmethoden zur Verfügung stellen. Genannt werden sollen hier beispielhaft die Essential Use Cases, die das User Interface Design unterstützen und ergänzend zu den traditionellen Objektorientierten Use Cases verwendet werden [CL99].

All diesen Methoden ist gemeinsam, dass sie nur einzelne Fragestellungen angehen. Sie bieten weder eine durchgängige Entwicklungsmethode an, noch sind sie industriell erprobt oder aufbereitet. Dies verhindert ihre praktische Anwendbarkeit für die Industrie im Allgemeinen und die KMU insbesondere und macht weiteren Entwicklungsbedarf notwendig.

Es wurden zahlreiche wissenschaftliche Ergebnisse zum Thema „Methoden und Vorgehensweisen zur Entwicklung von Benutzungsoberflächen“ publiziert. Die Gesamthematik wird vornehmlich unter Stichworten wie „Usability Engineering“ und „User / Usage Centered Design“ zusammengefasst.

Ziel dieser Methoden und Vorgehensweisen ist die ganzheitliche und durchgängige Betrachtung ergonomischer Aspekte während der Systementwicklung, so dass während der

Gesamtlaufzeit eines Projekts das Thema Usability ausreichend berücksichtigt wird. Insgesamt fällt auf, dass das Thema Usability Engineering oft noch als von der eigentlichen Systementwicklung losgelöste Methodik betrachtet wird, und so wird es in der industriellen Praxis auch oft gelebt. Ferner wird Usability Engineering zumeist aus der Sicht der Entwicklungsorganisation betrachtet. Tatsächlich lehren aber Erfahrungen aus der Praxis, dass wesentliche UE Aktivitäten in den frühen Phasen der Projektvorbereitung und Anforderungsanalyse auch stark durch die beauftragende Auftraggeberorganisation getrieben werden. Hier scheint eine Unterscheidung zwischen Aktivitäten auf Auftraggeber- und Auftragnehmer-/Entwicklerseite noch nicht hinreichend getroffen zu sein.

Die Bedeutung der Mensch-Maschine-Schnittstelle für die Gesamtqualität eines Systems wird oft noch stark unterbewertet. Den Hauptschaden tragen die Benutzer, die im täglichen Einsatz mit unkomfortablen und nicht ausreichend aufgabengerechten Dialogen und komplizierter Funktionalität arbeiten müssen.

Des Weiteren wird Usability Engineering oft noch als erheblicher Mehr- oder Zusatzaufwand der Systementwicklung betrachtet. Daher scheut das Management häufig bei ohnehin schon knappem Kostenrahmen davor zurück, Mitarbeiter in Richtung Usability Engineering gezielt auszubilden und für Projekte bereitzustellen.

Da die Schnittstellen zwischen UE und SE bisher nur unzureichend verstanden sind [May 99], gibt es bisher auch kein standardisiertes Modell, das eine Verknüpfung beider Disziplinen beschreibt. Unternehmen müssen diese Verbindung immer wieder neu erarbeiten, potentielle Synergieeffekte bleiben häufig ungenutzt.

Usability Engineering als einen integralen Bestandteil des Systementwicklungsprozesses und nicht als eigenen Teilprozess zu betrachten, das ist eine Aufgabe, die in der Zukunft zu lösen ist (Integration von Usability Engineering in Systems Engineering). Usability Engineering umfasst und berührt sehr viele Arbeiten, die in heutigen Systementwicklungsaktivitäten ohnehin bewerkstelligt werden müssen. Es geht darum, die Systementwicklung auch aus der Sicht der Ergonomie zu betrachten, wobei der Mensch ins Zentrum des Systems gestellt wird.

Ferner sollte die Seite der klassischen Auftraggeber aus Sicht des Usability Engineering stärker betrachtet werden. Denn aus der Erfahrung heraus liegt der Grund dafür, dass Usability Engineering in Entwicklungsprojekten nicht praktiziert wird, zu einem wesentlichen Teil auch daran, dass es Auftraggeber nicht als Leistung von den Anbietern eingefordert haben. Bietet eine Entwicklungsorganisation diese Methodik trotzdem mit an (was in der Regel zu gewissen Mehrkosten führt), ist sie oft nicht konkurrenzfähig gegenüber Anbietern, die ohne diese „Zusatzleistung“ anbieten. Von daher ist auch das Thema Usability Engineering verstärkt auf Auftraggeberseite in die entsprechenden Engineeringaktivitäten aufzunehmen, beispielsweise in die frühe Phase der Vertragsgestaltung und Anforderungsspezifikation sowie die späte Phase der Abnahme und Inbetriebnahme des Endsystems.

1.5 Ziele

Die Verbindung funktionaler mit nicht funktionalen Anforderungen schon im Designprozess zu unterstützen stellt eine der Herausforderungen der Zukunft dar. Usability wird meist ausschließlich mit Benutzeroberflächen (Mensch-Maschine-Interface, MMI) in Verbindung gebracht. Es existieren zahlreiche Leitfäden mit Gestaltungsrichtlinien für Oberflächen auf

Anwender- und Anbieterseite (Style-Guides). Große Anwenderorganisationen veröffentlichen Richtlinien zu Software-Ergonomie [VDMA04].

Die Benutzeroberfläche spielt zwar eine große Rolle bei der Bewertung der Nutzbarkeit eines Software-Systems, ist aber nicht allein für die Produktivität bei der Verwendung verantwortlich. Die notwendige Unterstützung der Anwender bei der Beherrschung der häufig hoch komplexen Funktionalität muss auch die implementierten Funktionen selbst in einem gegebenen MMI umfassen. Vielfach wird gerade in diesem Punkt den Anwendern deutlich zu viel zugemutet. Teilweise wird das Wissen der Entwickler benötigt, um eine Funktion auszuführen oder einen Prozess mit Hilfe von Software abzuwickeln. Unnötige Arbeitsschritte, um das Software-System zu bedienen (und nicht ausschließlich die Funktion zu nutzen), führen zu mangelnder Performance der Anwender. Vielfach ist dies auf die rein „informationstechnische“ Behandlung gegebener Aufgabenstellungen zurückzuführen. Nutzungskontext und Anforderungen an den Bediener (z.B. Qualifikation) werden selten berücksichtigt.

Die Motivation des Projektes USEKIT ist das Schließen der offensichtlichen Lücke zwischen SE- und UE-Methoden. Als grundsätzlicher Ansatz wird die Integration von UE-Methoden in den Software-Engineering-Prozess gewählt.

Ziel des Projekts ist die Entwicklung und Erprobung des nutzerzentrierten Vorgehensmodells USEKIT. Dies soll Methoden des Software Engineering (Schwerpunkt: Anforderungsphase und Testen) und des Usability Engineering integrieren und die spezifischen Randbedingungen von KMUs berücksichtigen. Der Schwerpunkt der Integration liegt dabei auf der Anforderungs- und der Testphase.

Die Ergebnisse sollen sich in einem für KMUs tauglichen und in bestehende Software-Engineering-Prozesse und -umgebungen integrierbaren Baukasten aus Methoden und Tools niederschlagen.

Technische und organisatorische Herausforderungen ergeben sich aus der notwendigen Integrierbarkeit der Prozesse und Tools des USEKIT Baukastens in die individuellen Entwicklungsprozesse einzelner Industrieunternehmen (hier beispielhaft die Anwendungspartner).

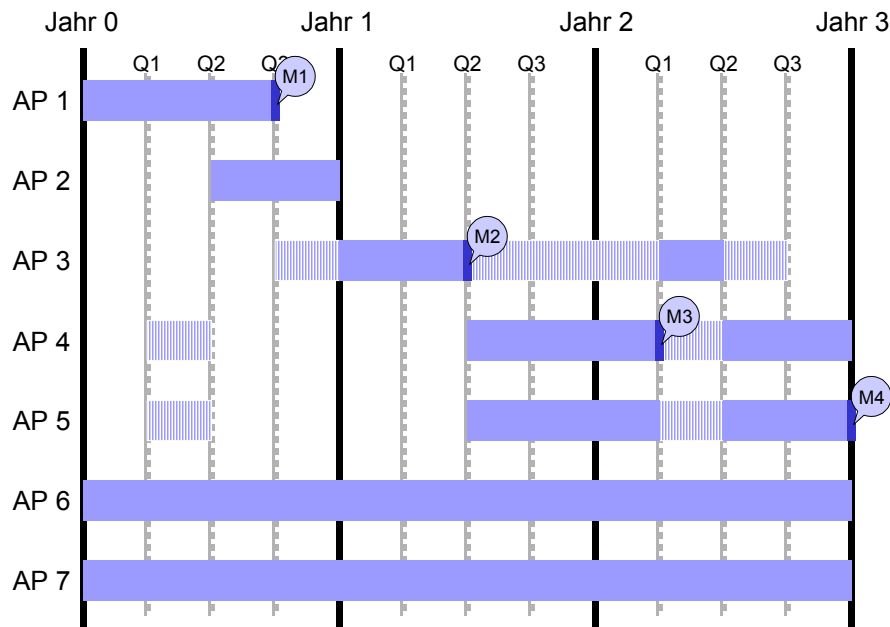
Im Projekt USEKIT werden Konzepte aus existierenden Ansätzen eingesetzt, die die Methoden-Provider einzeln schon erprobt haben. Die wissenschaftliche Herausforderung liegt in der Verzahnung der einzelnen Ansätze unter Ausnutzung von Synergien und deren quantitative Belegung. Die Zusammenstellung der Konsortialpartner mit ihren unterschiedlichen Kompetenzen sowie die Integration von Methodenentwicklung und Evaluation soll das Erreichen dieses Zieles sicherstellen.

Das Projekt USEKIT stellt aus heutiger Sicht einen wesentlichen Beitrag zur Entwicklung von Methoden für Software Systemen dar, die den heutigen divergenten Marktanforderungen gerecht werden müssen. Sowohl Software-Engineering Methoden alleine, als auch Usability Engineering Methoden für sich können dies derzeit nicht leisten, da jede Disziplin wichtige Aspekte außer Acht lässt.

Ein wesentlicher wissenschaftlicher Beitrag liegt auch in der Sicherstellung der Anwendbarkeit durch die Industriepartner, die sowohl durch eine systematische Erhebung der Anforderungen als auch durch iterative Evaluation sichergestellt wird. Das Projekt wird wichtigen Aufschluss über die Randbedingungen geben, die im empirischen Software-Engineering zu einer Methodenselektion führen können. Durch die Schaffung einer durchgängigen Fallstudie und die Sammlung und Auswertung von Return-on-Investment Daten, sollen für den deutschsprachigen Raum Anhaltspunkte zur quantitativen Beurteilung des Wertes einer solchen Methode gefunden werden. Dies liefert insbesondere im Bereich des empirischen Software-Engineering wertvolle Erkenntnisse.

2. Projektstatus USEKIT

2.1 Projektplan (Übersicht)



Das Gesamtvorhaben gliedert sich in sechs Arbeitsschwerpunkte und ein mitlaufendes Projektmanagement.

- AP1: Anforderungen an das nutzerzentrierte Vorgehensmodell
- AP2: Bewertung und Auswahl geeigneter SE und UE-Methoden
- AP3: Definition des USEKIT Baukastens
- AP4: Einführung des USEKIT Baukastens bei den Anwendungspartnern
- AP5: Inkrementelle empirische Evaluierung bei den Anwendungspartnern
- AP6: Aufbereitung und Verbreitung der Ergebnisse und Erfahrungen
- AP7: Projektmanagement

Zur Sicherstellung einer effektiven Projektkontrolle werden folgende Meilensteine definiert:

- M1: State of the practice Report , Ergebnis von AP 1
- M2: Baukasten, Ergebnis von AP 3:
- M3: Anwendungsbericht Ergebnis von AP 4
- M4: Abschlussbericht, Ergebnis von AP 5

2.2 Projektstatus

Das Projekt befindet sich heute (Juni 2004) in der Anfangsphase (AP1). Eine erste strukturierte Befragung von Unternehmen zu Projekt- entwicklung und -randbedingungen hat bereits stattgefunden.

Indifferente Anforderungen hinsichtlich „Benutzerfreundlichkeit“ werden von Auftraggebern gegenüber den Auftragnehmern formuliert. Eine Bewertung der Lösung hinsichtlich dieser geforderten Benutzerfreundlichkeit erfolgt nicht bzw. kann nicht durchgeführt werden, da keine Spezifikation vorliegt.

Die Verzahnung von UE und SE (dort wo der Begriff UE überhaupt eine Rolle spielt) findet über Meetings statt und Personen, die selbst (also persönlich) ein Interesse daran haben. Für die Unternehmen selbst ist das eher ein untergeordnetes bzw. oft kein strategisches Thema.

Die Sprache zwischen UE und SE ist nicht einheitlich, d.h. Begriffe im SE werden nicht mit UE in Verbindung gebracht und umgekehrt. So wird UE mit Interaktion via MMI (fast ausschließlich) gleichgesetzt. Funktionale Abläufe oder Gesamtprozesse als Objekt für UE scheinen keine Rolle zu spielen bzw. werden Abläufe und Interaktionen iterativ auf Anforderung des Kunden verbessert.

Das Wissen über UE liegt beim Kunden, nicht beim Hersteller von Software.

3. Ausblick

Es gibt derzeit ca. 19000 Unternehmen der Softwarebranche. Davon können ca. 10500 der Primärbranche und ca. 8500 der Sekundärbranche zugeordnet werden. Das Marktvolumen von Produkten, die direkt oder indirekt von der Softwareerstellung oder -anpassung abhängen, kann mit etwa 500 Mio. EUR beziffert werden. Dabei ist insbesondere die Primärbranche deutlich von kleinen Unternehmen geprägt [GfK00]. Der zunehmend internationale Wettbewerb in der Softwarebranche und der Markteintritt großer Softwareanbieter, wie der Eintritt Microsofts in den Markt für Business Solutions, bedroht die weniger kapitalstarken kleinen Unternehmen existenziell.

Besonders kleine und mittlere Unternehmen können nicht oder nur sehr eingeschränkt eigene Forschungsarbeit zum Thema Verknüpfung von UE und SE leisten. Daher nimmt sich dieses Projektkonsortium des Themas an. Die Resultate der Forschungsarbeit können grundsätzlich von allen Unternehmen genutzt werden, die direkt (Auftragnehmer) oder indirekt (Auftraggeber) mit dem Softwareentwicklungsprozess in seiner Gesamtheit zu tun haben. Sowohl Auftraggeber als auch Auftragnehmer können auf diese Weise ihre Position auf dem Markt verbessern.

- Auftragnehmer profitieren, in dem sie durch höhere Kundenzufriedenheit größere Marktanteile erlangen.
- Auftraggeber profitieren, in dem sie durch effizientere Unterstützung der Arbeitsprozesse zu einer höheren Produktivität gelangen.

Ziel der Forschungsarbeit ist es, das Verbesserungspotential durch den Einsatz von USEKIT mittels empirischer Daten zu untermauern. Neben der KMU-entsprechenden Aufbereitung der Ergebnisse würde dies einen wesentlichen Beitrag zur Verbreiterung der Ergebnisse leisten, da die Akzeptanz von Techniken in der Industrie durch quantitative Daten deutlich gesteigert wird.

Durch den hohen Eigenanteil, den die beteiligten Industriepartner in das Projekt mit einbringen, wird auch deutlich, dass ein großes Interesse vorhanden ist, die innerhalb von USEKIT entstandenen Projektergebnisse nach Projektabschluss fortzuführen.

4. Literaturverzeichnis

[Bau72] Bauer, F.L., "Software Engineering", Information Processing, 71, North Holland Publishing Co., Amsterdam, 1972

[BD95] Broehl, A.; Droeschel, W., Das V-Modell. Der Standard für die Softwareentwicklung mit Praxisleitfaden, Oldenbourg Verlag, 1995

[Bec99] Beck, K., *Extreme Programming Explained: Embrace Change*, Addison-Wesley, 1999.

[CGEY02] Cap Gemini Ernst & Young, ERP-Mittelstandsstudie 02/2002

[CHAOS99] CHAOS: A Recipe for Success, Standish Group, 1999

[CL99] Constantine, L.L. and Lockwood, L.A.D., Structure and Style in Use Cases for User Interface Design, in: Van Harmelen, M. (Hrsg.), Object-Modeling and User Interface Design, Addison-Wesley, 1999

[Coc02] Cockburn, A., *Agile Software Development*, Addison Wesley, 2002

[ISO 9241] EN ISO 9241: Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten. Teil 10: Grundsätze der Dialoggestaltung.

[GIL98] Gilb, T., "Principles of software engineering management" in "Usability is good business"

[GfK00] GfK Marktforschung GmbH; Fraunhofer-Institut für Experimentelles Software Engineering (IESE); Fraunhofer-Institut für Systemtechnik und Innovationsforschung (ISI): Analyse und Evaluation der Softwareentwicklung in Deutschland. Abschlussbericht einer Studie für das Bundesministerium für Bildung und Forschung, Dezember 2000. http://www.dlr.de/IT/IV/Studien/evasoft_abschlussbericht.pdf

[LAN95] Landauer, T. K., *The Trouble with computers: Usefulness, usability and productivity*, MIT Press

[May99] Mayhew, D., *The Usability Engineering Lifecycle*, Morgan Kaufmann Publishers, 1999

[Mar02] Marcus, A., Return on Investment for Usable UI Design, *User Experience*, Winter 2002

[NIEL93] Nielsen, J., *Usability Engineering*. Boston, San Diego, New York, 1993: AP Professional (Academic Press).

[PR02] Peissner, M.; Röse, K., Usability Engineering in Germany: Situation, Current Practice and Networking Strategies, European Usability Professionals Association Conference, Proceedings Volume 3, London, September 2002

[VDMA04] Leitfaden Software-Ergonomie, VDMA-Verlag 2004